Event-triggered Control and Filtering for Networked Systems Based on Network Dynamics

Submitted by
Yufeng Lin, Master of Science

A Thesis Submitted for the Degree of Doctor of Philosophy

Centre for Intelligent and Networked Systems
Central Queensland University

December 2014

Approved for Submission by Principal Supervisor

Laureate Professor Qing-Long Han

Declaration

I, Yufeng Lin, declare that this dissertation contains no material which has been previously submitted either in whole or in part for a degree at Central Queensland University or any other tertiary institution. To the best of my knowledge and belief, the material presented in this thesis is original except where due reference is made in the text.

Yufeng Lin

3 December 2014

Yufeng Lin

Abstract

With the development of computing and communication technology, networked systems in which information is exchanged through communication networks have become prevalent in a variety of practical applications. A considerable number of these systems have scarcely adequate computation and communication network resources. Notice that, network dynamics usually affect the quality of the performance of networked systems; network-based control or filtering design generally interferes with the quality of service of communication networks. Therefore, successful implementation of networked systems over communication networks requires the adequate integration of control or filtering with intelligent computation and communication networks. This research establishes an information scheduling middleware to efficiently utilise the limited communication resources and the computation capacity while preserving the desired performance of networked systems. In this proposed middleware, an event-triggered scheme is given to provide a tradeoff between the performance of networked systems and the utilisation of communication network resources. A scheduling mechanism is derived to avoid traffic congestion in the communication network by introducing a fluid-flow model. By using this information scheduling middleware, the issues of network-based control and filtering are investigated. First, a fluid-flow model of the communication network and an event-triggered scheme are integrated to develop a new framework, in which the criteria of stability and stabilisation are derived for networked systems under simultaneous consideration of control system performance and network dynamics. Second, based

on network dynamics, an online scheduling strategy is proposed to design the H_∞ filter for networked systems in the framework with the information scheduling middleware, the Information Dispatching Middleware (IDM). Third, the IDM is applied to distributed control for large-scale networked systems and a codesign method is obtained to determine the parameters of the IDM. Fourth, the mechanism of the IDM is analysed to investigate the cooperation between the Information Selection Module and the Congestion Avoidance Module in the IDM. Based on the IDM, distributed filters are designed considering the dynamics of communication networks. Finally, numerical and practical examples are given to demonstrate the effectiveness and advantages of the proposed information scheduling middleware.

Acknowledgements

I express my deep appreciation to my principal supervisor, Laureate Professor Qing-Long Han, for his enlightening guidance, invaluable discussions and insightful ideas throughout the past three years. What I have learned above all from being his student is profound respect his rigorous and diligent attitude to scientific research.

Further thanks go to Dr. Fuwen Yang and Dr. Yu-Long Wang, my associate supervisors, for their expert advice and genuine help with my research. I am pleased to thank Dr. Dennis Jarvis for his invaluable advice and sincere help not only for my research but also for my career. I also remain grateful to Dr. Xian-Ming Zhang who gave me professional guidance and technical advice on this research work.

My student life at Central Queensland University would have been more difficult if not for the friendships with the students and research staff in the Centre for Intelligent and Networked Systems.

Finally, I would like to express my deepest gratitude to my parents and parents-in-law, my wife, brother and sister for their love, encouragement, and support. My determination to complete this project owes a great deal to you, James Junyi Lin, my wonderful infant son.

Table of Contents

List of Tables

List of Figures

List of Symbols

$\mathbb{N}$	the set of positive integers
$\mathbb{Z}$	the set of non-negative integers
$\mathbb{R}$	the set of real numbers
$\mathbb{R}^+$	the set of positive real numbers
$\mathbb{R}^n$	the n-dimensional Euclidean space
$\mathbb{R}^{n\times m}$	the space of $n\times m$-dimensional real matrices
$\mathbb{E}\{x\}$	the mathematical expectation of the stochastic variable x
$\mathrm{Prob}\{\cdot\}$	the occurrence probability of the event "$\cdot$"
$\lambda_{\max}(A)$	the maximum eigenvalue of a symmetric matrix A
$\lambda_{\min}(A)$	the minimum eigenvalue of a symmetric matrix A
$\mathrm{diag}\{A_1, ..., A_n\}$	the block diagonal matrix
$\mathcal{L}_2[0, \infty)$	the space of square integrable functions on $[0, \infty)$
$(\Omega, \mathbb{F}, \mathbb{P})$	a complete probability space
$\mathrm{col}\{\cdots\}$	a block-column vector
$\mathbb{E}\{\cdot\}$	the mathematical expectation of the stochastic variable "$\cdot$"
A^T	the transpose of a matrix A
A^{-1}	the inverse of an invertible matrix A
$\|A\|$	induced 2-norm of a matrix A
$\|x\|$	Euclidean norm of a vector x
$A < B$	$A - B$ is negative definite for symmetric matrices A and B
I	an identity matrix
$\begin{bmatrix} A_{11} & A_{12} \\ \star & A_{22} \end{bmatrix}$	$\begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{bmatrix}$
$\begin{bmatrix} A_{11} & \star \\ A_{21} & A_{22} \end{bmatrix}$	$\begin{bmatrix} A_{11} & A_{21}^T \\ A_{21} & A_{22} \end{bmatrix}$
$\in$	belongs to
$\triangleq$	defined as
$\cup$	union

Chapter 1

Introduction

Networked systems are a class of systems controlled or estimated through some form of communication networks to ensure a certain quality of performance (QoP), as shown in Figure 1.1. Compared with traditional point-to-point wired systems, the

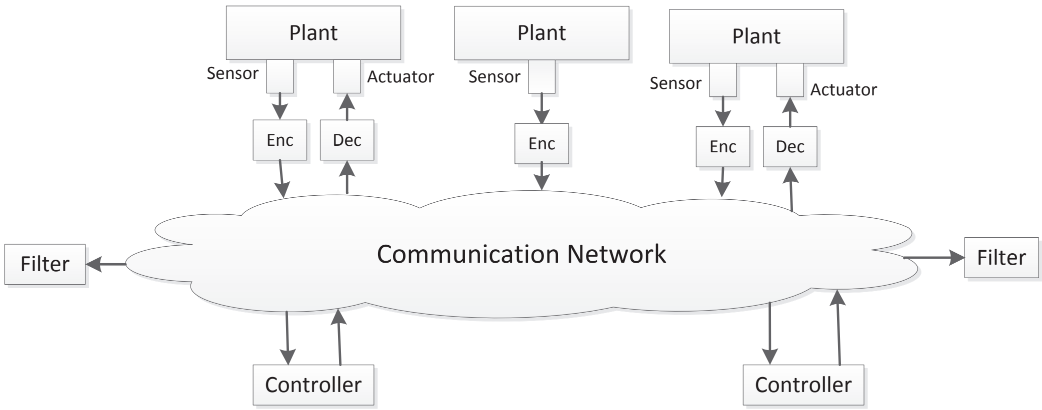

Figure 1.1: The architecture of networked systems

use of shared digital communication networks in networked systems provides several major benefits, such as low cost, ease of installation and maintenance, large flexibility and high reliability. Due to the advances in computation and communication technologies, networked systems are becoming prevalent in a wide range of applications. Examples include mobile robot vision systems, intelligent traffic control systems, smart power grids and automatic warehouse management systems. Since the mid-1980s, an amazing growth has occurred both in the theory of networked systems and in their prevalence. However, the employment of communication networks also raises several challenges that need to be addressed.

The most challenging aspect of investigating networked systems is that its study is an interdisciplinary research area, incorporating control systems, communication networks, information theory, and computing science [1]. Several studies in networked systems with emphasis on each of these research areas have been published. From the information theory point of view, the main task is analysis of protocols and communication sequences in order to improve their performance [2], [3], [4]. From the control systems point of view, the major issue is the modelling of networked systems and new control strategies that can deal with network induced constrains [5], [6]. In the last few years, increasing attention has been paid to intelligent computing technologies, where event-triggered communication schemes are adopted to schedule the control signals transmitted in the communication network [7], [8]. Overall, the literature about networked systems can be roughly divided into two main research areas: “control and filtering over networks” and “control of networks”.

1.1 Control and filtering of networked systems over communication networks

In this research area, the model of communication networks is usually presented in a simplified way, as a black box with a fixed network structure or pre-specified behaviour [9]. Networked control and filtering, the two main research issues of networked systems, are investigated to compensate for the communication network constraints such that some desired performance can be ensured, even under worst-case fluctuations of QoS (quality of service) of communication networks.

1.1.1 Network-based control

From the system control point of view, “control over networks” deals with the study of control strategies and control systems design in which data is exchanged through unreliable communication links. Due to communication constraints and limitations, the insertion of communication networks into the feedback control loop may exacerbate several issues, such as time delay, see e.g. [10], [11], [12], [13], [14], [15], data packet dropouts see, e.g. [16], [17], communication constraints, see e.g. [18], [19], [20], [21], [22] and quantisation, see e.g. [23], [24], [25], [26], [27], [28]. These issues may degrade system performance or even lead to system instability. On the other hand, due to the introduction of communication networks, the results using traditional control theory need to be reevaluated when investigating networked systems in non-ideal network environments. In general, several approaches are proposed in modelling, analysis and synthesis of networked systems, such as a *discrete-time model* approach [29], [30], a *continuous-time* approach [3], [19], [31] and a *sampled-data* approach [10], [20], [32], [33].

1.1.2 Network-based filtering

In the field of signal processing, state estimation has attracted considerable interest due to its wide ranging application in processing control, manufacturing, biomedical systems and so forth [34]. The well known Kalman filtering, one of the most sophisticated state estimation algorithms initially addressed in [35], has been extensively investigated in the research field for several decades. In some cases, external noise statistics are impossible to tabulate with complete accuracy. Hence, some researchers are resorting to the state estimation problem based on an H_∞ filtering approach, which was introduced by Elsayed and Grimble in [36]. The H_∞ filtering approach can provide a guaranteed noise attenuation level and is an effective method of solving the state estimation problem. The main advantage of H_∞ filtering is that no statistical assumption on the noise signal is required and that the noise sources can be arbitrary signals with bounded energy or bounded average power instead of being Gaussian.

Different from traditional H_∞ filtering schemes, network-based H_∞ filtering employs a shared communication network to communicate and exchange digital information between network nodes, such as sensors and filters. Thanks to the advantages of communication networks, this networked scenario has widely been used in the fields of embedded control systems, large manufacturing systems and automotive systems. However, due to communication constraints, networking the systems and filters inevitably introduces new challenging issues, such as time-varying delay, packet dropouts, information scheduling, inefficient utilisation of communication network resources, which need to be resolved under new frameworks for network-based H_∞ filtering issue of networked systems. In recent years, H_∞ filtering for networked systems has attracted increasing attention, see for example, H_∞ filtering for systems with a limited communication channel in [37], [38], network-based H_∞ filtering for continuous and discrete-time systems in [39], [40], respectively, H_∞ filtering for nonlinear networked systems in [41], [42] and the references cited therein.

From the above discussion, it should be pointed out that the fundamental difference between networked systems and traditional systems is the introduction of communication networks, which makes control systems more advanced while the analysis and synthesis of networked systems become more complicated.

1.1.3 Time-triggered schemes and event-trigged schemes

Over the past decade, the bulk of the work on networked systems has been concerned with analysis and synthesis of networked systems by using a sampled-data approach (see, e.g. [10], [43], [44] [45], [46], [47], [48], [49], [50]). These studies can be classified into two types of communication schemes: time-triggered schemes and event-triggered schemes.

In a time-triggered scheme, the time interval, i.e. between two consecutive transmissions, is usually constrained to be less than a fixed constant T, which is called *maximum allowable transmission interval* (MATI), see e.g. [3], [4], [43], [51]. T is generally set as small as technical implementation and communication network load permit in order to meet a desired performance. As pointed out in [51], such a time-triggered strategy, although easy to analyse and implement, represents a conservative solution that may unnecessarily overload the communication channel. Especially in distributed networked control systems, the MATI is often obtained in a centralized manner that is impractical for large-scale systems. As shown in [52], because the MATI is computed before the system is deployed, such a strategy must ensure different performance levels for all possible subsystem states. As a result, the time interval needs to be chosen short enough to assure a specified performance level for distributed networked systems. Consequently, larger bandwidth has to be allocated to ensure that the MATI is achieved. It is obvious that the time-triggered communication scheme leads to inefficient utilisation of the limited network resources and the wastage of energy resources. In this context, one would expect that the transmission intervals should not just satisfy a prefixed limit but rather be based on the current requirement of the system and the current dynamics of the network, the channel occupancy, and even the desired performance.

To overcome the drawbacks of the time-triggered communication scheme, some researchers, see, e.g. [49], [53], [54], [55], [56], [57], propose event-triggered schemes. One significant characteristic of event-triggered schemes is that the control or filtering tasks is executed only if a predefined event generated condition is violated. That is, event-triggered schemes only release sampled signals when “needed” [52]. In this case, “needed” means that the measurements of the system states or outputs satisfy a specified event generated condition. Compared with time-triggered schemes, an event-triggered scheme at least has the following advantages:

- It is executed in the same way that a human behaves as a manager [54];

- It trades available computation resources for precious communication network resources, such as network bandwidth and energy resources.

- Communication resources can be saved while the desired quality of performance can be ensured.

1.1.4 Event-triggered control and filtering

Due to the significant advantages of event-triggered schemes, event-triggered control for networked systems has become a hot research topic in recent years, see e.g. [7], [57], [58], [59], [60], [61], [62] and the references therein.

In [54], an implementation with an event-based proportional-integral-derivative controller was introduced by Årzén. The result showed that the system performance would not be degraded appreciably while the computation resources could be sufficiently reduced in comparison to a discrete-time implementation. Heemels *et al.* in 1999 [27] also did similar pioneering work showing the benefits of event-triggered control in their investigation of the synchronisation issues between two electrical motors. Analytical results of the event-based approach were first derived by Åström and Bernhardsson in 2002 [53], where an event-triggered approach was proposed as one of the aperiodic scheduling mechanisms to indicate that under certain circumstances such an approach gave better performance than the periodic sampling approach. Due to the benefits of event-based control, such as the reduction of communication bandwidth, computation resources and energy resources, better performance was achieved. Consequently, a variety of event-triggered conditions have been proposed in the implementation of networked systems to investigate control, filtering, synchronisation, and consensus issues.

For example, in [57], the problem of scheduling control tasks on embedded processors was studied. An event-triggered scheme was proposed to determine which data was executed at any given instant. Under this scheme, the performance of the control system was guaranteed and also traditional periodic execution requirements were relaxed. An execution model was proposed in [63], where the data was only sampled "when needed". Thus, a considerable amount of computational resources were saved. In [58], two implementations of event-triggered and self-triggered policies over sensor-actuator networks were introduced. In these two implementations, a feedback control law was designed such that the number of executions was reduced while the desired level of performance was guaranteed. In [52], a distributed event-triggering scheme for distributed networked control systems was proposed, where packet loss and transmission delays were considered. In this scheme, a subsystem broadcasts its state to its neighbors only when the subsystem's local state error exceeds a predefined threshold. Moreover, the maximal allowable number of successive data dropouts and the state-based deadlines for transmission delays were predicted. Decentralized event-triggered feedback schemes were proposed in [7] and [59] for linear and nonlinear networked control systems, respectively. An event-triggered scheme was given for perturbed linear systems in [55], where the control performance (practical stability) was analysed. Scheduling of event-triggered controllers over networks was considered in [60], in which different MAC protocols were compared in simulations.

The implementation of event-triggered schemes usually requires extra hardware to detect system states or outputs so that the next sampling time can be computed at the current sampling instants. When hardware is difficult to modify or update, alternative software schemes are required. These so-called self-triggered schemes are proposed in [50], [64], [65]. Under self-triggered schemes, an estimation of the next sampling interval is determined with online computation without any hardware.

In event-triggered control systems, the inter-event time between two consecutive events is an important issue, which is required to be larger than a certain positive constant; otherwise, the *Zeno* phenomenon may be exhibited [49], [66]. To avoid the occurrence of the undesired *Zeno* phenomenon, periodic event-triggered schemes were proposed in [67], [68], [69], [70]. The proposed schemes not only directly provide a guaranteed minimum inter-event time, at least longer than the sampling period $h > 0$, but also can easily be implemented by using embedded software architectures. An important event-triggered scheme — discrete event-triggered scheme —

is introduced by Yue *et al.* to investigate H_∞ control for linear systems in [67], in which both the controller gains and the parameters of the event-triggered scheme can be codesigned in terms of a set of linear matrix inequalities. Further results about the discrete event-triggered schemes have been derived by Zhang and Han in [68] and Peng and Han in [69]. The focus of these results is on the codesign issue to achieve a tradeoff between the desired quality of performance of the networked control systems and the utilisation of communication network resources.

In network-based filtering, along with event-based control, the event-triggered state estimation has received the same considerable attention and several important results have been addressed, see e.g. [71], [72], [73], [74], [75], [76], [77]. More specifically, in [71], event-based H_∞ filtering was studied for a class of networked systems with communication delays. In [72], a modified Kalman filter algorithm was presented by using a send-on-delta method based on an event-triggered sampling scheme. In [73], the state estimation problem was investigated by using the send-on-delta transmission method but without considering network-induced delays and data packet dropouts. In [74], a novel event-triggered estimator was proposed, where the measurements were updated when an event occurs rather than at each synchronous sampling instant. In [75], a particular scheme named the area-triggered method was proposed and a networked estimator problem based on a Kalman filter to estimate the states of the system was presented to deal with network-induced delays and data packet dropout. Based on an event-triggered sampling scheme, a modified fault isolation filter for a networked system with multiple faults was implemented in a novel form of the Kalman filter in [76]. An estimator problem for event-triggered sampling systems with packet dropout was studied in [77].

As stated above, the results of studies on event-triggered control and event-triggered filtering provide some techniques that are commonly referred to. It is also worth mentioning that some of the aforementioned results on event-triggered schemes, see e.g. [7], [57], [59], [78], are concerned with analysis issues and are based on a common assumption that the gains of the controllers or the filters must be known in advance, due to the difficulty of the design based on the criteria established by the studies. Moreover, the minimum inter-event time must be ensured in some forms of event-triggered schemes. However, in practice, the minimum inter-event time is difficult to obtain or may be shown not to exist. Furthermore, the above methods are usually specific to the control techniques and very few researchers have considered network dynamics, which enable the QoS of the supporting communication networks to be incorporated into the control or filtering loops.

Although the event-triggered scheme can save communication network resources, such kind of triggered scheme may also lead to inefficient utilisation. Triggered signals are of fundamental importance for networked systems. Their relationship poses two challenging issues:

- How to design an event-triggered scheme to respond to varying levels of QoS of a communication network and then determine rates of data transmission;
- How to schedule the communication network to maintain its QoS to serve the event-triggered scheme in order to achieve a tradeoff between the system performance and the QoS.

When both of these challenges are handled, the relationship between networked systems and communication networks can be established. Therefore, it is necessary to take network dynamics into consideration when the "control of network" is studied for networked systems.

1.2 Communication network dynamics

The rapid development of communication networks and computation technologies has provided several advantages and challenges for control system engineers to design and implement intelligent networked systems. How "control of networks" can dynamically serve networked systems within a unified framework becomes a new trend both in theory and in practice.

From the network dynamics point of view, "control of networks" is mainly concerned with providing a certain level of performance for a data flow to achieve efficient and fair utilisation of network resources. The research into "control of networks" spans several topics including congestion control, scheduling, routing and network protocols. Since a data transmission network is inherently a distributed system, in order to establish good values of the performance indicators, efficient means of resource sharing and traffic management should be implemented. Among the traffic regulation mechanisms, congestion control (or data flow control) plays a key role in ensuring coordinated access to the available resources.

1.2.1 Congestion control and fluid-flow models

Several techniques from operational research are adopted to study the design and control of modern communication networks such as optimisation, network programming and stochastic modelling. Stochastic modelling is the most widespread of these techniques because of the fact that this linearised model of the communication network, derived in [79], makes it possible to employ classical control system techniques to develop controllers subject to network parameters like load level, propagation delay, etc. The increasing number of applications in IP-based communication networks have made congestion control become a key issue to be addressed in meeting recent increasing demand for QoS in applications, such as networked real-time systems. Some of the recent accomplishments in the area of congestion control include the development of mathematical models for data flow control under various Internet protocols, see e.g. [80], [81], [82], [79], [83], [84]. Significant progress has been made in the theoretical understanding of network congestion control from an optimisation standpoint in a framework defined by Kelly in [85]. Here, the stability and fair allocation of a rate control algorithm for communication networks is analysed and the issue of how to share the available bandwidth within the network is addressed. The importance of an accurate fluid-flow model for the analysis and design of network dynamics has been illustrated in [86]. Since then, the effects of time delays and nonlinearities in the network fluid flow model have also been studied, see e.g. [87], [88], [89], [90], [91], [92], [93]. As an alternative to classical congestion control schemes developed using intuition and simple ad hoc control techniques, much effort has recently focussed on congestion control by using techniques such as optimal control [94], artificial neural networks [95], fuzzy systems [96] and nonlinear control [97], [98].

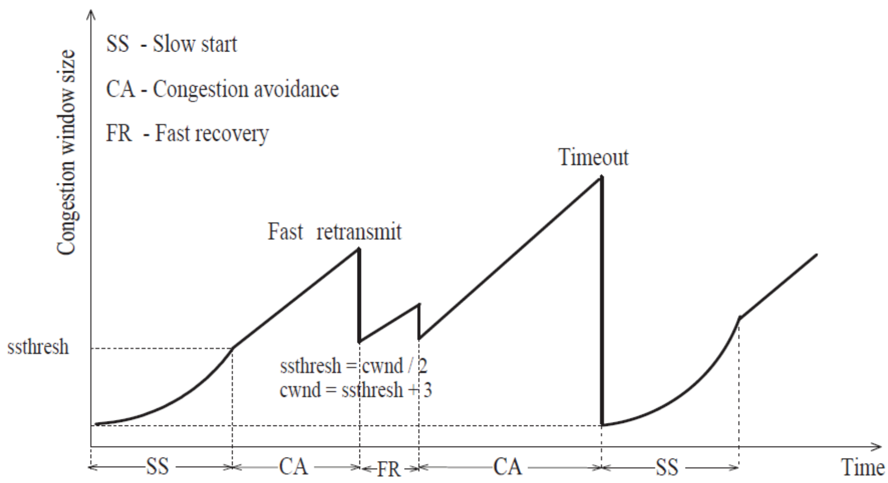

Figure 1.2: Evaluation of the window size in TCP Reno

Active Queue Management (AQM), an important scheduling policy, interacts with the TCP congestion control mechanism to manage the queue length. One of the most popular algorithms of AQM is Random Early Detection (RED), which was presented by Floyd and Jacobson in 1993 [99]. By using the RED strategy, the TCP congestion control mechanism involves four algorithms: slow start, congestion avoidance, fast retransmit, and fast recovery, which are in Figure 1.2, devised by Jacobson in [100].

To analyse the dynamic behaviour of the congestion control mechanism, we firstly introduce a discrete-time dynamic model of a TCP-RED congestion control system, as shown in Figure 1.3.

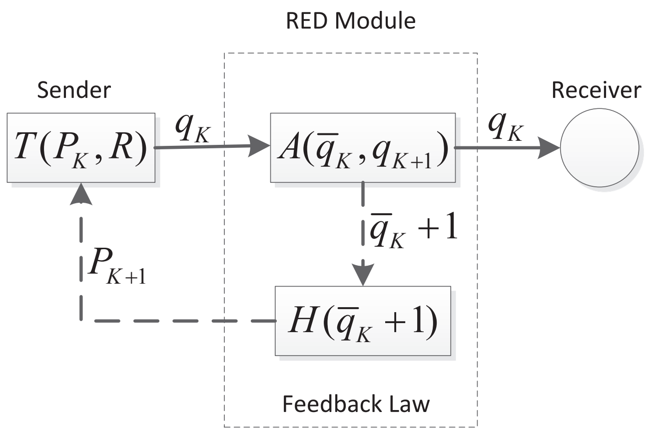

Figure 1.3: The TCP-RED feedback control system

In Figure 1.3, a complex throughput function $T(p_k, R)$ proposed in [101] is described as

$$T(p_k, R) = \frac{MK}{R\sqrt{p_k}} \tag{1.1}$$

where M is the packet size, $K \in [1, \sqrt{3/2}]$ is the constant parameter, R is the round trip time, p_k is the probability of packet loss at time t_k.

At time t_{k+1}, each sender adjusts its transmission rate based on the drop probability p_k to obtain a new queue length q_{k+1} as

$$q_{K+1} = \begin{cases} 0, & p_k \geq p_0 \\ B, & p_k \leq p_1 \\ (\dfrac{cKN}{(c-\lambda(1-p_k))\sqrt{p_k}} - \dfrac{R_0 c}{M}), & otherwise \end{cases} \tag{1.2}$$

where B is the buffer size, M is the packet size, c is the link capacity, N is the sessions, λ is the transmission rate of the UDP connection, R_0 is the round trip time, $p_0 = (NMK/R_0c)^2$, $p_1 = (NMK/(bM + R_0c))^2$.

The RED algorithm calculates the weighted moving average of the queue size. Let q_{k+1} be the current queue size and $w_q \in (0, 1)$ be the exponential averaging weight. The RED algorithm updates the average queue size as

$$\bar{q}_{k+1} = (1 - w_q)\bar{q}_k + w_q q_{k+1} \tag{1.3}$$

To manage the length of the queue by randomly dropping packets with a probability p_k, the obtained average queue size is determined by two parameters: minimum queue threshold $q_{\min}$ and maximum queue threshold $q_{\max}$. As shown in Figure 1.4, if the average queue size is smaller than $q_{\min}$, the packet is released to the queue; if the average queue size is between $q_{\min}$ and $q_{\max}$, the packet is dropped out with the probability p; if it exceeds $q_{\max}$, the packet is marked or discarded. The function for the average queue size can be then expressed as

$$p_{k+1} = \begin{cases} 0 & if\ \bar{q}_{k+1} \leq q_{\min} \\ \frac{\bar{q}_{k+1} - q_{\min}}{q_{\max} - q_{\min}} p_{\max} & if\ q_{\min} < \bar{q}_{k+1} < q_{\max} \\ 1 & if\ \bar{q}_{k+1} \geq q_{\max} \end{cases} \tag{1.4}$$

where $p_{\max}$ is the maximum packet drop probability.

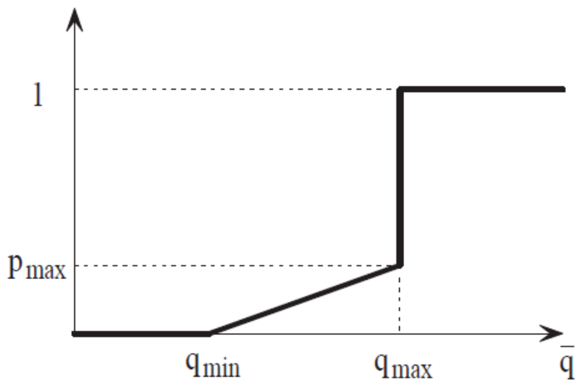

Figure 1.4: RED drop probability as the function of the average queue size

A nonlinear dynamic fluid-flow model was firstly proposed in [102] to examine the interactions of TCP flows with a single RED router as shown in (1.5). The dynamic behaviour model of the TCP congestion control mechanism was developed by using a fluid-flow and stochastic differential equation analysis approach. The window size $W(t)$ and the average queue length $q(t)$ are used as state variables.

$$\begin{cases} \dot{W}(t) = \dfrac{1}{R(t)} - \dfrac{W(t)}{2}\dfrac{W(t-R(t))}{2R(t-R(t))}p(t-R(t)) \\ \dot{q}(t) = \dfrac{N(t)}{R(t)}W(t) - C \end{cases} \tag{1.5}$$

where $W(t)$ is the expectation of the TCP window size; $q(t)$ is the expectation of the queue length; $R(t)$ is the round trip time; $N(t)$ is the load factor (number of TCP sessions); $p(t)$ is the probability of packet marked or dropped; C is the link capacity.

By linearising a fluid-based nonlinear TCP model, a linear fluid-flow model for TCP-RED congestion control mechanism was investigated in [79] and [103]. From a control theory point of view, the researchers performed the analysis of TCP interactions by using RED under equilibrium points. They presented effective methods by using control theory for choosing proper RED parameters which led to local stability of the dynamic system on the preset equilibrium points. The dynamic model linearised around the equilibrium points can be described as

$$\begin{cases} \delta\dot{W}(t) &= & -\frac{N}{R_0^2C}(\delta W(t) + \delta W(t-R_0)) \\ & & -\frac{1}{R_0^2C}(\delta q(t) - \delta q(t-R_0)) - \frac{R_0C^2}{2N^2}\delta p(t-C) \\ \delta\dot{q}(t) &=& \frac{N}{R_0}\delta W(t) - \frac{1}{R_0}\delta q(t) \end{cases} \tag{1.6}$$

where $\delta W = W - W_0$; $\delta q = q - q_0$; $\delta p = p - p_0$; $\{W_0,\ q_0,\ p_0\}$ is the set of operating points; W_0 is the expectation of TCP window size; q_0 is the expectation of queue length; R_0 is the round trip time; C is the link capacity; T_p is the propagation delay; N is the load factor (number of TCP sessions); and p is the probability of packet marked or dropped.

An accurate model of TCP with RED helps to understand and predict the dynamic behaviour of the network. In addition, the model helps to analyse the stability margins of the system and provides design methods for selecting network parameters. The design methods are important for the network designers to improve network robustness. Therefore, modelling TCP with RED is an essantial step to improve the service provided to control and filtering of networked systems and to efficiently utilise precious network resources. Motivated by this observation, this thesis will study the control and filtering of networked systems based on the fluid-flow model of network dynamics, which are based on the TCP congestion control mechanism with RED strategy, a widely deployed AQM policy.

As stated previously, although the advantages of networked systems are well understood and have been demonstrated through practical applications, to the best of the author's knowledge, there are no theoretical results in the literature for control and filtering of networked systems considering network dynamics. There are several problems that need to be addressed when network dynamics are involved in the analysis and synthesis of networked systems. More specifically, the following research problems need to be solved:

- How to reduce the complexity induced by a consideration of "control of networks" issues when investigating "control and filtering over networks" for networked systems;
- How to abstract the main research issues in both control systems and communication networks, and model them in a unified framework, which is indicated as a link between "control and filtering over networks" and "control of networks";
- How to schedule the information to efficiently utilise the communication network resources and computation capacity while preserving the required system performance;
- How to design online scheduling strategies for networked systems to adapt different QoS (quality of service) levels of communication network to achieve a better QoP (quality of performance) of networked systems while compensating network-induced time-varying delays;

- How to use the derived framework based on network dynamics to investigate control issues for a large-scale distributed system, and how to codesign the parameters of the controllers, the filters, the event-triggered scheme and the congestion strategies;

- How to analyse the dynamic interaction between networked systems and communication networks in the proposed unified framework to investigate the distributed event-triggered filtering for networked systems.

1.3 Significance of this research

From the practical point of view, networked systems have been deployed in a wide range of applications such as modern manufacturing, oil refinery, mineral processing, satellite and missile guidance, aircraft control and transport management and scheduling. It is well known that the insertion of communication networks into the loop between plants and controllers or filters enables remote execution of tasks, increases the flexibility of installation and reduces the cost of deployment. However, it also induces network-induced delays mainly due to the congestion of communication networks. These may have negative effects on the networked system and even cause system instability. In addition, it is noted that when network resources are limited, ignoring network dynamics may lead to degradation of the quality of service (QoS), especially when the communication network becomes congested, e.g., uni-processor, limited memory and restricted file operations. For "control of networks", proper communication protocols are required to maintain the network states such that the network quality of service (QoS) is guaranteed, and therefore advanced design methods are needed to guarantee the quality of performance (QoP) of networked systems.

Dynamic allocation of network resources according to the control systems' requirements will lead to more efficient resource utilisation. For "control over network", intelligent computation is incorporated into networked systems to schedule the sampled data. To integrate the communication networks, computation and networked systems, it is important to build a unified framework to assure both QoS of the communication networks and QoP of the networked systems simultaneously.

From the theoretical point of view, even though several studies have been dedicated to the research of networked systems, there are relatively few publications available that address both network dynamics and control system performance simultaneously, see for example, [9], [104], [105], [106], and the references cited therein. It is noted that for most of these results, there is an important assumption: either the QoS of the communication network is known in advance and controllers or filters are designed to obtain the desired performance for the networked system; or the performance level of the control system is known and the network is scheduled such that the desired performance of the networked system is guaranteed. It is obvious that these two assumptions separate the studies of the control systems and communication networks, which will lead to conservative results for the networked system. Therefore, a new framework would be necessary to integrate computation, communication networks and control systems. In this thesis, we will propose solutions to the problems mentioned at the end of Section 1.2.

To sum up, the study of the control and filtering for networked systems based on network dynamics is significant from both a practical and a theoretical point of view. Moreover, this research will not only enrich the theory of "control over networks" but also further develop the theory of "control of networks", and even more establish the codesign mechanism between them.

1.4 Contributions of this thesis

We aim at finding a link between "control over networks" and "control of networks". In this study, the main research issues are expressed by a novel framework with the construction of an Information Dispatching Middleware, whose concept is borrowed from the software engineering field. A middleware is a class of widely used application frameworks that hide a lot of low level details, and provide simpler as well as higher level usage models [107], [108]. Well designed usage models enable the components in the middleware to be reused in different applications.

In this thesis, a novel middleware framework is proposed to integrate computation and communication for real time information scheduling as shown in Figure 1.5.

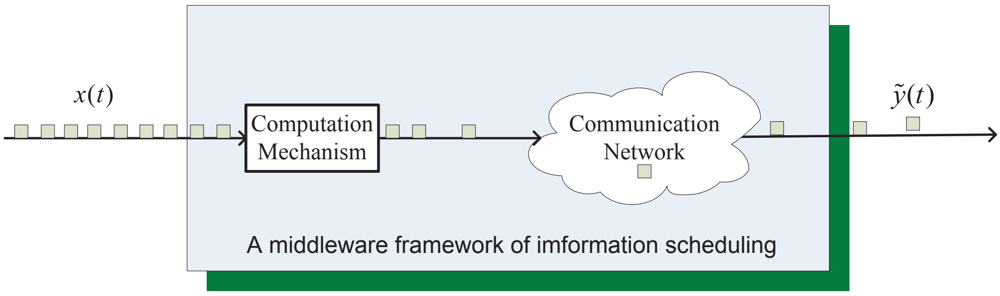

Figure 1.5: A middleware framework for information scheduling

More specifically, in this thesis, the link between "network dynamics" and "networked system" is proposed in the form of a middleware framework as shown in Figure 1.6 and 1.7, which integrates an advanced event-triggered communication scheme and a fluid-flow model of a dynamic network. This then enables control and filtering for networked systems to be investigated, such that the network resources are utilised efficiently and the desired performance of the networked system is preserved.

The contributions of this thesis can be classified under two headings: theoretical contributions and practical contributions. In the theoretical category, an event-triggered communication scheme considering both the system states and network dynamic characteristics is developed to effectively utilise the communication network resources while preserving the desired performance of networked systems. Several controller and filter design methods are proposed to regulate the utilisation of the

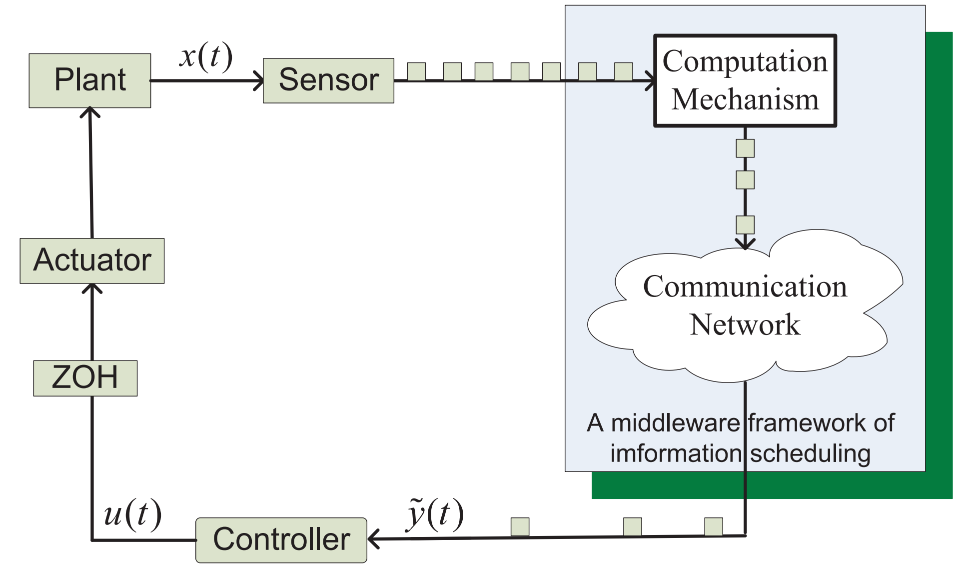

Figure 1.6: A middleware framework for networked control

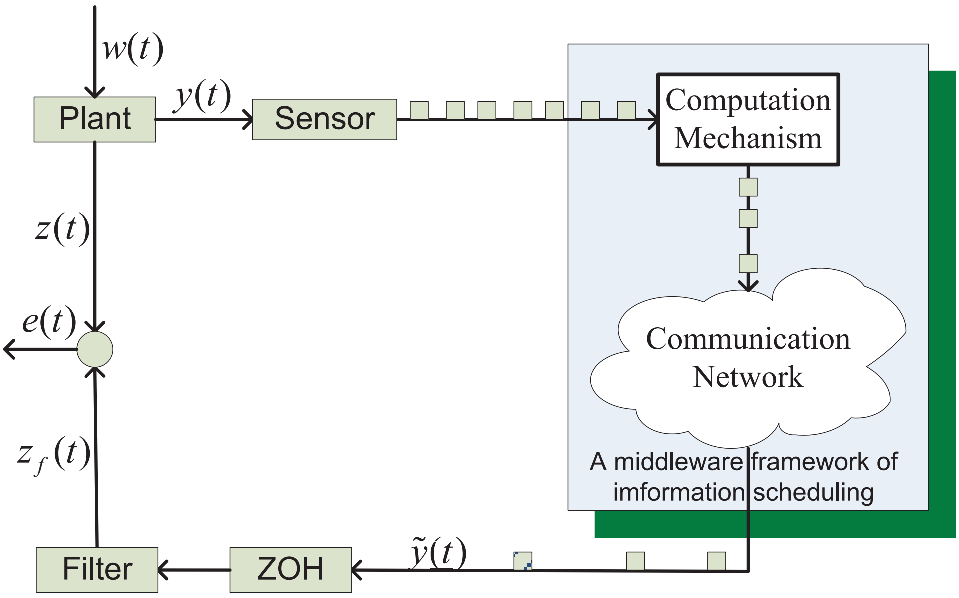

Figure 1.7: A middleware framework for networked filtering

scarce communication network resources and to make the control system satisfy the desired performance. In the practical category, a middleware framework is developed to regulate the utilisation of the scarce communication network resources. Such a middleware framework, integrating computation and communication, relates to both general usage and engineering applications.

The major contributions of this thesis are listed as:

- A novel scheduling middleware is proposed to balance the performance of networked systems and the utilisation of communication network resources. Under the proposed scheduling middleware, a new event-triggered transmission scheme is developed in an Information Selection Module to select the "needed" data, by taking both the states of control systems and network dynamics into account;

- An Information Dispatching Middleware is constructed for networked systems to investigate the H_∞ filtering issue. This middleware includes two modules: the Information Selection Module to regulate the transmission of the sampled data and the Congestion Avoidance Module to maintain the QoS of the communication network for the sampled data released by the Information Selection Module;

- New criteria are derived for network dynamic-based distributed H_∞ control of large-scale networked systems, where a distributed event-triggering scheme is proposed to trigger the data asynchronously. To achieve these results, both the networked system and network dynamics are modelled in an Information Dispatching Middleware to hide the low level details and to reduce the engineering burden;

- For distributed H_∞ filtering of networked systems, the interactive mechanism of the Information Dispatching Middleware is studied. An online scheduling policy that assigns priorities as an evaluation function is proposed. The operation of the evaluation function is based on two principal parameters: one is the QoS index of communication networks; the other is the scale of the error of each sensor measurement output between the current sampled signal and the latest transmitted signal. Then the event-triggered scheme is constructed based on the proposed evaluation function to schedule the sampled data to efficiently utilise the network resources.

1.5 Organisation of this thesis

This thesis focuses on establishing a link between “control over networks” and “control of networks”, by using an Information Dispatching Middleware to codesign networked systems and communication networks in a novel unified framework. The outline of this research is presented as follows:

Chapter 2 investigates the stability and stabilisation of networked systems under simultaneous consideration of control system performance and network dynamic behaviour. Intelligent computation and the communication network are integrated in a new framework to trade off the performance of the networked system and the utilisation of communication network resources. In terms of the given framework, a novel event-triggered transmission scheme considering both the states of the control system and network dynamics is developed to schedule the sampled data. Under this event-triggered scheme, only those sampled signals violating the preset judgement triggering condition will be transmitted. Then an augmented system of the proposed framework is formulated and by using Lyapunov-Krasovaskii functional theory, a stability criterion is derived. Based on this proposed condition, the feedback controllers are designed to asymptotically stabilise the networked control system and to regulate the utilisation of network communication resources. Finally, a numerical example is given to illustrate the merits and effectiveness of the proposed method.

Chapter 3 deals with the H_∞ filtering issue for networked systems by an online scheduling strategy based on network dynamics. First, an Information Dispatching Middleware is constructed to establish a novel filtering framework, where two modules namely the Information Selection Module and the Congestion Avoidance Module are introduced. The Information Selection Module aims to regulate the transmission of the sampled data in terms of a predefined event-triggering condition. The Congestion Avoidance Module is used to schedule the sampled data released by the Information Selection Module to the filter. Second, an online scheduling strategy is proposed under this framework to evaluate the network induced time-varying delay. Then the filtering error system based on network dynamics is formulated as a system with two interval time-varying delays. Third, a Lyapunov-Krasovskii functional approach is employed to formulate a new sufficient condition to ensure the stability and to guarantee a prescribed H_∞ noise attenuation performance for the filtering error system. Based on this condition, the H_∞ filter, network congestion controllers and event-triggering parameters can be codesigned provided that a set of linear matrix inequalities are feasible. Finally, the effectiveness of the proposed method is demonstrated through a mechanical system with two masses and two springs.

Chapter 4 is concerned with the problem of network dynamic-based distributed H_∞ control for large-scale distributed networked systems. The control signals are transmitted from distributed networked systems through a shared IP-based communication network to their controllers. Under the proposed Information Dispatching Middleware, the distributed networked systems' states are sampled periodically with network time synchronisation. In the Information Selection Module of the IDM, distributed systems are sampled into a group of event-triggered detectors to select the "needed" sampled signals, which violate the predefined event judgement condition, to be transmitted. The selected sampled signals are immediately encapsulated into a uni-packet by an event-generator. The uni-packet is then released to the shared IP-based communication network, whose network dynamics are regulated by the Congestion Avoidance Module of the Information Dispatching Middleware. A codesign method is given to trade off the transmission rate, reflected by the threshold of the event judgement function and the QoS of the communication network. By constructing a Lyapunov-Krasovskii functional, a sufficient condition is derived for the augmented system to be asymptotically stable with an H_∞ performance. Based on this criterion, the codesign method of the event-triggered scheme, distributed controllers and the congestion controllers to maintain the QoS of the communication network is correspondingly derived. Finally, a quadruple-tank process is used to illustrate the effectiveness of the proposed method.

Chapter 5 studies the distributed sensing and H_∞ filtering for networked systems where the interactive mechanism of the proposed Information Dispatching Middleware is investigated. First, an evaluation function is proposed based on two important parameters: one is the QoS index of the communication networks; the other is the scale of the error of the sensor's measurement output between the current sampled signal and the latest triggered signal. Then, by using the proposed evaluation function, an event-triggered scheme is constructed considering both dynamic measurement outputs and communication network states to efficiently utilise the precious network resources. Under the proposed event-triggered scheme, the sampled signals are determined to be transmitted or discarded depending on three factors: system measurement outputs, evaluation of the sampled signals and network dynamics. A sufficient condition is established for the existence of distributed H_∞ filters that would render the resulting filtering error system asymptotically stable and for which a prescribed disturbance attenuation performance index is guaranteed. Then the design method of event-triggered filters and the congestion controllers is posed in terms of linear matrix inequalities (LMIs). Finally, the simulation results illustrate the cooperation between networked systems and communication networks and the effectiveness of the proposed method.

Chapter 6 concludes the thesis and proposes several future research topics on networked systems.

Chapter 2

Stabilisation of networked control systems based on network dynamics

2.1 Introduction

This chapter aims at investigating the control issue for networked systems with a new framework based on network dynamics. During the past decade, the progress in digital computation and communication networks has promoted the development of networked control systems (NCSs) in which sensors and actuators are connected to a remote controller via a shared communication medium [109]. Due to the advances in computing and communication technology, NCSs have become increasingly prevalent in industrial control fields. Although introducing the communication network in the control loop provides several benefits, it raises a number of challenges that need to be addressed as mentioned in Chapter 1. In addition, it is acknowledged that the data transmitted in the communication network consumes network resources. So it is significant to investigate the efficient utilisation of network resources while preserving the desired level of performance of the control system.

In an NCS, data is usually sampled and transmitted according to time-triggered schemes due to the well-developed theory for sampled data control systems. The time-triggered scheme is executed according to the elapse of time and the choice of the period is usually based on the worst case, so the time-triggered scheme may lead to inefficient utilisation of precious network resources, processor usage, energy and so on [50]. To overcome the drawbacks of the time-triggered scheme, recent work on event-triggered schemes has appeared in the open literature that provides an efficient strategy to conserve communication resources while preserving the desired performance of the control system [60], [110], [111]. The event-triggered scheme uses a threshold condition to determine which sampled data should be transmitted through the communication network to the controller. By using the event-triggered scheme, the overall system performance is preserved while the real-time system's utilisation of communication network resources is reduced. Among the above-mentioned results, most of the event-triggered conditions are based on the control system's current state, the error between the current states and the latest transmitted states, while few results consider the dynamics of the communication networks.

It is recognised that the quality of service (QoS) of the communication network influences the performance of control systems, and whose design in turn impacts on the QoS of the network [1], [4], [104]. Hence, the network should be able to dynamically allocate necessary resources to the control system whenever needed, and also the control system should accept some network QoS degradation. A successful implementation of control systems should design the controllers considering both the control system performance and the network dynamics. In addition, the transmitted data determined by the event-triggered scheme is so important for the whole performance of the control system that how to transmit the "necessary" sampled data reliably becomes a new research issue. Therefore, it is important to consider network resource allocation and data scheduling when studying the stabilisation of networked systems. However, as far as we know, there is no published research that considers network dynamics and the control system performance in a unified framework, which motivates the current study.

Based on the above discussion, the objective of this chapter is to propose a new framework for a networked system to study its stability and stabilisation. In this study, both the performance of the control system and dynamics of the communication network are considered. First, the network dynamics are described in the form of a fluid-flow model. Then, based on the fluid-flow model and an event-triggered scheme, an augmented system with time-varying delays is modelled. By constructing a novel Lyapunov-Krasvikii functional and using a convex delay analysis approach, a stability criterion is derived, which establishes the relationship between the control system's performance and network dynamic behaviour such that the stability of the networked system is guaranteed while the network resources are utilised efficiently. Based on the stability criterion, the controller gains are derived both to preserve the stability of the system and to avoid the congestion of the communication network.

The organisation of this chapter is as follows. Section 2.2 proposes a new framework for modelling networked control systems based on network dynamics and an event-triggered scheme. Then the networked system and network dynamics are modelled in the form of an augmented system with two time-varying delays. The stability and stabilisation of the augmented system are investigated in Section 2.3 and Section 2.4, respectively. The effectiveness of the proposed method is illustrated in Section 2.5 through a simulation example. Section 2.6 concludes this chapter.

2.2 Integrate intelligent computation and a communication network for a networked control system

2.2.1 A new framework for a networked control system

Consider the following control plant

$$\dot{x}(t) = A_p x(t) + B_p u(t) \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ are the state vector and control input vector, respectively; A_p, B_p are known as constant matrices of appropriate dimensions. A new framework of the networked control system based on network dynamics is presented in Figure 2.1, where the network dynamics are modelled in the form of a fluid-flow model (FFM).

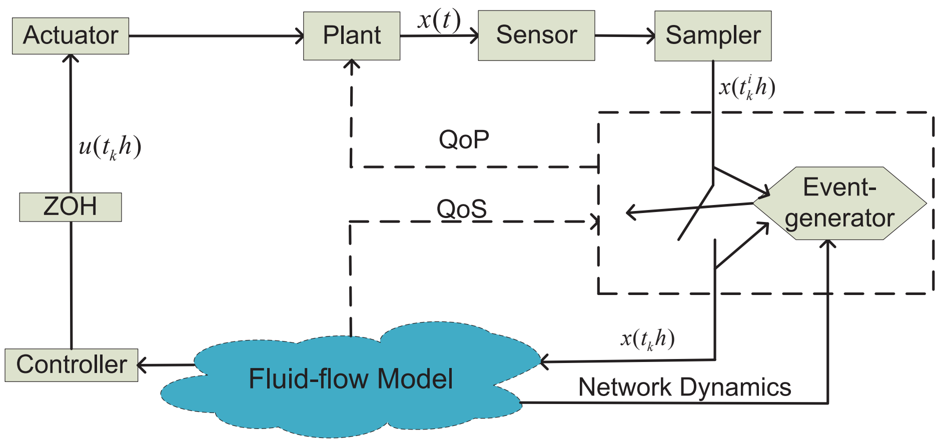

Figure 2.1: A new framework of an NCS based on an FFM

In this framework, the networked control system is comprised of a plant to be controlled and a sensor with a sampler, an actuator with a logic zero-order-holder (ZOH), and a controller whose operation is coordinated through the communication network. The sensor is clock-driven; the controller and the actuator are event-driven. Considering the effects of the inserted communication network, in order to utilise the network resources efficiently, a novel event-triggered communication scheme is proposed by employing a device called an event-generator to determine whether or not the sampled data should be transmitted to ensure a desired level of performance of the networked control system. How to schedule these “needed” sampled data in a reliable communication network within a bounded time delay has become a challenging issue. In this chapter, a fluid-flow model of a dynamic network based on TCP protocols is adopted to deal with this issue.

2.2.2 Network dynamics

A nonlinear model of TCP network behaviour is developed using a fluid-flow and stochastic differential equation analysis approach [102]. The model incorporates the average values of typical network variables such as the average TCP window size and the average queue length. The model can be described by the following coupled and nonlinear delay differential equations which ignore the TCP timeout mechanism

$$\begin{cases} \dot{W}(t) = \dfrac{1}{\tau(t)} - \dfrac{W(t)}{2}\dfrac{W(t-\tau(t))}{\tau(t-\tau(t))}p(t-\tau(t)) \\ \dot{q}(t) = \begin{cases} -C(t) + \dfrac{N(t)}{\tau(t)}W(t), q(t) > 0 \\ \max\{0, -C(t) + \dfrac{N(t)}{\tau(t)}W(t)\}, q(t) = 0, \end{cases} \\ \tau(t) = \dfrac{q(t)}{C(t)} + T_p, \end{cases} \tag{2.2}$$

where $W(t)$ is the TCP window size (in packets); $q(t)$ is the queue length in the router (in packets); T_p is the fixed propagation delay (in seconds); $\tau(t)$ is the round-trip time (in seconds); $C(t)$ is the available link capacity (in packets/second); $N(t)$ is the number of TCP sessions and $p(t)$ $(0 \leq p(t) \leq 1)$ is the packet-dropping probability function, which is the control input used to reduce the sending rate and to maintain the bottleneck queue.

