Robust and Resilient Model Predictive Control for Cyber-Physical Systems Against DoS Attacks

by

Yufan Dai

B.Sc., Wuhan University, 2020

A Thesis Submitted in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Mechanical Engineering

© Yufan Dai, 2023

University of Victoria

All rights reserved. This thesis may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

Robust and Resilient Model Predictive Control for Cyber-Physical Systems Against DoS Attacks

by

Yufan Dai

B.Sc., Wuhan University, 2020

Supervisory Committee

Dr. Yang Shi, Supervisor

(Department of Mechanical Engineering)

Dr. Daniela Constantinescu, Departmental Member

(Department of Mechanical Engineering)

ABSTRACT

With the development of Industrial 4.0, cyber-physical systems (CPSs) have been widely investigated due to their broad applications in a variety of areas. In a CPS, the cyber layer is integrated seamlessly with the physical components through a network-based structure, which dramatically alleviates the physical limitations at a low cost. However, great efficiency also comes with potential threats: The network-based structure is normally fragile and vulnerable to cyber attacks. These attacks can sabotage the elements in the system and tamper with the data, causing severe security problems, especially in the control system since the control system is the core infrastructure in most facilities. In this regard, model predictive control (MPC) stands out to be a promising solution to tackle attacks and ensure performance. Motivated by this fact, in this thesis, we focus on the robust and resilient MPC framework design and application against cyber attacks in CPSs.

In chapter 2, a robust and resilient MPC scheme is proposed and utilized to drive an autonomous underwater vehicle (AUV) to track a predesigned trajectory. Nevertheless, the remote controller-to-actuator channel suffers randomly existing DoS attacks. Thus, a compensation strategy must be developed to mitigate the risk of the AUV going out of control. Thus, the packet transmission strategy is utilized in this work to construct a candidate control sequence at each sampling instant. By updating the sequence in the buffer every time the channel is not suffering attacks, the AUV can at least receive the control input torque to achieve its original control objective. Furthermore, the robustness constraint approach is also introduced in this work to deal with external disturbances. The effectiveness of the proposed method is verified by simulation results and its advantages are reflected through comparison study with the standard MPC approach.

In chapter 3, a novel robust and resilient distributed MPC framework is proposed

for the multi-agent CPS, in which all the communication channels among agents suffer randomly existing DoS attacks. Existing work only focuses on designing control strategies for the channels inside each agent (controller-to-actuator channels and sensor-to-controller channels), but neglects the influence of neighbor agents. To address this issue, a lengthened packet transmission strategy is proposed. By lengthening the predicted state sequence at each sampling instant based on the state-feedback control law, each agent in the CPS is able to receive the necessary information to steer its state to the small region around the equilibrium no matter when the attacks occur. A new type of robustness constraint is also designed to enlarge the region of attraction of this problem. Numerical simulation results for a four-ground-vehicle system are presented to illustrate the advantages of the proposed framework.

In chapter 4, conclusions and potential future work are summarized and presented.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to Dr. Yang Shi, a professional and decent professor, for offering me this great opportunity to be a member of his wonderful group. Throughout my M.A.Sc. program, he generously shared invaluable advice and dedicated a significant amount of time to guiding me through the academic realm. In addition, his enthusiasm for doing research and mental support encouraged me all the time, particularly during the challenging times of the COVID-19 pandemic. Furthermore, he was devoting his heart to students – helped me proofread the report during flights and even revised my paper draft while recovering in the hospital after surgery. Once again, I extend my heartfelt appreciation to Dr. Shi for his thoughtfulness and guidance as my supervisor.

I would also express my appreciation to our ACIPL lab, which contributed a lot to my professional research career and daily life at UVic. Particularly, I would thank Dr. Kunwu Zhang for helping me understand the concept of control theory and revise my manuscript with professional standards. I am also grateful for being the study partner and roommate of Yue Song. The memory of taking the course via Zoom after midnight together will last forever. I would also say thank you to Xiang Sheng for offering me a McDonald's dinner and sending daily necessities to me during quarantine when I first came to Victoria two years ago. Moreover, I would thank Zehua Jia for bringing me basic concepts of control theories with laughter when I first came here. I would also thank Tianxiang Lu for introducing me to the lab two years ago and being my strongest basketball teammate all the time. Furthermore, I would also thank Xinxin Shang for telling me how to become a qualified teaching assistant. In addition, I would thank Dr. Henglai Wei, Dr. Tianyu Tan, Dr. Qian Zhang, Dr. Songlin Zhuang, Dr. Changxin Liu, Dr. Qi Sun, Yijia Xie, Haojiao Liang, Huiting Wang, Lei Xu, and many other students for experiencing all the happiness

and anxieties together.

At last, I must appreciate my family and my girlfriend Binyan Xu. Their unwavering support, attentive listening, and shared experiences have been a constant presence in my life, standing by me through both triumphs and setbacks. Nothing can substitute what they have devoted to me and I will forever cherish the deep emotions they have bestowed upon me.

Acronyms

CPS	cyber-physical system
DoS	denial-of-service
FDI	false data injection
C-A	controller-to-actuator
S-C	sensor-to-controller
MPC	model predictive control
RHC	receding horizon control
AMV	autonomous marine vehicle
AUV	autonomous underwater vehicle
MAS	multi-agent system

Chapter 1

Introduction

1.1 Research Background

1.1.1 Cyber-Physical Systems

Cyber-physical systems (CPSs), known as automated systems that integrate physical layers with cyber infrastructures [2, 26], gain increasing research attention in various areas since the seamless integration of cyber and physical domains has successfully stimulated many exciting applications in the era of Industry 4.0 [16, 94]. CPSs are also characterized by being large-scale, distributed, and heterogeneous interconnected systems that span over various application domains [29]. These systems can be described as intelligent systems that comprise a combination of hardware, software, and computational and physical components, closely interacting to continuously sense and control the changing state of the real world in real-time [16]. CPSs exhibit a level of freedom from the constraints of physical distances, enabling lower installation and maintenance costs in a broad spectrum of application areas [90], ranging from vehicle systems [81], healthcare systems [9] to industrial IoT systems [28, 82], as shown in Figure 1.1.

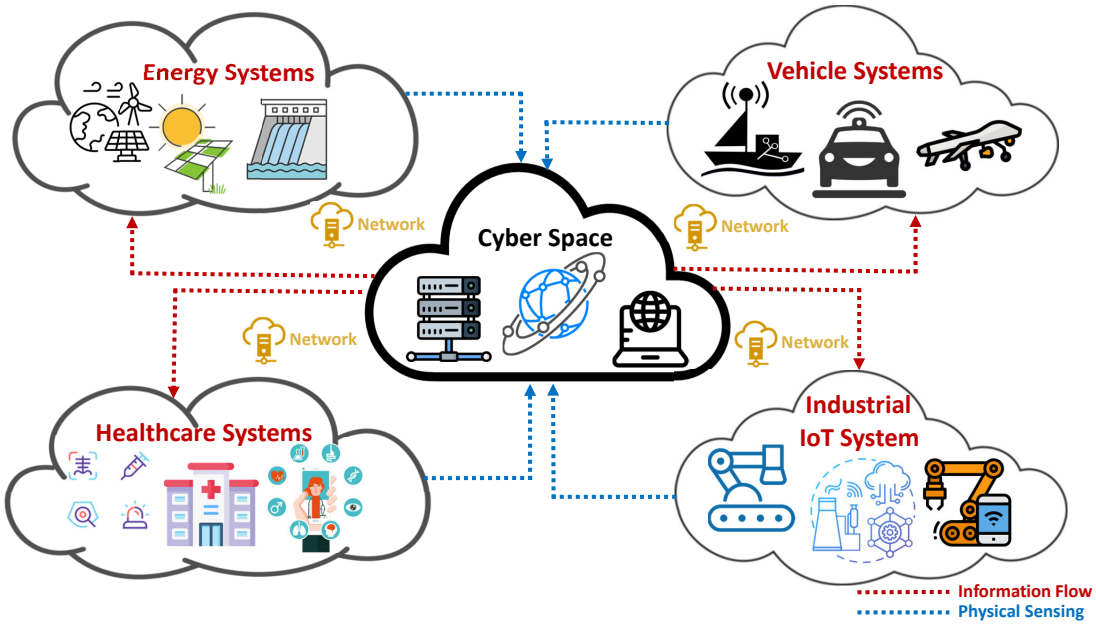

Figure 1.1: Applications of CPSs.

1.1.2 Cyber Attacks

The convenience in the interconnection among the subsystems in the CPSs is mainly owing to its network-based architecture. However, great efficiency with the advent of the network in control systems also introduces high risks of suffering cyber threats [43]. Due to the fact that the networked structure is highly exposed to adversarial hackers, the information transmitted in the networked channels is easy to be stolen or tampered with. In other words, attackers may gain access to launch a variety of cyber attacks on the overall systems. Cyber attacks, such as denial-of-service (DoS) attacks [1, 8], false data injection (FDI) attacks [35, 51], replay attacks [45], etc., aim at jamming the communication channels, stealing the data, or tampering with the signal, causing severe security threats or even potential damage to the society. It is worth mentioning that the security problems in the control systems are more severe since the control

systems are the core infrastructures in most facilities [57].

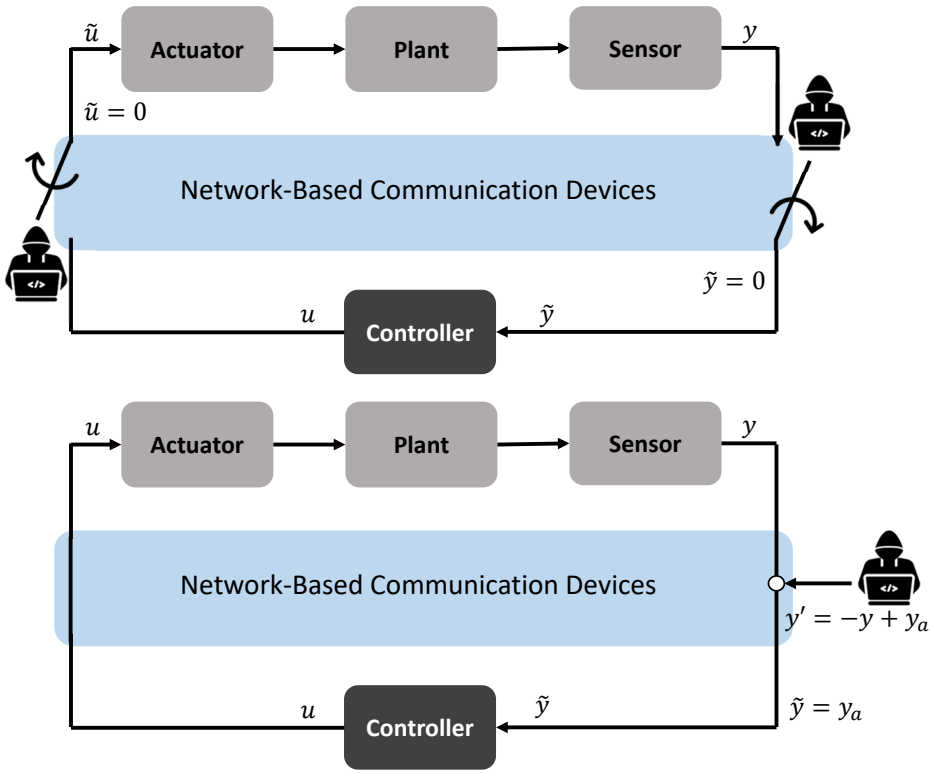

Figure 1.2: Illustrations for DoS attacks (upper) and FDI (lower).

Figure 1.2, for example, illustrates the typical launching strategy of DoS attacks and FDI attacks in a control system. Here, u and y represent the control input generated by the controller and the output collected by the sensor and $\tilde{u}$ and $\tilde{y}$ denote the corresponding signal after suffering the attacks, respectively. Note that DoS attacks usually launch on the controller-to-actuator (C-A) channel and sensor-to-controller (S-C) channel, jamming these channels to compromise devices and prevent the controller and sensor data from reaching their respective destinations [1]. While FDI attacks usually refer to the action that the attacker injects malicious measurement to deceive the state estimation process [84]. In addition, replay attacks can also affect the security of CPS by replacing the measurements generated through the sensor

or the control inputs coming from the controller with the previously recorded data. Other attacks such as zero dynamics attacks and convert attacks, among others, have not received extensive investigation in the field of control systems. Therefore, they will not be discussed in this thesis. However, all kinds of attacks can corrupt the data transmission process, and deter CPSs from achieving their control objectives.

1.1.3 Attack-Defense Strategy

Consequently, with the wide spread of CPSs, the vulnerability toward cyber attacks is still a challenge and, thus, an emerging research issue [13]. When these attacks are launched, the attackers are able to tamper with or block the data transmitted from one component to another to prevent the overall system from achieving its control objectives. In this regard, the need for designing a security control framework to mitigate the jeopardization caused by cyber attacks is of incremental urgency and paramount importance.

However, the challenges of establishing control frameworks to address cyber attacks persist. In order to completely safeguard the control system against the impact of such attacks, extensive research is being conducted on a comprehensive security service framework, which can be categorized into the following three aspects (illustrated in Figure 1.3):

(1) **Confidentiality**: To gain access to the network-based communication channels or construct deceptive data, adversarial attackers typically require specific information regarding the configuration and detailed transmission data of the control system. Consequently, enhancing the confidentiality of the system configuration and the privacy of data transmission becomes the primary focus at the first level of protecting the control system from cyber attacks. Therefore, the key question that arises is how to design an effective strategy to improve

confidentiality.

(2) **Resilient Control - Detection**: However, relying solely on a passive control strategy to mitigate their effects may not be sufficient, since attackers can decipher the encryption methods or implement some model-free attacks such as DoS attacks or replay attacks to the control systems. In this case, the concept of resilient control arises. Categorized by detection and mitigation processes, resilient controllers can alleviate the effect caused by cyber threats. In normal cases, the controller must be aware of the existence of the cyber attacks and utilize corresponding methods to mitigate the effect. Motivated by this, some researchers are focusing on proposing detection schemes to detect the attacks in real-time or determine what type of cyber attack it is. Hence, in this level, the key question is shown to be how to determine the baseline to be compared with the infected data to detect cyber attacks.

(3) **Resilient Control - Mitigation**: Closely following the previous process, when the attacks are detected in the networked channels (or some attacks may not interfere with the performance much, leading to no specific need for detection), the next step is to utilize the corresponding methods to correct the false data or generate a candidate data sequence to compensate for the lack of data, which results in enhancing the resilience of the overall CPS. Hence, the question of how to construct the corresponding candidate data for mitigating the effect induced by cyber attacks becomes crucial.

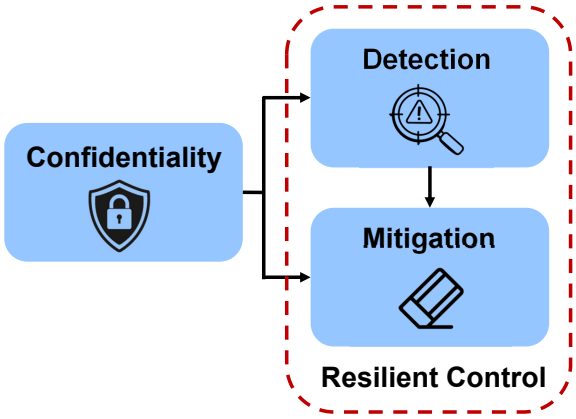

Figure 1.3: Three levels of security service.

1.2 Model Predictive Control

In this section, an overview of Model predictive control (MPC) and tightening constraint MPC is presented. Firstly, we introduce MPC with its strengths and detailed formulation and then introduce a tightening constraint approach to enhance its robustness. The following notations will be used in this section.

1.2.1 Overview of MPC

MPC, also known as receding horizon control (RHC), is an advanced optimal control strategy that integrates the feedback mechanism with the convex optimization problem. At each sampling instant, the controller will apply the mathematical model of the system to predict its future behavior, and then use this prediction to optimize the control sequence within the pre-designed prediction horizon as shown in Figure 1.4. This optimization problem is repeatedly solved under the inherent physical constraints based on the real system at each time instant. Owing to the excellent performance with solvable constraints, MPC has already been applied to various areas [17, 55, 72].

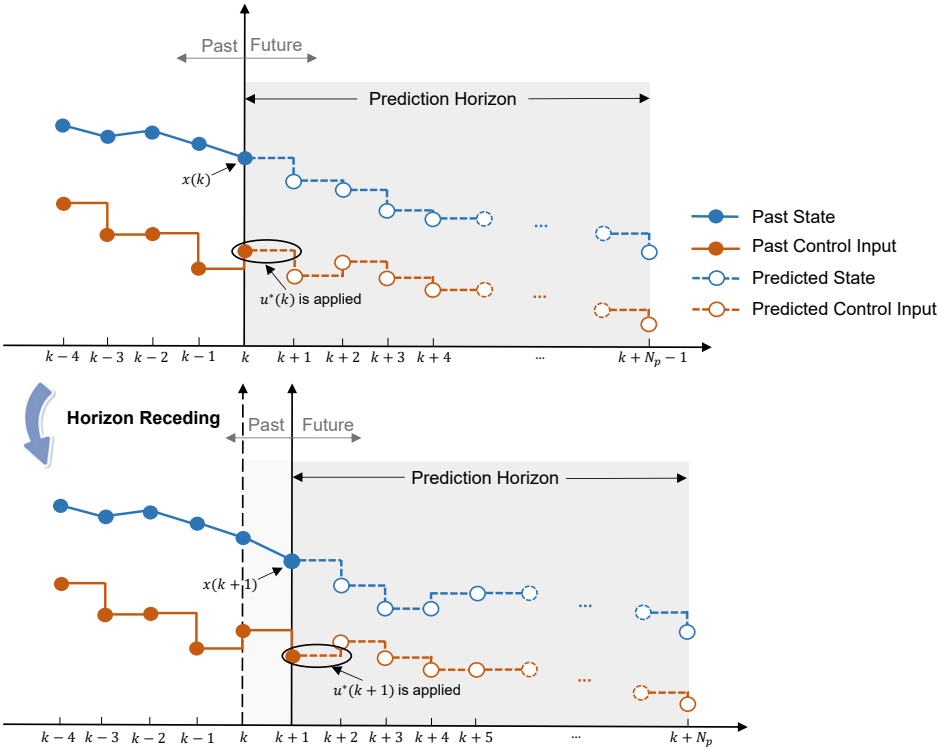

Figure 1.4: Illustration for MPC.

1.2.2 MPC Formulation

Before presenting the MPC formulation, firstly, we need to consider a class of systems to be controlled, which can be represented by the following original differential equation:

$$\dot{x}(t) = f\left(x(t),\, u(t)\right), \tag{1.1}$$

where $x(t) \in \mathbb{R}^{n_x}$ and $u(t) \in \mathbb{R}^{n_u}$ represent the state and control input vectors that are restricted in convex sets including the origin where the symbol $\mathbb{R}^n$ denotes the n-dimensional real space, which indicates $x(t) \in \mathbb{X}$ and $u(t) \in \mathbb{U}$. $f(\cdot) : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is a set of functions (containing both linear and nonlinear functions) with $\mathbf{0}_{n_x}$ being their equilibrium point, such that $f(\mathbf{0}_{n_x}, \mathbf{0}_{n_u}) = \mathbf{0}_{n_x}$. Note that the system

investigated here can be linearized at the origin, and the linearized system must be controllable.

In recent years, for the purpose of guaranteeing closed-loop stability with a finite prediction horizon, one classical method is to add a terminal constraint in the MPC algorithm at each time instant [12]. In this method, the infinite prediction horizon is divided into two parts–one part predicts the system behavior with predicted control inputs within finite simulation steps, while another part is under the control of a stabilizing feedback law, representing the subsequent infinite horizon. An additional constraint, named terminal constraint, is imposed to ensure that this control law meets the system constraints with guaranteed feasibility and feasibility analysis. The terminal constraint is chosen to steer the predicted state at the end of the prediction horizon into an invariant set, in which the state will never leave once enters. To be more specific, the MPC problem to be solved at the sampling instant k is formulated as

$$
\begin{aligned}
&\min_{\boldsymbol{u}^*(t)} \quad \{J\left(t, x(s|t), u(s|t)\right)\} \\
&\text{s.t.} \quad\; x(t|t) = x(t) && \text{(1.2a)} \\
&\qquad\; \dot{x}(s|t) = f\left(x(s|t), u(s|t)\right), && \text{(1.2b)} \\
&\qquad\; u(s|t) \in \mathbb{U}, && \text{(1.2c)} \\
&\qquad\; x(s|t) \in \mathbb{X}, && \text{(1.2d)} \\
&\qquad\; x(t+T|t) \in \mathbb{X}_f, && \text{(1.2e)} \\
&\qquad\; s \in [t, t+T]
\end{aligned}
$$

where $J\big(t, x(s|t), u(s|t)\big)$ is the cost function which usually has the form of

$$
J = \int_t^{t+T} \left[\|x(s)\|_Q^2 + u(s)\|_R^2\right] \mathrm{d}s + \|x(T)\|_P^2
$$

, (Q, R, and P are positive-definite symmetric weighting matrices with proper dimensions), where $\|\cdot\|_P$ is the P-weighted norm; $x(s|t)$ and $u(s|t)$ represent the predicted state and control inputs generated at time $s \in [t, t+T]$, where T is the prediction horizon; Eq. (1.2b) represents the predicted system evolution; $\mathbb{X}_f$ denotes the terminal region;

Here, the last term in the cost function $\|x(t+T|t)\|_P^2$ is called the terminal cost, and the corresponding terminal region can be chosen as $\mathbb{X}_f = \{x | \|x(t+T|t)\|_P^2 \leq \epsilon^2\}$. According to [12], the selection of the terminal penalty matrix P should be the solution to the Lyapunov equation

$$(A + BK + \kappa I)^{\mathrm{T}} P + P(A + BK + \kappa I) = -(Q + K^{\mathrm{T}} R K) \tag{1.3}$$

where $A = \frac{\partial f}{\partial \boldsymbol{x}}\big|_{(\mathbf{0}_{n_x}, \mathbf{0}_{n_u})}$ and $B = \frac{\partial f}{\partial \boldsymbol{u}}\big|_{(\mathbf{0}_{n_x}, \mathbf{0}_{n_u})}$ are the linearized system matrix; K is a linear stabilizing control gain; $\kappa \in \big(0, -\min\{(\lambda(A+BK)\}\big)$ is a constant; With these parameters chosen as required, the terminal region of the linearized system should satisfy two properties as follows:

- For $x(t) \in \mathbb{X}_f$, the state feedback control input always satisfies $u = Kx \in \mathbb{U}$.
- When $x(t) \in \mathbb{X}_f$, the optimal cost function of the optimization problem is decreasing step by step, which ensures that the origin of the state space is an asymptotically stable equilibrium point.

Recalling Figure 1.4 that at time t, the MPC algorithm utilizes the current state information $x(t)$ to solve the optimization problem under the constraints, generating the optimal control input sequence $\boldsymbol{u}^*(t)$ with the predicted state sequence $\boldsymbol{x}(t)$. After generating the predicted sequences, we apply the first element in the optimal control input sequence to the real system to guarantee performance with the closed-loop mechanism. Hence, by applying the MPC algorithm, the controller can predict

and optimize the system behavior at each sampling instant, improving the system performance with guaranteed stability analysis.

1.2.3 MPC with Robustness Constraint

However, in most practical cases, disturbances are ubiquitously existing in all kinds of systems. Although MPC has some intrinsic robustness due to its receding horizon nature [92], researchers are dedicated to studying a new type of MPC, robust MPC, to enhance and prove its robustness. Existing methods including tube-based MPC [41,42], min-max MPC [4], robustness constraint MPC [32,33], etc. Robustness constraint MPC, therein, aims at adding a new shrinking state constraint to the MPC algorithm at each time instant to ensure that the algorithm is still feasible and the closed-loop system is stable under the existence of disturbances. Compared to other robust MPC algorithms, this method only needs to solve the MPC algorithm with an additional constraint, in which all required parameters can be calculated offline, relatively requiring lesser computational burdens, which in this regard, is more suitable for the network-based control system.

Consider a nonlinear system that is disturbed by an additive uncertainty.

$$\dot{x}(t) = f\left(x(t),\, u(t)\right) + w(t), \tag{1.4}$$

where $w(t) \in \mathbb{R}^{n_x}$ denotes the vector of uncertainties and disturbances. In addition, $w(t)$ is assumed to be bounded as $w(t) \in \mathbb{W}$.

As the disturbances are unpredictable, it is not practical to directly apply the real system model in the MPC algorithm. Instead, in most of the robust MPC designs, the nominal system is applied to represent and predict the behavior of the real system,

which is often designed as:

$$\dot{\widehat{x}}(t) = f\left(\widehat{x}(t),\, u(t)\right), \tag{1.5}$$

where $\widehat{x}(t) \in \mathbb{R}^{n_x}$ represents the state of the nominal system. With the nominal system model (1.5), referring to [32, 33], the optimization problem in this robust MPC can be designed as

$$\begin{aligned}
&\min_{\boldsymbol{u}^*(t)} \ \left\{J\left(t, \widehat{x}(s|t), u(s|t)\right)\right\} \\
\text{s.t.} \quad & \widehat{x}(t|t) = x(t) && \text{(1.6a)} \\
& \dot{\widehat{x}}(s|t) = f\left(\widehat{x}(s|t), u(s|t)\right), && \text{(1.6b)} \\
& u(s|t) \in \mathbb{U}, && \text{(1.6c)} \\
& \|\widehat{x}(s|t)\|_P \leq \frac{T\alpha}{s-t}\epsilon, && \text{(1.6d)} \\
& s \in [t, t+T]
\end{aligned}$$

Here, (1.6d) is the robustness constraint, where ϵ is the terminal region level defined as (1.2e) and α is the designed parameter determining the shrinking rate of the constraint with the time instant increasing. By confining the nominal state in the shrinking region, under reasonable assumptions, the state of the real system defined in (1.4) can prove to be regulated in a small region containing the equilibrium and satisfies the original terminal constraint (1.2e).

Indeed, the tightening constraint method in [32,33] enhances the robustness in the MPC algorithm but has a relatively small region of attraction since the constraint (1.6d) is designed to be linearly related to the terminal constraint. Motivated by this, an integrated type of robustness constraint is proposed in [65], which replaces

the constraint (1.6d) by

$$\|\widehat{x}(s|t)\|_P \leq \frac{(t+T-s)M+s-t}{T}\alpha\epsilon,\ s \in [t, t+T],, \tag{1.7}$$

where M is the designed contraction rate. Due to the fact that the parameter α is designed similarly to (1.6d) and $M > 1$ only has a minimum value in proof, the original robustness constraint set at each time instant can be seen as the subset of the integral type of robustness constraint. Since then, the region of attraction of the robust MPC can be seen as enlarged.

