Strategic Control Methods for Wireless Networked Control Systems

March　2020

WANG　Zhaohong
王　昭鴻

Declaration

The materials contained in this dissertation are my original work and have not been previously submitted to this or any other university.

Wang Zhaohong

王　昭鴻

Abstract

Networked control system (NCS) has become one of popular research, which focuses in both academia and industry for recent few decades and has grown as a multidisciplinary area. Along with growing research trends, it is reasonable to take consideration of the latest knowledge and information to match the research needs. When a traditional feedback control system is closed by communication channels. Even more, the system may be shared with many other nodes outside the control system, then such a control system is categorized as an NCS. Note that all definitions found in literature for an NCS have a common feature in general. This key feature is that information is exchanged among elements in control system using a shared network. The information includes measured output, reference and control inputs, etc. Generally, control system elements in NCSs include sensors, controllers, and actuators.

This dissertation is organized in five chapters as follows.

Chapter I talks about a brief introduction of NCS in common. In this chapter, history of NCSs and wireless NCSs are discussed at first, followed by the main issues in the NCSs, which is also the main subject in this dissertation. An instance of wireless NCS based on Bluetooth is presented. Research towards those issues is categorized into two aspects, control of network and control over network. At the end of chapter I, we have presented some previous studies of other scholars.

In chapter II, two main issues of wireless NCSs, time-delay and packet dropout, are described. This chapter begins with problems affected by time-delay and packet dropout. The entire mathematic model is also established and studied. In this process, wireless NCS is reconstructed by merging target plant and network into a reconfigured target. Then a zero-order holder like compensator is proposed for packet dropout in control input, while packet dropout in measured output is compensated by an estimator.

An enhanced model predictive controller is proposed in chapter III since reconfigured function derived in chapter II is time-variant and multi-parametric, and such prediction model is not supported by common model predictive controller. Therefore, third-party Matlab toolbox YALMIP, a general

parser for linear matrix inequalities, and multi-parametric programming solver MPT3 (Multi-Parametric Toolbox) are introduced.

In chapter IV, several simulations and experiments are presented. The results of them are compared and discussed in criteria like stabilities, robustness, calculation cost and etc.

Chapter V includes the conclusion and possible future research directions in NCS.

List of Tables and Figures

Table of Content

Acknowledgement

Firstly, I would like to express my sincere gratitude to my advisor Prof. Fukushima for the continuous support of my Ph.D study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D study.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Hachino and Prof. Nishikawa, for their insightful comments and encouragement, but also for the hard question which incented me to widen my research from various perspectives.

Last but not the least, I would like to express my gratitude and love to my parents and my wife for supporting me throughout all my life in general, without them, I wouldn’t go this far. I owe my most sorrowful sorry to my grandfather, for absence in your funeral. Wish you rest in peace and may god bless you.

List of publications

- Zhaohong Wang, Seiji Fukushima, and Tomohiro Hachino, "Model predictive control approach on packet dropout prevention of networked control systems", Jul. 2018, Proc. 2018 5th International Conference on Information Science and Control Engineering (ICISCE)，pp.839－894.

- Zhaohong Wang and Seiji Fukushima, "Control Strategy for Networked Control systems with Time Delay and Packet Dropout using Linear Matrix Inequalities", EURASIP Journal on Wireless Communications and Networking ， doi:10.1186/s13638-019-1556-4. (Accepted in Sep. 2019)

I. Introduction

An NCS is a control system in which a data network is used as a feedback medium. The use of networks as media to interconnect the different components in an industrial control system is rapidly increasing, although the use of an NCS poses some challenges. One of the main problems to be addressed when considering an NCS is the size of the bandwidth required by each subsystem. It is clear that the reduction of bandwidth necessitated by the communication network in an NCS is a major concern. This can perhaps be addressed by two methods: the first method is to minimize the transfer of information between the sensor and the controller/actuator; the second method is to compress or reduce the size of the data transferred at each transaction. Since shared characteristics among popular industrial networks are a small transport time and a big overhead, using less bits per packet has a small impact on the overall bit rate. So, reducing the rate at which packets are transmitted brings better benefits than data compression in terms of the bit rate used.

The accelerated integration and convergence of communications, computing, and control over the last decade has inspired researchers and practitioners from a variety of disciplines to become interested in the emerging field of NCSs. In general, an NCS consists of sensors, actuators, and controllers whose operations are distributed at different geographical locations and coordinated through information exchanged over communication networks. Some typical characteristics of those systems are reflected in their asynchronous operations, diversified functions, and complicated organizational structures. The widespread applications of the Internet have been one of the major driving forces for research and development of NCS. More recently, the emergence of pervasive communication and computing has significantly intensified the effort of building such systems for control and management of various network-centric complex systems that have become more and more popular in process automation, computer integrated manufacturing, business operations, as well as public administration.

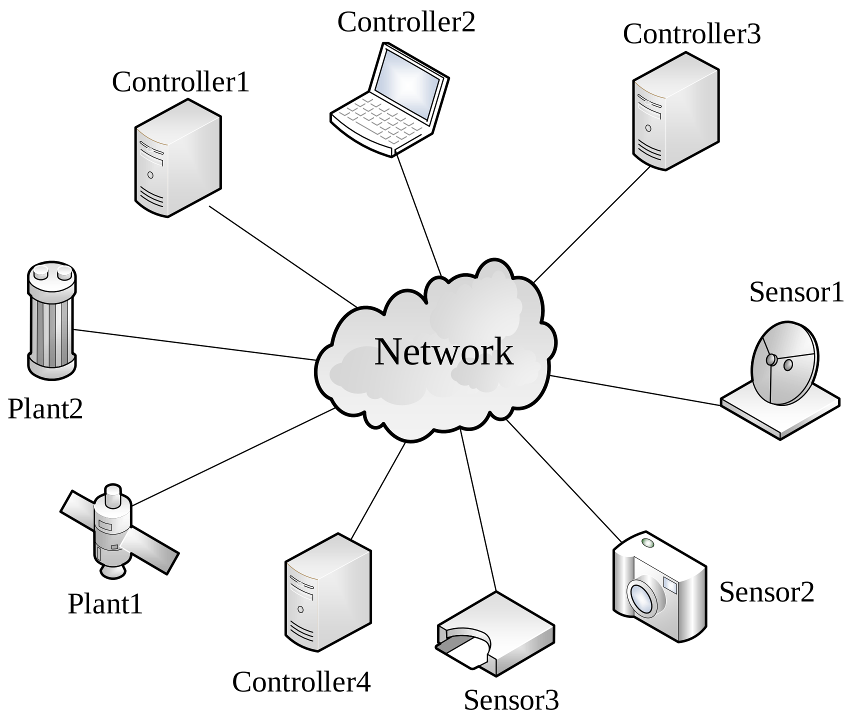


Figure 1 Typical structure of networked control system

Alongside of development of Internet of Things (IoT), we have paid more and more attention to NCSs, wherein the closed control loops are connected via networks [1]. Meanwhile, enhancement of System on Chip (SoC) and cloud computing have stimulated concerning over practical low-power networks, requiring for both security and low-power consumption, for instance, Bluetooth Low Energy (BLE), Zigbee, and etc [2]. Nowadays, NCSs are becoming more and more popular in both industry and daily life.

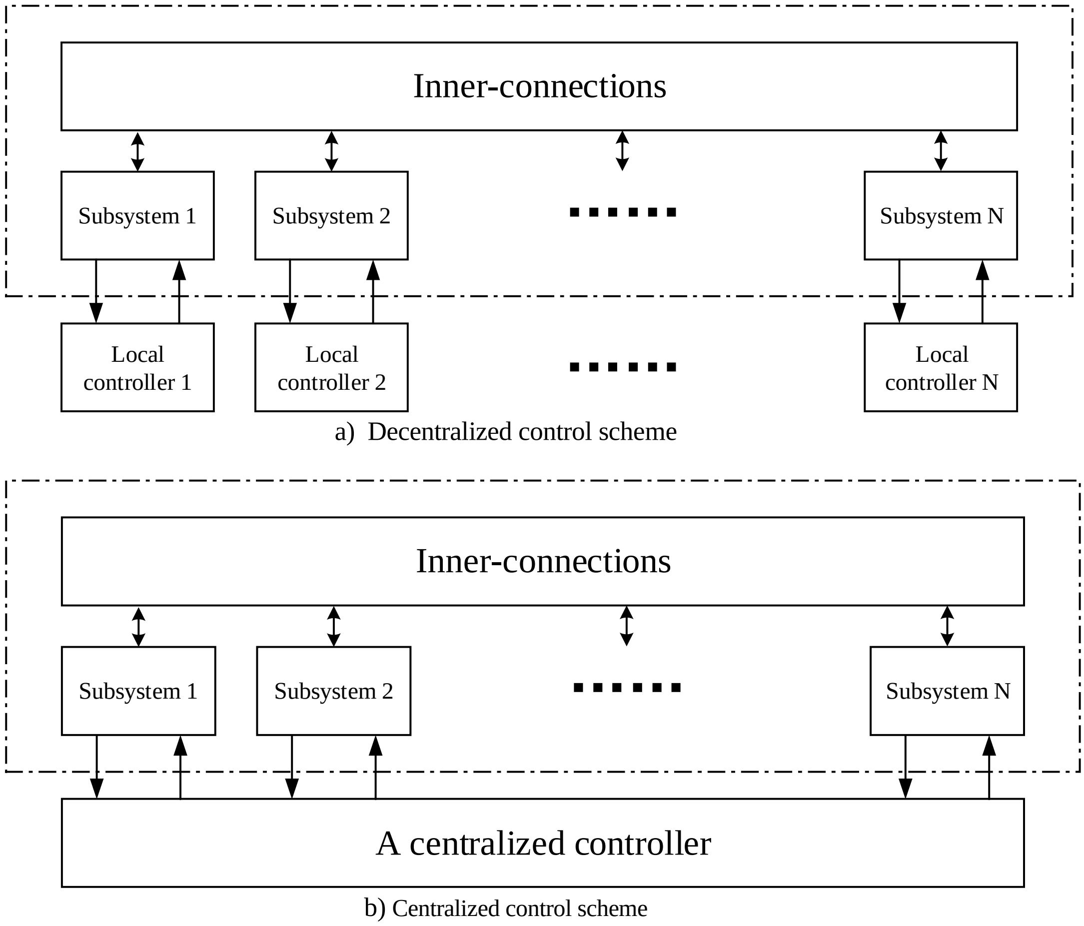


Figure 2 Decentralized and centralized control schemes

1.1 History of NCSs and wireless NCSs

The root of control systems may be traced back to 1868 when dynamics analysis of the centrifugal governor was conducted by the famous physicist J. C. Maxwell. The most significant achievement in conventional control systems happened when the Wright Brothers made their first successful test flight in 1903 [3]. The next significant progress was the fly-by-wire flight control system that was invented to eliminate the complexity; weight and fragility of the mechanical circuit of the hydro-mechanical flight control systems using an electrical circuit. The simplest and earliest configuration of an analog fly-by-wire flight control systems was first fitted to the Avro Vulcan in the 1950s. This is regarded as the first form of analog NCS. Digital computers became powerful tools in control system design and microprocessors added a new dimension to the capability of control systems. A modified NASA F-8C Crusader was the first digital fly-by-wire aircraft, in 1972. The next step in evolution was the distributed control system (DCS) that was introduced in 1975 [4]. Honeywell and Japanese electrical engineering firm Yokogawa introduced their own independently produced DCSs at around the same time, with the TDC 2000 and CENTUM system, respectively. As the expanding needs of industrial applications pushed the limit of point-to-point control, it became obvious that NCS was the solution to achieve remote control operations. Research in tele-operation was initiated with the concern for safety and convenience in hazardous environments; such as space projects and nuclear reactor power plants, and was made feasible only after further development of NCS. [5][6]

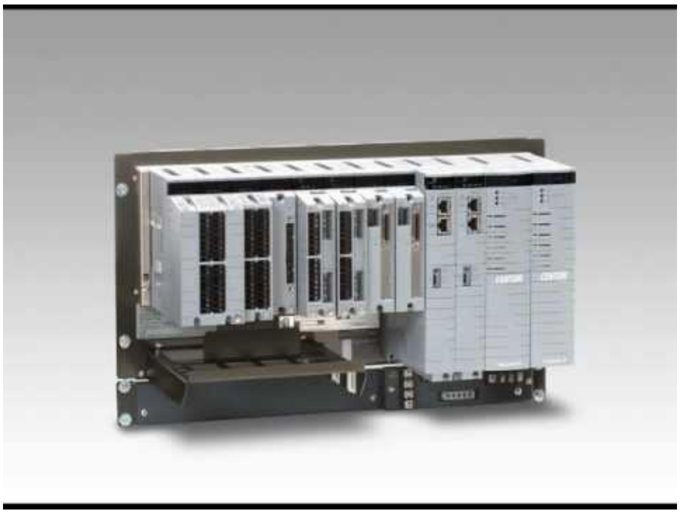


Figure 3 Industrial Distributed control systems - Yokogawa CENTUM

In the early development of control, the main feature was a centralized controller structure. Using electrical/electronic control as an example, all sensor/actuator/controller connections are point-to-point wired. There is no signal loss or time delay in signal transfer. This can be called an ideal centralized control. In the 1970s and 1980s, the application extended to distributed large-scale systems, which stimulated extensive study on decentralized control. A large-scale system "plant" is split into a number of subsystems with the interconnections between them. Under a decentralized control scheme, a number of local controllers are connected to each distributed subsystem and there is no signal transfer between different local controllers. Using a simplified term of feedback control matrix, it is a full matrix under a centralized control and a block-diagonal matrix under a decentralized control. Decentralized control was the only solution to some applications when modern communication techniques were not available. However, apart from the obvious performance limit (control matrix is limited to block-diagonal), it has already been established that, given a plant and a decentralized control structure, there may not exist any decentralized stabilizing controllers (not to mention a decentralized robust stabilizing controller). Indeed, Wang and Davison first introduced the notion of decentralized fixed modes associated with a controller structure [7]. A stabilizing controller exists if these fixed modes are stable. Among all important work in this area, Siljak developed state space and graph theoretic methods to address the problems of decentralized stability, decentralized controllability, fixed mode characterization, and decentralized controller design [8].