In practical networks, the available link capacity changes with time and it is taken as a disturbance in most of the studies [88], [112]. In this chapter, it is assumed that the nominal value C_0 is known. Take $(W(t), q(t))$ as the state and $p(t)$ as the input. For a given triplet of network parameters (N, C_0, T_0), any triple (W_0, q_0, p_0) that is in the set $\Upsilon = \{(W_0, q_0, p_0) : W_0 \in [0, \bar{W}], q_0 \in [0, \bar{q}], p_0 \in [0, 1], \tau_0 = \frac{q_0}{C_0} + T_p, W_0 = \frac{\tau_0 C_0}{N}, p_0 = \frac{2}{W_0^2}\}$ is a possible point, where $\bar{W}$ and $\bar{q}$ denote the maximum window size and the buffer capacity, respectively. Define

$$\begin{aligned} \delta W = W(t) - W_0,\ \delta q = q(t) - q_0, \\ \delta p = p(t) - p_0,\ \delta C = C(t) - C_0. \end{aligned} \tag{2.3}$$

Then linearise the equation (2.2) at the operating point such that the nonlinear model could be expressed in the form of the following linear time-delay system [113].

$$\left\{\begin{array}{rcl} \delta\dot{W}(t) & = & -\frac{N}{\tau_0^2 C_0}(\delta W(t) + \delta W(t-\tau_0)) \\ & & -\frac{1}{\tau_0^2 C_0}(\delta q(t) - \delta q(t-\tau_0)) - \frac{\tau_0 C_0^2}{2N^2}\delta p(t-\tau_0) \\ & & +\frac{\tau_0 - T_p}{\tau_0^2 C_0}(\delta C(t) - \delta C(t-\tau_0)), \\ \delta\dot{q}(t) & = & \frac{N}{\tau_0}\delta W(t) - \frac{1}{\tau_0}\delta q(t) - \frac{T_p}{\tau_0}\delta C(t), \end{array}\right. \tag{2.4}$$

Let $\tilde{x}_1(t) = \delta W(t)$, $\tilde{x}_2(t) = \delta q(t)$, $\tilde{x}(t) = [\tilde{x}_1(t)\ \tilde{x}_2(t)]^T$, $\tilde{u}_1(t) = \delta p(t)$. Then, based on the studies in [102], [113], we develop the congestion control algorithm and present a generalised fluid-flow model to codesign the communication network and networked systems' performance. The generalised fluid-flow model of network dynamic behaviour can be described as

$$\left\{\begin{array}{rcl} \dot{\tilde{x}}(t) & = & \tilde{A}\tilde{x}(t) + \tilde{A}_d\tilde{x}(t-\tau(t)) + \tilde{B}_1\tilde{u}_1(t-\tau(t)) + \tilde{B}_2\tilde{u}_2(t) + \tilde{D}v(t) \\ \tilde{z}(t) & = & \tilde{H}\tilde{x}(t) \end{array}\right. \tag{2.5}$$

where $\tilde{x}(t) \in \mathbb{R}^2$, $\tilde{u}_1(t) \in \mathbb{R}^1$, $\tilde{u}_2(t) \in \mathbb{R}^1$, $v(t) \in \mathbb{R}^2$, $\tilde{z}(t) \in \mathbb{R}^2$, represent the internal state, the internal control input, the external control strategy, the external disturbance and the output of the network, respectively; and $\tau(t)$ is the network-induced time delay bounded by $\tau_m \leq \tau(t) \leq \tau_M$, with

$$\begin{array}{rcl} \tilde{A} & = & \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & -\frac{1}{\tau_0^2 C_0} \\ \frac{N}{\tau_0} & -\frac{1}{\tau_0} \end{bmatrix}, \tilde{A}_d = \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & \frac{1}{\tau_0^2 C_0} \\ 0 & 0 \end{bmatrix}, \\ \tilde{B}_1 & = & \begin{bmatrix} -\frac{\tau_0^2 C_0}{2N^2} \\ 0 \end{bmatrix}, \qquad \tilde{D} = \begin{bmatrix} \frac{\tau_0 - T_p}{\tau_0^2 C_0} & -\frac{\tau_0 - T_p}{\tau_0^2 C_0} \\ -\frac{T_p}{\tau_0} & 0 \end{bmatrix}, \\ \tilde{B}_2 & = & \begin{bmatrix} -\frac{\tau_0^2 C_0}{2N^2} \\ 0 \end{bmatrix}, \qquad \tilde{H} = \begin{bmatrix} 0 & 1 \end{bmatrix}. \end{array}$$

Assume that the pair $(\tilde{A}, \tilde{B}_1)$ is controllable and the pair $(\tilde{A}, \tilde{H})$ is observable.

From the above discussion, one can design the state feedback controller as $\tilde{u}_1(t-\tau(t))$ and the external control strategy $\tilde{u}_2(t)$ to adjust the dynamic behaviour of the fluid-flow network. Equation (2.4) describes the network dynamics, which are set by the TCP protocols on the window sizes and the queue management scheme on the queue length, while a buffer is used to manage the packets.

2.2.3 An event-triggered scheme

The event-generator is the most important part in this framework, which provides a tradeoff between the communication resources and the system's performance. Under a periodic sampling mechanism and a network time synchronisation mechanism, both the state of the control plant and the state of the fluid-flow network are sampled synchronously at a constant period h. The set of sampled instants is represented by $\{jh|j \in \mathbb{N}\}$. The sampled data of both the control plant state $x(jh)$ and the fluid-flow model state $\tilde{x}(jh)$ reflecting the queue length and window size, is received by the event-generator. Then the event-generator determines whether or not the sampled data should be packaged and sent to the controller via the fluid-flow model for the dynamic network.

The released instants are defined as $t_k h$, $t_{k+1}h$,... and $x(t_k h)$, $x(t_{k+1}h)$,...are the corresponding states of the control plant, respectively. Hence the set of transmission instants can be represented by $\{t_k h|t_k \in \mathbb{N}\}$, where $t_1 = 1$ is the initial transmitted instant. Because of the imperfect communication network, for the transmitted data $x(t_k h)$, the fluid flow network introduces a communication delay $\tau(t_k h) = \frac{q(t_k h)}{C(t_k h)} + T_p$, with $\tau(t_k h) \in [\tau_m, \tau_M]$, $k = 1, 2, ...$, where $k \in \mathbb{N}$. Then the released signals $x(t_k h)$, $x(t_{k+1}h)$,... reach the controller at times $t_k h + \tau(t_k h)$, $t_{k+1}h + \tau(t_{k+1}h))$,.... By using the logic ZOH, the actuator keeps the data received at $t_k h + \tau(t_k h)$ available until the new data arrives at $t_{k+1}h + \tau(t_{k+1}h)$. We assume that the latest transmitted state $x(t_k h)$ of the control plant is stored in the storage of the event-generator. The following sampled data of the control plant is expressed as $x(t_k^i h)$, where $t_k^i h = t_k h + ih, i \in \mathbb{N}$. Meanwhile, the corresponding sampled state of the fluid-flow model is $\tilde{x}(t_k^i h)$. The state error of the control plant between the latest transmitted data $x(t_k h)$ and the corresponding sampled data $x(t_k^i h)$ can be calculated as

$$e(t_k^i h) = x(t_k^i h) - x(t_k h) \tag{2.6}$$

Taking the dynamic network behaviour into account, the sampled data $x(t_k^i h)$ of the control plant is packaged and transmitted by the event-generator only when the current sampled data $x(t_k^i h)$ of the control system, the state error $e(t_k^i h)$, the state of the fluid-flow network $\tilde{x}(t_k^i h)$ and dynamics priority evaluation function of the networked control system $\psi(e(t_k^i h), \tilde{x}(t_k^i h))$, satisfy the following judgement condition

$$\begin{bmatrix} e(t_k^i h) \\ \\ \psi(e(t_k^i h), \tilde{x}(t_k^i h)) \end{bmatrix}^T \Omega \begin{bmatrix} e(t_k^i h) \\ \psi(e(t_k^i h), \tilde{x}(t_k^i h)) \end{bmatrix} \geq \lambda \begin{bmatrix} x(t_k^i h) \\ \tilde{x}(t_k^i h) \end{bmatrix}^T \Omega \begin{bmatrix} x(t_k^i h) \\ \tilde{x}(t_k^i h) \end{bmatrix} \tag{2.7}$$

where λ is a scalar parameter, belonging to $(0, 1)$; and $\Omega \in \mathbb{R}^{(n+2)\times(n+2)}$ is a positive definite symmetric matrix to be determined.

Remark 2.1. *The dynamic priority evaluation function $\psi(e(t_k^i h), \tilde{x}(t_k^i h))$ comes from the Large Error First (LEF) online scheduling algorithm, which is proposed in [114]. The priorities are assigned by a pre-determined policy based on the function of the errors obtained from the plant states and the states of the network dynamics. The plant with the largest error has the highest priority.*

For ease of analysis and exposition of the essential feature of the event-generator, an event-triggered function $f(x_{t_k^i}, \tilde{x}_{t_k^i}, e_{t_k^i}, \psi_{t_k^i})$ of the event-generator is defined as

$$f(x_{t_k^i}, \tilde{x}_{t_k^i}, e_{t_k^i}, \psi_{t_k^i h}) = \begin{bmatrix} e_{t_k^i} \\ \psi_{t_k^i h} \end{bmatrix}^T \Omega \begin{bmatrix} e_{t_k^i} \\ \psi_{t_k^i h} \end{bmatrix} - \lambda \begin{bmatrix} x_{t_k^i} \\ \tilde{x}_{t_k^i} \end{bmatrix}^T \Omega \begin{bmatrix} x_{t_k^i} \\ \tilde{x}_{t_k^i} \end{bmatrix} \tag{2.8}$$

where $x_{t_k^i} \triangleq x(t_k^i h)$,$\tilde{x}_{t_k^i} \triangleq \tilde{x}(t_k^i h)$, $e_{t_k^i} \triangleq e(t_k^i h) = x(t_k^i h) - x(t_k h)$, $\psi_{t_k^i h} = \psi(e_{t_k^i}, \tilde{x}_{t_k^i})$. It can be seen from (2.7) and (2.8), when the event-triggered function $f(x_{t_k^i}, \tilde{x}_{t_k^i}, e_{t_k^i}, \psi_{t_k^i})$ < 0, the packet is not transmitted to the controller via the fluid-flow model of the communication network. Only the sampled data satisfying $f(x_{t_k^i}, \tilde{x}_{t_k^i}, e_{t_k^i}, \psi_{t_k^i h}) \geq 0$, is packaged and transmitted to the controller. In other words, the event-generator's decision is triggered when $f(x_{t_k^i}, \tilde{x}_{t_k^i}, e_{t_k^i}, \psi_{t_k^i h}) \geq 0$. Therefore, based on the above analysis, the next transmission instant corresponding to the transmitted instant $t_k h$ can be expressed as

$$t_{k+1}h = t_k h + \inf_{i \geq 1}\{ih | f(x_{t_k^i}, \tilde{x}_{t_k^i}, e_{t_k^i}, \psi_{t_k^i h}) \geq 0\} \tag{2.9}$$

Obviously, the set of transmission instants are $\{t_k h | t_1 h, t_2 h, \ldots, t_k h, ..., t_k \in \mathbb{N}\}$ which is a subset of the set of sampled instants $\{jh | j \in \mathbb{N}\}$. There may be some sampled data which are not triggered in the interval $t \in [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$. We now analyse the sampled data in the interval $t \in [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$. For technical convenience, consider the following two cases:

Case 1: if $t_k h + h + \tau(t_k h) > t_{k+1}h + \tau(t_{k+1}h)$, as shown in Figure 2.2

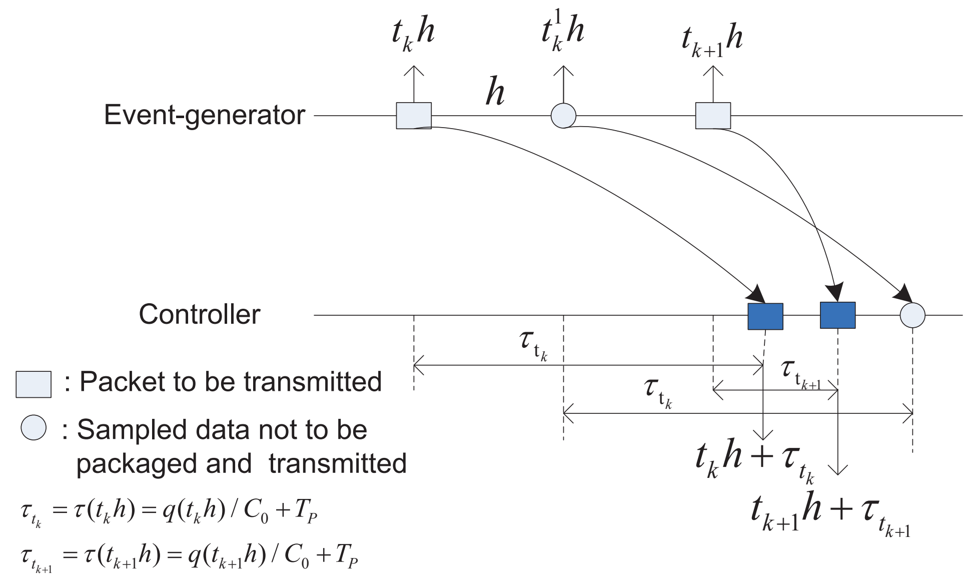

Figure 2.2: Case 1 for the package transmission

Define

$$d(t) = t - t_k h, for\ t \in [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h)) \quad (2.10)$$

Thus,

$$\tau_m \le \tau(t_k) \le d(t) \le (t_{k+1} - t_k)h + \tau(t_{k+1}h) \le h + \tau_M \quad (2.11)$$

Case 2: if $t_k h + h + \tau(t_k h) \le t_{k+1}h + \tau(t_{k+1}h)$, as shown in Figure 2.3

There exists an l_M such that $(t_k + l_M)h + \tau(t_k h) \le t_{k+1}h + \tau(t_{k+1}h)$ and $(t_k + l_M + 1)h + \tau(t_k h) > t_{k+1}h + \tau(t_{k+1}h)$. Then it can be seen that

$$[t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h)) = \Pi_1 \cup \Pi_2$$

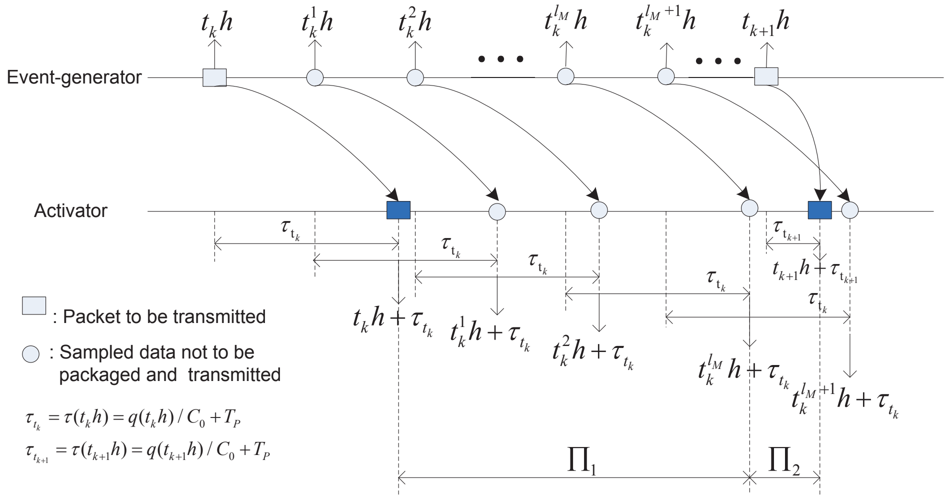

Figure 2.3: Case 2 for the package transmission

where $\Pi_1 = \cup_{i=0}^{l_M-1}\Pi_1^i, \Pi_1^i = [t_kh + ih + \tau(t_kh), t_kh + (i+1)h + \tau(t_kh))$, $\Pi_2 = [(t_k + l_M)h + \tau(t_kh), t_{k+1}h + \tau(t_{k+1}h))$.

Define

$$d(t) = \begin{cases} t - (t_k + i)h, & t \in \Pi_1^i \\ t - (t_k + l_M)h, & t \in \Pi_2 \end{cases}$$

with

$$\begin{cases} \tau_m \le \tau(t_kh) \le d(t) \le h + \tau(t_kh) \le h + \tau_M, & t \in \Pi_1^i \\ \tau_m \le \tau(t_kh) \le d(t) \le h + \tau(t_kh) \le h + \tau_M, & t \in \Pi_2 \end{cases}$$

Then

$$e(t_k^ih) = \begin{cases} x((t_k + i)h) - x(t_kh), & t \in \Pi_1^i \\ x((t_k + l_M)h) - x(t_kh), & t \in \Pi_2 \end{cases}$$

From the above discussion, we obtain that $[t_kh + \tau(t_kh), t_{k+1}h + \tau(t_{k+1}h)) = \Pi_1 \cup \Pi_2$. Let $l = t_{k+1} - t_k - 1$. If $\tau(t_kh) \le h + \tau(t_{k+1}h)$, $l_M = l$; if $\tau(t_kh) > h + \tau(t_{k+1}h)$, l_M is as defined in Case 2, with $t_k^ih = t_kh + ih$; $i = 0, ..., l$. t_k^ih is the sampling instant between two successful transmitted instants.

In the event-generator, when the sampled data $x(t_kh)$ are prepared to be transmitted, they are also used for the calculation of $e(t_k^ih)$. Since $e(t_k^ih) = x(t_k^ih) - x(t_kh)$, then $x(t_kh) = x(t_k^ih) - e(t_k^ih)$ and $d(t) = t - t_k^ih$, then

$$x(t_kh) = x(t - d(t)) - e(t_k^ih) \tag{2.12}$$

where $d(t)$ is a piecewise-linear with the derivative $\dot{d}(t) = 1$ and $\tau_m \leq d_m \leq d(t) \leq d_M < \infty$, for $t \neq t_k^i h + \tau(t_k^i h)$. $d_m = \min\{\tau(t_k^i h)\} \geq \tau_m$ and $d_M = h + \tau_M$, where $t_k^i \in \mathbb{N}$.

Remark 2.2. *From (2.9), one can see that the transmission events are dependent not only on the system's performance, but also on the network dynamics. This event-triggered scheme provides a tradeoff between system performance and network dynamics. Without considering the dynamic behaviour of the communication network, the scheduling strategy of the event-generator is degenerated to the case with the discrete event-triggered communication scheme as discussed in [110]. When* $\lambda = 0$*, it means all the sampled data are transmitted. Then it reduces to the time-triggered transmission scheme.*

Obviously, under the judgement algorithm (2.7), the event-generator reduces the the burden of the communication network and saves communication bandwidth. Furthermore, if the dynamic behaviour of the communication network reflects that the network is not congested at this time, then more sampled data of the control system is sent out such that the performance of the control system is enhanced and the network resources is efficiently utilised without any idle.

2.2.4 Modelling of networked control systems

In this framework, a dynamic priority evaluation function $\psi(t_k^i h)$ of the networked control system is employed to prevent congestion occurring in the communication network. The state error $e(t_k^i h)$ is viewed as the external disturbance of the network to schedule the other information, such as the short-term data flow related to this alteration. It can be viewed as the external disturbance of the network in the form of $\tilde{v}(t) = \tilde{F}e(t)$, where $\tilde{F}$ is the weight parameter of the external disturbance. Assume the feedback $u(t)$ controller of the networked system (2.1) and the congestion controllers $\tilde{u}_1(t)$, $\tilde{u}_2(t)$ of network dynamics (2.5) are in the form of

$$u(t) = Kx(t_k h), \tag{2.13}$$

$$\tilde{u}_1(t) = \tilde{K}_1 \tilde{x}(t - \tau(t_k h)), \tag{2.14}$$

$$\tilde{u}_2(t) = \tilde{K}_2 \psi(t_k^i h). \tag{2.15}$$

Therefore, using the above controllers and from (2.1) and (2.5), we obtain the augmented system as

$$\begin{aligned} \dot{\xi}(t) \quad = \quad & \mathcal{A}\xi(t) + (\mathcal{A} + \mathcal{B}_d \mathcal{K}_d)\xi(t - \tau(t_k h)) \\ & + \mathcal{B}_1 \bar{K}\xi(t - d(t)) + (\mathcal{D}_d + \mathcal{B}_2 \bar{K})\bar{e}(t_k^i h) \end{aligned} \tag{2.16}$$

where $t \in [t_k h + \tau(t_k h), t_{k+1} h + \tau(t_{k+1} h))$,

$$\xi(t) = \mathrm{col}\{x(t), \tilde{x}(t)\}, \tag{2.17}$$

$$\bar{e}(t_k^i h) = \mathrm{col}\{e(t_k^i h), \psi(t_k^i h)\}, \tag{2.18}$$

with

$$\begin{aligned} \mathcal{A} \quad &= \quad \begin{bmatrix} A_p & 0 \\ 0 & \tilde{A} \end{bmatrix}, \mathcal{A}_d = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{A}_d \end{bmatrix}, \quad \mathcal{B}_d = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{B}_1 \end{bmatrix}, \bar{K} = \begin{bmatrix} K & 0 \\ 0 & \tilde{K}_2 \end{bmatrix}, \\ \mathcal{K}_d \quad &= \quad \begin{bmatrix} 0 & 0 \\ 0 & \tilde{K}_1 \end{bmatrix}, \mathcal{D}_d = \begin{bmatrix} 0 & 0 \\ \tilde{D}\tilde{F} & 0 \end{bmatrix}, \mathcal{B}_1 = \begin{bmatrix} B_p & 0 \\ 0 & 0 \end{bmatrix}, \mathcal{B}_2 = \begin{bmatrix} -B_p & 0 \\ 0 & \tilde{B}_2 \end{bmatrix}. \end{aligned}$$

The initial condition of the state $\xi(t)$ is $\xi(t_0 + \theta) = \phi(\theta)$, where $\theta \in [-d_M, 0]$, with $\phi(0) = \xi_0$; $\phi \in \mathbb{W}$, where $\mathbb{W}$ denotes the Banach space of absolutely continuous functions $[-d_M,\ 0] \to \mathbb{R}^n$ with square-integrable derivative and with the norm

$$\|\phi\|_W^2 = \|\phi(0)\|^2 + \int_{-d_M}^{0} \|\phi(s)\|^2 ds + \int_{-d_M}^{0} \|\dot{\phi}(\theta)\|^2 d\theta$$

where the vector norm $\|\cdot\|$ represents the Euclidean norm.

By proposing a novel event-triggered scheme and a Lyapunov-Krasvikii functional and considering network-induced time-varying delay, we will derive a new stability criterion for the networked system, where both network behaviour and performance of the control system are considered. Based on the stability criterion, feedback controllers will be designed to allocate network resources dynamically and asymptotically stabilise the networked control system.

2.3 Stability analysis

In this section, by constructing a Lyapunov-Krasovskii functional for the augmented system (2.16), a sufficient condition under which the augmented system (2.16) can be asymptotically stable is derived.

The Lyapunov-Krasovskii functional is constructed as

$$V(t) = V_1(t) + V_2(t) + V_3(t) \tag{2.19}$$

where

$$\begin{aligned}
V_1(t) &= \xi^T(t)P_1\xi(t) + \int_{t-\tau_m}^{t} \xi^T(s)P_2\xi(s)ds + \int_{t-\tau_M}^{t-\tau_m} \xi^T(s)P_3\xi(s)ds \\
&\quad +\tau_m \int_{-\tau_m}^{0}\int_{t+\theta}^{t} \dot{\xi}^T(s)P_4\dot{\xi}(s)dsd\theta \\
&\quad +(\tau_M-\tau_m)\int_{-\tau_M}^{-\tau_m}\int_{t+\theta}^{t} \dot{\xi}^T(s)P_5\dot{\xi}(s)dsd\theta \\
V_2(t) &= \int_{t-d_m}^{t} \xi^T(s)R_1\xi(s)ds + \int_{t-d_M}^{t-d_m} \xi^T(s)R_2\xi(s)ds \\
&\quad +d_m \int_{-d_m}^{0}\int_{t+\theta}^{t} \dot{\xi}^T(s)R_3\dot{\xi}(s)dsd\theta \\
&\quad +(d_M-d_m)\int_{-d_M}^{-d_m}\int_{t+\theta}^{t} \dot{\xi}^T(s)R_4\dot{\xi}(s)dsd\theta \\
V_3(t) &= (d_M-d(t))(\xi(t)-\xi(t-\rho(t)))^T Q_1(\xi(t)-\xi(t-\rho(t))) \\
&\quad +(d_M-d(t))\int_{t-\rho(t)}^{t} \dot{\xi}^T(s)Q_2\dot{\xi}(s)ds
\end{aligned}$$

and where $\rho(t) = d(t) - \tau_{t_k}$.

Before carrying out the stability condition for the augmented system (2.16), the following lemmas are introduced.

Lemma 2.1. *[115]. For any constant matrix $R \in \mathbb{R}^{n\times n}$, $R = R^T > 0$, scalar $\tau > 0$ and vector function $\dot{x} : [t-\tau, t] \to \mathbb{R}^n$ such that the following integration is well defined, then*

$$-\tau \int_{t-\tau}^{t} \dot{x}^T(s)R\dot{x}(s)ds$$

$$\leq \begin{bmatrix} x(t) \\ x(t-\tau) \end{bmatrix}^T \begin{bmatrix} -R & R \\ \star & -R \end{bmatrix} \begin{bmatrix} x(t) \\ x(t-\tau) \end{bmatrix}$$

.

Lemma 2.2. *[116]. For any constant matrix $Q \in \mathbb{R}^{n\times n}$, scalars $\tau_m \leq \tau(t) \leq \tau_M$, and vector function $\dot{x} : [-\tau_M, -\tau_m] \to \mathbb{R}^n$ such that the following integration is well defined, then it holds that*

$$\begin{aligned} &-(\tau_M - \tau_m) \int_{t-\tau_M}^{t-\tau_m} \dot{x}^T(s) Q \dot{x}(s) ds \\ &\leq \psi^T(t) \begin{bmatrix} -Q & Q & 0 \\ \star & -2Q & Q \\ \star & \star & -Q \end{bmatrix} \psi(t) \end{aligned} \tag{2.20}$$

where $\psi^T(t) = \begin{bmatrix} x^T(t-\tau_m) & x^T(t-\tau(t)) & x^T(t-\tau_M) \end{bmatrix}$.

Lemma 2.3. *[117]. For a symmetric positive definite matrix $Q \in \mathbb{R}^{n\times n}$ and $W_l \in \mathbb{R}^{n\times n}$ $(l = 1, 2)$, such that the following integration is well defined, then it holds that*

$$\begin{aligned} -\int_{t-\rho(t)}^{t} \dot{x}^T(s) Q \dot{x}(s) ds \leq \eta^T(t) &\begin{bmatrix} W_1 + W_1^T & W_2 - W_1^T \\ \star & -W_2 - W_2^T \end{bmatrix} \eta(t) \\ &+\rho(t)\eta^T(t) \begin{bmatrix} W_1^T \\ W_2^T \end{bmatrix} Q^{-1} \begin{bmatrix} W_1^T \\ W_2^T \end{bmatrix}^T \eta(t) \end{aligned} \tag{2.21}$$

where $\eta^T(t) = \begin{bmatrix} \xi^T(t) & \xi^T(t-\rho(t)) \end{bmatrix}^T$.

Lemma 2.4. *[12]. Assume that there exist positive scalars ϵ_1, ϵ_2, ϵ_3 and a Lyapunov functional $V : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, for $t \geq t_0$ and $V(t)$ is continuously differentiable at $t \neq t_k h + \tau(t_k h)$, such that the corresponding solution $\xi(t)$ to system (2.16) and the Lyapunov functional $V(t)$ satisfy*

$$\epsilon_1 \|\xi(t)\|^2 \leq V(t) \leq \epsilon_2 \|\xi(t)\|^2, \tag{2.22}$$

$$\frac{dV(t)}{dt} \leq -\epsilon_3 \|\xi(t)\|^2, t \neq t_k h + \tau(t_k h), \tag{2.23}$$

$$\lim_{t\uparrow(t_k h + \tau(t_k h))} V(t) \geq V(t)|_{t=t_k h+\tau(t_k h)} \tag{2.24}$$

for any $t_k \in \mathbb{N}$. Then, the system (2.16) is asymptotically stable.

Applying Lemma 2.1 to Lemma 2.4, we have the following result.

Theorem 2.1. *For given parameters τ_m, τ_M, d_m, d_M, $\lambda > 0$ and controller gains K, $\tilde{K}_1$, $\tilde{K}_2$, the augmented system is asymptotically stable in network environments if there exist matrices $\Omega > 0$, $P_i > 0$, $R_v > 0$, $Q_1 > 0$, $Q_2 > 0$, with $i = 1, 2, ..., 5$, $v = 1, 2, 3, 4$, W_1 and W_2 with appropriate dimensions such that the following matrix inequalities hold*

$$\begin{bmatrix} \Psi_{111} & \Psi_{112} & \Psi_{113} \\ \star & \Psi_{122} & 0 \\ \star & \star & \Psi_{133} \end{bmatrix} < 0 \tag{2.25}$$

$$\begin{bmatrix} \Psi_{211} & \Psi_{212} & \Psi_{213} \\ \star & \Psi_{222} & \Psi_{223} \\ \star & \star & \Psi_{233} \end{bmatrix} < 0 \tag{2.26}$$

where

$$\Psi_{111} = \begin{bmatrix} \Lambda_{111} & \Lambda_{112} & \Lambda_{113} & \Lambda_{114} \\ \star & -2P_5 & 0 & 0 \\ \star & \star & -2R_4 + \lambda\Omega & 0 \\ \star & \star & \star & -\Omega \end{bmatrix},$$

$$\Lambda_{111} = P_1 A + A^T P_1 + P_2 - P_4 + R_1 - R_3 - Q_1 + W_1 + W_1^T$$
$$+ (d_M - d_m)(Q_1 A + A^T Q_1),$$

$$\Lambda_{112} = P_1 A_d + P_1 B_d K_d + (d_M - d_m) Q_1 (A_d + B_d K_d),$$

$$\Lambda_{113} = P_1 B_1 \bar{K} + (d_M - d_m) Q_1 B_1 \bar{K},$$

$$\Lambda_{114} = P_1 D_d + P_1 B_2 \bar{K} + (d_M - d_m) Q_1 (D_d + B_2 \bar{K}),$$

$$\Psi_{112} = \begin{bmatrix} P_4 & 0 & R_3 & 0 & Q_1 - W_1^T + W_2 - (d_M - d_m) A^T Q_1 \\ P_5 & P_5 & 0 & 0 & -(d_M - d_m)(A_d^T + K_d^T B_d^T) Q_1 \\ 0 & 0 & R_4 & R_4 & -(d_M - d_m) \bar{K}^T B_1^T Q_1 \\ 0 & 0 & 0 & 0 & -(d_M - d_m)(D_d^T + \bar{K}^T B_2^T) Q_1 \end{bmatrix},$$

$$\Psi_{122} = diag\{-P_2 + P_3 - P_4 - P_5, -P_3 - P_5, -R_1 + R_2 - R_3 - R_4,$$
$$- R_2 - R_4, -Q_1 - W_2 - W_2^T\},$$

$$\Psi_{113} = \begin{bmatrix} \tau_m A^T P_4 & \bar{\tau} A^T P_5 & d_m A^T R_3 & \bar{d} A^T R_4 & \bar{d} A^T Q_2 \\ \tau_m \bar{A}_d^T P_4 & \bar{\tau} \bar{A}_d^T P_5 & d_m \bar{A}_d^T R_3 & \bar{d} \bar{A}_d^T R_4 & \bar{d} \bar{A}_d^T Q_2 \\ \tau_m \bar{K}^T B_1^T P_4 & \bar{\tau} \bar{K}^T B_1^T P_5 & d_m \bar{K}_1^T B_1^T R_3 & \bar{d} \bar{K}^T B_1^T R_4 & \bar{d} \bar{K}^T B_1^T Q_2 \\ \tau_m \bar{D}_d^T P_4 & \bar{\tau} \bar{D}_d^T P_5 & d_m \bar{D}_d^T R_3 & \bar{d} \bar{D}_d^T R_4 & \bar{d} \bar{D}_d^T Q_2 \end{bmatrix},$$

$$\Psi_{133} = diag\{-P_4, -P_5, -R_3, -R_4, -\bar{d} Q_2\},$$

$$\Psi_{211} = \begin{bmatrix} \Lambda_{211} & \Lambda_{212} & \Lambda_{213} & \Lambda_{214} \\ \star & -2P_5 & 0 & 0 \\ \star & \star & -2R_4 + \lambda\Omega & 0 \\ \star & \star & \star & -\Omega \end{bmatrix},$$

$$\Lambda_{211} = P_1 A + A^T P_1 + P_2 - P_4 + R_1 - R_3 - Q_1 + W_1 + W_1^T,$$

$$\Lambda_{212} = P_1 A_d + P_1 B_d K_d, \quad \Lambda_{213} = P_1 B_1 \bar{K}, \quad \Lambda_{214} = P_1 D_d + P_1 B_2 \bar{K},$$

$$\Psi_{212} = \begin{bmatrix} P_4 & 0 & R_3 & 0 & Q_1 - W_1^T + W_2 \\ P_5 & P_5 & 0 & 0 & 0 \\ 0 & 0 & R_4 & R_4 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{222} = \Psi_{122},$$

$$\Psi_{213} = \begin{bmatrix} \tau_m A^T P_4 & \bar{\tau} A^T P_5 & d_m A^T R_3 & \bar{d} A^T R_4 & \bar{d} W_1^T \\ \tau_m \bar{A}_d^T P_4 & \bar{\tau} \bar{A}_d^T P_5 & d_m \bar{A}_d^T R_3 & \bar{d} \bar{A}_d^T R_4 & 0 \\ \tau_m \bar{K}^T B_1^T P_4 & \bar{\tau} \bar{K}^T B_1^T P_5 & d_m \bar{K}^T B_1^T R_3 & \bar{d} \bar{K}^T B_1^T R_4 & 0 \\ \tau_m \bar{D}_d^T P_4 & \bar{\tau} \bar{D}_d^T P_5 & d_m \bar{D}_d^T R_3 & \bar{d} \bar{D}_d^T R_4 & 0 \end{bmatrix},$$

$$\Psi_{223} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \bar{d} W_2^T \end{bmatrix}, \Psi_{233} = \Psi_{133},$$

with $\bar{\tau} = \tau_M - \tau_m, \bar{d} = d_M - d_m, \bar{A}_d = A_d + B_d K_d, \bar{D}_d = D_d + B_2 \bar{K}$.

Proof. Taking the derivative of $V(t)$ along the trajectory of the system (2.16) yields

$$\begin{aligned} \dot{V}(t) \quad = \quad & 2\xi^T(t) P_1 \dot{\xi}(t) + \xi^T(t) P_2 \xi(t) \\ & -\xi^T(t-\tau_m) P_2 \xi(t-\tau_m) + \xi^T(t-\tau_m) P_3 \xi(t-\tau_m) \\ & -\xi^T(t-\tau_M) P_3 \xi(t-\tau_M) + \tau_m^2 \dot{\xi}^T(t) P_4 \dot{\xi}(t) \\ & -\tau_m \int_{t-\tau_m}^{t} \dot{\xi}^T(s) P_4 \dot{\xi}(s) ds + (\tau_M - \tau_m)^2 \dot{\xi}^T(t) P_5 \dot{\xi}(t) \\ & -(\tau_M - \tau_m) \int_{t-\tau_M}^{t-\tau_m} \dot{\xi}^T(s) P_5 \dot{\xi}(s) ds \\ & +\xi^T(t) R_1 \xi(t) - \xi^T(t-d_m) R_1 \xi(t-d_m) \\ & +\xi^T(t-d_m) R_2 \xi(t-d_m) - \xi^T(t-d_M) R_2 \xi(t-d_M) \\ & +d_m^2 \dot{\xi}^T(t) R_3 \dot{\xi}(t) - d_m \int_{t-d_m}^{t} \dot{\xi}^T(s) R_3 \dot{\xi}(s) ds \\ & +(d_M - d_m)^2 \dot{\xi}^T(t) R_4 \dot{\xi}(t) \\ & -(d_M - d_m) \int_{t-d_M}^{t-d_m} \dot{\xi}^T(s) R_4 \dot{\xi}(s) ds \end{aligned}$$

$$
\begin{aligned}
&-(\xi(t)-\xi(t-\rho(t)))^T Q_1(\xi(t)-\xi(t-\rho(t))) \\
&+2(d_M-d(t))(\xi(t)-\xi(t-\rho(t)))^T Q_1\dot{\xi}(t) \\
&-\int_{t-\rho(t)}^{t}\dot{\xi}^T(s)Q_2\dot{\xi}(s)ds+(d_M-d(t))\dot{\xi}^T(t)Q_2\dot{\xi}(t)
\end{aligned}
$$

Applying Lemma 2.1, we have

$$
\begin{aligned}
&-\tau_m\int_{t-\tau_m}^{t}\dot{\xi}^T(s)P_4\dot{\xi}(s)ds \\
&\leq \begin{bmatrix}\xi(t)\\ \xi(t-\tau_m)\end{bmatrix}^T \begin{bmatrix}-P_4 & P_4\\ P_4 & -P_4\end{bmatrix}\begin{bmatrix}\xi(t)\\ \xi(t-\tau_m)\end{bmatrix}, \qquad (2.27)\\
&-d_m\int_{t-d_m}^{t}\dot{\xi}^T(s)R_3\dot{\xi}(s)ds \\
&\leq \begin{bmatrix}\xi(t)\\ \xi(t-d_m)\end{bmatrix}^T \begin{bmatrix}-R_3 & R_3\\ R_3 & -R_3\end{bmatrix}\begin{bmatrix}\xi(t)\\ \xi(t-d_m)\end{bmatrix} \qquad (2.28)
\end{aligned}
$$

By Lemma 2.2, the following formulas hold

$$
\begin{aligned}
&-(\tau_M-\tau_m)\int_{t-\tau_M}^{t-\tau_m}\dot{\xi}^T(s)P_5\dot{\xi}(s)ds \\
&\leq \begin{bmatrix}\xi(t-\tau_m)\\ \xi(t-\tau(t))\\ \xi(t-\tau_M)\end{bmatrix}^T \begin{bmatrix}-P_5 & P_5 & 0\\ \star & -2P_5 & P_5\\ \star & \star & -P_5\end{bmatrix}\begin{bmatrix}\xi(t-\tau_m)\\ \xi(t-\tau(t))\\ \xi(t-\tau_M)\end{bmatrix} \qquad (2.29)\\
&-(d_M-d_m)\int_{t-d_M}^{t-d_m}\dot{\xi}^T(s)R_4\dot{\xi}(s)ds \\
&\leq \begin{bmatrix}\xi(t-d_m)\\ \xi(t-d(t))\\ \xi(t-d_M)\end{bmatrix}^T \begin{bmatrix}-R_4 & R_4 & 0\\ \star & -2R_4 & R_4\\ \star & \star & -R_4\end{bmatrix}\begin{bmatrix}\xi(t-d_m)\\ \xi(t-d(t))\\ \xi(t-d_M)\end{bmatrix} \qquad (2.30)
\end{aligned}
$$

By Lemma 2.3, it is clear that

$$
\begin{aligned}
-\int_{t-\rho(t)}^{t}\dot{\xi}^T(s)Q_2\dot{\xi}(s)ds \leq{}& \begin{bmatrix}\xi(t)\\ \xi(t-\rho(t))\end{bmatrix}^T \begin{bmatrix}W_1+W_1^T & W_2-W_1^T\\ \star & -W_2-W_2^T\end{bmatrix}\begin{bmatrix}\xi(t)\\ \xi(t-\rho(t))\end{bmatrix} \\
&+\rho(t)\begin{bmatrix}\xi(t)\\ \xi(t-\rho(t))\end{bmatrix}^T \begin{bmatrix}W_1^T\\ W_2^T\end{bmatrix} Q_2^{-1}\begin{bmatrix}W_1^T\\ W_2^T\end{bmatrix}^T \begin{bmatrix}\xi(t)\\ \xi(t-\rho(t))\end{bmatrix} \qquad (2.31)
\end{aligned}
$$

For briefness of representation, let

$$
\psi(t)=\mathrm{col}\{\xi(t),\xi(t-\tau(t)),\xi(t-d(t)),\bar{e}(t_k^i h),\xi(t-\tau_m),\xi(t-\tau_M),\xi(t-d_m),\xi(t-d_M),\rho(t)\}
$$

From (2.27)-(2.31), we then obtain

$$\dot{V}(t) \leq \psi^T(t)\Sigma\psi(t) \tag{2.32}$$

where $\Sigma = \Psi + \Gamma^T\Theta\Gamma + (d_M - d(t))(2\Gamma_1^T Q_1\Gamma + \Gamma^T Q_2\Gamma) + (d(t) - d_m)\Gamma_2^T Q_2^{-1}\Gamma_2$, with

$$\Psi = \begin{bmatrix} \Psi_{211} & \Psi_{212} \\ \star & \Psi_{222} \end{bmatrix},$$

$$\Theta = \tau_m^2 P_4 + (\tau_M - \tau_m)^2 P_5 + d_m^2 R_3 + (d_M - d_m)^2 R_4,$$

$$\Gamma = \begin{bmatrix} A & (A_d + B_d K_d) & B_1\bar{K} & D_d + B_2\bar{K} & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Gamma_1 = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -I \end{bmatrix},$$

$$\Gamma_2 = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I \end{bmatrix},$$

and $\rho(t) = d(t) - \tau(t_k^i h) \leq d(t) - d_m$ is used in (2.31). It is clear that if $\Sigma < 0$, then there exists a scalar $\epsilon_3 > 0$ such that $\dot{V}(t) \leq -\epsilon_3\xi^T(t)\xi(t) < 0$, $(t \neq t_k h + \tau(t_k h))$, which implies that (2.23) is satisfied. By Lemma 2.4, one can conclude that the system (2.16) is asymptotically stable. Notice that Σ is a convex combination of $2\Gamma_1^T Q_1\Gamma + \Gamma^T Q_2\Gamma$ and $\Gamma_2^T Q_2^{-1}\Gamma_2$ on $d(t) \in [d_m,\ d_M]$. Therefore, $\Sigma < 0$ if

$$\Psi + \Gamma^T\Theta\Gamma + (d_M - d_m)(2\Gamma_1^T Q_1\Gamma + \Gamma^T Q_2\Gamma) < 0, \tag{2.33}$$

$$\Psi + \Gamma^T\Theta\Gamma + (d_M - d_m)\Gamma_2^T Q_2^{-1}\Gamma_2 < 0 \tag{2.34}$$

hold. Applying the Schur complement to (2.33) and (2.34), we arrive at (2.25) and (2.26). Then one can conclude that if the matrix inequalities (2.25) and (2.26) are satisfied, the augmented system (2.16) is asymptotically stable, which completes the proof. □

2.4 Controller design

Based on Theorem 2.1, we are in a position to design the controller of the control system under consideration with network dynamics.

Theorem 2.2. *For given scalars τ_m, τ_M, d_m, d_M $\lambda > 0$, the augmented system is asymptotically stable in network environments if there exist matrices $\tilde{\Omega} > 0$, $X > 0$,*

$\tilde{P}_i > 0$, $\tilde{R}_j > 0$, $\tilde{Q}_1 > 0$, $\tilde{Q}_2 > 0$, *for some given positive constants* ϵ, μ_l, *with* $(i = 2, ..., 5,\ j = 1, 2, 3, 4,\ l = 1, 2, 3, 4, 5)$, $\tilde{W}_1$, $\tilde{W}_2$, Y_1 *and* Y_2 *with appropriate dimensions such that*

$$\begin{bmatrix} \tilde{\Psi}_{111} & \tilde{\Psi}_{112} & \tilde{\Psi}_{113} \\ \star & \tilde{\Psi}_{122} & 0 \\ \star & \star & \tilde{\Psi}_{133} \end{bmatrix} < 0 \tag{2.35}$$

$$\begin{bmatrix} \tilde{\Psi}_{211} & \tilde{\Psi}_{212} & \tilde{\Psi}_{213} \\ \star & \tilde{\Psi}_{222} & \tilde{\Psi}_{223} \\ \star & \star & \tilde{\Psi}_{233} \end{bmatrix} < 0 \tag{2.36}$$

where

$$\begin{aligned}
\tilde{\Psi}_{111} &= \begin{bmatrix} \tilde{\Lambda}_{111} & \tilde{\Lambda}_{112} & \tilde{\Lambda}_{113} & \tilde{\Lambda}_{114} \\ \star & -2\tilde{P}_5 & 0 & 0 \\ \star & \star & -2\tilde{R}_4 + \lambda\tilde{\Omega} & 0 \\ \star & \star & \star & -\tilde{\Omega} \end{bmatrix}, \\
\tilde{\Lambda}_{111} &= AX + XA^T + \tilde{P}_2 - \tilde{P}_4 + \tilde{R}_1 - \tilde{R}_3 - \epsilon X + \tilde{W}_1 + \tilde{W}_1^T \\
&\quad + (d_M - d_m)(\epsilon AX + \epsilon XA^T), \\
\tilde{\Lambda}_{112} &= A_d X + B_d Y_2 + \epsilon(d_M - d_m)(A_d X + B_d Y_2), \\
\tilde{\Lambda}_{113} &= B_1 Y_1 + \epsilon(d_M - d_m) B_1 Y_1, \\
\tilde{\Lambda}_{114} &= D_d X + B_2 Y_1 + \epsilon(d_M - d_m)(D_d X + B_2 Y_1), \\
\tilde{\Psi}_{112} &= \begin{bmatrix} \tilde{P}_4 & 0 & \tilde{R}_3 & 0 & \epsilon X - \tilde{W}_1^T + \tilde{W}_2 - \epsilon(d_M - d_m)XA^T \\ \tilde{P}_5 & \tilde{P}_5 & 0 & 0 & -\epsilon(d_M - d_m)(XA_d^T + Y_2^T B_d^T) \\ 0 & 0 & \tilde{R}_4 & \tilde{R}_4 & -\epsilon(d_M - d_m)Y_1^T B_1^T \\ 0 & 0 & 0 & 0 & -\epsilon(d_M - d_m)(XD_d^T + Y_1^T B_2^T) \end{bmatrix}, \\
\tilde{\Psi}_{122} &= diag\{-\tilde{P}_2 + \tilde{P}_3 - \tilde{P}_4 - \tilde{P}_5, -\tilde{P}_3 - \tilde{P}_5, -\tilde{R}_1 + \tilde{R}_2 - \tilde{R}_3 - \tilde{R}_4, \\
&\quad -\tilde{R}_2 - \tilde{R}_4, -\epsilon X - \tilde{W}_2 - \tilde{W}_2^T\}, \\
\tilde{\Psi}_{113} &= \begin{bmatrix} \tau_m XA^T & \bar{\tau}XA^T & d_m XA^T & \bar{d}XA^T & \bar{d}XA^T \\ \tau_m \tilde{A}^T & \bar{\tau}\tilde{A}^T & d_m \tilde{A}^T & \bar{d}\tilde{A}^T & \bar{d}\tilde{A}^T \\ \tau_m Y_1^T B_1^T & \bar{\tau} Y_1^T B_1^T & d_m Y_1^T B_1^T & \bar{d} Y_1^T B_1^T & \bar{d} Y_1^T B_1^T \\ \tau_m \tilde{D}_d^T & \bar{\tau}\tilde{D}_d^T & d_m \tilde{D}_d^T & \bar{d}\tilde{D}_d^T & \bar{d}\tilde{D}_d^T \end{bmatrix}, \\
\tilde{\Psi}_{133} &= diag\{\mu_1^2 \tilde{P}_4 - 2\mu_1 X, \mu_2^2 \tilde{P}_5 - 2\mu_2 X, \mu_3^2 \tilde{R}_3 - 2\mu_3 X, \\
&\quad \mu_4^2 \tilde{R}_4 - 2\mu_4 X, \bar{d}(\mu_5^2 \tilde{Q}_2 - 2\mu_5 X)\}, \\
\tilde{\Psi}_{211} &= \begin{bmatrix} \tilde{\Lambda}_{211} & \tilde{\Lambda}_{212} & \tilde{\Lambda}_{213} & \tilde{\Lambda}_{214} \\ \star & -2\tilde{P}_5 & 0 & 0 \\ \star & \star & -2\tilde{R}_4 + \lambda\tilde{\Omega} & 0 \\ \star & \star & \star & -\tilde{\Omega} \end{bmatrix},
\end{aligned}$$

$$
\begin{aligned}
\tilde{\Lambda}_{211} &= AX + XA^T + \tilde{P}_2 - \tilde{P}_4 + \tilde{R}_1 - \tilde{R}_3 - \epsilon X + \tilde{W}_1 + \tilde{W}_1^T \\
\tilde{\Lambda}_{212} &= A_d X + B_d Y_2, \\
\tilde{\Lambda}_{213} &= B_1 Y_1, \\
\tilde{\Lambda}_{214} &= D_d X + B_2 Y_1, \\
\tilde{\Psi}_{212} &= \begin{bmatrix} \tilde{P}_4 & 0 & \tilde{R}_3 & 0 & \epsilon X - \tilde{W}_1^T + \tilde{W}_2 \\ \tilde{P}_5 & \tilde{P}_5 & 0 & 0 & 0 \\ 0 & 0 & \tilde{R}_4 & \tilde{R}_4 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \\
\tilde{\Psi}_{222} &= diag\{-\tilde{P}_2 + \tilde{P}_3 - \tilde{P}_4 - \tilde{P}_5, -\tilde{P}_3 - \tilde{P}_5, -\tilde{R}_1 + \tilde{R}_2 - \tilde{R}_3 - \tilde{R}_4, \\
& \quad -\tilde{R}_2 - \tilde{R}_4, -\epsilon X - \tilde{W}_2 - \tilde{W}_2^T\}, \\
\tilde{\Psi}_{213} &= \begin{bmatrix} \tau_m XA^T & \bar{\tau} XA^T & d_m XA^T & \bar{d} XA^T & \bar{d}\tilde{W}_1^T \\ \tau_m \tilde{A}^T & \bar{\tau}\tilde{A}^T & d_m \tilde{A}^T & \bar{d}\tilde{A}^T & 0 \\ \tau_m Y_1^T B_1^T & \bar{\tau} Y_1^T B_1^T & d_m Y_1^T B_1^T & \bar{d} Y_1^T B_1^T & 0 \\ \tau_m \tilde{D}_d^T & \bar{\tau}\tilde{D}_d^T & d_m \tilde{D}_d^T & \bar{d}\tilde{D}_d^T & 0 \end{bmatrix}, \\
\tilde{\Psi}_{223} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \bar{d}\tilde{W}_2^T \end{bmatrix}, \\
\tilde{\Psi}_{233} &= diag\{\mu_1^2 \tilde{P}_4 - 2\mu_1 X, \mu_2^2 \tilde{P}_5 - 2\mu_2 X, \mu_3^2 \tilde{R}_3 - 2\mu_3 X, \\
& \quad \mu_4^2 \tilde{R}_4 - 2\mu_4 X, -\bar{d}\tilde{Q}_2\},
\end{aligned}
$$

with $\bar{\tau} = \tau_M - \tau_m, \bar{d} = d_M - d_m, \tilde{A} = A_d X + B_d Y_2, \tilde{D}_d = D_d X + B_2 Y_1$. *Then the controller gains are* $K = Y_1 X^{-1}, K_d = Y_2 X^{-1}$ *and the weighting matrix is* $\Omega = X^{-1}\tilde{\Omega}X^{-1}$.