However, the integral type of robustness constraint is designed only based on the terminal constraint. When considering the state constraint, the right side of (1.7) needs to be proven to be lesser than the maximum value of $\|\widehat{x}(t+T)\|_P$, where $\widehat{x}(t) \in \mathbb{X}$, resulting in these types of robustness constraints hard-to-design and a relatively conservative region of attraction. In this regard, a new type of robustness constraint that is not only based on the terminal region can be investigated.

1.3 MPC-Based Cyber-Security Service

Based on the aforementioned discussions, MPC is often utilized due to its ability to handle physical constraints while ensuring optimal performance with respect to the preassigned indexes at the same time [63]. However, in addition to the ubiquitous physical constraints and external disturbances, as we mentioned earlier, cyber threats also cause trouble in CPSs. To fully utilize the predicted behavior against cyber threats with the ability to handle physical constraints and external disturbances, in recent years, various MPC-based secure control schemes have been done in recent years. Based on the discussions in 1.1.3, the MPC-based secure control schemes can also be classified as these processes.

1.3.1 Confidentiality

Firstly, the most intuitive way to defend cyber attacks is to enhance the confidentiality of the overall CPS, preventing the attackers from gaining access to know the structure and the data in advance. To protect the data from stolen in network-based communication channels, the encoding and decoding process, therein, offers a promising solution. An input quantization method is utilized in the output feedback MPC to quantize the input signal from the controller to the actuator [56], in which the quantizers serve as the encoder, improving the information confidentiality in CPS to some extent. However, the security perspective of the quantization approach is limited. In [64], an encoding and decoding method is utilized in the S-C channel, ensuring the data in this channel is difficult to decipher. Moreover, in [70], the encryption and decryption technique is extended to both C-A and S-C channels, further improving confidentiality. In addition to this, Sun and Shi proposed a cloud-edge framework [67], fully utilizing cloud-based computing with elliptic curve cryptography-based encryption, enhancing the data transmission security for both C-A and S-C channels with reduced computation load. Similar to the idea of the encoding technique, the blockchain technique is also investigated to ensure the safety of data exchanging the MPC [5]. Another perspective for enhancing the confidentiality of the CPS is to design parallel controllers. In [19], several auxiliary control laws of the Lyapunov-based MPC framework have been designed in parallel. By randomly selecting the auxiliary control law, the attack that launches at a specific controller may not be in effect.

Table 1.1: Related work of enhancing confidentiality in MPC

	Related Work
Quantization Methods	[56] [91] [61]
Encryption Methods	[64] [70] [67]
Blockchain Technique	[5]
Auxiliary Controller	[19]

1.3.2 Resilient MPC

When the first process in the security control framework, confidentiality, is missing or broken, the attacks can be purposely launched into the control system. To address this issue, a resilient MPC scheme should be utilized. As we mentioned in 1.1.3, the resilient MPC framework can also be divided into two processes, detection and mitigation, which will be illustrated as follows:

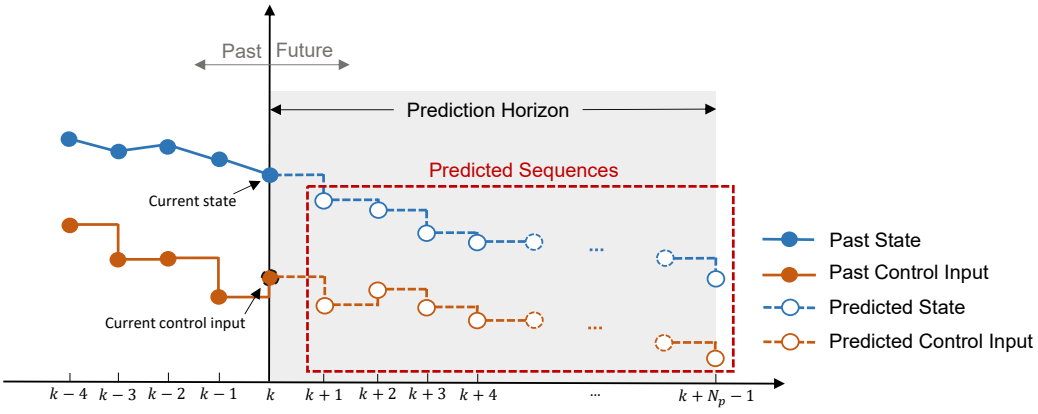

Figure 1.5: Predicted sequences in MPC.

-Detection

To initialize the resilient control framework, we need to consider how to detect and handle the data losses caused by cyber attacks. In general, there are two main themes of detection algorithms, which are the data-driven detection algorithm and the model-based detection algorithm [47]. Data-driven detection algorithms usually train the data with all potential attacks and collect a data set that contains all the possible information when attacks happen. After that, the attacks can be detected every time when the actual data matches the collected data set. Model-based detection techniques predict the system behavior and construct a set or triggering level based on the system model. When the attacks occur, the actual information will deviate much from the predicted ones, and then the attacks can be detected. In this regard, MPC has its intrinsic advantage. An explanatory illustration is shown in Figure 1.5. At each sampling instant, the controller generates not only the control input sequence including the current control input and the predicted control inputs but also the predicted state sequence, by performing the online optimization based on the plant model. Both predicted sequences can be deliberately utilized to play an instrumental role in proposing the resilient MPC strategy. Specifically, they can be effectively adopted in contrast to the false data injected by FDI or replay attacks.

Thereby, the predicted sequence generated at each time instant can be utilized in the model-based detection mechanism and therefore this mechanism is widely applied in the MPC approach. This kind of detector is integrated into the MPC framework in [11], but lacks comprehensive theoretical analysis. In [22], a model-based detector is set up to both tackle the replay attacks and the FDI attack. A predicted sequence-based set membership test is proposed to detect whether the attacks are underway or not. A similar approach is also shown in [20], with both communication channels affected, resulting in more scenarios to be considered. In [80], the model-

based predicted information is used to construct a triggering level. By comparing the measured information with the triggering level, cyber threats can be found. Another model-based detection mechanism is to calculate the value of the Lyapunov function of the measured information, comparing it with the predicted ones, and then detect the attacks, which is utilized in the Lyapunov-based MPC proposed in [49].As for the data-driven detection mechanisms, in [71] data-driven detection scheme is utilized in the MPC framework. By training the system for all the potential deceptive information, the long short-term memory fuzzy neural network is able to detect all the anomaly human-like data injection. All related results on the detection mechanisms are summarized in Table 1.2.

Note that in some work, the detection approaches are not discussed. The reasons are mainly threefold: (1) When considering tackling DoS attacks, the attacks can be detected every time if the receiver does not obtain any information at that time instant. Hence, it is unnecessary to design a specific detection mechanism for DoS attacks. (2) For other attacks, in some research, an assumption has been made that the attacks can be detected once occur. (3) In some resilient MPC approaches, e.g. [77], the resilient controller is designed to able to tackle the worst case induced by cyber attacks with known bounded attack duration. Consequently, the attacks can be tackled since their effect is less or equal to the worst case.

Table 1.2: Related work of the detection mechanisms in MPC

Detection Mechanisms		Related Work
Model-Based Detection	Direct Comparison	[11], [36]
	Set-Based Comparison	[22], [20], [39], [73]
	Triggering Level Construction	[80], [95]
	Lyapunov Based Comparison	[49]
Data-Driven Detection		[71]

-Mitigation

After detecting the attacks, the next step is to design a resilient control strategy to mitigate the effect induced by attacks. Some resilient MPC strategies have recently been emerging and reported in the literature [13]. Benefiting from the predicted sequence generated by MPC at each sampling instant, the first type of resilient MPC results aims to adopt the predicted sequence to compensate for or correct the interfered data. In [68], Qi Sun et. al. proposes a packet transmission strategy to fully utilize the predicted control input sequence, compensating for the lack of information caused by DoS attacks in the C-A channel for linear CPSs; Similarly, in [69], a buffer is designed to store the model-based predicted sequences to handle DoS attacks occurring on the C-A channel with the event-triggered mechanism to save the computational load. Furthermore, in [66], the packet transmission strategy is extended to be utilized in the nonlinear system, in which the external disturbances are tackled through robustness constraint MPC, and the event-triggered mechanism is also employed to reduce the computational and communication load. In [24], a self-triggered resilient MPC framework is proposed to tackle the FDI attacks on the C-A channel by generating control sample packages and reconstructing the data in a CPS. In [21],

Giuseppe Franze et. al. proposed a resilient control framework for a CPS to detect and mitigate the effect induced by the replay attacks on the S-C channel; an auxiliary control input sequence is constructed to substitute the false input. To move on, in [22], in addition to the controller buffer, a delay-MPC scheme is also proposed to tackle the replay attacks on the S-C channel, and an actuator buffer is designed for the FDI attacks on the C-A channel. Furthermore, in [20], both the actuator buffer and controller buffer are designed to store the predicted system behavior so as to mitigate the effect caused by the FDI attacks on both command and measurement channels. Another type of resilient MPC framework is to set up additional constraints on the duration of attacks or the control inputs and then enable the CPS to tolerate such attacks in the worst case (with the longest attack duration or maximum cost function). In [31], both DoS and FDI attacks are considered to interfere with the C-A channels in CPSs with limited energy upper bound. An additional input constraint based on the H_2 and H_∞ performances of this system is added to this system to help the control framework tolerate both attacks in the worst case. In spite of these attacks, the framework proposed in this work is able to regulate the states while reducing the computational load by using a dynamic event-triggered mechanism. A summary regarding resilient MPC that tackles different types of cyber attacks on their distinguished affected channels is listed in Table 1.3.

Table 1.3: Resilient MPC tackles different attacks on affected channels

Attack Type\Affected Channel	S-C Channel	C-A Channel	Both Channels
DoS Attack	N/A	[68], [31], [69] [66]	[36] [52]
FDI Attack	[71], [49], [80] [39], [77], [76] [78]	[22], [31], [24]	[36], [20], [11]
Replay Attack	[22], [21]	N/A	[95]

Based on the discussions above, the attacks' effect mitigation can be divided into two categories, model-based predicted sequence compensation and worst-case cyber attack tolerance. Most MPC-based resilient control strategies underscore the advantage of the predicted system behavior and fully utilize their predicted sequence to substitute the interfered information. While the worst-case attack tolerance methods usually assume tight constraints on attack duration, which leads to relatively more conservative results. The detailed mitigation classifications can be summarized and shown in Table 1.4.

Table 1.4: Classification of attack mitigation

	Model-Based Compensation	Worst-Case Tolerance
Linear System	[68], [69], [21], [22], [77], [95] [20], [52], [36], [39], [76]	[31], [78]
Nonlinear System	[66], [24], [80], [49], [11]	N/A

It can also be concluded in Table 1.4 that most current results mainly focus on cyber security issues in linear systems, which may be due to the fact that the behaviors of nonlinear systems are relatively hard to predict and optimize. Thus, more mature theories or applications should be proposed to address cyber security issues in more general scenarios.

Additionally, there is another type of attack, direct attack, that cannot be handled by the above listed-MPC methodologies. In this attack setting, adversarial attackers can purposely hack the control system and isolate one unit from it. For instance, in [15], the power system is designed to be connected with an analog proportional integral controller in parallel with a converter and digital MPC. When the converter is hijacked, the digital MPC is of no use, but the candidate proportional-integral controller can still achieve the control objective. However, such attacks are not common in CPSs, so they will not be involved in this thesis.

1.4 Research Objectives and Proposed Methodologies

Based on the aforementioned discussions for the existing cyber attack-defense strategy in CPSs, the objectives of this M.A.Sc. thesis are to

- Deal with the packet losses in different channels induced by randomly existing DoS attacks in both single-agent and multi-agent CPSs.
- Tackle the external disturbances in the nonlinear CPSs.
- Demonstrate the effectiveness of the proposed algorithms in real-world application scenarios.

To achieve these objectives, we formulate the MPC-based resilient methodologies, which can be described as follows:

- To mitigate the effect caused by DoS attacks, the lengthened sequence transmission strategy is proposed to store the state sequences generated by the MPC algorithm and lengthened by a stabilizing control law. In this way, the lack of state information can be compensated by the lengthened predicted sequences.
- Robustness constraint MPC is utilized and improved in this work. The robustness of the MPC algorithm is proven to tolerate disturbances with a comparable performance by adding a robustness constraint. Furthermore, the robustness constraint is designed by utilizing both state constraint and terminal constraint to enlarge the region of attraction.
- The robust and resilient MPC strategy is utilized in a cyber-physical autonomous unmanned vehicle (AUV) system and a multi-agent CPS, while DoS attacks occur on the C-A channels and the communication channels among the agents, respectively. The performance and analysis of both scenarios are guaranteed.

1.5 Thesis Organization

To demonstrate the effectiveness of the robust and resilient MPC framework in both theory and application, the remainder of this thesis is organized as follows:

Chapter 2 involves a robust and resilient MPC scheme to tackle DoS attacks in a typical CPS, remote AUV trajectory tracking control, in which the communication channel between the cyber layer (remote controller) to the physical component (autonomous underwater vehicle) suffers randomly existing DoS attacks.

Chapter 3 proposes a novel robust and resilient distributed MPC scheme to solve the cooperation regulation problem for the multi-agent CPSs. Different from

the attacks in single-agent systems, DoS attacks in this case occur on all the communication channels among the vehicles. The theoretical analysis is involved and numerical simulation results for a group of ground vehicles are given to show the effectiveness.

Chapter 4 gives the conclusions of this thesis and the potential future research areas.

Chapter 2

Robust and Resilient MPC for AUV Trajectory Tracking Control Against DoS Attacks

2.1 Overview

CPS has a wide range of applications owing to its seamless integration of cyber and physical domains. This feature greatly enables the tightly-coupled locomotion, computational and communication components, enabling potential advantages in a cyber-physical vehicle system [6]. One area that can greatly benefit from these advantages is the control of autonomous marine vehicles (AMVs). By establishing close communication between the remote controller (cyber layers) and the vessel's body (physical domain), AMVs can be controlled in real-time under specific conditions to efficiently accomplish designated tasks [25, 48]. This chapter primarily focuses on a typical CPS control application: remote AMV control with a special emphasis on addressing security concerns.

The pursuit of exploring the ocean has never ceased for humanity. In the past thousands of years, humans have been driven by an insatiable curiosity to uncover the mysteries of the ocean. From ancient seafarers embarking on perilous voyages using a canoe to nowadays advancements in maritime technology, our quest to understand and explore the oceanic realm has only grown stronger.

In recent years, AMV has been widely investigated since it is a powerful tool for exploring the ocean more efficiently, further broadening our understanding and research of the ocean. Its wide applications including bathymetric [23, 30], environmental mapping [38], rescuing [46], etc. On the other hand, broad applications of AMV also stimulate a wide range of control problems, such as dynamic positioning control, path following, trajectory tracking, and cooperative control. Among the diverse control problems, trajectory tracking, enforcing the vehicle to track a temporal and spatial trajectory, is one of the most fundamental tasks. Plenty of research interests have been gathered in this topic due to its development prospects [58,60,83].

Furthermore, modern AMVs are frequently assigned tasks in offshore environments to achieve further goals such as investigating the ocean. Due to the computational limitations of the onboard embedded computers, AMVs are usually considered to be controlled by remote controllers (e.g. a ground station or marine surface vehicles) to enhance real-time tracking accuracy and reliability. In these scenarios, the communication channel between the AMV and the remote controller becomes crucial, as the AMV would be considered completely out of control or even lost without a functioning communication channel. With the emergence of network technologies, remote control of AMVs is typically achieved through network-based architectures, enabling real-time information exchange between the AMV and the remote controller.

However, with great efficiency also come potential drawbacks. The network-based channel connecting the AMV and the remote controller is fragile and susceptible

to various cyber attacks. Adversarial attackers can exploit these channels to steal information or launch cyber attacks, hindering AMV from accomplishing its intended objectives. In the worst-case scenario, these attacks can result in significant damage to the AMV. Among the different types of cyber attacks, the DoS attack stands out as an easily executable method for attackers. Its simplicity lies in the fact that it does not require comprehensive knowledge of the entire system, making it a common threat in our daily lives. To tackle this issue, some research has already been done to tackle the attacks when controlling an AMV. In [40], Yong Ma et. al. proposed a full-state regulation control algorithm for the linearized marine vehicle model stabilize the marine vehicles against DoS attacks on the C-A channel. Zehua Ye et. al. proposed a semi-Markovian jumping system approach to deal with the DoS attacks occurring on the C-A channel of the marine vehicle control for a state-stabilizing control [89]. In [93], the attack-resilient property is extended to defense against the attacks that occur on both C-A and S-C channels: An intelligent event-based resilient control strategy using Takagi–Sugeno (T-S) fuzzy switched approach is proposed to regulate the state of an AMV. Furthermore, in [87], this vehicle system under the proposed framework is proven to be finite-time stable. In [86], in addition to the intelligent event-based T-S fuzzy switched system approach, the quantized sliding mode control strategy is also involved to mitigate the effect caused by the attacks on the S-C and C-A channels. The attack-resilient approach is also investigated for tracking control. In [88], an adaptive event-based control strategy is proposed to control the marine vehicle to track specific reference signals. However, only the yaw angle is able to be tracked in this work. Although some resilient trajectory-tracking control approaches have already been discussed, the investigation of the trajectory-tracking control for an actual nonlinear marine vehicle against DoS attacks still remains blank. In addition to this, to control a marine vehicle in real-life, the constraints need to be considered as

well. None of the research above can tackle the physical constraints that widely exist in all marine vehicles. The question remains unsolved: How to consider the constraint while also fulfilling the tracking objective? How to compensate for the effect caused by the DoS attacks in a nonlinear marine vehicle model? As we mentioned in chapter 1, there exists a powerful control scheme MPC that is able to optimize the control performance while also considering the constraints on both states and inputs. In addition to this, MPC also has its intrinsic advantage of tackling DoS attacks since the predicted sequences that are generated at each sampling instant can be utilized to compensate for the lack of information.

In this chapter, we develop a robust and resilient MPC framework for an extensively studied type of AMV - autonomous underwater vehicle (AUV) to track a reference trajectory against external disturbances and DoS attacks that randomly exist on the C-A channel. The main challenges in designing the control strategy are mainly two-fold:

- An actual nonlinear AUV model is utilized in this chapter to improve tracking accuracy. Unlike the linearized system in [40, 88], the action of the nonlinear system is harder to optimize, which makes the trajectory tracking control difficult to design. To track the predesigned trajectory in real time, a precise error dynamics model is needed.

- For the purpose of tracking the reference signal under the disturbances and DoS attacks, a compensation method should be investigated to mitigate the effect caused by the DoS attacks. Meanwhile, a robust control strategy is needed to tackle external disturbances.

The remainder of this chapter is organized as follows. Section 2.2 mainly introduces the modeling of AWV, derivation of the error dynamics, and the formation of

the trajectory tracking problem. Section 2.3 develops the robust and resilient MPC framework via packet transmission strategy and robustness constraint. In Section 2.4, the simulation results including the comparison study between the proposed method and the standard MPC approach are involved, demonstrating the effectiveness of the proposed design. Finally, the conclusion is summarized in Section 2.5.

Notations

The notations used in this chapter are fairly standard. The symbol $\mathbb{R}^n$ denotes the n-dimensional real space. Given a matrix P, $P \succ 0$ and $P \succeq 0$ denote that matrix P is positive definite and positive semidefinite, respectively. For a vector $x \in \mathbb{R}^{n\times 1}$, $\|x\|$ denotes the Euclidean norm and $\|x\|_P$ denotes the P weighted Euclidean norm as $\sqrt{x^{\mathrm{T}}Px}$, where the matrix $P \succ 0$.

2.2 Preliminaries and Problem Statement

2.2.1 AUV Modeling

In this chapter, we consider the motion of a Saab SeaEye Falcon open-frame AUV (shown in Figure 2.1) in the local level plane.

Figure 2.1: Saab SeaEye Falcon open-frame AUV [54].

The three-degree-of-freedom kinematic equations can be expressed as follows:

$$\begin{aligned}\dot{\eta} &= R(\psi) \\ &= \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} \\ &\triangleq J(\eta)\mathrm{v},\end{aligned} \tag{2.1}$$

where $\eta = [x,\, y,\, \psi]^{\mathrm{T}}$ denotes the position and orientation vector represented in the earth-fixed reference frame; $\mathrm{v} = [u,\, v,\, r]^{\mathrm{T}}$ denotes the velocity vector containing the surge, sway, and yaw velocities represented in the body reference frame. For the purpose of simplifying the model, we need to propose the following assumption:

Assumption 1. *Following [79], for the AUV investigated in this chapter with low-speed motion, we simply assume that*

- *The AUV is symmetric at three planes.*
- *The mass distribution of the AUV is homogeneous.*

- *The pitch and roll motions of the AUV are neglected.*

With the assumption above, the dynamics of the AUV can be expressed as:

$$M\dot{\mathrm{v}} + C(\mathrm{v})\mathrm{v} + D(\mathrm{v})\mathrm{v} + g(\eta) = F + w. \tag{2.2}$$

where M is the inertia and $C(\mathrm{v})$ represents the state-dependent matrix of Coriolis and centripetal terms; $D(\mathrm{v})$ stands for the hydrodynamic damping and lift matrix; $g(\eta)$ denotes the vector of gravitational forces and moments; $w = [w_x,\, w_y,\, w_\psi]^{\mathrm{T}}$ is the additive disturbances. To be more specific, the matrices in the AUV dynamics are formulated as:

$$M = \begin{bmatrix} M_{\dot{u}} & 0 & 0 \\ 0 & M_{\dot{v}} & 0 \\ 0 & 0 & M_{\dot{r}} \end{bmatrix}, \tag{2.3a}$$

$$C(\mathbf{v}) = \begin{bmatrix} 0 & 0 & -M_{\dot{v}}v \\ 0 & 0 & M_{\dot{u}}u \\ M_{\dot{v}}v & -M_{\dot{u}}u & 0 \end{bmatrix}, \tag{2.3b}$$

$$D(\mathbf{v}) = \begin{bmatrix} X_u + D_u|u| & 0 & 0 \\ 0 & Y_v + D_v|v| & 0 \\ 0 & 0 & N_r + D_r|r| \end{bmatrix}, \tag{2.3c}$$

where $M_{\dot{u}} = m - X_{\dot{u}}$, $M_{\dot{v}} = m - X_{\dot{v}}$, and $M_{\dot{r}} = I_z - N_{\dot{r}}$ are the inertia coefficients including add mass; X_u, Y_v, and N_r are linear drag coefficients; D_u, D_v and D_r are corresponding quadratic drag coefficients;

Assumption 2. *Due to the fact that the disturbances on the yaw velocity may cause a tremendous effect on trajectory tracking control, we simply assume that the disturbance on $\dot{r}$ is too small to count, i.e. $w_\psi = 0$.*

Based on the formulations and *Assumption 2*, the AUV model studied in this chapter can be constructed as below:

$$\dot{u} = \frac{M_{\dot{v}}}{M_{\dot{u}}} vr - \frac{X_u}{M_{\dot{u}}} u - \frac{D_u}{M_{\dot{u}}} u|u| + \frac{F_u}{M_{\dot{u}}} + w_x \tag{2.4a}$$

$$\dot{v} = -\frac{M_{\dot{u}}}{M_{\dot{v}}} ur - \frac{Y_v}{M_{\dot{v}}} v - \frac{D_v}{M_{\dot{v}}} v|v| + \frac{F_v}{M_{\dot{v}}} + w_y \tag{2.4b}$$

$$\dot{r} = \frac{M_{\dot{u}} - M_{\dot{v}}}{M_{\dot{r}}} uv - \frac{N_r}{M_{\dot{r}}} r - \frac{D_r}{M_{\dot{r}}} r|r| + \frac{F_r}{M_{\dot{r}}}. \tag{2.4c}$$

Since the disturbances that occur in the systems are unknown and unpredictable, instead of directly studying the behavior of the real system dynamics (2.1) (2.2), we can construct a nominal system model without the interference of the disturbances. By optimizing the behavior of the nominal system under certain conditions, the robustness and resilience subject to disturbances and DoS attacks in the actual system can be guaranteed.

Define $\widehat{\mathrm{x}} = [\widehat{x}, \widehat{y}, \widehat{\psi}, \widehat{u}, \widehat{v}, \widehat{r}]^{\mathrm{T}}$ to be the nominal system state of the AUV. Then, according to the dynamics of the real system (2.1) (2.2), the dynamics of the nominal system can be formulated as:

$$\begin{bmatrix} \dot{\widehat{x}} \\ \dot{\widehat{y}} \\ \dot{\widehat{\psi}} \\ \dot{\widehat{u}} \\ \dot{\widehat{v}} \\ \dot{\widehat{r}} \end{bmatrix} = \begin{bmatrix} \widehat{u}\cos\widehat{\psi} - \widehat{v}\sin\widehat{\psi} \\ \widehat{u}\sin\widehat{\psi} + \widehat{v}\cos\widehat{\psi} \\ \widehat{r} \\ \frac{M_{\dot{v}}}{M_{\dot{u}}}\widehat{v}\widehat{r} - \frac{X_u}{M_{\dot{u}}}\widehat{u} - \frac{D_u}{M_{\dot{u}}}\widehat{u}|\widehat{u}| + \frac{F_u}{M_{\dot{u}}} \\ -\frac{M_{\dot{u}}}{M_{\dot{v}}}\widehat{u}\widehat{r} - \frac{Y_v}{M_{\dot{v}}}\widehat{v} - \frac{D_v}{M_{\dot{v}}}\widehat{v}|\widehat{v}| + \frac{F_v}{M_{\dot{v}}} \\ \frac{M_{\dot{u}} - M_{\dot{v}}}{M_{\dot{r}}}\widehat{u}\widehat{v} - \frac{N_r}{M_{\dot{r}}}\widehat{r} - \frac{D_r}{M_{\dot{r}}}\widehat{r}|\widehat{r}| + \frac{F_r}{M_{\dot{r}}} \end{bmatrix} \tag{2.5}$$

After deriving (2.5), the actual system behavior without disturbances can be studied by utilizing the nominal system.

2.2.2 Error Dynamics

For the purpose of tracking the reference signal and simplifying the numerical model operated in the MPC algorithm, constructing an error model for the path-following problem is necessary. The following error system is set up following the idea from [62].

Firstly, we need to define the reference signal x_r to track. The reference signal contains the reference path $\eta_r = [x_r,\, y_r,\, \psi_r]^{\mathrm{T}}$ and the reference body-fixed velocity $\mathrm{v}_r = [u_r,\, v_r,\, r_r]^{\mathrm{T}}$, where x_r and y_r are pre-designed reference position, and

$$\psi_r = \operatorname{atan2}(\dot{y}_r, \dot{x}_r), \tag{2.6}$$

where atan2 is the four-quadrant inverse tangent operator.