The accelerated integration and convergence of communications, computing, and control over the last decade has inspired researchers and practitioners from a variety of disciplines to become interested in the emerging field of NCSs. In general, an NCS consists of sensors, actuators, and controllers whose operations are distributed at different geographical locations and coordinated through information exchanged over communication networks. Some typical characteristics of those systems are reflected in their asynchronous operations, diversified functions, and complicated organizational structures. The widespread applications of the Internet have been one of the major driving forces for research and development of NCS. More recently, the emergence of pervasive communication and computing has significantly intensified the effort of building such systems for control and management of various network-centric complex systems that have become more and more popular in process automation, computer integrated manufacturing, business operations, as well as public administration.

Compared to traditional wired control systems, wireless NCSs show great advantages such as enhanced mobility, remote operation, and improved safety. Wireless tracking control systems have been widely used in industry and daily life. The controller wirelessly communicates with the actuator and sensor through wireless access points and stations. Once the controller receives sensor signals, it calculates and transmits controller signals to the actuator based on the embedded control algorithm and reference signals. However, as is well known, a wireless NCS is inherently less reliable than the traditional wired control system.

Demands on diversity, complexity, and real-time performance for networked operations have brought new technological challenges to NCS. Today, many fundamental questions regarding the stability of interconnected dynamical systems, the effects of communication on the performance of control systems, etc., remain open and to be answered. Even from the perspective of control field alone, we need to think about what the new direction for research and application in this age of connected world would be. One potential approach is to extend the concept of "code on demands" with agent programming to "control on demands" with agent-based control. In other words, can we liberate control algorithms that are fixed to plants to be controlled to control agents that are free and mobile in a connected world? Once this is accomplished, various innovative methods based on connectivity can be employed for control and management, e.g., using "local simple, remote complex" principle to design low cost yet high performance and intelligent NCS that require less computing power, small memory space, and little upgrading. Indeed, there are many new, exciting, and challenging ideas, problems, and concepts in the emerging field of NCSs.

1.2 Main Issues of NCSs

We have to beware of two main defects that NCSs may have problems such as time delay and packet dropout. Considering time delay and packet dropout in NCSs, especially in WNCSs. The reasons of them are varied; some are due to distance, and others are due to disturbances or intrinsic property. For example, Bluetooth has an intrinsic time delay even if a face-to-face connection. Fortunately, time delay and packet dropout can be ignored in most cases, and some of the rest may be compensated by a more robust connection. However, in the NCS requiring for higher real-time characteristics, time delay and packet dropout should be taken into consideration of control strategy.

In a view of the essence of NCSs, the control loops that are closed over a communication network, from a simple field bus to the nested Internet, get more and more common as the hardware devices for network and network nodes become cheaper. In recent years, much attention has been paid to the use of real-time networks in a control loop and the networks have been increasingly used as a medium to interconnect different components in large-scale plants and in geographically distributed systems. Examples are industrial automation, example, it reduces cost of cabling and offers modularity and flexibility in system design. There is currently a strong research interest in NCS among control communities [9][10][11]. Many researchers study the stability [12][13], performance and compensator design with regard to random delay caused by network communication in an NCS [14]. Research in network and control has received considerable attention during last few years. Several special issues on the topics have taken place or are taking place. A recent special issue is focusing on NCS. In this paper, different from all other existing studies, we explore the impact and potential of NCS viewed from the controller structure for large complex systems.

1.3 research trend of NCSs

A communication network is the backbone of the NCS. Reliability, security, ease of use and availability are the main issues while choosing the communication type. The ARPANET developed by ARPA (Advanced Research Projects Agency Network) of the US Department of Defense in 1969, was the world's first operational packet switching network, and the predecessor of the Internet [15]. Later came Fieldbus (around since 1988) - an industrial network system for real-time distributed control. Fieldbus is a generic-term which describes a modern industrial digital communications network intended to replace the existing 4-20mA analog signal standard. This network is a digital, bi-directional, multi-drop, serial-bus used to link isolated field devices especially in the automated manufacturing environment. Profibus (Process Field Bus) is a standard for fieldbus communication in automation technology and was first promoted in 1989 by BMBF (German department of education and research). CAN is one of the other fieldbus standards - a serial, asynchronous, multi-master communication protocol designed for applications needing high-level data integrity and data rates of up to 1 Mbit/s. CAN was introduced in 1980s by Robert Bosch GmbH for connecting ECUs for automotive applications (vehicle bus) following the fly-by-wire technology in flight control [16]. CAN-based DCS have two main restrictions. They are the size of distributed area and the need for communication (1) with other LANs; and (2) with remote CAN segments. Thus, there is a wide variety of competing fieldbus standards and therefore many times interoperability becomes an issue. Some of the proposed solutions for this are extensible device description based on XML [17] and integrated fieldbus network architecture [18]. Another communications network used in NCS - Ethernet - has evolved into the most widely implemented physical and link layer protocol today, mainly because of the low cost of the network components and their backward compatibility with the existing Ethernet infrastructure. Now we have, fast Ethernet (10 to 100 Mbit/s) and Gigabit Ethernet (1000 Mbit/s). Recently, switched Ethernet became a very promising alternative for real-time industrial application due to the elimination of uncertainties in traditional Ethernet [19].

The basis capabilities of any NCS are information acquisition (sensors / users), command (controllers / users), communication, and network and control (actuators). In broader terms, NCS research is categorized into two parts:

(1) Control of network: Study and research on communications and networks to make them suitable for real-time NCS, e.g. routing control, congestion reduction, efficient data communication, networking protocol etc.

(2) Control over network: This deals more with control strategies and control systems design over the network to minimize the effect of adverse network parameters on NCS performance such as network delay.

The previous aspect includes measures making networks suitable for real-time NCSs, e.g. congestion reduction, efficient networking protocol. Meantime, control over network pays more attention to control strategies and control systems design over the network to enhance the performance of control system against packet dropout, time delay, and etc. [20]

1.4 Previous research

Our study is under the domain of control over network. While in this area, there have been many remarkable studies discussing such issue of time delay and packet dropout in NCSs. H. Mehrivadsh and M. Shafiei have discussed a robust model predictive control for discrete-time delayed systems, when most practical NCSs are continuous-time delayed [21]. G. Torres have considered an NCS with time delayed less than sampling period, the application area of which is limited and the study has more concerning over modelling [22]. An explanation of Markov chains process in packet dropout has been introduced by J. Wu and T. Chen [23][27]. P. Seiler and R. Sengupta modelled packet dropout as a Markov jumping linear system and discussed an H-∞ approach for it. The global H-∞ gain is excellent; however, an approximation of bilinear matrix inequalities problem is not well proved [24]. A standard model predictive controller has been raised with fast response speed in 40 steps but the robustness of such control design has not been discussed [25]. D. Nesic and A. Teel have employed a perturbation theory to analyze the stability of NCSs that could be applied in the forward path from a sensor to a controller, although it is not feasible to approximate system as a continuous one in real control practice [26]. F. Yang and Q. Han discussed an H-∞ controller solved by linear matrix inequalities at a significant low cost while the entire control system is at a risk of instable since only optimal situations are under consideration, leaving alone the worst situations [28]. A modified preview control for a wireless tracking control system with packet loss is proposed by W. Zhang, J. Bae and M. Tomizuka in need of some deterministic future segment while most NCSs in reality have only stochastic future [29].

Wang presents the history and issues of agent-based control and management for NCS from the perspective of his own research group. He argues and calls for a paradigm shift from control algorithms to control agents so that agent-based control can be established as the new control mechanism for operation and management of networked devices and systems. The goal of his agent-based approach is to transform “code on demand” in programming into “control on demand” in control, and provides a platform for designing and building low cost but high-performance networked equipment in the age of connectivity [30].

Some researchers considered the fact that the design and implementation of many digital systems have been based on the emulation of idealized continuous-time blocks, and in analogy with sampled-data control system design, Tabbara, Nesic, and Teel explore an emulation-based approach to the analysis and design of NCS. For this purpose, they survey a selection of emulation-type NCS results in the literature and highlight the crucial role that scheduling between disparate components of the control systems plays. They then detail several different properties that scheduling protocols need to verify together with appropriate bounds on inter-transmission times such that various notions of input-output stability of the nominal "network-free" system is preserved when deployed as an NCS. This could be an important method for designing NCS in the future. Liu addresses issues in analysis and design of NCS based on a novel control strategy, termed networked predictive control. The stability of the closed-loop networked predictive control system is analyzed. The analytical criteria are obtained for both fixed and random communication time delays. The on-line and real-time simulation of networked predictive control systems is presented in detail [31].

Yue, Han, and Lam discuss the design of robust $H\infty$ controllers and $H\infty$ filters for uncertain NCS with the effects of both network induced delay and data dropout taken into consideration. In this chapter, a new analysis method for $H\infty$ performance of NCS is provided by introducing slack matrix variables and employing the information of the lower bound of the network-induced delay. Numerical examples and simulation results are given to illustrate the effectiveness of their proposed method. Nikolakopoulos, Panousopoulou, and Tzes propose a switched output feedback control scheme for networked systems, and apply the scheme to client–server architectures where the feedback control loop is closed over a general-purpose wireless communication channel between the plant (server) and the controller (client). To deal with network delay effects, a linear quadratic regulator (LQR)-output feedback control scheme is introduced, whose parameters are tuned according to the variation of the measured round-trip latency times. The overall scheme resembles a gain scheduler controller with the latency times playing the role of scheduling parameter. The proposed control scheme is applied in both experimental and simulation studies to an NCS over different communication channels [32].

Yang and Zhang have developed a guaranteed cost networked control (GCNC) method and established the corresponding stability for Takagi–Sugeno (T–S) fuzzy systems with time delay [33]. Both analytical studies and simulation results show the validity of their proposed control scheme. A robust $H\infty$ networked control method for T–S fuzzy systems with uncertainty and time delay is also presented in this chapter, along with sufficient conditions for robust stability with $H\infty$ performance. Sun and Wu have proposed a discrete-time jump fuzzy system for the modeling and control of a class of nonlinear NCS with random but bounded communication delays and packets dropout [34]. Here a guaranteed cost control with state feedback is developed by constructing a sub-optimal performance controller for the discrete-time jump fuzzy systems in such a way that a piecewise quadratic Lyapunov function (PQLF) can be used to establish the global stability of the resulting closed-loop fuzzy control system. When not all states are available, an output feedback controller is designed. For the NCS based on the mixed networks, a neuro-fuzzy controller is developed. Simulation examples are carried out to show the effectiveness of their proposed approaches. Chen investigates the boundary control of damped wave equations using a boundary measurement in an NCS setting [35][36]. In his approach, induced delays in this networked boundary control system are lumped as the boundary measurement delay. The Smith predictor is applied to this problem and the instability problem due to large delays is solved and the scheme is proved to be robust against a small difference between the assumed delay and the actual delay. He also analyzes the robustness of the time-fractional order wave equation with a fractional order boundary controller subject to delayed boundary measurement.

	method	target	issues	cost	speed
others	robust MPC []	LTI	both but discrete	NaN	~20 steps

			delayed time		
	Markov chain []	LTI	only packet dropout	NaN	~20 steps
	H-∞ []	LTI	only packet dropout	2.50%	NaN
	MPC []	LTI	only time delay	NaN	~30steps
proposition in this study	enhanced robust MPC	LTI	both	3.67%	~15steps

Table 1 Comparison of proposed method and some other research

In our proposition, Pade approximation is introduced in the modelling process, with the help with that, sections of time delay are eliminated. Among numerous approximation measures, Pade approximation may still work when target function's Taylor series does not converge, and it is widely used in computer calculation especially in rational functions of given order. Then a zero-order holder like compensator is proposed for packet dropout in control input, while packet dropout in measured output is compensated by estimation. Then the control target is reformed as a control problem of an uncertain time-varying multi-parametric state space. Moreover, in order to apply toolbox YALMIP and solver MPT3, such state space is rewritten as a linear parametric-varying prediction model. Finally, a robust model predictive controller is completed by solving a constrained minimax problem of predictive cost function. In order to show the advantages of our proposition, Table 1 compares some other research with proposed method in their performances.