Proof. Multiply both the left- and right- sides of (2.25) by diag$\{P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, I, I, I, I, I\}$, and both the sides of (2.26) by diag$\{P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, P_1^{-1}, I, I, I, I, P_1^{-1}\}$, respectively. Define the new variables as $X = P_1^{-1}$, $\tilde{\Omega} = X\Omega X$, $\tilde{P}_2 = XP_2X$, $\tilde{P}_3 = XP_3X$, $\tilde{P}_4 = XP_4X$, $\tilde{P}_5 = XP_5X$, $\tilde{R}_1 = XR_1X$, $\tilde{R}_2 = XR_2X$, $\tilde{R}_3 = XR_3X$, $\tilde{R}_4 = XR_4X$, $\tilde{Q}_1 = XQ_1X$, $\tilde{Q}_2 = XQ_2X$, $\tilde{W}_1 = XW_1X$, $\tilde{W}_2 = XW_2X$ and $\tilde{Q}_1 = \epsilon P_1$, with $\epsilon > 0$ is given. By the inequality as $(\mu_1 P_4 - X)^T P_4^{-1}(\mu_1 P_4 - X) \geq 0$, we then obtain (2.35) and (2.36), where the terms $-P_5$, $-R_3$, $-R_4$, $-Q_2$ are solved in the same way as in solving P_4^{-1}. The proof of Theorem 2.2 is completed □

Remark 2.3. *In Theorem 2.1 and 2.2, the parameters τ_m, τ_M, d_m, d_M, λ are given prior. The parameters τ_m, τ_M are the minimum and the maximum queue delays of the fluid-flow model; The parameters d_m, d_M are the minimum and the maximum artificial time delays of the networked system; the parameter λ is the threshold of the event-triggering scheme in the event-generator. The parameter τ_m equals to T_p, the fixed propagation delay in the network environment; the parameter τ_M can be determined by maximum queue size, the available link capacity and the fixed propagation delay; the parameter d_m equals to τ_m and the parameter d_M equals to $\tau_M + h$, which are modelled in Subsection 2.2.3; the threshold λ of the event-triggering scheme can be chosen in the range of $(0, 1)$ to determine whether the sampled signals should be released.*

In the following, we will show the merits and effectiveness of the proposed codesign methods by a numerical example.

2.5 A numerical example

In this section, an illustrative example is presented to demonstrate the effectiveness of the proposed framework. A low speed network is employed to transmit the released data. The choice of its parameters is as Table 2.1.

Table 2.1: Parameters of the communication network

TCP session number N	50
Link capacity C	5000 packets/s
Queen Delay t_0	0.18 s
Propagation delay T_p	0.1 s
Queue size q_0	900 packets
Window size W_0	20 packets
Probability of packet mark p_0	0.005

In this situation, the parameters of the fluid-flow model are as

$$\begin{aligned}\tilde{A} &= \begin{bmatrix} -0.3086 & -0.0062 \\ 277.7778 & -5.5556 \end{bmatrix}, \tilde{A}_d = \begin{bmatrix} -0.3086 & 0.0062 \\ 0 & 0 \end{bmatrix}, \tilde{B}_1 = \begin{bmatrix} -0.0324 \\ 0 \end{bmatrix}, \\ \tilde{B}_2 &= \begin{bmatrix} -0.0324 \\ 0 \end{bmatrix}, \tilde{D} = \begin{bmatrix} 0.005 & -0.005 \\ -0.5556 & 0 \end{bmatrix}, F = \begin{bmatrix} 0.0001 & 0.0005 \\ 0.0002 & 0.0008 \end{bmatrix}.\end{aligned}$$

The control system that we studied has the following parameters

$$A_p = \begin{bmatrix} 0.2562 & 0.1824 \\ 0.1443 & -0.1011 \end{bmatrix}, B_p = \begin{bmatrix} 0.7506 \\ -0.6797 \end{bmatrix}.$$

Under this network environment, suppose that the sampling period $h = 0.15s$ and $\lambda = 0.04$. By using Theorem 2.2, we obtain the system feedback controller's gain as $K = \begin{bmatrix} -10.1726 & 0.1798 \end{bmatrix}$. The congestion controller gains are $\tilde{K}_1 = \begin{bmatrix} -1.5138 & -0.6639 \end{bmatrix}$, $\tilde{K}_2 = \begin{bmatrix} -8.5443 & 0.1539 \end{bmatrix}$. And the triggering matrix is

$$\Omega = \begin{bmatrix} 16.7481 & -8.3653 & -0.0000 & 0.0021 \\ -8.3653 & 12.2868 & 0.0000 & -0.0040 \\ -0.0000 & 0.0000 & 0.0127 & 0.1765 \\ 0.0021 & -0.0040 & 0.1765 & 17.7985 \end{bmatrix}.$$

The state responses of the control system and the control input held in the logic ZOH are shown in Figure 2.4 and Figure 2.5, respectively.

The scheduling of the sampled data is shown in Figure 2.6 within a simulation time of $T = 30$ s.

It is obvious that under this new scheme, the sampled data is efficiently scheduled by the event-generator based on network dynamics. To be specific, as shown in Figure 2.6 and Figure 2.7, 51.72% communication tasks executed in the first $5s$ which accounts for 7.5% of the total sampled data. Few sampled data is transmitted after $5s$ due to the asymptotical stability of the control system. As can be seen from Figure 2.7, when the ability to transmit is increased in the fluid-flow network model, more sampled data are packaged and triggered by the event-generator and the corresponding control performance is guaranteed. Compared with the period of the simulation time from 0 s to 5 s, the frequency of sampled data packaged and transmitted in the fluid-flow network model after $5s$ is smaller because the system

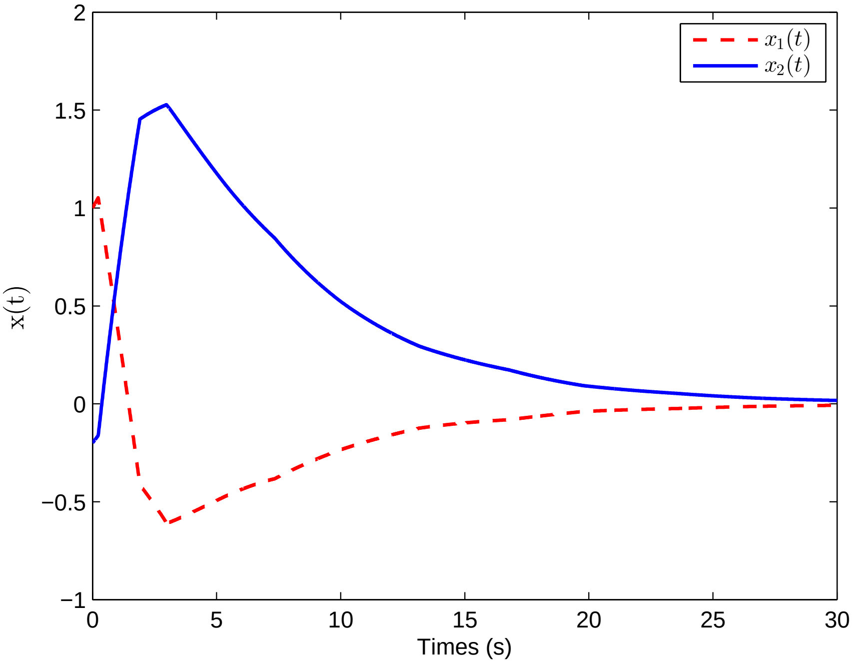

Figure 2.4: The state responses of the control system

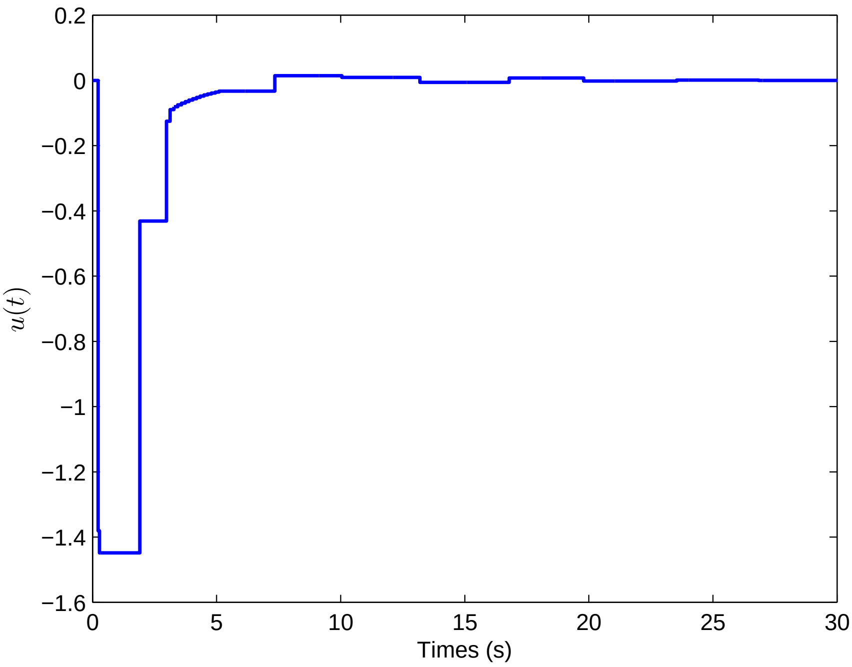

Figure 2.5: The control input of the networked system

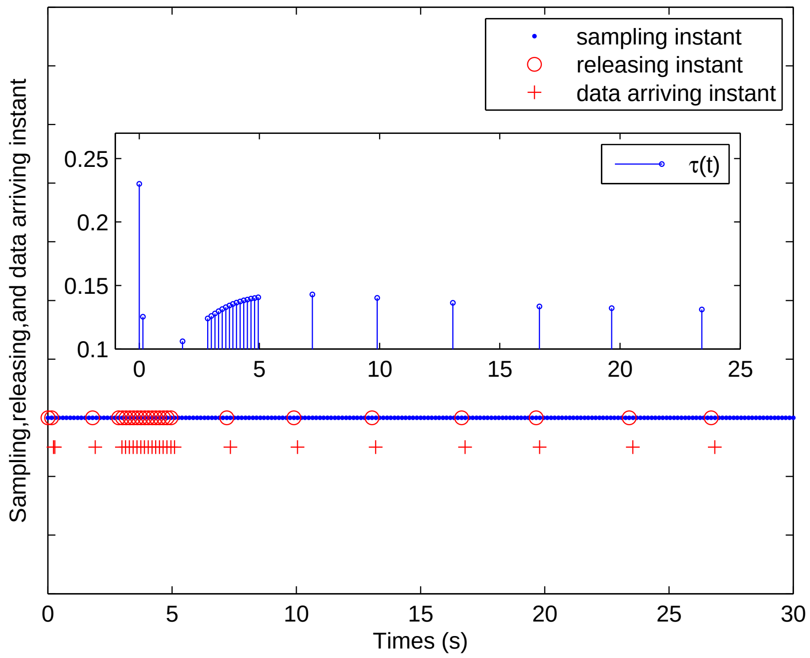

Figure 2.6: Sampling, releasing, and data arriving instants

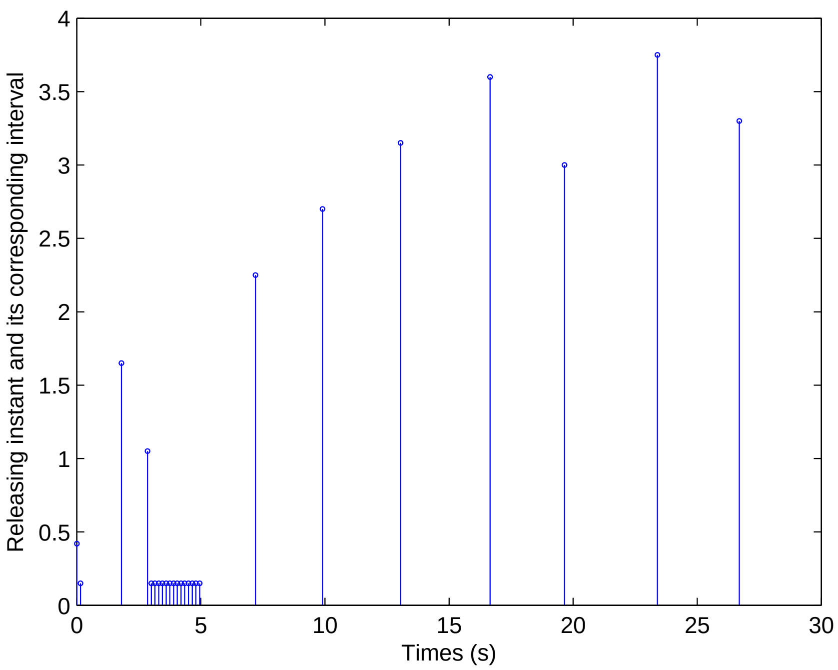

Figure 2.7: Releasing instants and the corresponding intervals

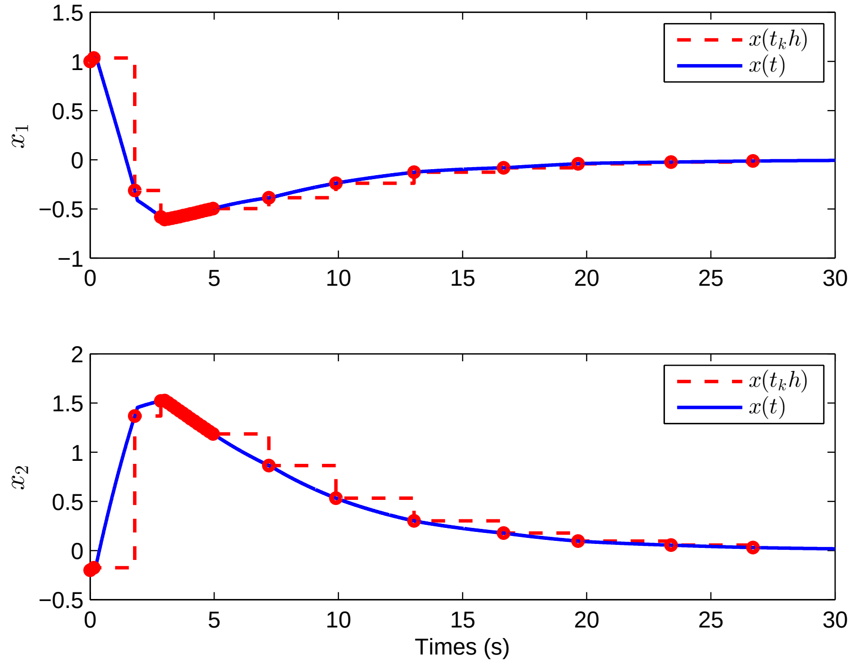

Figure 2.8: The state responses of the networked system with released packages

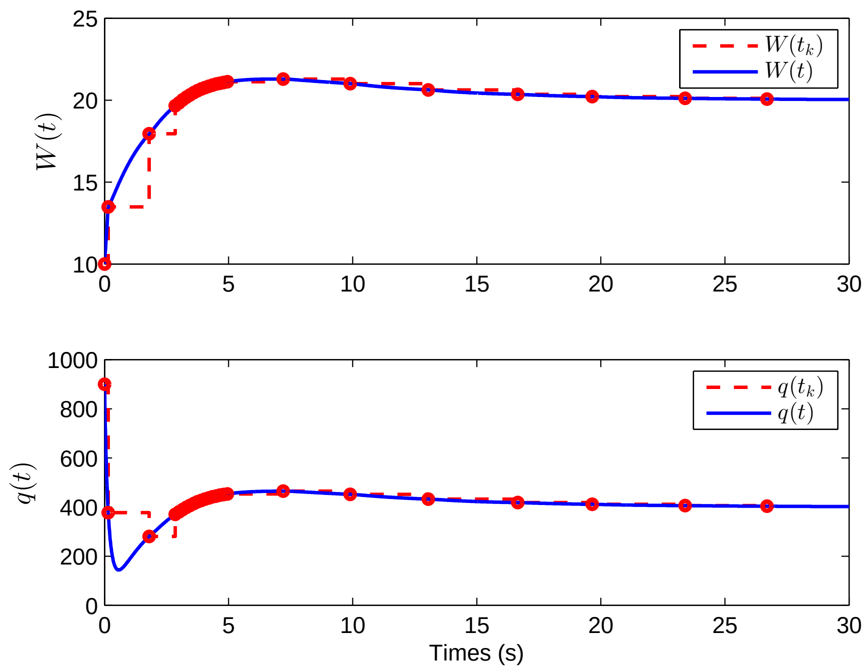

Figure 2.9: The state responses of the FFM with released instants

is approaching the equilibrium point. In total, less communication resources are required; in other words, the limited network resources are saved.

Therefore, from this illustrative example, we can conclude that the sampled data is packaged and transmitted by the event-generator not only based on the performance of the networked system, but also on the network dynamics.

2.6 Conclusion

This chapter has addressed the stability and stabilisation of networked systems based on network dynamics. A novel event-triggered scheme which provides a condition to determine which sampled data should be transmitted is proposed. Based on fluid-flow model indicating the dynamics of the communication network, a new framework is proposed. The stability criterion and the controller gain design method have been obtained, taking into account network-induced time-varying delays. The criterion builds the relationship between the performance of the control system and network dynamics. Based on this criterion, the resources of the network are utilised efficiently without congestion while the performance of the control system is ensured. Simulation results have demonstrated the effectiveness of the proposed method.

Chapter 3

H_∞ filter design for networked systems based on network dynamics

3.1 Introduction

Focussing on a research problem described in Chapter 1, this chapter aims to investigate the H_∞ filter design for networked systems based on network dynamics. H_∞ filtering is an important issue in the fields of signal processing and system control. The main feature of H_∞ filtering is that, compared with the famous Kalman filtering [35], it does not require that both the system model under study and the statistical properties of external noises are known exactly. Due to its theoretical and practical significance, H_∞ filtering has been intensively studied in the last ten years, see e.g. [117], [118], [119], [120] and the references therein.

In recent years, with the rapid development of digital and intelligent technologies, H_∞ filtering for networked systems has gained increasing attention [121], [122]. In the networked system framework, the communication network is introduced to transmit data packets from a physical plant to a filter. Due to the imperfection of the communication network and the limitation of network resources, the output of the plant is not exactly equal to the input of the filter. Moreover, most of the available results are based on time-triggered schemes where all the sampled data is sent through a communication network with a fixed period. Under those schemes, the system output is first sampled with a constant period and then is encapsulated with its time stamp and transmitted over the communication network to the filter. By using time-triggering schemes to design H_∞ filters, the sampling period is determined under the worst case operating conditions, which leads to inefficient utilisation of the precious network resources.

Recently, event-triggered schemes have been proposed as a means to reduce the burden on communication networks, see e.g. [123], [124], [125], [126] and the references therein. Among those results, most of the event-triggering conditions are based on the system's behaviour while few results consider the dynamics of communication networks which reflect the level of the quality of service (QoS). It is obvious that the main purpose of those kinds of event-triggered schemes is to reduce the demand on usage of network resources. However, if the saved network resources are not reused, those kinds of event-triggered schemes will lead to an idle network, which constitutes a wastage of network resources. To overcome this drawback, it is necessary to design an event-triggering scheme which takes network dynamics into account so as to adapt to different levels of the QoS of the communication network. For example, to investigate the H_∞ filtering issue, when the communication channel is idle, more sampled data can be released to the network to be transmitted to the H_∞ filters. Then the network resources will be utilised more efficiently and better performance may be achieved.

The consideration of network dynamics in the modelling of H_∞ filtering for networked systems inevitably enhances the complexity in the study. To deal with this issue, in this chapter, a novel framework is developed for the networked system. More specifically, the implementation of such a framework involves at least the following two aspects:

1. How to abstract the main characteristics of the communication network to

codesign the event-triggering scheme with system dynamic behaviour such that network resources can be dynamically allocated to transmit the released sampled data;

2. How to design an online scheduling strategy for the filter to adapt the network dynamics in order to achieve better H_∞ performance.

In this chapter, we develop a novel framework with an information scheduling middleware to investigate the event-triggered H_∞ filtering issue for networked systems. Both the networked system and network dynamics are abstracted and modelled within the Information Dispatching Middleware, which reduce the burden on the system engineers to simplify and shorten the codesign development of a networked system with network dynamics. In the middleware, two modules namely the Information Selection Module and the Congestion Avoidance Module are developed. The Information Selection Module aims to regulate the transmission of the sampled data in terms of a predefined event-triggering condition. The Congestion Avoidance Module is used to schedule those sampled data released by the Information Selection Module to the filter. With the proposed middleware, a tradeoff between the desired H_∞ filtering performance and communication network resources utilisation is obtained. Moreover, a linear estimation method is derived to compensate network-induced influence on the released signals. The online scheduling strategy is designed based on network dynamics with different levels of the H_∞ filtering performance to adapt the different levels of network QoS. Furthermore, based on the proposed framework, the filtering error system is modelled as a linear system with two time-varying delays. By employing a Lyapunov-Krasovskii functional, a sufficient condition such that the filtering error system is asymptotically stable with a prescribed H_∞ level, is derived. As a result, the H_∞ filter, the congestion controllers and the event-triggered scheme can be codesigned in terms of solutions to a set of linear matrix inequalities (LMIs). Finally, the effectiveness of the proposed method is illustrated by a mechanical system with two masses and two springs.

This chapter is organised into six sections. After a brief introduction, Section 3.2 proposes a novel middleware framework for networked system to analyse and synthesize the filtering issue. In Section 3.3, H_∞ performance of the networked system is analysed and the H_∞ filter is designed in Section 3.4. Simulation results are presented in Section 3.5 and the chapter is recapitulated in Section 3.6.

3.2 Modelling and problem formulation

3.2.1 A novel H_∞ filtering framework for a networked system

Consider the continuous-time linear system described by

$$\begin{cases} \dot{x}_p(t) = A_p x_p(t) + B_p w(t), \;\; x_p(0) = x_0 \\ y_p(t) = C_p x_p(t) \\ z_p(t) = L_p x_p(t) \end{cases} \tag{3.1}$$

where $x_p(t) \in \mathbb{R}^n$ is the system state vector; $y_p(t) \in \mathbb{R}^m$ is the measurement output vector; and $z_p(t) \in \mathbb{R}^r$ is the signal to be estimated. The input signal $w(t) \in \mathbb{R}^l$ belongs to $\mathcal{L}_2[0, +\infty)$. The system matrices A_p, B_p, C_p and L_p are constant matrices with appropriate dimensions.

The novel H_∞ filtering framework for the networked system with network dynamics is established by introducing an Information Dispatching Middleware, which is illustrated in Figure 3.1. To achieve a tradeoff between the desired performance of the H_∞ filtering and the utilisation of communication network resources, a codesign method is proposed to fill the "interaction gap" between the QoS of the communication network and the QoP of the filtering design.

In the proposed framework, the plant is described in (3.1); the sensor and the sampler are clock-driven; the smart zero-order-holder (S-ZOH) and the filter are event-driven. The released packets including "needed" measurement output signals of the plant, time stamp and the state of network dynamics, are scheduled by the

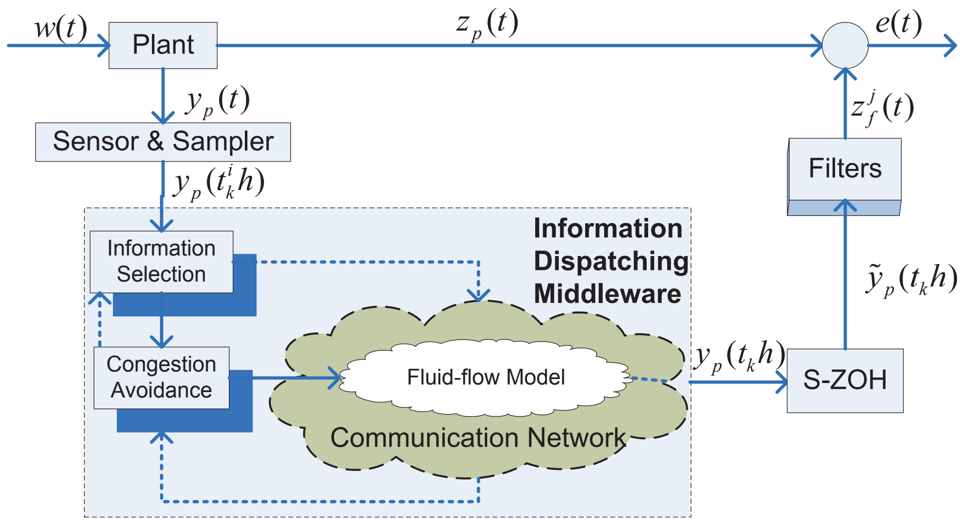

Figure 3.1: A novel framework with the Information Dispatching Middleware

Information Dispatching Middleware. In this middleware, there are two modules:

- The Information Selection Module with an event-triggering scheme;
- The Congestion Avoidance Module generated from a fluid-flow model under equilibrium points based on TCP/AQM communication networks.

Scheduled by the middleware, the released packets are transmitted to the S-ZOH, which has a "hook" to detect the time-delay in order to design a suitable filter to estimate the controlled output signal $z_p(t)$. Network time synchronisation is employed to sample the output of the networked system and the state of the communication network synchronously.

3.2.2 Network dynamics in the Congestion Avoidance Module

In this section, we will analyse the network dynamics in the Congestion Avoidance Module of the Information Dispatching Middleware in the proposed framework. As discussed in Chapter 2, a nonlinear model without timeout mechanism for a TCP network is developed by using a fluid-flow and stochastic differential equation analysis approach [102]. The model is specified as

$$
\begin{cases}
\dot{W}(t) = \dfrac{1}{\tau(t)} - \dfrac{W(t)}{2}\dfrac{W(t-\tau(t))}{\tau(t-\tau(t))}p(t-\tau(t)) \\
\dot{q}(t) = \begin{cases} -C(t) + \dfrac{N(t)}{\tau(t)}W(t), q(t) > 0 \\ \max\{0, -C(t) + \dfrac{N(t)}{\tau(t)}W(t)\}, q(t) = 0, \end{cases} \\
\tau(t) = \dfrac{q(t)}{C(t)} + T_p,
\end{cases}
\tag{3.2}
$$

where $W(t)$ is the TCP window size; $q(t)$ is the queue length; T_p is the propagation delay; $\tau(t)$ is the information time delay; $C(t)$ is the available link capacity; $N(t)$ is the number of TCP sessions; and $p(t)$ ($0 \leq p(t) \leq 1$) is the packet-dropping probability function, which is the control input used to maintain the bottleneck queue.

In this chapter, we consider the estimation of released data from the Information Selection Module based on the effects of time-varying delays induced from the fluid-flow model.

It is assumed that the information delay $\tau(t)$ is under a single router. Take $(W(t), q(t))$ as the network state reflecting network dynamical characteristics. Based on the desired H_∞ performance, for a chosen triplet of network parameters (N, C_0, τ_0), a triple (W_0, q_0, p_0) in the set $\Upsilon = \{(W_0, q_0, p_0) : \tau_0 = \frac{q_0}{C_0} + T_p, W_0 = \frac{\tau_0 C_0}{N}, p_0 = \frac{2}{W_0^2}\}$ is an equilibrium point. Define $\delta W = W(t) - W_0$, $\delta q = q(t) - q_0$, $\delta p = p(t) - p_0$, $\delta C = C(t) - C_0$. Then linearise the equation (3.2) at the equilibrium point such that the nonlinear model could be expressed in the form of the following linear time-delay model [79]

$$\begin{cases} \delta\dot{W}(t) = -\frac{N}{\tau_0^2 C_0}(\delta W(t) + \delta W(t-\tau(t))) \\ -\frac{1}{\tau_0^2 C_0}(\delta q(t) - \delta q(t-\tau(t))) - \frac{\tau_0 C_0^2}{2N^2}\delta p(t-\tau(t)) \\ +\frac{\tau_0 - T_p}{\tau_0^2 C_0}(\delta C(t) - \delta C(t-\tau(t))), \\ \delta\dot{q}(t) = \frac{N}{\tau_0}\delta W(t) - \frac{1}{\tau_0}\delta q(t) - \frac{T_p}{\tau_0}\delta C(t) \end{cases} \tag{3.3}$$

Set $\tilde{x}(t) = [\delta W(t)\ \delta q(t)]^T$, $\tilde{u}_1(t) = \delta p(t)$. Then a generalised fluid-flow model of TCP/AQM communication networks for the H_∞ issue of networked systems can be described as

$$\begin{cases} \dot{\tilde{x}}(t) &= \tilde{A}\tilde{x}(t) + \tilde{A}_d\tilde{x}(t-\tau(t)) + \tilde{B}_1\tilde{u}_1(t-\tau(t)) \\ &+ \tilde{B}_2\tilde{u}_2(t) + \tilde{D}\tilde{v}(t) \\ \tilde{y}(t) &= \tilde{f}(y_p(t_k h)) \end{cases} \tag{3.4}$$

where $\tilde{x}(t) \in \mathbb{R}^2$, $\tilde{u}_1(t) \in \mathbb{R}^1$, $\tilde{u}_2(t) \in \mathbb{R}^1$, $v(t) \in \mathbb{R}^2$ represent the internal dynamics of the network, the internal control input for the network, the external control strategy for the network and the external disturbance of the network, respectively. $\tilde{f}(\cdot)$ denotes the network effects on the released signal $y_p(t_k h)$ at the released instant $t_k h$, with

$$\begin{aligned} \tilde{A} &= \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & -\frac{1}{\tau_0^2 C_0} \\ \frac{N}{\tau_0} & -\frac{1}{\tau_0} \end{bmatrix}, \tilde{A}_d = \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & \frac{1}{\tau_0^2 C_0} \\ 0 & 0 \end{bmatrix}, \\ \tilde{B}_1 &= \tilde{B}_2 = \begin{bmatrix} -\frac{\tau_0^2 C_0}{2N^2} \\ 0 \end{bmatrix}, \tilde{D} = \begin{bmatrix} \frac{\tau_0 - T_p}{\tau_0^2 C_0} & -\frac{\tau_0 - T_p}{\tau_0^2 C_0} \\ -\frac{T_p}{\tau_0} & 0 \end{bmatrix}. \end{aligned}$$

The packets, including the "needed" sampled data to be packaged and released from the event-generator in the Information Selection Module, will be scheduled by the Congestion Avoidance Module of the Information Dispatching Middleware.

3.2.3 An event-triggered scheme in the Information Selection Module

In the proposed middleware, the event-generator is a major part of the Information Selection Module. It receives the sampled measurement output data $y_p(sh)$ and the network state $\tilde{x}(sh)$ with a constant period $h > 0$, under a periodic sampling mechanism and a network time synchronisation. The set of sampled instants can be represented by $\{sh|s \in \mathbb{N}\}$. The set of released instants $\{t_k h|k \in \mathbb{N}\}$ is a subset of the sampled instants and define $t_k^i h = t_k h + ih$ as the corresponding following sampled instants for the released instant $t_k h$, with $i = 0, \ldots, n$, $n = t_{k+1} - t_k$ when $i = 0$, $t_k^i h = t_k h$; $i = n$, $t_k^i h = t_{k+1} h$.

Considering the effects of the communication network on the transmitted measurement output $y_p(t_k h)$, it is assumed that the communication network introduces a communication delay $\tau(t_k h) = \frac{q(t_k h)}{C_0} + T_p$, in which the link capacity C_0 is fixed with $\tau_m \leq \tau(t_k h) \leq \tau_M$, $k \in \mathbb{N}$. τ_m is the fixed propagation delay T_p; τ_M is the maximum allowable transmission network-induced delay.

The measurement output error $e_k(t_k^i h)$ of the plant between the latest transmitted output sampled data $y_p(t_k h)$ and the current sampled data $y_p(t_k^i h)$ can be calculated as

$$e_k(t_k^i h) = y_p(t_k h) - y_p(t_k^i h). \tag{3.5}$$

The S-ZOH keeps the data received at $t_k h + \tau(t_k h)$ available until the new data arrives at $t_{k+1} h + \tau(t_{k+1} h)$. The holding zone of the S-ZOH is $[t_k h + \tau(t_k h), t_{k+1} h + \tau(t_{k+1} h)) = \bigcup_{i=0}^{n-1} \Theta_k^i$, where $\Theta_k^i = [t_k^i h + \tau(t_k^i h), t_k^{i+1} h + \tau(t_k^{i+1} h))$. By introducing an artificial time delay $d(t) = t - t_k^i h$, $t \in \Theta_k^i$, we obtain

$$y_p(t_k h) = y_p(t - d(t)) + e_k(t_k^i h) \tag{3.6}$$

where $d(t)$ is piecewise-linear with the derivative $\dot{d}(t) = 1$.

To keep the right order for the $y_p(t_k^i h)$, in this chapter, we assume that $d_m \leq d(t) \leq d_M < \infty$, for $t \neq t_k^i h + \tau(t_k^i h)$ with $d_m = \min\{\tau(t_k h)\}$, $d_M = h + \max\{\tau(t_k h)\}$.

In the Information Selection Module, at the sampling instant $t_k^i h$, taking the network dynamics $\tilde{x}(t_k^i h)$, the measurement output $y(t_k^i h)$ and the dynamic priority evaluation function $\psi(t_k^i h) = \psi(e_k(t_k^i h), \tilde{x}(t_k^i h))$ into account, we define a judgement function $f(t_k^i h)$ to select the sampled measurement output $y_p(t_k^i h)$ as

$$f(t_k^i h) = \begin{bmatrix} e_k(t_k^i h) \\ \psi(t_k^i h) \end{bmatrix}^T \Omega \begin{bmatrix} e_k(t_k^i h) \\ \psi(t_k^i h) \end{bmatrix} - \lambda \begin{bmatrix} y_p(t_k^i h) \\ \tilde{x}(t_k^i h) \end{bmatrix}^T \Omega \begin{bmatrix} y_p(t_k^i h) \\ \tilde{x}(t_k^i h) \end{bmatrix}$$

where λ is a threshold and Ω is a positive definite weighting matrix, which can be codesigned based on network dynamics.

The judgement function $f(t_k^i h)$ determines whether or not the sampled measurement output $y_p(t_k^i h)$ should be packaged as $(SN, t_{k+1}h, y_p(t_{k+1}h), \tilde{x}(t_{k+1}h))$ and transmitted through the communication network. $t_0 h$ is the initial released instant and the next transmission instant of the $t_k h$ can be expressed as

$$t_{k+1}h = t_k h + \inf_{i \geq 1}\{ih | f(t_k^i h) \geq 0\}. \tag{3.7}$$

3.2.4 An online scheduling strategy and an H_∞ filtering error model

Based on the previous analysis, an online scheduling mechanism will be developed for the H_∞ filtering of the networked system. In order to reduce the negative effects of the network-induced delay on the released sampled data, an online delay-evaluation approach is proposed to compensate for the network-induced delay. Considering the effects of the time-varying delays induced from the fluid-flow model and inspired by [127], [128], [129], the estimation of the released data in (3.4) is formulated as

$$\tilde{y}(t) = \tilde{l}(\tau(t_k h))y_p(t_k h) \tag{3.8}$$

where $t \in [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$ and the parameter $\tilde{l}(\tau(t_k h))$ is a function of $\tau(t_k h)$. Generally, $\tilde{l}(\tau(t_k h))$ should be worked out online to compensate for the network-induced delay $\tau(t_k h)$. However, the processing power in the nodes is limited and time required for calculating $\tilde{l}(\tau(t_k h))$ is intolerable. In this context, $\tilde{y}(t)$ will be discretised as

$$\tilde{y}(t) = \begin{cases} (1 - \frac{1}{\tau_M}(\frac{\tau_0+\tau_1}{2} - \tau_m))y_p(t_k h), \tau_0 \leq \tau(t_k h) < \tau_1 \\ (1 - \frac{1}{\tau_M}(\frac{\tau_1+\tau_2}{2} - \tau_m))y_p(t_k h), \tau_1 \leq \tau(t_k h) < \tau_2 \\ \cdots \\ (1 - \frac{1}{\tau_M}(\frac{\tau_{N-1}+\tau_N}{2} - \tau_m))y_p(t_k h), \tau_{N-1} \leq \tau(t_k h) < \tau_N \end{cases} \tag{3.9}$$

where $\tau_0 = \tau_m$, $\tau_N = \tau_M$.

An online scheduling mechanism is developed in the proposed framework to adapt the dynamic behaviors of the communication network in order to achieve a better H_∞ performance. There are two steps to schedule the sampled data in the Information Dispatching Middleware and in S-ZOH.

In the proposed IDM, the Information Selection Module receives the sampled data from the networked system and from the communication network periodically and simultaneously. According to the network dynamics, the event-generator chooses a predefined threshold λ and a positive definite weighting matrix Ω to determine whether or not the sampled data $y_p(t_k^i h)$ is triggered and encapsulated into a packet with sequence number SN, time stamp $t_{k+1}h$ and the network state $\tilde{x}(t_{k+1}h)$ as $(SN, t_{k+1}h, y(t_{k+1}h), \tilde{x}(t_{k+1}h))$ to be transmitted through the communication channel.

Once the S-ZOH receives the released packet $(SN, t_k h, y(t_k h), \tilde{x}(t_k h))$, an online scheduling strategy is designed to choose a suitable filter such that the prescribed filtering performance can be satisfied. The online scheduling algorithm can be concisely described as in Algorithm 1.

Algorithm 1 Online scheduling strategy In S-ZOH.

1: Unpack the received packets $(SN, t_k h, y(t_k h), \tilde{x}(t_k h))$;
2: Detect the network dynamic $\tilde{x}(t_k h)$ by a 'hook' in S-ZOH;
3: Calculate $\tau(t_k h)$, $\tilde{y}(t_k h)$;
4: **switch** $\tau(t_k h)$ **do**
5:     **case** $\tau(t_k h) \in [\tau_0, \tau_1)$ S-ZOH switches to Filter 1;
6:     **case** $\tau(t_k h) \in [\tau_1, \tau_2)$ S-ZOH switches to Filter 2;
7:     $\cdots$
8:     **case** $\tau(t_k h) \in [\tau_{j-1}, \tau_j)$ S-ZOH switches to Filter j;
9:     $\cdots$
10:     **case** $\tau(t_k h) \in [\tau_{\varrho-1}, \tau_\varrho)$ S-ZOH switches to Filter ϱ;
11: Wait for the next arrived packet, then jump to 1.

Define $\tilde{l}(j) = 1 - \frac{1}{\tau_M}(\frac{\tau_{j-1}+\tau_j}{2} - \tau_m)$, with $j = 1, 2, ..., N$. Then $\tilde{y}(t) = \tilde{l}(j) y_p(t_k h)$.

The filter to be designed is a stable and full order linear dynamic filter in the form of

$$\begin{cases} \dot{x}_f(t)=A_f x_f + B_f \tilde{y}(t), \ \ x_f(0)=0 \\ z_f(t)=C_f x_f(t) \end{cases} \tag{3.10}$$

where A_f, B_f, and C_f are filter parameters to be designed based on the network dynamics.

Denote

$$\begin{aligned} e(t) &= z_p(t) - z_f(t), \\ \eta(t) &= \mathrm{col}\{x_p(t), \tilde{x}(t)\}, \\ \xi(t) &= \mathrm{col}\{\eta(t), x_f(t)\}, \\ \bar{e}(t_k^i h) &= \mathrm{col}\{e_k(t_k^i h), \psi(t_k^i h)\}. \end{aligned}$$

Combining (3.1) and (3.4), with $\tilde{u}_1(t) = \tilde{K}_1\tilde{x}(t-\tau(t))$ and the external control strategy for the fluid-flow model $\tilde{u}_2(t) = \tilde{K}_2\psi(t_k^i h)$, the filtering error system can be expressed as

$$\begin{cases} \dot{\xi}(t) &= \bar{\bar{A}}\xi(t) + \bar{\bar{A}}_d\eta(t-\tau(t)) \\ & \quad +\bar{\bar{B}}_f\eta(t-d(t)) + \bar{\bar{D}}\bar{e}(t_k^i h) + \bar{\bar{B}}w(t) \\ e(t) &= L\xi(t) \\ \xi(\theta) &= \mathrm{col}\{x_0, \tilde{x}_0, 0\}, \ \ \theta \in [-d_M, 0], \end{cases} \tag{3.11}$$

for $t \in [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$, with $\tau_{j-1} \le \tau(t_k h) \le \tau_j$, $j = 1, \cdots, N$, where

$$\begin{aligned} &\bar{\bar{A}} = \mathrm{diag}\{\bar{A}, A_f\}, \bar{A} = \mathrm{diag}\{A_p, \tilde{A}\}, \bar{\bar{A}}_d = \mathrm{col}\{\bar{A}_d, 0\}, L = \begin{bmatrix} L_p E & -C_f \end{bmatrix}, \\ &\bar{A}_d = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{A}_d + \tilde{B}_1\tilde{K}_1 \end{bmatrix}, \bar{\bar{B}}_f = \begin{bmatrix} 0 \\ B_f\tilde{l}(j)C_pE \end{bmatrix}, \bar{\bar{B}} = \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix}, \bar{B} = \begin{bmatrix} B_p \\ 0 \end{bmatrix} \\ &E = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \end{bmatrix}, \bar{\bar{D}} = \mathrm{col}\{\bar{D}, B_f\bar{L}\}, \bar{D} = \begin{bmatrix} 0 & 0 \\ \tilde{D}\tilde{F} & \tilde{B}_2\tilde{K}_2 \end{bmatrix}, \bar{L} = \begin{bmatrix} \tilde{l}(j) & 0 \end{bmatrix}. \end{aligned}$$

The event-triggered H_∞ filtering problem can then be stated as: Given prescribed levels of disturbance attenuation $\gamma_j > 0, j = 1, 2, \cdots, N$, determine the weighting matrix Ω_j and the parameters A_f, B_f and C_f of the filters in the form of (3.10), such that

1) the filter error system (3.11) with $w(t) = 0$ is asymptotical stable; and

2) under zero initial condition, the filter error system (3.11) achieves an H_∞ performance level γ_j, i.e. the filtering error $e(t)$ satisfies $\|e(t)\|_2 \le \gamma_j \|w(t)\|_2$, for any nonzero $w(t) \in \mathcal{L}_2[0,\infty)$.

To proceed with, we first introduce three lemmas, which are useful in solving the above problem.

Lemma 3.1. *[130] Given any real matrices X, Y and Z with appropriate dimensions and such that $Y > 0$ and is symmetric, we have*

$$X^T Y X + X^T Z + Z^T X + Z^T Y^{-1} Z \ge 0. \tag{3.12}$$

Lemma 3.2. *[131] For a given matrix $R > 0$, the following inequality holds for any continuously differentiable function $w : [a, b] \to \mathbb{R}^n$*

$$\int_a^b \dot{w}^T(u) R \dot{w}(u) du \ge \frac{1}{b-a}(\tilde{\Gamma}_1^T R \tilde{\Gamma}_1 + 3\tilde{\Gamma}_2^T R \tilde{\Gamma}_2) \tag{3.13}$$

where $\tilde{\Gamma}_1 = w(b) - w(a)$ and

$$\tilde{\Gamma}_2 = w(b) + w(a) - \frac{2}{b-a}\int_a^b w(u)du.$$

Lemma 3.3. *[132] Let $\tau(t)$ be a continuous function satisfying $0 \le h_1 \le \tau(t) \le h_2$. For any $n \times n$ real matrix $R_1 > 0$ and a vector $\dot{x} : [-h_2,\ 0] \to \mathbb{R}^\ltimes$ such that the integration concerned below is well defined, the following inequality holds for any $2n \times 2n$ real matrix S_1 satisfying $\begin{bmatrix} \tilde{R}_1 & S_1 \\ S_1^T & \tilde{R}_1 \end{bmatrix} \ge 0$*

$$\begin{aligned} &-(h_2 - h_1)\int_{t-h_2}^{t-h_1} \dot{x}^T(s) R_1 \dot{x}(s) ds \\ &\le 2\psi_{11}^T S_1 \psi_{21} - \psi_{11}^T \tilde{R}_1 \psi_{11} - \psi_{21}^T \tilde{R}_1 \psi_{21} \end{aligned} \tag{3.14}$$

where $\tilde{R}_1 := diag\{R_1, 3R_1\}$; and

$$\begin{cases} \psi_{11} = \begin{bmatrix} x(t-\tau(t)) - x(t-h_2) \\ x(t-\tau(t)) + x(t-h_2) - 2v_2(t) \end{bmatrix} \\ \psi_{21} = \begin{bmatrix} x(t-h_1) - x(t-\tau(t)) \\ x(t-h_1) - x(t-\tau(t)) - 2v_1(t) \end{bmatrix} \end{cases}$$

with

$$\begin{cases} v_1(t) = \dfrac{1}{\tau(t)-h_1}\displaystyle\int_{t-\tau(t)}^{t-h_1} x(s)ds \\ v_2(t) = \dfrac{1}{h_2-\tau(t)}\displaystyle\int_{t-h_2}^{t-\tau(t)} x(s)ds. \end{cases}$$

3.3 H_∞ filtering performance analysis

In this section, let the filter parameters in (3.10) and the congestion controllers in (3.4) first be given. Then, based on a Lyapunov-Krasovskii functional, a sufficient condition for H_∞ performance analysis of the filtering error system (3.11) is given by the following theorem.

Theorem 3.1. *For given A_f, B_f, C_f, $\tilde{K}_1$, $\tilde{K}_2$ and scalars τ_{j-1}, τ_j, $(j = 1, \cdots, N)$, d_m, d_M, $\lambda \in (0,\ 1)$ and $\gamma_j > 0$, the filtering error system (3.11) is asymptotically stable with an H_∞ performance γ_j, if there exist matrices $\Omega_j > 0$, $P = \left[\begin{smallmatrix} P_1 & E^T P_2 \\ \star & P_3 \end{smallmatrix}\right] > 0$, $R_i > 0$, $Q_i > 0$ $(i = 1,2,3,4)$, $S_1 = \left[\begin{smallmatrix} S_{11} & S_{12} \\ S_{13} & S_{14} \end{smallmatrix}\right] > 0$, $S_2 = \left[\begin{smallmatrix} S_{21} & S_{22} \\ S_{23} & S_{24} \end{smallmatrix}\right] > 0$, with appropriate dimensions such that*

$$\begin{bmatrix} \tilde{Q}_4 & S_1 \\ \star & \tilde{Q}_4 \end{bmatrix} \geq 0, \quad \begin{bmatrix} \tilde{R}_4 & S_2 \\ \star & \tilde{R}_4 \end{bmatrix} \geq 0 \tag{3.15}$$

$$\begin{bmatrix} \Psi & \tau_{j-1}\Gamma Q_3 & (\tau_j - \tau_{j-1})\Gamma Q_4 & d_m \Gamma R_3 & (d_M - d_m)\Gamma R_4 \\ \star & -Q_3 & 0 & 0 & 0 \\ \star & \star & -Q_4 & 0 & 0 \\ \star & \star & \star & -R_3 & 0 \\ \star & \star & \star & \star & -R_4 \end{bmatrix} < 0 \tag{3.16}$$

where $\tilde{Q}_4 = diag\{Q_4, 3Q_4\}$, $\tilde{R}_4 = diag\{R_4, 3R_4\}$ and

$$\Gamma = col\{\bar{A}^T, 0, 0, \bar{A}_d^T, 0, 0, 0, 0, \bar{D}^T, \bar{B}^T, 0, 0, 0, 0, 0, 0, \}$$

$$\Psi = \begin{bmatrix} \Psi_{11} & \Psi_{12} & \Psi_{13} & \Psi_{14} & 0 \\ \star & \Psi_{22} & 0 & \Psi_{24} & \Psi_{25} \\ \star & \star & \Psi_{33} & \Psi_{34} & \Psi_{35} \\ \star & \star & \star & \Psi_{44} & 0 \\ \star & \star & \star & \star & \Psi_{55} \end{bmatrix} \tag{3.17}$$

with

$$\Psi_{11} = \begin{bmatrix} \Lambda_{11} & E^T P_2 A_f + \bar{A}^T E^T P_2 - E^T L_p^T C_f \\ \star & P_3 A_f + A_f^T P_3^T + C_f^T C_f \end{bmatrix},$$

$$\Psi_{12} = \begin{bmatrix} -2Q_3 & P_1\bar{A}_d & 0 \\ 0 & P_2^T E\bar{A}_d & 0 \end{bmatrix},$$
$$\Psi_{13} = \begin{bmatrix} -2R_3 & E^T P_2 B_f \tilde{l} C_p E & 0 \\ 0 & P_3 B_f \tilde{l} C_p E & 0 \end{bmatrix},$$
$$\Psi_{14} = \begin{bmatrix} P_1\bar{D} + E^T P_2 B_f \bar{L} & P_1\bar{B} & 6Q_3 & 6R_3 \\ P_2^T E\bar{D} + P_3 B_f \bar{L} & P_2^T E\bar{B} & 0 & 0 \end{bmatrix},$$
$$\Psi_{22} = \begin{bmatrix} \Lambda_{33} & \Lambda_{34} & \Lambda_{35} \\ \star & \Lambda_{44} & \Lambda_{45} \\ \star & \star & -Q_2 - 4Q_4 \end{bmatrix},$$
$$\Psi_{24} = \begin{bmatrix} 0 & 0 & 6Q_3 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$
$$\Psi_{25} = \begin{bmatrix} -2S_{13}^T - 2S_{14}^T + 6Q_4 & 0 & 0 & 0 \\ 2S_{13}^T - 2S_{14}^T + 6Q_4 & -2S_{12} - 2S_{14} + 6Q_4 & 0 & 0 \\ 0 & 2S_{12} - 2S_{14} + 6Q_4 & 0 & 0 \end{bmatrix},$$
$$\Psi_{33} = \begin{bmatrix} \Lambda_{66} & \Lambda_{67} & \Lambda_{68} \\ \star & \Lambda_{77} & \Lambda_{78} \\ \star & \star & -R_2 - 4R_4 \end{bmatrix},$$
$$\Psi_{34} = \begin{bmatrix} 0 & 0 & 0 & 6R_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$
$$\Psi_{35} = \begin{bmatrix} 0 & 0 & -2S_{23}^T - 2S_{24}^T & 6R_4 \\ 0 & 0 & 2S_{23}^T - 2S_{24}^T + 6R_4 & -2S_{22} - 2S_{24} + 6R_4 \\ 0 & 0 & 6R_4 & 2S_{22} - 2S_{24} \end{bmatrix},$$
$$\Psi_{44} = diag\{-\Omega_j, -\gamma_j^2 I, -12Q_3, -12R_3\},$$
$$\Psi_{55} = \begin{bmatrix} -12Q_4 & 4S_{14} & 0 & 0 \\ \star & -12Q_4 & 0 & 0 \\ \star & \star & -12R_4 & 4S_{24} \\ \star & \star & \star & -12R_4 \end{bmatrix},$$
$$\Lambda_{11} = P_1\bar{A} + \bar{A}^T P_1^T + Q_1 + R_1 - 4Q_3 - 4R_3 + E^T L_p^T L_p E,$$
$$\Lambda_{33} = -Q_1 + Q_2 - 4Q_4 - 4Q_3$$
$$\Lambda_{34} = S_{11}^T + S_{12}^T + S_{13}^T + S_{14}^T - 2Q_4,$$
$$\Lambda_{35} = -S_{11}^T + S_{13}^T - S_{12}^T + S_{14}^T,$$
$$\Lambda_{44} = -S_{11} - S_{11}^T - S_{13} - S_{13}^T + S_{12} + S_{12}^T + S_{14} + S_{14}^T - 8Q_4,$$
$$\Lambda_{45} = S_{11}^T - S_{13}^T - S_{12}^T + S_{14}^T - 2Q_4,$$
$$\Lambda_{66} = -R_1 + R_2 - 4R_4 - 4R_3,$$

$$\Lambda_{67} = S_{21}^T + S_{22}^T + S_{23}^T + S_{24}^T - 2R_4,$$

$$\Lambda_{68} = -S_{21}^T + S_{23}^T - S_{22}^T + S_{24}^T,$$

$$\Lambda_{77} = -S_{21} - S_{21}^T - S_{23} - S_{23}^T + S_{22} + S_{22}^T + S_{24} + S_{24}^T - 8R_4 + \lambda \bar{C}^T \Omega_j \bar{C},$$

$$\Lambda_{78} = S_{21}^T - S_{23}^T - S_{22}^T + S_{24}^T - 2R_4, \bar{C} = diag\{C_p, I\}.$$

Proof. Choose the following Lyapunov-Krasovskii functional

$$V(t) = \xi^T(t)P\xi(t) + V_1(t) + V_2(t) \tag{3.18}$$

where

$$\begin{aligned}
V_1(t) = &\int_{t-\tau_{j-1}}^{t} \eta^T(s)Q_1\eta(s)ds + \int_{t-\tau_j}^{t-\tau_{j-1}} \eta^T(s)Q_2\eta(s)ds \\
&+ \tau_{j-1}\int_{t-\tau_{j-1}}^{t}\int_{\theta}^{t} \dot{\eta}^T(s)Q_3\dot{\eta}(s)dsd\theta \\
&+ (\tau_j - \tau_{j-1})\int_{t-\tau_j}^{t-\tau_{j-1}}\int_{\theta}^{t} \eta^T(s)Q_4\eta(s)dsd\theta \\
V_2(t) = &\int_{t-d_m}^{t} \eta^T(s)R_1\eta(s)ds + \int_{t-d_M}^{t-d_m} \eta^T(s)R_2\eta(s)ds \\
&+ d_m\int_{t-d_m}^{t}\int_{\theta}^{t} \dot{\eta}^T(s)R_3\dot{\eta}(s)dsd\theta \\
&+ (d_M - d_m)\int_{t-d_M}^{t-d_m}\int_{\theta}^{t} \dot{\eta}^T(s)R_4\dot{\eta}(s)dsd\theta
\end{aligned}$$

where $P = \left[\begin{smallmatrix} P_1 & E^TP_2 \\ \star & P_3 \end{smallmatrix}\right] > 0$, $Q_1 > 0$, $Q_2 > 0$, $Q_3 > 0$, $Q_4 > 0$, $R_1 > 0$, $R_2 > 0$, $R_3 > 0$, $R_4 > 0$ to be determined.