Recalling (2.1), the body-fixed velocity v_r can be formulated as:

$$\mathrm{v}_r = R^{-1}(\psi)\dot{\eta}_r = \begin{bmatrix} \cos\psi_r \dot{x}_r + \sin\psi_r \dot{y}_r \\ -\sin\psi_r \dot{x}_r + \cos\psi_r \dot{y}_r \\ \frac{-\dot{y}_r}{\dot{x}_r^2 + \dot{y}_r^2} \end{bmatrix}. \tag{2.7}$$

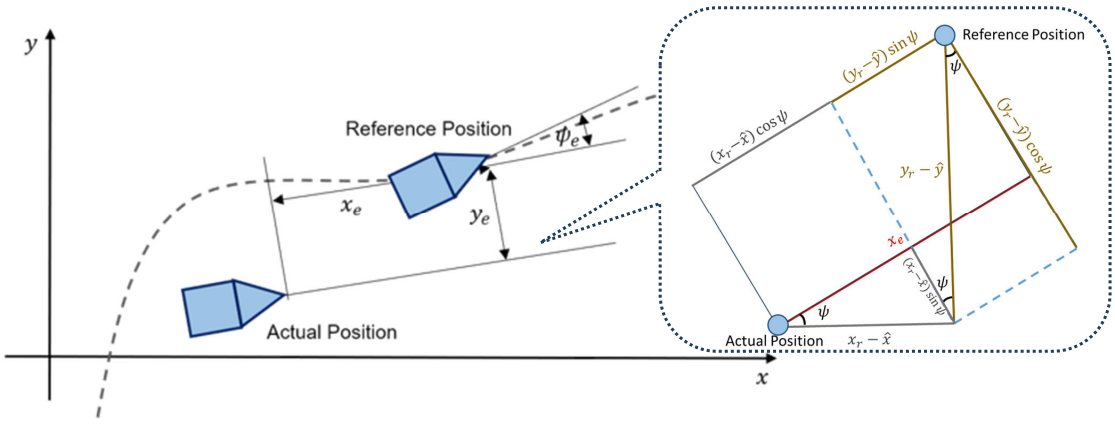

Figure 2.2: Illustration for the tracking error.

Equivalently, the reference path can also be viewed as the reference state trajectory generated by a reference system with the same kinematic model of the real vehicle:

$$\dot{\eta}_r = \begin{bmatrix} \cos\psi_r & \sin\psi_r & 0 \\ -\sin\psi_r & \cos\psi_r & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix}. \tag{2.8}$$

Here, we define the error state $e = [x_e,\, y_e,\, \psi_e,\, u_e,\, v_e,\, r_e]^{\mathrm{T}}$ as shown in Figure 2.2 to denote the tracking error between the nominal system's states and the reference system's states in the body reference frame. As illustrated in Figure 2.2, with the reference signal defined above, the kinematic state error $\eta_e = [x_e,\, y_e,\, \psi_e]^{\mathrm{T}}$ in the AUV parallel reference frame can be calculated as

$$\eta_e = \begin{bmatrix} x_e \\ y_e \\ \psi_e \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - \widehat{x} \\ y_r - \widehat{y} \\ \psi_r - \widehat{\psi} \end{bmatrix}. \tag{2.9}$$

Furthermore, the body-fixed velocity of the error system $\mathrm{v}_e = [u_e,\, v_e,\, r_e]^{\mathrm{T}}$ can be defined as:

$$\begin{aligned} u_e &= \widehat{u} - u_r\cos\psi_e + v_r\sin\psi_e \\ v_e &= \widehat{v} - u_r\sin\psi_e - v_r\cos\psi_e \\ r_e &= \widehat{r} - r_r \end{aligned} \tag{2.10}$$

By differentiating (2.9) and (2.10) with the approximation $\sin\psi \approx 0$ and $\cos\psi \approx 1$,

we can derive the kinematic error equations:

$$\begin{aligned}
\dot{x}_e &= y_e r_r - u_e + y_e r_e \\
\dot{y}_e &= -x_e r_r - v_e - x_e r_e \\
\dot{\psi}_e &= -r_e \\
\dot{u}_e &= \dot{\widehat{u}} - \dot{u}_r \cos\psi_e - u_r \sin\psi_e r_e + \dot{v}_r \sin\psi_e - v_r \cos\psi_e r_e \\
\dot{v}_e &= \dot{\widehat{v}} - \dot{u}_r \sin\psi_e + u_r \cos\psi_e r_e - \dot{v}_r \cos\psi_e - v_r \sin\psi_e r_e \\
\dot{r}_e &= \dot{\widehat{r}} - \dot{r}_r
\end{aligned} \tag{2.11}$$

Substituting (2.4) into (2.11), we can obtain

$$\begin{aligned}
\dot{u}_e =& \frac{M_{\dot{v}}}{M_{\dot{u}}}\widehat{v}\widehat{r} - \frac{X_u}{M_{\dot{u}}}\widehat{u} - \frac{D_u}{M_{\dot{u}}}\widehat{u}|\widehat{u}| + \frac{F_u}{M_{\dot{u}}} \\
& - \dot{u}_r \cos\psi_e - u_r \sin\psi_e r_e + \dot{v}_r \sin\psi_e - v_r \cos\psi_e r_e \\
\dot{v}_e =& -\frac{M_{\dot{u}}}{M_{\dot{v}}}\widehat{u}\widehat{r} - \frac{Y_v}{M_{\dot{v}}}\widehat{v} - \frac{D_v}{M_{\dot{v}}}\widehat{v}|\widehat{v}| + \frac{F_v}{M_{\dot{v}}} \\
& - \dot{u}_r \sin\psi_e + u_r \cos\psi_e r_e - \dot{v}_r \cos\psi_e - v_r \sin\psi_e r_e \\
\dot{r}_e =& \frac{M_{\dot{u}} - M_{\dot{v}}}{M_{\dot{r}}}\widehat{u}\widehat{v} - \frac{N_r}{M_{\dot{r}}}\widehat{r} - \frac{D_r}{M_{\dot{r}}}(r_e + r_r)|\widehat{r}| + \frac{F_r}{M_{\dot{r}}} - \dot{r}_r.
\end{aligned} \tag{2.12}$$

Integrating (2.12) with (2.10), the error speed dynamics can be formulated as:

$$\begin{aligned}
\dot{u}_e =& \frac{M_{\dot{v}}}{M_{\dot{u}}}(v_e + u_r \sin\psi_e + v_r \cos\psi_e)(r_e + r_r) - \frac{X_u}{M_{\dot{u}}}(u_e + u_r \cos\psi_e - v_r \sin\psi_e) \\
& - \frac{D_u}{M_{\dot{u}}}(u_e + u_r \cos\psi_e - v_r \sin\psi_e)|u_e + u_r \cos\psi_e - v_r \sin\psi_e| + \frac{F_u}{M_{\dot{u}}} \\
& - \dot{u}_r \cos\psi_e - u_r \sin\psi_e r_e + \dot{v}_r \sin\psi_e - v_r \cos\psi_e r_e \\
\dot{v}_e =& -\frac{M_{\dot{u}}}{M_{\dot{v}}}(u_e + u_r \cos\psi_e - v_r \sin\psi_e)(r_e + r_r) - \frac{Y_v}{M_{\dot{v}}}(v_e + u_r \sin\psi_e + v_r \cos\psi_e) \\
& - \frac{D_v}{M_{\dot{v}}}(v_e + u_r \sin\psi_e + v_r \cos\psi_e)|v_e + u_r \sin\psi_e + v_r \cos\psi_e| + \frac{F_v}{M_{\dot{v}}} \\
& - \dot{u}_r \sin\psi_e + u_r \cos\psi_e r_e - \dot{v}_r \cos\psi_e - v_r \sin\psi_e r_e \\
\dot{r}_e =& \frac{M_{\dot{u}} - M_{\dot{v}}}{M_{\dot{r}}}(u_e + u_r \cos\psi_e - v_r \sin\psi_e)(v_e + u_r \sin\psi_e + v_r \cos\psi_e) \\
& - \frac{N_r}{M_{\dot{r}}}(r_e + r_r) - \frac{D_r}{M_{\dot{r}}}(r_e + r_r)|r_e + r_r| + \frac{F_r}{M_{\dot{r}}} - \dot{r}_r.
\end{aligned} \tag{2.13}$$

To simplify the dynamic model in the optimization problem with comparable performance, we further design $F = [F_u,\, F_v,\, F_r]^{\mathrm{T}}$ as follows:

$$\begin{aligned}
F_u =& M_{\dot{v}}(v_e + u_r \sin\psi_e + v_r \cos\psi_e)(r_e + r_r) - X_u(u_e + u_r \cos\psi_e - v_r \sin\psi_e) \\
& - D_u(u_e + u_r \cos\psi_e - v_r \sin\psi_e)|u_e + u_r \cos\psi_e - v_r \sin\psi_e| \\
& + M_{\dot{u}}(-\dot{u}_r \cos\psi_e - u_r \sin\psi_e r_e + \dot{v}_r \sin\psi_e - v_r \cos\psi_e r_e) + M_{\dot{u}}\tau_u \\
F_v =& -M_{\dot{u}}(u_e + u_r \cos\psi_e - v_r \sin\psi_e)(r_e + r_r) - Y_v(v_e + u_r \sin\psi_e + v_r \cos\psi_e) \\
& - D_v(v_e + u_r \sin\psi_e + v_r \cos\psi_e)|v_e + u_r \sin\psi_e + v_r \cos\psi_e| \\
& + M_{\dot{v}}(-\dot{u}_r \sin\psi_e + u_r \cos\psi_e r_e - \dot{v}_r \cos\psi_e - v_r \sin\psi_e r_e) + M_{\dot{v}}\tau_v \\
F_r =& (M_{\dot{u}} - M_{\dot{v}})(u_e + u_r \cos\psi_e - v_r \sin\psi_e)(v_e + u_r \sin\psi_e + v_r \cos\psi_e) \\
& - N_r(r_e + r_r) - D_r(r_e + r_r)|r_e + r_r| - M_{\dot{r}}\tau_r.
\end{aligned} \tag{2.14}$$

By defining $\tau = [\tau_u,\, \tau_v,\, \tau_r]^{\mathrm{T}}$ as above, together with (2.11), we can derive the error

dynamics for the AUV tracking control:

$$\dot{e} = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\psi}_e \\ \dot{u}_e \\ \dot{v}_e \\ \dot{r}_e \end{bmatrix} = \begin{bmatrix} y_e r_r - u_e + y_e r_e \\ -x_e r_r - v_e - x_e r_e \\ -r_e \\ \tau_u \\ \tau_v \\ \tau_r \end{bmatrix} = f_e(e, \tau). \tag{2.15}$$

As a result, the simplified error model for the trajectory tracking problem is constructed. By applying this simplified model, the trajectory-tracking problem on the AUV can be transformed into a regulation problem for the error dynamics, which is much simpler to design a controller.

2.2.3 DoS Attacks

To control the AUV remotely to achieve allocated tasks, the communication channels between the controller and the AUV are often set using network-based structures due to their convenience. However, the network-based channels in the AUV control are thought fragile and vulnerable to cyber attacks [85]. In this work, we especially consider the DoS attacks that occur on the remote controller to actuator channel, which is illustrated in Figure 2.3.

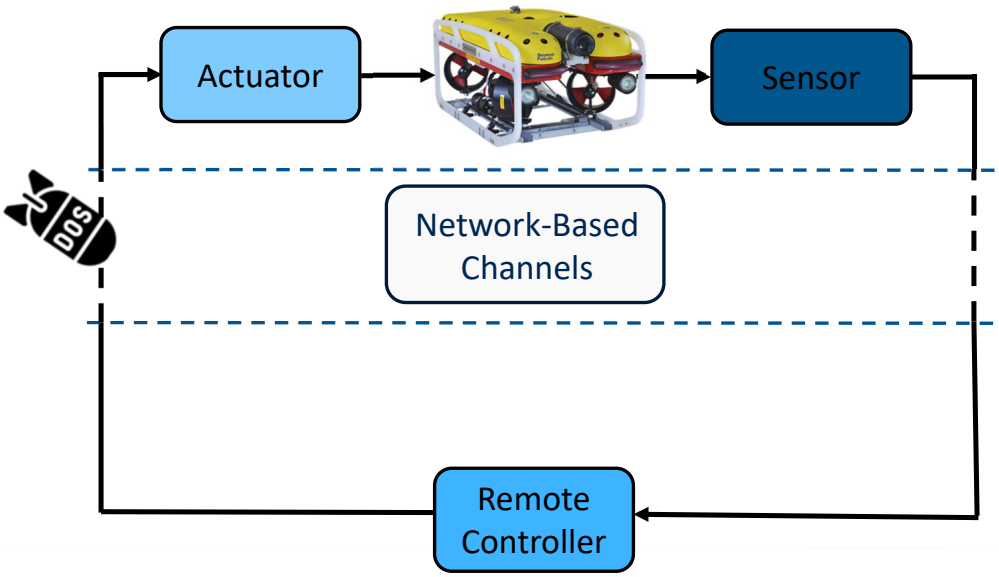

Figure 2.3: DoS attacks on the C-A channel.

Let $\mathcal{T}^a \triangleq \{t_l^a\}$ and $\mathcal{D}^a \triangleq \{d_l^a\}$ denote all the launching time instants and their corresponding duration of the DoS attacks on the C-A channel, respectively. Here, l denotes the lth launching. We define

$$\Xi(t_0, t_1) \triangleq \bigcup_{l \in \mathbb{N}} [t_l^a,\, t_l^a + d_l^a), \tag{2.16}$$

to represent the total DoS attacks activation duration during the time period $[t_0, t_1)$. With this definition, when $t \in \Xi(t_0, t_1)$, the communication channel from the controller to the AUV will be totally blocked ($F = \mathbf{0}$), making the AUV out of control, and preventing it from achieving its control objective. However, it is impractical when the duration of DoS attacks approaches infinity. In this regard, an assumption should be made.

Assumption 3. *The maximum duration of the DoS attacks investigated in this work is defined as T_a and is assumed to be bounded. Furthermore, the interval between the adjacent DoS attacks launching time should at least contain one sampling instant.*

Regarding *Assumption 3*, we assume that the DoS attacks targeting the C-A channel of the network-based structure have a finite duration and occur with a minimum interval. We propose that our framework can effectively handle and mitigate these attacks.

2.2.4 Control Objective

For the AUV whose dynamic model can be described in (2.2), we aim to develop a robust and resilient MPC framework to control the AUV following a pre-designed trajectory under the existence of

- External disturbances directly in the AUV model;
- DoS attacks that randomly occur on the C-A channel.

2.3 Robust and Resilient Controller Design for the Trajectory Tracking Problem

In this section, to address the aforementioned issues, methodologies including robust and resilient MPC design and packet transmission strategy are introduced.

2.3.1 Robust and Resilient MPC Design

To tackle the constraints of the AUV and mitigate the effect caused by the disturbances and the DoS attacks, we apply a robust and resilient MPC mechanism to this AUV trajectory tracking problem. The resilient MPC technique is utilized to stabilize the error dynamics so as to make the actual AUV system track the reference path.

The objective is realized through the minimization of the following cost function:

$$J\left(t,\, e(t),\, \tau(t)\right) = \int_{t}^{t+T} \mathcal{L}(e(s),\, \tau(s))\mathrm{d}s + \mathcal{L}_f\left(e(t+T)\right), \tag{2.17}$$

where t is the current time; T is the prediction horizon; $e(t)$ is the error state, which as the system dynamics defined in (2.15); $\mathcal{L}(e,\, \tau) = \|e\|_Q^2 + \|\tau\|_R^2$ is the stage cost, where $Q \succ 0$ and $R \succeq 0$ are the weighting matrices with appropriate dimensions; $\mathcal{L}_f\left(e\right)$ is the terminal penalty satisfying $\mathcal{L}_f\left(\mathbf{0}\right) = 0$ and $\mathcal{L}_f\left(e\right) > 0$ for any $e \neq 0$. With the definition above, the optimization problem to be solved at each sampling instant is formulated as follows:

$$\begin{aligned}
\boldsymbol{\tau}^* = &\arg\min_{\boldsymbol{\tau}} \quad \left\{J\left(t,\, e(t),\, \tau(t)\right)\right\} \\
\text{s.t.} \quad & \dot{e}(s|t) = f_e\left(e(s|t), \tau(s|t)\right) && \text{(2.18a)} \\
& \tau(s|t) \in \mathbb{T}_e && \text{(2.18b)} \\
& \eta_e(s|t) \in \left(1 - \frac{s-t}{T}\zeta\right) C_e && \text{(2.18c)} \\
& e(t+T) \in \bar{\mathbb{X}}_f && \text{(2.18d)} \\
& s \in [t,\, t+T]
\end{aligned}$$

Here, $\boldsymbol{\tau}^*$ is the optimized control input sequence; (2.18b) is the input constraint, where $\mathbb{T}_e$ is the compact control input set; (2.18c) is the tightening state constraint, where η_e, defined in (2.9), is the first three elements in e; C_e is the manually set state constraint, where each element i in η_e satisfies that $i \in \{i|\|i\| \leq c_e\}$ (c_e is a tuning parameter); (2.18d) is the tightening terminal constraint, where $\bar{\mathbb{X}}_f = \{e|\|e\|_P^2 \leq \xi\epsilon^2\}$ is the terminal constraint on the error state and P is the terminal penalty matrix; Note that (2.18c) and (2.18d) are the robust constraints, in which ζ and ξ are the designed shrinking parameters. The main idea of the shrinking parameters is to guarantee that

the actual system state is confined to the original state constraint and satisfies the terminal constraint when the nominal system state satisfies the tightening ones.

Remark 1. *Following the idea in [59], for the AUV trajectory-tracking control, the constraints on the actual torques are usually set as*

$$\begin{aligned} F_{u,min} \leq F_u \leq F_{u,max} \\ F_{v,min} \leq F_v \leq F_{v,max} \\ F_{r,min} \leq F_r \leq F_{r,max}. \end{aligned} \tag{2.19}$$

Since the optimization problem (2.18) *does not contain the integral torque F, we can transfer the original torque constraints to a new input constraint $\tau \in \mathbb{T}_e$, where $\mathbb{T}_e$ is a compact region that satisfies*

$$\begin{aligned} \frac{F_{u,\min} - F_{1,\max}}{M_{\dot{u}}} \leq \tau_u \leq \frac{F_{u,\max} - F_{1,\min}}{M_{\dot{u}}} \\ \frac{F_{v,\min} - F_{2,\max}}{M_{\dot{v}}} \leq \tau_u \leq \frac{F_{v,\max} - F_{2,\min}}{M_{\dot{v}}} \\ \frac{F_{r,\min} - F_{3,\max}}{M_{\dot{r}}} \leq \tau_r \leq \frac{F_{r,\max} - F_{3,\min}}{M_{\dot{r}}}. \end{aligned} \tag{2.20}$$

Here, $F_1 = F_u - M_{\dot{u}}\tau_u$, $F_2 = F_v - M_{\dot{v}}\tau_v$, $F_3 = F_r - M_{\dot{r}}\tau_r$, and the subscript min *and* max *represents the minimum and maximum value of these three integral torques within the limited trajectory. Nevertheless, in some trajectories, this constraint-transferring method may be too conservative to operate. In this case, we can also design a new decision variable F here with the definition of* (2.14)*. By adding this nonlinear constraint, the constraint on the torque can also be solved but requires relatively more computational burden.*

At each sampling instant, the optimization problem (2.18) is solved with the current error state $e(t)$ that is generated through (2.9) and (2.10) to generate the optimal control input sequence $\boldsymbol{\tau}^*(t)$ and the predicted error state sequence $\boldsymbol{e}^a(t)$. Note that

the control input $\boldsymbol{\tau}^*(t)$ generated at each sampling instant cannot be directly applied to the AUV since it is not the actual torque. After deriving $\boldsymbol{\tau}^*(t)$, we need to generate $F(t)$ according to (2.14) to be the integral torque. At the next sampling instant $t+\delta$, where δ is the sampling period, the optimization problem (2.18) is solved again by utilizing the latest measured state data $e(t+\delta)$. By solving the optimization problem recursively, the trajectory-tracking goal can be achieved.

2.3.2 Packet Transmission Strategy

In this remote AUV control problem, the communication channel from the remote controller to the actuator (AUV) is considered to suffer random existing DoS attacks. Hence, the AUV may receive nothing when the attacks happen and become totally out of control. To overcome this issue, refer to [66], packet transmission strategy is utilized in this work to mitigate the lack of control input caused by DoS attacks.

After solving the optimization problem (2.18) at each sampling instant, the optimized control input sequence $\boldsymbol{\tau}^*$ can be derived and the predicted error state sequence $\boldsymbol{e}^a$ can be calculated through:

$$\begin{aligned} e^a(t|t) &= e(t|t) \\ \tau^*(s|t) &= \tau(s) \\ \dot{e}^a(s|t) &= f_e\left(e^a(s|t), \tau(s)\right), \quad s \in [t, t+T). \end{aligned} \tag{2.21}$$

where $\tau^*(s|t)$ and $e^a(s|t)$ represents the control input and the error state at time s that is predicted at time t, respectively.

However, the control input sequence cannot be directly applied to the AUV since the actual torque has not been generated yet. According to (2.14), the optimized

torque input sequence $\boldsymbol{F}^a$ can be generated as

$$F^a(s|t) = \begin{bmatrix} F_u^a(s|t) \\ F_v^a(s|t) \\ F_r^a(s|t) \end{bmatrix} \tag{2.22}$$

where $F_u^a(s|t)$, $F_v^a(s|t)$, and $F_r^a(s|t)$ are formulated as

$$\begin{aligned}
F_u^a(s|t) =& M_{\dot{v}}(v_e(s) + u_r(s)\sin\psi_e(s) + v_r(s)\cos\psi_e(s))(r_e(s) + r_r(s)) \\
&- X_u(u_e(s) + u_r(s)\cos\psi_e(s) - v_r(s)\sin\psi_e(s)) \\
&- D_u(u_e(s) + u_r(s)\cos\psi_e(s) - v_r(s)\sin\psi_e(s)) \\
&\times |u_e(s) + u_r(s)\cos\psi_e(s) - v_r(s)\sin\psi_e(s)| \\
&+ M_{\dot{u}}(-\dot{u}_r(s)\cos\psi_e(s) - u_r(s)\sin\psi_e(s)r_e(s) \\
&+ \dot{v_r}(s)\sin\psi_e(s) - v_r(s)\cos\psi_e(s)r_e(s)) + M_{\dot{u}}\tau_u(s) \qquad (2.23a) \\
F_v^a(s|t) =& - M_{\dot{u}}(u_e(s) + u_r(s)\cos\psi_e(s) - v_r(s)\sin\psi_e(s))(r_e(s) + r_r(s)) \\
&- Y_v(v_e(s) + u_r(s)\sin\psi_e(s) + v_r(s)\cos\psi_e(s)) \\
&- D_v(v_e(s) + u_r(s)\sin\psi_e(s) + v_r(s)\cos\psi_e(s)) \\
&\times |v_e(s) + u_r(s)\sin\psi_e(s) + v_r(s)\cos\psi_e(s)| \\
&+ M_{\dot{v}}(-\dot{u}_r(s)\sin\psi_e(s) + u_r(s)\cos\psi_e(s)r_e(s) \\
&- \dot{v_r}(s)\cos\psi_e(s) - v_r(s)\sin\psi_e(s)r_e(s)) + M_{\dot{v}}\tau_v(s) \qquad (2.23b) \\
F_r^a(s|t) =& (M_{\dot{u}} - M_{\dot{v}})(u_e(s) + u_r(s)\cos\psi_e(s) - v_r(s)\sin\psi_e(s)) \\
&\times (v_e(s) + u_r(s)\sin\psi_e(s) + v_r(s)\cos\psi_e(s)) - N_r(r_e(s) + r_r(s)) \\
&- D_r(r_e(s) + r_r(s))|r_e(s) + r_r(s)| - M_{\dot{r}}\tau_r(s), \qquad (2.23c)
\end{aligned}$$

where $s \in [t, t+T]$. Instead of only sending an integral torque input to the actuator of the AUV, at each time instant, the controller utilizes the optimal control input

sequence $\boldsymbol{\tau}^*(t)$ and the predicted error state sequence $\boldsymbol{e}(t)$ to construct an optimal integral torque sequence $\boldsymbol{F}^a(t)$ according to (2.23) and send this sequence to a buffer on the actuator, which is shown in Figure 2.4.

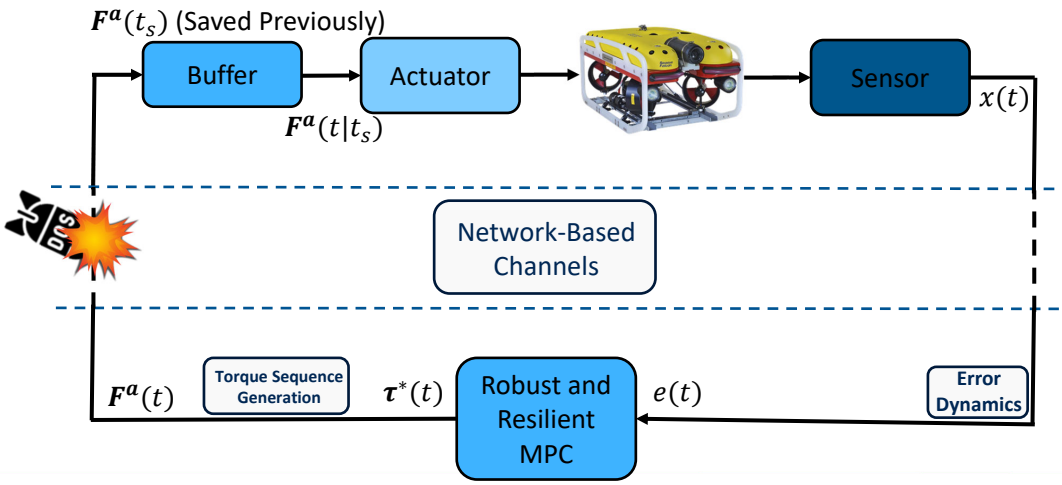

Figure 2.4: Robust and resilient MPC framework.

However, considering the randomly existing DoS attacks on the C-A channel, the utilization of the packet generated at each sampling instant needs to be discussed in detail. Before showing details, we first define t_s as the latest successful transmission time from the remote controller to the AUV and $F(t)$ to be the actual torque that is applied on the AUV at time t. Then the following two cases can be considered:

- When the network-based C-A channel does not suffer DoS attacks at time t, the buffer can successfully receive the integral torque sequence $\boldsymbol{F}^a(t)$. In this case, the actuator can directly utilize the first column of the sequence in $\boldsymbol{F}^a(t)$.

$$F(t) = \boldsymbol{F}^a(t|t) \tag{2.24}$$

  Then, the latest successful transmission time needs to be updated with $t_s = t$ to record that the information stored in the buffer is generated at time t.

- When the network-based C-A channel suffers DoS attacks at time t, the buffer receives nothing from the controller. Under this circumstance, the buffer can send the predicted integral torque input that represents the proper input at this moment $\boldsymbol{F}^a(t_s)$ (saved at time t_s) to the actuator, which can be represented as

$$F(t) = \boldsymbol{F}^a(t|t_s) \tag{2.25}$$

  In this way, the candidate torque can still be utilized in the actuator even when the attacks exist on the C-A channel.