II. Basics and problem formation

As is shown in Figure 4, the overall NCS is consisted of target plant, controller and network between them while the NCS is reconstructed by merging target plant and network into a reconfigured target plant in Figure 5. In the modelling process, sections of time delay are eliminated through Pade approximation. Among numerous approximation measures, Pade approximation may still work when target function’s Taylor series does not converge, and it is widely used in computer calculation especially in rational functions of given order [37]. Then a zero-order holder like compensator is proposed for packet dropout in control input, while packet dropout in measured output is compensated by estimation.

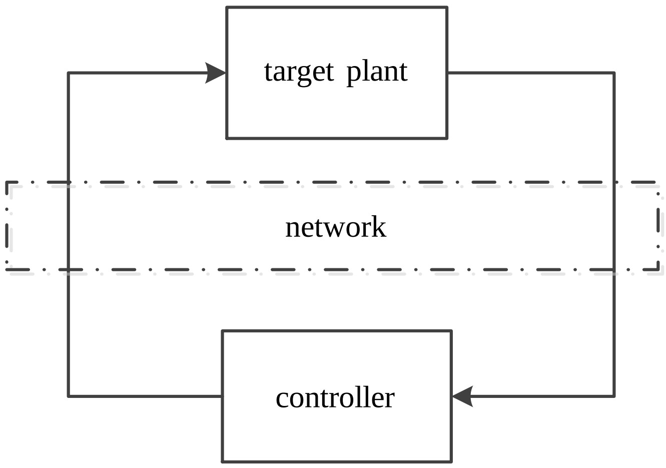


Figure 4 Brief diagram of typical networked control system

2.1 Basics

In this section, three preliminary elements in NCSs are introduced in brief, as well as main issues in NCS, such as time-delay, packet dropout, etc.

2.1.1 Sensors, actuators and networks

Sensors and actuators are two critical components of every closed loop control system. Such a system is also called a mechatronics system. A typical mechatronics system as shown in Figure 1 consists of a sensing units, controllers, and actuating units. A sensing unit can be as simple as a single sensor or can consist of additional components such as filters, amplifiers, modulators, and other signal conditioners. The controller accepts the information from the sensing unit, makes decisions based on the control algorithm, and outputs commands to the actuating unit. The actuating unit consists of an actuator and optionally a power supply and a coupling mechanism. [38]

Sensor is a device that when exposed to a physical phenomenon (temperature, displacement, force, etc.) produces a proportional output signal (electrical, mechanical, magnetic, etc.). The term transducer is often used synonymously with sensors. However, ideally, a sensor is a device that responds to a change in the physical phenomenon. On the other hand, a transducer is a device that converts one form of energy into another form of energy. Sensors are transducers when they sense one form of energy input and output in a different form of energy. For example, a thermocouple responds to a temperature change (thermal energy) and outputs a proportional change in electromotive force (electrical energy). Therefore, a thermocouple can be called a sensor and or transducer.

Normally, the output from a sensor requires post processing of the signals before they can be fed to the controller. The sensor output may have to be demodulated, amplified, filtered, linearized, range quantized, and isolated so that the signal can be accepted by a typical analog-to-digital converter of the controller. Some sensors are available with integrated signal conditioners, such as the microsensors. All the electronics are integrated into one microcircuit and can be directly interfaced with the controllers.

Linear and rotational position sensors are two of the most fundamental of all measurements used in a typical mechatronics system. In general, the position sensors produce an electrical output that is proportional to the displacement they experience. There are contact type sensors such as strain gage, LVDT (Linear Variable Differential Transformer), RVDT (Rotary Variable Differential Transformer), tachometer, etc. The noncontact type includes encoders, hall effect, capacitance, inductance, and interferometer type. They can also be classified based on the range of measurement. Usually the high-resolution type of sensors such as hall effect, fiber optic inductance, capacitance, and strain gage are suitable for only very small range (typically from 0.1 mm to 5 mm). The differential transformers on the other hand, have a much larger range with good resolution. Interferometer type sensors provide both very high resolution (in terms of microns) and large range of measurements (typically up to a meter). However, interferometer type sensors are bulky, expensive, and requires large set up time.

Actuators are basically the muscle behind a mechatronics system that accepts a control command (mostly in the form of an electrical signal) and produces a change in the physical system by generating force, motion, heat, flow, etc. Normally, the actuators are used in conjunction with the power supply and a coupling mechanism. The power unit provides either AC or DC power at the rated voltage and current. The coupling mechanism acts as the interface between the actuator and the physical system. Typical mechanisms include rack and pinion, gear drive, belt drive, lead screw and nut, piston, and linkages.

Actuators are essentially of electrical, electromechanical, electromagnetic, hydraulic, or pneumatic type. The new generations of actuators include smart material actuators, micro-actuators, and Nanoactuators. Actuators can also be classified as binary and continuous based on the number of stable-state outputs. A relay with two stable states is a good example of a binary actuator. Similarly, a stepper motor is a good example of continuous actuator. When used for a position control, the stepper motor can provide stable outputs with very small incremental motion.

2.1.2 Issues

a) Time-delay

The network access in a NCSs typically introduces delays, since nodes may have to wait until the network becomes available. Depending on the implementation or context, these delays may either affect the times between consecutive sampling, in cases where sensors and controllers respond on demand to network availability, or may introduce significant delays on the data received from plant and controller, in case the sensors hold past information until the network becomes available for transmission. Further delays may be taken into account such as transmission and processing delays.

It is well-known that transmission delays affect the dynamic performance and stability of feedback control systems. This is particularly true of the NCS. Network delays are the various delays with different lengths due to sharing a common network medium and mainly include two types of time delays: delay from controller to the corresponding actuator; delay from sensor to the corresponding controller.

Nilson classified the network delays into three cases [39]: (1) constant delays; (2) random delays with the independence; and (3) random delays with the distributions governed by a Markov chain. Case (1) is the easiest and many techniques, e.g. Smith-Estimator and its modifications, are available in conventional control theory to handle a system with a constant time delay. A state observer-based control strategy for NCS is proposed in to deal with the NCS with a constant time delay. Case (2) assumes that the current transfer delay is independent of the previous delays have different probability distributions. Case (3) uses the memory to model network queues and varying network loads. A Markov-Bernoulli process is used to model the network queue with delays [40]. The delay robustness of a simple NCS is investigated. To reduce the delay effect, a multi-rate control method of NCS is studied by using a two-level control structure with a fast controller at the lower level and a slow controller at the higher level. A predictive control approach with the random network delay for NCS is investigated [41]. A threshold control policy is proposed to improve the network QoS.

b) Data packet drop-out

Data packet drop-out is due to the unreliable data transmission paths which not only cause the transmission delay but also may generate the transmission lost. Reducing network packet drop-out rate has a significant impact on the QoS. One of the main differences between NCS and conventional control is that synchronization between different sensors, actuators and control units is not guaranteed, which cause the congestion in NCS which may lead to the package dropout to reduce the queue size. The drop-out rate is one important measure of QoS. There are many researches in reducing the drop-out rate and maintaining the system stability. For example, stability of NCS in the presence of packet losses caused by network congestion avoidance mechanisms is studied [42].

c) Sampling time

The sampling time has a significant impact to the performance and stability of a closed-loop control system. In general, the smaller the sampling time, the better the performance of the closed-loop system. However, when the sampling time decreases beyond a certain critical point, the system performance will decrease and even become unstable. This is due to the uncertainty in the system. On the other hand, the smaller sampling time will increase the network transfer load. This may cause network traffic saturation and longer delay which will degrade the control performance. Due to the above issues, selection of the best sampling time period becomes an important task and most times a compromise has to be taken. In practical applications, most researchers use the simulation and experimental to decide the best sampling time. Selection of the best sampling time of NCS is an interesting research topic and should be studied in line with the network scheduling. The multi-rate control strategy investigated may provide a sensible solution to choosing a compromising sampling time for an NCS [43].

d) Jitter

Jitter is generally defined as any distortion of a signal or image caused by poor synchronization, which is short-term variations of the significant instances of digital signals from their ideal positions in time. It is defined by the IEEE as "time-related, abrupt, spurious variations in the duration of any specified related interval", and arises due to clock drift, branching in the code, scheduling, communication, and use of certain computer hardware structures, e.g. cache memory. Jitter can be classified into two types: delay jitter and rate (sampling) jitter. The goal of delay jitter is to minimize the difference between delay times of different packets. The goal of rate jitter is to minimize the difference between inter-arrive times. Jitter has a significant effect on the performance of NCS [44].

e) Stability

Stability plays a fundamentally important role in a closed loop control system and has to be considered first. This is also true in the design of an NCS. Due to the challenging nature of NCS, e.g. delay, sampling time, jitter, stability is getting hard to maintain. There are many researches in study of stability of NCS.

Among these, it is worthwhile to pay attention to the study of stability for model-based NCSs. In some other work, an explicit model of the plant is used to produce an estimate of the plant behavior between feedback transmission times. The stability of model-based NCSs is studied when the controller/actuator is updated with the sensor feedback data at non-constant time intervals. For NCSs with transmission times that are time varying within a time interval, sufficient conditions for Lyapunov stability are derived. For systems with transmission times driven by stochastic processes with identical independently distributed or Markov-chain transmission times, sufficient conditions for almost sure stability as well as mean square stability are presented [45].

2.2 Bluetooth

Bluetooth Low Energy (BLE), which was introduced as part of the Bluetooth 4.0 specification, is an exciting wireless technology that gives mobile application developers unprecedented access to external hardware and provides hardware engineers with easy and reliable access to their devices from every major mobile operating system. In our study, Bluetooth is assumed to be the prototype of network connections in NCSs.

Not like other wireless network communication methods, Bluetooth has an intrinsic time-delay in the transmission. The main reasons for latency in Bluetooth communication lie in several aspects [46]:

1. Operation type
2. Connection intervals
3. Frames in every connection event
4. Length of frame data
5. Software delays

While Bluetooth Low Energy is a good technology on its own merit, what makes BLE genuinely exciting—and what has pushed its phenomenal adoption rate so far so quickly - is that it's the right technology, with the right compromises, at the right time. For a relatively young standard, BLE has seen an uncommonly rapid adoption rate, and the number of product designs that already include BLE puts it well ahead of other wireless technologies at the same point of time in their release cycles.

Compared to other wireless standards, the rapid growth of BLE is relatively easy to explain: BLE has gone further faster because its fate is so intimately tied to the phenomenal growth in smartphones, tablets, and mobile computing. Early and active adoption of BLE by mobile industry heavyweights like Apple and Samsung broke open the doors for wider implementation of BLE.

While the mobile and tablet markets become increasingly mature and costs and margins are decreasing, the need for connectivity with the outside world on these devices has a huge growth potential, and it offers peripheral vendors a unique opportunity to provide innovative solutions to problems people might not even realize that they have today. So many benefits have converged around BLE, and the doors have been opened wide for small, nimble product designers to gain access to a potentially massive market with task-specific, creative, and innovative products on a relatively modest design budget.

All-in-one radio-plus-microcontroller (system-on-chip) solutions today are only under $2 per chip and in low volumes, which is well below the total overall price point of similar wireless technologies such as WiFi, GSM, Zigbee, etc. And BLE allows you to design viable products today that can talk to any modern mobile platform using chips, tools, and standards that are easy to access. Perhaps one of the less visible key factors contributing to the success of BLE is that it was designed to serve as an extensible framework to exchange data. This is a fundamental difference with classic Bluetooth, which focused on a strict set of use cases. BLE, on the other hand, was conceived to allow anyone with an idea and a bunch of data points coming from an accessory to realize it without having to know a huge amount about the underlying technology. The smartphone vendors understood the value of this proposition early on, and they provided flexible and relatively low-level APIs to give mobile application developers the freedom to use the BLE framework in any way they see fit. Devices that talk to smartphones or tablets also offer another easy-to-underestimate advantage for product designers: they have an unusually low barrier to adoption. Users are already accustomed to using the handsets or tablets in their possession, which means the burden of learning a new UI is limited, as long as we respect the rich visual language that people have grown accustomed to in the platforms they use. With a relatively easy-to-understand data model, no intrusive licensing costs, no fees to access the core specs, and a lean overall protocol stack, it should be clear why platform designers and mobile vendors see a winner in BLE.