Then, taking the derivative of $V(t)$ with respect to t along the trajectory of the system (3.11) yields

$$\begin{aligned}
\dot{V}(t) = &2\xi^T(t)P\dot{\xi}(t) + \eta^T(t)(Q_1 + R_1)\eta(t) \\
&- \eta^T(t-\tau_j)Q_2\eta(t-\tau_j) - \eta^T(t-d_M)R_2\eta(t-d_M) \\
&+ \eta^T(t-\tau_{j-1})(Q_2 - Q_1)\eta(t-\tau_{j-1}) + \eta^T(t-d_m)(R_2 - R_1)\eta(t-d_m) \\
&+ \dot{\eta}^T(t)(\tau_{j-1}^2Q_3 + (\tau_j - \tau_{j-1})^2Q_4 + d_m^2R_3 + (d_M - d_m)^2R_4)\dot{\eta}(t)
\end{aligned}$$

$$\begin{aligned}&-\tau_{j-1}\int_{t-\tau_{j-1}}^{t}\dot{\eta}^T(\theta)Q_3\dot{\eta}(\theta)d\theta-(\tau_j-\tau_{j-1})\int_{t-\tau_j}^{t-\tau_{j-1}}\dot{\eta}^T(\theta)Q_4\dot{\eta}(\theta)d\theta\\&-d_m\int_{t-d_m}^{t}\dot{\eta}^T(\theta)R_3\eta(\theta)d\theta-(d_M-d_m)\int_{t-d_M}^{t-d_m}\dot{\eta}^T(\theta)R_4\dot{\eta}(\theta)d\theta\end{aligned}\tag{3.19}$$

Applying Lemma 3.2, we have

$$-\tau_{j-1}\int_{t-\tau_{j-1}}^{t}\dot{\eta}^T(\theta)Q_3\dot{\eta}(\theta)d\theta\leq-\psi_0^T\begin{bmatrix}Q_3&0\\0&3Q_3\end{bmatrix}\psi_0\tag{3.20}$$

where

$$\psi_0=\begin{bmatrix}\eta(t)-\eta(t-\tau_{j-1})\\\eta(t)+\eta(t-\tau_{j-1})-v_0\end{bmatrix}$$

with

$$v_0(t)=\frac{2}{\tau_{j-1}}\int_{t-\tau_{j-1}}^{t}\eta(s)ds.$$

Applying Lemma 3.2 again, we obtain

$$-d_m\int_{t-d_m}^{t}\dot{\eta}^T(\theta)R_3\dot{\eta}(\theta)d\theta\leq-\tilde{\psi}_0^T\begin{bmatrix}R_3&0\\0&3R_3\end{bmatrix}\tilde{\psi}_0\tag{3.21}$$

where

$$\tilde{\psi}_0=\begin{bmatrix}\eta(t)-\eta(t-d_m)\\\eta(t)+\eta(t-d_m)-\tilde{v}_0\end{bmatrix}$$

with

$$\tilde{v}_0(t)=\frac{2}{d_m}\int_{t-d_m}^{t}\eta(s)ds.$$

By using Lemma 3.3, it is clear that

$$-(\tau_j-\tau_{j-1})\int_{t-\tau_j}^{t-\tau_{j-1}}\dot{\eta}^T(\theta)Q_4\dot{\eta}(\theta)d\theta\leq2\psi_{11}^TS_1\psi_{12}-\psi_{11}^T\tilde{Q}_4\psi_{11}-\psi_{12}^T\tilde{Q}_4\psi_{12}\tag{3.22}$$

where $\tilde{Q}_4=\mathrm{diag}\{Q_4,3Q_4\}$ and

$$\begin{cases}\psi_{11}=\begin{bmatrix}\eta(t-\tau(t))-\eta(t-\tau_j)\\\eta(t-\tau(t))+\eta(t-\tau_j)-2v_{11}(t)\end{bmatrix}\\\psi_{12}=\begin{bmatrix}\eta(t-\tau_{j-1})-\eta(t-\tau(t))\\\eta(t-\tau_{j-1})+\eta(t-\tau(t))-2v_{12}(t)\end{bmatrix}\end{cases}$$

with

$$\begin{cases} v_{11}(t) = \dfrac{1}{\tau_j - \tau(t)} \displaystyle\int_{t-\tau_j}^{t-\tau(t)} \eta(s)ds \\ v_{12}(t) = \dfrac{1}{\tau(t) - \tau_{j-1}} \displaystyle\int_{t-\tau(t)}^{t-\tau_{j-1}} \eta(s)ds \end{cases}$$

Applying Lemma 3.3 again to obtain

$$-(d_M - d_m)\int_{t-d_M}^{t-d_m} \dot{\eta}^T(\theta)R_4\dot{\eta}(\theta)d\theta \leq 2\psi_{21}^T S_2\psi_{22} - \psi_{21}^T\tilde{R}_4\psi_{21} - \psi_{22}^T\tilde{R}_4\psi_{22} \quad (3.23)$$

where $\tilde{R}_4 = \text{diag}\{R_4, 3R_4\}$ and

$$\begin{cases} \psi_{21} = \begin{bmatrix} \eta(t-d(t)) - \eta(t-d_M) \\ \eta(t-d(t)) + \eta(t-d_M) - 2v_{21}(t) \end{bmatrix} \\ \psi_{22} = \begin{bmatrix} \eta(t-d_m) - \eta(t-d(t)) \\ \eta(t-d_m) + \eta(t-d(t)) - 2v_{22}(t) \end{bmatrix} \end{cases}$$

with

$$\begin{cases} v_{21}(t) = \dfrac{1}{d_M - d(t)} \displaystyle\int_{t-d_M}^{t-d(t)} \eta(s)ds \\ v_{22}(t) = \dfrac{1}{d(t) - d_m} \displaystyle\int_{t-d(t)}^{t-d_m} \eta(s)ds \end{cases}$$

Based on (3.7), it is clear that for $t \in \Theta_k^i$

$$\begin{bmatrix} e_k(t_k^i h) \\ \psi(t_k^i h) \end{bmatrix}^T \Omega_j \begin{bmatrix} e_k(t_k^i h) \\ \psi(t_k^i h) \end{bmatrix} < \lambda \begin{bmatrix} y_p(t_k^i h) \\ \tilde{x}(t_k^i h) \end{bmatrix}^T \Omega_j \begin{bmatrix} y_p(t_k^i h) \\ \tilde{x}(t_k^i h) \end{bmatrix} \quad (3.24)$$

Substituting (3.20)-(3.24) into (3.19) yields

$$\dot{V}(t) + e^T(t)e(t) - \gamma_j^2 w^T(t)w(t) \leq \zeta^T(t)\Upsilon\zeta(t) \quad (3.25)$$

where $\zeta(t) = \text{col}\{\eta(t), x_f(t), \eta(t-\tau_{j-1}), \eta(t-\tau(t)), \eta(t-\tau_j), \eta(t-d_m), \eta(t-d(t)), \eta(t-d_M), \bar{e}(t-d(t)), w(t), v_0(t), \tilde{v}_0(t), v_{11}(t), v_{12}(t), v_{21}(t), v_{22}(t)\}$, and

$$\Upsilon = \Psi + \Gamma\left[\tau_{j-1}^2 Q_3 + (\tau_j - \tau_{j-1})^2 Q_4 + d_m^2 R_3 + (d_M - d_m)^2 R_4\right]\Gamma^T$$

If the matrix inequality in (3.16) is satisfied, then application of the Schur complement gives $\Upsilon < 0$. Thus, there exists a scalar $\sigma > 0$ such that

$$\dot{V}(t) + e^T(t)e(t) - \gamma_j^2 w^T(t)w(t) \leq -\sigma\zeta^T(t)\zeta(t) \leq -\sigma\xi^T(t)\xi(t) \quad (3.26)$$

When $w(t) \equiv 0$, from (3.26), it can be seen that

$$\begin{aligned}\dot{V}(t) &\leq -e^T e(t) - \sigma\xi^T(t)\xi(t) \\ &\leq -\sigma\xi^T(t)\xi(t) < 0, \quad \text{for } \xi(t) \neq 0\end{aligned} \tag{3.27}$$

Therefore, we can conclude that the filtering error system (3.11) with $w(t) \equiv 0$ is asymptotically stable.

When $w(t) \neq 0$, from (3.26), for any $t \geq 0$, we have

$$\int_0^t -e^T(s)e(s) + \gamma_j^2 w^T(s)w(s)ds \geq V(t) - V(0) \tag{3.28}$$

Under zero initial conditions,

$$\|e(t)\|_2 < \gamma_j \|w(t)\|_2 \tag{3.29}$$

This completes the proof. □

3.4 H_∞ filter design

It is obvious that in Theorem 3.1, the filtering parameters A_f, B_f and C_f are coupled with the Lyapunov matrix P. Hence, we can not obtain the filter parameters directly from Theorem 3.1. In the following, we will provide a method to codesign the filter parameters, the congestion controllers and the event-triggering scheme parameters.

Theorem 3.2. *For given scalars τ_{j-1}, τ_j, d_m, d_M, $\lambda \in (0,\ 1)$ and $\gamma_j > 0$, the event-triggered filtering problem is solvable, if there exist matrix $\Omega_j > 0$, $(j = 1, \cdot, N)$, $P_1 > 0$, $R_i > 0$, $Q_i > 0$ $(i = 1, 2, 3, 4)$, $S_1 = \left[\begin{smallmatrix} S_{11} & S_{12} \\ S_{13} & S_{14}\end{smallmatrix}\right] > 0$, $S_2 = \left[\begin{smallmatrix} S_{21} & S_{22} \\ S_{23} & S_{24}\end{smallmatrix}\right] > 0$, $W > 0$, $P_1 > E^T W E$, $\hat{A}_f$, $\hat{B}_f$ and $\hat{C}_f$ such that the following matrix inequalities hold*

$$\begin{bmatrix} \tilde{Q}_4 & S_1 \\ \star & \tilde{Q}_4 \end{bmatrix} \geq 0, \quad \begin{bmatrix} \tilde{R}_4 & S_2 \\ \star & \tilde{R}_4 \end{bmatrix} \geq 0 \tag{3.30}$$

$$\begin{bmatrix} \tilde{\Psi} & \Gamma_0 & \tau_{j-1}\Gamma_1 Q_3 & (\tau_j-\tau_{j-1})\Gamma_1 Q_4 & d_m\Gamma_1 R_3 & (d_M-d_m)\Gamma_1 R_4 & \Gamma_2 & \Gamma_3 \\ \star & -I & 0 & 0 & 0 & 0 & 0 & 0 \\ \star & \star & -Q_3 & 0 & 0 & 0 & 0 & 0 \\ \star & \star & \star & -Q_4 & 0 & 0 & 0 & 0 \\ \star & \star & \star & \star & -R_3 & 0 & 0 & 0 \\ \star & \star & \star & \star & \star & -R_4 & 0 & 0 \\ \star & \star & \star & \star & \star & \star & -P_1 & 0 \\ \star & \star & \star & \star & \star & \star & \star & -P_1 \end{bmatrix} < 0 \tag{3.31}$$

where $\tilde{Q}_4 = diag\{Q_4, 3Q_4\}$, $\tilde{R}_4 = diag\{R_4, 3R_4\}$, *and*

$$\begin{aligned}
\Gamma_0 &= col\{-E^T L_p^T, \hat{C}_f^T, \underbrace{0, \cdots, 0}_{14}\},\\
\Gamma_1 &= col\{\bar{A}^T, 0, 0, \tilde{\bar{A}}_d^T, 0, 0, 0, 0, \tilde{\bar{D}}^T, \bar{B}^T, 0, 0, 0, 0, 0, 0, \},\\
\Gamma_2 &= col\{0, WE, \underbrace{0, \cdots, 0,}_{14} \tau_{j-1}Q_3, (\tau_j-\tau_{j-1})Q_4, d_m R_3, (d_M-d_m)R_4\},\\
\Gamma_3 &= col\{0, 0, 0, \tilde{\bar{B}}_1^T Y_1^T, 0, 0, 0, 0, \tilde{\bar{B}}_2^T Y_2^T, \underbrace{0, \cdots, 0,}_{12}\},
\end{aligned}$$

$$\tilde{\Psi} = \begin{bmatrix} \tilde{\Psi}_{11} & \tilde{\Psi}_{12} & \tilde{\Psi}_{13} & \tilde{\Psi}_{14} & 0 \\ \star & \Psi_{22} & 0 & \Psi_{24} & \Psi_{25} \\ \star & \star & \Psi_{33} & \Psi_{34} & \Psi_{35} \\ \star & \star & \star & \Psi_{44} & 0 \\ \star & \star & \star & \star & \Psi_{55} \end{bmatrix} \tag{3.32}$$

with

$$\begin{aligned}
\tilde{\Psi}_{11} &= \begin{bmatrix} \tilde{\Lambda}_{11} & E^T\hat{A}_f + \bar{A}^T E^T W^T \\ \star & \hat{A}_f + \hat{A}_f^T \end{bmatrix},\\
\tilde{\Psi}_{12} &= \begin{bmatrix} -2Q_3 & P_1\tilde{\bar{A}}_d + Y_1\bar{\bar{B}}_1 & 0 \\ 0 & W^T E\tilde{\bar{A}}_d & 0 \end{bmatrix},\\
\tilde{\Psi}_{13} &= \begin{bmatrix} -2R_3 & E^T\hat{B}_f\tilde{l}C_pE & 0 \\ 0 & \hat{B}_f\tilde{l}C_pE & 0 \end{bmatrix},\\
\tilde{\Psi}_{14} &= \begin{bmatrix} P_1\tilde{\bar{D}} + E^T\hat{B}_f\bar{L} + Y_2\bar{\bar{B}}_2 & P_1\bar{B} & 6Q_3 & 6R_3 \\ WE\tilde{\bar{D}} + \hat{B}_f\bar{L} & WE\bar{B} & 0 & 0 \end{bmatrix},\\
\tilde{\Lambda}_{11} &= P_1\bar{A} + \bar{A}^T P_1^T + Q_1 + R_1 - 4Q_3 - 4R_3,\\
\bar{\bar{B}}_1 &= \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\bar{B}}_1 \end{bmatrix}, \bar{\bar{B}}_2 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\bar{B}}_1 \end{bmatrix}, \tilde{\bar{B}}_1 = diag\{-\frac{\tau_0^2 C_0}{2N^2}, -\frac{\tau_0^2 C_0}{2N^2}\}.
\end{aligned}$$

Moreover, the parameters of the filter for the filtering error system (3.11) are given as

$$A_f = \hat{A}_f W^{-1},\ B_f = \hat{B}_f,\ C_f = \hat{C}_f W^{-1} \tag{3.33}$$

Proof. Notice that $W > 0$. Then there exists a nonsingular real matrices P_2 and a real matrix $P_3 > 0$ such that $W = P_2 P_3^{-1} P_2^T$. Since $P_1 - E^T W E > 0$, we have $P_1 - E^T P_2 P_3^{-1} P_2^T E > 0$, which leads to $P = \left[\begin{smallmatrix} P_1 & E^T P_2 \\ \star & P_3 \end{smallmatrix}\right] > 0$. Denote

$$A_f = P_2^{-1} \hat{A}_f W^{-1} P_2,\ B_f = P_2^{-1} \hat{B}_f,\ C_f = \hat{C}_f W^{-1} P_2 \tag{3.34}$$

Define $\mathcal{J} := \text{diag}\{I, P_2 P_3^{-1}, \underbrace{I, \cdots, I}_{18}\}$ and pre- and post-multiply (3.16) by $\mathcal{J}$ and $\mathcal{J}^T$, respectively. Applying the Schur complement, and based on Lemma 3.1, one can deduce that the matrix inequality (3.31) implies (3.16). Therefore, if the conditions of Theorem 3.2 are satisfied, then so are the conditions of Theorem 3.1, which means that the filtering error system (3.11) is asymptotically stable with H_∞ performance γ_j. Finally, we prove the filter matrix parameters A_f, B_f and C_f can be obtained by (3.39). In fact, an observation from (3.34) is that

$$\begin{bmatrix} A_f & B_f \\ C_f & 0 \end{bmatrix} = \begin{bmatrix} P_2^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{A}_f W^{-1} & \hat{B}_f \\ \hat{C}_f W^{-1} & 0 \end{bmatrix} \begin{bmatrix} P_2 & 0 \\ 0 & I \end{bmatrix}$$

Then, it can be concluded that the filter (3.10) with parameters (A_f, B_f, C_f) is algebraically equivalent to the filter (3.10) with the parameters $(\hat{A}_f W^{-1}, \hat{B}_f, \hat{C}_f W^{-1})$, which completes the proof. □

If the network dynamics is not considered, the event-triggered communication scheme proposed in this thesis is no longer available. Nevertheless, we can also present the sufficient condition to design suitable filters to estimate the states of the control system. We first rewrite the corresponding filtering error system as

$$\begin{cases} \dot{\zeta}(t) &= \hat{A}\zeta(t) + \hat{E}\hat{H}\zeta(t - d(t)) + \hat{B}_e \hat{e}(t_k^i h) + \hat{B}_w w(t) \\ e(t) &= \hat{C}\zeta(t) \\ \zeta(\theta) &= \text{col}\{x_0, 0\},\ \theta \in [-d_M, 0], \end{cases} \tag{3.35}$$

where

$$\zeta(t) = \text{col}\{x_p(t), x_f(t)\}, \hat{A} = \text{diag}\{A_p, A_f\}, \hat{C} = \begin{bmatrix} L_p & -C_f \end{bmatrix},$$
$$\hat{E} = \begin{bmatrix} 0 \\ B_f C_p \end{bmatrix}, \hat{B}_w = \begin{bmatrix} B_p \\ 0 \end{bmatrix}, \hat{B}_e = \begin{bmatrix} 0 \\ B_f \end{bmatrix}, \hat{H} = \begin{bmatrix} I & 0 \end{bmatrix}.$$

Then, following the line of the proofs of Theorem 3.1 and 3.2, we have the following result.

Corollary 3.1. *For given scalars d_m, d_M, $\lambda \in (0,\ 1)$ and $\gamma > 0$, the event-triggered filtering problem is solvable, if there exist matrices $\Omega > 0$, $P_1 > 0$, $R_i > 0$ $(i = 1, 2, 3, 4)$, $S_2 = \left[\begin{smallmatrix} S_{21} & S_{22} \\ S_{23} & S_{24} \end{smallmatrix}\right] > 0$, $W > 0$, $P_1 > W$, $\hat{A}_f$, $\hat{B}_f$ and $\hat{C}_f$ such that the following matrix inequalities hold*

$$\begin{bmatrix} \tilde{R}_4 & S_2 \\ \star & \tilde{R}_4 \end{bmatrix} \geq 0 \tag{3.36}$$

$$\begin{bmatrix} \chi & \hat{\Gamma}_0 & d_m \hat{\Gamma}_1 R_3 & (d_M - d_m)\hat{\Gamma}_1 R_4 \\ \star & -I & 0 & 0 \\ \star & \star & -R_3 & 0 \\ \star & \star & \star & -R_4 \end{bmatrix} < 0 \tag{3.37}$$

where $\tilde{R}_4 = diag\{R_4, 3R_4\}$, and

$$\hat{\Gamma}_0 = col\{-L_p^T, \hat{C}_f^T, \underbrace{0, \cdots, 0}_{8}\},$$
$$\hat{\Gamma}_1 = col\{A_p^T, 0, 0, 0, 0, 0, B_p, 0, 0, 0\},$$

$$\chi = \begin{bmatrix} \chi_{11} & \chi_{12} & \chi_{13} & 0 \\ \star & \chi_{22} & \chi_{23} & \chi_{24} \\ \star & \star & \chi_{33} & 0 \\ \star & \star & \star & \chi_{44} \end{bmatrix} \tag{3.38}$$

with

$$\chi_{11} = \begin{bmatrix} P_1 A_p + A_p^T P_1^T + R_1 - 4R_3 & \hat{A}_f + A_p^T W^T \\ \star & \hat{A}_f + \hat{A}_f^T \end{bmatrix},$$
$$\chi_{12} = \begin{bmatrix} -2R_3 & \hat{B}_f \tilde{C}_p & 0 \\ 0 & \hat{B}_f C_p & 0 \end{bmatrix},$$

$$\chi_{13} = \begin{bmatrix} \hat{B}_f & P_1 B_p & 6R_3 \\ \hat{B}_f & W B_p & 0 \end{bmatrix},$$
$$\chi_{22} = \begin{bmatrix} \vartheta_{33} & \vartheta_{34} & \vartheta_{35} \\ \star & \vartheta_{44} & \vartheta_{45} \\ \star & \star & -R_2 - 4R_4 \end{bmatrix},$$
$$\chi_{24} = \begin{bmatrix} 0 & 0 & 6R_3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$\chi_{24} = \begin{bmatrix} -2S_{23}^T - 2S_{24}^T & 6R_4 \\ 2S_{23}^T - 2S_{24}^T + 6R_4 & -2S_{22} - 2S_{24} + 6R_4 \\ 6R_4 & 2S_{22} - 2S_{24} \end{bmatrix},$$
$$\chi_{33} = diag\{-\Omega, -\gamma^2 I, -12R_3\},$$
$$\chi_{44} = \begin{bmatrix} -12R_4 & 4S_{24} \\ \star & -12R_4 \end{bmatrix},$$
$$\vartheta_{33} = -R_1 + R_2 - 4R_4 - 4R_3,$$
$$\vartheta_{34} = S_{21}^T + S_{22}^T + S_{23}^T + S_{24}^T - 2R_4,$$
$$\vartheta_{35} = -S_{21}^T + S_{23}^T - S_{22}^T + S_{24}^T,$$
$$\vartheta_{44} = -S_{21} - S_{21}^T - S_{23} - S_{23}^T + S_{22} + S_{22}^T + S_{24} + S_{24}^T - 8R_4 + \lambda C_p^T \Omega C_p,$$
$$\vartheta_{45} = S_{21}^T - S_{23}^T - S_{22}^T + S_{24}^T - 2R_4.$$

Moreover, the parameters of the filter for the filtering error system (3.35) are given as

$$A_f = \hat{A}_f W^{-1},\ B_f = \hat{B}_f,\ C_f = \hat{C}_f W^{-1} \tag{3.39}$$

Proof 3.1. *The proof is omitted due to its similarity to those of Theorem 3.1 and 3.2.*

The example in Section 3.5 shows that Corollary 3.1 can achieve less conservative results than some existing ones.

3.5 An application to a mechanical system with two masses and two springs

Assume that the plant to be considered is as shown in Figure 3.2, where x_1 and

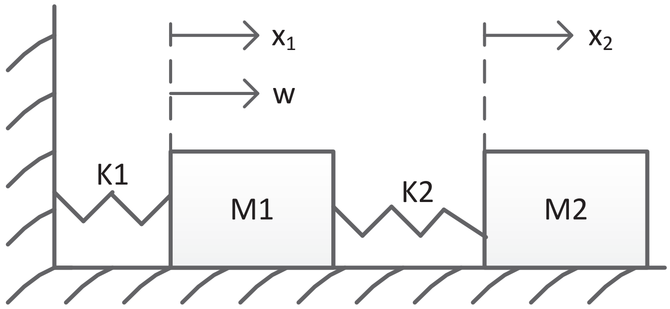

Figure 3.2: A mass-spring system

x_2 are the positions of the mass m_1 and m_2, respectively; and k_1, k_2 are spring constants. As specified in [125], $m_1 = 1$, $m_2 = 0.5$, $k_1 = 1$, $k_2 = 1$ and the viscous friction coefficient is 0.5. Then the networked system (3.1) with the parameters as

$$A_p = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2 & 1 & -0.5 & 0 \\ 2 & -2 & 0 & -1 \end{bmatrix}, B_p = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, C_p = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \tag{3.40}$$

and L_p is chosen as $L_p = \begin{bmatrix} 0.1 & 0.1 & 0.1 & 0.1 \end{bmatrix}$.

In this application, we assume that the data is transmitted over a low speed network, with the TCP/AQM communication network's parameters chosen as in Table 3.1.

Table 3.1: A low speed network

TCP session number N	50
Link capacity C	5000 packets/second
Round trip time τ_0	0.2s
Propagation delay T_p	0.1s
Queue size q_0	500 packets
Window size W_0	25 packets
Probability of packet mark p_0	0.0032

In this network environment, we choose the sampling period $h = 0.2s$. We decompose the delay interval [0.1 0.64] into 7 subintervals. Then employing Theorem 3.2, we obtain the 7 sets of the online scheduling parameters as in Table 3.2

The parameters for the online filter F_j, $(j = 1, \cdots, 7)$ can be obtained by em-

Table 3.2: Online congestion controller gains and filters

γ_j	λ	$\tilde{K}_1$	$\tilde{K}_2$	$\tau(t_k h)$	Filter
1	0.015	$[-66.0974\ 1.7643$	$[-0.0229\ -0.0009]$	$[0.10, 0.16)$	F_1
0.8	0.03	$[-35.3524\ 0.9085]$	$[-0.0084\ -0.0004]$	$[0.16, 0.22)$	F_2
1.2	0.045	$[-46.3352\ 1.1557]$	$[-0.0130\ -0.0005]$	$[0.22, 0.28)$	F_3
1.4	0.06	$[-44.1108\ 1.0829]$	$[-0.0120\ -0.0005]$	$[0.28, 0.34)$	F_4
1.6	0.075	$[-34.1376\ 0.8319]$	$[-0.0077\ -0.0004]$	$[0.34, 0.40)$	F_5
1.8	0.09	$[-54.2958\ 1.3065]$	$[-0.0163\ -0.0008]$	$[0.40, 0.46)$	F_6
2.0	0.105	$[-62.9985\ 1.5246]$	$[-0.0276\ -0.0010]$	$[0.46, 0.64]$	F_7

ploying Theorem 3.2. These parameters are as

$$
\begin{aligned}
A_f^1 &= \begin{bmatrix} -0.0296 & 0.0417 & 0.0614 & -0.0075 \\ -0.0084 & -0.0212 & -0.0426 & 0.0201 \\ -0.0465 & 0.0119 & -0.0203 & 0.0133 \\ 0.0292 & -0.0428 & -0.0311 & -0.0134 \end{bmatrix}, \\
B_f^1 &= \mathrm{col}\{-0.0081\ -0.0010\ -0.0006\ -0.0004\}, \\
C_f^1 &= \begin{bmatrix} -0.1983 & -0.0920 & -0.1876 & -0.0953 \end{bmatrix}.
\end{aligned}
$$

$$
\begin{aligned}
A_f^2 &= \begin{bmatrix} -0.0217 & 0.0272 & 0.0415 & -0.0039 \\ -0.0073 & -0.0149 & -0.0282 & 0.0134 \\ -0.0361 & 0.0071 & -0.0167 & 0.0075 \\ 0.0171 & -0.0288 & -0.0199 & -0.0111 \end{bmatrix}, \\
B_f^2 &= \mathrm{col}\{-0.0082\ -0.0012\ -0.0006\ -0.0007\}, \\
C_f^2 &= \begin{bmatrix} -0.1936 & -0.0880 & -0.1787 & -0.0924 \end{bmatrix}.
\end{aligned}
$$

$$
\begin{aligned}
A_f^3 &= \begin{bmatrix} -0.0245 & 0.0326 & 0.0489 & -0.0050 \\ -0.0079 & -0.0172 & -0.0336 & 0.0157 \\ -0.0400 & 0.0089 & -0.0180 & 0.0096 \\ 0.0213 & -0.0338 & -0.0241 & -0.0119 \end{bmatrix}, \\
B_f^3 &= \mathrm{col}\{-0.0081\ -0.0011\ -0.0006\ -0.0006\}, \\
C_f^3 &= \begin{bmatrix} -0.1961 & -0.0897 & -0.1829 & -0.0934 \end{bmatrix}.
\end{aligned}
$$

$$
\begin{aligned}
A_f^4 &= \begin{bmatrix} -0.0245 & 0.0326 & 0.0489 & -0.0050 \\ -0.0079 & -0.0172 & -0.0336 & 0.0157 \\ -0.0400 & 0.0089 & -0.0180 & 0.0096 \\ 0.0213 & -0.0338 & -0.0241 & -0.0119 \end{bmatrix}, \\
B_f^4 &= \mathrm{col}\{-0.0081\ -0.0011\ -0.0006\ -0.0006\},
\end{aligned}
$$

$$C_f^4 = \begin{bmatrix} -0.1961 & -0.0897 & -0.1829 & -0.0934 \end{bmatrix}.$$

$$\begin{aligned} A_f^5 &= \begin{bmatrix} -0.0216 & 0.0278 & 0.0423 & -0.0041 \\ -0.0080 & -0.0148 & -0.0285 & 0.0139 \\ -0.0369 & 0.0075 & -0.0166 & 0.0080 \\ 0.0172 & -0.0291 & -0.0202 & -0.0110 \end{bmatrix}, \\ B_f^5 &= \mathrm{col}\{-0.0082 - 0.0011 - 0.0006 - 0.0006\}, \\ C_f^5 &= \begin{bmatrix} -0.1930 & -0.0884 & -0.1783 & -0.0920 \end{bmatrix}. \end{aligned}$$

$$\begin{aligned} A_f^6 &= \begin{bmatrix} -0.0265 & 0.0369 & 0.0547 & -0.0054 \\ -0.0088 & -0.0188 & -0.0374 & 0.0172 \\ -0.0428 & 0.0097 & -0.0189 & 0.0112 \\ 0.0241 & -0.0375 & -0.0277 & -0.0123 \end{bmatrix}, \\ B_f^6 &= \mathrm{col}\{-0.0079 - 0.0010 - 0.0006 - 0.0005\}, \\ C_f^6 &= \begin{bmatrix} -0.1979 & -0.0907 & -0.1854 & -0.0939 \end{bmatrix}. \end{aligned}$$

$$\begin{aligned} A_f^7 &= \begin{bmatrix} -0.0290 & 0.0422 & 0.0619 & -0.0069 \\ -0.0099 & -0.0208 & -0.0423 & 0.0200 \\ -0.0469 & 0.0116 & -0.0200 & 0.0135 \\ 0.0284 & -0.0427 & -0.0318 & -0.0130 \end{bmatrix}, \\ B_f^7 &= \mathrm{col}\{-0.0078 - 0.0009 - 0.0006 - 0.0004\}, \\ C_f^7 &= \begin{bmatrix} -0.1979 & -0.0923 & -0.1887 & -0.0960 \end{bmatrix}. \end{aligned}$$

Based on the network dynamics and by employing Theorem 3.2, the weighting matrices under the online scheduling strategy can be obtained as

$$\Omega_1 = \begin{bmatrix} 0.0801 & -0.0005 & -0.0000 \\ -0.0005 & 0.4396 & -0.0084 \\ -0.0000 & -0.0084 & 0.0008 \end{bmatrix}, \Omega_2 = \begin{bmatrix} 0.0676 & -0.0003 & -0.0000 \\ -0.0003 & 0.2410 & -0.0047 \\ -0.0000 & -0.0047 & 0.0005 \end{bmatrix},$$

$$\Omega_3 = \begin{bmatrix} 0.0723 & -0.0004 & -0.0000 \\ -0.0004 & 0.3127 & -0.0061 \\ -0.0000 & -0.0061 & 0.0006 \end{bmatrix}, \Omega_4 = \begin{bmatrix} 0.0707 & -0.0003 & -0.0000 \\ -0.0003 & 0.3012 & -0.0059 \\ -0.0000 & -0.0059 & 0.0005 \end{bmatrix},$$

$$\Omega_5 = \begin{bmatrix} 0.0661 & -0.0003 & -0.0000 \\ -0.0003 & 0.2392 & -0.0047 \\ -0.0000 & -0.0047 & 0.0005 \end{bmatrix}, \Omega_6 = \begin{bmatrix} 0.0728 & -0.0004 & -0.0000 \\ -0.0004 & 0.3756 & -0.0073 \\ -0.0000 & -0.0073 & 0.0007 \end{bmatrix},$$

$$\Omega_7 = \begin{bmatrix} 0.0754 & -0.0006 & -0.0000 \\ -0.0006 & 0.3660 & -0.0062 \\ -0.0000 & -0.0062 & 0.0008 \end{bmatrix}.$$

Under the proposed scheduling strategy, the sampled measurement outputs (SN, $t_kh, y(t_kh), \tilde{x}(t_kh)$) of the networked system (3.1) are scheduled as shown in Table 3.3.

Table 3.3: An online scheduling strategy

SN	t_kh	$y(t_kh)$	$\tilde{x}(t_kh)$	Filter
S1	1h	0.0200	$[10.00, -500.0]^T$	F_1
S2	2h	0.0731	$[20.00, 336.3]^T$	F_3
S3	3h	0.2428	$[26.37, 891.0]^T$	F_5
S4	4h	0.3116	$[28.45, 1202]^T$	F_6
S5	9h	0.2389	$[15.79, 1009]^T$	F_6
S6	10h	0.1995	$[10.83, 78.78]^T$	F_5
S7	11h	0.1553	$[7.440, 563.6]^T$	F_4
...	...	...	...	...
S10	19h	0.0627	$[0.0368, 1.992]^T$	F_2^*
...	...	...	...	...

Given the above filters and the initial conditions of the system as $x_p(0) = \text{col}\{0.02, -0.02, 0.02, -0.02\}$ and $w(t) = 5e^{-t^2}\sin(3\pi t)$, the evolution of the filtering error signals are shown in Figure 3.3. The sampling, releasing and arriving instants of the transmitted data are illustrated in Figure 3.4. The network dynamics are shown in Figure 3.5.

On the other hand, in the time interval $[0,\ 10s]$, the number of sampled data packets is 50, while under the Information Dispatching Middleware, only 16 packets are transmitted to the filter. Then it can be seen from the above results that in this network environment, the measurement outputs are effectively scheduled by the proposed Information Dispatching Middleware framework. To be specific, nearly 32% of communication resources are saved. Moreover, the released packages are transmitted by the congestion module under the allowable transmission network-induced delay interval. Furthermore, the middleware framework parameters are set and suitable H_∞ filters are designed on the basis of network dynamics.

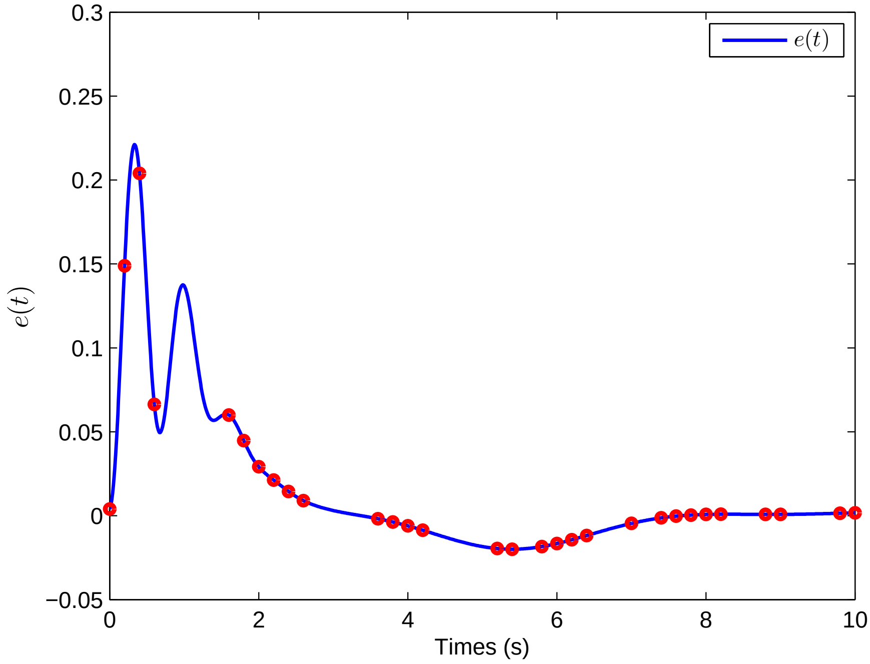

Figure 3.3: Estimation error e(t) with the event-triggered scheme

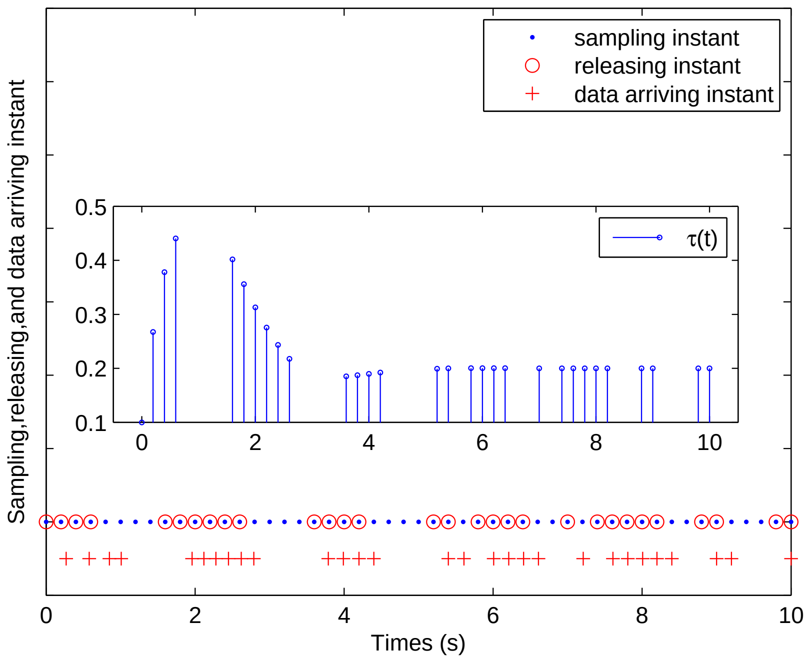

Figure 3.4: Sampling, releasing, arriving instants and network-induced delays

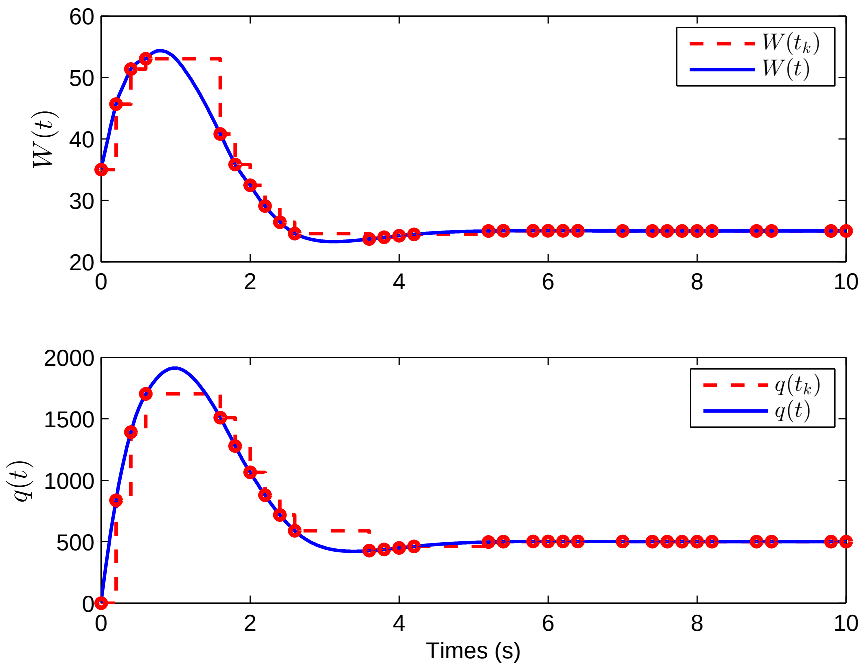

Figure 3.5: Network state responses with event-triggered instants

In order to compare with some existing results, we consider the case that the communication network is given in advance. Thus, it is assumed that the network dynamics is not considered in the filters design. For various λ, we calculate the minimum H_∞ performance level $\gamma_{\min}$ of $\gamma > 0$ using Corollary 3.1, and the obtained results are list in Table 3.4, from which one can see clearly that Corollary 3.1 in this thesis outperforms the ones in [71].

Table 3.4: Minimum values of γ for different λ.

λ	0.1	0.2	0.3	0.4	0.5
[71]	0.3341	0.3597	0.3772	0.3890	0.3972
Corollary 3.1	0.1891	0.1933	0.1962	0.1983	0.2001

3.6 Conclusion

In this chapter, event-triggered H_∞ filtering for networked systems has been investigated by taking network dynamics into account. An Information Dispatching Middleware has been constructed to establish a novel framework for networked systems, where two modules, namely the Information Selection Module and the Congestion Avoidance Module, are introduced. An online scheduling strategy has been proposed based on this framework. Then, the filtering error system based on network dynamics has been formulated as a system with two time-varying delays. By using Lyapunov-Krasovskii functional theory, a sufficient condition to ensure stability and to guarantee the prescribed H_∞ noise attenuation performance for the filtering error system has been derived. Based on the condition, the codesign method of the online filter and network congestion controllers has been proposed in terms of a set of linear matrix inequalities. Finally, a mechanical system with two masses and two springs has been used to illustrate the merits and effectiveness of the method proposed in this chapter.

Chapter 4

Network dynamic-based H_∞ control of large-scale distributed networked systems

4.1 Introduction

Distributed networked systems consist of several coupled subsystems that are geographically distributed and spatially connected through communication networks. There are a number of applications of this kind of system, for example, electrical power grid and transportation networks. In such large-scale systems, a group of physical distributed systems transmit data through a shared communication network. The introduction of a communication network provides several advantages such as cost effectiveness and ease of installation and maintenance, as previously described in Chapter 1. However, due to the limited network resources and the characteristics of large-scale systems, the analysis and control of distributed networked systems are significantly different from those of traditional control systems. Several results on analysis and design of distributed networked systems have been reported in the literature, see, e.g. [133], [134], [135] and the references therein.

For large-scale systems, there are three main control strategies: centralised control, decentralised control and distributed control. The centralised strategy requires the measurement outputs of all subsystems to be collected and transmitted to a centralised controller simultaneously in order to execute the tasks [136]. Moreover, the centralised strategy requires a very powerful communication network in order to transmit all the measurement outputs at the same time. To integrate these problems, decentralised architectures have been proposed. In a decentralised architecture, a controller is assigned to each subsystem through a communication network so that tasks can be executed based on local measurements. With this strategy, there is no signal transferred between different local controllers [137]. However, in some situations, while most of the signals are collected locally, some signals still need to be transmitted between the controllers to maintain the performance of the overall system. In this case, a distributed control strategy is preferable, in which the performance of the system is preserved while the burden on the communication network is reduced compared to the centralised strategy.

For efficient utilisation of limited network resources, it is important to introduce an event-triggered transmission scheme into the control of large-scale distributed systems to reduce some unnecessary transmissions. A big challenge in introducing the event-triggered scheme into a distributed control system is the asynchronous transmission of each subsystem, which makes the design of the event-triggered thresholds much more complicated than in the centralised case. To deal with this problem, several results have been reported in the literature. In [7] and [138], decentralised event-triggered feedback schemes are proposed in order to analyse the asymptotical stability of the linear and nonlinear systems, respectively. Based on a centralised approach, an event-triggered scheme is introduced into sensor-actuator networks [139], where the measurement error of the full system state is upper bounded by the centralised event-triggered threshold. This requires the measurement of all the subsystems to be transmitted to the central controller synchronously. More recently, an event-triggered scheme for distributed control systems is developed in [7], [51], [52]. In these results, the current state of the subsystem is sampled and released for transmission only if the local measurement error of the subsystem state exceeds a specified threshold. It should be pointed out that the results mentioned above require the states to be measured continuously. In these schemes, extra hardware is needed to detect the violation of the event-triggered condition. In addition, due to the continuous detection, a nonzero minimum inter-event time needs to be guaranteed for each sensor node to avoid the *Zeno* phenomenon. In practice, when communication networks take place over large-scale control systems, the data is transmitted in discrete packets. Therefore, when the data for each subsystem is detected at discrete sampling instants, hardware is not needed and the *Zeno* phenomenon is avoided.

In this chapter, we propose a distributed discrete event-triggered scheme where each subsystem triggers data asynchronously, which is different from synchronous transmission as studied in [139]. Although it brings difficulties in system modelling and analysis, asynchronous transmission can increase the flexibility and ease the implementation. In this chapter, a framework for large-scale systems in network environments is developed. In this framework, an Information Dispatching Middleware is invoked to schedule the sampled data by using the Information Selection Module and the Congestion Avoidance Module. The Information Selection Module contains an event-triggered scheme, which considers both the subsystem alert and network dynamics. It provides a tradeoff to balance networked system performance and network resources utilisation. In the Congestion Avoidance Module, the released packets are scheduled by congestion avoidance within the allowable network-induced delay. To achieve such a tradeoff, both the networked system and network dynamics are modelled in the Information Dispatching Middleware, which hides the low level details and reduces the engineers' burden. The parameters of the middleware and the controllers are codesigned by solving a set of linear matrix inequalities (LMIs). The design method is applied to a quadruple-tank process to illustrate the effectiveness of the proposed framework.

4.2 Modelling and problem formulation

4.2.1 A framework for a distributed networked system

Consider a large-scale distributed system, which consists of $\mathcal{N}$ coupled linear time-invariant subsystems described by

$$\begin{cases} \dot{x}_i(t) = A_{ii}x_i(t) + B_iu_i(t) + \sum\limits_{j\in\mathcal{N}_i} A_{ij}x_j(t) + B_w^i w_i(t) \\ y_i(t) = C_ix_i(t) \\ x_i(0) = x_i^0, \end{cases} \tag{4.1}$$

where $x_i(t) \in \mathbb{R}^{n_i}$ is the state vector of subsystem i; $u_i(t) \in \mathbb{R}^{m_i}$ is the control input; $w_i(t) \in \mathbb{R}^q$ is the exogenous disturbance, which belongs to $\mathcal{L}_2[0, +\infty)$; $y_i(t) \in \mathbb{R}^{s_i}$ is the measurement output vector; matrices A_i, A_{ij} B_i, B_w^i and C_i are known real matrices with appropriate dimensions; and ϕ_i is the initial condition.

The objective of this chapter is to control the distributed system (4.1) with remotely distributed controllers by employing an event-triggered scheme. It should be recalled that for existing event-triggered methods, most of them only consider the stability or the performance of the control system, while few of them consider the dynamics of the communication network, which is important for the control of distributed networked systems. In this chapter, we focus on the event-triggered distributed control for large-scale systems taking into account network dynamics, as shown in Figure 4.1.

The Information Dispatching Middleware illustrated in Figure 4.1 consists of two modules. One is the Information Selection Module; the other is the Congestion Control Module. In the Information Selection Module, there are two devices called a signal-receiver and an event-generator to deal with the received signals from distributed sensors, as shown in Figure 4.2. Between the signal-receiver and the event-generator, there are several event-detectors for different sensors to select the "needed" signal to be transmitted under some level of threshold λ, which determines the number of transmitted data and guarantees the prescribed performance

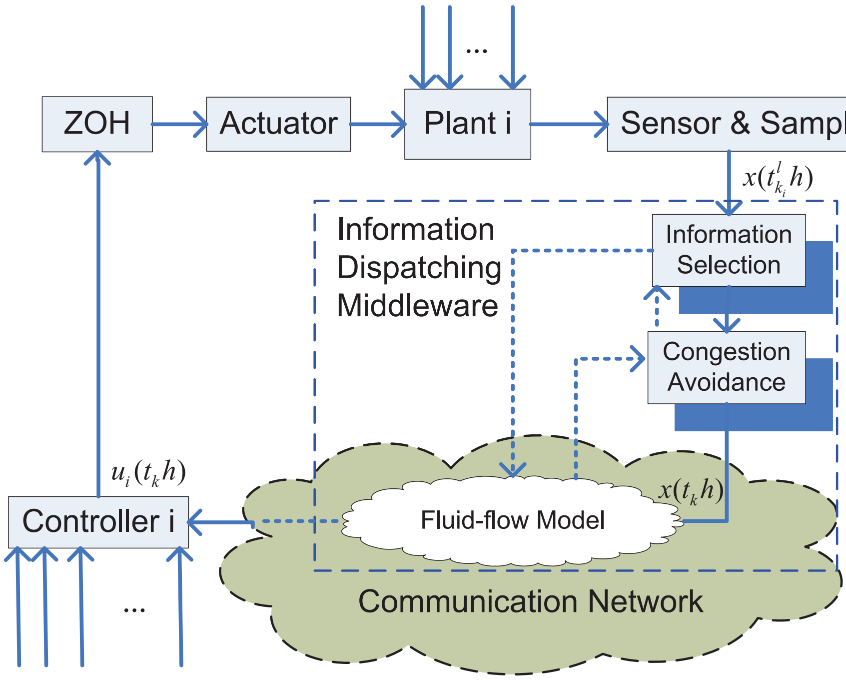

Figure 4.1: A framework for a distributed control system

of the distributed large-scale system.

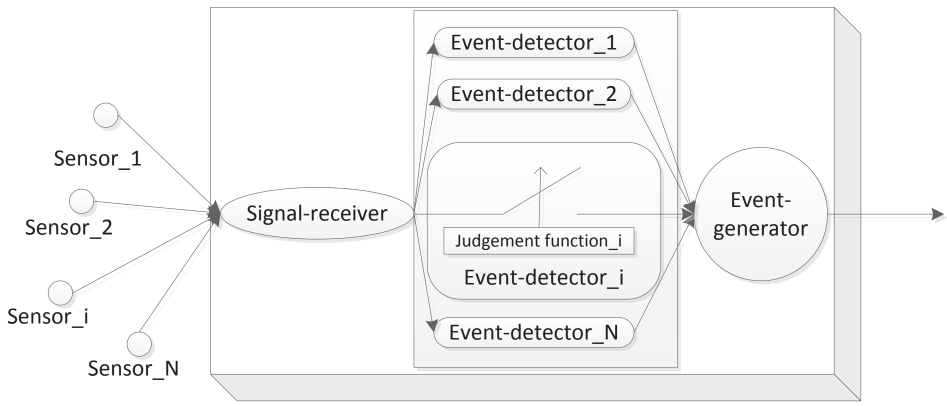

Figure 4.2: The Information Selection Module

More specifically, for the event-detector i, the main task is to select the "needed" sampled data to be transmitted from sensor i to the corresponding distributed controller $u_i(t)$ by using a predefined event-triggered scheme. The event-detector i receives not only the sampled data $x_i(sh)$ of sensor i, but also the network dynamics $\tilde{x}(sh)$, with a constant period $h > 0$ under some form of network time synchronisation mechanism. The set of sampled instants can be represented by $\{sh|s \in \mathbb{N}\}$;

the set of released instants $\{t_{k_i}h | t_{k_i} \in \mathbb{N}\}$ is a subset of the sampled instants and we define $t_{k_i}^l h = t_{k_i}h + lh$ as the corresponding following sampled instants for the released instant $t_{k_i}h$, with $l = 0, \ldots, L_{k_i}$, $L_{k_i} = t_{k_i+1} - t_{k_i}$ when $l = 0$, $t_{k_i}^l h = t_{k_i}h$; $l = L_{k_i}$, $t_{k_i}^l h = t_{k_i+1}h$.