To conclude, by applying the packet transmission strategy, there will always be integral torque that can be utilized in the actuator whenever the DoS attacks occur.

Remark 2. *According to the packet transmission strategy, to compensate for the lack of information caused by DoS attacks, the controller must fully utilize the predicted sequence generated at a previous time. Hence, when $t = 0$, we need to assume that the attacks do not exist. In addition, the successful data transmission time t_s is updated only when the communication channel is not attacked at such time. As a result, at time t, the buffer can determine whether DoS attacks have occurred on their information-receiving channels by comparing the values of t_s and t. When t_s does not match the current time t, it indicates that DoS attacks have affected the C-A channel.*

2.3.3 Robust and Resilient MPC Framework for the AUV

Based on the aforementioned discussions, the flow chart of this control framework can be summarized as

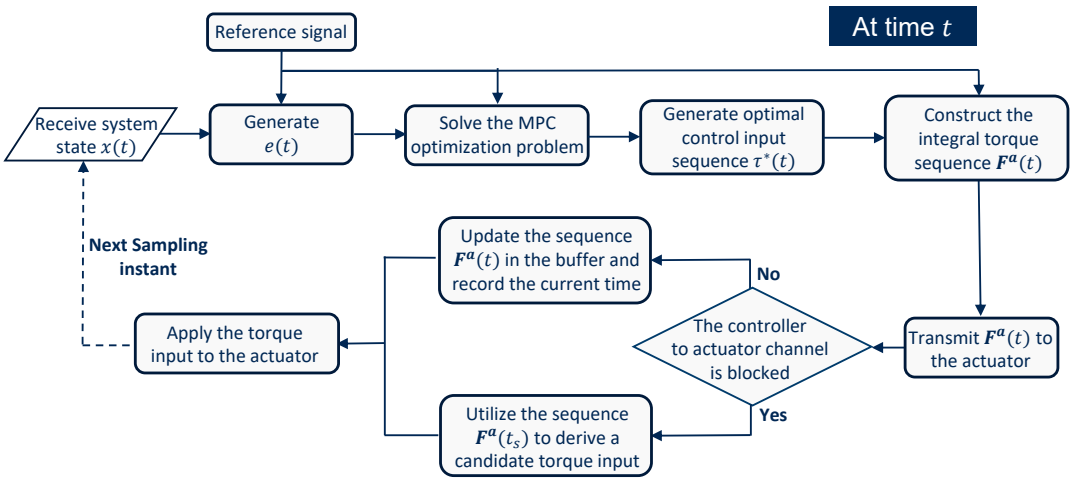

Figure 2.5: Robust and resilient MPC algorithm illustration.

As illustrated in Figure 2.5, assume that the current time is t, after receiving the state information from the sensor of AUV, the controller will generate the error state $e(t)$ based on (2.11). Then, the resilient MPC optimization problem (2.18) can be solved to generate the optimized control input sequence $\boldsymbol{\tau}^*$. However, this sequence cannot be directly applied to the AUV since it is not the actual torque. Consequently, we need to construct the integral torque sequence $\boldsymbol{F}^a(t)$ according to (2.23). Having derived this sequence, a discussion should be made to tackle DoS attacks that randomly affect the data transmission on the C-A channel, which is illustrated in Section 2.3.2. After that, at time $t + \delta$, all the steps will be operated again using the latest measured state information $\mathrm{x}(t + \delta)$. To sum up, the resilient MPC for the AUV to track a predesigned trajectory can be conducted in *Algorithm 1*.

By performing this algorithm, the AUV is able to track the pre-designed reference path under the DoS attacks and disturbances. The effectiveness of the whole robust and resilient MPC framework will be verified through simulation results in the next section.

Algorithm 1 Robust and Resilient MPC algorithm.

Require: Given initial system state $\mathbf{x}(0)$, the weighting matrices Q and R; the reference path η_r and the reference body-fixed velocity v_r the terminal penalty matrix P; the sampling period δ; the prediction horizon T; the terminal set level ϵ; scaling parameters ξ and ζ; Set $t = 0$, and $t_s = 0$.

1: **while** the control action is not stopped **do**
2: Receive the state $\mathrm{x}(t)$ of the AUV.
3: Generate the error state $e(t)$ referring to (2.11).
4: Solve the optimization problem (2.18), and generate the optimal control input sequence $\boldsymbol{\tau}^*(t)$.
5: Transfer $\boldsymbol{\tau}^*(t)$ into the integrated torque sequence $\boldsymbol{F}^a(t)$ according to (2.23).
6: Transmit the packet $\boldsymbol{F}^a(t)$ to the buffer.
7: **if** The network-based channel is not being attacked **then**
8: Utilize the first column in the packet to generate the integral torque (2.24) and send it to the actuator.
9: Update t_s with $t_s = t$.
10: **else**
11: Utilize the information that represents the predicted current input torque in the buffer to generate the integral torque (2.25) and send it to the actuator.
12: **end if**
13: $t = t + \delta$.
14: **end while**

Table 2.1: Parameters of the Saab SeaEye Falcon open-frame AUV

Inertia Term	Linear drag	Quadratic drag
$M_{\dot{u}} = 283.6\text{kg}$	$X_u = 26.9\text{kg/s}$	$D_u = 241.3\text{kg/s}$
$M_{\dot{v}} = 593.2\text{kg}$	$Y_v = 35.8\text{kg/s}$	$D_v = 503.8\text{kg/s}$
$M_{\dot{r}} = 29.0\text{kg}$	$N_r = 3.5\text{kg m}^2\text{/s}$	$D_r = 76.9\text{kg m}^2$

2.4 Simulation Study

To show the effectiveness of the proposed control framework, in this section, we perform a simulation here to control the Saab SeaEye Falcon open-frame AUV to track a pre-designed trajectory under the disturbances and the DoS attacks that randomly exist on the C-A channel.

2.4.1 Parameter Configuration

In this chapter, we only consider the vehicle in the local level plane. The system parameter of this Saab SeaEye Falcon open-frame AUV is shown in Table 2.4.1

To tune the control algorithm, weighting matrices $Q = \text{diag}([20,\ 8,\ 1,\ 1,\ 1,\ 1])$ and $R = \text{diag}([0.01,\ 0.01,\ 0.01])$; terminal penalty cost is designed as $\mathcal{L}_f\left(e(t+T)\right) = \|e(t+T)\|_P^2$ with $P = \text{diag}([0.5,\ 0.5,\ 0.5,\ 0.5,\ 0.5,\ 0.5])$; terminal set level $\epsilon = 0.01$, which is set the same as in [62]; shrinking parameters $\xi = 0.8$ and $\zeta = 0.25$; the sampling period is set as $\delta = 0.1\text{s}$ and the prediction horizon is set as $T = 8\delta$; the integral torque constraint is set as $F_{u,\max} = F_{v,\max} = F_{r,\max} = 9000$ and $F_{u,\min} = F_{v,\min} = F_{r,\min} = -5500$; as discussed in (2.20), the control input constraints are set as $-39.55 \leq \tau_u \leq 11.58$, $-16.54 \leq \tau_v \leq 7.90$, and $-189.71 \leq \tau_r \leq 310.30$; the disturbances in this simulation are $w_x = \frac{1}{800}\sin(\frac{t}{20} + \frac{\pi}{4})$, and $w_y = \frac{1}{1000}\cos(\frac{t}{25})$; the manually set error constraint is configured as $c_e = 0.5$.

2.4.2 Simulation Results

To show the effectiveness of this control framework, the proposed method is also compared with the standard MPC approach [12] without the packet transmission strategy and the robustness constraint. To make a comprehensive comparison, the simulation results contain one trajectory with three different attack scenarios.

The reference trajectory is designed as

$$\begin{cases} x_r(t) = 5t \\ y_r(t) = \sin\dfrac{t}{18} \end{cases}.$$

In addition, the initial state is $\mathrm{x}(0) = [-0.1, 0.1, 0.0349, 5, \cos(\frac{\pi}{18}), 0]^{\mathrm{T}}$.

Scenario 1: No attack case

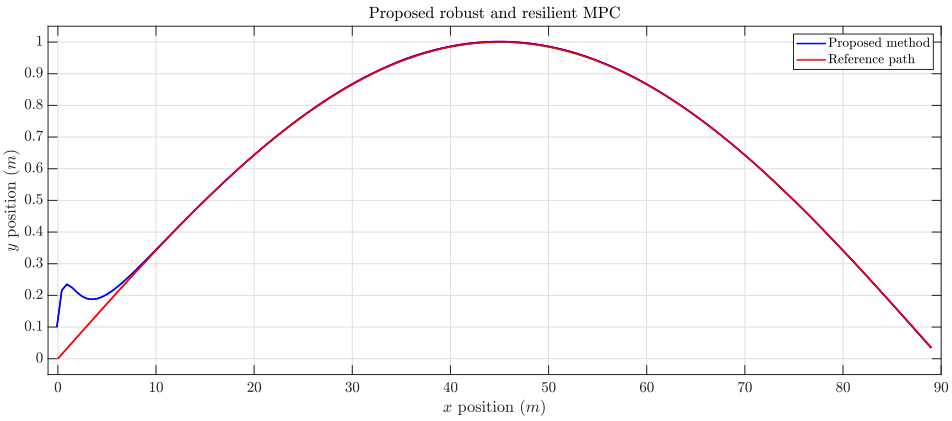

Figure 2.6: Tracking performance on XoY plane in Scenario 1.

In this case, the C-A channel does not suffer DoS attacks. It can be shown in Figure 2.6 that when there is no DoS attack, the proposed method can track the pre-designed trajectory well.

Scenario 2: DoS attacks with short duration

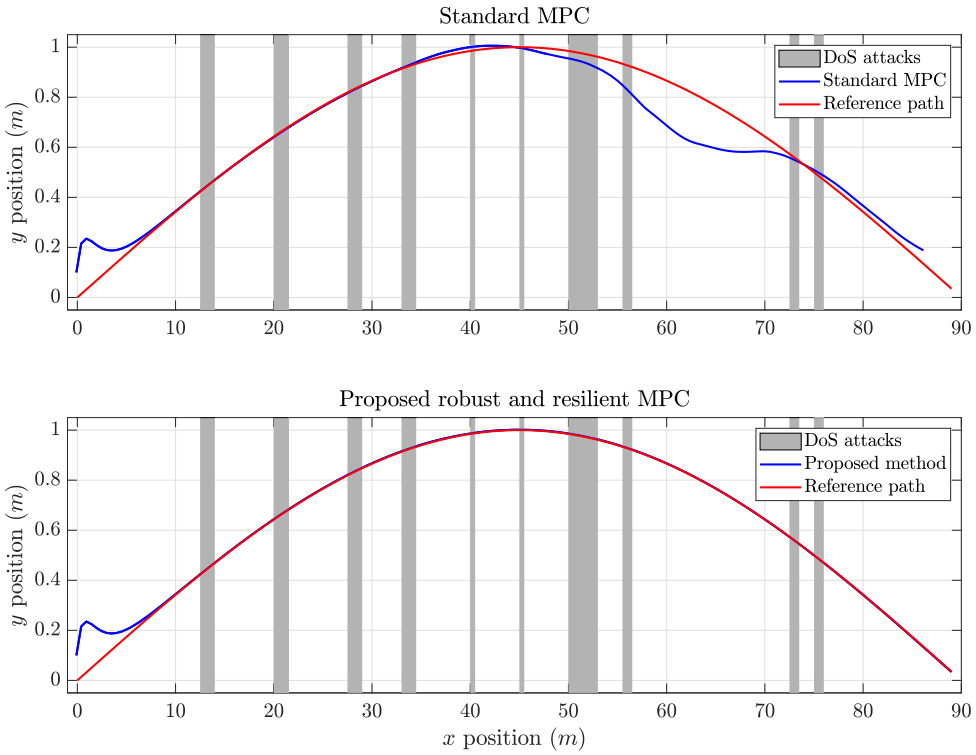

Figure 2.7: Tracking performance comparison on XoY plane in Scenario 2.

According to Figure 2.7, in which the grey area represents the existence time duration of DoS attacks, the standard MPC approach will suffer a huge deviation from the reference path when the duration of DoS attacks is relatively long, while the proposed method can successfully overcome this effect. In other words, when the activation time of the DoS attacks is relatively short, the proposed method can tackle this issue and achieve the tracking goal.

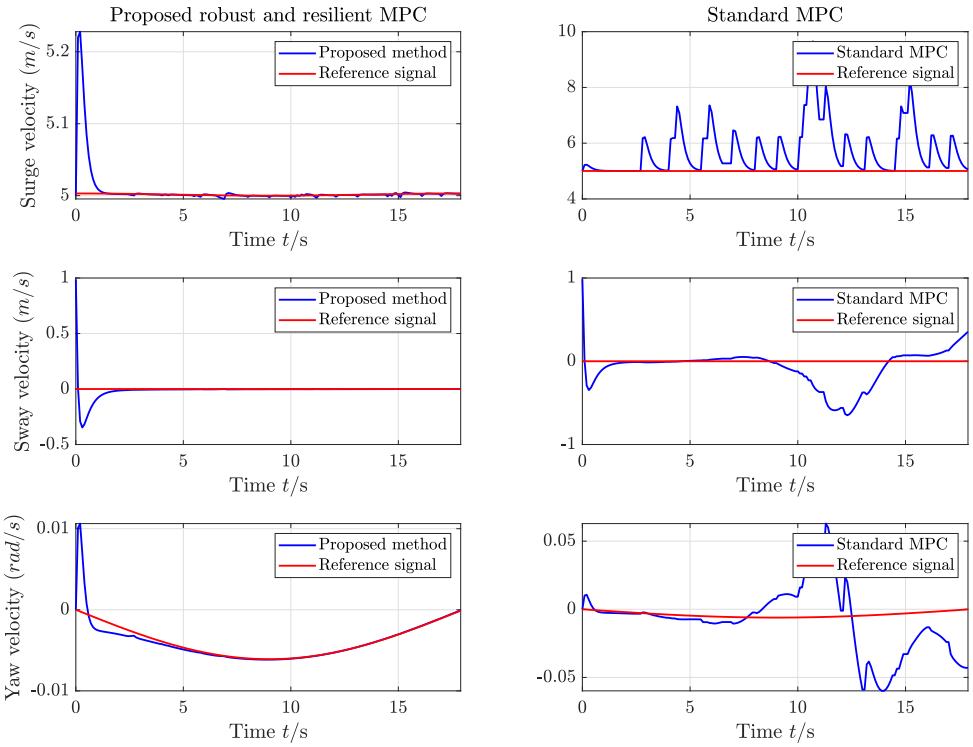

Figure 2.8: Comparison results of tracking yaw, surge, and sway velocities in scenario 2.

Fig 2.8 further shows the performance of tracking the yaw, surge, and sway velocities. As illustrated in this figure, the proposed method is able to not only track the position in the trajectory well but also follow the reference trajectory with exactly the same speed within the finite time intervals. In contrast, the standard MPC approach cannot track the velocities at all, which leads to the deviation between the actual path and the reference trajectory.

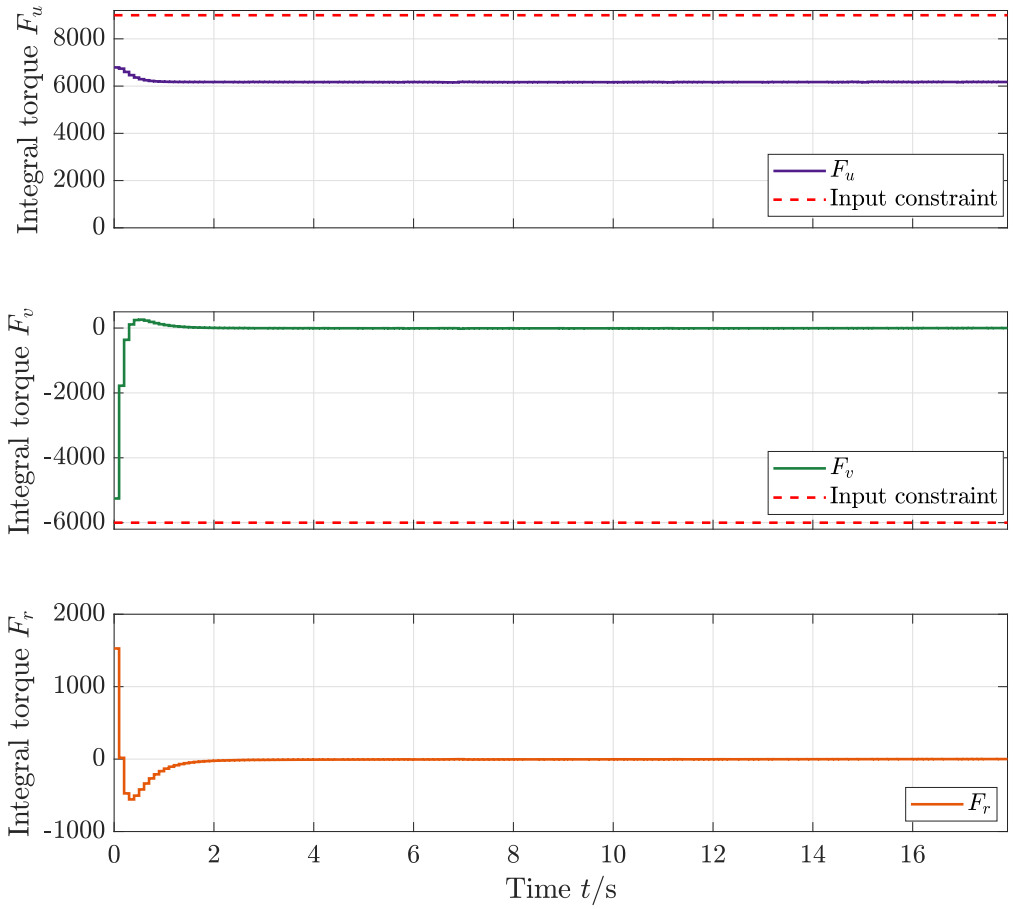

Figure 2.9: Illustration for the proposed control inputs.

In Figure 2.9, the control input sequence is shown. It can be clearly shown that the input constraint is satisfied by applying the proposed method, which meet the physical constraint on the actual AUV.

Scenario 3: DoS attacks with long duration

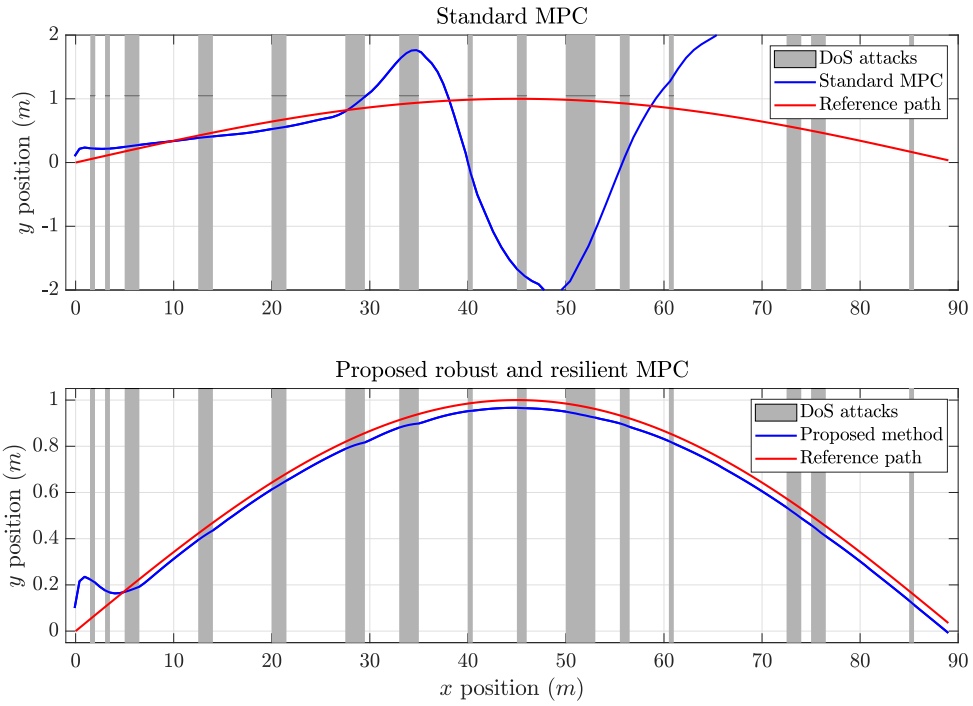

Figure 2.10: Tracking performance comparison on XoY plane in Scenario 2.

In this case, the duration of DoS attacks is set longer and more frequent than the previous one. As illustrated in Figure 2.10, the standard MPC method fails to track the reference signal. However, the proposed method, despite being slightly affected, can still effectively track the reference trajectory and successfully achieve the control objective, which further illustrate how powerful the proposed framework is.

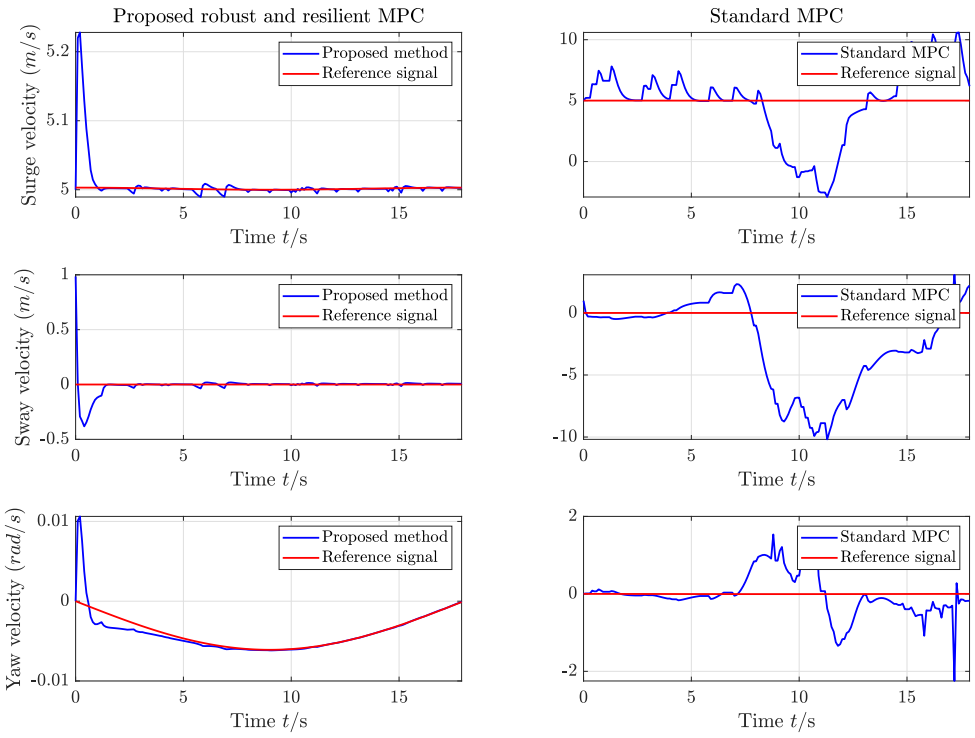

Figure 2.11: Comparison results of tracking yaw, surge, and sway velocities in scenario 3.

Figure 2.11 provides an explanation for the significant disparity in tracking performance. Specifically, when examining the tracking of yaw, sway, and yaw velocities, the two methods exhibit distinct tracking capabilities. In the case of more frequent DoS attacks, the standard MPC method fails to mitigate their impact, resulting in a complete loss of control over the AUV.

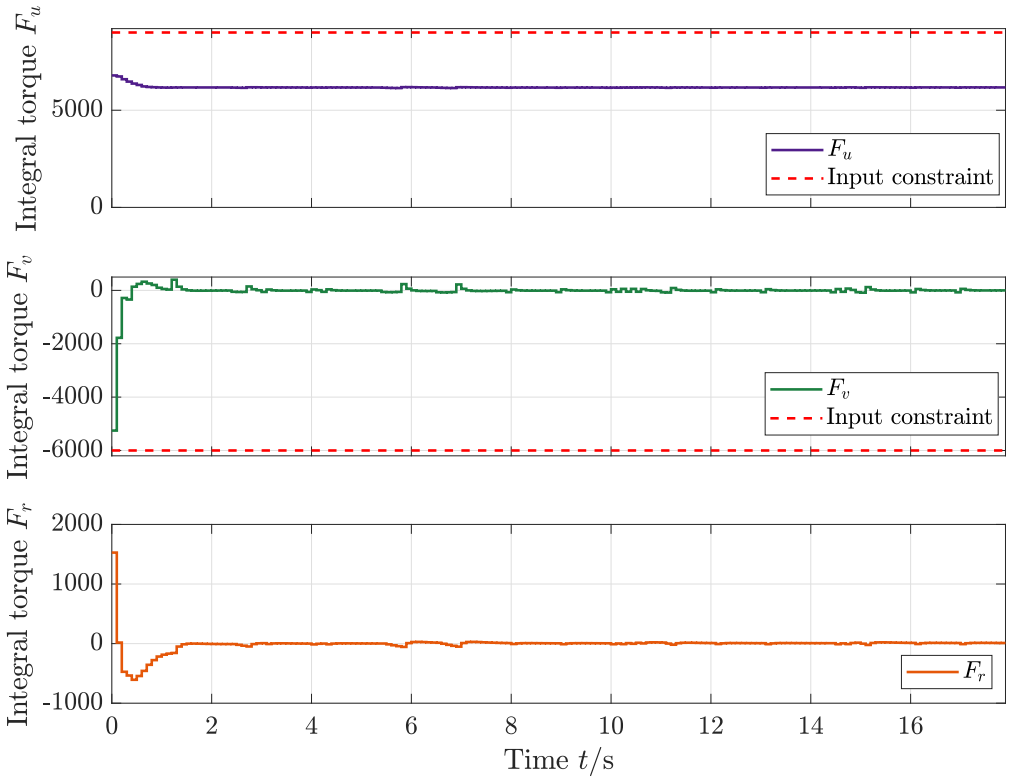

Figure 2.12: Illustration for the proposed control inputs.

In Figure 2.12, the control input torque sequences that are applied on the AUV are shown. It can be obtained that although the C-A channel suffers severe DoS attacks, the control input constraints can still be satisfied, which verifies the effectiveness of the proposed method.

2.5 Conclusion

In this chapter, the trajectory tracking problem for AUV is studied. A robust and resilient MPC framework is proposed to tackle the external disturbances and the DoS attacks that randomly exist on the C-A channel. According to the simulation results, the packet transmission strategy can successfully compensate for the lack of data

caused by the DoS attacks while the robustness constraint approach can ensure the feasibility of the tracking objective. As a result, the AUV under the proposed control framework is able to track the pre-designed trajectory, and comparison results clearly show the effectiveness of the proposed approach against DoS attacks. Note that the wonderful tracking results may be due to the fact that the disturbances cannot be set too large up to now, but future studies may involve the tube-based MPC structure, which can tackle disturbances with larger magnitude. However, when the duration of the DoS attacks is too long, there might exist a slight difference between the actual and the reference trajectories. This is mainly because the simplification process of generating the error model is coupled with the disturbances, which disturbs the prediction of the MPC. More accurate error dynamics need to be applied to narrow the deviation between the actual trajectory and the reference ones when the attacks last long.