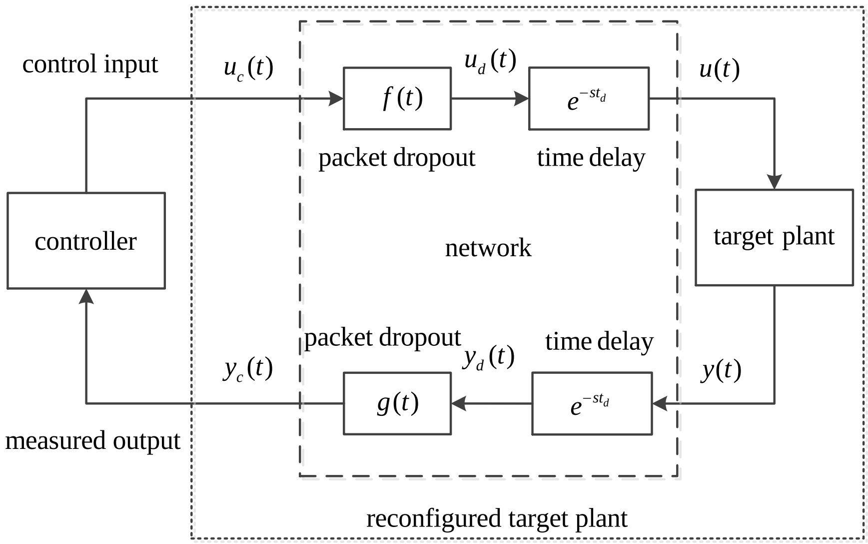
2.3 Problem formation

A typical NCS consisting of a target plant, a controller and network connecting is shown in Figure 4. Not like conventional unnetworked control systems, NCSs usually suffer from packet dropout and/or time delay. These abnormalities may incur instability or latency to the control system. Here we discuss control strategies with respects to both packet dropout and time delay in NCS.

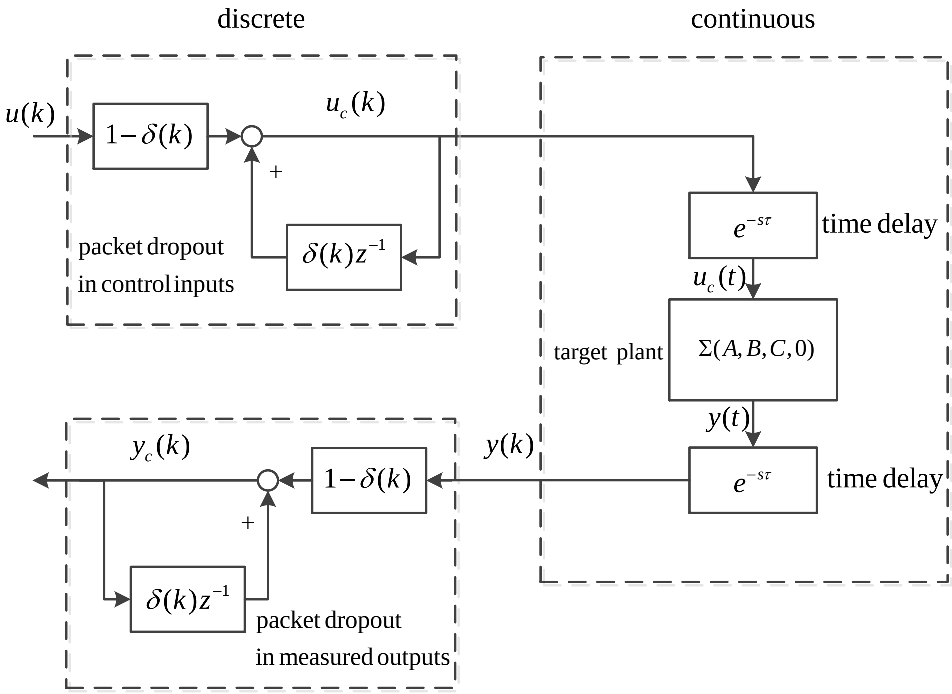
A detailed NCS is shown in Figure 5. Both data transmissions of control input and measured output are networked and both suffer from packet dropout and time delay. Sections of the target plant and network are merged and reconfigured into a new target plant. Since the issue we focus on is network in this study, for simplicity, a linear time-invariant (LTI) target plant is considered, whose state space is:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \tag{1}$$

where $x(t)$, $u(t)$ and $y(t)$ are states, input and output of the target plant respectively, while A, B and C are transition matrix, input gain and output gain of the LTI system.



a) Overall reconfiguration of networked control system



b) Details of reconfigured target plant

Figure 5 Reconfiguration of networked control system

As shown in the network section of Figure 5, the sequential order of time delay and packet dropout may be ignored due to commutative property of multiplication. Therefore, the positions of time delay and packet dropout can be switched. In order to keep consistency, two time delay sections are placed coherently to target plant, while packet dropout sections coherently to controller. Note the NCS in this study is not distributed, NCS with one controller and one actuator is considered while distributed NCSs with time-varying delay will be studied in future. The time delay equations are

$$\begin{cases} u(t) = u_d(t - t_d) \\ y_d(t) = y(t - t_d) \end{cases} \tag{2}$$

where t_d is the expectation of delayed time in the NCS, which is assumed to be a constant known to us measured and calculated ahead of time. This is due to the fact that delayed time in most NCSs does not change a lot and it is possible to be compensated by a robust control strategy. $u_d(t)$ is variable designated for control input before time delay, while $y_d(t)$ is variable designated for measured output after time delay. Besides, the dropout equations of packet dropout are

$$\begin{cases} u_d(t) = f(t)u_c(t) \\ y_c(t) = g(t)y_d(t) \end{cases} \tag{3}$$

where $f(t)$ and $g(t)$ are uncertain binary dropout functions irrelevant to each other, $u_c(t)$ is the control input derived directly from controller, while $y_c(t)$ is the very measured output controller receives. Packet dropout out takes place when the values of them are zero, vice versa. In a view of practical instance of NCS, data transmission is usually conveyed in packets intermittently at every transmission interval. As a result, dropout functions are actually discrete functions of massive dots. For convenience, $f(t)$ and $g(t)$ are usually proposed as piecewise defined function. Though overall NCS in Figure 5 differs from a normal control system consisting of only target plant and controller, we may reconfigure it with the help of equations (1)-(3) by merging sections of the target plant and network into a new reconfigured target plant, the state space is derived as

$$\begin{cases} \dot{x}(t) = Ax(t) + f(t - t_d)Bu_c(t - t_d) \\ y_c(t) = g(t)Cx(t - t_d) \end{cases} \tag{4}$$

The control problem of NCS is finally reformed as a control problem of a non-linear delayed target plant of state space (4). The control target of our study is to model and stabilize the reconfigured target plant (4), seeking an enhanced performance in such NCSs.

III. Modeling and control strategy design

3.1 Modelling

In this section, a further modelling process is required not only for the prediction model of model predictive controller but also a prerequisite for the usage of YALMIP toolbox in chapter III. There are three steps in this modelling process. Discretization is the very first. The reconfigured target plant is actually a hybrid system of both continuous and discrete when studying the dynamics of reconfigured NCS in (4). Whereas in typical NCSs, data is transmitted in packets every transmission interval, so it is rational to discretize the NCS at transmission period, designated as T_s, $T_s < t_d$. However, the time delay sections, of which the delayed time be utilized as sampling rate in normal occasion, is not easy to handle in such dual-time-rate system of T_s and t_d. Therefore, secondly, an augmented state space by Pade approximation is employed to eliminate the time-delay. Furthermore, a zero-order holder like measure for the packet dropout in control input and estimation for the measured output are proposed at last.

Whereas parameter A, B in (1) are constant known to us, discretized transition function of original target plan and time delay in control input at sampling rate T_s is supposed to be

$$x(k+1) = A_d x(k) + B_d U_d(k) \tag{5}$$

where $A_d$and B_d are corresponding discrete parameters, mark $U_d(t)$=$u_d(t - t_d)$. Then a discrete Pade approximation is applied to $U_d(k)$ as

$$\begin{cases} U_d(k) = C_\phi \phi(k) + D_\phi u_d(k) \\ \phi(k+1) = A_{\phi d} \phi(k) + B_{\phi d} u_d(k) \end{cases} \tag{6}$$

Similarly, Pade approximation in measured output may be written as

$$\begin{cases} X(k) = C_\varphi \varphi(k) + D_\varphi x(k) \\ \varphi(k+1) = A_{\varphi d} \varphi(k) + B_{\varphi d} x(k) \end{cases} \tag{7}$$

The corresponding parameters of state space in equations (6), (7) are derived by Matlab function *pade*. For simplicity, a state feedback is supposed instead of state observer in this study, since we are not concerned about an observer. Thus, $y(k) = Ix(k) = x(k)$ where I is identical matrix. Afterwards, mark $X(t) = x(t - t_d)$ likely.

Considering the packet dropout sections of both the control input and measured output, it is impossible to acknowledge whether packet dropout happens or not in control input. Besides, by checking the validity of data transmitted, the occurrence of packet dropout in measured output is possible to be detected. Therefore, two difference measures are proposed for each packet dropout sections. Before that, the property of the dropout functions $f(t)$, $g(t)$ should be learned at first. It is obvious that dropout functions $f(t)$and $g(t)$ are stochastic uncertainties unable to learn. According to that, it is rational to discretize $f(t)$and $g(t)$ as $F(k)$ and $G(k)$, respectively, which are also binary stochastic functions.

On the control input's side, we propose a zero-order holder that, the data of control input transmitted would not drop to zero when packet dropout happens, but keep the same with the last time,

$$u_d(k) = F(k)u_c(k) + [1 - F(k)]u_d(k-1) \tag{8}$$

On the measured output's side, we propose a measure that estimation updated time to time since packet dropout happens or not is detectable checking the data validity. If packet dropout happens, the measured output must be invalid. Therefore, an estimated measured output may be obtained by

$$y_c(k) \approx \hat{X}(k) = E[G(k)]X(k)|_{F(k-1)=1} + \{1 - E[G(k)]\}X(k)|_{F(k-1)=0} \tag{9}$$

where $E[G(k)] = {}^{n}/_{k}$, n is the times that $G(i) = 1, i = 1, \dots, k$. Finally, we can get the augmented simultaneous functions (10) from equations (5)-(8) as

$$\begin{bmatrix} x(k+1) \\ u_d(k) \\ \phi(k+1) \\ \varphi(k+1) \end{bmatrix} = \begin{bmatrix} A_d & B_d[1-F(k)]D_\phi & B_dC_\phi & 0 \\ 0 & 1-F(k) & 0 & 0 \\ 0 & [1-F(k)]B_{\phi d} & A_{\phi d} & 0 \\ B_{\varphi d} & 0 & 0 & A_{\varphi d} \end{bmatrix} \begin{bmatrix} x(k) \\ u_d(k-1) \\ \phi(k) \\ \varphi(k) \end{bmatrix} + \begin{bmatrix} F(k)B_dD_\phi \\ F(k) \\ F(k)B_{\phi d} \\ 0 \end{bmatrix} u_c(k) \tag{10}$$

For simplicity, we designate equation (10) as

$$\xi(k+1) = A_{\xi d}(k)\xi(k) + B_{\xi d}(k)u_c(k) \tag{11}$$

where

$$\xi(k) = \begin{bmatrix} x(k) \\ u_d(k-1) \\ \phi(k) \\ \varphi(k) \end{bmatrix}, A_{\xi d}(k) = \begin{bmatrix} A_d & B_d[1-F(k)]D_\phi & B_d C_\phi & 0 \\ 0 & 1-F(k) & 0 & 0 \\ 0 & [1-F(k)]B_{\phi d} & A_{\phi d} & 0 \\ B_{\varphi d} & 0 & 0 & A_{\varphi d} \end{bmatrix}$$

$$B_{\xi d}(k) = [F(k)B_d D_\phi \quad F(k) \quad F(k)B_{\phi d} \quad 0]^T \tag{12}$$

3.2 Control strategy

An enhanced model predictive controller is proposed since reconfigured transition function (11) is time-variant and multi-parametric and such prediction model is not supported by common model predictive controller. Therefore, third-party Matlab toolbox YALMIP [47], a general parser for linear matrix inequalities and multi-parametric programming solver MPT3 (Multi-Parametric Toolbox) are introduced [48]. Firstly, the control target may be written as

$$arg\ min_{u_c(0)} max_{\Xi} \sum_{k=0}^{N-1}[\Delta x^T(k)Q\Delta x(k) + u_c^T(k)Ru_c(k)] + \Delta x^T(N)P\Delta x(N) \tag{13a}$$

subject to

$$\Xi: \xi(k+1) = A_{\xi d}(k)\xi(k) + B_{\xi d}(k)u_c(k) \tag{13b}$$

$$x(k) \in [x_{min}, x_{max}], u_c(k) \in [u_{c_{min}}, u_{c_{max}}], F(k) \in \{0, 1\} \tag{13c}$$

where $\Delta x = x - x_{ref}$, x_{ref} is the reference signal, Q, R and P are weighting constants, x_{min}, x_{max} and $u_{c_{min}}, u_{c_{max}}$ are lower and upper limits of $x(k)$ and $u_c(k)$, N is the predictive horizon of measured output and control input of model predictive controller.