The measurement state error $e_i(t_{k_i}^l h)$ of the sensor node i between the latest transmitted data $x_i(t_{k_i}h)$ and the corresponding sampled data $x_i(t_{k_i}^l h)$ can be calculated as

$$e_i(t_{k_i}^l h) = x_i(t_{k_i}^l h) - x_i(t_{k_i}h). \tag{4.2}$$

In event-detector i, the event-triggered comparator to check if the current data should be transmitted, is given by

$$\begin{bmatrix} e_i(t_{k_i}^l h) \\ \psi_i(t_{k_i}^l h) \end{bmatrix}^T \Omega_i \begin{bmatrix} e_i(t_{k_i}^l h) \\ \psi_i(t_{k_i}^l h) \end{bmatrix} \geq \lambda(\tilde{\gamma}) \begin{bmatrix} x_i(t_{k_i}^l h) \\ \tilde{x}(t_{k_i}^l h) \end{bmatrix}^T \Omega_i \begin{bmatrix} x_i(t_{k_i}^l h) \\ \tilde{x}(t_{k_i}^l h) \end{bmatrix} \tag{4.3}$$

where $x_i(t_{k_i}^l h)$, $\tilde{x}(t_{k_i}^l h)$, $e_i(t_{k_i}^l h)$ and $\psi_i(t_{k_i}^l h)$ are the state of subsystem i at the instant $t_{k_i}^l h$, the state of network dynamics at the instant $t_{k_i}^l h$, the state error of subsystem i at the instant $t_{k_i}^l h$, and the dynamic priority evaluation function of subsystem i at the instant $t_{k_i}^l h$, respectively; and $\lambda(\tilde{\gamma})$ is a function of the communication network performance index $\tilde{\gamma}$ to determine whether or not the sampled state $x_i(t_{k_i}^l h)$ should be packaged and transmitted. The network dynamics $\tilde{x}(t_{k_i}^l h)$, the dynamic priority evaluation function $\psi_i(t_{k_i}^l h)$ and the communication network performance index $\tilde{\gamma}$ are provided by the Congestion Control Module.

Remark 4.1. *In contrast to the event-triggered conditions in [70], [125], [140], the proposed event-triggered comparator in the Information Selection Module not only considers the state error $e_i(t_{k_i}^l h)$ of the sensor node i between the latest transmitted data $x_i(t_{k_i}h)$ and the following sampled data $x_i(t_{k_i}^l h)$, but also is concerned with the dynamic priority evaluation $\psi_i(t_{k_i}^l h)$ and the network dynamics $\tilde{x}(t_{k_i}^l h)$, which shows the coordination between the Information Selection Module and the Congestion*

Avoidance Module. Moreover, the transmitted packet is evaluated by $\psi_i(t^l_{k_i}h)$ based on its importance or significance, then the packets are assigned their own priority to access the communication network.

The selected sampled data $x_i(t^l_{k_i}h)$ is packaged as $(SN, t_{k_i+1}h, x_i(t_{k_i+1}h), \tilde{x}(t_{k_i+1}h))$ and transmitted through the communication network. t_0h is the initial released instant and the next transmission instant of the sensor node i can be expressed as

$$t_{k_i+1}h = t_{k_i}h + \inf_{l\geq 1}\{lh|f_i(t^l_{k_i}h) \geq 0, \quad i = 1, \cdots, \mathcal{N}\}. \tag{4.4}$$

where $f_i(t) = \left[\begin{smallmatrix} e_i(t) \\ \psi_i(t) \end{smallmatrix}\right]^T \Omega_i \left[\begin{smallmatrix} e_i(t) \\ \psi_i(t) \end{smallmatrix}\right] - \lambda(\tilde{\gamma}) \left[\begin{smallmatrix} x_i(t) \\ \tilde{x}(t) \end{smallmatrix}\right]^T \Omega_i \left[\begin{smallmatrix} x_i(t) \\ \tilde{x}(t) \end{smallmatrix}\right]$.

Remark 4.2. *An important parameter in the proposed event-triggered comparator is $\lambda(\tilde{\gamma})$, which reflects the level at which the packets are released by the Information Select Module. The relationship between the parameter λ and the H_∞ performance index $\tilde{\gamma}$ of the communication network is established, and a codesign method is provided in Section 4.3 to show how to determine the parameter $\lambda(\tilde{\gamma})$ based on the prescribed H_∞ performance index $\tilde{\gamma}$.*

In the Congestion Avoidance Module, network dynamics of a shared TCP/IP communication network and the network performance which shows the quality of the service (QoS) can be formulated as a nonlinear fluid-flow model by using a stochastic differential equation analysis approach [102]. As discussed in Chapter 2, a generalised fluid-flow model of TCP/AQM communication networks for large-scale distributed systems can be described as

$$\begin{cases} \dot{\tilde{x}}(t) &= \tilde{A}\tilde{x}(t) + \tilde{A}_d\tilde{x}(t-\tau(t)) + \tilde{B}_1\tilde{u}_1(t-\tau(t)) \\ & \quad +\tilde{B}_2\tilde{u}_2(t) + \tilde{D}\tilde{v}(t) + \tilde{B}_w\tilde{w}(t) \\ \tilde{y}(t) &= \tilde{H}\tilde{x}(t) \end{cases} \tag{4.5}$$

where $\tilde{x}(t) \in \mathbb{R}^2$, $\tilde{u}_1(t) \in \mathbb{R}^1$, $\tilde{u}_2(t) \in \mathbb{R}^1$, $\tilde{v}(t) \in \mathbb{R}^2$, $\tilde{w}(t) \in \mathbb{R}^2$ represent the internal state of the network, the internal control input for the network, the external control strategy for the network, the short-lived information need to be transmitted, and the external disturbance of the IP-based communication network, such as a birth-and death process of introducing short-lived http flows into the router, which belongs to $\mathcal{L}_2[0,\ \infty)$, with $\tilde{H} = \begin{bmatrix} 0 & 1 \end{bmatrix}$,

$$\tilde{A} = \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & -\frac{1}{\tau_0^2 C_0} \\ \frac{N}{\tau_0} & -\frac{1}{\tau_0} \end{bmatrix}, \tilde{A}_d = \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & \frac{1}{\tau_0^2 C_0} \\ 0 & 0 \end{bmatrix},$$

$$\tilde{B}_1 = \begin{bmatrix} -\frac{\tau_0^2 C_0}{2N^2} \\ 0 \end{bmatrix}, \qquad \tilde{B}_2 = \begin{bmatrix} -\frac{\tau_0^2 C_0}{2N^2} \\ 0 \end{bmatrix},$$

$$\tilde{D} = \begin{bmatrix} \frac{\tau_0 - T_p}{\tau_0^2 C_0} & -\frac{\tau_0 - T_p}{\tau_0^2 C_0} \\ -\frac{T_p}{\tau_0} & 0 \end{bmatrix}, \qquad \tilde{B}_w = \begin{bmatrix} \frac{\tau_0 - T_p}{\tau_0^2 C_0} & -\frac{\tau_0 - T_p}{\tau_0^2 C_0} \\ -\frac{T_p}{\tau_0} & 0 \end{bmatrix}.$$

It is assumed that the pair $(\tilde{A}, \tilde{B}_1)$ is controllable and the pair $(\tilde{A}, \tilde{H})$ is observable.

One of the purposes of the Congestion Avoidance Module is to design control laws such that the communication network system (4.5) has a prescribed H_∞ performance $\tilde{\gamma}$, that is

1) the system (4.5), with $\tilde{w}(t) \equiv 0$ is asymptotically stable; and

2) the H_∞ performance $\|\tilde{y}\|_2 < \tilde{\gamma}\|\tilde{w}(t)\|_2$ is guaranteed for all nonzero $\tilde{w}(t) \in \mathcal{L}_2[0, \infty)$.

The $\tilde{\gamma}$ is employed to design the threshold $\lambda(\tilde{\gamma})$ for the event-triggered scheme in the Information Selection Module, which is presented in Section 4.3.

Remark 4.3. *One of the purposes of introducing the Congestion Avoidance Module is to control the network dynamics approaching the equilibrium point of the network states and to avoid the queue size exceeding the maximum queue size $q_M = (\tau_M - \tau_P)C_0$ in the fluid-flow model. This means that the released packets are transmitted via some form of TCP/IP communication network within an allowable time delay.*

In this chapter, the released data is transmitted in the TCP/IP communication network, which means that the data packets released by the generator can be successfully transmitted to the ZOH without becoming disordered, while the network-induced delays are unavoidable. Considering the effects of the communication network, for the transmitted data $x_i(t_{k_i}h)$, it is assumed that the fluid-flow network introduces a communication delay $\tau(t_{k_i}h) = \frac{q(t_{k_i}h)}{C_0} + T_p$, in which the link capacity C_0 is fixed with $\tau_m \le \tau(t_{k_i}h) \le \tau_M$, $k_i \in \mathbb{N}$. τ_m is the fixed propagation delay T_p; τ_M is the maximum allowable transmission network-induced delay. When the packet $(SN_i, t_{k_i}h, x_i(t_{k_i}h), \tilde{x}(t_{k_i}h))$ reaches the distributor at $t_{k_i}h+\tau(t_{k_i}h)$, the packet is unpacked, then the data $x_i(t_{k_i}h)$ is released to the corresponding controller $u_i(t)$, and its ZOH keeps the data available until the new data arrives at $t_{k_i+1}h + \tau(t_{k_i+1}h)$. The holding zone of the ZOH is $[t_{k_i}h + \tau(t_{k_i}h), t_{k_i+1}h + \tau(t_{k_i+1}h)) = \bigcup_{l=0}^{L_{k_i}-1} \Theta^l_{k_i}$, where

$$\Theta^l_{k_i} = [t^l_{k_i}h + \tau(t^l_{k_i}h), t^{l+1}_{k_i}h + \tau(t^{l+1}_{k_i}h)) \tag{4.6}$$

with $t^l_{k_i}h = t_{k_i}h + lh$, $(l = 1, 2, \cdots, L_{k_i})$. $\tau(t^l_{k_i}h)$ can be chosen while keeping the right order of the start-point and the end-point of the $\Theta^l_{k_i}$.

In this chapter, a novel method is proposed to package the "needed" sampled signals to be transmitted through communication networks. At some sampled instants, if the judgement condition in any of the event-detectors is violated, the sampled signals released from the event-generator are packaged into one uni-packet as $(SN, t_kh, x_i(t_kh), x_j(t_kh), \cdots, \tilde{x}(t_kh))$ and then released to the communication network.

Figure 4.3 shows how the Information Selection Module works and how the distributor on the other side updates the stores of the distributed controllers. For example, at instant t_kh, the event-detector 2 and the event-detector $\mathcal{N}$'s judgement conditions are violated and these two sampled signals are triggered. Then the event-generator packages these two signals into a uni-packet as $(SN, t_kh, (x_2(t_kh), x_{\mathcal{N}}(t_kh)), \tilde{x}(t_kh))$. When the distributor receives this uni-packet and unpacks it, the stores of the ZOHs for both the controller 2 and controller $\mathcal{N}$ are updated.

For the distributed system, a set of sampled instants is chosen as $\{t^l_kh | t^l_k \in \mathbb{N}\}$ and the set of released instants of the event-generator as $\{t_kh | k \in \mathbb{N}\}$. The corresponding sampled instants for the released instant t_kh are $t^l_kh = t_kh + lh$, with

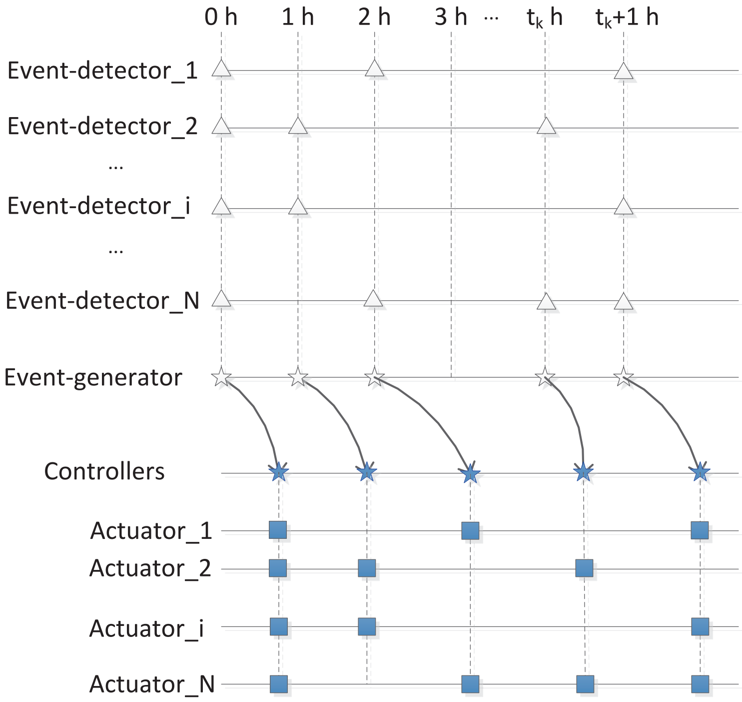

Figure 4.3: The uni-packet and actuator update

$l = 0, \ldots, L$, $L = t_{k+1} - t_k$ when $l = 0$, $t_k^l h = t_k h$; $l = L$, $t_k^l h = t_{k+1} h$. $t_0 h$ is set as the initial released instant of the uni-packet. The next transmission instant for the distributed system can be derived as

$$t_{k+1}h = t_k h + \inf_{l \geq 1}\{lh | \forall f_i(t_{k_i}^l h) \geq 0\}. \tag{4.7}$$

The holding zone of the distributor is $\bigcup_{k=0}^{\infty}[t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h)) = \bigcup_{k=0}^{\infty}\bigcup_{l=0}^{L-1}\Theta_k^l$, where $\Theta_k^l = [t_k^l h + \tau(t_k^l h), t_k^{l+1} h + \tau(t_k^{l+1} h))$. By introducing an artificial time delay $d(t) = t - t_k^l h$, $t \in \Theta_k^l$, from (4.2), we obtain

$$x_i(t_{k_i} h) = x_i(t - d(t)) - e_i(t_k^l h) \tag{4.8}$$

where $d(t)$ is a piecewise function satisfying $d_m \leq d(t) \leq d_M$, with $d_m = \tau_m$ and $d_M = h + \tau_M$. Based on (4.4), it is clear that for $t \in \Theta_k^l$

$$\begin{bmatrix} e_i(t_k^l h) \\ \psi_i(t_k^l h) \end{bmatrix}^T \Omega_i \begin{bmatrix} e_i(t_k^l h) \\ \psi_i(t_k^l h) \end{bmatrix} < \lambda(\tilde{\gamma}) \begin{bmatrix} x_i(t_k^l h) \\ \tilde{x}(t_k^l h) \end{bmatrix}^T \Omega_i \begin{bmatrix} x_i(t_k^l h) \\ \tilde{x}(t_k^l h) \end{bmatrix}. \tag{4.9}$$

Under the event-triggered scheme, the expression of $u_i(t)$ can be described as

$$\begin{aligned} u_i(t) &= K_i(x_i(t_{k_i}h) + \sum_{j\in\mathcal{N}_i} a^c_{ij} x_j(t_{k_j}h)) \\ &= K_i(x_i(t^l_k h) - e_i(t^l_k h) + \sum_{j\in\mathcal{N}_i} a^c_{ij}(x_j(t^l_k h) - e_j(t^l_k h))) \end{aligned} \tag{4.10}$$

where a^c_{ij} is a weighted adjacency between $x_i(t)$ and $x_j(t)$.

4.2.2 The closed-loop of the distributed H_∞ control system

From the previous analysis, the closed-loop of the distributed system based on the proposed framework with Information Dispatching Middleware can be modelled as a time-delay system.

For the subsystem i, from (4.1) and (4.10), we obtain

$$\begin{aligned} \dot{x}_i =& A_{ii}x_i + \sum_{j\in\mathcal{N}_i} A_{ij}x_j + B_iK_i[x_i(t^l_k h) + \sum_{j\in\mathcal{N}_i} a_{ij}x_j(t^l_k h)] \\ &- B_iK_i[e_i(t^l_k h) + \sum_{j\in\mathcal{N}_i} a_{ij}e_j(t^l_k h)] + B^i_w w_i(t) \end{aligned} \tag{4.11}$$

Therefore, under the proposed scheduling middleware, the large-scale distributed networked system can be expressed as

$$\begin{cases} \dot{x}(t) =Ax(t) + BKA^c x(t-d(t)) \\ \qquad - BKA^c e(t-d(t)) + B_w w(t) \\ y(t) =Cx(t), \quad t\in\Theta^l_k \end{cases} \tag{4.12}$$

where

$$\begin{aligned} x(t) &= \mathrm{col}\{x_1(t), x_2(t), \cdots, x_{\mathcal{N}}(t)\}, \\ e(t-d(t)) &= \mathrm{col}\{e_1(t-d(t)), e_2(t-d(t)), \cdots, e_{\mathcal{N}}(t-d(t))\}, \\ w(t) &= \mathrm{col}\{w_1(t), w_2(t), \cdots, w_{\mathcal{N}}(t)\}, \end{aligned}$$

with

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1\mathcal{N}} \\ A_{21} & A_{22} & \cdots & A_{1\mathcal{N}} \\ \vdots & \vdots & \ddots & \vdots \\ A_{\mathcal{N}1} & A_{\mathcal{N}2} & \cdots & A_{\mathcal{N}\mathcal{N}} \end{bmatrix}, A^c = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1\mathcal{N}} \\ a_{21} & 1 & \cdots & a_{1\mathcal{N}} \\ \vdots & \vdots & \ddots & \vdots \\ a_{\mathcal{N}1} & a_{\mathcal{N}2} & \cdots & 1 \end{bmatrix},$$

$$B = \mathrm{diag}\{B_1, B_2, \cdots, B_{\mathcal{N}}\}, \quad B_w = \mathrm{diag}\{B_w^1, B_w^2, \cdots, B_w^{\mathcal{N}}\},$$
$$K = \mathrm{diag}\{K_1, K_2, \cdots, K_{\mathcal{N}}\}, \quad C = \mathrm{diag}\{C_1, C_2, \cdots, C_{\mathcal{N}}\}.$$

In the proposed framework, $\psi(t_k^i h)$ is the dynamic priority evaluation of subsystem i for the network QoS adaptation mechanism at the instant $t_k^i h$. By using the Congestion Avoidance Module, the fluid-flow model for TCP/AQM communication network becomes

$$\begin{cases} \dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{A}_d\tilde{x}(t-\tau(t)) + \tilde{B}_1\tilde{K}_1\tilde{x}(t-\tau(t)) \\ \qquad + \tilde{B}_2\tilde{K}_2\psi(t_k^l h) + \tilde{D}\tilde{F}e(t_k^l h) + \tilde{B}_w\tilde{w}(t) \\ \tilde{y}(t) = \tilde{H}\tilde{x}(t) \end{cases} \tag{4.13}$$

where

$$\tilde{B}_2 = [\tilde{B}_2^1, \tilde{B}_2^2, \cdots, \tilde{B}_2^{\mathcal{N}}], \quad \tilde{K}_2 = \mathrm{diag}\{\tilde{K}_{21}, \tilde{K}_{22}, \cdots, \tilde{K}_{2\mathcal{N}}\},$$
$$\tilde{D} = [\tilde{D}_1, \tilde{D}_2, \cdots, \tilde{D}_{\mathcal{N}}], \quad \tilde{F} = \mathrm{diag}\{\tilde{F}_1, \tilde{F}_2, \cdots, \tilde{F}_{\mathcal{N}}\},$$
$$\psi(t_k^l h) = \mathrm{diag}\{\psi(t_{k_1}^l h), \psi(t_{k_2}^l h), \cdots, \psi(t_{k_{\mathcal{N}}}^l h)\},$$
$$e(t_k^l h) = \mathrm{diag}\{e(t_{k_1}^l h), e(t_{k_2}^l h), \cdots, e(t_{k_{\mathcal{N}}}^l h)\}.$$

Therefore, considering distributed system (4.12), the network dynamics (4.13), and (4.8) and the distributed controllers (4.10) we obtain the following augmented system

$$\begin{cases} \dot{\eta}(t) = \bar{A}\eta(t) + (\bar{A}_d + \bar{B}_1\bar{K}_1)\eta(t-\tau(t)) \\ \qquad + (-\bar{B}\bar{K}\bar{A}^c + \bar{D} + \bar{B}_2\bar{K}_2)\bar{e}(t-d(t)) \\ \qquad + \bar{B}\bar{K}\bar{A}^c\eta(t-d(t)) + \bar{B}_w\bar{w}(t) \\ \bar{y}(t) = \bar{C}\eta(t), \quad t \in \Theta_k^l \end{cases} \tag{4.14}$$

where $\eta(t) = \mathrm{col}\{x(t), \tilde{x}(t)\}$, $\bar{y}(t) = \mathrm{col}\{y(t), \tilde{y}(t)\}$, $\bar{w}(t) = \mathrm{col}\{w(t), \tilde{w}(t)\}$, $\bar{e}(t -$

$d(t)) = \mathrm{col}\{e(t-d(t)), \psi(t-d(t))\}$,

$$\begin{aligned}
\bar{A} &= \begin{bmatrix} A & 0 \\ 0 & \tilde{A} \end{bmatrix}, \ \bar{A}_d = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{A}_d \end{bmatrix}, \ \bar{B}_1 = \bar{B}_2 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{B}_1 \end{bmatrix}, \\
\bar{K}_1 &= \begin{bmatrix} 0 & 0 \\ 0 & \tilde{K}_1 \end{bmatrix}, \ \bar{B} = \begin{bmatrix} B_1 & 0 \\ 0 & 0 \end{bmatrix}, \ \bar{K} = \begin{bmatrix} K & 0 \\ 0 & 0 \end{bmatrix}, \\
\bar{A}^c &= \begin{bmatrix} A^c & 0 \\ 0 & 0 \end{bmatrix}, \ \bar{D} = \begin{bmatrix} 0 & 0 \\ \tilde{D}\tilde{F} & 0 \end{bmatrix}, \ \bar{K}_2 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{K}_2 \end{bmatrix}, \\
\bar{B}_w &= \mathrm{diag}\{B_w, \tilde{B}_w\}, \bar{C} = \mathrm{diag}\{C, \tilde{H}\}.
\end{aligned}$$

Under the proposed scheduling framework, the distributed H_∞ control problem to be addressed is stated as follows. Given a scalar $\bar{\gamma} = \mathrm{diag}\{\gamma, \tilde{\gamma}\}$, design controller parameters K, $\tilde{K}_1$ and $\tilde{K}_2$, such that the augmented system (4.14) with $\bar{w}(t) = 0$ is asymptotically stable; and the H_∞ performance $\|\bar{y}(t)\|_2 < \bar{\gamma}\|\bar{w}(t)\|_2$ is guaranteed for all nonzero $\bar{w}(t) \in \mathcal{L}_2[0,\ \infty)$ and a prescribed $\bar{\gamma} > 0$ under the condition $\eta(t) = 0$, $\forall t \in [-d_M,\ 0]$.

To this end, we introduce the following lemma, which is useful in solving the above problem.

Lemma 4.1. *For a symmetric positive definite matrix $R \in \mathbb{R}^{n\times n}$ and $S > 0 \in \mathbb{R}^{n\times n}$, the following inequality holds for any matrix $S > 0$ satisfying* $\begin{bmatrix} R & S \\ \star & R \end{bmatrix} \geq 0$

$$\begin{aligned}
&-(\tau_M - \tau_m)\int_{t-\tau_M}^{t-\tau_m} \dot{\eta}^T(s) R \dot{\eta}(s) ds \\
&\leq \xi(t)^T \begin{bmatrix} -R & R-S & S \\ \star & S+S^T-2R & R-S \\ \star & \star & -R \end{bmatrix} \xi(t)
\end{aligned} \tag{4.15}$$

where $\xi(t) = col\{\eta(t-\tau_m), \eta(t-\tau(t)), \eta(t-\tau_M)\}$.

4.3 The analysis and synthesis of the proposed framework for the distributed system

4.3.1 H_∞ performance analysis

In this subsection, we focus on the H_∞ performance analysis of the augmented system (4.14). Controller design is discussed in the next subsection.

Theorem 4.1. *For a given $\bar{\gamma}$, the augmented system (4.14) is asymptotically stable with the H_∞ performance $\bar{\gamma}$, if there exist real matrices $\Omega > 0$, $P > 0$, $S_1 > 0$, $S_2 > 0$, $Q_i > 0$, $R_i > 0$ $(i = 1, 2, 3, 4)$ and real nonsingular matrices M_1, M_2 of appropriate dimensions such that*

$$\begin{bmatrix} R_2 & S_1 \\ \star & R_2 \end{bmatrix} \geq 0, \quad \begin{bmatrix} R_4 & S_2 \\ \star & R_4 \end{bmatrix} \geq 0 \tag{4.16}$$

$$\begin{bmatrix} \Psi_{11} & \Psi_{12} & \Psi_{13} \\ \star & \Psi_{22} & \Psi_{23} \\ \star & \star & \Psi_{33} \end{bmatrix} < 0 \tag{4.17}$$

$$\Psi_{11} = \begin{bmatrix} \Lambda_1^1 & R_1 & P\bar{A}_d + P\bar{B}_1\bar{K}_1 + M_1^T\bar{A}_d + M_1^T\bar{B}_1\bar{K}_1 \\ \star & \Lambda_2^1 & R_2 - S_1 \\ \star & \star & S_1 + S_1^T - 2R_2 \end{bmatrix},$$

$$\Psi_{12} = \begin{bmatrix} 0 & R_3 & P\bar{B}\bar{K}\bar{A}^c + M_1^T\bar{B}\bar{K}\bar{A}^c & 0 \\ S_1 & 0 & 0 & 0 \\ R_2 - S_1 & 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{13} = \begin{bmatrix} \Lambda_3^1 & \bar{A}^c - M_1^T + \bar{A}^TM_2 & P\bar{B}_w + M_1^T\bar{B}_w \\ 0 & 0 & 0 \\ 0 & (\bar{A}_d + \bar{B}_1\bar{K}_1)^TM_2 & 0 \end{bmatrix},$$

$$\Psi_{22} = \begin{bmatrix} -Q_2 - R_2 & 0 & 0 & 0 \\ \star & \Lambda_4^1 & R_4 - S_2 & S_2 \\ \star & \star & \Lambda_5^1 & R_4 - S_2 \\ \star & \star & \star & -Q_4 - R_4 \end{bmatrix},$$

$$\Psi_{23} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & (\bar{B}\bar{K}\bar{A}^c)^TM_2 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{33} = \begin{bmatrix} -H_2^T\Omega H_2 & (\bar{D} - \bar{B}\bar{K}\bar{A}^c + \bar{B}_2\bar{K}_2)^TM_2 & 0 \\ \star & \Lambda_6^1 & 0 \\ \star & \star & -\bar{\gamma}^2I \end{bmatrix},$$

$$\Lambda_1^1 = P\bar{A} + \bar{A}^TP + M_1^T\bar{A} + \bar{A}^TM_1 + Q_1 + Q_3 - R_1 - R_3 + \bar{C}^T\bar{C},$$

$$\Lambda_2^1 = -Q_1 + Q_2 - R_1 - R_2,$$

$$\Lambda_3^1 = (M_1^T + P)(\bar{D} - \bar{B}\bar{K}\bar{A}^c + \bar{B}_2\bar{K}_2),$$

$$\Lambda_4^1 = -Q_3 + R_4 - R_3 - R_4,$$

$$\Lambda_5^1 = S_2 + S_2^T - 2R_4 + \lambda(\tilde{\gamma})H_1^T\Omega H_1,$$

$$\Lambda_6^1 = -M_2 - M_2^T + \tau_m^2R_1 + d_m^2R_3 + (\tau_M - \tau_m)^2R_2 + (d_M - d_m)^2R_4.$$

Proof. Choose the following Lyapunov-Krasovskii functional

$$V(t) = \eta^T(t)P\eta(t) + V_1(t) + V_2(t) \tag{4.18}$$

where

$$\begin{aligned}
V_1(t) &= \int_{t-\tau_m}^{t} \eta^T(s)Q_1\eta(s)ds + \int_{t-\tau_M}^{t-\tau_m} \eta^T(s)Q_2\eta(s)ds \\
&+ \int_{t-d_m}^{t} \eta^T(s)Q_3\eta(s)ds + \int_{t-d_M}^{t-d_m} \eta^T(s)Q_4\eta(s)ds \\
V_2(t) &= \tau_M \int_{-\tau_M}^{0} \int_{t+\theta}^{t} \dot{\eta}^T(s)R_1\dot{\eta}(s)dsd\theta \\
&+ (\tau_M - \tau_m) \int_{-\tau_M}^{-\tau_m} \int_{t+\theta}^{t} \dot{\eta}^T(s)R_2\dot{\eta}(s)dsd\theta \\
&+ d_m \int_{-d_m}^{0} \int_{t+\theta}^{t} \dot{\eta}^T(s)R_3\dot{\eta}(s)dsd\theta \\
&+ (d_M - d_m) \int_{-d_M}^{-d_m} \int_{t+\theta}^{t} \dot{\eta}^T(s)R_4\dot{\eta}(s)dsd\theta
\end{aligned}$$

Then, taking the derivative of $V(t)$ with respect to t along the trajectory of the system (4.14) yields

$$\begin{aligned}
\dot{V}(t) &= 2\eta^T(t)P\dot{\eta}(t) + \eta^T(t)(Q_1 + Q_3)\eta(t) \\
&+ \eta^T(t-\tau_m)(Q_2 - Q_1)\eta(t-\tau_m) \\
&- \eta^T(t-\tau_M)Q_2\eta(t-\tau_M) \\
&+ \eta^T(t-d_m)(Q_4 - Q_3)\eta(t-d_m) \\
&+ \dot{\eta}^T(t)\Lambda\dot{\eta}(t) - \eta^T(t-d_M)Q_4\eta(t-d_M) \\
&- \tau_m \int_{t-\tau_m}^{t} \dot{\eta}^T(\theta)R_1\dot{\eta}(\theta)d\theta \\
&- d_m \int_{t-d_m}^{t} \dot{\eta}^T(\theta)R_3\eta(\theta)d\theta \\
&- (\tau_M - \tau_m) \int_{t-\tau_M}^{t-\tau_m} \dot{\eta}^T(\theta)R_2\dot{\eta}(\theta)d\theta \\
&- (d_M - d_m) \int_{t-d_M}^{t-d_m} \dot{\eta}^T(\theta)R_4\dot{\eta}(\theta)d\theta
\end{aligned} \tag{4.19}$$

where $\Lambda = \tau_m^2 R_1 + d_m^2 R_3 + (\tau_M - \tau_m)^2 R_2 + (d_M - d_m)^2 R_4$. Applying Lemma 4.1, we have

$$\begin{aligned}
&-(\tau_M - \tau_m)\int_{t-\tau_M}^{t-\tau_m} \dot{\eta}^T(s) R_2 \dot{\eta}(s) ds \\
&\leq \xi_1^T(t) \begin{bmatrix} -R_2 & R_2 - S_1 & S_1 \\ \star & S_1 + S_1^T - 2R_2 & R_2 - S_1 \\ \star & \star & -R_2 \end{bmatrix} \xi_1(t)
\end{aligned} \tag{4.20}$$

$$\begin{aligned}
&-(d_M - d_m)\int_{t-d_M}^{t-d_m} \dot{\eta}^T(s) R_4 \dot{\eta}(s) ds \\
&\leq \xi_2^T(t) \begin{bmatrix} -R_4 & R_4 - S_2 & S_2 \\ \star & S_2 + S_2^T - 2R_4 & R_4 - S_2 \\ \star & \star & -R_4 \end{bmatrix} \xi_2(t)
\end{aligned} \tag{4.21}$$

where $\xi_1(t) = \mathrm{col}\{\eta(t-\tau_m), \eta(t-\tau(t)), \eta(t-\tau_M)\}$, $\xi_2(t) = \mathrm{col}\{\eta(t-d_m), \eta(t-d(t)), \eta(t-d_M)\}$ with

$$\begin{bmatrix} R_2 & S_1 \\ \star & R_2 \end{bmatrix} \geq 0, \quad \begin{bmatrix} R_4 & S_2 \\ \star & R_4 \end{bmatrix} \geq 0 \tag{4.22}$$

It is clear that there exist real nonsingular matrices M_1 and M_2 such that

$$\begin{aligned}
&(2\eta^T(t)M_1^T + 2\dot{\eta}^T(t)M_2^T)(\bar{A}\eta(t) + (\bar{A}_d + \bar{B}_1\bar{K}_1)\eta(t-\tau(t)) \\
&+ (-\bar{B}\bar{K}\bar{A}^c + \bar{D} + \bar{B}_2\bar{K}_2)\bar{e}(t-d(t)) \\
&+ \bar{B}\bar{K}\bar{A}^c\eta(t-d(t)) + \bar{B}_w\bar{w}(t) - \dot{\eta}(t)) = 0
\end{aligned} \tag{4.23}$$

Substituting (4.9) and (4.20)-(4.23) into (4.19) yields

$$\dot{V}(t) + \bar{y}^T(t)\bar{y}(t) - \bar{\gamma}^2\bar{w}^T(t)\bar{w}(t) \leq \zeta^T(t)\Psi\zeta(t) \tag{4.24}$$

where $\zeta(t) = \mathrm{col}\{\eta(t), \eta(t-\tau_m), \eta(t-\tau(t)), \eta(t-\tau_M), \eta(t-d_m), \eta(t-d(t)), \eta(t-d_M), \bar{e}(t-d(t)), \dot{\eta}(t)\}$ and

$$\Psi = \begin{bmatrix} \Psi_{11} & \Psi_{12} & \Psi_{13} \\ \star & \Psi_{22} & \Psi_{23} \\ \star & \star & \Psi_{33} \end{bmatrix}$$

First, we consider the asymptotic stability of the augmented system (4.14) with $\bar{w}(t) = 0$. When $\bar{w}(t) = 0$, (4.24) implies

$$\dot{V}(t) \leq \zeta^T(t)\Psi\zeta(t) \tag{4.25}$$

(4.17) yields $\Psi < 0$. Thus, from (4.25), we have $\dot{V}(t) \leq \rho\|\eta(t)\|^2$, where $\rho = -\lambda_{\min}(-\Psi)$. Therefore, system (4.14) is asymptotically stable according to Theorem 4.1.

Next, we prove the H_∞ performance of the augmented system (4.14) for all nonzero $\bar{w}(t) \in \mathcal{L}_2[0,\ \infty)$. If Theorem 4.1 is satisfied, we have

$$\dot{V}(t) < -\bar{y}^T(t)\bar{y}(t) + \bar{\gamma}^2\bar{w}^T(t)\bar{w}(t) \tag{4.26}$$

Integrating both sides of (4.26) from 0 to ∞ on t, and under the zero initial condition, one can obtain that

$$\int_0^\infty \bar{y}^T(t)\bar{y}(t)dt < \bar{\gamma}^2 \int_0^\infty \bar{w}^T(t)\bar{w}(t)dt \tag{4.27}$$

that is $\|\bar{y}\|_2 < \bar{\gamma}\|\bar{w}\|_2$ for all nonzero $\bar{w}(t)$, which completes the proof. □

4.3.2 H_∞ controller design

In this subsection, based on Theorem 4.1, a solution to the distributed H_∞ control problem for the large-scale distributed networked system (4.1) and network dynamic system (4.13) within the proposed framework is obtained based on the feasibility of a set of LMIs. Then, the codesign method for the design of the Information Selection Module and the Congestion Avoidance Module is introduced.

Theorem 4.2. *For given scalars $\bar{\gamma}$, $\beta > 0$, the distributed H_∞ control problem for the augmented system (4.14) is solvable, if there exist real matrices $\Omega > 0$, $P > 0$, $P_0 > 0$, $S_1 > 0$, $S_2 > 0$, $Q_i > 0$, $R_i > 0$ $(i = 1, 2, 3, 4)$, real nonsingular matrices $M_1 > 0$, $M_2 > 0$ and Y_1, Y_2 of appropriate dimensions such that (4.16) and*

$$\begin{bmatrix} -\beta I & \bar{B}_1^T P - P_0 \bar{B}_1^T \\ P\bar{B}_1 - \bar{B}_1 P_0 & -I \end{bmatrix} \leq 0 \tag{4.28}$$

$$\begin{bmatrix} \bar{\Psi}_{11} & \bar{\Psi}_{12} & \bar{\Psi}_{13} \\ \star & \bar{\Psi}_{22} & \bar{\Psi}_{23} \\ \star & \star & \bar{\Psi}_{33} \end{bmatrix} < 0 \tag{4.29}$$

where

$$\bar{\Psi}_{11} = \begin{bmatrix} \Lambda_1^2 & R_1 & \Lambda_2^2 & 0 \\ \star & \Lambda_3^2 & R_2 - S_1 & S_1 \\ \star & \star & S_1 + S_1^T - 2R_2 & R_2 - S_1 \\ \star & \star & \star & -Q_2 - R_2 \end{bmatrix},$$

$$\bar{\Psi}_{12} = \begin{bmatrix} R_3 & \bar{B}Y\bar{A}^c & 0 & M_1^T\bar{D} + P\bar{D} - \bar{B}Y\bar{A}^c + Y_2\bar{\bar{B}}_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\bar{\Psi}_{13} = \begin{bmatrix} -M_1^T + \bar{A}^T M_2 & PB_w + M_1^T\bar{B}_w & M_1^T & 0 \\ 0 & 0 & 0 & 0 \\ \bar{A}_d^T M_2 & 0 & 0 & \bar{\bar{B}}_1^T Y_1^T \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\bar{\Psi}_{22} = \begin{bmatrix} \Lambda_4^2 & R_4 - S_2 & S_2 & 0 \\ \star & \Lambda_5^2 & R_4 - S_2 & 0 \\ \star & \star & -Q_4 - R_4 & 0 \\ \star & \star & \star & -H_2^T\Omega H_2 \end{bmatrix},$$

$$\bar{\Psi}_{23} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{A}^{cT}Y^T\bar{B}^T \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\bar{A}^{cT}Y^T\bar{B}^T + \bar{\bar{B}}_2^T Y_2^T \end{bmatrix},$$

$$\bar{\Psi}_{33} = \begin{bmatrix} \Lambda_6^2 & M_2^T\bar{B}_w^T & M_2^T & 0 \\ \star & -\bar{\gamma}^2 I & 0 & 0 \\ \star & \star & -P & 0 \\ \star & \star & 0 & -P \end{bmatrix},$$

$$\Lambda_1^2 = P\bar{A} + \bar{A}^T P + M_1^T\bar{A} + \bar{A}^T M_1 + Q_1 + Q_3 - R_1 - R_3 + \bar{C}^T\bar{C},$$

$$\Lambda_2^2 = P\bar{A}_d + Y_1\bar{\bar{B}}_1 + M_1^T\bar{A}_d,$$

$$\Lambda_3^2 = -Q_1 + Q_2 - R_1 - R_2,$$

$$\Lambda_4^2 = -Q_3 + Q_4 - R_3 - R_4,$$

$$\Lambda_5^2 = S_2 + S_2^T - 2R_4 + \lambda(\tilde{\gamma})H_1^T\Omega H_1,$$

$$\Lambda_6^2 = -M_2 - M_2^T + \tau_m^2 R_1 + d_m^2 R_3 + (\tau_M - \tau_m)^2 R_2 + (d_M - d_m)^2 R_4,$$

$$\bar{\bar{B}}_1 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\bar{B}}_1 \end{bmatrix}, \bar{\bar{B}}_2 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\bar{B}}_2 \end{bmatrix},$$

$$\tilde{\bar{B}}_1 = diag\{-\frac{\tau_0^2 C_0}{2N^2}, -\frac{\tau_0^2 C_0}{2N^2}\}, \tilde{\bar{B}}_2 = diag\underbrace{\tilde{\bar{B}}_1, \cdots, \tilde{\bar{B}}_1}_{\mathcal{N}}\}.$$

Moreover, the gain matrices in (4.14) are given as

$$\bar{K} = P_0^{-1}Y, \quad \bar{\bar{K}}_1 = P^{-1}Y_1, \quad \bar{\bar{K}}_2 = P^{-1}Y_2 \tag{4.30}$$

and $\bar{\bar{K}}_1 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\bar{K}}_1 \end{bmatrix}$, $\bar{\bar{K}}_2 = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{\bar{K}}_2 \end{bmatrix}$ *where* $\tilde{\bar{K}}_1 = \begin{bmatrix} \tilde{K}_1 \\ 0 \end{bmatrix}$ *and* $\tilde{\bar{K}}_2 = \begin{bmatrix} \tilde{K}_2 \\ 0 \end{bmatrix}$.

Proof. First, the linear equality condition

$$P\bar{B}_1 = \bar{B}_1 P_0 \tag{4.31}$$

can be equivalently converted to

$$\mathrm{tr}((P\bar{B}_1 - \bar{B}_1 P_0)^T (P\bar{B}_1 - \bar{B}_1 P_0)) = 0 \tag{4.32}$$

which can be expressed as

$$(P\bar{B}_1 - \bar{B}_1 P_0)^T (P\bar{B}_1 - \bar{B}_1 P_0) \leq \beta I \tag{4.33}$$

where β is a small enough positive scalar. Based on Schur's complement, (4.33) equals to (4.28). Then it is clear that $P\bar{B}\bar{K} = \bar{B}P_0\bar{K}$, if condition (4.28) is satisfied.

Next, if $\tilde{K}_1 = \begin{bmatrix}\tilde{K}_{11} & \tilde{K}_{12}\end{bmatrix}$ and $\tilde{K}_2 = \begin{bmatrix}\tilde{K}_{21} & \tilde{K}_{22}\end{bmatrix}$, from the definition of $\tilde{B}_1$ and $\tilde{B}_2$, it is obvious that

$$\bar{B}_1\bar{K}_1 = \bar{\bar{K}}_1\bar{\bar{B}}_1, \quad \bar{B}_2\bar{K}_2 = \bar{\bar{K}}_2\bar{\bar{B}}_2 \tag{4.34}$$

From Lemma 4.1 and the above discussion, (4.17) equals to (4.29), which completes the proof. □

For a given prescribed H_∞ performance $\tilde{\gamma} > 0$ of the communication network, the threshold λ in the Information Selection Module can be determined and mapped in the range of $(0, 1)$ for each event-detector. By using Theorem 4.2, we can codesign the H_∞ distributed controllers and the Information Dispatching Middleware including the Information Selection Module and the Congestion Avoidance Module.

How to determine the proper threshold λ under the prescribed communication network's H_∞ performance $\tilde{\gamma}$ is an important issue, which requires a codesign s-trategy to investigate the relationship between these two main parameters. In this chapter, we design

$$\lambda(\tilde{\gamma}) = arctan(\alpha\tilde{\gamma}) \tag{4.35}$$

where $\alpha > 0$ is a scalar regulation parameter.

Algorithm 2 Codesign the H_∞ controllers and the Information Dispatching Middleware

1: Preset some initial parameters, such as the sampling time $h > 0$, an H_∞ performance $\tilde{\gamma}$ and so on;
2: Set a range $(0, \tilde{\gamma}]$, where the corresponding optimal H_∞ performance $\tilde{\gamma}$ can be obtained and let $\theta_1 = 0$ and $\theta_2 = \tilde{\gamma}$;
3: **if** Theorem 4.2 is satisfied, that is, LMIs (4.16), (4.28) and (4.29) are feasible with θ_2 and $\lambda(\theta_2)$ **then**
4: Obtain the corresponding optimal H_∞ performance $\tilde{\gamma}_m = \theta_2$ by using the binary search method;
5: **end if**
6: With the desired performance, by using Theorem 4.2, the parameters of the Information Dispatching Middleware and the H_∞ controllers for the large-scale distributed networked system (4.1) and network dynamic system (4.13) can be codesigned.

Therefore, under the prescribed H_∞ performance $\bar{\gamma}$, we can codesign the distributed controllers for the large-scale distributed networked system (4.1) and the Information Dispatching Middleware. By using the proposed method, the optimal H_∞ performance $\tilde{\gamma}$ for the network dynamic system (4.13) can be obtained with the following Algorithm 2.

4.4 An application to the quadruple-tank process

In this section, the model of a quadruple-tank process is considered as an example to illustrate the effectiveness of the proposed method. The process consists of four interconnected water tanks and two pumps. The schematic diagram of the quadruple-tank process from [141] is shown in Figure 4.4. The aim is to control the level in the lower two tanks with two pumps though a communication network. A schematic diagram of the distributed control of the process is shown in Figure 4.5. The inputs $v_1(t)$ and $v_2(t)$ are the voltages to the two pumps and the outputs $y_1(t)$ and $y_2(t)$ are the water levels in the lower two tanks. The coupled tanks can be modelled by means of the following nonlinear model [142]:

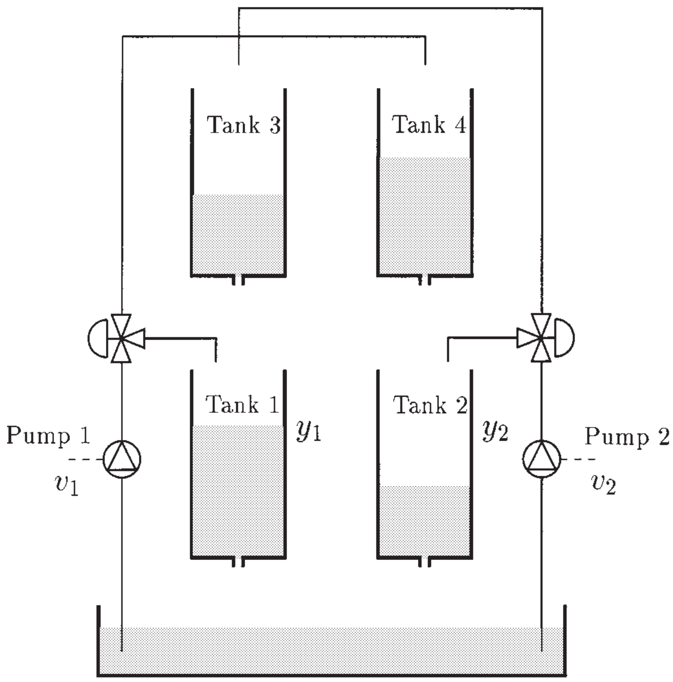

Figure 4.4: The schematic diagram of the quadruple-tank process

$$\begin{cases} \frac{dh_1(t)}{dt} &= -\frac{a_1}{A}\sqrt{2gh_1(t)} + \frac{\gamma_1 k_1}{A_1}v_1(t) \\ & \quad -\frac{a_{13}}{A}\sqrt{2g(h_1(t)-h_3(t))} \\ \frac{dh_2(t)}{dt} &= \frac{a_1}{A}\sqrt{2gh_1(t)} - \frac{a_2}{A}\sqrt{2gh_2(t)} \\ \frac{dh_3(t)}{dt} &= -\frac{a_3}{A}\sqrt{2gh_3(t)} + \frac{(1-\gamma_2)k_2}{A_3}v_2(t) \\ & \quad +\frac{a_{13}}{A}\sqrt{2g(h_1(t)-h_3(t))} \\ \frac{dh_4(t)}{dt} &= \frac{a_3}{A}\sqrt{2gh_3(t)} - \frac{a_4}{A}\sqrt{2gh_4(t)} \\ y_i &= kh_i, i = 1, 2 \end{cases} \tag{4.36}$$

where $h_i(t)(i = 1, ..., 4)$ denotes the water level in the tanks; $v_i(i = 1, 2)$ are voltages applied to the pumps; $a_i(i = 1, ..., 4)$ are the outlet areas of the tanks; a_{13} is the outlet area between tanks 1 and 3; A is the cross-sectional area of the tanks; and g is the gravitational constant. The parameter values of the laboratory process are

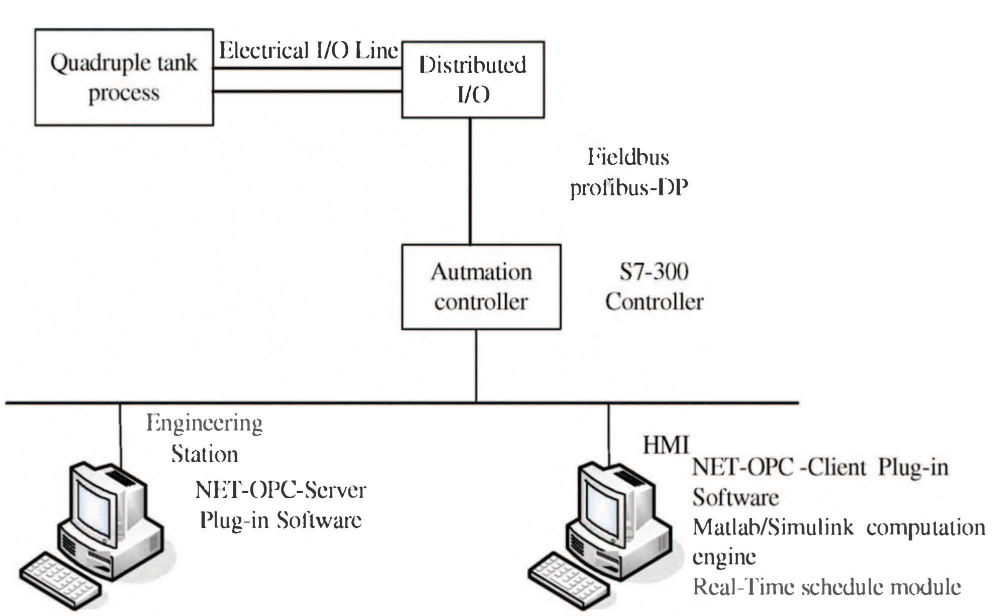

Figure 4.5: Distributed networked control systems for quadruple-tank process

given as $A = 28\text{cm}^2$, $a_1 = a_3 = 0.071\text{cm}^2$, $a_2 = a_4 = 0.057\text{cm}^2$, $a_{13} = 0.057\text{cm}^2$, $g = 981\text{cm/s}^2$ and $k = 0.5\text{V/cm}$. The operating-range of the system is

$$0 \le h_i \le 25, i = 1, ...4, \quad 0 \le v_i \le 5, i = 1, 2 \tag{4.37}$$

The chosen operating points correspond to the following parameter values

$$\begin{array}{rcl} h_1^0 &=& 12.4\text{cm}, \quad h_2^0 = 12.7\text{cm}, \quad h_3^0 = 1.8\text{cm}, \\ h_4^0 &=& 1.4\text{cm}, \quad v_1^0 = 3.3\text{V}, \quad v_2^0 = 2.6\text{V}, \quad \gamma_1 = 0.8, \\ \gamma_2 &=& 0.3, \quad k_1 = 3.33\text{cm}^3\text{V/s}, \quad k_2 = 3.35\text{cm}^3\text{V/s}. \end{array}$$

This system is linearised around the equilibrium point given by h_i^0 and u_i^0, yielding

$$\left\{ \begin{array}{rcl} \dot{x}(t) &=& Ax(t) + Bu(t) + B_w w(t) \\ y(t) &=& Cx(t) \end{array} \right.$$

where $x(t) = \text{col}\{h_1(t) - h_1^0, h_2(t) - h_2^0, h_3(t) - h_3^0, h_4(t) - h_4^0\}$, $u(t) = \text{col}\{v_1(t) - v_1^0, v_2(t) - v_2^0\}$ and $w(t) = \text{col}\{w_1(t), w_2(t)\}$. With a proper approximation of the nonlinear equations of the model (4.36), we have

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \quad B_w = \begin{bmatrix} B_w^1 \\ B_w^2 \end{bmatrix}$$

with

$$A_{11} = \begin{bmatrix} -0.0332 & 0 \\ 0.0159 & -0.0127 \end{bmatrix}, A_{12} = \begin{bmatrix} 0.0172 & 0 \\ 0 & 0 \end{bmatrix},$$

$$A_{22} = \begin{bmatrix} -0.0591 & 0 \\ 0.0419 & -0.0381 \end{bmatrix}, B_1 = \begin{bmatrix} 0.0951 \\ 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0.0359 \\ 0 \end{bmatrix}, \tag{4.38}$$
$$B_w^1 = B_w^2 = \begin{bmatrix} 0.0001 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0.5 & 0.2 & 0 & 0 \\ 0 & 0 & 0.5 & 0.1 \end{bmatrix}.$$

Assume there is a communication network between the system and the distributed controllers. The parameters of the network are shown in Table 4.1.