Chapter 3

Robust and Resilient Distributed MPC for Cyber-Physical Systems Against DoS Attacks

3.1 Background of Resilient Distributed MPC

According to the aforementioned discussions, MPC offers a promising solution for CPSs tackling cyber attacks on both S-C and C-A channels in a single-agent architecture. In this regard, much research has been done from the perspective of enhancing confidentiality to designing a resilient control framework for the single-agent CPS. However, most CPSs contain two or more agents or subsystems, controlling them to achieve one goal cooperatively. Under this circumstance, the communication channels are much more complicated than the single-agent ones.

It is worthwhile mentioning that cyber attacks can be purposely launched on the communication channels among agents, thus adversely affecting information transmission among agents and aiming to destroy the safe system operation but there

exist very few results dedicated to this issue. In 2017, Velarde et. al. proposed a dual decomposition distributed MPC approach to robustify the original control framework by designing a model-based trust-worthy reference value for the whole system to follow to relieve the effect induced by fake reference sent by the attackers [74]. In [73], this method is improved to consider more attack actions such as selfish attacks (attacks that can modify the parameters in the cost function). However, this approach is only effective when the system is linear and it can only alleviate the effect, not tackle completely compensate for attacks. In [3], a distributed MPC strategy is developed for a platoon problem, but it only considers DoS attacks that are launched on the channels between two nonconsecutive neighboring agents. In [14], the authors developed an event-triggered resilient distributed MPC method for the platoon problem of a linearized network-based vehicle system, in which a special case that DoS attacks directly target on the agent (the attacked agent cannot communicate with all its neighbors) is considered. However, the resilient distributed MPC framework against adversarial attacks randomly launching at all communication channels has not been adequately investigated. It is underscored that the following research questions need to be addressed: How to make full use of the "prediction" feature of MPC to efficiently compensate for the effect caused by DoS attacks? How to analyze the effect of DoS attacks on the theoretical properties of the resulting overall system? How to construct a resilient and robust distributed MPC strategy? The development and main results of this chapter will provide affirmative answers to the above questions.

In this chapter, a resilient and robust distributed MPC is proposed for multi-agent CPSs, specifically considering DoS attacks randomly occurring on the communication channels among agents. The main contributions of this work are three-fold:

- Aiming at tackling the disturbances in the system, a new type of robustness constraint is constructed in the MPC optimization problem. Existing work about

robustness constraint [32–34] mainly constructs the constraint only based on the terminal constraint, resulting in a small region of attraction. The proposed robustness constraint is designed based on both the state constraint and the terminal constraint, which can enlarge the region of attraction with comparable control performance;

- A lengthened predicted state sequence scheme, depending on the attack duration, is proposed to facilitate the information transmission and efficient compensation for the lost information due to DoS attacks;

- The rigorous theoretical analysis of the performance of the proposed robust and resilient distributed MPC are provided. Sufficient conditions on guaranteeing the recursive feasibility of the proposed method and stability of the closed-loop system are derived, respectively. A numerical example and comparison results are shown to illustrate the effectiveness of the resulting method.

The remainder of the chapter is organized as follows: Section 3.2 formulates the problem with some preliminary results. Section 3.3 describes the robust and resilient control framework. In Section 3.4, sufficient conditions for ensuring recursive feasibility and closed-loop stability are established, respectively. Numerical examples and comparison results are illustrated in Section 3.5. Finally, the conclusion and future work are presented in Section 3.6.

Notations

The notations used in this chapter are fairly standard. The symbol $\mathbb{R}^n$ denotes the n-dimensional real space. The symbols $\mathbb{N}$ and $\mathbb{N}^+$ denote the set of all natural numbers and the set of all positive integers, respectively. Let $\mathbb{N}_{[a,\,b)}$ denote all the integers in the interval $[a,\,b], a < b$. Given a matrix P, $P \succ 0$ and $P \succeq 0$ denote that matrix P

is positive definite and positive semidefinite, respectively. For a vector $x \in \mathbb{R}^{n\times 1}$, $\|x\|$ denotes the Euclidean norm and $\|x\|_P$ denotes the P weighted Euclidean norm as $\sqrt{x^{\mathrm{T}}Px}$, where the matrix $P \succ 0$. The difference between the two sets is defined as $A \setminus B \triangleq \{x|x \in A, x \notin B\}$. $\lambda_{\max}(P)$ and $\lambda_{\min}(P)$ denote the largest and the smallest eigenvalues of matrix P, respectively. $\lceil r \rceil$ rounds r to the nearest integer toward positive infinity.

3.2 Problem Formulation

3.2.1 System Description

The communication topology of a nonlinear multi-agent CPS consisting of M agents can be illustrated as a directed graph $\mathscr{G} \triangleq \{\mathscr{M}, \mathscr{E}\}$, where $\mathscr{M} = \{i|i = 1, 2, ..., M\}$ represents the set of all agents and $\mathscr{E} \subset \mathscr{M} \times \mathscr{M}$ is the collection of all the communication channels among agents. Furthermore, the neighbor set of agent i is denoted as $\mathscr{N}_i$. With such definition of the neighbor set, if agent i can receive information from its neighbors j, then $j \in \mathscr{N}_i$.

At the discrete-time sampling instant $k \in \mathbb{N}$, the model of agent $i \in \mathscr{M}$ is described as

$$x_i(k+1) = f_i(x_i(k), u_i(k)) + \omega_i(k), \tag{3.1}$$

where $f_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$, $x_i(k) \in \mathbb{R}^{n_x}$ is the system state; $u_i(k) \in \mathbb{R}^{n_u}$ is the control input; $\omega_i(k) \in \mathbb{R}^{n_x}$ is the external disturbance. Here, the control input $u_i(k)$, the state $x_i(k)$, and the additive disturbance $\omega_i(k)$ are constrained by the following

compact sets:

$$u_i(k) \in \mathbb{U}_i \subset \mathbb{R}^{n_u},\ x_i(k) \in \mathbb{X}_i \subset \mathbb{R}^{n_x},\ \omega_i(k) \in \mathbb{W}_i \subset \mathbb{R}^{n_x}, \tag{3.2}$$

with $\mathbb{U}_i$ and $\mathbb{X}_i$ containing the origin. In addition, $\rho \triangleq \sup_{\omega_i(k)\in\mathbb{W}_i} \|\omega_i(k)\|$ is defined as the upper bound of external disturbances.

For agent i, $i \in \mathscr{M}$, the nominal system of the system (3.1) is written as

$$\widehat{x}_i(k+1) = f_i(\widehat{x}_i(k), u_i(k)), \tag{3.3}$$

where $\widehat{x}_i(k)$ is the nominal state satisfying the state constraint $\widehat{x}_i(k) \in \mathbb{X}_i \subset \mathbb{R}^{n_x}$.

Assumption 4. *For the system in (3.1), the following Lipschitz condition holds for all x_i, $z_i \in \mathbb{X}_i$ and $u_i \in \mathbb{U}_i$:*

$$\|f_i(x_i, u_i) - f_i(z_i, u_i)\| \leq L_{fi}\left(\|x_i - z_i\|\right), \tag{3.4}$$

where $L_{fi} > 0$ is the Lipschitz constant.

Assumption 5. *For the nominal system (3.3), assume that:*

- *The point $(0,0)$ is the equilibrium of the system, i.e., $f_i(0,0) = 0$, and $f_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ can be linearized at $(0,0)$, and the linearized system is described as:*

$$\widehat{x}_i(k+1) = A\widehat{x}_i(k) + Bu_i(k), \tag{3.5}$$

  where $A = \partial f_i/\partial x_i|_{(0,0)}$ and $B = \partial f_i/\partial u_i|_{(0,0)}$.

- *For the linearized nominal system (3.5), there exists a state feedback control law $u_i(k) = K_i\widehat{x}_i(k)$ to make $A_{Ki} \triangleq A + BK_i$ stable.*

Define the Lyapunov function $V_i(\widehat{x}_i(k)) \triangleq \|\widehat{x}_i(k)\|^2_{P_i}$ and the terminal set $\widehat{\Omega}_i(\epsilon_i) \triangleq \left\{x_i | x_i^{\mathrm{T}} P_i x_i \leq \epsilon_i^2\right\}$, where $P_i \succ 0$ is the terminal penalty matrix.

Assumption 6. *For the system in (3.3), suppose Assumption 5 holds. Define* $Q_i^* \triangleq Q_i + K_i^T R_i K_i$, *where* $Q_i \succ 0$ *and* $R_i \succeq 0$ *are two predesigned matrices with appropriate dimensions. There exist a constant* $\epsilon_i > 0$, *such that when* $\widehat{x}_i \in \widehat{\Omega}_i(\epsilon_i)$.

- $V_i(\widehat{x}_i) = \widehat{x}_i^T P_i \widehat{x}_i$ *is chosen as the Lyapunov function to the system* $\widehat{x}_i(k+1) = f_i(\widehat{x}_i, K_i\widehat{x}_i)$ *and satisfies* $V_i\left(\widehat{x}_i(k+1)\right) - V_i(\widehat{x}_i(k)) \leq -\|\widehat{x}_i(k)\|^2_{Q_i^*}$;
- *The candidate input* $u_i = K_i\widehat{x}_i$ *satisfies* $K_i\widehat{x}_i \in \mathbb{U}_i$.

Note that the aforementioned assumptions are general and widely used in the related literature on MPC, e.g., [50, 53].

3.2.2 DoS Attacks

Due to the vulnerability of the communication channel, DoS attacks can interfere all the channels in CPS. Specifically, attacks in this work are set in an intermittent or random manner. When the DoS attack is in effect at any time instant, it will block the agent from broadcasting the predicted state sequence to its neigbors. Let $\mathcal{T}_{ij}^a \triangleq \left\{k_{ij,l}^a\right\}$ and $\mathcal{D}_{ij}^a \triangleq \left\{d_{ij,l}^a\right\}$ denote all the launching time instants and their corresponding duration of the DoS attacks on the channel from agent i to agent j, respectively. Here, l denotes the lth launching. We define

$$\Xi_{ij}(k_0, k_1) \triangleq \bigcup_{l \in \mathbb{N}} \mathbb{N}_{[k_{ij,l}^a,\, k_{ij,l}^a + d_{ij,l}^a)} \tag{3.6a}$$

$$\Theta_{ij}(k_0, k_1) \triangleq \mathbb{N}_{(k_0,\, k_1)} \setminus \Xi_{ij}(0,\, \infty) \tag{3.6b}$$

to represent the total activation time periods of DoS attacks, and the overall successful transmission time periods from i to j in the time interval $[k_0,\, k_1]$, respectively. The

launching time instants and their corresponding duration satisfy $k^a_{ij,l} > k_0$, $k^a_{ij,l}+d^a_{ij,l} \leq k_1$, where $k_0 \in \mathbb{N}$, $k_1 \in \mathbb{N}^+$, and $k_1 > k_0$.

Consequently, the effect of DoS attacks on the information transmission from i to j can be described as

$$\Phi_{ij,\Xi}(k) = \begin{cases} 1, & k \in \Xi_{ij}(k_0,\, k_1) \\ 0, & k \in \Theta_{ij}(k_0,\, k_1) \end{cases}, \tag{3.7}$$

where $\Phi_{ij,\Xi} = 1$ represents that the information transmission from i to j is blocked, and $\Phi_{ij,\Xi} = 0$ indicates a successful transmission. We recall the following assumption from [10] to characterize DoS attacks within finite time horizon.

Assumption 7. *With the DoS attacks activation time being set as (3.6a), there exist constants $\alpha \geq 0$ and $\gamma \in (0, 1)$, such that for all $k_0 \geq 0$ and $k_1 \geq k_0$,*

$$|\Xi_{ij}(k_0, k_1)| = \sum_{k=k_0}^{k_1} \Phi_{ij,\Xi}(k) \leq \alpha + \gamma(k_1 - k_0), \tag{3.8}$$

where $|\Xi_{ij}(k_0, k_1)|$ represents all the activation time instants of DoS attacks between the time interval $[k_0,\, k_1]$.

Remark 3. *Note that γ in (3.8) is defined as $\lim\limits_{k_1\to\infty} \frac{|\Xi_{ij}(k_0,k_1)|}{k_1-k_0}$, which depicts the ratio of the total attack duration in considered time intervals. As a result, the upper bound of the duration of DoS attacks can be derived as $N_a \triangleq \lceil \alpha/(1-\gamma) \rceil$.*

Remark 4. *Different from the attacks that directly launch on the agents [14], we consider attacks on the arbitrary communication channels among agents. In general, attacks on the agents block the information receiving and broadcasting channels simultaneously, which consequently can be seen as a special case of the problem formulated in our chapter. An illustrative example with three agents is given in the following for*

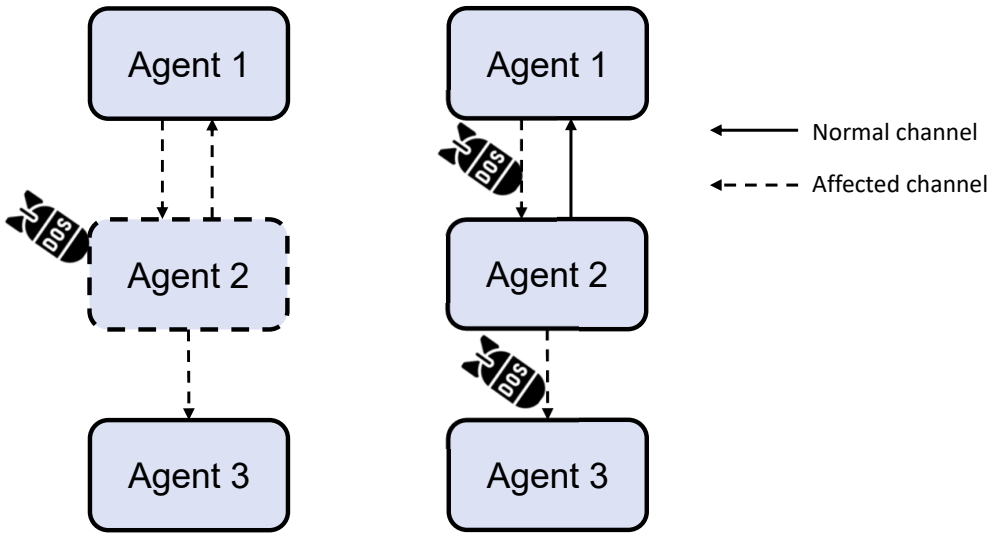

Figure 3.1: DoS attacks occuring on agents (left) and communication channels among agents (right).

the explanation. For the left figure in Figure 3.1, based on the attack policy in [14], assume that agent 2 *is attacked. Then it cannot receive information from agent* 1*, and cannot broadcast information to agent* 1 *and agent* 3*. This situation can be equivalently represented by the proposed formulation, in which all three communication channels are suffering DoS attacks, as shown in the right figure in Figure 3.1. As a result, the attack policy in [14] can be considered as a special case of the proposed formulation.*

3.2.3 Control Objectives

Consider a multi-agent CPS consisting of a group of agents whose model can be described in (3.1) and their connections are characterized as a directed graph. In this work, we aim to develop a robust and resilient distributed MPC strategy such that even under unknown disturbances and DoS attacks on the channels among agents, the states of all agents can be steered to a small region around the equilibrium in a

cooperative manner.

3.3 Robust and Resilient Distributed MPC against DoS Attacks

In this section, a robust and resilient distributed MPC strategy is proposed, as illustrated in Figure 3.2. Firstly, the distributed MPC algorithm is introduced. By imposing a new robustness constraint to the MPC optimization problem, the effect of external disturbances is predicted to be confined within the tightened state constraint. Thus, the robustness of the MPC is enhanced. Furthermore, based on the optimal control sequence and state sequence from the MPC algorithm, a lengthened sequence transmission strategy is proposed to mitigate the effect caused by DoS attacks. Finally, a dual-mode scheme is proposed to reduce the unnecessary computational burden.

3.3.1 Distributed MPC with Robustness Constraint

To handle the external disturbances, we design the robustness constraint for the optimization problem of each agent. The robustness constraint is developed to shrink the state of the nominal system step by step in a predesigned manner to counter the effect caused by external disturbances. Specifically, the robustness constraint in this work is designed based on both state constraint and terminal constraint for the purpose of enlarging the region of attraction compared to [33]. Having added this

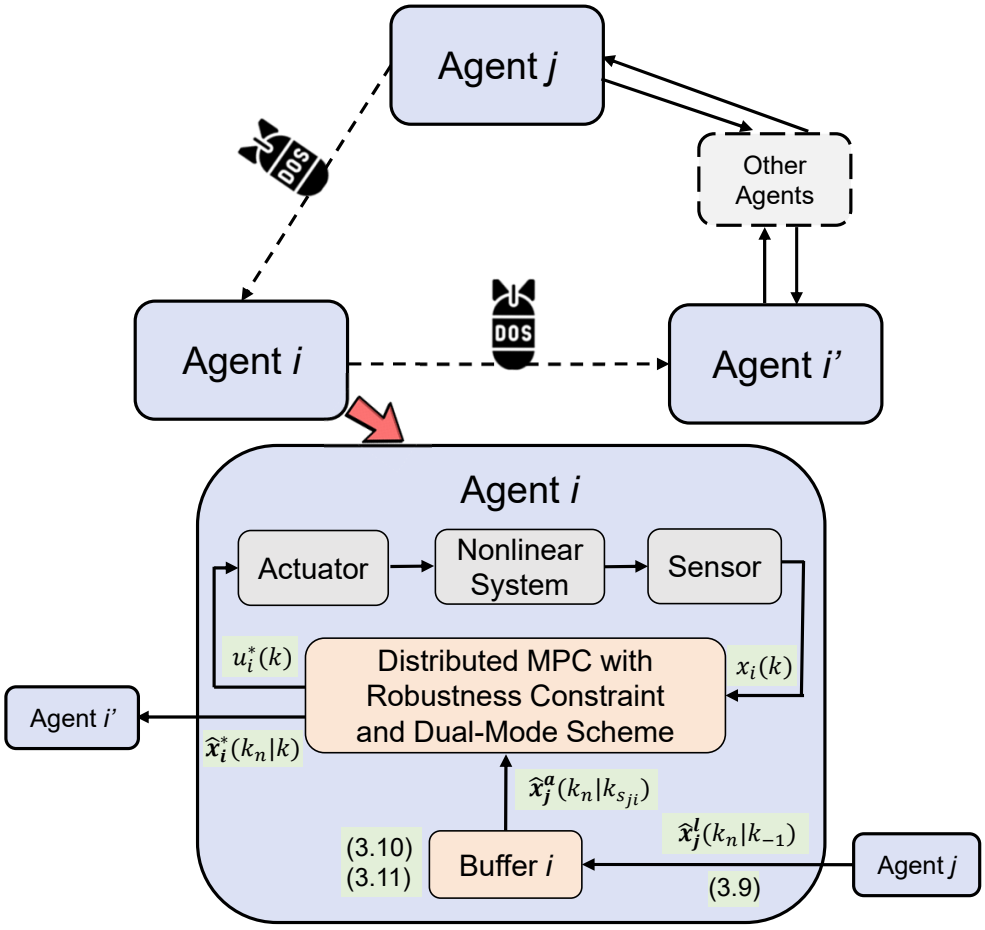

Figure 3.2: Control framework of an multi-agent CPS at time instant k (focusing on Agent i).

constraint, we formulate the optimization problem $\mathcal{P}_i$ as:

$$
\begin{aligned}
\min_{\boldsymbol{u}_i(k_n|k)} \quad & \left\{ J_i \left(\widehat{x}_i(k_n|k), u_i(k_n|k), \widehat{x}^a_{-i}(k_n|k) \right) \right\} \\
\text{s.t.} \quad & \widehat{x}_i(k|k) = x_i(k), \\
& \widehat{x}_i(k_{n+1}|k) = f_i \left(\widehat{x}_i(k_n|k), u_i(k_n|k) \right), \\
& u_i(k_n|k) \in \mathbb{U}_i, \\
& \|\widehat{x}_i(k_n|k)\| \leq \left(1 - \frac{n}{N_p} \zeta_i \right) c_i, \\
& \|\widehat{x}_i(k_{N_p}|k)\|_{P_i} \leq \xi_i \epsilon_i, \quad n = 0, 1, ..., N_p - 1,
\end{aligned}
$$

where k_n denotes $k+n$, k_{N_p} denotes $k+N_p$, in which N_p is the prediction horizon. $\widehat{\boldsymbol{x}}_i(k_n|k)$ is defined as the state sequence of the predicted nominal system state, which is generated through the second constraint with the control input sequence $\boldsymbol{u}_i(k_n|k)$. $\widehat{x}^a_{-i}(k_n|k)$ is the collection of the assumed state of agent i's neighbors. The third constraint is the input constraint. The last two constraints are the robustness constraint, where ζ_i, ξ_i are scaling parameters for the tightened state constraint and terminal constraint, respectively. In addition, the positive constant $c_i = \arg\max_{c_i} \{c_i \in \mathbb{R} : \mathbb{B}_i(c_i) \subseteq \mathbb{X}_i\}$, where $\mathbb{B}_i(c_i) \triangleq \left\{\widehat{x}_i \middle| \|\widehat{x}_i\| \leq c_i\right\}$. By solving the optimization problem at time instant k, we can obtain the optimal input and state sequences as $\boldsymbol{u}^*_i(k_n|k) \triangleq \{u^*_i(k|k), u^*_i(k_1|k), ...u^*_i(k_{N_p-1}|k)\}$ and $\widehat{\boldsymbol{x}}^*_i(k_n|k) \triangleq \{\widehat{x}^*_i(k|k), \widehat{x}^*_i(k_1|k), ...\widehat{x}^*_i(k_{N_p}|k)\}$, respectively.

Here, the objective function is defined as $J_i\left(\widehat{x}_i(k), u_i(k), \widehat{x}^a_{-i}(k)\right) = \sum_{n=0}^{N_p-1}$ $\left\{\|\widehat{x}_i(k_n|k)\|^2_{Q_i} + \|u_i(k_n|k)\|^2_{R_i} + \sum_{j\in\mathcal{N}_i} \|\widehat{x}_i(k_n|k) - \widehat{x}^a_j(k_n|k)\|^2_{Q_{ji}}\right\} + \|\widehat{x}_i(k_{N_p}|k)\|^2_{P_i}$, where $\widehat{x}^a_j(k_n|k)$ is the assumed state sent from agent j, $j \in \mathcal{N}_i$, which will be discussed in 3.3.2. $Q_i \succ 0$ and $R_i \succeq 0$ are the weighting matrices, and $Q_{ji} \succ 0$ is the cooperation matrix.

3.3.2 Lengthened Sequence Transmission Strategy

To tackle the randomly occurring DoS attacks on the communication channels among agents, we design a lengthened sequence transmission strategy. Considering three agents in a multi-agent system, i, j, and i'. Agent i needs to receive information from agent j and agent i' needs to receive information from agent i at each time instant. Considering the attacks on the j to i channel at the time instant k, agent j generates

lengthened state and control input sequences using the following strategy:

$$\boldsymbol{u}_j^l(k_n|k) = \begin{cases} u_j^*(k_n|k) & \text{if } n \in \mathbb{N}_{[0,N_p-1]}, \\ K_i\widehat{x}_i^l(k_n|k) & \text{if } n \in \mathbb{N}_{[N_p,N_p+N_a-1]} \end{cases}$$
$$\widehat{x}_j^l(k_{n+1}|k) = f_j\left(\widehat{x}_j^l(k_n|k), u_j^l(k_n|k)\right),\ n \in \mathbb{N}_{[0,N_p+N_a-1]} \tag{3.9}$$

where $\boldsymbol{u}_j^l(k_n|k)$ and $\widehat{\boldsymbol{x}}_j^l(k_n|k)$ are the lengthened control input and state sequences, N_a is the upper of the duration of the DoS attacks defined in *Remark* 3, and $\widehat{x}_j^l(k|k) = x_j(k)$.

The lengthened sequence to be sent from agent j to agent i depends on whether the channel from j to i is attacked, which is explained in the following. Before showing details, we firstly define $k_{s_{ji}}$ as the latest successful transmission instant from agent j to agent i. Then the following two cases are considered:

- When the communication channel from j to i is not attacked at the time instant $k-1$, agent i can receive the information generated from agent j, and save this sequence in the buffer. In this case, at time instant k, the MPC controller in agent i directly utilizes the state sequence at the time instant $k-1$ with

$$\widehat{\boldsymbol{x}}_j^a(k_n|k) = \widehat{\boldsymbol{x}}_j^l(k_n|k_{-1}),\ n \in \mathbb{N}_{[0,N_p-1]}. \tag{3.10}$$

  After employing the sequence in the optimization problem $\mathcal{P}_i$, update $k_{s_{ji}} = k-1$. Having done this, the optimization problem can be solved to generate two sequences $\boldsymbol{u}_i^*(k_n|k)$ and $\widehat{\boldsymbol{x}}_i^*(k_n|k)$. Finally, agent i lengthens the state and control input sequences with the same step as (3.9), and sends the lengthened state sequence to agent i'.

- When the communication channel from j to i is being attacked at the time

instant $k-1$, agent i cannot receive the information generated from agent j. In this case, select part of the state sequence saved at the time instant $k_{s_{ji}}$ in the buffer with

$$\widehat{\boldsymbol{x}}_j^a(k_n|k) = \widehat{\boldsymbol{x}}_j^l(k_n|k_{s_{ji}}),\ n \in \mathbb{N}_{[0,N_p-1]}. \tag{3.11}$$

Then, the MPC controller adopts $\widehat{x}_j^a(k_n|k)$ as the neighbor's state sequence and generates two optimal sequences $\boldsymbol{u}_{\boldsymbol{i}}^{\boldsymbol{*}}(k_n|k)$ and $\widehat{\boldsymbol{x}}_{\boldsymbol{i}}^{\boldsymbol{*}}(k_n|k)$. Finally, similar to the first case, after lengthening the state and control input sequences by following (3.9), agent i broadcasts the lengthened state sequence to agent i'.

Remark 5. *Note that there is a special case when $k = 0$, the strategy above is not applicable since the sequence $\widehat{\boldsymbol{x}}_i^a(k_n|k)$, $i \in \mathscr{M}$, does not exist. In this regard, we set $\widehat{\boldsymbol{x}}_i^a(n|0)$ to be an all-zeros sequence.*

Remark 6. *Under the lengthened sequence transmission strategy, the successful data transmission time instant $k_{s_{ji}}$ is updated only when the communication channel is not attacked at that time instant. As a result, at time instant k, each agent in the CPS can determine whether DoS attacks have occurred on their information-receiving channels by comparing the values of $k_{s_{ji}}$ and $k-1$. If $k_{s_{ji}}$ does not match the last time instant, it indicates that DoS attacks have affected the communication channel.*

3.3.3 Dual-Mode Control Framework

Dual-mode control has been widely applied in a variety of the MPC schemes; see, e.g., [18, 33, 44]. It is known that it can help reduce the computational burden, because when the states of all the agents enter the terminal region at the time instant k_o:

$$\|\widehat{x}_i(k_o|k_o)\|_{P_i}^2 \leq {\epsilon_i}^2, \tag{3.12}$$

the control scheme is changed to the state-feedback control law

$$u_i(k_o) = K_i x_i(k_o), \tag{3.13}$$

rather than solving Problem $\mathcal{P}_i$. Furthermore, when the channel is suffering from DoS attacks, the dual-mode control can help enhance the resilience against attacks, since the state-feedback control law does not require the information from neighbor agents. In addition, we assume that there exists a detection mechanism for each agent, such that each agent can know whether the states of other agents enter the terminal region. As a result, the attacks launching on the communication channels will not affect the control input generation.