A conventional method to handle optimization function (13a) is rewriting it as

$$\min \omega$$

subject to

$$\sum_{k=0}^{N-1}[\Delta x^T(k)Q\Delta x(k) + u_c^T(k)Ru_c(k)] + \Delta x^T(N)P\Delta x(N) < \omega \tag{14}$$

where ω is epigraph variable [49]. Therefore, the optimization problem is reformed into a linear matrix inequality (LMI) (14). There have been plenty of studies and tutorials about how to solve LMI problems. Unconstrained ones are usually solved by Riccati equation [50], while constrained ones may be expanded into a high order LMI via Schur complement then solved by Matlab toolboxes [51].

In the occasion of NCS of this study, however, prediction model (13b) is uncertain and stochastic, the optimization problem is impossible to be solved instinctively by a conventional method. In a view of robustness, all possible states of prediction model (13b) in future should be taken into consideration. Still more, uncertainty of transition function (13b) at time k may be regarded as a linear parametric-varying function according to parameter $F(k)$, hence,

$$A_{\xi d}(k)|_{F(k)=1} = \begin{bmatrix} A_d & 0 & B_d C_\phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & A_{\phi d} & 0 \\ B_{\varphi d} & 0 & 0 & A_{\varphi d} \end{bmatrix},$$

$$A_{\xi d}(k)|_{F(k)=0} = \begin{bmatrix} A_d & B_d D_\phi & B_d C_\phi & 0 \\ 0 & I & 0 & 0 \\ 0 & B_{\phi d} & A_{\phi d} & 0 \\ B_{\varphi d} & 0 & 0 & A_{\varphi d} \end{bmatrix},$$

$$B_{\xi d}(k)|_{F(k)=1} = [B_d D_\phi \quad I \quad B_{\phi d} \quad 0]^T,$$

$$B_{\xi d}(k)|_{F(k)=0} = [0 \quad 0 \quad 0 \quad 0]^T \tag{15}$$

Furthermore, uncertain variables $\theta_0(k)$ and $\theta_1(k)$ are introduced as

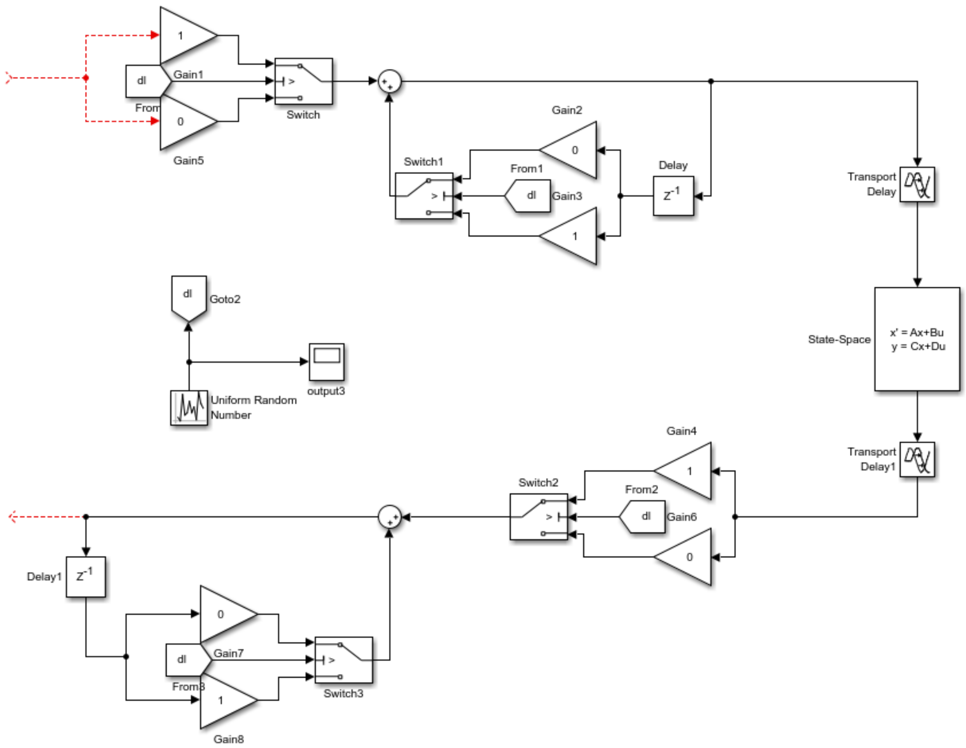
$$\begin{cases} \theta_0(k) = 0 \\ \theta_1(k) = 1 \end{cases}, F(k) = 1, \begin{cases} \theta_0(k) = 1 \\ \theta_1(k) = 0 \end{cases}, F(k) = 0 \tag{16}$$

Then, the transition function (13b) may be rewritten in linear parametric-varying format that is supported by YALMIP as follows

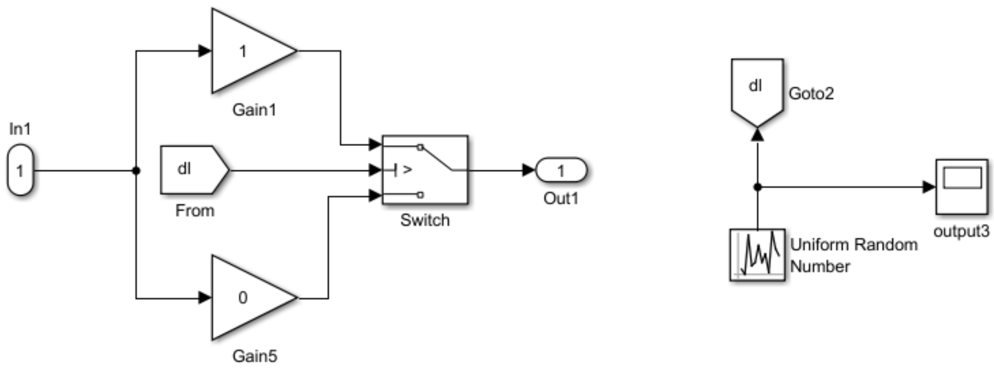
$$\xi(k+1) = \theta_1(k) A_{\xi d}(k)|_{F(k)=1} \xi(k) + \theta_0(k) A_{\xi d}(k)|_{F(k)=0} \xi(k) +$$

$$\theta_1(k)B_{\xi d}(k)|_{F(k)=1}u_c(k) + \theta_0(k)B_{\xi d}(k)|_{F(k)=0}u_c(k) \quad (17)$$

Notice programming problems in (13), (16) and (17) are not supported by official toolbox of Matlab while third-party solvers such as MPT3, SeDuMi, and etc. are useful extensions of Matlab, moreover, YALMIP is a general parser choosing solver, and translating programming problem from mathematical expressions into functions that third-party solvers can handle [52]. In this case, solver MPT3 is called by YALMIP. Firstly, YALMIP parses the optimization problem (13), enumerating all possible occasions and eliminating the uncertain variables $\theta_0(k)$ and $\theta_1(k)$. Then the Karush-Kuhn-Tucker conditions [53] of expanded LMI are examined by MPT3 in this way and critical regions are derived and optimizer function in each critical region is feasible to be worked out.



a) Time-delay and packet dropout in Simulink model



b) Details of packet dropout

Figure 6 Modeling partition in Simulink model

Figure 6 shows some partition model in Simulink project of our study, and some Matlab codes are also shown as below:

```
% YALMIP options
  yalmip('clear')
  yopts = sdpsettings('robust.polya',1);
% Model data
  A1 =[];
  A2 = [];
  B1 = [];
  B2 = [];
  B{1} = B1;
  B{2} = B2;
% System sizes
  nx   = 2; % Number of states
  nu   = 1; % Number of inputs
  ndyn = 2; % Number of vertex systems
```

```
% State and input constraints
   xmin = [-10;-10];
   xmax = [  8;  8];
   umin = -0.5;
   umax =   1 ;
% MPC data
   Q   = eye(nx);
   R   = 0.01;
   N   = 3;
   xref = [0;0];
% States x(k), ..., x(k+N)
   x = sdpvar(repmat(nx,1,N),repmat(1,1,N));
% Inputs u(k), ..., u(k+N) (last one not used)
   u = sdpvar(repmat(nu*ndyn,1,N),repmat(1,1,N));
% Scheduling parameter
   th = binvar(repmat(ndyn,1,N),repmat(1,1,N));
% Epigraph variable
   sdpvar w;
   for k = N:-1:1   % shifted: N-1:-1:0
   % Parameter simplex
     F = [uncertain(th{k}), sum(th{k}) == 1, 0 <= th{k} <= 1];
   % Epigraph variable, MPT requires a bounded set so let us add artificial upper bound
     F = [F, 0 <= w <= 10000];
   % Uncertain predictions and control
     uth = kron(th{k},eye(nu))'*u{k};    % u(th)
     Ath = [A1 A2]*kron(th{k},eye(nx));  % A(th) = A1*th1 + A2*th2 + ...
     Bth = [B1 B2]*kron(th{k},eye(nu));  % B(th)
```

```
  xp  = Ath*x{k} + Bth*uth;
% Input constraints
  F = [F, repmat(umin,ndyn,1) <= u{k} <=  repmat(umax,ndyn,1)];
% State constraints
  F = [F, xmin  <= x{k} <= xmax];
% Insert step-specific code here

  if k == N
 % Initial step
 % |x| written as max(a[x;u]+b)
     a = [kron(eye(nx),[1 -1]') zeros(2*nx,nu)];
     b = zeros(2*nx,1);
 % |u| written as max(c[x;u]+d)
     c = [zeros(2*nu,nx) kron(eye(nu),[1 -1]')];
     d = zeros(2*nu,1);
 % |x|+|u| written as max(aa[x;u]+bb)
     aa = repmat(a,2*nu,1) + kron(c,ones(size(a,1),1));
     bb = repmat(b,2*nu,1) + kron(d,ones(size(a,1),1));
 % Final state constraints
     F = [F, xmin  <= xp   <= xmax];
 % Cost function
     F   = [F, aa*[Q*(xp-xref);R*uth]+bb + norm(Q*(x{k}-xref),inf) <= w];
     obj = w;

  elseif k > 1
 % Intermediate step
 % Get the hyperplanes for cost-to-go
```

```
unique(([reshape([sol{k+1}{1}.Bi{:}]' ,nx,[])'  reshape
([sol{k+1}{1}.Ci{:}]' ,[],nu)]),'rows');
        a = [S(:,1:nx) zeros(size(S,1),nu)];
        b = S(:,nx+1:end);
    % Jp+|u| = max(a[x;u]+b)+|u|=max(a[x;u]+b)+max(c[x;u]+d)=max(aa[x;u]+bb)
        aa = repmat(a,2*nu,1) + kron(c,ones(size(a,1),1));
        bb = repmat(b,2*nu,1) + kron(d,ones(size(a,1),1));
    % Constrain predicted state
        [H,K] = double(sol{k+1}{1}.Pfinal);
        F     = [F, H*xp <= K];
    % Cost function
        F   = [F, aa*[xp;R*uth]+bb <= w];
        obj = norm(Q*(x{k}-xref),inf) + w;

      else
    % Final step
    % Get the hyperplanes for cost-to-go
        S = unique(([reshape([sol{k+1}{1}.Bi{:}]',nx,[])' reshape
([sol{k+1}{1}.Ci{:}]',[],nu)]),'rows');
        a = [S(:,1:nx) zeros(size(S,1),nu)];
        b = S(:,nx+1:end);
    % Jp+|u| = max(a[x;u]+b)+|u|=max(a[x;u]+b)+max(c[x;u]+d)=max(aa[x;u]+bb)
        aa = repmat(a,2*nu,1) + kron(c,ones(size(a,1),1));
        bb = repmat(b,2*nu,1) + kron(d,ones(size(a,1),1));
    % State update equation
        xp = x{k} + Bth*uth; % x{1} = z
    % Constrain predicted state
```

```
        [H,K] = double(sol{k+1}{1}.Pfinal);
        F     = [F, H*xp <= K];
    % Cost function
        F   = [F, aa*[xp;R*uth]+bb <= w];
        obj = w;
    % add eps-penalty on vertex predictions
        for v = 1:ndyn
            xpv{v} = x{k} + B{v}*u{k}(v);
            obj = obj + 0.001*norm(Q*xpv{v},inf);
        end
      end

    % Determine robust counterpart
      [F,obj] = robustify(F,obj,yopts,th{k});
    % Solve multi-parametric problem
      [sol{k},diagnost{k},Uz{k},J{k},Optimizer{k}] = solvemp(F,obj,yopts,x{k},u{k});
    end
}
```

In order to increase solve speed and reduce computation load, we introduce a discrete dynamic programming method which solves programming problems optimally by breaking them into sub-problems and finding the optimal solution recursively.