Table 4.1: A low speed IP-based communication network

TCP session number N	100
Link capacity C	5000 packets/second
Round trip time τ_0	0.25s
Propagation delay T_p	0.1s
Queue size q_0	750 packets
Window size W_0	25 packets
Probability of packet mark p_0	0.0032

The weighted adjacency matrix is as

$$A^c = \begin{bmatrix} 1 & 0.5 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 & 0.5 \\ 0.5 & 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 0.5 & 1 \end{bmatrix} \tag{4.39}$$

For a given H_∞ performance $\gamma = 1$ of the large-scale system, when Algorithm 2 is applied, we obtain the admissible minimum H_∞ performance $\tilde{\gamma} = 2.0010$ of the communication network. Using (4.35) and Theorem 4.2, the parameters of the event-triggered scheme in the Information Selection Module are obtained as

$$\begin{aligned}
\lambda &= 0.0127, \\
\Omega_1 &= \begin{bmatrix} 670.8354 & 45.0406 & -0.0207 & 0.6645 \\ 45.0406 & 658.7630 & -0.0041 & -0.1585 \\ -0.0207 & -0.0041 & 688.4885 & -5.8668 \\ 0.6645 & -0.1585 & -5.8668 & 23.5755 \end{bmatrix},
\end{aligned}$$

$$\Omega_2 = \begin{bmatrix} 688.0072 & 33.5328 & -0.0105 & 0.4170 \\ 33.5328 & 772.4596 & -0.0043 & -0.1348 \\ -0.0105 & -0.0043 & 688.4885 & -5.8668 \\ 0.4170 & -0.1348 & -5.8668 & 23.5750 \end{bmatrix}.$$

The distributed controller gain K and the congestion controller gains $\tilde{K}_1$, $\tilde{K}_2$ are as

$$K = \begin{bmatrix} -1.2717 & -0.1482 & -0.4733 & -0.4733 \\ -0.7808 & -0.7808 & -2.0442 & -0.2983 \end{bmatrix}, \tag{4.40}$$

$$\tilde{K}_1 = \begin{bmatrix} -12.3866 & 0.8865 \end{bmatrix}, \tag{4.41}$$

$$\tilde{K}_2 = \begin{bmatrix} -0.0001 & -0.0011 & -0.0002 & -0.0015 \end{bmatrix}. \tag{4.42}$$

We can also obtain the decentralised controller gain K and the congestion controller gains as

$$K = \begin{bmatrix} -2.1488 & -0.1957 & 0 & 0 \\ 0 & 0 & -4.8141 & -0.9180 \end{bmatrix},$$
$$\tilde{K}_1 = \begin{bmatrix} -12.2613 & 0.8857 \end{bmatrix},$$
$$\tilde{K}_2 = 10^{-3} \times \begin{bmatrix} -0.0495 & 0.0341 & -0.0473 & 0.1068 \end{bmatrix}.$$

In the same way, we can also derive the centralised controller gain K and the congestion controller gains $\tilde{K}_1$, $\tilde{K}_2$ as

$$K = \begin{bmatrix} -0.2518 & -0.2518 & -0.2518 & -0.2518 \\ -0.5453 & -0.5453 & -0.5453 & -0.5453 \end{bmatrix},$$
$$\tilde{K}_1 = \begin{bmatrix} -12.4837 & 0.8875 \end{bmatrix},$$
$$\tilde{K}_2 = \begin{bmatrix} -0.0001 & -0.0012 & -0.0001 & -0.0009 \end{bmatrix}.$$

From the above result, it can be seen that by employing Theorem 4.2, we can obtain the distributed controller gains, decentralised controller gains and the centralised controller gain, which means that this method is practical to a class of large-scale distributed systems in network environments.

In the simulation, the initial states of the subsystems are given as $x_1(0) = \mathrm{col}\{1, -0.3\}$, $x_2(0) = \mathrm{col}\{1, -0.2\}$ and the external disturbances $w_1(t) = w_2(t) = 5e^{-t^2}\sin(3\pi t)$. The sampling period is set to $h = 0.2$s. The disturbance of the IP-based communication network is a birth-and-death process, in the form of

$$\tilde{w}(t) = BavK(t)$$

where $Bav = 200$ packets/s is the average transmission rate of the http flows; $K(t)$ is a birth-and-death process, where $-K_{\max} \leq K(t) < K_{\max}$, with $K_{\max} = 25$.

By Theorem 4.2, we obtain the distributed controller gain as in (4.40). The state responses of subsystem 1 and subsystem 2 are shown in Figure 4.6 and Figure 4.7, respectively. Figure 4.8 and Figure 4.9 illustrate the sampling instants, releasing

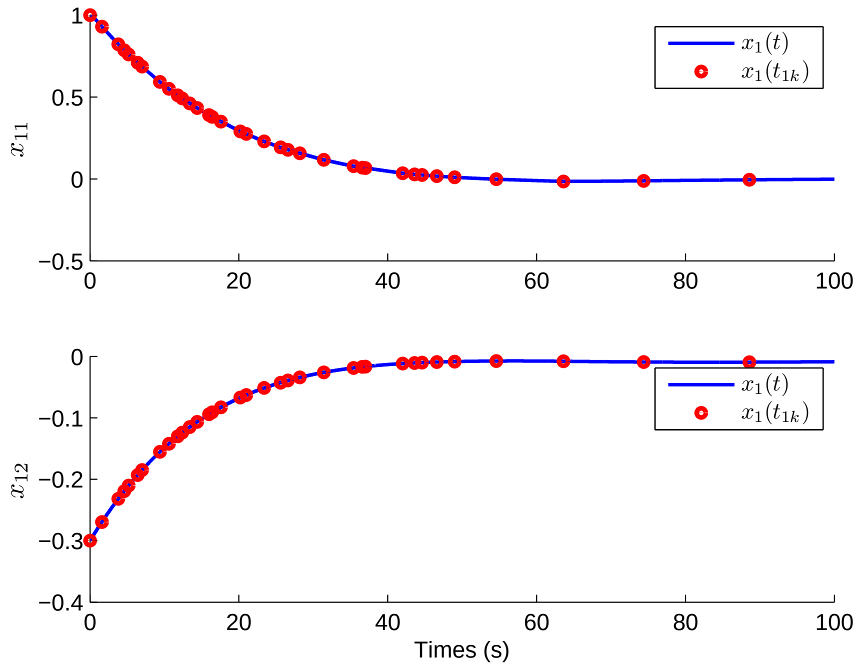

Figure 4.6: The state responses of subsystem 1

instants and arrival instants of subsystem 1 and subsystem 2 on the interval $[0,\ 100\text{s})$, respectively. Figure 4.10 and Figure 4.11 show the release instants and the intervals of each subsystem. The sampled data transmission rates of each subsystem are 8.2% and 7.2%, respectively. These results show that the number of transmitted data is significantly reduced by using the proposed event-triggered strategy. Next, we depict the states of the communication network in Figure 4.12. From this figure, it can be seen that the data can be scheduled and transmitted depending on the dynamics of the communication network. At the same time, network resources are utilised efficiently, which illustrates the effectiveness of the proposed design method.

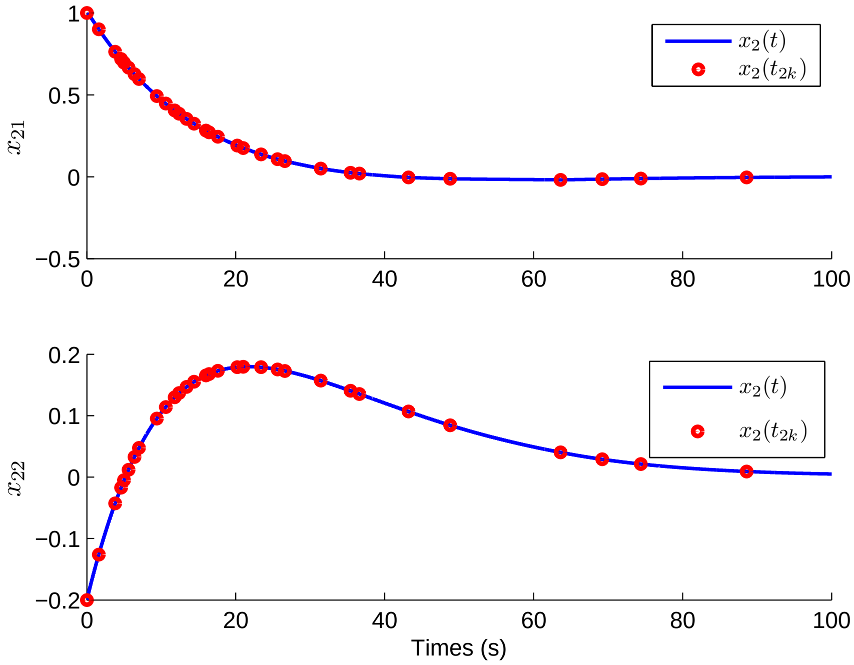

Figure 4.7: The state responses of subsystem 2

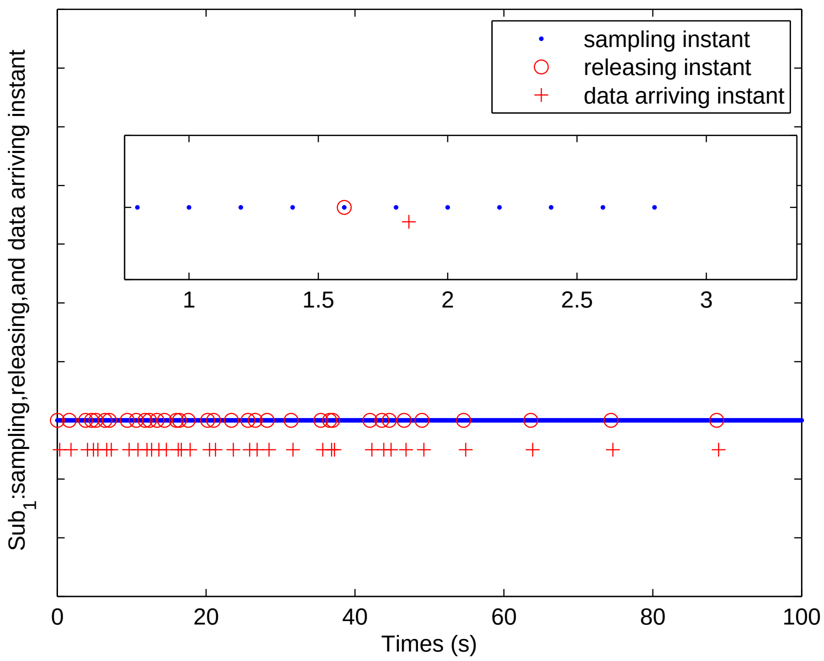

Figure 4.8: Sampling, releasing and data arriving instants of subsystem 1

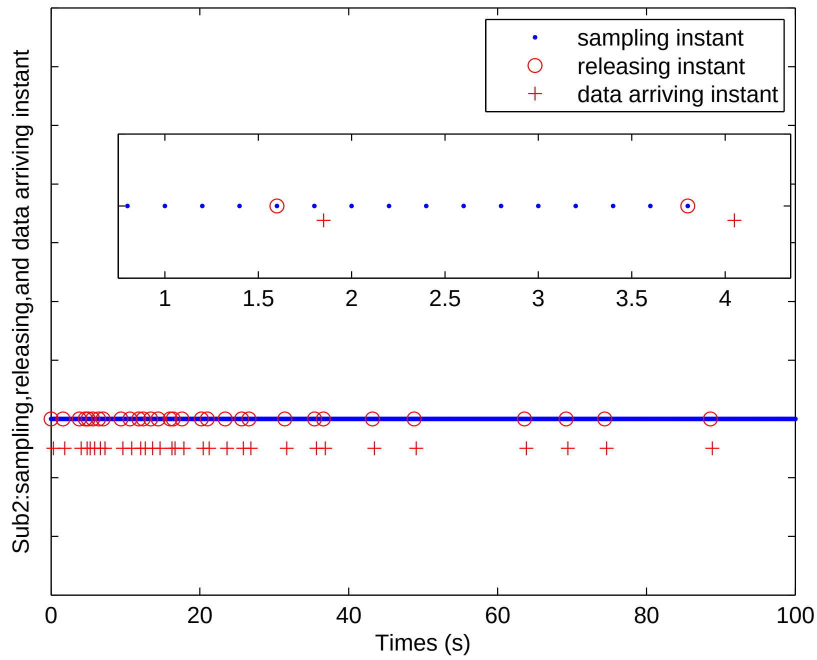

Figure 4.9: Sampling, releasing and data arriving instants of subsystem 2

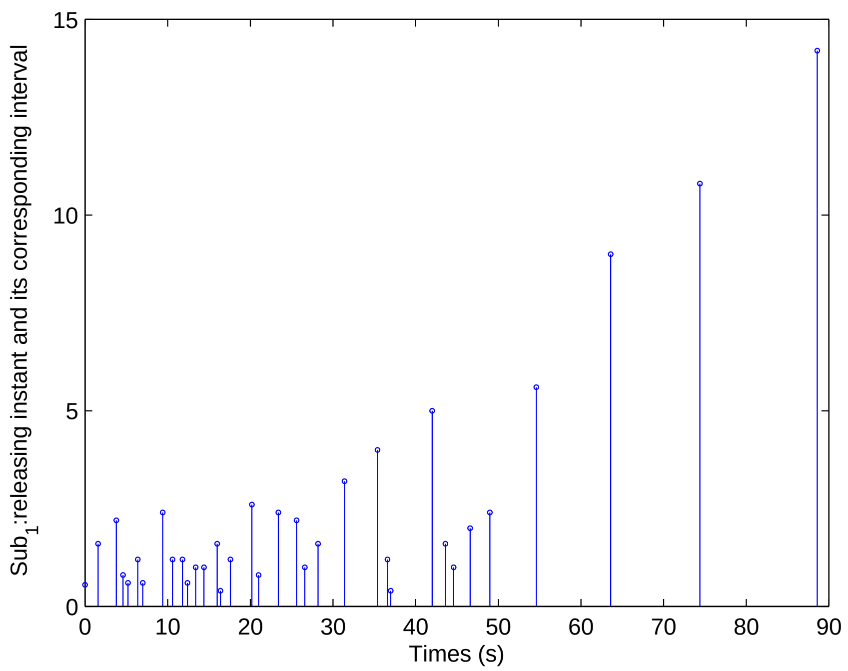

Figure 4.10: The release instants and intervals of subsystem 1

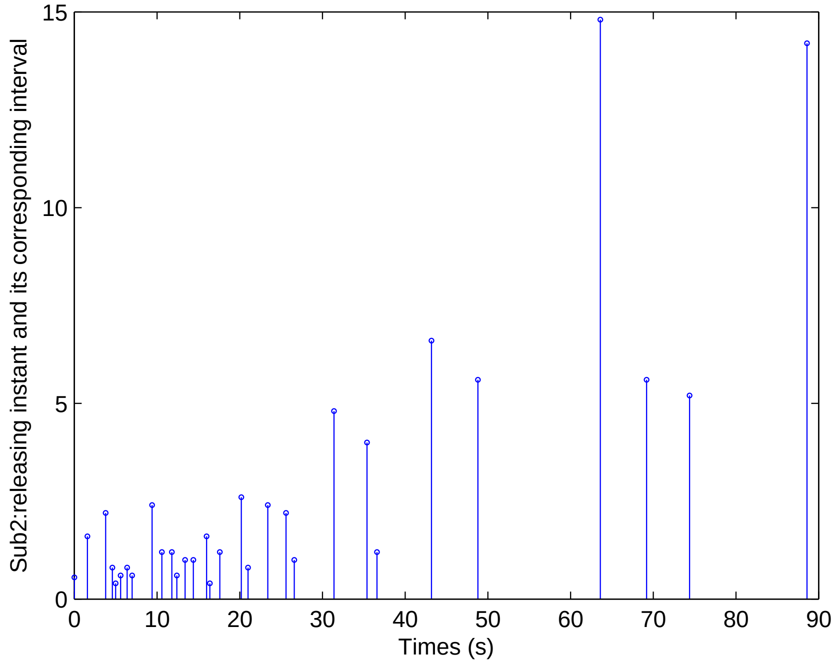

Figure 4.11: The release instants and intervals of subsystem 2

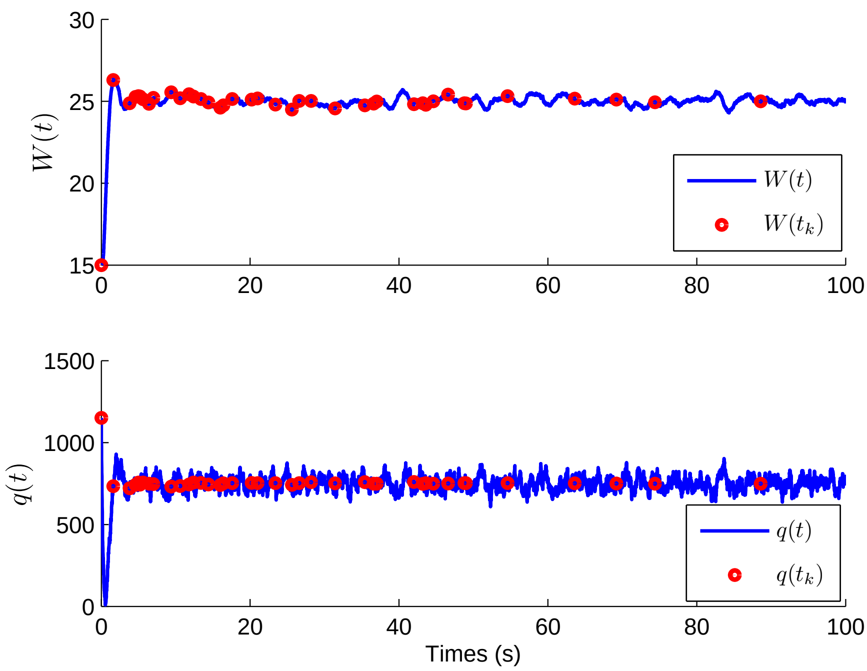

Figure 4.12: The state responses of the communication network

4.5 Conclusion

A novel distributed event-triggered transmission strategy for a large-scale distributed system has been proposed. In this strategy, the event-triggered condition not only considers the states of the distributed systems but also considers the network dynamics. The event-triggered condition is only checked periodically at the sampling instants, which avoids the *Zeno* phenomenon. Following this strategy, a novel middleware is developed to schedule the sampled data based on a fluid flow model. Then, a distributed protocol is proposed such that the distributed networked system can be transformed into a linear system with two time-varying delays. Then, a sufficient condition has been obtained such that the distributed system is asymptotically stable with a prescribed H_∞ noise attenuation performance. Correspondingly, based on this condition, the method to determine the controller gain for the distributed system and the congestion controller gains of the communication network is presented. Finally, an example is used to illustrate the effectiveness of the proposed method.

Chapter 5

Distributed H_∞ filtering for networked systems based on an Information Dispatching Middleware

5.1 Introduction

In this chapter, the proposed Information Dispatching Middleware is used to develop a framework to investigate distributed sensing and H_∞ filtering issues of networked systems.

As discussed in Chapter 3, during the past decade, H_∞ filtering for networked systems has attracted considerable attention due to its theoretical and practical significance in the areas of system control and signal processing. Due to the widespread availability of cheap network capable devices, distributed filtering has been undergoing a revolution in recent years. For distributed filtering, the asynchronous sensor network is comprised of a large number of sensor nodes with computing and communication capabilities. In contrast to systems with a single sensor, each individual sensor node in a sensor network not only gathers measurement information from itself, but also gathers information from its neighbouring sensors according to the sensing topology of the sensor network. The past decade has seen successful applications of distributed filtering in a wide range of practical areas such as military sensing, physical security, distributed robots and manufacturing automation. A schematic diagram for distributed filtering in a networked system is shown in Figure 5.1.

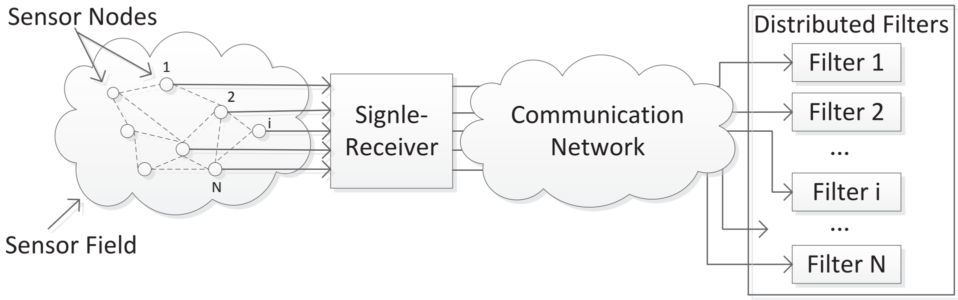

Figure 5.1: Distributed sensing and filtering for a networked system

Recently, theoretical research on distributed H_∞ filtering has attracted increasing attention. To implement such filtering for networked systems, three major issues need to be taken into account. The first issue is the complicated coupling between one specific sensor and its neighbouring sensors. Compared with a single sensor, smart sensors are usually deployed densely in the region of interest for the control plant. Each sensor, equipped with a radio transceiver, has the ability to gather the measurement information of the control plant not only from itself, but also from its neighbouring sensors according to the sensing topology of the sensor nodes. The coupling among the variety of sensors makes the designing and analysis of control systems complex [143], [144], [145]. The second issue is the network-induced delay generated by the shared communication network between the sensor nodes and the distributed filters. It is well known that the most significant delays are network-induced delays and these delays are regarded as the main sources of performance degradation and/or even divergence of the implemented filtering design [146], [147], [148], [149], [150]. The third issue is how to efficiently use the limited shared communication network resources while maintaining the quality of service (QoS) to avoid network congestion, which is the main cause of network-induced delays and packet dropouts [151], [152]. Therefore, considering these three issues, the study of distributed H_∞ filtering for networked systems with a wide variety of smart sensors is a significant challenge, and which motivates the current study.

In the past decade, employing an event-triggering scheme as a means of reducing the communication traffic load of a network has been paid considerable attention, see for example [123], [124], [125] ,[126] and the references therein. From most of these results, the event-triggering conditions are based on the current measurement output of the control system, the error between the current sampled data and the latest transmitted data. However, few results consider the dynamics of the communication network, which dynamically reflects different levels of quality of service (QoS) of the communication network. It should be noted that the QoS of the communication network affects the performance of control systems, whose design in turn influences the QoS of the communication network [4], [104], [153], [154]. Hence, under event-triggered schemes, a successful implementation of distributed H_∞ filtering for networked systems over communication networks requires the codesign of filters and network congestion control strategies. Furthermore, the transmitted data determined by the event-triggering scheme is so important for the performance of the whole networked systems that it is necessary to choose suitable network protocols to transmit the released data successfully [61], [155]. Therefore, it is of significance to study the distributed H_∞ filtering of networked systems considering both H_∞ filtering performance and network dynamics. This motivates us to develop a framework to provide a tradoff between the quality of performance (QoP) and the quality of service (QoS) of the communication networks.

Based on the above discussion, in this chapter, a novel framework is established by constructing an Information Dispatching Middleware to investigate the distributed H_∞ filtering for networked systems with multiple smart sensors. Both an event-triggered scheme and network dynamics are abstracted and modelled within the proposed Information Dispatching Middleware. In the middleware, two modules named the Information Selection Module and the Congestion Avoidance Module are developed. The Information Selection Module aims to regulate the transmission of the sampled data in terms of event-detectors with predefined event-triggering conditions. The Congestion Avoidance Module is used to schedule packets including those sampled data released by the Information Selection Module to the distributed H_∞ filters. With the proposed middleware, a tradeoff between the desired distributed H_∞ filtering performance and communication network resources utilisation is obtained. Based on the proposed framework, the filtering error system is modelled as a linear system with two time-varying delays. A sufficient condition such that the filtering error system is asymptotically stable with a prescribed H_∞ level, is derived. Correspondingly, distributed H_∞ filters, congestion controllers and the event-triggered scheme can be codesigned in terms of a set of linear matrix inequalities (LMIs). Finally, a numerical example is given to demonstrate the effectiveness of the proposed results.

5.2 Problem formulation and modelling

In this section, we present a framework for distributed H_∞ filtering by taking network dynamics into account. To begin with, we define the sensor network topology by using graph theory.

5.2.1 Sensor network topology

The topology of a sensor network considered in this chapter is represented by a directed graph $\mathcal{G} = (\mathscr{V}, \mathscr{E}, \mathscr{A})$, where $\mathscr{V} = \{1, 2, \cdots, \varrho\}$ is an index set of ϱ sensor nodes; $\mathscr{E}$ is the set of edges, which is a subset of $\mathscr{V} \times \mathscr{V}$; and $\mathscr{A} = (a_{ij})_{n\times n}$ is the nonnegative adjacency matrix associated with the edges of the graph. We denote by (i, j) $(i, j \in \mathscr{V})$ the edge of the graph originating at node i and ending at node j. $a_{ij} > 0$ if and only if $(i, j) \in \mathscr{E}$, which means that there is information transmission

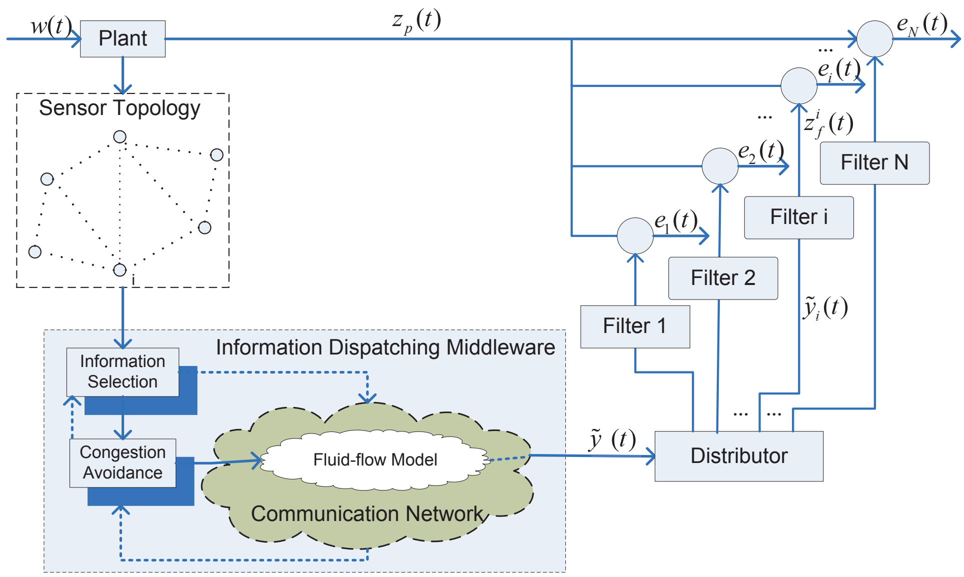

Figure 5.2: A framework for distributed H_∞ filtering with an Information Dispatching Middleware

from node i to node j. In this chapter, we set $a_{ii} = 1$ $(i \in \mathscr{V})$, which indicates that all the sensor nodes are self-connected, i.e. $(i, i) \in \mathscr{E}$. If $(j, i) \in \mathscr{E}$ $(j \neq i)$, then node j is called a neighbour of node i. The set of node i's neighbours and itself is denoted by $\mathcal{N}_i \triangleq \{j \in \mathscr{V} | (j, i) \in \mathscr{E}\}$, which means that in the sensor network, the sensor node i receives the information from the nodes $j \in \mathcal{N}_i$.

5.2.2 A framework for distributed H_∞ filtering

The framework for distributed H_∞ filtering is shown in Figure 5.2, where the plant is considered as a continuous-time linear system described by

$$\begin{cases} \dot{x}_p(t) = A_p x_p(t) + B_p w_p(t) \\ z_p(t) = E_p x_p(t) \\ x_p(t_0) = \phi_0 \end{cases} \tag{5.1}$$

where $x_p(t) \in \mathbb{R}^n$ is the system state vector; $z_p(t)$ is the controlled output vector; $w_p(t) \in \mathbb{R}^l$ is the exogenous disturbance vector belonging to $\mathcal{L}_2[0, \infty)$; ϕ_0 is the initial condition; and A_p, B_p and E_p are known real matrices with appropriate dimensions. The other components in Figure 5.2 are described in detail as

Sensor topology with ϱ nodes

Suppose that ϱ sensor nodes are connected according to the graph $\mathcal{G}$. Each sensor node measures the physical plant's output and receives signals from its neighbouring nodes. Specifically, at node i ($i \in \mathscr{V}$), the measurement output from the physical plant is given by

$$y_i(t) = C_p^i x_p(t) \tag{5.2}$$

where $y_i(t) \in \mathbb{R}^m$ and C_p^i is a known real matrix. Thus, the received signal of node i from the physical plant and its neighbouring nodes can be expressed as

$$\bar{y}_i(t) = \sum_{j \in \mathcal{N}_i} a_{ji} y_j(t) = \sum_{j \in \mathcal{N}_i} a_{ji} C_p^j x_p(t) \tag{5.3}$$

The signal $\bar{y}_i(t)$ ($i \in \mathscr{V}$) is transmitted through a communication network so that an H_∞ filter corresponding to the sensor node i is designed to estimate the system state. The transmission task is completed via the following Information Dispatching Middleware (IDM).

The Information Dispatching Middleware (IDM)

An important feature of the framework is the introduction of an IDM. Based on network characteristics, the IDM dynamically regulates the network traffic to choose "necessary" data packets to be transmitted such that some desired H_∞ performance of the resultant filtering error system can be ensured. The flow chat of the mechanism of the IDM is as shown in Figure 5.3. The IDM consists of two modules, which are described below.

1) The Information Selection Module (ISM).

The Information Selection Module (ISM) includes ϱ nodes corresponding to the ϱ sensor nodes above. The i-th node of ISM is shown in Figure 5.4.

In this figure, the Receiver i receives the signal $\bar{y}_i(t)$ from the sensor node i and the network dynamic signal $\tilde{x}(t)$ from the communication network. The Sampler

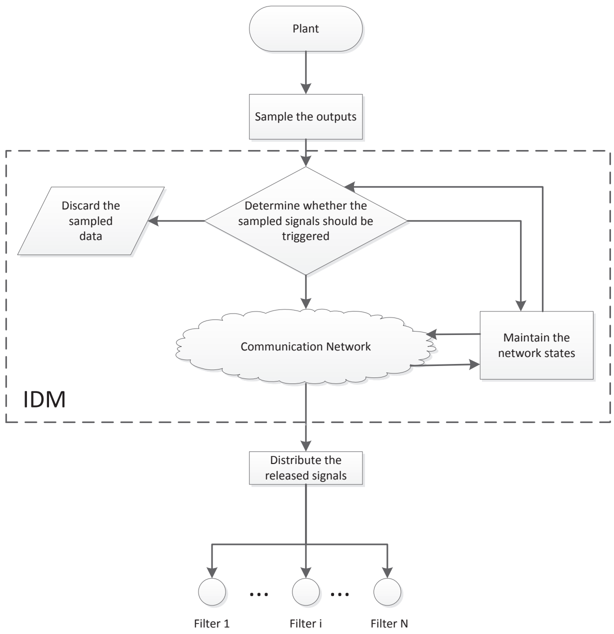

Figure 5.3: The mechanism of the IDM

Figure 5.4: The i-th node in the Information Selection Module

i is used to sample the received signal $(\bar{y}_i(t), \tilde{x}(t))$ with a period $h > 0$, where h is a constant. Whether or not the sampled signal is transmitted through a communication network is determined by the event-trigger processor ETP i. The ETP i consists of a buffer and an event-trigger. The buffer is used to store the latest released signal $\bar{y}_i(t_{i,k}h)$ and the current sampled signal $\bar{y}_i((t_{i,k}+l)h)$, where l is a positive integer. In order to define an efficient event-triggering scheme (ETS) in the i-th ETP, we introduce an error function $e_i(t^l_{i,k}h)$ and an evaluation function $\psi_i(t^l_{i,k}h)$, where $t^l_{i,k}h = t_{i,k}h + lh$. The error function $e_i(t^l_{i,k}h)$ reflecting the error between the current sampled signal $\bar{y}_i(t^l_{i,k}h)$ and the latest released signal $\bar{y}_i(t_{i,k}h)$ is given by

$$e_i(t^l_{i,k}h) = \bar{y}_i(t_{i,k}h) - \bar{y}_i(t^l_{i,k}h) \tag{5.4}$$

where $\bar{y}_i(t^l_{i,k}h)$ is the current sampled signal and $\bar{y}_i(t_{i,k}h)$ is the latest triggered signal.

The evaluation function $\psi_i(t^l_{i,k}h)$ is defined based on two important parameters: the current QoS index $\delta_i(t^l_{i,k}h)$ of the communication network and a certain scale of error signal $e_i(t^l_{i,k}h)$. Inspired by [156] and [157], the QoS index $\delta_i(t^l_{i,k}h)$ specification encompasses on (i) the level of the service α, (ii) and the flow performance, which are reflected by the window size $W(t^l_{i,k}h)$ and queue size $q(t^l_{i,k}h)$. Specifically, the QoS index, which is closely related to the network dynamics, is defined as

$$\delta_i(t^l_{i,k}h) = \mathrm{col}\{\frac{\alpha}{|W(t^l_{i,k}h) - W_0| + W_0}, \frac{\alpha}{|q(t^l_{i,k}h) - q_0| + q_0}\} \tag{5.5}$$

where $W(t^l_{i,k}h)$ and $q(t^l_{i,k}h)$ are the window size and queue length at the time instant $t^l_{i,k}h$, respectively; W_0 and q_0 are the the ideal window size and queue size; α is the QoS level of the service for the control system. From the definition of the QoS, one can see that when the window size and the queue size are approaching their ideal size, the QoS index increases, which means that the QoS of the network is improving. Thus, the scheduling function $\psi_i(t^l_{i,k}h)$ can be given in the following form of

$$\psi_i(t^l_{i,k}h) = \delta_i(t^l_{i,k}h)\|e_i(t^l_{i,k}h)\| \tag{5.6}$$

Remark 5.1. *The proposed evaluation function $\psi_i(t^l_{i,k}h)$ can be used to assign the priority to schedule the sampled signals dynamically. Compared with the Large Error First (LEF) scheduling algorithm derived by Yepez et al. in [114], $\psi_i(t^l_{i,k}h)$ is constructed not only from the QoS of the communication network, but also from the characteristics of the sampled signal. It is helpful to construct an event-triggered scheme, which determines whether or not the current sampled signal $\bar{y}_i(t^l_{i,k}h)$ should be triggered.*

Suppose that the latest signal released by ETP i is denoted by $(\bar{y}_i(t_{i,k}h), \tilde{x}(t_{i,k}h))$, and the current signal sampled from the node i is $(\bar{y}_i(t^l_{i,k}h), \tilde{x}(t^l_{i,k}h))$. Denote

$$f_i(t^l_{i,k}) = \Im_i(t^l_{i,k})^T \Omega_i \Im_i(t^l_{i,k}) - \lambda \wp_i(t^l_{i,k})^T \Omega_i \wp_i(t^l_{i,k}) \tag{5.7}$$

where $\lambda > 0$ is an event-triggering threshold parameter and Ω_i is a weighting matrix satisfying $\Omega_i = \Omega_i^T > 0$, and

$$\Im_i(t^l_{i,k}) = \begin{bmatrix} e_i(t^l_{i,k}h) \\ \psi_i(t^l_{i,k}h) \end{bmatrix}, \ \wp_i(t^l_{i,k}) = \begin{bmatrix} \bar{y}_i(t^l_{i,k}h) \\ \tilde{x}(t^l_{i,k}h) \end{bmatrix}.$$

Then the ETS involved in ETP i can be described as follows.

If the current sampled signal $(\bar{y}_i(t^l_{i,k}h), \tilde{x}(t^l_{i,k}h))$ satisfies $f_i(t^l_{i,k}h) > 0$, then

- the current signal $(\bar{y}_i(t^l_{i,k}h)$ together with its time-stamp $t^l_{i,k}$ and the node number i is encapsulated into a data packet $(i, t^l_{i,k}, \bar{y}_i(t^l_{i,k}h))$;

- this data packet is immediately released to the communication network by ETP i;

Otherwise, the current sampled signal is discarded directly.

From the description of the above ETS, it can be clearly seen that the time sequence $\{t_{i,1}h, t_{i,2}h, \cdots, t_{i,k}h, \cdots\}$ indicating when ETP i releases data packets can be obtained by

$$\begin{cases} t_{i,k+1} = t_{i,k} + \min\left\{ \ l \ \middle| \ f_i(t^l_{i,k}) > 0, \ l = 1, 2, \cdots \right\} \\ t_{i,0} = 0, \quad k = 1, 2, \cdots. \end{cases} \tag{5.8}$$

To demonstrate some characteristics of the proposed ETS (5.7), which is different to existing event-triggered schemes, we choose

$$\Omega^i = \begin{bmatrix} \Omega^i_{11} & \Omega^i_{12} \\ \Omega^{iT}_{12} & \Omega^i_{22} \end{bmatrix} \tag{5.9}$$

then, from (5.7), the proposed event-triggering scheme can be expressed as

$$f'_i(t^l_{i,k}) = \Pi^i_1 + \Pi^i_2 + \Pi^i_3. \tag{5.10}$$

where

$$\begin{cases} \Pi_1^i = e_i(t_{i,k}^l h)^T \Omega_{11}^i e_i(t_{i,k}^l h) - \lambda_{i1} \bar{y}_i(t_{i,k}^l h)^T \Omega_{11}^i \bar{y}_i^T(t_{i,k}^l h) \\ \Pi_2^i = \psi_i(t_{i,k}^l h)^T \Omega_{12}^{iT} e_i(t_{i,k}^l h) + e_i(t_{i,k}^l h)^T \Omega_{12}^i \psi_i(t_{i,k}^l h) \\ \qquad - \lambda(\tilde{x}(t_{i,k}^l h) \Omega_{12}^{iT} \bar{y}_i(t_{i,k}^l h) + \bar{y}_i(t_{i,k}^l h)^T \Omega_{12}^i \tilde{x}(t_{i,k}^l h)) \\ \Pi_3^i = \psi_i(t_{i,k}^l h)^T \Omega_{22}^i \psi_i(t_{i,k}^l h) - \lambda_{i1} \tilde{x}(t_{i,k}^l h)^T \Omega_{22}^i \tilde{x}(t_{i,k}^l h) \end{cases} \tag{5.11}$$

Regarding the sampled signal $\bar{y}_i((t_{i,k}+l)h)$ of the distributed sensor i, whether or not it is triggered to be encapsulated into the data packet $(i, t_{i,k+1}, \bar{y}_i((t_{i,k+1})h)$ depends on the following cases:

Case 1: $f_i'(t_{i,k}^l) > 0$, which means the sampled signal $\bar{y}_i(t_{i,k}^l h)$ is triggered, then it yields

if $\Pi_1^i + \Pi_3^i \leq 0$ which means that the coupling between the system and the communication network has a certain influence on the triggered signals.

else if $\Pi_1^i > 0$ & $\Pi_3^i > 0$ which shows that both the system dynamics and signal characteristics evaluated by the function $\psi_i(t)$ jointly decide that the signal should be triggered.

else if $Pi_1^i > 0$ & $\Pi_3^i \leq 0$ which means that system dynamics determine that the signal should be triggered.

else $\Pi_1^i \leq 0$ & $\Pi_3^i > 0$ which means that the signal is self triggered by its own characteristics.

Case 2: $f_i(t_{i,k}^l) \leq 0$, which means the sampled signal $\bar{y}_i(t_{i,k}^l h)$ is discarded, then we have

if $\Pi_1^i + \Pi_3^i > 0$ which means that the coupling between the system and the communication network has a certain influence on restraining the sampled signals.

else if $\Pi_1^i \leq 0$ & $\Pi_3^i \leq 0$ which shows that both the system dynamics and the signal characteristics jointly decide that the signal should be discarded.

else if $\Pi_1^i < 0$ & $\Pi_3^i \geq 0$ which means that system dynamics determine that the signal should be discarded.

else $\Pi_1^i \geq 0$ & $\Pi_3^i < 0$ which means that communication network congestion prevents the sampled signal from being triggered.

Observe that, Π_{i1} is related to the system outputs, while Π_{i3} is related to network dynamics. Compared with the event-triggered schemes in [140] and [158], it is clear that, under the above event-triggered scheme, whether or not an event is triggered is determined not only by the current sampled signal and the latest released signal but also by the current network dynamics. More specifically, if $\Pi_1^i \leq 0$, $\Pi_3^i > 0$, but $\Pi_1^i + \Pi_3^i > 0$, under this event-triggered scheme, the sampled data is released. The reason for this case is that although transmission of this data is not necessary for the control system, the QoS index and the flow performance (window size and queue size) illustrate that the network is not congested, which means that the QoS of the communication network is adequate to transmit more data. Consequently, the sampled data is transmitted, thus improving the performance of the network and utilising network resources more efficiently. On the other hand, if $\Pi_1^i > 0$, $\Pi_3^i \leq 0$, but $\Pi_1^i + \Pi_3^i < 0$, then the sampled data is not transmitted. This situation is due to the fact that although the sampled data should be transmitted to guarantee the control system's performance, the QoS and the flow dynamics show that the network is congested, and so no data can be transmitted.

Remark 5.2. *In this chapter, we employ the Large Error First scheduling policy, which is different from the scheduling policy in [159]. In [159], the scheduler uses a mean value of the states of the applications as a form of performance index. While based on event-triggering characteristics, it is more efficient to use the error of measurement outputs to assign priorities to different filters.*

Remark 5.3. *As shown in* (5.7), $\psi_i(t_{i,k}^l h)$ $(i = 1, 2, \cdots, \varrho)$ *is used in the event-*

triggering scheme with network dynamics to jointly decide whether or not the current sampled signal needs to be triggered. Moreover, from Π_{i3} *in* (5.11)*, this evaluated function* $\psi_i(t)$ *can be applied in the fluid-flow model as an external control strategy to avoid communication network congestion. Therefore, such a novel evaluated function* $\psi_i(t)$ *is a link to implement the codesign between networked systems and communication network under the proposed Information Dispatching Middle framework for investigation of the distributed* H_∞ *filtering issue.*

2) The Congestion Avoidance Module (CAM).

The Congestion Avoidance Module (CAM) is used to design a congestion control strategy for a transmission control protocol (TCP) communication network such that data-packets released from ISM can be successfully transmitted to a zero-order-hold to be held for filter design. To begin with, a dynamic model of TCP behaviour, which is based on a fluid-flow model, is given by [102]

$$\begin{cases} \dot{W}(t) = \dfrac{1}{\tau(t)} - \dfrac{W(t)}{2}\dfrac{W(t-\tau(t))}{\tau(t-\tau(t))}p(t-\tau(t)) \\ \dot{q}(t) = \begin{cases} -C(t) + \dfrac{N(t)}{\tau(t)}W(t), & q(t) > 0 \\ \max\left\{0, -C(t) + \dfrac{N(t)}{\tau(t)}W(t)\right\}, & q(t) = 0 \end{cases} \\ \tau(t) = \dfrac{q(t)}{C(t)} + T_p \end{cases} \tag{5.12}$$

where $W(t)$ is the TCP window size; $q(t)$ is the queue length; T_p is the propagation delay; $\tau(t)$ is the round-trip time, which represents the network-induced delay; $C(t)$ is the available link capacity; $N(t)$ is the number of TCP sessions; and $p(t)$ is the probability of packet marking satisfying $0 \leq p(t) \leq 1$. This model reflects a fundamental characteristic named 'additive increase and multiplicative decrease' in typical TCP/IP communication networks. Moreover, simulation shows that this model captures well the qualitative behavior of TCP traffic flows [160]. In the past decade, active queue management (AQM) has played a key role in controlling congestion in a TCP/IP communication network, because TCP with an AQM router can be considered as a feedback control system, allowing us to design congestion controllers using feedback control approaches. Consequently, a number of results on AQM-based congestion control have been reported in the literature [161], [113] and references therein.

Suppose that $N(t)$ is a constant N, and that $W(t)$ and $q(t)$ satisfy $0 \leq W(t) \leq \bar{W}$ and $0 \leq q(t) \leq \bar{q}$, where $\bar{W}$ and $\bar{q}$ denote maximum window size and buffer capacity, respectively. Take $(W(t), q(t))$ as the state and $p(t)$ as the input. For given network parameters (N, C_0, T_p) with C_0 being the nominal value of $C(t)$, the operating point (W_0, q_0, p_0) defined by $\dot{W}(t) = 0$ and $\dot{q}(t) = 0$ satisfies $W_0 \in [0, \bar{W}], q_0 \in [0, \bar{q}], p_0 \in [0, 1]$, and $\tau_0 = \frac{q_0}{C_0} + T_p, W_0 = \frac{\tau_0 C_0}{N}$ and $p_0 = \frac{2}{W_0^2}$. Denote $\tilde{x}(t) = \mathrm{col}\{W(t) - W_0, q(t) - q_0\}$, $\tilde{u}(t) = p(t) - p_0$ and $\tilde{v}(t) = \mathrm{col}\{C(t) - C_0, C(t - \tau_0) - C_0\}$. Then at the operating point (W_0, q_0, p_0), the nonlinear fluid-flow model (5.12) can be linearised in the following form

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{A}_d\tilde{x}(t - \tau(t)) + \tilde{B}\tilde{u}(t - \tau(t)) + \tilde{D}\tilde{v}(t) \tag{5.13}$$

where

$$\tilde{A} = \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & -\frac{1}{\tau_0^2 C_0} \\ \frac{N}{\tau_0} & -\frac{1}{\tau_0} \end{bmatrix}, \qquad \tilde{A}_d = \begin{bmatrix} -\frac{N}{\tau_0^2 C_0} & \frac{1}{\tau_0^2 C_0} \\ 0 & 0 \end{bmatrix},$$

$$\tilde{B} = \begin{bmatrix} -\frac{\tau_0^2 C_0}{2N^2} \\ 0 \end{bmatrix}, \qquad \tilde{D} = \begin{bmatrix} \frac{\tau_0 - T_p}{\tau_0^2 C_0} & -\frac{\tau_0 - T_p}{\tau_0^2 C_0} \\ -\frac{T_p}{\tau_0} & 0 \end{bmatrix}.$$

From the codesign point of view, in (5.13), we choose the variation of the available link capacity $\delta C(t) = C(t) - C_0$ as the networked system output error between the most recent sampled signal and the latest released signal. This is mapped to schedule the related short-lived information, such as state of system variation, alarm information and so on. From the linearisation process of the fluid-flow model, the delayed disturbance $\delta C(t - \tau(t)) = C(t - \tau(t)) - C_0$ also exists in (5.13). For the distributed filtering issue, we construct $\delta C_i(t)$ and $\delta C_i(t-\tau(t))$ as

$$\begin{cases} \delta C_i(t) = L_{i1}(e_i(t)) \\ \delta C_i(t-\tau(t)) = L_{i2}(e_i(t)) \end{cases}$$

where $L_{i1},\ L_{i2}$ are functions that map the networked system output error to its related short-lived information.

To incorporate the above details into the codesign process, we define the disturbance $\tilde{v}_i(t)$ of the fluid-flow model at the equilibrium as

$$\tilde{v}_i(t) = [\delta C_i(t), \delta C_i(t-\tau(t))]^T = [L_{i1}(e_i(t)), L_{i2}(e_i(t))]^T.$$

then

$$\tilde{v}(t) = \sum_{i=1}^{m} [L_{i1}(e_i(t)), L_{i2}(e_i(t))]^T. \tag{5.14}$$

In this chapter, it is assumed that

$$\tilde{v}(t) = \sum_{i=1}^{m} [F_{i1}, F_{i2}]^T e_i(t) \tag{5.15}$$

where $F_{i1},\ F_{i2} \in \mathbb{R}^{1\times m}$ are known real matrices.

From [113], it is true that the linearisation model in (5.13) offers an effective approach to the congestion control of a TCP/IP communication network. A state feedback controller can be designed such that the system (5.13) is asymptotically stable. Under the designed controller, the network dynamics of the TCP/IP communication network remain stable at the equilibrium point, which means that a satisfactory quality of service of the communication network can be ensured. On the other hand, the evaluation functions $\psi_i(t^l_{i,k}h)$ $(i = 1, 2, \cdots, \varrho)$ introduced in ISM are used to dynamically allocate transmission priority of those data-packets triggered from ETP i $(i = 1, 2, \cdots, \varrho)$. Hence, the dynamic evaluation functions, regarded as an external control input, can be used to design suitable congestion controllers to *enhance* the stability of network dynamics. To this end, we slightly modify the system (5.13) to become

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{A}_d\tilde{x}(t-\tau(t)) + \tilde{B}\tilde{u}_1(t-\tau(t)) + \tilde{B}\tilde{u}_2(t) + \tilde{D}\tilde{v}(t). \tag{5.16}$$

The objective of the CAM is to design congestion controllers in the form of

$$\begin{cases} \tilde{u}_1(t) = K_1 x(t) \\ \tilde{u}_2(t) = \sum_{i=1}^{\varrho} K_{2i}\psi_i(t), \;\; t \in [t_{i,k}h, t_{i,k+1}h) \end{cases} \tag{5.17}$$

such that the closed-loop system formed by (5.16) and (5.17) is asymptotically stable.

The zero-order-hold (ZOH)

Under the proposed Information Dispatching Middleware, the zero-order-holder (ZOH) of each distributed H_∞ filter is event-driven. More specifically, the ZOH of filter i receives a data packet $(i, t_{i,k}, \bar{y}_i(t_{i,k}h)$ from the sensor node i through the IP-based communication network. Due to the effects of the communication network, for the data packet $(i, t_{i,k}, \bar{y}_i(t_{i,k}h)$, the network-induced delay induced by the fluid-flow model is $\tau(t_{i,k}) = \frac{q(t_{i,k}h)}{C_0} + T_p$, where the link capacity C_0 is fixed and $\tau_m \le \tau(t_{i,k}) \le \tau_M$, $k_i \in \mathbb{N}$, with τ_m the fixed propagation delay T_p and τ_M the maximum allowable transmission network-induced delay. Then, the ZOH of the filter i receives its data packets with the time sequence $\{t_{1_i}h + \tau(t_{1_i}), t_{2_i}h + \tau(t_{2_i}), \cdots, t_{i,k}h + \tau(t_{i,k}), \cdots\}$. This means that the ZOH i keeps the data available until the new data arrives at $t_{k_i+1}h + \tau(t_{k_i+1}h)$. Hence, the holding zone of the filter i's ZOH can be expressed as

$$[t_{0_i} + \tau_0, \;\; +\infty) = \bigcup_{k=1}^{\infty} [t_{i,k}h + \tau(t_{i,k}), \;\; t_{i,k+1}h + \tau(t_{i,k+1}))$$

where $t_0^i + \tau_0^i = t_{1_i}h + \tau(t_{1_i})$.

Distributed H_∞ filters

In this chapter, the distributed H_∞ filter i can be designed as a linear dynamic filter in the form of

$$\begin{cases} \dot{x}_f^i(t) = A_f^i x_f^i(t) + B_f^i \bar{y}_i(t_{i,k}h) \\ z_f^i(t) = C_f^i x_f^i(t). \end{cases} \tag{5.18}$$

Using the proposed Information Dispatching Middleware, once the ZOH receives the packet $(i, t_{i,k}, \bar{y}_i(t_{i,k}h)$, it actuates the distributed H_∞ filter i with the signal

$\bar{y}_i(t_{i,k}h)$. Hence, we have

$$\tilde{y}(t) = \bar{y}(t_{i,k}h),\ t_{i,k}h + \tau(t_{i,k}h) \le t < t_{i,k+1}h + \tau(t_{i,k+1}h) \tag{5.19}$$

Substituting (5.19) into (5.18) yields

$$\begin{cases} \dot{x}_f^i(t) = A_f^i x_f^i(t) + B_f^i \tilde{y}(t) \\ z_f^i(t) = C_f^i x_f^i(t) \end{cases} \tag{5.20}$$

where $t_{i,k}h + \tau(t_{i,k}h) \le t < t_{i,k+1}h + \tau(t_{i,k+1}h)$, and A_f^i, B_f^i, C_f^i are filter parameters to be determined.