Based on the previous discussions, the proposed robust and resilient distributed MPC strategy will be implemented in a dual-mode control manner, which is summarized in Algorithm 2.

3.4 Theoretical Analysis

This section shows the proof for the recursive feasibility of the formulated optimization problem and the closed-loop stability of the multi-agent CPS by applying the proposed resilient and robust distributed MPC approach.

For agent i, $i \in \mathscr{M}$, construct a candidate control sequence at the time instant $k+1$

$$\widetilde{\boldsymbol{u}}_i(k_n|k_1) \triangleq \left\{\widetilde{u}_i(k_1|k_1),\, \widetilde{u}_i(k_2|k_1), ...,\, \widetilde{u}_i(k_{N_p}|k_1)\right\},$$

Algorithm 2 Robust and resilient distributed MPC algorithm

Require: For agent i, $i \in \mathscr{M}$, the weighting matrices Q_i, Q_{ji}, R_i; the state-feedback control gain K_i; the terminal penalty matrix P_i; the prediction horizon N_p; the terminal set level ϵ_i; scaling parameters ξ_i and ζ_i; the initial state $x_i(0)$; the upper bound of the duration of DoS attacks N_a. Set $k = 0$, and $k_{s_{ji}} = 0$.

while the control action is not stopped **do**
  For all agents, sample the system states.
  if (3.12) is not satisfied for all the agents **then**
    for agent i, $i \in \mathscr{M}$ **do**
      if the communication channels from j to i, $j \in \mathscr{N}_i$, is not being attacked **then**
        Receive state sequence $\widehat{\boldsymbol{x}}_j^l(k_n|k_{-1})$ from its neighbors j.
        Save the state sequence $\widehat{\boldsymbol{x}}_j^l(k_n|k_{-1})$ in the buffer and update $k_{s_{ji}} = k-1$.
      else
        Adopt the sequence $\widehat{\boldsymbol{x}}_j^l(k_n|k_{s_{ji}})$ saved in the buffer.
      end if
    end for
    Construct the state sequence by following (3.10), (3.11), and send it to the MPC controller.
    Solve the optimization problem $\mathcal{P}_i$ to generate the sequence $\boldsymbol{u}_i^*(k_n|k)$ and $\widehat{\boldsymbol{x}}_i^*(k_n|k)$.
    Apply $u_i(k) = u_i^*(k|k)$ to agent i.
    Construct the lengthened state and control input sequences by applying (3.9), and then broadcast the state sequence to its neighbors.
  else
    for agent i, $i \in \mathscr{M}$ **do**
      Construct the control input by applying (3.13).
    end for
  end if
  $k = k + 1$;
end while

and its corresponding state sequence

$$\widetilde{\boldsymbol{x}}_i(k_n|k_1) \triangleq \left\{\widetilde{x}_i(k_1|k_1),\, \widetilde{x}_i(k_2|k_1), ...,\, \widetilde{x}_i(k_{N_p+1}|k_1)\right\},$$

where the candidate control sequence $\widetilde{\boldsymbol{u}}_i(k_{n+1}|k_1),\ n \in \mathbb{N}_{[0,\,N_p-1]}$ can be represented as

$$\widetilde{\boldsymbol{u}}_i(k_{n+1}|k_1) = \begin{cases} u_i^*(k_{n+1}|k) & \text{if } n \in \mathbb{N}_{[0,N_p-2]} \\ K_i\widehat{x}_i^*(k_{N_p}|k) & \text{if}\, n = N_p - 1 \end{cases}, \tag{3.14}$$

and the predicted state sequence $\widetilde{\boldsymbol{x}}_i(k_{n+1}|k_1)$ are constructed according to the nominal system dynamics

$$\widetilde{x}_i(k_{n+1}|k_1) = f_i\left(\widetilde{x}_i(k_n|k_1), \widetilde{u}_i(k_n|k_1)\right),\ n \in \mathbb{N}_{[0,N_p-1]}. \tag{3.15}$$

For the rest of this section, we will prove that the candidate input sequence and its corresponding predicted state sequence can be the feasible solution of optimization problem $\mathcal{P}_i$ under certain conditions, and the multi-agent CPS is stable based on the feasibility analysis.

3.4.1 Recursive Feasibility

In this subsection, we conduct the feasibility analysis of the formulated optimization problem and derive the sufficient conditions of ensuring the recursive feasibility. Before proceeding, we present the following assumption and lemma that will be used to establish the main results.

Assumption 8. *Assume that there exists an initially feasible region $\mathbb{X}_N \subseteq \mathbb{X}_i$, such that for all the initial state $x_0 \in \mathbb{X}_N$, the optimization problem in (3.9) admits a*

feasible solution when its initial value is set as x_0.

To guarantee recursive feasibility, essentially it suffices to prove that the predicted state sequences generated at the time instant $k+1$ under the candidate control input sequence is feasible. More specifically, it does need the following three requirements to be fulfilled.

(R1) The predicted state at the time instant $k+N_p$ satisfies the terminal constraint:

$$\|\widetilde{x}_i(k_{N_p}|k_1)\|_{P_i} \leq \epsilon_i. \tag{3.16}$$

(R2) The predicted state at the time instant $k+N_p+1$ satisfies the tightened terminal constraint:

$$\|\widetilde{x}_i(k_{N_p+1}|k_1)\|_{P_i} \leq \xi_i \epsilon_i. \tag{3.17}$$

(R3) The predicted state satisfies the tightened state constraint:

$$\|\widetilde{x}_i(k_{n+1}|k_1)\| \leq \left(1 - \frac{n}{N_p}\zeta_i\right) c_i. \tag{3.18}$$

Lemma 1. *For agent* $i \in \mathscr{M}$*, with the system dynamics in (3.1), suppose Assumptions 4 and 5 hold, and optimization problem* $\mathcal{P}_i$ *has a feasible solution at the time instant* k*, then* $\|\widetilde{x}_i(k_n|k_1) - \widehat{x}_i^*(k_n|k)\| \leq {L_{fi}}^{n-1}\rho$, $n \in \mathbb{N}_{[1,N_p]}$.

Proof. Recall that at the time instant k, the optimal input sequence is denoted as $\boldsymbol{u}_i^*(k_n|k)$, $n \in \mathbb{N}_{[0,\,N_p-1]}$ and the optimal state sequence is denoted as $\widehat{\boldsymbol{x}}_i^*(k)$, $n \in \mathbb{N}_{[0,\,N_p]}$. The upper bound of the difference between the optimal state and the actual

state can be derived as:

$$\begin{aligned}
&\|\widehat{x}_i^*(k_n|k_{n-1}) - x_i(k_n)\| \\
=&\|f_i\left(\widehat{x}_i^*(k_{n-1}|k_{n-1}), u_i(k_{n-1})\right) \\
&- f_i\left((x_i(k_{n-1}), u_i(k_{n-1})\right) - w_i(k_{n-1})\| \\
\leq&\rho,
\end{aligned} \tag{3.19}$$

where $n \in \mathbb{N}_{[1,N_p]}$.

Similarly, the difference between the nominal state and the optimal sequence can be derived as

$$\begin{aligned}
&\|\widetilde{x}_i(k_n|k_1) - \widehat{x}_i^*(k_n|k)\| \\
=&\|f_i\left(\widetilde{x}_i(k_{n-1}|k_1), \widetilde{u}_i(k_{n-1}|k_1)\right) \\
&- f_i\left(\widehat{x}_i^*(k_{n-1}|k), u_i^*(k_{n-1}|k)\right)\| \\
\overset{(3.4)}{\leq}& L_{fi}\|\widetilde{x}_i(k_{n-1}|k_1) - \widehat{x}_i^*(k_{n-1}|k)\| \\
\leq&{L_{fi}}^{n-1}\|\widetilde{x}_i(k_1|k_1) - \widehat{x}_i^*(k_1|k)\| \\
=&{L_{fi}}^{n-1}\|x_i(k_1) - \widehat{x}_i^*(k_1|k)\| \\
\overset{(3.19)}{\leq}& {L_{fi}}^{n-1}\rho,
\end{aligned} \tag{3.20}$$

where $n \in \mathbb{N}_{[1,N_p]}$. □

After deriving the difference between the nominal state and the optimal state at the same predicted time instant, we can obtain the following theorem to prove the recursive feasibility of the optimization problem $\mathcal{P}_i$.

Theorem 1. *For agent $i \in \mathscr{M}$, suppose Assumptions 4, 5, 6, and 8 hold, and the optimization problem $\mathcal{P}_i$ is feasible at the time instant k, Algorithm 2 is also feasible*

if the following conditions are satisfied:

$$\frac{\lambda_{max}(P_i^{\frac{1}{2}})}{\lambda_{max}(P_i^{\frac{1}{2}}) + \lambda_{min}(Q_i^{*\frac{1}{2}})} \leq \xi_i \leq 1 - \frac{\rho\lambda_{max}(P_i^{\frac{1}{2}}){L_{fi}}^{N_p-1}}{\epsilon_i} \tag{3.21a}$$

$$\zeta_i \geq \frac{N_p\rho{L_{fi}}^{N_p-1}}{c_i} \tag{3.21b}$$

where ξ_i and ζ_i are positive constants for agent $i \in \mathscr{M}$.

Proof. To complete the proof, we need to show that the candidate input sequence (3.14) and the corresponding predicted state sequence (3.15) satisfy the input constraint and conditions (R1)-(R3) when ξ_i and ζ_i satisfy the conditions above, which is explained in the following.

Before showing the details, we firstly demonstrate that the input constraint is satisfied. As shown in (3.14), the candidate input sequence $\widetilde{\boldsymbol{u}}_i(k_{n+1}|k_1)$ is constructed based on the optimal control input sequence $\boldsymbol{u}_i^*(k_n|k)$ and the feedback control law $K_i\widehat{x}_i^*(k_{N_p}|k)$. According to *Assumption 6*, the candidate input sequence always satisfies the input constraint.

(R1) At the time instant $k+1$, we need to prove that (3.16) is satisfied. By applying (3.20), one has

$$\|\widetilde{x}_i(k_{N_p}|k_1) - \widehat{x}_i^*(k_{N_p}|k)\|_{P_i} \leq \lambda_{\max}(P_i^{\frac{1}{2}}){L_{fi}}^{N_p-1}\rho. \tag{3.22}$$

By recalling the tightened terminal constraint at time instant k, $\|\widehat{x}_i^*(k_{N_p}|k)\|_{P_i} \leq \xi_i\epsilon_i$, and applying the triangle inequality, we can derive

$$\begin{aligned} &\|\widetilde{x}_i(k_{N_p}|k_1) - \widehat{x}_i^*(k_{N_p}|k)\|_{P_i} + \|\widehat{x}_i^*(k_{N_p}|k)\|_{P_i} \\ &\overset{(3.22)}{\leq} \lambda_{\max}(P_i^{\frac{1}{2}}){L_{fi}}^{N_p-1}\rho + \xi_i\epsilon_i. \end{aligned} \tag{3.23}$$

Since ξ_i satisfies (3.21a), it suffices to impose that

$$\lambda_{\max}(P_i^{\frac{1}{2}}){L_{fi}}^{N_p-1}\rho + \xi_i\epsilon_i \leq \epsilon_i.$$

Then it can be derived that (3.16) is satisfied.

(R2) We also need to ensure that (3.17) holds at the time instant $k+1$. Recalling the control input sequence constructed as (3.14), according to *Assumption 6*, we have:

$$\|\widetilde{x}_i(k_{N_p+1}|k_1)\|_{P_i} \leq \|\widetilde{x}_i(k_{N_p}|k_1)\|_{P_i} - \|\widetilde{x}_i(k_{N_p}|k_1)\|_{Q_i^*}.$$

With (3.21a) being met, the following inequality holds:

$$\max\left\{\|\widetilde{x}_i(k_{N_p}|k_1)\|_{P_i} - \|\widetilde{x}_i(k_{N_p}|k_1)\|_{Q_i^*}\right\} \leq \xi_i\epsilon_i,$$

which is the equivalent requirement of (3.17).

(R3) Finally, at the time instant $k+1$, we need to prove that (3.18) is satisfied. By applying (3.20), it can be obtained that

$$\|\widetilde{x}_i(k_n|k_1)\| \leq \|\widehat{x}_i^*(k_n|k)\| + {L_{fi}}^{n-1}\rho,$$

where $n \in \mathbb{N}_{[1,N_p]}$. Due to the fact that ζ_i satisfies (3.21b), the following inequality is held:

$$\|\widetilde{x}_i(k_n|k_1)\| \leq \|\widehat{x}_i^*(k_n|k)\| + {L_{fi}}^{n-1}\rho \leq \left(1 - \frac{n-1}{N_p}\zeta_i\right)c_i.$$

Thus, the requirement (R3) is met.

In summary, according to *Theorem* 1, if the given conditions are satisfied, *Algorithm* 2 is recursively feasible. □

3.4.2 Stability Analysis

As discussed in *Theorem* 1, the recursive feasibility of the formulated optimization problem can be guaranteed if a set of conditions can be satisfied. In this subsection, we concentrate on the closed-loop stability analysis of the multi-agent CPS by applying the proposed resilient and robust MPC strategy as illustrated in *Algorithm* 2.

Theorem 2. *For the multi-agent CPS (3.1) using Algorithm 1 under the conditions of Theorem 2 with Assumptions 4-8 held. Given a constant* β_i *with* $\frac{\lambda_{max}(P_i^{\frac{1}{2}})}{\lambda_{max}(P_i^{\frac{1}{2}})+\lambda_{min}(Q_i^{*\frac{1}{2}})} \geq \beta_i \geq \frac{2\lambda_{max}(P_i)^{\frac{3}{2}}\rho}{\epsilon_i \lambda_{min}(Q_i^*)}$*, if the cooperation matrices* Q_{ji} *satisfies*

$$\sum_{j\in\mathcal{N}_i} \lambda_{max}(Q_{ji}) < \frac{\epsilon_i^2 \frac{\lambda_{min}(Q_i)}{\lambda_{max}(P_i)} - \mathcal{B}_i}{c_j}, \tag{3.24}$$

where $\mathcal{B}_i$ *and* c_j *are defined as:*

$$\begin{aligned}
\mathcal{B}_i \triangleq & \sum_{n=1}^{N_p-1} \Big[\big({L_{fi}}^{n-1} \rho \lambda_{max}(Q_i^{\frac{1}{2}}) + 2\left(1 - \frac{n}{N_p}\zeta_i\right) c_i \big) \\
& \times \left({L_{fi}}^{n-1} \rho \lambda_{max}(Q_i^{\frac{1}{2}}) \right) \Big], \\
c_j \triangleq & \sum_{n=0}^{-k_S+N_p-1} \sum_{j\in\mathcal{N}_i} \Big[\big(c_i + c_j - \frac{n+1}{N_p}(\zeta_j c_j + \zeta_i c_i) \\
& - \frac{k_S}{N_p}\zeta_j c_j \big) + {L_{fi}}^n \rho \Big]^2 \\
& + \sum_{n=-k_S+N_p}^{N_p-1} \sum_{j\in\mathcal{N}_i} \Big[\frac{\xi_j \epsilon_j}{\lambda_{min}(P_j^{\frac{1}{2}})} + \left(1 - \frac{n+1}{N_p}\zeta_i\right) c_i + {L_{fi}}^n \rho \Big]^2 \\
& + \sum_{j\in\mathcal{N}_i} \left(\frac{\epsilon_j}{\lambda_{min}(P_j^{\frac{1}{2}})} + \frac{\xi_i \epsilon_i}{\lambda_{min}(P_i^{\frac{1}{2}})} \right)^2 .
\end{aligned}$$

Here, ζ_i*,* ζ_j*,* ξ_i*, and* ξ_j *are the designed parameters that satisfy the conditions in Theorem 1, then the overall system state will converge to the convergence set* $\Omega_1^* \times \Omega_2^* \times \cdots \times \Omega_M^*$*, where* $\Omega_i^* \triangleq \left\{ x_i | {x_i}^T P_i x_i \leq \left(1 + \frac{\lambda_{min}(Q_i^*)}{\lambda_{max}(P_i)}\right) \beta_i {\epsilon_i}^2 \right\}$.

Proof. At the time instant $k+1$, construct the cost function with the candidate input sequence and the predicted state sequence for agent i, $i \in \mathscr{M}$. Then, the difference of the cost function between the two adjacent time instants can be represented as:

$$\begin{aligned}\triangle(J_i) \triangleq\ & J_i\left(\widetilde{x}_i(k_{n+1}|k_1), \widetilde{u}_i(k_{n+1}|k_1), \widetilde{x}^a_{-i}(k_{n+1}|k_1)\right) \\ & - J_i\left(\widehat{x}^*_i(k_n|k), u^*_i(k_n|k), \widehat{x}^a_{-i}(k_n|k)\right),\ n \in \mathbb{N}_{[0,\, N_p-1]}.\end{aligned}$$

We then split $\triangle(J_i)$ with three time intervals:

$$\begin{aligned}\triangle(J_i) =& \mathcal{T}_1 + \mathcal{T}_2 + \mathcal{T}_3 \\ =& \sum_{n=1}^{N_p-1} \big\{\|\widetilde{x}_i(k_n|k_1)\|^2_{Q_i} + \|\widetilde{u}_i(k_n|k_1)\|^2_{R_i} \\ & - \|\widehat{x}^*_i(k_n|k)\|^2_{Q_i} - \|u^*_i(k_n|k)\|^2_{R_i}\big\} \\ & + \|\widetilde{x}_i(k_{N_p}|k_1)\|^2_{Q_i} + \|\widetilde{u}_i(k_{N_p}|k_1)\|^2_{R_i} \\ & - \|\widehat{x}^*_i(k|k)\|^2_{Q_i} - \|u^*_i(k|k)\|^2_{R_i} \\ & + \|\widetilde{x}_i(k_{N_p+1}|k_1)\|^2_{P_i} - \|\widehat{x}^*_i(k_{N_p}|k)\|^2_{P_i} \\ & + \sum_{n=0}^{N_p-1} \sum_{j \in \mathcal{N}_i} \|\widetilde{x}_i(k_{n+1}|k_1) - \widehat{x}^a_{-i}(k_{n+1}|k_1)\|^2_{Q_{ji}} \\ & - \sum_{n=0}^{N_p-1} \sum_{j \in \mathcal{N}_i} \|\widetilde{x}_i(k_n|k) - \widehat{x}^a_{-i}(k_n|k)\|^2_{Q_{ji}}.\end{aligned}$$

On the right-hand side of this equation, the first part can be bounded as

$$\begin{aligned}
\mathcal{T}_1 = & \sum_{n=1}^{N_p-1} \left\{ \|\widetilde{x}_i(k_n|k_1)\|_{Q_i}^2 + \|\widetilde{u}_i(k_n|k_1)\|_{R_i}^2 \right. \\
& \left. -\|\widehat{x}_i^*(k_n|k)\|_{Q_i}^2 - \|u_i^*(k_n|k)\|_{R_i}^2 \right\} \\
\overset{(3.14)}{=} & \sum_{n=1}^{N_p-1} \|\widetilde{x}_i(k_n|k_1)\|_{Q_i}^2 - \|\widehat{x}_i^*(k_n|k)\|_{Q_i}^2 \\
\leq & \sum_{n=1}^{N_p-1} (\|\widetilde{x}_i(k_n|k_1)\|_{Q_i} + \|\widehat{x}_i^*(k_n|k)\|_{Q_i}) \\
& \times \|\widetilde{x}_i(k_n|k_1) - \widehat{x}_i^*(k_n|k)\|_{Q_i} \\
\leq & \sum_{n=1}^{N_p-1} \left\{ \left({L_{fi}}^{n-1} \rho \lambda_{\max}(Q_i^{\frac{1}{2}}) + 2 \left(1 - \frac{n}{N_p} \zeta_i \right) c_i \right) \right. \\
& \left. \times \left({L_{fi}}^{n-1} \rho \lambda_{\max}(Q_i^{\frac{1}{2}}) \right) \right\}.
\end{aligned}$$

For the second part

$$\begin{aligned}
\mathcal{T}_2 = & \|\widetilde{x}_i(k_{N_p}|k_1)\|_{Q_i}^2 + \|\widetilde{u}_i(k_{N_p}|k_1)\|_{R_i}^2 \\
& + \|\widetilde{x}_i(k_{N_p+1}|k_1)\|_{P_i}^2 - \|\widehat{x}_i^*(k_{N_p}|k)\|_{P_i}^2,
\end{aligned} \tag{3.25}$$

it is proven that the candidate state at the time instant $k + N_p$ enters the terminal region. According to *Assumption 6*, the following inequality holds true:

$$\begin{aligned}
& \|\widehat{x}_i^*(k_{N_p}|k)\|_{P_i}^2 - \|\widetilde{x}_i(k_{N_p+1}|k_1)\|_{P_i}^2 \\
& \geq \|\widehat{x}_i(k_{N_p})\|_{Q_i^*}^2 = \|\widetilde{x}_i(k_{N_p}|k_1)\|_{Q_i}^2 + \|\widetilde{u}_i(k_{N_p}|k_1)\|_{R_i}^2,
\end{aligned}$$

which means $\mathcal{T}_2 \leq 0$.

The third part can be evaluated as follows:

$$
\begin{aligned}
&\mathcal{T}_3 = \\
&\sum_{n=0}^{N_p-1} \sum_{j\in\mathcal{N}_i} \|\widetilde{x}_i(k_{n+1}|k_1) - \widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}}^2 \\
&- \sum_{n=0}^{N_p-1} \sum_{j\in\mathcal{N}_i} \|\widehat{x}_i(k_n|k) - \widehat{x}_j^a(k_n|k)\|_{Q_{ji}}^2 \\
&\leq \sum_{n=0}^{N_p-1} \sum_{j\in\mathcal{N}_i} \|\widetilde{x}_i(k_{n+1}|k_1) - \widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}}^2 \\
&\leq \sum_{n=0}^{N_p-1} \sum_{j\in\mathcal{N}_i} \left(\|\widetilde{x}_i(k_{n+1}|k_1)\|_{Q_{ji}} + \|\widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}}\right)^2 \\
&= \sum_{n=0}^{-k_S+N_p-1} \sum_{j\in\mathcal{N}_i} \left(\|\widetilde{x}_i(k_{n+1}|k_1)\|_{Q_{ji}} + \|\widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}}\right)^2 \\
&+ \sum_{n=-k_S+N_p}^{N_p-2} \sum_{j\in\mathcal{N}_i} \left(\|\widetilde{x}_i(k_{n+1}|k_1)\|_{Q_{ji}} + \|\widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}}\right)^2 \\
&+ \sum_{j\in\mathcal{N}_i} \left(\|\widetilde{x}_i(k_{N_p+1}|k_1)\|_{Q_{ji}} + \|\widehat{x}_j^a(k_{N_p+1}|k_1)\|_{Q_{ji}}\right)^2,
\end{aligned}
$$

where $k_S = k - k_{s_{ji}}$.

To evaluate the upper bound of $\mathcal{T}_3$, we can derive the upper bound of each element in the polynomial. Considering the term $\|\widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}}$, we have:

$$
\begin{aligned}
&\|\widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}} \overset{(3.10),(3.11)}{=\!=\!=} \|\widehat{x}_j^*(k_{n+1}|k_{s_{ji}})\|_{Q_{ji}} \\
&\leq \left(1 - \frac{k_S+n+1}{N_p}\zeta_j\right) c_j \lambda_{\max}(Q_{ji}^{\frac{1}{2}}),
\end{aligned}
$$

with $n \in \mathbb{N}_{[0,-k_S+N_p-1]}$, and

$$
\|\widehat{x}_j^a(k_{n+1}|k_1)\|_{Q_{ji}} \leq \frac{\lambda_{\max}(Q_{ji}{}^{\frac{1}{2}})}{\lambda_{\min}(P_j{}^{\frac{1}{2}})}\xi_j\epsilon_j.
$$

with $n \in \mathbb{N}_{[-k_S+N_p,N_p-1]}$.

Similarly, considering the term $\|\widetilde{x}_i(k_{n+1}|k_1)\|_{Q_{ji}}$, and recalling (3.20), we can obtain

$$\begin{aligned}\|\widetilde{x}_i(k_{n+1}|k_1)\|_{Q_{ji}} &\leq \left\{\|\widehat{x}_i^*(k_{n+1}|k)\|_{Q_{ji}} + {L_{fi}}^n \rho \lambda_{\max}(Q_{ji}^{\frac{1}{2}})\right\} \\ &\leq \left\{\left(1 - \frac{n+1}{N_p}\zeta_i\right) c_i \lambda_{\max}(Q_{ji}^{\frac{1}{2}}) + {L_{fi}}^n \rho \lambda_{\max}(Q_{ji}^{\frac{1}{2}})\right\},\end{aligned}$$

with $n \in \mathbb{N}_{[0,N_p-2]}$, and

$$\|\widetilde{x}_i(k_{N_p+1}|k_1)\|_{Q_{ji}} \leq \frac{\lambda_{\max}({Q_{ji}}^{\frac{1}{2}})}{\lambda_{\min}({P_i}^{\frac{1}{2}})}\xi_i\epsilon_i.$$

Consequently, the upper bound of $\mathcal{T}_3$ can be formulated as:

$$\begin{aligned}\mathcal{T}_3 \leq & \sum_{n=0}^{-k_S+N_p-1} \sum_{j\in\mathcal{N}_i} \Big\{ {L_{fi}}^n \rho \lambda_{\max}(Q_{ji}^{\frac{1}{2}}) \\ & + \left(c_i + c_j - \frac{n+1}{N_p}(\zeta_j c_j + \zeta_i c_i) - \frac{k_S}{N_p}\zeta_j c_j\right) \lambda_{\max}(Q_{ji}^{\frac{1}{2}}) \Big\}^2 \\ & + \sum_{n=-k_S+N_p}^{N_p-1} \sum_{j\in\mathcal{N}_i} \Big\{ \frac{\lambda_{\max}({Q_{ji}}^{\frac{1}{2}})}{\lambda_{\min}({P_j}^{\frac{1}{2}})}\xi_j\epsilon_j + {L_{fi}}^n \rho \lambda_{\max}(Q_{ji}^{\frac{1}{2}}) \\ & + \left(1 - \frac{n+1}{N_p}\zeta_i\right) c_i \lambda_{\max}(Q_{ji}^{\frac{1}{2}}) \Big\}^2 \\ & + \sum_{j\in\mathcal{N}_i} \left(\frac{\lambda_{\max}({Q_{ji}}^{\frac{1}{2}})}{\lambda_{\min}({P_j}^{\frac{1}{2}})}\epsilon_j + \frac{\lambda_{\max}({Q_{ji}}^{\frac{1}{2}})}{\lambda_{\min}({P_i}^{\frac{1}{2}})}\xi_i\epsilon_i \right)^2 \\ & \triangleq \sum_{j\in\mathcal{N}_i} \lambda_{\max}(Q_{ji}) c_j.\end{aligned}$$

To sum up, we can derive that

$$\begin{aligned}\triangle(J_i) =& \mathcal{T}_1 + \mathcal{T}_2 + \mathcal{T}_3 - \|\widehat{x}_i^*(k|k)\|_{Q_i}^2 - \|u_i^*(k|k)\|_{R_i}^2 \\ \leq& \mathcal{T}_1 + \mathcal{T}_3 - \|\widehat{x}_i^*(k|k)\|_{Q_i}^2.\end{aligned} \tag{3.26}$$

Since $\widehat{x}_i^*(k|k) = x_i(k)$, according to 3.24, the following inequality is satisfied:

$$\mathcal{T}_1 + \mathcal{T}_3 < \epsilon_i^2 \frac{\lambda_{\min}(Q_i)}{\lambda_{\max}(P_i)} \leq \|\widehat{x}_i^*(k|k)\|_{Q_i}^2. \tag{3.27}$$

when $x_i \notin \widehat{\Omega}_i$.