For programming problem of dynamics $x_{i+1} = f(x_i, u_i)$ with cost function

$$J_0(x, U_0) = \sum_{i=0}^{N-1} l(x_i, u_i) + l_f(x_N),$$

$$U_0 = \{u_0, u_1, \dots, u_{N-1}\} \tag{18}$$

where l is the running cost and l_f is the final cost. Define partial cost function as

$$J_i(x, U_i) = \sum_{j=i}^{N-1} l(x_j, u_j) + l_f(x_N),$$

$$U_i = \{u_i, u_{i+1}, \dots, u_{N-1}\} \tag{19}$$

Thus, value function may be written as

$$V(x, i) = \min_{U_i} J_i(x, U_i) \tag{20}$$

when $i = 0$, $V(x, 0) = \min_{U_0} J_0(x, U_0)$ is the final objective of programming problem. Notice the value function (20) may be decomposed into Bellman equation [54] format. Therefore,

$$V(x, i) = \min_{U_i}[l(x_i, u_i) + V(x, i+1)] = \min_{u_i}[l(x_i, u_i) + V(f(x_i, u_i), i+1)] \tag{21}$$

Conclusively in this study, running cost is $\Delta x^T(k) Q \Delta x(k) + u_c^T(k) R u_c(k)$, and final cost is $\Delta x^T(k) P \Delta x(N)$, the programming problem of model predictive controller may be solved by solving partial value function $V(x, i)$ recursively from $i = N - 1$ to $i = 0$.

IV. Numeric simulations and experiments

In order to demonstrate the performance of proposed method, two simulations are presented. The first one is based on a fixed-wing aircraft model. In the first simulation, step responses of attitude are studied in order to show feasibility and stability of proposed method. A numeric simulation of second order LTI system is carried out in the other example, where the robustness as well as real-time property of proposed method is studied.

4.1 Example 1:

Consider a MIMO (Multiple-Input and Multiple-Output) aircraft system [55] as target plant in NCS with the state space as

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} -0.0151 & -60.565 & 0 & -32.174 \\ -0.0001 & -1.3411 & 0.9929 & 0 \\ 0.00018 & 43.254 & -0.8694 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ \qquad * x(t) + \begin{bmatrix} -2.516 & -13.136 \\ -0.1689 & -0.2514 \\ -17.251 & -1.5766 \\ 0 & 0 \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(t) \end{cases} \tag{22}$$

The aircraft model is shown in Figure 7. Control inputs $u(t)$ are elevator and flaperon angles shown in Figure 7, ranging between ± 25 and ± 75 degrees, respectively, while measured output $y(t)$ are attack and pitch angles ranging between ± 1 and ± 75 degrees respectively, which are also designated as α and β in Figure 7. Sampling period also known as transmission interval in this study is 0.025 seconds.

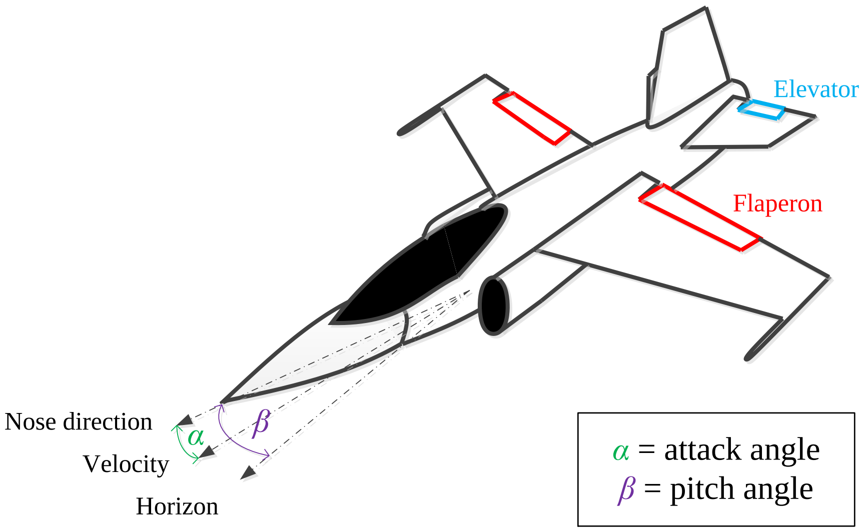
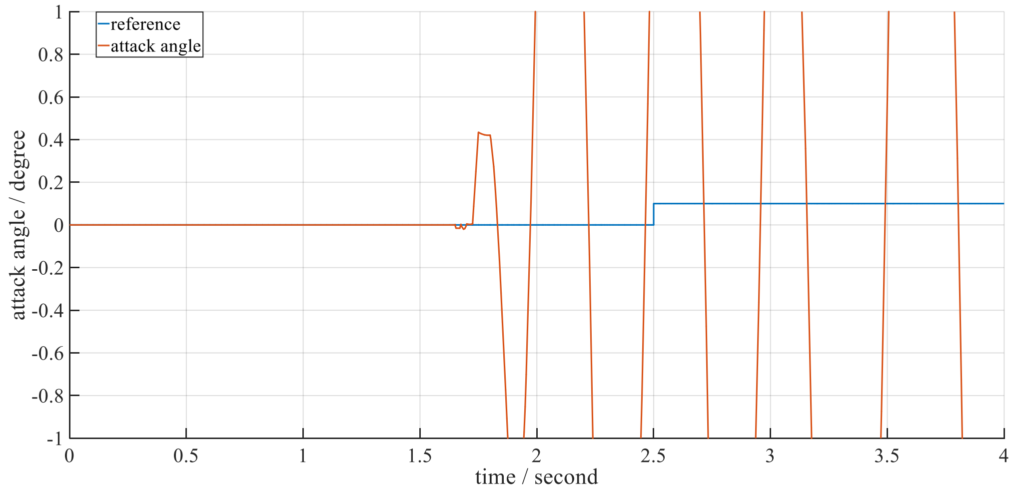
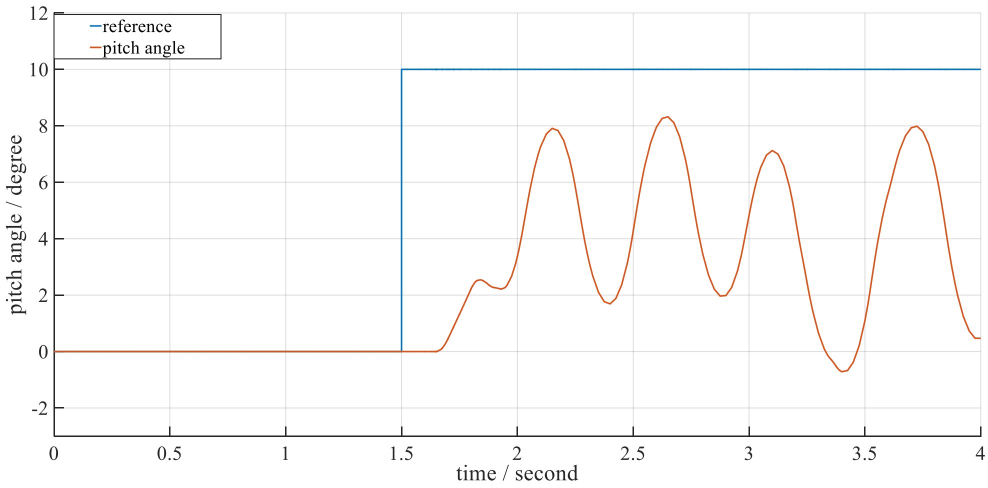


Figure 7 Brief diagram of aircraft model of (22)

Step response results of NCS in Figure 5 with the target plant (22) are shown in Figure 8, which is regulated by a conventional model predictive controller. Time delay is 0.1 seconds and dropout rate is 0.1 of both control input and measured output networks.



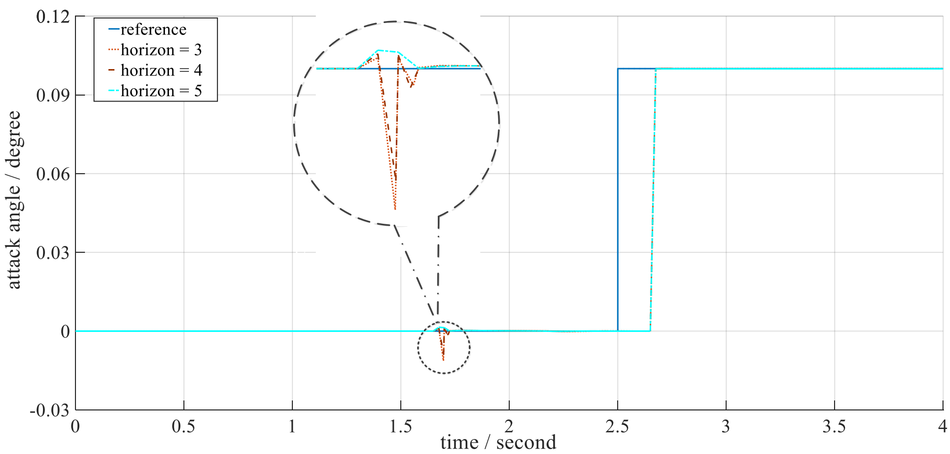
a) Waveform of attack angle



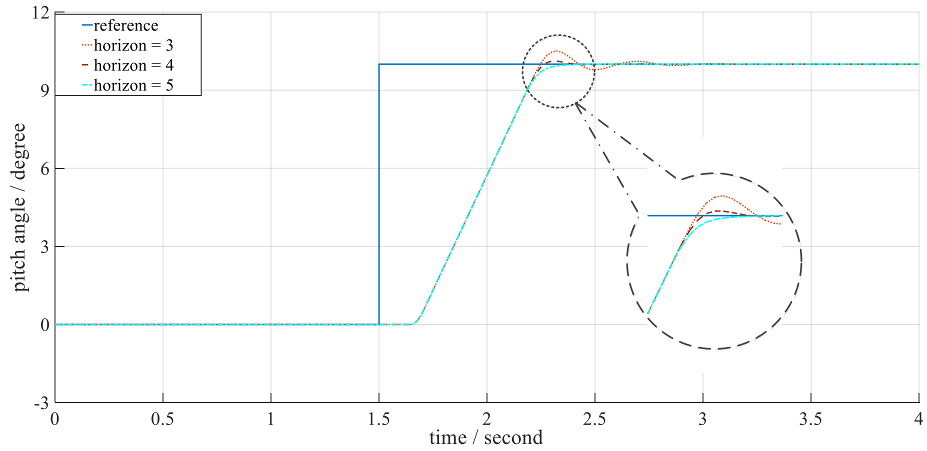
b) Waveform of pitch angle

Figure 8 Step responses of networked control target plant (22) by a conventional model predictive controller

The latency and stochasticity in NCS incur severe instability of studied flight system as is shown in Figure 8, especially in Figure 8 a), the curves overshoot more than 30 outside the figure. In order to demonstrate the advantages of proposed method, simulation results employing proposition under severer conditions are shown in Figure 5, with a delayed time of 0.15 seconds and a dropout rate of 0.25.



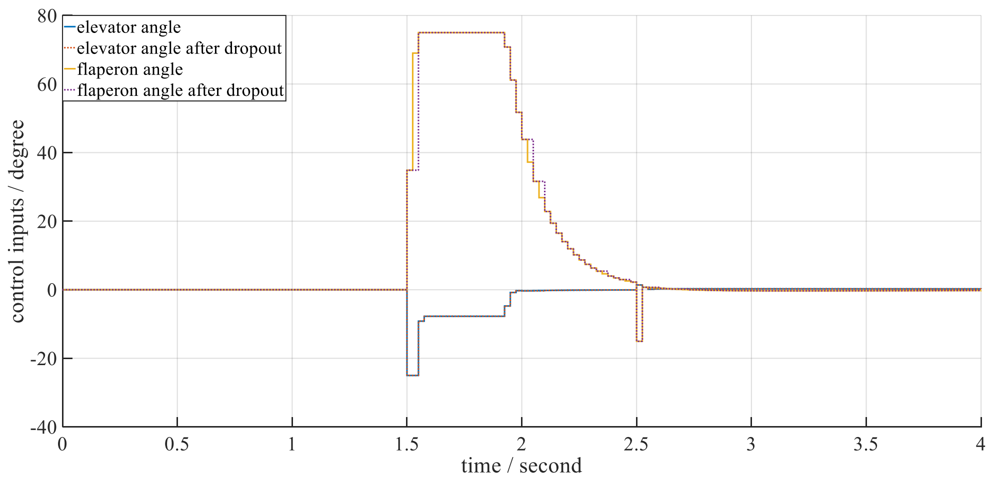
a) Partial waveforms of attack angle under variant prediction horizons



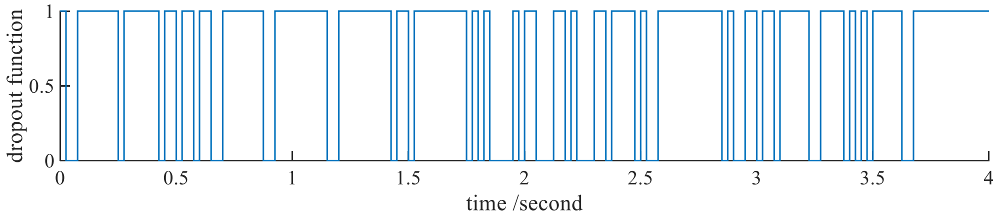
b) Waveforms of pitch angle under variant prediction horizons

Figure 9 Step responses of networked control target plant (22) by proposed method under variant prediction horizon from 3 to 5

Variant prediction horizons are inspected as shown in Figure 9, from 3 to 5 time/second. All of the predictive horizons inspected can keep the NCS stable. As prediction horizon increases, undershoot of attack angle around 1.7 seconds in Figure 9 a) is smoothed gradually, vibration of pitch angle around 2.3 seconds in Figure 9 b) is flattened. The performance is adequately excellent when prediction horizon grows as large as 5.