5.2.3 The model of the distributed H_∞ filtering error system

Based on the previous analysis, the distributed H_∞ filtering problem for networked systems is investigated by using the proposed the Information Dispatching Middleware including the Information Selection Module and the Congestion Avoidance Module.

We choose a set of sampled instants as $\{sh|s \in \mathbb{N}\}$ and a set of released instants of each event-trigger processor as $\{t_k h|k \in \mathbb{N}\}$. Define the corresponding following sampled instants for the current released instant $t_k h$ as $t_k^l h = t_k h + lh$, with $t_k^l = 0, \ldots, L$. Set $t_1 h$ as the initial released instant and the next transmission instant for the $t_k h$ instant can be derived as

$$t_{k+1}h = t_k h + \inf_{l \ge 1}\{lh|\forall f_i(t_k^l h) > 0\}. \tag{5.21}$$

The holding zone of the distributor is $[t_1 h, \infty) = \bigcup_{k=1}^{\infty}[t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$. The interval $[t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$ can be divided into the following subsets as shown in Figure 5.5.

As can be seen from Figure 5.5, we have

$$[t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h)) = \bigcup_{l=0}^{L} \Theta_k^l. \tag{5.22}$$

where if $L = 0$, $\Theta_k^l = [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$; if $L \in \mathbb{N}$ exists, such that $(t_k + L)h + \tau(t_k h) \le t_{k+1}h + \tau(t_{k+1}h)$ and $(t_k + L + 1)h + \tau(t_k h) > t_{k+1}h + \tau(t_{k+1}h)$.

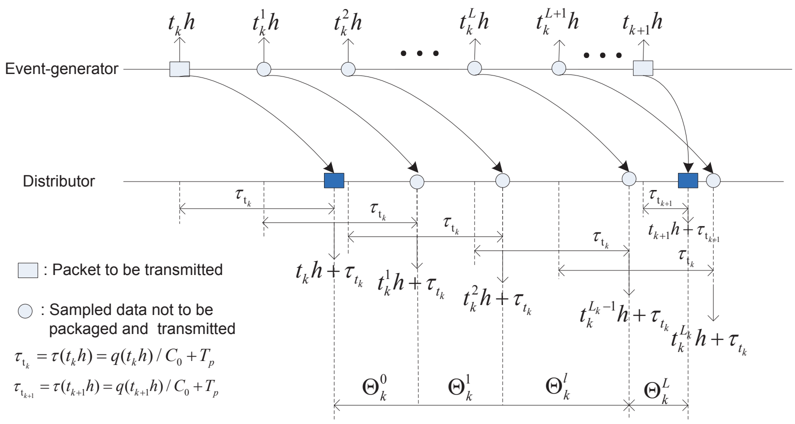

Figure 5.5: The subsets of the time interval $[t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$

$\Theta_k^l = [t_k^l h + \tau(t_k h), t_k^l h + h + \tau(t_k h))$, with $t_k^l = t_k + lh$, when $l = 0, 1, \cdots, L-1$; and $\Theta_k^l = [(t_k + L_k - 1)h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$ when $l = L$.

Then, we define

$$d(t) = t - t_k^l h, t \in \Theta_k^l, l \in \{0, 1, \cdots, L\}. \tag{5.23}$$

If $L = 0$, we have

$$\tau_m \leq \tau(t_k h) \leq d(t) \leq (t_{k+1} - t_k)h \leq h + \tau_M; \tag{5.24}$$

If $L \neq 0$, it yiels

$$\tau_m \leq \tau(t_k h) \leq d(t) \leq h + \tau(t_k h) \leq h + \tau_M. \tag{5.25}$$

It is clear that $d(t)$ is a piecewise-line function satisfying

$$\tau_m \leq \tau(t_k h) \leq d(t) \leq h + \tau_M. \tag{5.26}$$

From (5.4), we obtain

$$e_i(t_k^l h) = \bar{y}_i(t_{i,k}h) - \bar{y}_i(t_k^l h), t \in \Theta_k^l \tag{5.27}$$

where $\bar{y}_i(t_{i,k}h)$ is the latest released sampled data in the event-detector i for the corresponding instant $t_k h$.

Then, the input signal is obtained as

$$\tilde{y}_i(t) = \bar{y}_i(t_k h) = \bar{y}_i(t_k^l h) + e_i(t_k^l h) \tag{5.28}$$

where $t \in [t_k h + \tau(t_k h), t_{k+1}h + \tau(t_{k+1}h))$

Denote

$$\begin{aligned}
\xi(t) &= [\bar{x}_p(t), \tilde{x}(t), \bar{x}_f(t)]^T, \ \ \bar{x}_p(t) = \mathrm{col}_N\{x_p(t)\}, \\
\bar{x}_f(t) &= \mathrm{col}_N\{x_f^i(t)\}, \quad \varphi(t) = [\bar{e}_k(t-d(t)), \bar{\psi}(t-d(t))]^T, \\
\bar{e}_k(t-d(t)) &= \mathrm{col}_N\{e_k^i(t-d(t))\}, \ \ \bar{\psi}(t) = \mathrm{col}_N\{\psi_i(t-d(t))\}, \\
w(t) &= \bar{w}_p(t), \quad \bar{w}_p(t) = \mathrm{col}_N\{w_p(t)\}.
\end{aligned}$$

Combining (5.1), (5.15), (5.15), (5.16), (5.17) and (5.20), the filtering error system can be expressed as

$$\begin{cases}
\dot{\xi}(t) = A\xi(t) + A_d H\xi(t-\tau(t)) + B_1 H\xi(t-d(t)) \\
\qquad\qquad + B_2\varphi(t) + B_3 w(t) \\
e(t) = E\xi(t) \\
\xi(\theta) = \mathrm{col}\{\xi(t_0), 0\}, \ \ \theta \in [-d_M, 0],
\end{cases} \tag{5.29}$$

for $t \in [t_k^l h + \tau(t_k^l h), t_k^{l+1}h + \tau(t_k^{l+1}h))$, where

$$A = \begin{bmatrix} \bar{A}_p & 0 & 0 \\ 0 & \tilde{A} & 0 \\ 0 & 0 & \bar{A}_f \end{bmatrix}, A_d = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{A}_d + \tilde{B}_1\tilde{K}_1 \\ 0 & 0 \end{bmatrix}, H = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \bar{B}_f(\mathcal{A}\otimes I)\breve{C} & 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0 & 0 \\ \tilde{D}\bar{F} & \tilde{B}_2\bar{K}_2 \\ \bar{B}_f & 0 \end{bmatrix}, B_3 = \begin{bmatrix} \bar{B}_p \\ 0 \\ 0 \end{bmatrix},$$

$$\begin{aligned}
&\bar{A}_p = \mathrm{diag}_N\{A_p\}, \bar{A}_f = \mathrm{diag}_N\{A_f^i\}, \breve{C} = \mathrm{vec}_N^1\{\mathrm{col}_N\{C_p^i\}\}, \\
&\tilde{F} = \mathrm{vec}_N\{\tilde{F}_i\}, \tilde{K}_2 = \mathrm{vec}_N\{\tilde{K}_{2i}\}, \bar{B}_p = \mathrm{col}_N\{B_p\}, \bar{E}_p = \mathrm{diag}_N\{E_p\}, \\
&\bar{B}_f = \mathrm{diag}_N\{B_f^i\}, \bar{C}_f = \mathrm{diag}_N\{C_f^i\}, E = [\bar{E}_p \ 0 \ -\bar{C}_f].
\end{aligned}$$

In this chapter, the distributed H_∞ filtering problem is stated as follows: given scalars τ_m, τ_M, d_m, d_M, $\lambda > 0$, and a prescribed level of disturbance attenuation $\gamma > 0$, design some suitable distributed filters in the form of (5.18), such that

- The filter error system (5.29) with $w(t) = 0$ is asymptotically stable;
- Under zero initial condition, the filtering error system achieves a prescribed H_∞ performance level γ, *i.e.* the filtering error $e(t)$ satisfies $\|e(t)\|_2 \leq \gamma \|w(t)\|_2$, for any nonzero $w(t) \in \mathcal{L}_2[0, \infty)$.

5.3 Analysis and synthesis

5.3.1 Distributed H_∞ filtering performance analysis

In this section, given the filter parameters in (5.18), a sufficient condition for H_∞ performance analysis of the filtering error system (5.29) based on the Lyapunov-Krasovskii functional approach will be presented in the following theorem.

To begin with, we first introduce two lemmas, which are useful to solve the above problem.

Lemma 5.1. *[162] Let $z(t) \in W[a,\ b)$ and $z(a) = 0$. Then for any $n \times n$-matrix $R > 0$, the following inequality holds*

$$\int_a^b z^T(\theta) R z(\theta) d\theta \leq \frac{4(b-a)^2}{\pi^2} \int_a^b \dot{z}^T(\theta) R \dot{z}(\theta) d\theta \tag{5.30}$$

Lemma 5.2. *[130] Let $\Phi = \Phi^T$, Γ_1 and Γ_2 be real matrices of appropriate dimensions, then*

$$\Phi + \Gamma_1 \Lambda(T) \Gamma_2 + \Gamma_2^T \Lambda^T(T) \Gamma_1^t < 0$$

for all $\Lambda(t)$ satisfying $\Lambda^T(t)\Lambda(t) \leq I$, if and only if there exists a scalar $\varepsilon > 0$ such that

$$\Phi + \varepsilon^{-1} \Gamma_1 \Gamma_1^t + \varepsilon \Gamma_2^T \Gamma_2 < 0.$$

In this chapter, these inequalities are employed to investigate the issue of distributed H_∞ filtering.

Theorem 5.1. *For given scalars τ_m, τ_M, d_m, d_M, $\lambda \in (0,\ 1)$ and $\gamma > 0$, the filtering error system (5.29) is asymptotically stable with an H_∞ performance γ, if there exist* *matrices* $\Omega > 0$, $P > 0$, $S > 0$, $Q_i > 0$, $R_i > 0$ $(i = 1, 2, 3, 4)$, *with appropriate dimensions such that*

$$\begin{bmatrix} \Psi_{11} & \Psi_{12} & \Psi_{13} & \Psi_{14} \\ \star & \Psi_{22} & \Psi_{23} & \Psi_{24} \\ \star & \star & \Psi_{33} & \Psi_{34} \\ \star & \star & \star & \Psi_{44} \end{bmatrix} < 0 \tag{5.31}$$

where

$$\Psi_{11} = \begin{bmatrix} \Lambda_{11} & H^T Q_3 & PA_d & 0 \\ \star & -Q_1+Q_2-Q_3-Q_4 & Q_4 & 0 \\ \star & \star & -2Q_4 & Q_4 \\ \star & \star & \star & -Q_2-Q_4 \end{bmatrix},$$

$$\Psi_{12} = \begin{bmatrix} H^T R_3 & PB_1 & 0 & PB_2 & PB_3 & \frac{\pi^2}{4} H^T S \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{13} = \begin{bmatrix} \tau_m A^T H^T Q_3 & \bar{\tau} A^T H^T Q_4 & d_m A^T H^T R_3 \\ 0 & 0 & 0 \\ \tau_m A_d^T H^T Q_3 & \bar{\tau} A_d^T H^T Q_4 & d_m A_d^T H^T R_3 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{14} = \begin{bmatrix} \bar{d} A^T H^T R_4 & \bar{d} A^T H^T S & E^T \\ 0 & 0 & 0 \\ \bar{d} A_d^T H^T R_4 & \bar{d} A_d^T H^T S & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{22} = \begin{bmatrix} \bar{R}_1 & R_4 & 0 & 0 \\ \star & \bar{R}_2 & R_4 & 0 \\ \star & \star & -R_2 - R_4 & 0 \\ \star & \star & \star & -H_1^T \Omega H_1 \end{bmatrix},$$

$$\Psi_{23} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \tau_m B_1^T H^T Q_3 & \bar{\tau} B_1^T H^T Q_4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \tau_m B_2^T H^T Q_3 & \bar{\tau} B_2^T H^T Q_4 \end{bmatrix},$$

$$\Psi_{24} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ d_m B_1^T H^T R_3 & \bar{d} B_1^T H^T R_4 & \bar{d} B_1^T H^T S & 0 \\ 0 & 0 & 0 & 0 \\ d_m B_2^T H^T R_3 & \bar{d} B_2^T H^T R_4 & \bar{d} B_2^T H^T S & 0 \end{bmatrix},$$

$$\Psi_{33} = \begin{bmatrix} -\gamma^2 I & 0 & \tau_m B_3^T H^T Q_3 & \bar{\tau} B_3^T H^T Q_4 \\ \star & -\frac{\pi^2}{4} S & 0 & 0 \\ \star & \star & -Q_3 & 0 \\ \star & \star & \star & -Q_4 \end{bmatrix},$$

$$\Psi_{34} = \begin{bmatrix} d_m B_3^T H^T R_3 & \bar{d} B_3^T H^T R_4 & \bar{d} B_3^T H^T S & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{44} = \begin{bmatrix} -R_3 & 0 & 0 & 0 \\ 0 & -R_4 & 0 & 0 \\ 0 & 0 & -S & 0 \\ 0 & 0 & 0 & -I \end{bmatrix}, G = \begin{bmatrix} (\mathcal{A} \otimes I)\check{C} & 0 \\ 0 & I \end{bmatrix}$$

$$\Lambda_{11} = PA + A^T P + H^T Q_1 H - H^T Q_3 H + H^T R_1 H - H^T R_3 H - \frac{\pi^2}{4} H^T S H,$$

$$\bar{\tau} = \tau_M - \tau_m, \bar{d} = d_M - d_m, \Omega = diag\{\Omega_1, \Omega_2, \cdots, \Omega_\varrho\},$$

$$\bar{R}_1 = -R_1 + R_2 - R_3 - R_4, \bar{R}_2 = -2R_4 + \lambda G^T H_2^T \Omega H_2 G,$$

$$H_1 = \begin{bmatrix} I_{1\times m} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & I_{1\times 2} & 0 & \cdots & 0 \\ 0 & I_{1\times m} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & I_{1\times 2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & \cdots & I_{1\times m} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & I_{1\times 2} \end{bmatrix},$$

$$H_2 = \begin{bmatrix} I_{1\times m} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & I_{1\times 2} \\ 0 & I_{1\times m} & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & I_{1\times 2} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & I_{1\times m} & 0 \\ 0 & 0 & \cdots & 0 & 0 & I_{1\times 2} \end{bmatrix}.$$

Proof. Choose a Lyapunov-Krasovskii functional candidate as

$$V(t,\xi(t)) = V_1(t,\xi(t)) + V_2(t,\xi(t)) + V_3(t,\xi(t)) \tag{5.32}$$

where

$$\begin{aligned} V_1(t,\xi(t)) &= \xi^T(t) P \xi(t) \\ V_2(t,\xi(t)) &= \int_{t-\tau_m}^{t} \xi^T(s) H^T Q_1 H \xi(s) ds \\ &\quad + \int_{t-\tau_M}^{t-\tau_m} \xi^T(s) H^T Q_2 H \xi(s) ds \\ &\quad + \tau_m \int_{t-\tau_m}^{t} \int_{s}^{t} \dot{\xi}^T(\theta) H^T Q_3 H \dot{\xi}(\theta) d\theta ds \end{aligned}$$

$$
\begin{aligned}
&+ \bar{\tau} \int_{t-\tau_M}^{t-\tau_m} \int_s^t \dot{\xi}^T(\theta) H^T Q_4 H \dot{\xi}(\theta) d\theta ds \\
V_3(t, \xi(t)) = &\int_{t-d_m}^{t} \xi^T(s) H^T R_1 H \xi(s) ds \\
&+ \int_{t-d_M}^{t-d_m} \xi^T(s) H^T R_2 H \xi(s) ds \\
&+ d_m \int_{t-d_m}^{t} \int_s^t \dot{\xi}^T(\theta) H^T R_3 H \dot{\xi}(\theta) d\theta ds \\
&+ \bar{d} \int_{t-d_M}^{t-d_m} \int_s^t \dot{\xi}^T(\theta) H^T R_4 H \dot{\xi}(\theta) d\theta ds \\
&+ \bar{d}^2 \int_{t-\rho(t)}^{t} \dot{\xi}^T(s) H^T S H \dot{\xi}(s) ds \\
&- \frac{\pi^2}{4} \int_{t-\rho(t)}^{t} \bar{\xi}^T(s) H^T S H \bar{\xi}(s) ds
\end{aligned}
$$

and $P > 0$, $Q_i > 0$, $R_i > 0$ $(i = 1, 2, 3, 4)$, $S > 0$, $\bar{\xi}(s) = \xi(s) - \xi(s - \rho(t))$ and $\rho(t) = d(t) - \tau(t_k h)$.

By using Lemma 5.1, we can derive that

$$
\bar{d}^2 \int_{t-\rho(t)}^{t} \dot{\xi}^T(s) H^T S H \dot{\xi}(s) ds \geq \frac{\pi^2}{4} \int_{t-\rho(t)}^{t} \bar{\xi}^T(s) H^T S H \bar{\xi}(s) ds
$$

which means that the Lyapunov-Krasovskii functional candidate $V(t, \xi(t)) \geq 0$.

Taking the derivative of $V(t, \xi(t))$ with respect to t along the trajectory of system (5.29) yields

$$
\dot{V}(t, \xi(t)) = \dot{V}_1(t, \xi(t)) + \dot{V}_2(t, \xi(t)) + \dot{V}_3(t, \xi(t)) \tag{5.33}
$$

where

$$
\begin{aligned}
\dot{V}_1(t, \xi(t)) &= 2\xi^T(t) P (A\xi(t) + A_d H \xi(t - \tau(t)) \\
&\quad + B_1 H \xi(t - d(t)) + B_2 \varphi(t - d(t)) + B_3 \omega(t)) \\
\dot{V}_2(t, \xi(t)) &= \xi^T(t) H^T Q_1 H \xi(t) \\
&\quad - \xi^T(t - \tau_m) H^T Q_1 H \xi(t - \tau_m)
\end{aligned}
$$

$$
\begin{aligned}
& + \xi^T(t-\tau_m)H^T Q_2 H\xi(t-\tau_m) \\
& - \xi^T(t-\tau_M)H^T Q_2 H\xi(t-\tau_M) \\
& + \tau_m^2 \dot{\xi}^T(t)H^T Q_3 H\dot{\xi}(t) \\
& - \tau_m \int_{t-\tau_m}^{t} \dot{\xi}^T(s)H^T Q_3 H\dot{\xi}(s)ds \\
& + \bar{\tau}^2 \dot{\xi}^T(t)H^T Q_4 H\dot{\xi}(t) \\
& - \bar{\tau} \int_{t-\tau_M}^{t-\tau_m} \dot{\xi}^T(s)H^T Q_4 H\dot{\xi}(s)ds \\
\dot{V}_3(t,\xi(t)) = & \ \xi^T(t)H^T R_1 H\xi(t) \\
& - \xi^T(t-d_m)H^T R_1 H\xi(t-d_m) \\
& + \xi^T(t-d_m)H^T R_2 H\xi(t-d_m) \\
& - \xi^T(t-d_M)H^T R_2 H\xi(t-d_M) \\
& + d_m^2 \dot{\xi}^T(t)H^T R_3 H\dot{\xi}(t) \\
& - d_m \int_{t-d_m}^{t} \dot{\xi}^T(s)H^T R_3 H\dot{\xi}(s)ds \\
& + \bar{d}^2 \dot{\xi}^T(t)H^T R_4 H\dot{\xi}(t) \\
& - \bar{d} \int_{t-d_M}^{t-d_m} \dot{\xi}^T(s)H^T R_4 H\dot{\xi}(s)ds \\
& + \bar{d}^2 \dot{\xi}^T(t)H^T S H\dot{\xi}(t) \\
& - \frac{\pi^2}{4}\xi^T(t)H^T S H\xi(t) \\
& + \frac{\pi^2}{4}\xi^T(t)H^T S H\xi(t-\rho(t)) \\
& + \frac{\pi^2}{4}\xi^T(t-\rho(t))H^T S H\xi(t) \\
& - \frac{\pi^2}{4}\xi^T(t-\rho(t))H^T S H\xi(t-\rho(t))
\end{aligned}
$$

From the definition of $t_{i,k}^l h$, it is clear that $t_{i,k}^l h \in [t_{i,k}h+\tau(t_{i,k}),\ t_{i,k+1}h+\tau(t_{i,k+1})$. Then from the event-triggering scheme (5.7), it is known that the next transmitted instant is $t_{k+1}h$, which means that

$$
\Im_i(t_{i,k}^l)^T \Omega_i \Im_i(t_{i,k}^l) < \lambda \wp_i(t_{i,k}^l)^T \Omega_i \wp_i(t_{i,k}^l) \tag{5.34}
$$

Then, for all distributed sensors, we have

$$\varphi^T(t_k^l h)H_1^T\Omega H_1\varphi(t_k^l h) < \lambda\xi^T(t_k^l h)G^T H_2^T\Omega H_2 G\xi(t_k^l h) \tag{5.35}$$

Thus

$$\begin{aligned}\dot{V}(t,\xi(t)) \leq V(t,\xi(t)) &+ e^T(t)e(t) - \gamma^2 w^T(t)w(t)\\ &- e^T(t)e(t) + \gamma^2 w^T(t)w(t)\\ &+ \lambda\xi^T(t_k^l h)G^T H_2^T\Omega H_2 G\xi(t_k^l h)\\ &- \varphi^T(t_k^l h)H_1^T\Omega H_1\varphi(t_k^l h).\end{aligned} \tag{5.36}$$

By using Jensen integral inequalities from [115] and [116], the following inequalities hold

$$\begin{gathered}
-\tau_m\int_{t-\tau_m}^{t}\dot{\xi}^T(s)H^TQ_3H\dot{\xi}(s)ds \leq\\
\begin{bmatrix}\xi(t)\\ H\xi(t-\tau_m)\end{bmatrix}^T\begin{bmatrix}-H^TQ_3H & H^TQ_3\\ \star & -Q_3\end{bmatrix}\begin{bmatrix}\xi(t)\\ H\xi(t-\tau_m)\end{bmatrix},\\
-(\tau_M-\tau_m)\int_{t-\tau_M}^{t-\tau_m}\dot{\xi}^T(s)H^TQ_4H\dot{\xi}(s)ds \leq\\
\begin{bmatrix}H\xi(t-\tau_m)\\ H\xi(t-\tau(t))\\ H\xi(t-\tau_M)\end{bmatrix}^T\begin{bmatrix}-Q_4 & Q_4 & 0\\ \star & -2Q_4 & Q_4\\ \star & \star & -Q_4\end{bmatrix}\begin{bmatrix}H\xi(t-\tau_m)\\ H\xi(t-\tau(t))\\ H\xi(t-\tau_M)\end{bmatrix},\\
-d_m\int_{t-d_m}^{t}\dot{\xi}^T(s)H^TR_3H\dot{\xi}(s)ds \leq\\
\begin{bmatrix}\xi(t)\\ H\xi(t-d_m)\end{bmatrix}^T\begin{bmatrix}-H^TR_3H & H^TR_3\\ \star & -R_3\end{bmatrix}\begin{bmatrix}\xi(t)\\ H\xi(t-d_m)\end{bmatrix},\\
-(d_M-d_m)\int_{t-d_M}^{t-d_m}\dot{\xi}^T(s)H^TR_4H\dot{\xi}(s)ds \leq\\
\begin{bmatrix}H\xi(t-d_m)\\ H\xi(t-d(t))\\ H\xi(t-d_M)\end{bmatrix}^T\begin{bmatrix}-R_4 & R_4 & 0\\ \star & -2R_4 & R_4\\ \star & \star & -R_4\end{bmatrix}\begin{bmatrix}H\xi(t-d_m)\\ H\xi(t-d(t))\\ H\xi(t-d_M)\end{bmatrix}.
\end{gathered}$$

Then we have

$$\begin{aligned}
&\dot{V}(t,\xi(t)) + e^T(t)e(t) - \gamma^2 w^T(t)w(t) \\
&\le \eta^T(t)(\Xi - \tilde{\Psi}_1\Psi_{33}\tilde{\Psi}_1^T - \tilde{\Psi}_2\Psi_{44}\tilde{\Psi}_2^T)\eta(t)
\end{aligned} \tag{5.37}$$

where

$$\Xi = \begin{bmatrix} \Psi_{11} & \Psi_{12} \\ \star & \Psi_{22} \end{bmatrix}, \tilde{\Psi}_1 = \begin{bmatrix} \Psi_{13} \\ \Psi_{23} \end{bmatrix}, \tilde{\Psi}_2 = \begin{bmatrix} \Psi_{14} \\ \Psi_{24} \\ \Psi_{34} \end{bmatrix},$$

with

$\eta(t) = \mathrm{col}\{\xi(t), H\xi(t-\tau_m), H\xi(t-\tau(t)), H\xi(t-\tau_M), H\xi(t-d_m), H\xi(t-d(t)), H\xi(t-d_M), \varphi(t), w(t), \xi(t-\rho(t))\}$.

From (5.31) and Schur complement, we obtain

$$\Xi - \tilde{\Psi}_1\Psi_{33}\tilde{\Psi}_1^T - \tilde{\Psi}_2\Psi_{44}\tilde{\Psi}_2^T < 0.$$

Therefore, from the above discussion, for $t \in [t_{i,k}h + \tau(t_{i,k}),\ t_{i,k+1}h + \tau(t_{i,k+1})$, if $w(t) \neq 0$, we have

$$\dot{V}(t,\xi(t)) \le -e^T(t)e(t) + \gamma^2 w^T(t)w(t) \tag{5.38}$$

Under zero initial condition, integrating both sides of (5.38) from t and letting $t \to \infty$, we have $\|e(t)\|_2 \le \gamma\|w(t)\|_2$, which means that the filtering error system satisfies an H_∞ performance γ.

When $w(t) = 0$, (5.37) reflects that there exists a $\rho > 0$, such that

$$\dot{V}(t,\xi(t)) < -\rho\xi^(t)\xi(t) \tag{5.39}$$

for $\xi(t) \neq 0$. Therefore, it can be concluded that the filtering error system is asymptotically stable if the matrix inequalities in (5.31) are true. This completes the proof. $\square$

5.3.2 Co-design of distributed H_∞ filters, congestion controllers and the event-triggering scheme

In this section, we provide a method to codesign the distributed H_∞ filter parameters, the congestion controllers and the event-triggering scheme parameters based on Theorem 5.1.

Theorem 5.2. *For given parameters $\lambda \in (0,\ 1)$ and $\gamma > 0$, the distributed H_∞ filtering problem for the resulting filter error system (5.29) is solvable if there exist $\epsilon > 0$, matrices $P_1 > W > 0$, $R_i > 0$, $Q_i > 0$ $(i = 1, 2, 3, 4)$, $S > 0$, $\Omega > 0$, Y_1, Y_2, $\hat{A}_f$, $\hat{B}_f$, $\hat{C}_f$ with appropriate dimensions, such that*

$$\begin{bmatrix} \tilde{\Psi}_{11} & \tilde{\Psi}_{12} & \tilde{\Psi}_{13} & \tilde{\Psi}_{14} \\ \star & \tilde{\Psi}_{22} & \tilde{\Psi}_{23} & \tilde{\Psi}_{24} \\ \star & \star & \tilde{\Psi}_{33} & \tilde{\Psi}_{34} \\ \star & \star & \star & \tilde{\Psi}_{44} \end{bmatrix} < 0 \tag{5.40}$$

$$\begin{bmatrix} P_1 & EW \\ \star & W \end{bmatrix} > 0 \tag{5.41}$$

where

$$\tilde{\Psi}_{11} = \begin{bmatrix} \tilde{\Lambda}_{11} & \bar{E} & Q_3 & P_1\bar{A}_d & 0 \\ \star & \hat{A}_f + \hat{A}_f^T & 0 & WE^T\bar{A}_d & 0 \\ \star & \star & \bar{Q} & Q_4 & 0 \\ \star & \star & \star & -2Q_4 & Q_4 \\ \star & \star & \star & \star & -Q_2 - Q_4 \end{bmatrix},$$

$$\tilde{\Psi}_{12} = \begin{bmatrix} R_3 & E\hat{B}_f\bar{\mathcal{A}} & 0 & P_1\bar{D} + E\hat{B}_f\bar{I} & PB_{31} \\ 0 & \hat{B}_f\bar{\mathcal{A}} & 0 & WE^T\bar{D} + \hat{B}_f\bar{I} & WE^TB_{31} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\tilde{\Psi}_{13} = \begin{bmatrix} \frac{\pi^2}{4}S & \tau_m A_{11}^T Q_3 & \bar{\tau} A_{11}^T Q_4 & d_m A_{11}^T R_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \tau_m \bar{A}_d^T Q_3 & \bar{\tau} \bar{A}_d^T Q_4 & d_m \bar{A}_d^T R_3 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\tilde{\Psi}_{14} = \begin{bmatrix} \bar{d}A_{11}^T R_4 & \bar{d}A_{11}^T S & \tilde{E}_p^T & P_1 & 0 \\ 0 & 0 & -\hat{C}_f^T & WE^T & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \bar{d}\bar{A}_d^T R_4 & \bar{d}\bar{A}_d^T S & 0 & 0 & \bar{\bar{B}}_1^T Y_1^T \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\tilde{\Psi}_{22} = \begin{bmatrix} \bar{R}_1 & R_4 & 0 & 0 \\ \star & \bar{R}_2 & R_4 & 0 \\ \star & \star & -R_2 - R_4 & 0 \\ \star & \star & \star & -H_1^T \Omega H_1 \end{bmatrix},$$

$$\tilde{\Psi}_{23} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \tau_m \bar{D}^T Q_3 & \bar{\tau} \bar{D}^T Q_4 \end{bmatrix},$$

$$\tilde{\Psi}_{24} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ d_m \bar{D}^T R_3 & \bar{d}\bar{D}^T R_4 & \bar{d}\bar{D}^T S & 0 & 0 & \bar{\bar{B}}_{21}^T Y_2^T \end{bmatrix},$$

$$\tilde{\Psi}_{33} = \begin{bmatrix} -\gamma^2 I & 0 & \tau_m \bar{B}_{31}^T Q_3 & \bar{\tau} \bar{B}_{31}^T Q_4 \\ \star & -\frac{\pi^2}{4} S & 0 & 0 \\ \star & \star & -Q_3 & 0 \\ \star & \star & \star & -Q_4 \end{bmatrix},$$

$$\tilde{\Psi}_{34} = \begin{bmatrix} d_m \bar{B}_{31}^T R_3 & \bar{d}\bar{B}_{31}^T R_4 & \bar{d}\bar{B}_{31}^T S & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tau_m Q_3 & 0 \\ 0 & 0 & 0 & 0 & \bar{\tau} Q_4 & 0 \end{bmatrix},$$

$$\tilde{\Psi}_{44} = \begin{bmatrix} -R_3 & 0 & 0 & 0 & d_m R_3 & 0 \\ 0 & -R_4 & 0 & 0 & \bar{d} R_4 & 0 \\ 0 & 0 & -S & 0 & \bar{d} S & 0 \\ 0 & 0 & 0 & -I & 0 & 0 \\ 0 & 0 & 0 & 0 & -\epsilon^{-1} I & 0 \\ 0 & 0 & 0 & 0 & 0 & -\epsilon P_1 \end{bmatrix},$$

$$\bar{Q} = -Q_1 + Q_2 - Q_3 - Q_4, \bar{E} = E\hat{A}_f + A_{11}^T E W^T,$$

$$\tilde{\Lambda}_{11} = P_1 A_{11} + A_{11}^T P_1 + Q_1 - Q_3 + R_1 - R_3 - \frac{\pi^2}{4} S,$$

with

$$\hat{A}_f = P_2 \bar{A}_f P_3^{-T} P_2^T, \hat{B}_f = P_2 \bar{B}_f, \hat{C}_f = P_2 P_3^{-1} \bar{C}_f,$$

$$W = P_2 P_3^{-1} P_2^T, \bar{I} = \begin{bmatrix} I & 0 \end{bmatrix}, E = \begin{bmatrix} 0 & I \end{bmatrix},$$

$$\begin{aligned}
A_{11} &= \begin{bmatrix} \bar{A} & 0 \\ 0 & \tilde{A} \end{bmatrix}, \ \bar{A}_d = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{A}_d \end{bmatrix}, \ \bar{D} = \begin{bmatrix} 0 & 0 \\ \tilde{D}\tilde{F} & 0 \end{bmatrix}, \\
\bar{\mathcal{A}} &= \begin{bmatrix} (\mathcal{A} \otimes I)\check{C} & 0 \end{bmatrix}, \tilde{E}_p = \begin{bmatrix} \bar{E}_p & 0 \end{bmatrix}, \\
B_{31} &= \begin{bmatrix} \bar{B}_p & 0 \\ 0 & 0 \end{bmatrix}, \bar{\bar{B}}_1 = \begin{bmatrix} 0 & 0 \\ 0 & \hat{B}_1 \end{bmatrix}, \bar{\bar{B}}_2 = \begin{bmatrix} 0 & 0 \\ 0 & \hat{B}_2 \end{bmatrix} \\
\hat{B}_1 &= diag\{-\frac{\tau_0^2 C_0}{2N^2}, -\frac{\tau_0^2 C_0}{2N^2}\}, \\
\hat{B}_2 &= diag\{-\frac{\tau_0^2 C_0}{2^2}, \cdots, -\frac{\tau_0^2 C_0}{2N^2}\}_{2\varrho \times 2\varrho}.
\end{aligned}$$

Moreover, if the above conditions are feasible, the parameters of the distributed filters are given as

$$\bar{A}_f = \hat{A}_f W^{-1}, \ \bar{B}_f = \hat{B}_f, \ \bar{C}_f = \hat{C}_f W^{-1}.$$

The congestion controller parameters are as

$$\tilde{K}_1 = P_1^{-1} Y_1, \ \tilde{K}_2 = P_1^{-1} Y_2.$$

Proof. Introduce the following matrix variables:

$$J_1 = \begin{bmatrix} I & 0 \\ 0 & P_2 P_3^{-1} \end{bmatrix}, \ P = \begin{bmatrix} P_1 & EP_2 \\ P_2^T E^T & P_3 \end{bmatrix}.$$

By using the Schur complement, we can infer from (5.41) that $P_1 - EWE^T > 0$. We then perform congruence transformations to (5.31) by diag$\{J_1^T, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I\}$ and define new variables $W = P_2 P_3^{-1} P_2^T$, $\hat{A}_f = P_2 \bar{A}_f P_3^{-T} P_2^T$, $\hat{B}_f = P_2 \bar{B}_f$, $\hat{C}_f = P_2 P_3^{-1} \bar{C}_f$ and set $\hat{B}_1 = \text{diag}\{-\frac{\tau_0^2 C_0}{2N^2}, -\frac{\tau_0^2 C_0}{2N^2}\}$, $\hat{B}_2 = \text{diag}\{-\frac{\tau_0^2 C_0}{2N^2}, \cdots, -\frac{\tau_0^2 C_0}{2N^2}\}_{2\varrho \times 2\varrho}$, $Y_1 = P_1 \tilde{K}_1$, $Y_2 = P_2 \tilde{K}_2$. Employing Lemma 5.2 and the Schur complement, we can then derive the condition (5.40) to design the distributed H_∞ filters and congestion controllers.

In addition, according to the definitions of $\hat{A}_f$, $\hat{B}_f$ and $\hat{C}_f$, we obtain

$$\begin{bmatrix} \bar{A}_f & \bar{B}_f \\ \bar{C}_f & 0 \end{bmatrix} = \begin{bmatrix} P_2^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{A}_f & \hat{B}_f \\ \hat{C}_f & 0 \end{bmatrix} \begin{bmatrix} P_2^{-T} P_3 & 0 \\ 0 & I \end{bmatrix}$$

Then, it is easy to obtain the distributed H_∞ filter parameters, which are as

$$\bar{A}_f = \hat{A}_f W^{-1},\ \bar{B}_f = \hat{B}_f,\ \bar{C}_f = \hat{C}_f W^{-1}.$$

where $W = P_2 P_3^{-1} P_2^T$.

Therefore, we can conclude that the filter $(\bar{A}_f, \bar{B}_f, \bar{C}_f)$ defined in (5.20) guarantees the filtering error system (5.29) to be asymptotically stable with an H_∞ noise attenuation level bound γ. This completes the proof. □

As can be seen from Theorem 5.2, for a given prescribed distributed H_∞ filtering performance $\gamma > 0$, and choosing the proper threshold λ in the Information Selection Module for each event-detector, we can obtain H_∞ distributed filters codesigned with the Information Dispatching Middeleware that include the Information Selection Module and the Congestion Avoidance Module.

5.4 A simulation example

In this section, a numerical example is given to illustrate the effectiveness of the proposed scheduling framework for networked systems to investigate the distributed H_∞ filtering issue.

Consider a networked system (5.1) with the following parameters

$$\begin{aligned}
A_p &= \begin{bmatrix} -1.1595 & 0.1890 & -0.2713 \\ -0.1251 & 0 & -0.1 \\ -0.230 & 0.6740 & -0.152 \end{bmatrix}, B_p = \begin{bmatrix} 0.1350 & 0.0128 \\ 0.0128 & 0.0510 \\ 0.1021 & 0.1250 \end{bmatrix}, \\
C_p^1 &= \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}, C_p^2 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, C_p^3 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \\
C_p^4 &= \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, E_p = \begin{bmatrix} 0 & 0 & 0.1 \end{bmatrix}.
\end{aligned}$$

The scenario to be considered has four different sensors deployed in a distributed area to monitor the networked system outputs $y_i(t) (i = 1, 2, 3, 4)$ for the corresponding distributed H_∞ filters. Figure 5.6 shows the communication among these

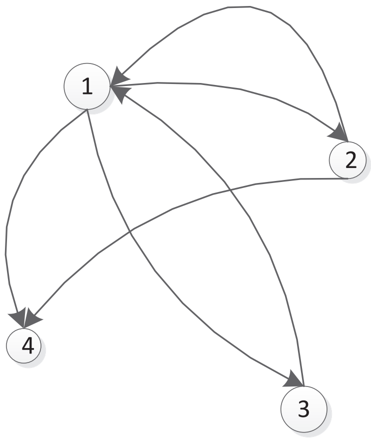

Figure 5.6: The coupling among these 4 sensors

sensors. Based on such a communication topology, the adjacency matrix $\mathcal{A}$ with coupled weights is as

$$\mathcal{A} = \begin{bmatrix} 1 & 0.01 & 0.015 & 0 \\ 0.012 & 1 & 0 & 0.08 \\ 0.08 & 0 & 1 & 0 \\ 0.026 & 0 & 0 & 1 \end{bmatrix}$$

To simplify, we consider a low speed TCP/AQM communication network with one router, and the network parameters for the fluid-flow model (5.16) are shown in Table 5.1.

Table 5.1: A low speed TCP/AQM communication network

TCP session number N	50
Link capacity C	5000 packets/second
Round trip time τ_0	0.25s
Propagation delay T_p	0.1s
Queue size q_0	750 packets
Window size W_0	25 packets
Probability of packet mark p_0	0.0032

For this environment, we choose the sampling period $h = 0.2s$, $\tau_m = 0.1s$, $\tau_M = 0.35s$ and the exogenous disturbance noise $\omega(t) = [10e^{-t^2}\sin(3\pi t); 0]$.

The threshold λ in the Information Selection Module for each event-trigger processor is set as $\lambda = 0.2$ and the prescribed distributed H_∞ performance is set as $\gamma = 0.5$, which is larger than the optimal H_∞ performance 0.2887 under the proposed

Theorem 5.2.

In the proposed Information Dispatching Middleware, the Information Selection Module has a generator with four event-trigger processors. By using the proposed codesign strategy in Theorem 5.2, we can obtain its weighting matrices as

$$\begin{aligned}
\Omega_{11}^{1} &= \begin{bmatrix} 62.6412 & -62.5527 \\ -62.5527 & 62.7606 \end{bmatrix}, \Omega_{22}^{1} = \begin{bmatrix} 2.8702 & 0.0382 \\ 0.0382 & 0.0095 \end{bmatrix}, \\
\Omega_{12}^{1} &= \begin{bmatrix} -145.5193 & -5.7456 \\ 145.4572 & 5.0615 \end{bmatrix} \times 10^{-6} \\
\Omega_{11}^{2} &= \begin{bmatrix} 3.7377 & 0.2538 \\ -1.2538 & 0.7248 \end{bmatrix}, \Omega_{22}^{2} = \begin{bmatrix} 2.8702 & 0.0382 \\ 0.0382 & 0.0095 \end{bmatrix}, \\
\Omega_{12}^{2} &= \begin{bmatrix} 427.5793 & -165.4494 \\ -521.8813 & -2.3041 \end{bmatrix} \times 10^{-8} \\
\Omega_{11}^{3} &= \begin{bmatrix} 0.1288 & 0.0242 \\ 0.0242 & 0.3136 \end{bmatrix}, \Omega_{22}^{3} = \begin{bmatrix} 2.8702 & 0.0382 \\ 0.0382 & 0.0095 \end{bmatrix}, \\
\Omega_{12}^{3} &= \begin{bmatrix} -48.0044 & -3.6159 \\ 62.9386 & -4.8465 \end{bmatrix} \times 10^{-7} \\
\Omega_{11}^{4} &= \begin{bmatrix} 118.1876 & -118.0471 \\ -118.0471 & 118.1876 \end{bmatrix}, \Omega_{22}^{4} = \begin{bmatrix} 2.8702 & 0.0382 \\ 0.0382 & 0.0095 \end{bmatrix}, \\
\Omega_{12}^{4} &= \begin{bmatrix} -31.5623 & -6.4842 \\ -31.5623 & -6.4842 \end{bmatrix} \times 10^{-7}.
\end{aligned}$$

Remark 5.4. *It can be seen from these four weighting matrices,* $\Omega_{22}^{1} = \Omega_{22}^{2} = \Omega_{22}^{3} = \Omega_{22}^{4}$*, which shows that the network dynamics have the same influence on each event-triggering processor in the Information Selection Module of the Information Dispatching Middleware.*

In this simulation, the controller gains for the Congestion Avoidance Module are

$$\tilde{K}_1 = \begin{bmatrix} -1.9936 & 0.0436 \end{bmatrix}, \tilde{K}_2 = \begin{bmatrix} -0.0008 & -0.0203 \\ -0.0487 & -0.0097 \\ 0.0102 & -0.0237 \\ -0.3001 & -0.0612 \end{bmatrix} \times 10^{-8}.$$

By using such a control strategy, the network dynamics can be shown in Figure 5.7.

Under the scheduling strategy in the Congestion Avoidance Module, the released data packet sequences from the sensor nodes to the distributed H_∞ filters are shown in Figure 5.8–5.11. It should be pointed out that by using the derived scheduling strategy in the Congestion Avoidance Module of the Information Dispatching

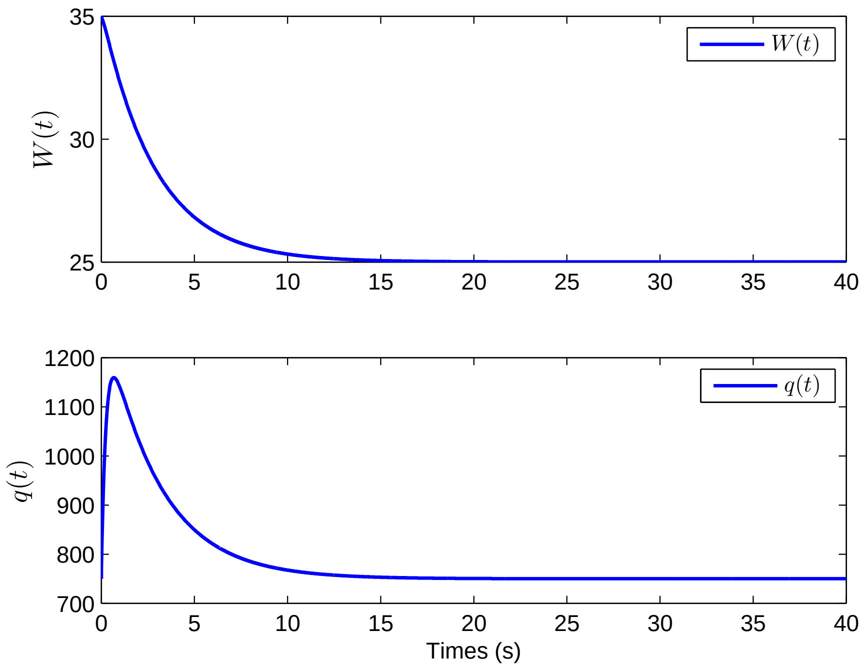

Figure 5.7: Network dynamics

Middleware, all the packets released from the Information Selection Module are transmitted within the the preset allowable time delay interval $[0.1s,\ 0.35s]$.

With the desired H_∞ performance and the proposed codesign strategy, the corresponding distributed H_∞ filters are obtained as

$$A_f^1 = \begin{bmatrix} -12.2308 & 1.2669 & 0.4129 \\ 2.3354 & -4.7254 & -1.0726 \\ -1.9755 & 1.0211 & -0.9095 \end{bmatrix}$$
$$B_f^1 = \begin{bmatrix} -0.1946 & -0.1779 \\ -0.3916 & -0.3553 \\ 0.0367 & 0.0331 \end{bmatrix}$$
$$C_f^1 = \begin{bmatrix} 0.1791 & -0.1402 & -1.1956 \end{bmatrix}$$
$$A_f^2 = \begin{bmatrix} -12.6563 & 1.9947 & -1.5902 \\ 2.7139 & -6.8801 & -4.4038 \\ -1.9891 & 1.0343 & -0.9298 \end{bmatrix}$$
$$B_f^2 = \begin{bmatrix} -0.6389 & -0.1141 \\ -0.0614 & -1.1392 \\ 0.3054 & -0.1061 \end{bmatrix}$$
$$C_f^2 = \begin{bmatrix} 0.1251 & -0.2682 & -1.0751 \end{bmatrix}$$

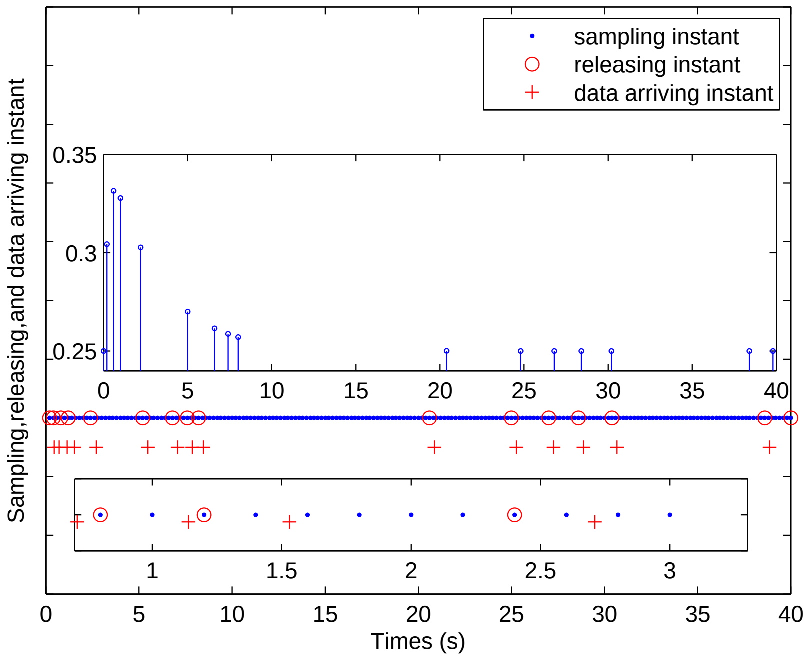

Figure 5.8: Sampling, releasing and data arriving instants for filter 1

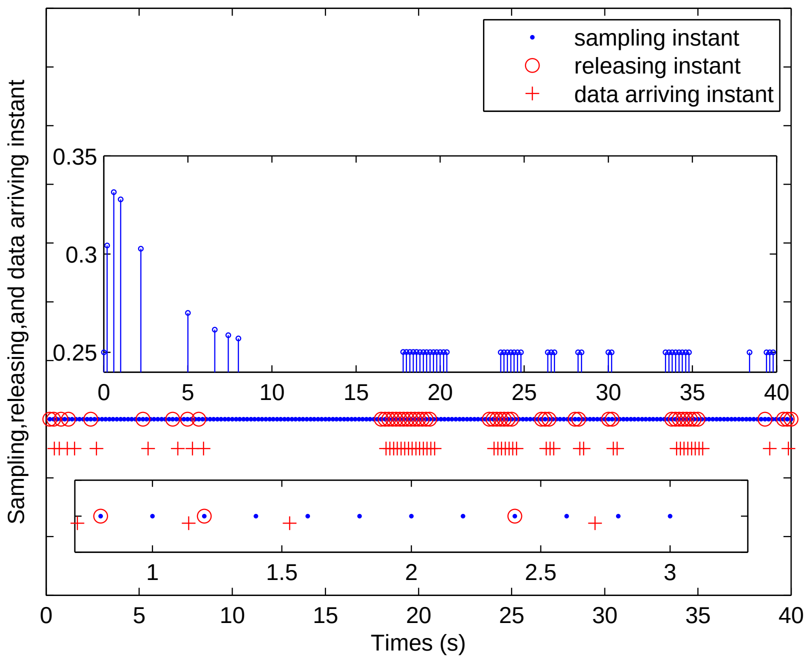

Figure 5.9: Sampling, releasing and data arriving instants for filter 2

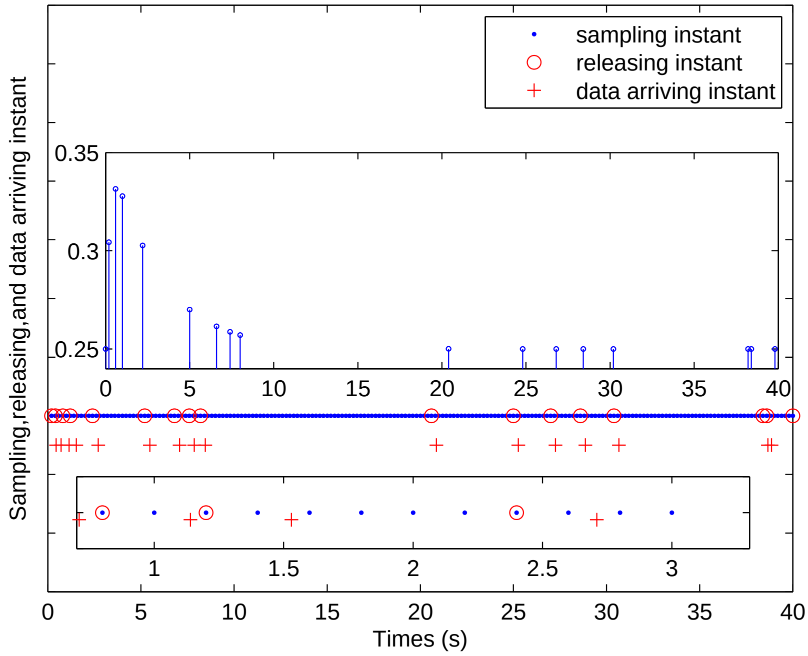

Figure 5.10: Sampling, releasing and data arriving instants for filter 3

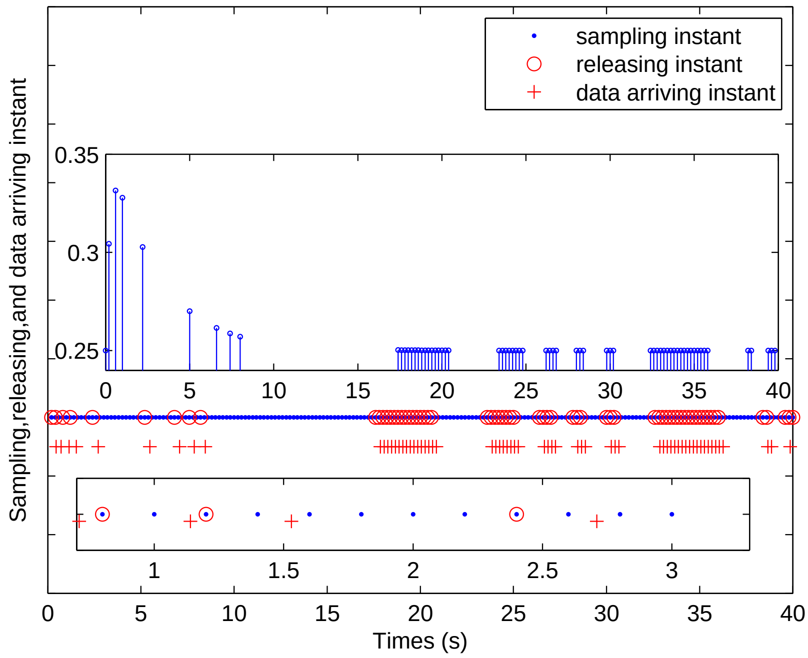

Figure 5.11: Sampling, releasing and data arriving instants for filter 4

$$A_f^3 = \begin{bmatrix} -12.2628 & 1.4983 & -2.3040 \\ 2.4629 & -5.5115 & -2.3511 \\ -1.9817 & 1.06551 & -0.8017 \end{bmatrix}$$
$$B_f^3 = \begin{bmatrix} -0.2702 & -0.2358 \\ 1.6422 & -1.2605 \\ 0.4709 & -0.0990 \end{bmatrix}$$
$$C_f^3 = \begin{bmatrix} 0.1464 & -0.1891 & -1.0868 \end{bmatrix}$$
$$A_f^4 = \begin{bmatrix} -12.3862 & 1.9581 & -1.9217 \\ 2.6880 & -6.0575 & -3.5146 \\ -1.9966 & 1.0603 & -0.8083 \end{bmatrix}$$
$$B_f^4 = \begin{bmatrix} 0.4235 & -0.4760 \\ -1.1816 & 0.0091 \\ -0.4119 & 0.2563 \end{bmatrix}$$
$$C_f^4 = \begin{bmatrix} 0.1285 & -0.2362 & -1.0505 \end{bmatrix}$$

The output signal $z_p(t)$ of the networked system and filter output signals $z_f^1(t)$, $z_f^2(t)$, $z_f^3(t)$, $z_f^4(t)$ are shown in Figure 5.12.