Therefore, it is shown in (3.27) that the state x_i can be steered into the terminal region $\widehat{\Omega}_i$.

Based on the discussion above, if the largest eigenvalues of Q_{ji} satisfy (3.24), the states of all agents will be steered into the terminal region by solving the optimization problem $\mathcal{P}_i$. In the following steps, we can prove that the state of each agent will converge to the region $\Omega_i \triangleq \left\{x_i^{\mathrm{T}} P_i x_i \leq \beta_i {\epsilon_i}^2\right\}$, where $\beta_i \geq \frac{2\lambda_{\max}({P_i}^{\frac{3}{2}})\rho}{\epsilon_i \lambda_{\min}(Q_i^*)}$. Assume that there exist a constant $\eta_i \in (\sqrt{\beta_i}, 1)$ and a region $\Omega_i^{\eta_i} \triangleq \left\{x | {x_i}^{\mathrm{T}} P_i x_i \leq {\eta_i}^2 {\epsilon_i}^2\right\}$.

When the state of agent i has entered $\widehat{\Omega}_i$ but has not entered $\Omega_i^{\eta_i}$, we have:

$$\begin{aligned}
&{x_i}^{\mathrm{T}}(k+1) P_i x_i(k+1) - {x_i}^{\mathrm{T}}(k) P_i x_i(k) \\
\leq& - \|\widehat{x}_i^*(k|k)\|_{Q_i^*}^2 + 2\widehat{x}_i^*(k|k)^{\mathrm{T}} P_i \omega_i(k) \\
\leq& - \frac{\lambda_{\min}(Q_i^*)}{\lambda_{\max}(P_i)} {\eta_i}^2 {\epsilon_i}^2 + 2\epsilon_i \lambda_{\max}({P_i}^{\frac{1}{2}})\rho \\
\leq& \left(-\eta^2 + \beta_i\right) \frac{\lambda_{\min}(Q_i^*)}{\lambda_{\max}(P_i)} \epsilon_i^2.
\end{aligned} \tag{3.28}$$

Since $\eta_i \in (\sqrt{\beta_i}, 1)$, it can be concluded that when $x_i \notin \Omega_i^{\eta_i}$, the difference between two adjacent time instants of the Lyapunov function designed for the system is always negative, which implies that the state of each agent will converge to the region Ω_i in finite time.

Furthermore, we need to prove that the state will not leave Ω_i^*. Let $\mathcal{K}_i$ be the set of all the time instants that the state of agent enters Ω_i, with $\kappa_i \in \mathcal{K}_i$ being an

arbitrary time instant in this set. It can be obtained that:

$$x_i{}^{\mathrm{T}}(\kappa_i)P_i x_i(\kappa_i) \leq \beta_i {\epsilon_i}^2.$$

Suppose that at time instant $\kappa_i + 1$, the state x_i leaves the region Ω_i. Therefore, based on (3.28), the following inequality will hold true:

$$\begin{aligned} x_i{}^{\mathrm{T}}(\kappa_i+1)P_i x_i(\kappa_i+1) &\leq x_i{}^{\mathrm{T}}(\kappa_i)P_i x_i(\kappa_i) + \beta_i \frac{\lambda_{\min}(Q_i^*)}{\lambda_{\max}(P_i)}\epsilon_i^2 \\ &\leq \left(1 + \frac{\lambda_{\min}(Q_i^*)}{\lambda_{\max}(P_i)}\right)\beta_i {\epsilon_i}^2. \end{aligned}$$

which implies $x_i(\kappa_i + 1) \in \Omega_i^*$.

Since $\Omega_i \subset \Omega_i^*$, according to (3.28), the state of agent i will then converge to Ω_i in finite time. Consequently, the state of agent i, $i \in \mathscr{M}$ will be confined in the small region Ω_i^*, which also implies that the overall system state will converge to the convergence set $\Omega_1^* \times \Omega_2^* \times \cdots \times \Omega_M^*$. □

Remark 7. *Inequality (3.24) shows the upper bound of the eigenvalues of matrix Q_{ji}. In fact, the disturbance bound can also influence the selection of Q_{ji}. Specifically, larger eigenvalues of matrix Q_{ji} mean lesser tolerance to the disturbance on the agent i. Therefore, we are motivated to propose another kind of robustness constraint to enhance the robustness against DoS attacks and external disturbances with slight sacrifice to the region of attraction. To achieve this purpose, the tightened state constraint and the tightened terminal constraint can be fused into one constraint, which is represented as $\|\widehat{x}_i(k_n|k)\|_{P_i} \leq \Xi(\xi_i, n)$, where $\Xi(\xi_i, n) \triangleq \left(1 - \frac{n}{N_p}\right) c_i \xi_i \lambda_{min}({P_i}^{\frac{1}{2}}) + \frac{n}{N_p}\xi_i\epsilon_i$, $n \in \mathbb{N}_{[0,\, N_p]}$. In this way, the conditions in Theorem 1 and Theorem 2 can be modified as $\frac{N_p \lambda_{max}({P_i}^{\frac{1}{2}}){L_{fi}}^{N_p-1}\rho}{\xi_i\epsilon_i + \xi_i c_i \lambda_{max}({P_i}^{\frac{1}{2}})} \leq \xi_i \leq \frac{\epsilon_i - \rho\lambda_{max}({P_i}^{\frac{1}{2}}){L_{fi}}^{N_p-1}}{\epsilon_i}$, and $c_j \leq \sum_{j\in\mathcal{N}_i}\left\{\sum_{n=0}^{N_p-1}\left[\Xi(\xi_i,(n+1))\right.\right.$ $\left.\left.+\Xi(\xi_j,(n+1)) + {L_{fj}}^n\rho\right]^2 + \left(\frac{\lambda_{max}(Q_{ji}{}^{\frac{1}{2}})}{\lambda_{min}(P_j{}^{\frac{1}{2}})}\epsilon_j + \frac{\lambda_{max}(Q_{ji}{}^{\frac{1}{2}})}{\lambda_{min}(P_i{}^{\frac{1}{2}})}\xi_i\epsilon_i\right)^2\right\}$, thereby enhancing the*

robustness. In summary, the selection of different types of robustness constraints can be seen as a trade-off between the robustness and the size of the region of attraction.

3.5 Simulation Study

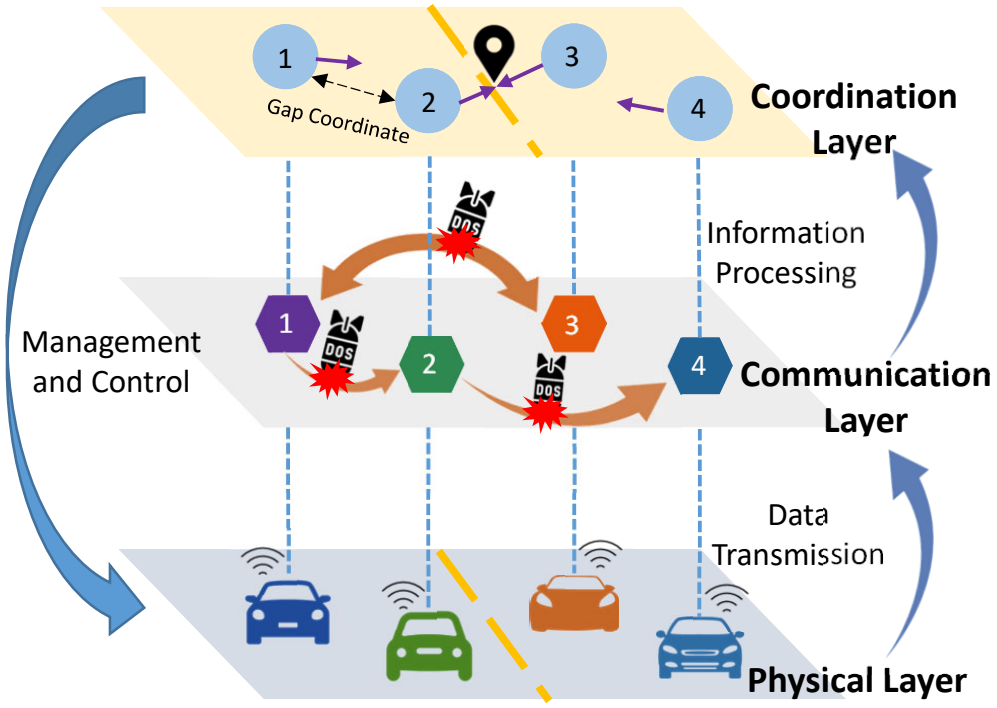

Figure 3.3: Cooperative regulation problem for a CPS consisting of four ground vehicles.

In this section, we consider a CPS comprising four ground vehicles to test the performance of the proposed approach. As shown in Figure 3.3, this multi-agent CPS has a hierarchical architecture consisting of a physical, communication, and coordination layers. Each vehicle in the physical layer has a communication module in the network layer with the communication topology $\mathcal{N}_1 = \{3\}$, $\mathcal{N}_2 = \{1\}$, $\mathcal{N}_3 = \{1\}$, and $\mathcal{N}_4 = \{2\}$. In the meantime, DoS attacks will affect the communication among these communication modules in a random manner, to block each vehicle from

Table 3.1: Parameters of the four vehicles

Vehicle Index	M_i (kg)	C_i ($\mathrm{N \cdot s^2 \cdot m^{-2}}$)	R_i (m)
1	1000	0.99	0.30
2	1200	1.1	0.38
3	1500	1.3	0.39
4	1400	1.2	0.37

receiving information from its neighbors, leading to degraded cooperative regulation performance. To guarantee the performance under DoS attacks and the disturbances on each vehicle, we apply the proposed robust and resilient distributed MPC strategy to this system.

3.5.1 System Model and Parameter Configuration

In this chapter, we only consider the vehicle longitudinal dynamics as adopted in [96]. For the purpose of striking a balance between accuracy and conciseness, the following assumptions have been made: 1) the vehicle body is rigid and strictly left-right symmetric; 2) no tire slip in the longitudinal direction; 3) the driving and braking torques are integrated to one general torque. With the assumptions above, the vehicle i, $i \in \mathscr{M}$ in this system has the nonlinear dynamic model, which is given by:

$$\begin{cases} s_i(k+1) = s_i(k) + T_c v_i(k) \\ v_i(k+1) = v_i(k) + \dfrac{T_c}{M_i}\Big(\dfrac{T_i(k)}{R_i} - F_i\big(v_i(k)\big)\Big) + w_i(k) \end{cases},$$

where $T_c = 0.3$s is the sampling period; $x_i(k) = [s_i(k),\, v_i(k)]^{\mathrm{T}}$ is the system state; $s_i(k)$ and $v_i(k)$ represent the position and velocity of vehicle i, respectively; M_i is the vehicle mass; $T_i(k)$ is the integrated driving/braking torque; R_i is the tire radius; $F_i\big(v_i(k)\big) = C_i {v_i}^2(k)$ denotes these aerodynamic drag, where C_i is a aerodynamic coefficient. The vehicle coefficients are shown in the TABLE 3.1. To simplify the

algorithm, let $u_i(k) = \frac{T_i(k)}{M_i R_i}$ be the control input. Hence, the integrated torque $T_i(k)$ for each vehicle can be derived through a simple linear transformation after deriving the control input $u_i(k)$. The state constraint for each vehicle is assumed to be same, which is given by $\mathbb{X}_i = \big\{[s_i, v_i]^{\mathrm{T}} | -1.5\,\mathrm{m} \leq s_i \leq 1.5\,\mathrm{m},\ -1.5\,\mathrm{m/s} \leq v_i \leq 1.5\,\mathrm{m/s}\big\}$; the torque constraints are given as $\mathbb{U}_1 = \big\{T_1 | -1300\,\mathrm{N} \leq T_1 \leq 1300\,\mathrm{N}\big\}$. $\mathbb{U}_2 = \big\{T_2 | -2000\,\mathrm{N} \leq T_2 \leq 2000\,\mathrm{N}\big\}$, $\mathbb{U}_3 = \big\{T_3 | -2500\,\mathrm{N} \leq T_3 \leq 2500\,\mathrm{N}\big\}$, and $\mathbb{U}_4 = \big\{T_4 | -2300\,\mathrm{N} \leq T_4 \leq 2300\,\mathrm{N}\big\}$, respectively; the disturbances in the four agents are $w_1(k) = 0.0015\sin(\frac{\pi k}{15})$, $w_2(k) = 0.0015\cos(\frac{\pi k}{10})$, $w_3(k) = 0.0015\cos(\frac{\pi k}{5})$, and $w_4(k) = 0.0015\cos(\frac{\pi k}{5} - \frac{\pi}{4})$, respectively. The initial states of the three agents are set as $x_1(0) = [-0.95, -0.3]^{\mathrm{T}}$, $x_2(0) = [-1.4, 0.1]^{\mathrm{T}}$, $x_3(0) = [-1.1, -1.2]^{\mathrm{T}}$, and $x_4(0) = [-1.0, -1.3]^{\mathrm{T}}$ respectively. In the meantime, DoS attacks are set as occurring on all the communication channels among agents at arbitrary time instants, and the launching time of DoS attacks in this simulation is illustrated in Figure 3.4.

The design parameters for the proposed robust and resilient distributed MPC algorithm are given in the following. The prediction horizon is set as $N_p = 10$; the weighting matrices Q_1, Q_2, Q_3, and Q_4 are set as [1.1, 0; 0, 1.1], with R_1, R_2, R_3 and R_4 being 1. The corresponding feedback control gain K_i is designed as $K_1 = K_2 = K_3 = K_4 = [0.80, 1.62]$; According to *Assumption 6*, the terminal penalty matrices are derived as $P_1 = P_2 = P_3 = P_4 = [2.23, 0.63; 0.63, 4.69]$. Under these circumstances, the terminal region levels are derived as $\epsilon_1 = \epsilon_2 = \epsilon_3 = \epsilon_4 = 0.70$. Based on [27], the Lipschitz constants L_{f_i} are calculated as $L_{f_1} = 1.17$, $L_{f_2} = 1.17$, $L_{f_3} = 1.16$, and $L_{f_4} = 1.16$, respectively. Furthermore, by following the presented sufficient conditions in *Theorem 1*, we can choose the scaling parameters: $\zeta_i = 0.25$ and $\xi_i = 0.91$. In this simulation, the cooperation matrix Q_{ij} are designed as $Q_{12} = Q_{13} = Q_{31} = Q_{24} = [0.022, 0; 0, 0.022]$.

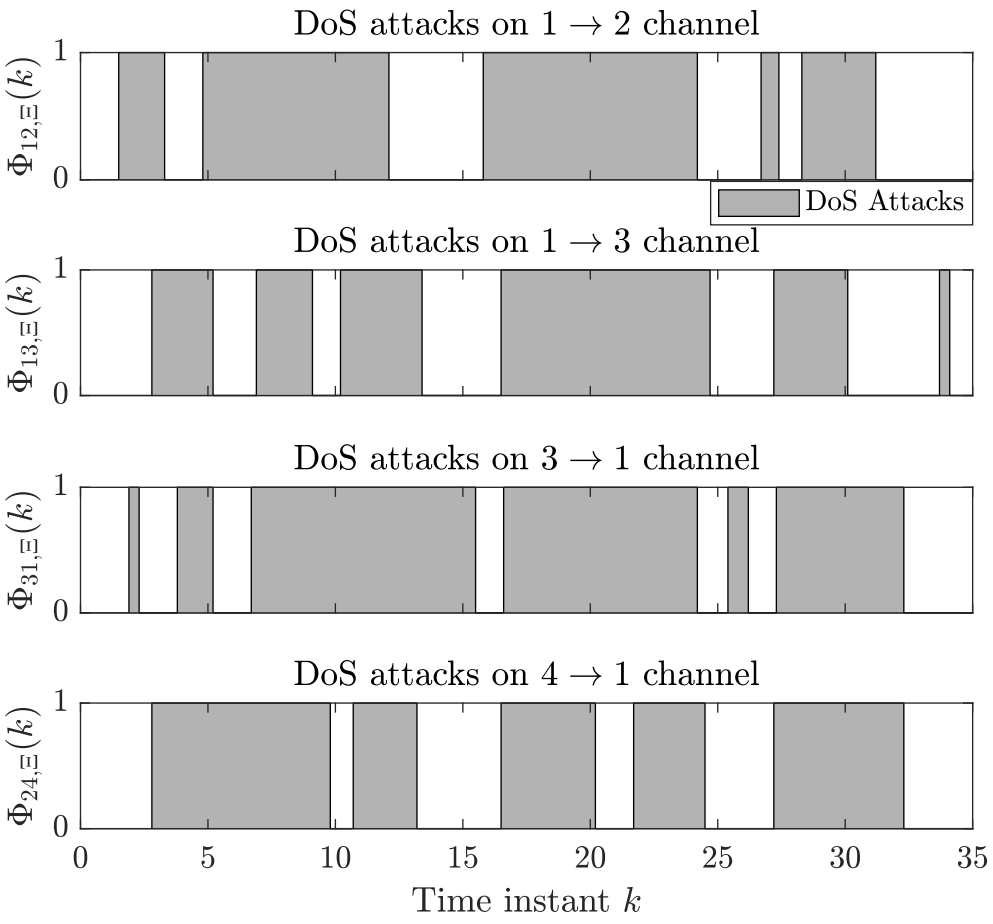

Figure 3.4: Launching time of DoS attacks.

3.5.2 Simulation Results Analysis

The optimization problem $\mathcal{P}_i$ is solved with the nonlinear programming solver IPOPT [75] via the YALMIP [37] toolbox in MATLAB. The state trajectories of the cooperative regulation problem of this multi-agent CPS are demonstrated in Figure 3.5. It can be observed that the states of this multi-agent CPS are finally steered into the region $\Omega_1^* \times \Omega_2^* \times \Omega_3^* \times \Omega_4^*$, which verifies the *Theorem 2*. Furthermore, Figure 3.6 illustrates the torque input sequences for the four subsystems, respectively. Based on the results above, it can be verified that the proposed robust and resilient distributed MPC

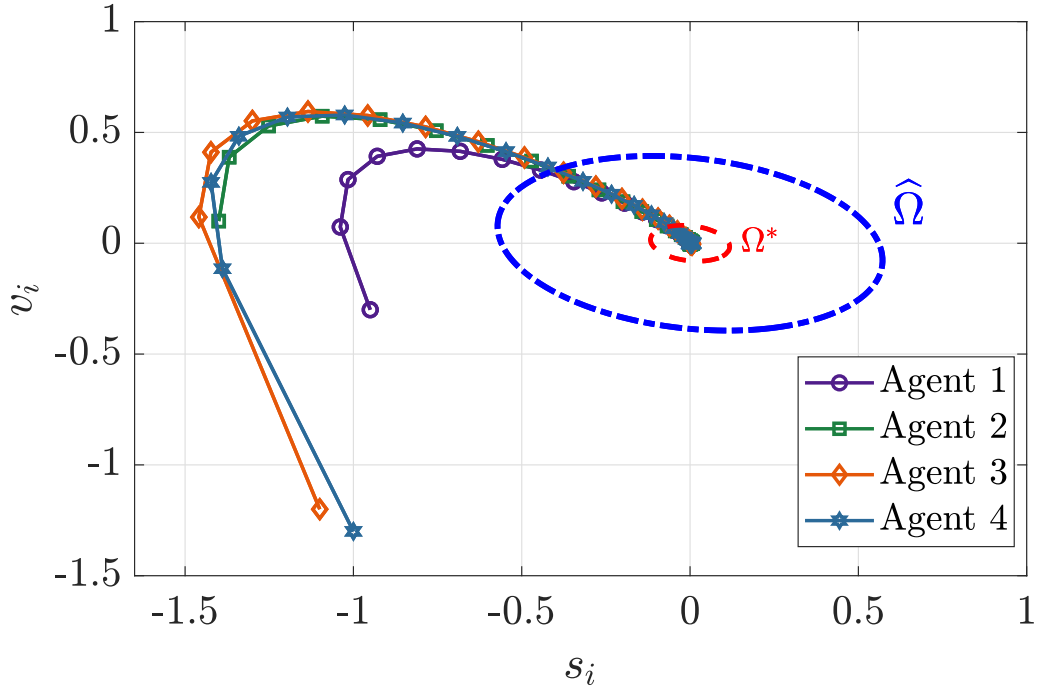

Figure 3.5: State trajectories of the CPS.

strategy can achieve the cooperation regulation goal with guaranteed input constraint satisfaction under randomly existing DoS attacks.

For the purpose of verifying the effectiveness, the proposed method is compared with the standard distributed MPC method [7] in simulation. To make a fair comparison, we choose the same control parameters for both distributed MPC methods. To guarantee the implementation of the distributed MPC method, we set each controller in this system to use all zero sequences to represent the neigbors' states when attacks occur on this channel. To further show the effectiveness of the proposed method, we compare the deviation between the actual state and the center of the system generated by using the distributed MPC method with robustness constraint and the proposed method, respectively. The result is shown in Figure 3.7.

Let $s_a(k) = \frac{1}{M}\sum_{i=1}^{M} s_i$ and $v_a(k) = \frac{1}{M}\sum_{i=1}^{M} v_i$ be the average of the position and the velocity of the four vehicles. Then we can introduce $d_{s,i}(k)$ and $d_{v,i}(k)$ to be the

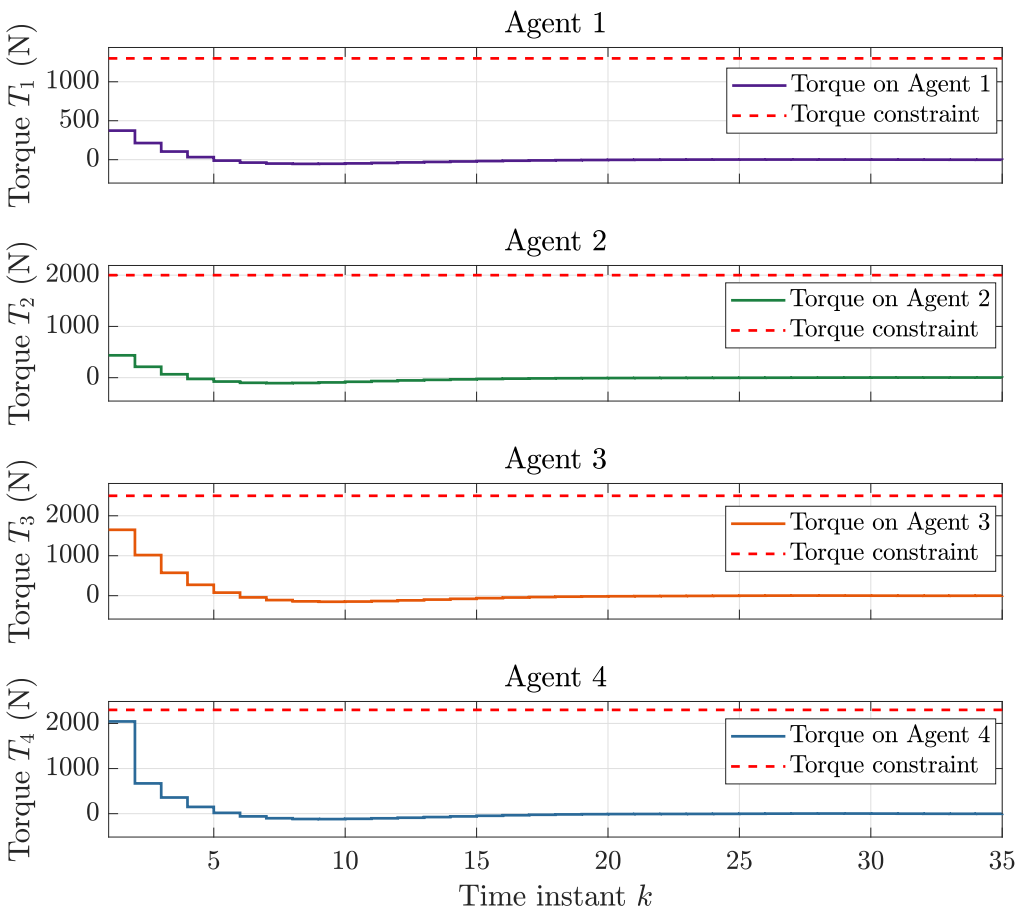

Figure 3.6: Integrated torques for four ground vehicles.

deviation between each state and the center of the system, respectively. With the definition above, these indexes are calculated as:

$$d_{s,i}(k) = \|s_i(k) - s_a(k)\|,\ d_{v,i}(k) = \|v_i(k) - v_a(k)\|.$$

Figure 3.7 shows the deviation comparison result between the proposed method and the distributed MPC method with robustness constraint. It can be observed that the proposed method accelerates the speed of convergence under the DoS attacks. To conclude, by applying the robust and resilient distributed MPC control strategy, the

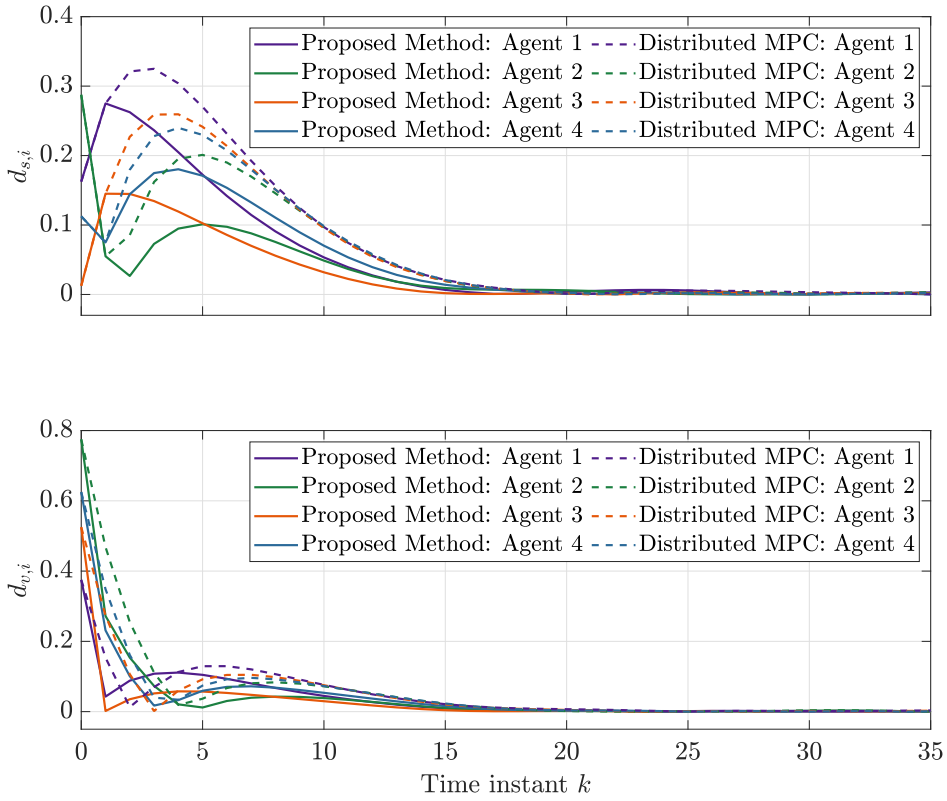

Figure 3.7: Deviation between each state and the center of all vehicles.

states of the multi-agent CPS can be steered into the region $\Omega_1^* \times \Omega_2^* \times \Omega_3^* \times \Omega_4^*$ under the bounded disturbance and randomly occurring DoS attacks, which meets the theoretical analysis in 3.4.