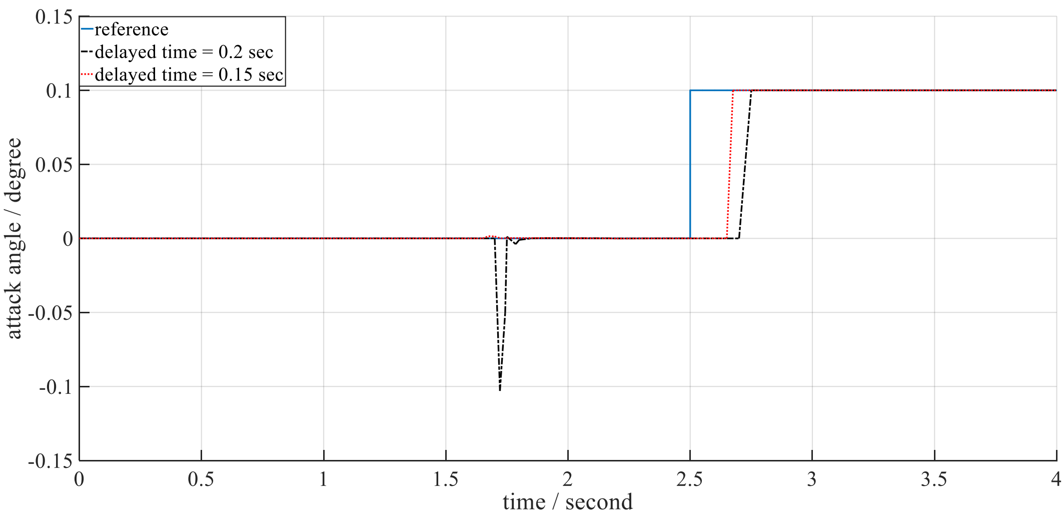
a) Waveforms of control inputs



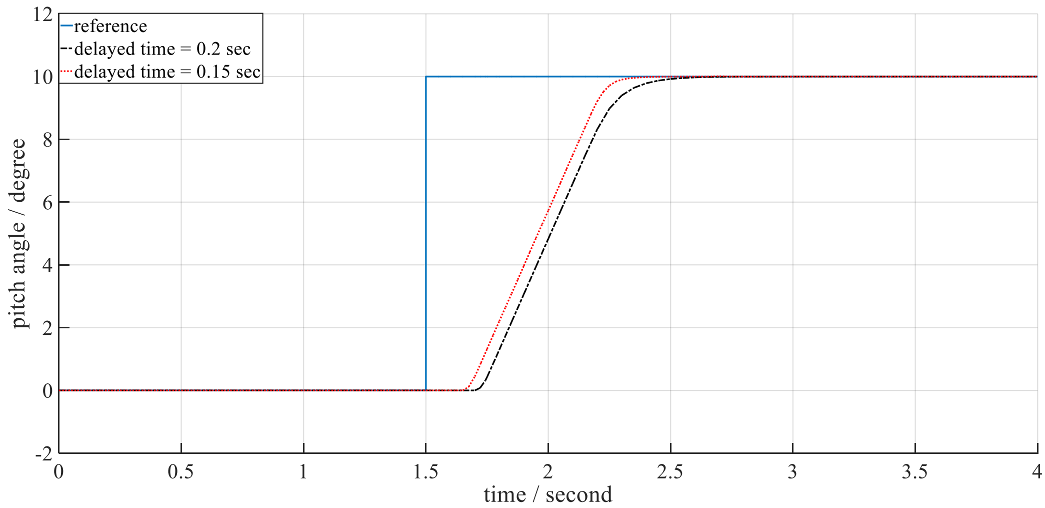
b) Values of dropout function

Figure 10 a) Control inputs of networked control target plant (22) by proposed method under prediction horizon of 5, before and after packet dropout section, b) dropout function

Control inputs of elevator and flaperon angle using proposed method are shown in Figure 10, of which the prediction horizon is fixed to 5. Curves of control inputs ($u_c(t)$ in Figure 5) after packet dropout section ($u_d(t)$ in Figure 5) are well compensated by proposed zero-order holder compensator as shown in Figure 10 a). The values may not dropout to zero when packet dropout happens in packet dropout section of Figure 5. Figure 10 b) is the stairstep graph of dropout function, pack dropout happens when the value is 0, vice versa.



a) Waveforms of attack angle under variant delayed times



b) Waveforms of pitch angle under variant delayed times

Figure 11 Step responses of networked control target plant (22) under variant delayed times regulated by proposed method with prediction horizon of 5 and predictive model delay of 0.15 seconds

Figure 11 demonstrates step response curves of proposed method as delayed time changes slightly. The parameters and solution of proposed method are intended to deal with networked control target plant (22) under delayed time of 0.15 seconds. Results in Figure 11 shows that the whole control system stays stable when delayed time shifts to 0.2 seconds. While the performance is slightly downgraded as some peak undershoots takes place in attack angle as shown in Figure 11 a).

4.2 Example 2:

Another numeric simulation is presented, in which the target plant is in second order. Therefore, 3-dimensional graphs of value and control input function are possible to be plotted. In this way, more properties of proposed method are possible to be acknowledged.

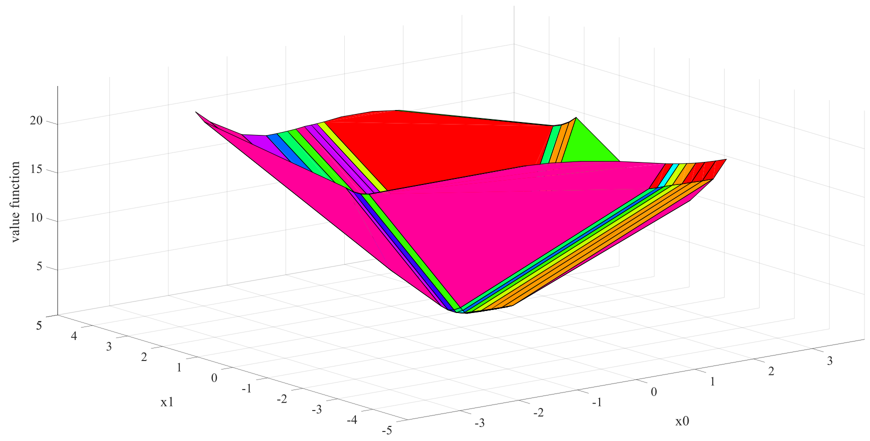
Consider target plant of

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t) \\ z(t) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} x(t) \end{cases} \tag{23}$$

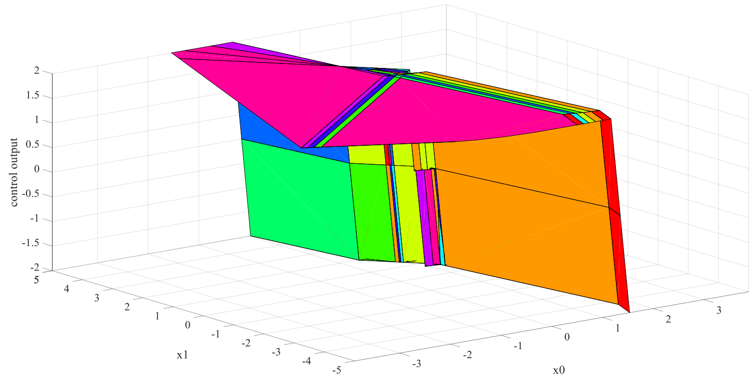
with transmission interval of 0.1 seconds, control input and measured output delay both 0.3 seconds, prediction horizon N of 4. $y(t)$ is measured output, and $z(t)$ is the desired output. Boundary constraints of states and control inputs are

$$\begin{cases} -5 \le x(k) \le 5 \\ -2 \le u(k) \le 2 \end{cases} \tag{24}$$

The discretization and Pade approximation processes are carried out by Matlab functions. After the discretization and Pade approximation, we would like to solve the minimax problem of inequalities by a zero initial states in order to derive the explicit solution, which is in the form of piece-wise affine function sets in each critical region, including value function and control input function. The results are shown in Figure 12.



a) Value function



b) Control input function

Figure 12 Explicit solution of proposed method with regard to target plant (23)

Hyperplane in Figure 12 a) demonstrates the cost function, which is employed to weigh the cost of a robust control strategy. When comparing to simulation result of [56], proposed method is 3.67% less, which means method in this paper has a remarkable robustness relatively. Hyperplane in Figure 12 b) is the control input function. The value of control input may be derived from the hyperplane when parameters of reference and target plant are unchanged. Accordingly, the minimax problem is necessarily to be solved if only parameters of NCS are updated, while conventional method needs to solve programming problem every time. Moreover, a single minimax problem of 4-step model predictive control with proposed discrete dynamic programming method takes 67.68 seconds by an AMD FX8310 CPU to solve, while it takes 211.24 seconds for same control strategy without discrete dynamic programming method. As a result, proposed method can effectively save more time of solving, however, the algorithm needs further optimized to make such NCS more practical. A step response of this example is also presented as below.

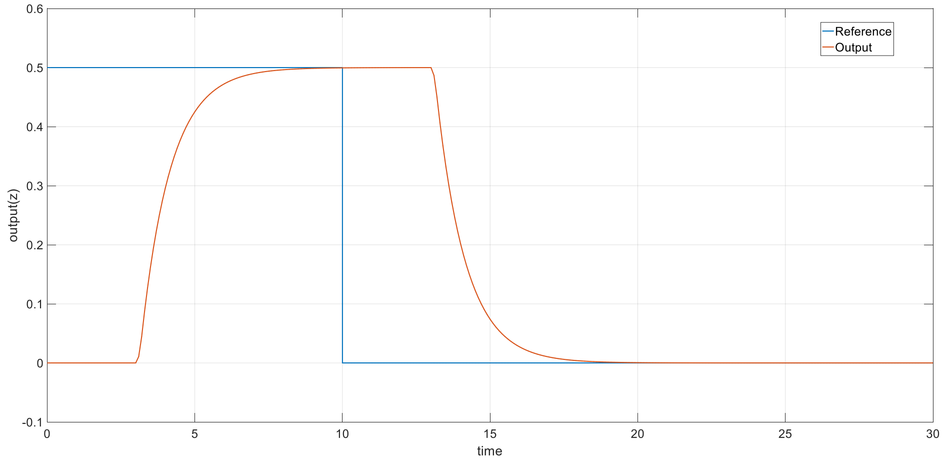


Figure 13 Step response of target plant (23) using proposed method

The curves of output in Figure 13 take 5 steps to converge with reference after the transmission delay of 3 steps, that is, 0.3 seconds. For comparison, control group of PID control method [56] underperforms unstable results in dealing with such NCSs, of which the curves diverge rapidly within several steps.

V. Conclusion

In Chapter I and II, we introduce networked control systems and some basic conceptions around them. Firstly, we introduce the background and history of networked control systems, then the four elements, controller, sensor, actuator and network, in networked control systems are also been discussed. Accordingly, the main issues in networked control systems are expressed. We also express control of network and control over network, two main measures in networked control system as well. Bluetooth is chosen as the prototype network meanwhile. Dozens of literatures are reviewed and compared, with both their innovative propositions and defects.

Towards issues of time-delay and packet dropout, we have proposed a novel model predictive controller towards networked control systems featured with time delay and packet dropout in Chapter III and IV. First of all, target plant and network are reconstructed and reformed into a reconfigured target plant, then in the modelling process, sections of time delay are eliminated by Pade approximation, moreover, a zero-order holder like compensator is utilized for packet dropout in control input, meanwhile packet dropout in measured output is compensated by estimation. Followed the control target is reformed as a control problem of an uncertain time-varying multi-parametric state space. Furthermore, in order to apply toolbox YALMIP and solver MPT3, such a state space is rewritten as a linear parametric-varying prediction model. At last, a robust model predictive controller is completed by solving a constrained minimax problem of predictive cost function. The simulation results have shown a robust performance of proposed method towards packet dropout and time delay in networked control system. The proposition in this study performs outstanding stability, also takes advantages in shortening solving time and enhancing the robustness of networked control systems with time delay and packet dropout. Meanwhile, we have considered an LTI target plant in this study for simplicity since the subject we focus on is networked control systems with time delay and packet dropout. That does not mean proposed method is not suitable for other target plant, say, nonlinear ones.