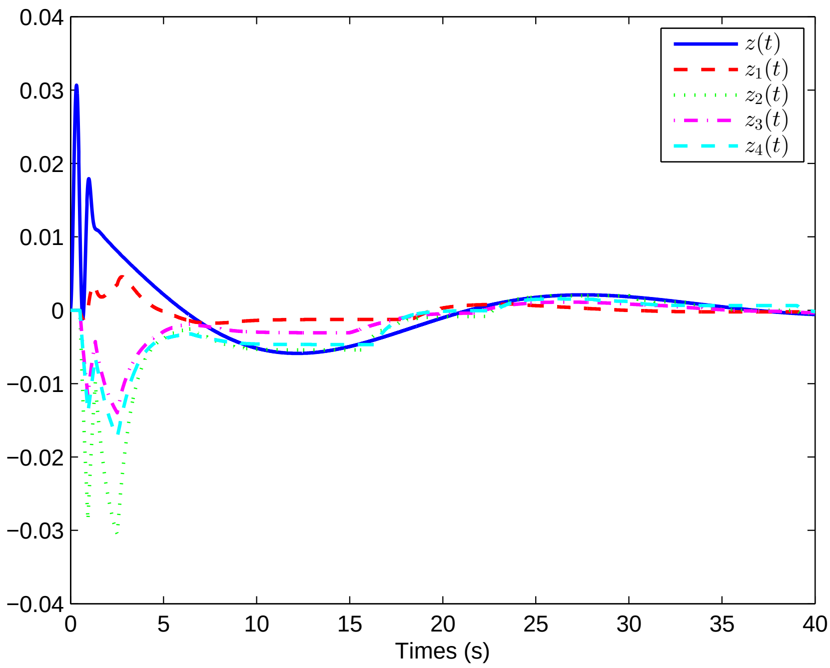

Figure 5.12: Estimation signals $z(t)$ and $z_f^1(t)$, $z_f^2(t)$, $z_f^3(t)$, $z_f^4(t)$

In the proposed Information Selection Module, whether or not the sampled signals are released depends on the mechanism of the event-triggering processors, which are designed not only from the systems dynamics, but also from the network dynamics. As discussed in the Information Selection Module, there are two cases for the sampled signals, which indicate whether the sampled signals are triggered or discarded. The rules are designed as

Case 1: $\Pi_1^i + \Pi_2^i + \Pi_3^i > 0$

- **T1**: $\Pi_1^i + \Pi_3^i \leq 0$,
- **T2**: $\Pi_1^i > 0$ & $\Pi_3^i > 0$,
- **T3**: $\Pi_1^i + \Pi_3^i > 0$ & $\Pi_1^i > 0$ & $\Pi_3^i \leq 0$,
- **T4**: $\Pi_1^i + \Pi_3^i > 0$ & $\Pi_1^i \leq 0$ & $\Pi_3^i > 0$;

Case 2: $\Pi_1^i + \Pi_2^i + \Pi_3^i \leq 0$

- **D1**: $\Pi_1^i + \Pi_3^i > 0$,
- **D2**: $\Pi_1^i \leq 0$ & $\Pi_3^i \leq 0$,
- **D3**: $\Pi_1^i + \Pi_3^i \leq 0$ & $\Pi_1^i < 0$ & $\Pi_3^i \geq 0$,
- **D4**: $\Pi_1^i + \Pi_3^i \leq 0$ & $\Pi_1^i \geq 0$ & $\Pi_3^i < 0$.

where Π_1^i, Π_2^i, Π_3^i are defined in (5.11).

The simulation results show the event-triggering processor's interaction between the dynamics of the networked system and the dynamics of the communication network. All the sampled data can be classified in Table 5.2

In this table, from the rules T3 and D2, it can be seen that the system dynamics are the major role, which determines whether or not the sampled signals should be triggered. However, under the prescribed H_∞ performance, the communication network congestion also prevents the transmission of some signals, which ought to be triggered from a system perspective, as shown in the rule D4. Both system

Table 5.2: The classification of all sampled signals in 40s

	Case 1				Case 2			
	T1	T2	T3	T4	D1	D2	D3	D4
Filter 1	0	5	9	8	0	116	0	62
Filter 2	0	5	36	4	0	103	0	52
Filter 3	0	6	79	3	0	59	0	52
Filter 4	0	5	23	4	0	103	0	55

dynamics and the signal characteristics evaluated by the function $\psi_i(t)$ can trigger some sampled signals as shown in the rule T2 while the evaluation function itself has the ability to release the sampled signals as implied by the rule T4.

Remark 5.5. *The simulation results in Table 5.2 indicate that there is no signal triggered by the rules T1 and D1, that is, the coupling between the system and the communication network does not affect the triggering mechanism of the ETPs in the Information Selection Module. This result is helpful for engineering implementation in practical applications by separation of systems and communication networks because the minimal but adequate linkage between then makes it possible to work on each independently.*

Remark 5.6. *The results also show that there is no signal triggered under rule D3 in Case 2, which implies that if the evaluation of the sampled signal is larger or the communication network is not congested, the sampled signals are triggered and released to the communication network. Hence, the rule D3 in Case 2 can be mapped to the rule T4 in Case 1.*

The interactive mechanism of the ETPs in the Information Selection Module can be clearly seen in Figure 5.13 – 5.16.

Under the proposed Theorem 5.2, the Information Dispatching Middleware and distributed H_∞ filters are obtained. One can see from the above simulation results that by using the proposed Information Dispatching Middleware, communication resources, such as the bandwidth allocated for the distributed filters, can be signifi-

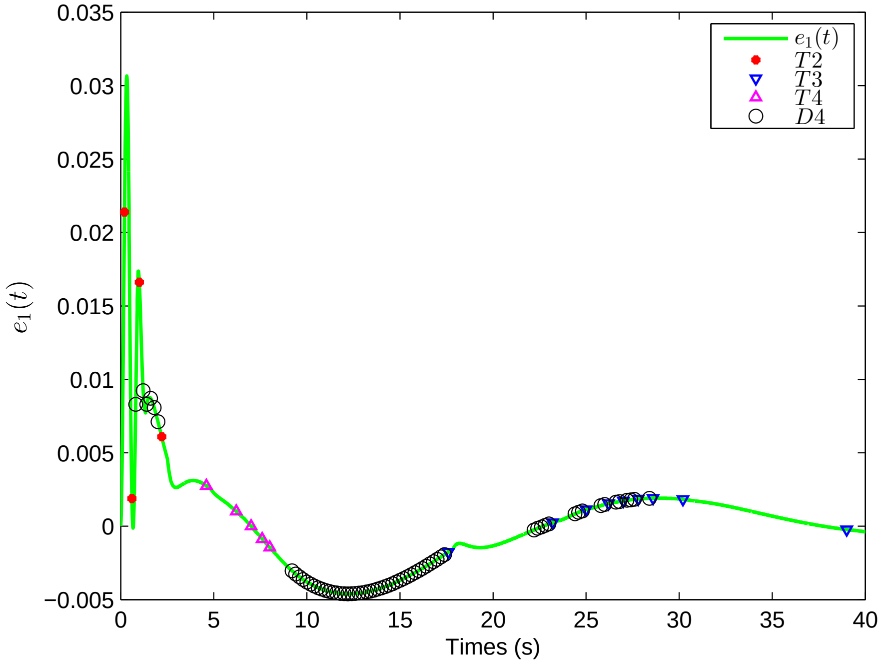

Figure 5.13: The filtering error $e_1(t)$ of Filter 1

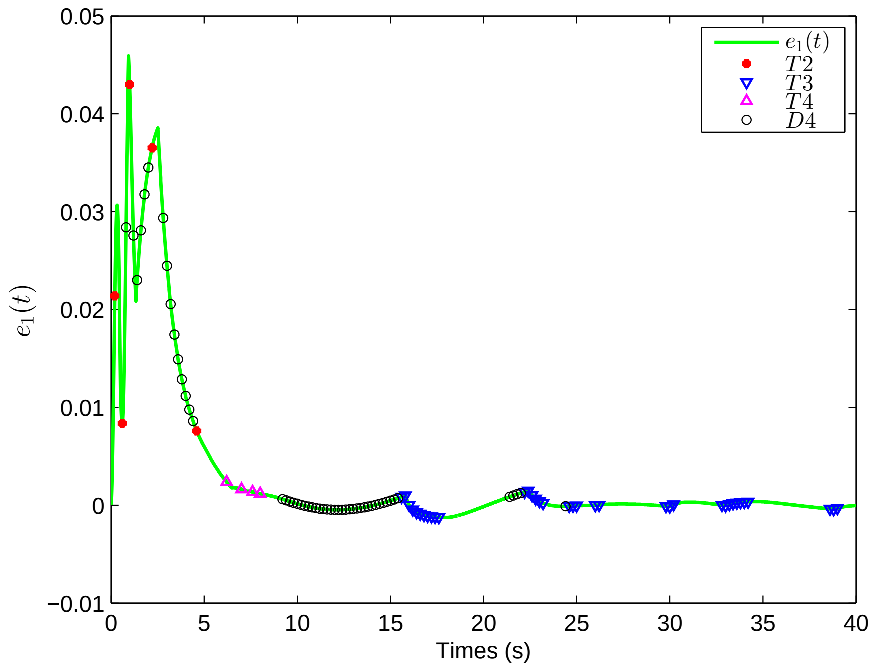

Figure 5.14: The filtering error $e_2(t)$ of Filter 2

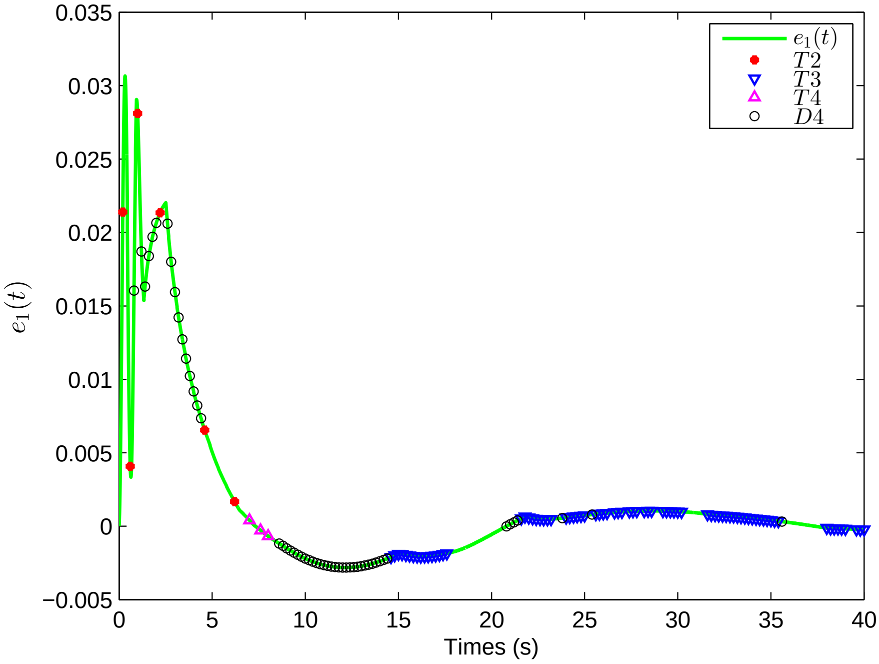

Figure 5.15: The filtering error $e_3(t)$ of Filter 3

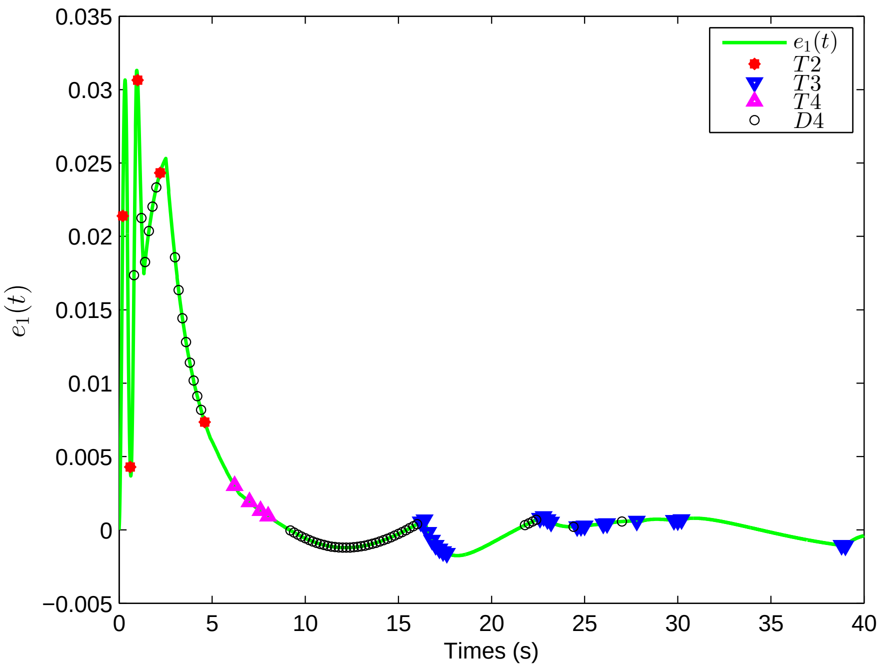

Figure 5.16: The filtering error $e_4(t)$ of Filter 4

cantly conserved. With a threshold of $\lambda = 0.2$ in the Information Selection Module, 23.38% of the filtering tasks are executed, which means that less communication resources are used and more bandwidth is made available for other transmission purposes. Figure 5.8 to Figure 5.11 show that the released packets from the Information Selection Module are scheduled in the fluid-flow model by the congestion control strategy within the limited time-delay.

5.5 Conclusion

The key factor in the investigation of distributed H_∞ filtering for networked systems has been to maintain a constant focus on the significant contribution of network dynamics to the control and filtering problems. In this chapter, An Information Dispatching Middleware has been constructed to establish a novel framework for networked systems, where two modules called the Information Selection Module and the Congestion Avoidance Module are introduced. The scheduling strategy has been proposed based on this framework. Then, the filtering error system based on network dynamics has been formulated as a system with two time-varying delays. A sufficient condition to ensure the stability and to guarantee the prescribed H_∞ noise attenuation performance for the filtering error system has been derived. Based on that condition, the codesign method of the distributed filters, the event-triggering parameter and network congestion controllers has been proposed. Finally, an example has been given to illustrate the merits and effectiveness of the method proposed in this chapter.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis has been concerned with the control and filtering of networked systems considering network dynamics. A novel middleware framework has been proposed to integrate computation and communication for real time information scheduling, such that the network resources are utilised efficiently while the networked systems' performance is preserved. Some criteria have been derived to analyse and synthesise the control and filtering issues for networked systems. Several examples are provided to illustrate the effectiveness of the proposed criteria. More specifically, the main results of this thesis are as follows:

- Stability and stabilisation of networked systems under simultaneous consideration of network dynamics have been studied within a unified framework. A innovative scheduling middleware has been proposed to trade off the performance of the control system and the utilisation of communication network resources. Based on the proposed scheduling middleware, a new type of event-triggered scheme is developed to select the necessary data to be transmitted, where both the states of the control system and the network dynamics are considered. By using such an event-triggered scheme, the stability and stabilisation criterion has been derived. Correspondingly, the controllers have been obtained to stabilise the networked system and to regulate the utilisation of network resources. A numerical example has shown the validity of the proposed method.

- H_∞ filtering for networked systems by an online scheduling strategy based on network dynamics has been investigated. An Information Dispatching Middleware has been constructed to establish a novel framework between the control system and network dynamics. This Information Dispatching Middleware consists of two modules: the Information Selection Module to regulate the transmission of the sampled data and the Congestion Avoidance Module to schedule the sampled data released by the Information Selection Module. Under this framework, an online scheduling strategy has been proposed to evaluate the network-induced time-varying delay. Then, the filtering error system with two time-varying delays has been formulated based on network dynamics. A sufficient condition has been established such that the resulting system is stable with a prescribed H_∞ noise attenuation performance. The design method for the filter and the congestion controllers has been proposed in the form of linear matrix inequalities. The effectiveness of the proposed method has been shown by a mechanical system with two masses and two springs.

- The distributed H_∞ control for large-scale distributed networked systems has been considered by taking network dynamics into account. Under the proposed Information Dispatching Middleware, the data are sampled into a group of event-triggered detectors to select the "needed" sampled data and then released to the shared IP-based communication network. The Congestion Avoidance Module is introduced to allocate the precious network resources. A codesign method has been presented to trade off the transmission rate reflected by the threshold of the event judgement function, and the QoS of the communication network. A stability criterion for the augmented system to be asymptotically stable has been established. Based on the stability condition, a codesign method for the event-triggered scheme, distributed controllers and congestion controllers has been developed to ensure that the large-scale distributed networked systems remain stable without network congestion. The effectiveness of the design method has been demonstrated through a quadruple-tank process.

- The distributed sensing and H_∞ filtering for networked systems have been studied. The interactive mechanism of the proposed Information Dispatching Middleware has been investigated, where an evaluation function has been proposed to construct the event-triggered scheme, taking both dynamic measurement outputs and communication network states into account. A discontinuous Lyapunov-Krasovskii functional has been constructed to derive a delay-dependent criterion for the existence of the distributed H_∞ filters that render the resulting filtering error system asymptotically stable with a prescribed disturbance attenuation performance index. The design method for the event-triggered filters and the congestion controllers has been correspondingly proposed in terms of linear matrix inequalities. The effectiveness of the proposed method has been shown by the simulation results.

6.2 Future Work

It should be pointed out that the issues of control and filtering for networked systems based on network dynamics have not yet been thoroughly investigated. The current research has focused on some fundamental issues, such as modelling, constructing middleware, performance analysis and controllers or filters design in the presence of network-induced delays, based on the event-triggered scheme. Some related topics for future research are listed below:

- The Information Dispatching Middleware presented in this thesis is designed specifically to enable control and filtering issues in networked systems to be studied. As such, its focus was the scheduling of sampled signals. The first natural extension of such scheduling middleware is to investigate other attractive issues, such as synchronization and consensus issues for distributed multi-agent systems. Furthermore, the network security can also be considered in the Information Selection Module of the proposed information scheduling middleware. For example, if one of the event-detectors in the Information Selection Module is attacked, the key point of research question is how to detect the security issue and to change the control or filtering strategies to ensure the rest of the subsystems meet the prescribed performance level. This area poses new challenging research issues for networked systems.

- The proposed information scheduling middleware (IDM) contains two function modules: the Information Selection Module to determine whether or not the current sampled data should be released to the communication channel and the Congestion Avoidance Module to guarantee the QoS of the communication network for the transmission of the released signals. The IDM is an open middleware, that is, more kinds of function modules can be introduced, not only to express more aspects of the communication network characteristics for the integration of computation, communication network and control systems but also to address more levels of network-induced imperfections. Therefore, in future research work, it will be meaningful to enrich the function blocks of the IDM, for example by providing a function module for quantisation of the released signals, for coding of the released signals and so on.

- It was observed that communication network congestion is a major problem, essentially leading to unanticipated phenomena, such as network accessing delays, packet losses or disorder and so on. In this thesis, for the released sampled signals from the Information Selection Module, reliable protocols in an IP-based communication network are considered to transmit triggered signals.

Based on these protocols, coupled with a congestion avoidance mechanism, a fluid-flow model is developed to abstract the dynamic characteristics of the communication network for investigating the control and filtering issues of networked systems. In future research work, other kinds of models, such as the gene regulatory network or chemical reaction processes, could be employed to model the dynamic characteristics of communication networks for the construction of new information scheduling middlewares. In those middlewares, a variety of codesign methods can be derived to trade off between the desired performance of networked systems and the efficient utilisation of the scarce communication network resources. New suitable protocols for scheduling the data flow in the communication network could also be developed for networked systems.

- In the Information Selection Module of the Information Dispatching Middleware, the proposed event-triggered scheme is advanced not only because it can avoid the *Zeno* phenomenon, but also because it is designed to adapt network dynamics, and even considers the QoS of the communication network, which is indicated by the evaluation function used in the event-triggered scheme. Future work may target how this scheme can be extended to involve more aspects, such as different levels of quantisation or coding of the sampled signals, uncertain or partially unknown statistical knowledge of networked systems and so on.

- The application of the proposed information scheduling middleware may become an attractive research topic. The middleware, which considers the event-triggered scheme, congestion control strategies, the tradeoff between the QoS of communication networks and the QoP of networked systems and so on, can be applied to develop an advanced message-oriented model (AMOM), which is one of the key technologies for building large-scale enterprise systems. AMOM may then become the glue that binds other independent and autonomous subsystems together and turns them into integrated distributed networked systems. These applications can be built by using diverse modules and can run on different platforms. Due to the unified characteristic of the Information Dispatching Middleware, users benefit from the functions provided by different modules in the proposed middleware. They are not required to rewrite their existing applications. This could be achieved by placing a queue between sender and receiver (as the control or filtering loop discussed in this thesis), to provide a level of indirection during communication to ensure the desired QoP of networked systems.

References

[1] J. Baillieul and P. Antsaklis, "Control and communication challenges in networked real-time systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 9–28, Jan. 2007.

[2] L. Zhang and D. Hristu-Varsakelis, "Communication and control co-design for networked control systems," *Automatica*, vol. 42, no. 6, pp. 953–958, 2006.

[3] D. Nesic and A. Teel, "Input-output stability properties of networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, no. 10, pp. 1650–1667, 2004.

[4] G. Walsh and H. Ye, "Scheduling of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 57–65, 2001.

[5] L. Montestruque and P. Antsaklis, "On the model-based control of networked systems," *Automatica*, vol. 39, no. 10, pp. 1837–1843, 2003.

[6] Y. Tipsuwan and M. Chow, "Control methodologies in networked control systems," *Control Engineering Practice*, vol. 11, no. 10, pp. 1099–1111, 2003.

[7] X. Wang and M. Lemmon, "Event-triggered broadcasting across distributed networked control systems," in *Proceedigns of the 2008 American Control Conference*, USA, Jun. 2008, pp. 3139–3144.

[8] J. Giraldo and N. Quijano, "Current results and research trends in networked control systems," in *Processings of the 2011 IEEE IX Latin American on*

Robotics Symposium and the IEEE Colombian Conference on Automatic Control and Industry Applications, USA, Oct. 2011, pp. 1–6.

[9] N. Vatanski, J.-P. Georges, C. Aubrun, E. Rondeau, and S.-L. Jämsä-Jounela, "Networked control with delay measurement and estimation," *Control Engineering Practice*, vol. 17, no. 2, pp. 231–244, 2009.

[10] D. Yue, Q.-L. Han, and C. Peng, "State feedback controller design of networked control systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 51, no. 11, pp. 640–644, 2004.

[11] H. Fujioka, "Stability analysis for a class of networked/embedded control systems: A discrete-time approach," in *Proceedings of the 2008 American Control Conference*, USA, Jun. 2008, pp. 4997–5002.

[12] P. Naghshtabrizi, J. Hespanha, and A. Teel, "Exponential stability of impulsive systems with application to uncertain sampled-data systems," *Systems & Control Letters*, vol. 57, no. 5, pp. 378–385, 2008.

[13] A. Balluchi, P. Murrieri, and A. Sangiovanni-Vincentelli, "Controller synthesis on non-uniform and uncertain discrete–time domains," in *Hybrid Systems: Computation and Control*. Springer, 2005, pp. 118–133.

[14] M. Cloosterman, N. van de Wouw, M. Heemels, and H. Nijmeijer, "Robust stability of networked control systems with time-varying network-induced delays," in *Proceedings of the 45th IEEE Conference on Decision and Control*, CA, Dec. 2006, pp. 4980–4985.

[15] M. Cloosterman, N. van de Wouw, W. Heemels, and H. Nijmeijer, "Stability of networked control systems with uncertain time-varying delays," *IEEE Transactions on Automatic Control*, vol. 54, no. 7, pp. 1575–1580, 2009.

[16] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[17] S. Smith and P. Seiler, "Estimation with lossy measurements: jump estimators for jump systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 12, pp. 2163–2171, 2003.

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

[19] D. Dacic and D. Nesic, "Quadratic stabilization of linear networked control systems via simultaneous protocol and controller design," *Automatica*, vol. 43, no. 7, pp. 1145–1155, 2007.

[20] D. Hristu and K. Morgansen, "Limited communication control," *Systems & Control Letters*, vol. 37, no. 4, pp. 193–205, 1999.

[21] H. Rehbinder and M. Sanfridson, "Scheduling of a limited communication channel for optimal control," *Automatica*, vol. 40, no. 3, pp. 491–500, 2004.

[22] H. Ishii and B. Francis, "Stabilization with control networks," *Automatica*, vol. 38, no. 10, pp. 1745–1751, 2002.

[23] D. Liberzon, "On stabilization of linear systems with limited information," *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 304–307, 2003.

[24] R. Brockett and D. Liberzon, "Quantized feedback stabilization of linear systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 7, pp. 1279–1289, 2000.

[25] G. Nair and R. Evans, "Stabilizability of stochastic linear systems with finite

feedback data rates," *SIAM Journal on Control and Optimization*, vol. 43, no. 2, pp. 413–436, 2005.

[26] D. Delchamps, "Stabilizing a linear system with quantized state feedback," *IEEE Transactions on Automatic Control*, vol. 35, no. 8, pp. 916–924, 1990.

[27] W. Heemels, R. Gorter, A. van Zijl, P. Van den Bosch, S. Weiland, W. Hendrix, and M. Vonder, "Asynchronous measurement and control: a case study on motor synchronization," *Control Engineering Practice*, vol. 7, no. 12, pp. 1467–1482, 1999.

[28] W. Heemels, H. Siahaan, A. Juloski, and S. Weiland, "Control of quantized linear systems: an L_1-optimal control approach," in *Proceedings of the 2003 American Control Conference*, vol. 4, USA, Jun. 2003, pp. 3502–3507.

[29] Y. Halevi, A. Ray *et al.*, "Integrated communication and control systems: Part I-analysis," *Journal of Dynamic Systems, Measurement and Control*, vol. 110, no. 4, pp. 367–373, 1988.

[30] A. Ray and Y. Halevi, "Integrated communication and control systems: Part II-design considerations," *Journal of Dynamic Systems, Measurement and Control*, vol. 110, no. 4, pp. 374–381, 1988.

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

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

[33] E. Fridman, A. Seuret, and J. Richard, "Robust sampled-data stabilization of linear systems: an input delay approach," *Automatica*, vol. 40, no. 8, pp. 1441–1446, 2004.

[34] H. Zhang, Y. Shi, and A. Mehr, "Robust weighted H_∞ filtering for networked systems with intermittent measurements of multiple sensors," *International Journal of Adaptive Control and Signal Processing*, vol. 25, no. 4, pp. 313–330, 2011.

[35] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[36] A. Elsayed and M. Grimble, "A new approach to the H_∞ design of optimal digital linear filters," *IMA Journal of Mathematical Control and Information*, vol. 6, no. 2, pp. 233–251, 1989.

[37] B. Shen, "A stochastic sampled-data approach to distributed H_∞ filtering in sensor networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 9, pp. 2237–2246, 2011.

[38] H. Gao and T. Chen, "H_∞ estimation for uncertain systems with limited communication capacity," *IEEE Transactions on Automatic Control*, vol. 52, no. 11, pp. 2070–2084, 2007.

[39] D. Yue and Q.-L. Han, "Network-based robust filtering for uncertain linear systems," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4293–4301, 2006.

[40] X. Zhang and Q.-L. Han, "Network-based H_∞ filtering for discrete-time systems," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 956–961, 2012.

[41] B. Jiang, Z. Mao, and P. Shi, "H_∞ filter design for a class of networked control systems via T-S fuzzy-model approach," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 1, pp. 201–208, 2010.

[42] H. Dong, Z. Wang, and H. Gao, "Robust H_∞ filtering for a class of nonlinear networked systems with multiple stochastic communication delays and packet dropouts," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 1957–1966, 2010.

[43] D. Carnevale, A. Teel, and D. Nesic, "A Lyapunov proof of an improved maximum allowable transfer interval for networked control systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 892–897, 2007.

[44] H. Song, L. Yu, and W. Zhang, "Networked H_2/H_∞ filtering for linear discrete-time systems," *Information Sciences*, vol. 181, no. 3, pp. 686–696, 2011.

[45] L. Montestruque and P. Antsaklis, "Stability of model-based networked control systems with time-varying transmission times," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1562–1572, 2004.

[46] L. Zhang, Y. Shi, T. Chen, and B. Huang, "A new method for stabilization of networked control systems with random delays," *IEEE Transactions on Automatic Control*, vol. 50, no. 8, pp. 1177–1181, 2005.

[47] L. Bakule and M. d. l. Sen, "Non-fragile controllers for a class of time-delay nonlinear systems," *Kybernetika*, vol. 45, no. 1, pp. 15–32, 2009.

[48] L. Bakule and M. de la Sen, "Decentralized resilient H_∞ observer-based control for a class of uncertain interconnected networked systems," in *Proceedings of the 2010 American Control Conference*, USA, Jun. 2010, pp. 1338–1343.

[49] M. Lemmon, "Event-triggered feedback in control, estimation, and optimization," in *Networked Control Systems.* Springer, 2010, pp. 293–358.

[50] A. Anta and P. Tabuada, "To sample or not to sample: Self-triggered control for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2030–2042, 2010.

[51] R. Postoyan, P. Tabuada, D. Nesic, and A. Anta, "Event-triggered and self-triggered stabilization of distributed networked control systems," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, USA, Dec. 2011, pp. 2565–2570.

[52] X. Wang and M. Lemmon, "Event-triggering in distributed networked control systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 586–601, 2011.

[53] K. Astrom and B. Bernhardsson, "Comparison of Riemann and Lebesgue sampling for first order stochastic systems," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 2, USA, Dec. 2002, pp. 2011–2016.

[54] K. Årzén, "A simple event-based PID controller," in *Proceedings of the 14th IFAC World Congress*, vol. 18, South Africa, Aug. 1999, pp. 423–428.

[55] W. Heemels, J. Sandee, and P. Van Den Bosch, "Analysis of event-driven controllers for linear systems," *International Journal of Control*, vol. 81, no. 4, pp. 571–590, 2008.

[56] P. Otanez, J. Moyne, and D. Tilbury, "Using deadbands to reduce communication in networked control systems," in *Proceedings of the 2002 American Control Conference*, vol. 4, USA, May 2002, pp. 3015–3020.

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

[58] M. Mazo and P. Tabuada, "On event-triggered and self-triggered control over

sensor/actuator networks," in *Proceedings of the 47th IEEE Conference on Decision and Control*, Mexico, Dec. 2008, pp. 435–440.

[59] X. Wang and M. Lemmon, "Decentralized event-triggered broadcasts over networked control systems," in *Hybrid Systems: Computation and Control*. Springer, 2008, pp. 674–677.

[60] A. Cervin and T. Henningsson, "Scheduling of event-triggered controllers on a shared network," in *Proceedings of the 47th IEEE Conference on Decision and Control*, Mexico, Dec. 2008, pp. 3601–3606.

[61] P. Wan and M. Lemmon, "An event-triggered distributed primal-dual algorithm for network utility maximization," in *Proceedings of the 48th IEEE Conference on Decision and Control, jointly with the 28th Chinese Control Conference*, China, Dec. 2009, pp. 5863–5868.

[62] C. Peng, Q.-L. Han, and D. Yue, "A discrete event-triggered communication scheme for networked TS fuzzy systems," in *Proceedings of the 37th Annual Conference on IEEE Industrial Electronics Society*, Australia, Dec. 2011, pp. 2282–2287.

[63] M. Velasco, P. Martí, and E. Bini, "Control-driven tasks: Modeling and analysis," in *Real-Time Systems Symposium*, 2008, pp. 280–290.

[64] M. Mazo and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2456–2461, 2011.

[65] X. Wang, "Event-triggering in cyber-physical systems," Ph.D. dissertation, University of Notre Dame, 2009.

[66] A. D. Ames, P. Tabuada, and S. Sastry, "On the stability of Zeno Equilibria," in *Hybrid Systems: Computation and Control*. Springer, 2006, pp. 34–48.

[67] D. Yue, E. Tian, and Q.-L. Han, "A delay system method to design of event-triggered control of networked control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 475–481, 2013.

[68] X.-M. Zhang and Q.-L. Han, "Event-triggered mixed H_∞ and passive control of linear systems via dynamic output feedback," in *Proceedings of the 39th Annual Conference of the IEEE Industrial Electronics Society*, Vienna, Nov. 2013, pp. 5080–5085.

[69] C. Peng and Q.-L. Han, "A novel event-triggered transmission scheme and control co-design for sampled-data control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 10, pp. 2620–2626, 2013.

[70] W. Heemels and M. Donkers, "Model-based periodic event-triggered control for linear systems," *Automatica*, vol. 49, no. 3, pp. 698–711, 2013.

[71] S. Hu and D. Yue, "Event-based H_∞ filtering for networked system with communication delay," *Signal Processing*, pp. 2029–2039, 2012.

[72] Y. Suh, V. Nguyen, and Y. Ro, "Modified Kalman filter for networked monitoring systems employing a send-on-delta method," *Automatica*, vol. 43, no. 2, pp. 332–338, 2007.

[73] V. Nguyen and Y. Suh, "Improving estimation performance in networked control systems applying the Send-on-delta transmission method," *Sensors*, vol. 7, no. 10, pp. 2128–2138, 2007.

[74] J. Sijs and M. Lazar, "On event based state estimation," in *Hybrid Systems: Computation and Control*. Springer, 2009, pp. 336–350.

[75] V. Nguyen and Y. Suh, "Networked estimation with an area-triggered transmission method," *Sensors*, vol. 8, no. 2, pp. 897–909, 2008.

[76] S. Li, D. Sauter, and B. Xu, “Fault isolation filter for networked control system with event-triggered sampling scheme,” *Sensors*, vol. 11, no. 1, pp. 557–572, 2011.

[77] V. Nguyen and Y. Suh, “Networked estimation for event-based sampling systems with packet dropouts,” *Sensors*, vol. 9, no. 4, pp. 3078–3089, 2009.

[78] X. Wang and M. Lemmon, “Self-triggered feedback control systems with finite-gain L_2 stability,” *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 452–467, 2009.

[79] C. Hollot, V. Misra, D. Towsley, and W. Gong, “Analysis and design of controllers for AQM routers supporting TCP flows,” *IEEE Transactions on Automatic Control*, vol. 47, no. 6, pp. 945–959, 2002.

[80] H. Zhang, M. Liu, V. Vukadinović, and L. Trajković, “Modeling TCP/RED: a dynamical approach,” in *Complex Dynamics in Communication Networks.* Springer, 2005, pp. 251–278.

[81] S. Mascolo, “Congestion control in high-speed communication networks using the smith principle,” *Automatica*, vol. 35, no. 12, pp. 1921–1935, 1999.

[82] S. Martínez, J. Cortés, and F. Bullo, “Motion planning and control problems for underactuated robots,” in *Control Problems in Robotics.* Springer, 2003, pp. 59–74.

[83] K. Jacobsson, H. Hjalmarsson, and K. Johansson, “Towards accurate congestion control models: Validation and stability analysis,” in *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, Japan, Jul. 2006, pp. 672–682.

[84] S. Deb and R. Srikant, “Rate-based versus queue-based models of congestion

control," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 32, no. 1, 2004, pp. 246–257.

[85] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, 1998.

[86] A. Tang, K. Jacobsson, L. Andrew, and S. Low, "An accurate link model and its application to stability analysis of FAST TCP," in *Proceedings of the 26th IEEE International Conference on Computer Communications, joint the 2007 INFOCOM*, USA, May 2007, pp. 161–169.

[87] T. Alpcan and T. Basar, "A globally stable adaptive congestion control scheme for internet-style networks with delay," *IEEE/ACM Transactions on Networking*, vol. 13, no. 6, pp. 1261–1274, 2005.

[88] X. Fan, M. Arcak, and J. T Wen, "Robustness of network flow control against disturbances and time-delay," *Systems & control letters*, vol. 53, no. 1, pp. 13–29, 2004.

[89] X. Fan, S. Kalyanaraman, M. Arcak, and Y. Chan, "Delay robustness and AQM in networks," in *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, Japan, Jul. 2006, pp. 695–700.

[90] P. Quet and H. Ozbay, "On the design of AQM supporting TCP flows using robust control theory," *IEEE Transactions on Automatic Control*, vol. 49, no. 6, pp. 1031–1036, 2004.

[91] P. Yan, Y. Gao, and H. Ozbay, "A variable structure control approach to active queue management for TCP with ECN," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 2, pp. 203–215, 2005.

[92] H. Han, S. Shakkottai, C. Hollot, R. Srikant, and D. Towsley, "Multi-path TCP: a joint congestion control and routing scheme to exploit path diversity in the internet," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1260–1271, 2006.

[93] L. Ying, G. Dullerud, and R. Srikant, "Global stability of internet congestion controllers with heterogeneous delays," *IEEE/ACM Transactions on Networking*, vol. 14, no. 3, pp. 579–591, 2006.

[94] A. Segall, "Optimal distributed routing for virtual line-switched data networks," *IEEE Transactions on Communications*, vol. 27, no. 1, pp. 201–209, 1979.

[95] Y. Liu and C. Douligeris, "Rate regulation with feedback controller in AT-M networks-a neural network approach," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 2, pp. 200–208, 1997.

[96] A. Pitsillides and A. Sekercioglu, "Fuzzy logic based effective congestion control," in *TCD workshop on Applications of Computational Intelligence to T-elecommunications*, vol. 16, London, 1999.

[97] A. Pitsillides, P. Ioannou, M. Lestas, and L. Rossides, "Adaptive nonlinear congestion controller for a differentiated-services framework," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 94–107, 2005.

[98] A. Pitsillides, P. Ioannou, and L. Rossides, "Congestion control for differentiated-services using non-linear control theory," in *Proceedings of the Sixth IEEE Symposium on Computers and Communications*, Tunisia, Jul. 2001, pp. 726–733.

[99] S. Floyd and V. Jacobson, "Random early detection gateways for congestion

avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, 1993.

[100] V. Jacobson, "Congestion avoidance and control," in *ACM SIGCOMM Computer Communication Review*, vol. 18, no. 4, 1988, pp. 314–329.

[101] P. Ranjan, E. H. Abed, and R. J. La, "Nonlinear instabilities in TCP-RED," *IEEE/ACM Transactions on Networking*, vol. 12, no. 6, pp. 1079–1092, 2004.

[102] V. Misra, W. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," *ACM SIGCOMM Computer Communication Review*, vol. 30, no. 4, pp. 151–160, 2000.

[103] C. Hollot, V. Misra, D. Towsley, and W. Gong, "A control theoretic analysis of RED," in *Proceedings of the 2001 INFOCOM, joint the Twentieth Annual IEEE Computer and Communications Societies*, vol. 3, Apr. 2001, pp. 1510–1519.

[104] P. Martí, J. Yépez, M. Velasco, R. Villà, and J. Fuertes, "Managing quality-of-control in network-based control systems by controller and message scheduling co-design," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 6, pp. 1159–1167, 2004.

[105] Y.-Q. Song, "Networked control systems: From independent designs of the network QoS and the control to the co-design," in *Proceedings of the 14th IFAC International Conference on Fieldbuses and networks in industrial and embedded systems*, Republic of Korea, May 2009.

[106] U. Tiberi, C. Fischione, K. H. Johansson, and M. D. Di Benedetto, "Energy-efficient sampling of networked control systems over IEEE 802.15. 4 wireless networks," *Automatica*, vol. 49, no. 3, pp. 712–724, 2013.

[107] S. Graham, G. Baliga, and R. Kumar, "Issues in the convergence of control with communication and computing: Proliferation, architecture, design, services, and middleware," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, vol. 2, Bahamas, Dec. 2004, pp. 1466–1471.

[108] G. Baliga and P. Kumar, "A middleware for control over networks," in *Proceedings of the 44th IEEE Conference on Decision and Control, joint the 2005 European Control Conference*, Spain, Dec. 2005, pp. 482–487.

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

[110] D. Yue, E. Tian, and Q.-L. Han, "A delay system method for designing event-triggered controllers of networked control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 475–481, 2013.

[111] X. Wang and M. Lemmon, "Event-triggering in distributed networked systems with data dropouts and delays," in *Hybrid Systems: Computation and Control*. Springer, 2009, pp. 366–380.

[112] D. Cavendish, M. Gerla, and S. Mascolo, "A control theoretical approach to congestion control in packet networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 5, pp. 893–906, 2004.

[113] F. Zheng and J. Nelson, "An H_∞ approach to the controller design of AQM routers supporting TCP flows," *Automatica*, vol. 45, no. 3, pp. 757–763, 2009.

[114] J. Yépez, P. Martí, and J. Fuertes, "Control loop scheduling paradigm in distributed control systems," in *Proceedings of the 29th Annual Conference on IEEE Industrial Electronics Society*, vol. 2, USA, Nov. 2003, pp. 1441–1446.

[115] Q.-L. Han, "Absolute stability of time-delay systems with sector-bounded nonlinearity," *Automatica*, vol. 41, no. 12, pp. 2171–2176, 2005.

[116] C. Peng, D. Yue, E. Tian, and Z. Gu, "A delay distribution based stability analysis and synthesis approach for networked control systems," *Journal of the Franklin Institute*, vol. 346, no. 4, pp. 349–365, 2009.

[117] X. Zhang and Q.-L. Han, "Robust H_∞ filtering for a class of uncertain linear systems with time-varying delay," *Automatica*, vol. 44, no. 1, pp. 157–166, 2008.

[118] H. Emara-Shabaik, M. Mahmoud, and Y. Shi, "H_2 and H_∞ filtering for linear systems with unknown inputs and polytopic uncertainty," *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 5, pp. 2211–2220, 2010.

[119] B. Shen, Z. Wang, H. Shu, and G. Wei, "H_∞ filtering for nonlinear discrete-time stochastic systems with randomly varying sensor delays," *Automatica*, vol. 45, no. 4, pp. 1032–1037, 2009.

[120] M. Mahmoud, "Delay-dependent H_∞ filtering of a class of switched discrete-time state delay systems," *Signal Processing*, vol. 88, no. 11, pp. 2709–2719, 2008.

[121] J. Wu, X. Chen, and H. Gao, "H_∞ filtering with stochastic sampling," *Signal Processing*, vol. 90, no. 4, pp. 1131–1145, 2010.

[122] D. Yue and Q.-L. Han, "Network-based robust H_∞ filtering for uncertain linear systems," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4293–4301, 2006.

[123] S. Li, D. Sauter, and B. Xu, "Fault isolation filter for networked control system

with event-triggered sampling scheme," *Sensors*, vol. 11, no. 1, pp. 557–572, 2011.

[124] V. H. Nguyen and Y. S. Suh, "Networked estimation with an area-triggered transmission method," *Sensors*, vol. 8, no. 2, pp. 897–909, 2008.

[125] X.-M. Zhang and Q.-L. Han, "On designing event-based H_∞ filters for sampled-data systems," in *Proceedings of the 51st Annual Conference on the Decision and Control*, Hawaii, Dec. 2012, pp. 6048–6053.

[126] J. Sijs and M. Lazar, "On event based state estimation," in *Hybrid Systems: Computation and Control*. Springer, 2009, pp. 336–350.

[127] Z. Wei, X.-H. Ma, and J.-Y. Xie, "A novel control scheme based on online delay evaluation for a class of networked control systems," in *Proceedings of the 2003 American Control Conference*, vol. 5, USA, Jun. 2003, pp. 4143–4148.

[128] K. J. Åström and B. Wittenmark, *Computer-controlled systems: theory and design*. Courier Dover Publications, 2011.

[129] Y.-L. Wang and Q.-L. Han, "Observer-based continuous-time networked control system design," in *Proceedings of the 2012 American Control Conference*, Canada, Jun. 2012, pp. 5694–5699.

[130] L. Xie and C. E. de Souza, "Robust H_∞ control for linear systems with norm-bounded time-varying uncertainty," *IEEE Transactions on Automatic Control*, vol. 37, no. 8, pp. 1188–1191, 1992.

[131] A. Seuret and F. Gouaisbaut, "Wirtinger-based integral inequality: application to time-delay systems," *Automatica*, vol. 49, no. 9, pp. 2860–2866, 2013.

[132] X.-M. Zhang and Q.-L. Han, "Global asymptotic stability analysis for delayed

neural networks using a matrix-based quadratic convex approach," *Neural Networks*, vol. 54, pp. 57–69, 2014.

[133] C. Langbort, R. S. Chandra, and R. D'Andrea, "Distributed control design for systems interconnected over an arbitrary graph," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1502–1519, 2004.

[134] J. K. Yook, D. M. Tilbury, and N. R. Soparkar, "Trading computation for bandwidth: Reducing communication in distributed control systems using state estimators," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 4, pp. 503–518, 2002.

[135] C. Peng, Q.-L. Han, and D. Yue, "Communication-delay-distribution-dependent decentralized control for large-scale systems with IP-based communication networks," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 820–830, 2013.

[136] T. C. Yang, "Networked control system: a brief survey," *IEE Proceedings-Control Theory and Applications*, vol. 153, no. 4, pp. 403–412, 2006.

[137] M. Vaccarini, S. Longhi, and M. R. Katebi, "Unconstrained networked decentralized model predictive control," *Journal of Process Control*, vol. 19, no. 2, pp. 328–339, 2009.

[138] X. Wang and M. D. Lemmon, "Decentralized event-triggered broadcasts over networked control systems," in *Hybrid Systems: Computation and Control*. Springer, 2008, pp. 674–677.

[139] M. Mazo and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2456–2461, 2011.

[140] D. Yue, E. Tian, and Q.-L. Han, "A delay system method for designing event-triggered controllers of networked control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 475–481, 2013.

[141] K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 3, pp. 456–465, 2000.

[142] P. Millán, L. Orihuela, C. Vivas, F. Rubio, D. V. Dimarogonas, and K. H. Johansson, "Sensor-network-based robust distributed control and estimation," *Control Engineering Practice*, vol. 21, no. 9, pp. 1238–1249, 2013.

[143] B. Shen, Z. Wang, Y. S. Hung, and G. Chesi, "Distributed filtering for polynomial nonlinear stochastic systems in sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1971–1979, 2011.

[144] J. Teng, H. Snoussi, and C. Richard, "Decentralized variational filtering for target tracking in binary sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 10, pp. 1465–1477, 2010.

[145] W. Yang, X. Wang, and H. Shi, "Optimal consensus-based distributed estimation with intermittent communication," *International Journal of Systems Science*, vol. 42, no. 9, pp. 1521–1529, 2011.

[146] M. Basin, E. Sanchez, and R. Martinez-Zuniga, "Optimal linear filtering for systems with multiple state and observation delays," *International Journal of Innovative Computing, Information and Control*, vol. 3, no. 5, pp. 1309–1320, 2007.

[147] F. Yang, Z. Wang, G. Feng, and X. Liu, "Robust filtering with randomly varying sensor delay: the finite-horizon case," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 3, pp. 664–672, 2009.

[148] M. Basin, P. Shi, and D. Calderon-Alvarez, "Approximate finite-dimensional filtering for polynomial states over polynomial observations," *International Journal of Control*, vol. 83, no. 4, pp. 724–730, 2010.

[149] Y. Hung and F. Yang, "Robust H_∞ filtering with error variance constraints for discrete time-varying systems with uncertainty," *Automatica*, vol. 39, no. 7, pp. 1185–1194, 2003.

[150] M. Basin, P. Shi, and D. Calderon-Alvarez, "Joint state filtering and parameter estimation for linear stochastic time-delay systems," *Signal Processing*, vol. 91, no. 4, pp. 782–792, 2011.

[151] L. Chai, B. Hu, and P. Jiang, "Distributed state estimation based on quantized observations in a bandwidth constrained sensor network," in *Proceedings of the 7th World Congress on Intelligent Control and Automation*, China, Jun. 2008, pp. 2411–2415.

[152] Y.-S. Ding, X.-J. Lu, K.-R. Hao, L.-F. Li, and Y.-F. Hu, "Target coverage optimisation of wireless sensor networks using a multi-objective immune co-evolutionary algorithm," *International Journal of Systems Science*, vol. 42, no. 9, pp. 1531–1541, 2011.

[153] J. Baillieul and P. Antsaklis, "Control and communication challenges in networked real-time systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 9–28, Jan. 2007.

[154] S. Zampieri, "Trends in networked control systems," in *Proceedings of the 17th IFAC World Congress*, Korea, Jul. 2008, pp. 2886–2894.

[155] D. Lehmann and J. Lunze, "Event-based control with communication delays and packet losses," *International Journal of Control*, vol. 85, no. 5, pp. 563–577, 2012.

[156] C. Aurrecoechea, A. T. Campbell, and L. Hauw, "A survey of QoS architectures," *Multimedia systems*, vol. 6, no. 3, pp. 138–151, 1998.

[157] D. Di Sorte, M. Femminella, and G. Reali, "A QoS index for IP services to effectively support usage-based charging," *IEEE communications letters*, vol. 8, no. 11, pp. 686–688, 2004.

[158] C. Peng, Q.-L. Han, and D. Yue, "To transmit or not to transmit: a discrete event-triggered communication scheme for networked Takagi-Sugeno fuzzy systems," *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 1, pp. 164–170, 2013.

[159] A. Cervin, J. Eker, B. Bernhardsson, and K.-E. Årzén, "Feedback–feedforward scheduling of control tasks," *Real-Time Systems*, vol. 23, no. 1-2, pp. 25–53, 2002.

[160] S. Manfredi, M. di Bernardo, and F. Garofalo, "Design, validation and experimental testing of a robust AQM control," *Control Engineering Practice*, vol. 17, no. 3, pp. 394–407, 2009.

[161] K. Chavan, R. G. Kumar, M. N. Belur, and A. Karandikar, "Robust active queue management for wireless networks," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 6, pp. 1630–1638, 2011.

[162] K. Liu, V. Suplin, and E. Fridman, "Stability of linear systems with general sawtooth delay," *IMA Journal of Mathematical Control and Information*, vol. 27, no. 4, pp. 419–436, 2010.