3.6 Conclusion

In this work, we have developed the robust and resilient distributed MPC framework for discrete-time nonlinear multi-agent CPS subject to external disturbances and randomly occurring DoS attacks to achieve the cooperative regulation goal. A new type of robustness constraint approach is proposed to enhance the robustness of the MPC algorithm while also enlarging the region of attraction compared to the original

one. Furthermore, a lengthened sequence transmission strategy is also applied to utilize and lengthen the predicted state and control input sequences to mitigate the information block out among the agents induced by DoS attacks. We have proven that the proposed algorithm is recursively feasible and the state of the closed-loop multi-agent CPS can be steered into a small region containing the equilibrium. Numerical results also show the advantages of the proposed work.

Chapter 4

Conclusions and Future Plans

4.1 Conclusions

In this thesis, a robust and resilient MPC framework has been addressed to tackle the external disturbances and DoS attacks in both single-agent and multi-agent CPSs.

Chapter 2 involves an AUV trajectory-tracking control application. In this problem, DoS attacks occur on the remote controller to the actuator channel in a random manner, hindering the AUV from achieving its original control objective. To address this issue, firstly, an error model for the MPC to solve is derived and simplified. Furthermore, the tightening constraint approach is utilized to tackle the bounded additive disturbances that affect the AUV. Moreover, the packet-transmission strategy is also utilized in this work to store the predicted integral torque sequence in the buffer. Thus, the actuator can use the predicted control input saved previously as the candidate input to complete the tracking task. Finally, an numerical simulation result is given to show the effectiveness of the proposed control scheme to track the reference trajectory in real time compared with the standard MPC with no resilient control design.

In chapter 3, we propose a novel robust and resilient distributed MPC framework for nonlinear multi-agent CPSs. In this work, DoS attacks are purposely launched to the network-based communication channels among agents, blocking each agent from sending their state information to their neighbors. To tackle this issue, we design a lengthened sequence transmission strategy in the control framework: At each sampling instant, the controller in each agent generates both the predicted state and control input sequence, and lengthens the sequence using the state-feedback control law regarding the longest duration of the DoS attacks. By achieving this, each agent can receive the required state sequence whenever the attacks launch. Furthermore, a novel tightening constraint approach based on both the state constraint and the terminal constraint is designed in this control framework, enhancing the region of attraction compared to the original ones to tackle the disturbances. The simulation result is also given to show the effectiveness of a multi-vehicle system.

4.2 Research plan

In this thesis, a robust and resilient MPC framework is proposed to tackle the DoS attacks for both single-agent CPS and multi-agent CPS. However, there still exist some limitations in each work. In addition, many interesting areas and problems related to this topic are worth investigating. We list some of them here.

- In chapter 1, a cyber-security defense framework is introduced. However, in this thesis, we mainly focus on the mitigation process, which means that we can only tackle the attacks when they occur (No specific detection mechanism is required since DoS attacks can always be found when the receiver obtains nothing). Nevertheless, other attacks such as FDI attacks and replay attacks cannot be detected in this simple way. In this regard, a detection mechanism

needs to be investigated to consider other kinds of attacks in the multi-agent CPS.

- Another potential research is to enhance confidentiality in CPS. At this point, an encode and decode process can be considered when exchanging data. After the encryption is achieved, it will be much harder for the attackers to steal the information from the original communication channels, which prevents potential cyber attacks from the source.

- In both chapter 2 and chapter 3, a packet rather than a vector need to be generated and transmitted at each sampling instant, which increases the burden of data transmission. To address this issue, we can design a quantized method in the communication channel. Quantized method has already been integrated with MPC design in the networked-control systems. However, current quantized control schemes mainly focus on embedding a quantizer onto the C-A channels or the S-C channel, without considering the interference from the communication channels in a CPS with multi-agent architecture. Hence, the question of how to efficiently integrate the distributed MPC framework with the quantizer on the communication channels remains unsolved.

Bibliography

[1] Saurabh Amin, Alvaro A. Cárdenas, and S. Shankar Sastry. Safe and secure networked control systems under denial-of-service attacks. In *Proceedings of Hybrid Systems: Computation and Control: 12th International Conference (HSCC)*, pages 31–45, Berlin, Heidelberg, April 13 - 15, 2009. Springer Berlin Heidelberg.

[2] Radhakisan Baheti and Helen Gill. Cyber-physical systems. *The Impact of Control Technology*, 12(1):161–166, 2011.

[3] Mohammad Hossein Basiri, Nasser L. Azad, and Sebastian Fischmeister. Attack resilient heterogeneous vehicle platooning using secure distributed nonlinear model predictive control. In *Proceedings of 2020 28th Mediterranean Conference on Control and Automation (MED)*, pages 307–312, September 15 - 18, 2020.

[4] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on Automatic Control*, 48(9):1600 – 1606, 2003.

[5] Masoud Bonyani, Mohammad Mehdi Ghanbarian, and Mohsen Simab. Blockchain technology based exchanged information security for demand-side management of grid-connected microgrid using model predictive control. *IET Generation, Transmission & Distribution*, 2022.

[6] Justin M. Bradley and Ella M. Atkins. Optimization and control of cyber-physical vehicle systems. *Sensors*, 15(9):23020–23049, 2015.

[7] E. Camponogara, D. Jia, B.H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Systems Magazine*, 22(1):44–52, 2002.

[8] Glenn Carl, George Kesidis, Richard R. Brooks, and Suresh Rai. Denial-of-service attack-detection techniques. *IEEE Internet Computing*, 10(1):82 – 89, 2006.

[9] Alberto Huertas Celdrán, Manuel Gil Pérez, Félix J García Clemente, and Gregorio Martínez Pérez. Sustainable securing of medical cyber-physical systems for the healthcare of the future. *Sustainable Computing: Informatics and Systems*, 19:138–146, 2018.

[10] Ahmet Cetinkaya, Kaito Kikuchi, Tomohisa Hayakawa, and Hideaki Ishii. Randomized transmission protocols for protection against jamming attacks in multi-agent consensus. *Automatica*, 117:108960, 2020.

[11] Mohammadreza Chamanbaz, Fabrizio Dabbene, and Roland Bouffanais. A physics-based attack detection technique in cyber-physical systems: A model predictive control co-design approach. In *Proceedings of 2019 Australian & New Zealand Control Conference (ANZCC)*, pages 18–23. IEEE, November 27 - 29, 2019.

[12] Hong Chen and Frank Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.

[13] Jicheng Chen and Yang Shi. Stochastic model predictive control framework for resilient cyber-physical systems: Review and perspectives. *Philosophical Transactions of the Royal Society*, 379(2207):1–13, 2021.

[14] Jicheng Chen, Hui Zhang, and Guodong Yin. Distributed dynamic event-triggered secure model predictive control of vehicle platoon against DoS attacks. *IEEE Transactions on Vehicular Technology*, pages 1–14, 2022.

[15] Yu Chen, Weikang Qiu, Xinmin Liu, and Yong Kang. A parallel control framework of analog proportional integral and digital model predictive controllers for enhancing power converters cybersecurity. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 10(1):1258–1269, 2019.

[16] Armando W. Colombo, Stamatis Karnouskos, Okyay Kaynak, Yang Shi, and Shen Yin. Industrial cyberphysical systems: A backbone of the fourth industrial revolution. *IEEE Industrial Electronics Magazine*, 11(1):6–16, 2017.

[17] Ján Drgoňa, Javier Arroyo, Iago Cupeiro Figueroa, David Blum, Krzysztof Arendt, Donghun Kim, Enric Perarnau Ollé, Juraj Oravec, Michael Wetter, Draguna L. Vrabie, and Lieve Helsen. All you need to know about model predictive control for buildings. *Annual Reviews in Control*, 50:190–232, 2020.

[18] William B. Dunbar. Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Transactions on Automatic Control*, 52(7):1249–1263, 2007.

[19] Helen Durand. State measurement spoofing prevention through model predictive control design. In *Proceedings of 6th IFAC Conference on Nonlinear Model Predictive Control*, volume 51, pages 543–548, Madison, Wisconsin, USA, August, 19 - 22, 2018. Elsevier.

[20] Giuseppe Franzè, Domenico Famularo, Walter Lucia, and Francesco Tedesco. Cyber–physical systems subject to false data injections: A model predictive control framework for resilience operations. *Automatica*, 152:110957, 2023.

[21] Giuseppe Franzè, Francesco Tedesco, and Walter Lucia. Resilient control for cyber-physical systems subject to replay attacks. *IEEE Control Systems Letters*, 3(4):984–989, 2019.

[22] Giuseppe Franzè, Walter Lucia, and Francesco Tedesco. Resilient model predictive control for constrained cyber-physical systems subject to severe attacks on the communication channels. *IEEE Transactions on Automatic Control*, 67(4):1822–1836, 2022.

[23] Nikolai Gershfeld, Tyler M. Paine, and Michael R. Benjamin. Adaptive and collaborative bathymetric channel-finding approach for multiple autonomous marine vehicles. *IEEE Robotics and Automation Letters*, 8(7):4028–4035, 2023.

[24] Ning He, Kai Ma, and Huiping Li. Resilient predictive control strategy of cyber–physical systems against FDI attack. *IET Control Theory & Applications*, 2022.

[25] Hai Hu, Furong Wang, Fan Zhang, Weijia Jia, and Ge Tang. Automatic mobile vehicle for adaptive real-time communication relay. In *Proceedings of 2009 29th IEEE International Conference on Distributed Computing Systems Workshops*, pages 32–37, Montreal, Quebec, Canada, June 22 - 26, 2009.

[26] Nasser Jazdi. Cyber physical systems in the context of Industry 4.0. In *Proceedings of 2014 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, Cluj-Napoca, Romania, May 22-24, 2014.

[27] H.K. Khalil. *Nonlinear Systems*. Pearson Education. Prentice Hall, 2000.

[28] Jin Ho Kim. A review of cyber-physical system research relevant to the emerging it trends: Industry 4.0, IoT, big data, and cloud computing. *Journal of Industrial Integration and Management*, 2(03):1750011, 2017.

[29] Kyoung-Dae Kim and P. R. Kumar. Cyber–physical systems: A perspective at the centennial. *Proceedings of the IEEE*, 100:1287–1308, 2012.

[30] Byung-Cheol Kum, Dong-Hyeok Shin, Seok Jang, Seung Yong Lee, Jung Han Lee, TaeJun Moh, Dong Gil Lim, Jong-Dae Do, and Jin Hyung Cho. Application of unmanned surface vehicles in coastal environments: Bathymetric survey using a multibeam echosounder. *Journal of Coastal Research*, 95(SI):1152–1156, 06 2020.

[31] Bin Li, Xinglian Zhou, Zhaoke Ning, Xiaoyi Guan, and Ka-Fai Cedric Yiu. Dynamic event-triggered security control for networked control systems with cyber-attacks: A model predictive control approach. *Information Sciences*, 612:384–398, 2022.

[32] Huiping Li and Yang Shi. Robust distributed model predictive control of constrained continuous-time nonlinear systems: A robustness constraint approach. *IEEE Transactions on Automatic Control*, 59(6):1673–1678, 2013.

[33] Huiping Li and Yang Shi. Distributed receding horizon control of large-scale nonlinear systems: Handling communication delays and disturbances. *Automatica*, 50(4):1264–1271, 2014.

[34] Huiping Li and Yang Shi. *Robust Receding Horizon Control for Networked and Distributed Nonlinear Systems*. Springer, 2017.

[35] Gaoqi Liang, Junhua Zhao, Fengji Luo, Steven R Weller, and Zhao Yang Dong. A review of false data injection attacks against modern power systems. *IEEE Transactions on Smart Grid*, 8(4):1630–1638, 2016.

[36] Yuezhi Liu, Yong Chen, and Meng Li. Event-based model predictive damping control for power systems with cyber-attacks. *ISA Transactions*, 136:687–700, 2023.

[37] J. Lofberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of 2004 IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, April 26-May 1, 2004.

[38] Martin Ludvigsen and Asgeir J. Sørensen. Towards integrated autonomous underwater operations for ocean mapping and monitoring. *Annual Reviews in Control*, 42:145–157, 2016.

[39] Rui Ma, Sagnik Basumallik, Sara Eftekharnejad, and Fanxin Kong. Recovery-based model predictive control for Cascade mitigation under cyber-physical attacks. In *Proceedings of 2020 IEEE Texas Power and Energy Conference (TPEC)*, pages 1–6, College Station, TX, USA, February 6 - 7, 2020. IEEE.

[40] Yong Ma, Zongqiang Nie, Songlin Hu, Zhixiong Li, Reza Malekian, and M. Sotelo. Fault detection filter and controller co-design for unmanned surface vehicles under DoS attacks. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1422–1434, 2021.

[41] D.Q. Mayne and W. Langson. Robustifying model predictive control of constrained linear systems. *Electronics Letters*, 37(23):1422 – 1423, 2001.

[42] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.

[43] Anthony R. Metke and Randy L. Ekl. Security technology for smart grid networks. *IEEE Transactions on Smart Grid*, 1(1):99 – 107, 2010.

[44] H. Michalska and D.Q. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 38(11):1623–1633, 1993.

[45] Yilin Mo and Bruno Sinopoli. Secure control against replay attacks. In *Proceedings of 2009 47th Annual Allerton Conference on Communication, Control, and Computing, Monticello, Illinois, October 1-3*, 2009.

[46] Robin R. Murphy, Karen L. Dreger, Sean Newsome, Jesse Rodocker, Eric Steimle, Tetsuya Kimura, Kenichi Makabe, Fumitoshi Matsuno, Satoshi Tadokoro, and Kazuyuki Kon. Use of remotely operated marine vehicles at Minamisanriku and Rikuzentakata Japan for disaster recovery. In *Proceedings of 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, pages 19–25, November 1 - 5, 2011.

[47] Ahmed S. Musleh, Guo Chen, and Zhao Yang Dong. A survey on the detection algorithms for false data injection attacks in smart grids. *IEEE Transactions on Smart Grid*, 11(3):2218–2234, 2020.

[48] Edin Omerdic, Daniel Toal, and Zoran Vukic. User interface for interaction with heterogeneous vehicles for cyber-physical systems. In *Proceedings of 2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1–5, Phuket, Thailand, November 13 - 15, 2016.

[49] Henrique Oyama and Helen Durand. Integrated cyberattack detection and resilient control strategies using Lyapunov-based economic model predictive control. *AIChE Journal*, 66(12):e17084, 2020.

[50] T. Parisini, M. Sanguineti, and R. Zoppoli. Nonlinear stabilization by receding-horizon neural regulators. *International Journal of Control*, 70(3):341–362, 1998.

[51] Datian Peng, Jianmin Dong, Jianan Jian, Qinke Peng, Bo Zeng, and Zhi-Hong Mao. Economic-driven FDI attack in electricity market. In *Proceedings of the 1st International Conference on Science of Cyber Security*, pages 216–224, Beijing, China, August 12 - 14, 2018. Springer International Publishing.

[52] Thomas Pierron, Teresa Árauz, Jose Maria Maestre, A Cetinkaya, and C Stoica Maniu. Tree-based model predictive control for jamming attacks. In *Proceedings of 2020 European Control Conference (ECC)*, pages 948–953, Saint Petersburg, Russia, May 12-15, 2020. IEEE.

[53] Gilberto Pin, Davide M. Raimondo, Lalo Magni, and Thomas Parisini. Robust model predictive control of nonlinear systems with bounded and state-dependent uncertainties. *IEEE Transactions on Automatic Control*, 54(7):1681–1687, 2009.

[54] Alison A Proctor. *Semi-autonomous guidance and control of a Saab SeaEye Falcon ROV*. PhD thesis, Department of Mechanical Engineering, University of Victoria, Victoria, Canada, 2014.

[55] S Joe Qin and Thomas A Badgwell. A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764, 2003.

[56] Hongchun Qu, Yu Li, and Wei Liu. Output feedback model predictive control for NCSs with input quantization. *Complexity*, 2022, 2022.

[57] Helem S Sánchez, Damiano Rotondo, Teresa Escobet, Vicenç Puig, and Joseba Quevedo. Bibliographical review on cyber attacks from a control oriented perspective. *Annual Reviews in Control*, 48:103–128, 2019.

[58] Chao Shen and Yang Shi. Distributed implementation of nonlinear model predictive control for AUV trajectory tracking. *Automatica*, 115:108863, 2020.

[59] Chao Shen, Yang Shi, and Brad Buckham. Integrated path planning and tracking control of an auv: A unified receding horizon optimization approach. *IEEE/ASME Transactions on Mechatronics*, 22(3):1163–1173, 2017.

[60] Chao Shen, Yang Shi, and Brad Buckham. Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control. *IEEE Transactions on Industrial Electronics*, 65(7):5796–5805, 2018.

[61] Ting Shi, Peng Shi, and Huiyan Zhang. Model predictive control of distributed networked control systems with quantization and switching topology. *International Journal of Robust and Nonlinear Control*, 30(12):4584–4599, 2020.

[62] Yang Shi, Chao Shen, Henglai Wei, and Kunwu Zhang. *Lyapunov-Based Model Predictive Control for Dynamic Positioning and Trajectory-Tracking Control of an AUV*, pages 49–75. Springer International Publishing, Cham, 2023.

[63] Yang Shi and Kunwu Zhang. Advanced model predictive control framework for autonomous intelligent mechatronic systems: A tutorial overview and perspectives. *Annual Reviews in Control*, 52:170–196, 2021.

[64] Yan Song, Zidong Wang, Lei Zou, and Shuai Liu. Endec-decoder-based N-step model predictive control: Detectability, stability and optimization. *Automatica*, 135:109961, 2022.

[65] Qi Sun, Jicheng Chen, and Yang Shi. Integral-type event-triggered model predictive control of nonlinear systems with additive disturbance. *IEEE Transactions on Cybernetics*, 51(12):5921–5929, 2021.

[66] Qi Sun, Jicheng Chen, and Yang Shi. Event-triggered robust MPC of nonlinear cyber-physical systems against DoS attacks. *Science China Information Sciences*, 65(1):1–17, 2022.

[67] Qi Sun and Yang Shi. Model predictive control as a secure service for cyber–physical systems: A cloud-edge framework. *IEEE Internet of Things Journal*, 9(22):22194–22203, 2021.

[68] Qi Sun, Kunwu Zhang, and Yang Shi. Resilient model predictive control of cyber-physical systems under DoS attacks. *IEEE Transactions on Industrial Informatics*, 16(7):4920–4927, 2019.

[69] Yuan-Cheng Sun and Guang-Hong Yang. Robust event-triggered model predictive control for cyber-physical systems under denial-of-service attacks. *International Journal of Robust and Nonlinear Control*, 29(14):4797–4811, 2019.

[70] Atharva Suryavanshi, Aisha Alnajdi, Mohammed Alhajeri, Fahim Abdullah, and Panagiotis D Christofides. Encrypted model predictive control design for security to cyberattacks. *AIChE Journal*, page e18104, 2023.

[71] Vo-Van Thanh and Wencong Su. Data-driven model predictive control-based proactive scheduling for commercial microgrid considering anomaly detection. *IEEE Systems Journal*, 17(2), 2022.

[72] Sergio Vazquez, Jose I Leon, Leopoldo G Franquelo, Jose Rodriguez, Hector A Young, Abraham Marquez, and Pericle Zanchetta. Model predictive control:

A review of its applications in power electronics. *IEEE Industrial Electronics Magazine*, 8(1):16–31, 2014.

[73] Pablo Velarde, José M Maestre, Hideaki Ishii, and Rudy R Negenborn. Vulnerabilities in lagrange-based distributed model predictive control. *Optimal Control Applications and Methods*, 39(2):601–621, 2018.

[74] Pablo Velarde, Jose Maria Maestre, Hideaki Ishii, and Rudy R Negenborn. Scenario-based defense mechanism for distributed model predictive control. In *Proceedings of 2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 6171–6176, Melbourne, Australia, December 12 - 15, 2017. IEEE.

[75] A Wächter and LT Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106:25–57, 2006.

[76] Jianhua Wang, Yan Song, Shuai Liu, and Sunjie Zhang. Security in H_2-sense for polytopic uncertain systems with attacks based on model predictive control. *Journal of the Franklin Institute*, 353(15):3769–3785, 2016.

[77] Jianhua Wang, Yan Song, and Guoliang Wei. Security-based resilient robust model predictive control for polytopic uncertain systems subject to deception attacks and RR protocol. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(8):4772–4783, 2021.

[78] Jun Wang, Baocang Ding, and Jianchen Hu. Security control for LPV system with deception attacks via model predictive control: A dynamic output feedback approach. *IEEE Transactions on Automatic Control*, 66(2):760–767, 2020.

[79] Henglai Wei, Chao Shen, and Yang Shi. Distributed lyapunov-based model predictive formation tracking control for autonomous underwater vehicles subject to

disturbances. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(8):5198–5208, 2021.

[80] Zhe Wu, Fahad Albalawi, Junfeng Zhang, Zhihao Zhang, Helen Durand, and Panagiotis D Christofides. Detecting and handling cyber-attacks in model predictive control of chemical processes. *Mathematics*, 6(10):173, 2018.

[81] Gang Xiong, Fenghua Zhu, Xiwei Liu, Xisong Dong, Wuling Huang, Songhang Chen, and Kai Zhao. Cyber-physical-social system in intelligent transportation. *IEEE/CAA Journal of Automatica Sinica*, 2(3):320–333, 2015.

[82] Hansong Xu, Wei Yu, David Griffith, and Nada Golmie. A survey on industrial internet of things: A cyber-physical systems perspective. *IEEE Access*, 6:78238–78259, 2018.

[83] Zheng Yan and Jun Wang. Model predictive control for tracking of underactuated vessels based on recurrent neural networks. *IEEE Journal of Oceanic Engineering*, 37(4):717–726, 2012.

[84] Qingyu Yang, Jie Yang, Wei Yu, Dou An, Nan Zhang, and Wei Zhao. On false data-injection attacks against power system state estimation: Modeling and countermeasures. *IEEE Transactions on Parallel and Distributed Systems*, 25(3):717 – 729, 2014.

[85] Zehua Ye, Ying Xu, Jia-Hao Dong, Juntong Chen, and Dan Zhang. Resilient sliding mode control of multiple autonomous underwater vehicles under stochastic DoS attack. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, 237(2):498–507, 2023.

[86] Zehua Ye, Dan Zhang, Jun Cheng, and Zheng-Guang Wu. Event-triggering and quantized sliding mode control of UMV systems under DoS attack. *IEEE Transactions on Vehicular Technology*, 71(8):8199–8211, 2022.

[87] Zehua Ye, Dan Zhang, Chao Deng, Huaicheng Yan, and Gang Feng. Finite-time resilient sliding mode control of nonlinear UMV systems subject to DoS attacks. *Automatica*, 156:111170, 2023.

[88] Zehua Ye, Dan Zhang, and Zheng-Guang Wu. Adaptive event-based tracking control of unmanned marine vehicle systems with DoS attack. *Journal of the Franklin Institute*, 358(3):1915–1939, 2021.

[89] Zehua Ye, Dan Zhang, Huaicheng Yan, and Zheng-Guang Wu. A semi-markovian jumping system approach to secure DPC of nonlinear networked unmanned marine vehicle systems with DoS attack. *Journal of the Franklin Institute*, 2021.

[90] Rajaa Vikhram Yohanandhan, Rajvikram Madurai Elavarasan, Premkumar Manoharan, and Lucian Mihet-Popa. Cyber-physical power system (CPPS): A review on modeling, simulation, and analysis with cyber security applications. *IEEE Access*, 8:151019–151064, 2020.

[91] Jimin Yu, Liangsheng Nan, Xiaoming Tang, and Ping Wang. Model predictive control of non-linear systems over networks with data quantization and packet loss. *ISA Transactions*, 59:1–9, 2015.

[92] Shuyou Yu, Marcus Reble, Hong Chen, and Frank Allgöwer. Inherent robustness properties of quasi-infinite horizon nonlinear model predictive control. *Automatica*, 50(9):2269–2280, 2014.

[93] Dan Zhang, Zehua Ye, Gang Feng, and Hongyi Li. Intelligent event-based fuzzy dynamic positioning control of nonlinear unmanned marine vehicles under DoS attack. *IEEE Transactions on Cybernetics*, 52(12):13486–13499, 2022.

[94] Kunwu Zhang, Yang Shi, Stamatis Karnouskos, Thilo Sauter, Huazhen Fang, and Armando Walter Colombo. Advancements in industrial cyber-physical systems: An overview and perspectives. *IEEE Transactions on Industrial Informatics*, 19(1):716–729, 2023.

[95] Dong Zhao, Yang Shi, Steven X. Ding, Yueyang Li, and Fangzhou Fu. Replay attack detection based on parity space method for cyber-physical systems. *arXiv*, arXiv:2306.02020, 2023.

[96] Yang Zheng, Shengbo Eben Li, Keqiang Li, Francesco Borrelli, and J. Karl Hedrick. Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies. *IEEE Transactions on Control Systems Technology*, 25(3):899–910, 2017.