Theoretically, the work in this thesis is partial and insufficient, more aspects and subjects, such as more control strategic control methods, nonlinearity and uncertainties in target plants, distributed time-delay and etc., would be well complementation to this thesis, all these aspects and subjects should be taken into consideration when continuing future studies to this thesis.

References

[1] H. Chaouchi, The Internet of Things. London: Wiley-ISTE, 2010, pp. 1-35

[2] R. Aburukba, A. R. Al-Ali, N. Kandil, and D. AbuDamis, “Configurable ZigBee-based Control System for People with Multiple Disabilities in Smart Homes,”, International Conference on Industrial Informatics and Computer Systems (CIICS 16), Mar. 2016, pp. 1–5, doi:10.1109/ICCSII.2016.7462435

[3] Crouch, T. D., “First flight: The Wright Brothers and the invention of the airplane”, published by Washington, D.C.: Produced by the Division of Publications, Harpers Ferry Center National Park Service, U.S. Department of the Interior, 2002.

[4] Mahalik, N.N. P.; Kim, “K.K.A Prototype for Hardware-in-the-Loop Simulation of a Distributed Control Architecture”, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 38, Issue 2, pp.189 –200, March 2008.

[5] Chow, M-Y; Tipsuwan, Y, “Time sensitive network-based control systems and applications”, IEEE IES Network Based Control Newsletter 2005.

[6] S. Wang and E. J. Davison, “On the stabilization of decentralized control systems”, IEEE Trans. Automatic Control, Vol.AC-18, no.5, pp.473-478, 1973.

[7] F. Y. Wang, D. Liu, S. X. Yang and L. Li,. “Networking, Sensing, and Control for Networked Control Systems: Architectures, Algorithms, and Applications” in the IEEE Transactions on Systems, Man, and Cybernetics–Part C, vol. 37, no. 2, Mar. 2007.

[8] D. Siljak, “Decentralized Control of Complex Systems”, New York: Academic, 1991.

[9] R. M. Murray, K. J. Astrom, S. P. Boyd, R. W. Brockett and G Stein, "Future Directions in Control in an Information-Rich World", IEEE Control Systems Magazine, April 2003.

[10] Silviu-Iulian Niculescu, "On delay robustness analysis of a simple control algorithm in high-speed networks", Automatica, 38, 2002, pp. 885-889.

[11] J. Yu, S. Yu and H. Wang, "Survey on the Performance Analysis of Networked Control Systems", IEEE Conference on Systems, Man and Cybernetics, The Hague, Netherlands, 2004.

[12] G. C. Walsh, Y. Hong and L. G. Bushnell, "Stability Analysis of Networked Control Systems", IEEE Trans. On Control Systems Technology, 10(3), May 2002, pp. 438-46.

[13] W. Zhang, M. S. Branicky and S. M. Philips, "Stability of Networked Control Systems", IEEE Control System Magazine, 21(1), Feb. 2001, pp. 84-99.

[14] L. Guan, M. E. Woodward and I. U. Awan, "Performance Analysis of Maintaining Delay Constrains in a Discrete-time Finite Capacity Queue", Procs of the 5th Informatics Workshop for Research Students edited by H Yu, School of Informatics, Bradford University, UK, 17 March 2004.

[15] Mahalik, N.N. P.; Kim, K.K.; A Prototype for Hardware-in-the-Loop Simulation of a Distributed Control Architecture", IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 38, Issue 2, March 2008, pp.189 –200.

[16] Farsi, M, Ratcliff, K and Barbosa, M. "An overview of controller area network", Computing and control engineering journal, vol. 10, June 1999, pp. 113-120.

[17] Wang Z, Yu H, Wang H; Xu A and Zhou Y. "The research of XML-based extensible device description for FF fieldbus system", IECON 2004, pp.2600 – 2603.

[18] Park Z. W and Kim M. K, "Design of an integrated fieldbus gateway", Proceedings of Russian-Korean International Symposium on Science and Technology, 2005, pp.843 – 846.

[19] Lee, K. C, Lee, S and Lee, M. H, "Worst Case Communication Delay of Real-Time Industrial Switched Ethernet With Multiple Levels," IEEE Transactions on Industrial Electronics, vol.53, no.5, Oct. 2006, pp.1669-1676.

[20] R. A. Gupta and M. Chow, "Networked Control System: Overview and Research Trends," IEEE Trans. On Industrial Electronics, vol.57(7), Jul. 2017, pp. 2527-2535, doi: 10.1109/TIE.2009.2035462

[21] H. Mehrivash and M. H. Shafiei, "Robust Model Predictive Control of discrete-time delayed positive systems," 2017 5th International Conference on Control, Instrumentation, and Automation (ICCIA), Nov. 2017, pp. 150-154, doi: 10.1109/ICCIAutom.2017.8258669

[22] G. Torres, M. Velasco, P. Marti, J. M. Fuertes and E. X. Martin, "An Alternative Discrete-Time Model for Networked Control Systems with Time Delay Less than the Sampling Period," 39th Annual Conference of the IEEE industrial Electronics Society, Nov. 2013, pp. 5627-5631, doi: 10.1109/IECON.2013.6700056

[23] J. Wu and T. Chen, "Design of networked control systems with packet dropouts," IEEE Trans. on Automatic Control, vol. 52, July 2007, pp. 1314-1319, doi: 10.1109/TAC.2007.900839

[24] P. Seiler and R. Sengupta, "An H∞ approach to networked control," IEEE Trans. on Automatic Control, vol. 50, March 2005, pp. 356-364, doi: 10.1109/TAC.2005.844177

[25] J. Wu, L. Zhang, and T. Chen, "An MPC approach to networked control design," Proc. of the 26th Chinese Control Conference (CCC 07), Jul. 2007, pp. 10-14, doi: 10.1109/CHICC.2006.4346915

[26] D. Nesic and A. Teel, "Input-Output stability properties of networked control systems," IEEE Trans. on Automatic Control, vol. 49, Oct. 2004, pp. 1650-1667, doi: 10.1109/TAC.2004.835360

[27] J. Wu and T. Chen, "Design of networked control systems with packet dropouts," IEEE Trans. on Automatic Control, vol. 52, Jul. 2007, pp. 1314-1319, doi: 10.1109/TAC.2007.900839

[28] F. Yang and Q. Han, "H∞ Control for Networked Systems with Multiple Packet Dropouts," Information Sciences, vol. 252, Dec. 2013, pp. 106-117, doi: 10.1016/J.INS.2013.06.043

[29] W. Zhang, J. Bae and M. Tomizuka, "Modified preview control for a wireless tracking control system with packet loss," IEEE/ASME Trans. On Mechatronics, vol. 20(1), Feb. 2015, pp.299-307, doi: 10.1109/ TMECH.2013.2297151

[30] Wang F. Y, "Overview of Agent-based Control and Management for NCS". In: Wang FY., Liu D. (eds) Networked Control Systems. Springer, London, 2008

[31] Carnevale D, Teel A. R and Nesic D, "A Lyapunov proof of an improved maximum allowable transfer interval for networked control systems". IEEE Trans Automat Control 52(5), 2007, pp. 892–897.

[32] Liu G. P, Rees D, Chai S. C and Nie X. Y, "Design, simulation and implementation of networked predictive control systems". Measurement & Control 38, 2005, pp.17–21.

[33] Zhang H. G and Cai L. L, "Decentralized nonlinear control of a HVAC system", IEEE Trans Systems Man Cybernetics-Part C: Applications and Reviews 32, 2002, pp.493–498.

[34] Sun F and Wu F, "A Discrete-time Jump Fuzzy System Approach to NCS Design". In: Wang FY., Liu D. (eds) Networked Control Systems. Springer, London, 2008.

[35] Liang JS, Zhang W. W and Chen Y. Q, "Robustness of boundary control of damped wave equations with large delays at boundary measurement". In: Proceedings of the IFAC World Congress, Prague, Czeck Republic, 2005.

[36] Liang J. S Chen Y. Q and Fullmer R, "Simulation studies on the boundary stabilization and disturbance rejection for fractional diffusion-wave equation", In: Proceedings of the 2004 IEEE American Control Conference, Boston, MA, 2004, pp.5010–5015.

[37] R. Singh, V. M. Mishra and J. singh, "Model order reduction via routh Hurwitz array and improved Pade approximations," 2018 International Conference on Power Energy, Environment and Intelligent Control (PEEIC), Apr. 2018, pp. 755-758, doi: 10.1109 /PEEIC.2018.8665590

[38] Genta G, "Actuators and Sensors", In: Introduction to the Mechanics of Space Robots. Space Technology Library, vol 26. Springer, Dordrecht, 2002, pp.427-478

[39] J Nilson, Real-time Control Systems with Delays, PhD Dissertation, Dept. of Automatic Control, Lund Institute of Technology, Lund, Sweden, 1998.

[40] S. H. Yang and C. Dai, “Multi-rate Control in Internet Based Control Systems”, Proc. of Control 2004, University of Bath, UK, September 2004.

[41] G. P. Liu, J. X. Mu and D. Rees, “Networked Predictive Control of Systems with Random Communication Delay”, Proc. of Control 2004, University of Bath, UK, September, 2004.

[42] M. S. Branicky, S. M. Phillips and W. Zhang, “Scheduling and Feedback Co-Design for Networked Control Systems”, Procs. IEEE Conference on Decision and Control, Las Vegas, December 10-13, 2002.

[43] Anton Cervin, Dan Henriksson, Bo Lincoln, Johan Eker and Karl-Erik Årzén: "How Does Control Timing Affect Performance?" IEEE Control Systems Magazine, June 2003, pp. 16-30.

[44] Anton Cervin, Dan Henriksson, Bo Lincoln and Karl-Erik Årzén: "Jitterbug and TrueTime: Analysis Tools for Real-Time Control Systems." In Proceedings of the 2nd Workshop on Real-Time Tools, Copenhagen, Denmark, August 2002.

[45] H. Ishii and B. A. Francis, Quadratic stabilization of sampled-data systems with quantization, Automatica, 39, 2003, pp.1793-1800.

[46] Cho K, and etc., “Analysis of latency performance of bluetooth low energy (BLE) networks”. Sensors 2015, 15, pp. 59-78.

[47] J. Lofberg, Minimax approaches to robust model predictive control. Linkoping, Sweden: Linkoping Studies in Science and Technology. Dissertations, 2003, pp. 39-62

[48] M. Herceg, M. Kvasnica, C. N. Jones and M. Morari, “Multi-Parametric Toolbox 3.0,” 2013 European Control Conference (ECC), Jul. 2013, pp. 502-510, doi: 10.23919/ECC.2013.6669862

[49] C.D. Aliprantis and K. C. Border, Infinite Dimensional Analysis: A hitchhiker's Guide (3rd edition), Springer Science & Business Media, pp. 8-10

[50] D. Simon, J. Lofberg and T. Glad, "Reference tracking MPC using dynamic terminal set transformation," IEEE Trans. On Automatic Control, vol. 59(10), Mar. 2014, pp. 2790-2795, doi: 10.1109/TAC.2014.2313767

[51] Y. Lan and H. Huang, "Observer-based robust control of time-delay uncertain systems with application to engine idle speed control," International Jounal of Dynamical Systems and Differential Equations, vol. 4(3), 2012, pp. 274-285, doi: 10.1504/IJDSDE.2012.049905

[52] T. Besselmann, J. Lofberg, and M. Morari, "Explicit MPC for LPV Systems: Stability and Optimality," IEEE Trans. on Automatic Control, vol. 57(9), Sep. 2012, pp. 2322-2332, doi: 10.1109/TAC.2012.2187400

[53] F. Borrelli, A. Bemporad and M. Morari, Predictive Control for Linear and Hybrid Systems. London: Cambridge university press, 2015, pp. 181-196

[54] J. Miao, Economic Dynamics in Discrete Time: MIT Press, 2014, pp. 134-143

[55] A. Bemporad, A. Casavola and E. Mosca, "Nonlinear control of constrained linear systems via predictive reference management," IEEE Trans. on Automatic Control, vol. 42(3), Mar. 1997, pp. 340-349, doi: 10.1109/9.557577

[56] Z. Wang, S. Fukushima and T. Hachino, "Model predictive control approach on packet dropout prevention of networked control systems," 2018 5th International Conference on Information Science and Control Engineering (ICISCE), Jul. 2018, pp. 839-843, doi: 10.1109/ICISCE.2018.00176