Doctor of Philosophy Engineering

Vision-Based Adaptive Cruise Control in Vehicle Platooning Navigation

The Graduate School

of the University of Ulsan

Department of Electrical Engineering

Yang Yu

Vision-Based Adaptive Cruise Control in Vehicle Platooning Navigation

Supervisor: Kang-Hyun Jo

A Dissertation

Submitted to

the Graduate School of the University of Ulsan

In partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy Engineering

by

Yang Yu

Department of Electrical Engineering

Ulsan, Korea

November 2018

Yang Yu 의 공학박사학위 논문을 인준함

심사위원장	조강현	(인)
심사위원	강희준	(인)
심사위원	서영수	(인)
심사위원	김성현	(인)
심사위원	강현덕	(인)

울산대학교 대학원

2018 년 11 월

Vision-Based Adaptive Cruise Control in Vehicle Platooning Navigation

This certifies that the dissertation of Yang Yu is approved.

Professor Kang-Hyun Jo

Committee Chair

Professor Hee-Jun Kang

Committee Member

Professor Young-Soo Suh

Committee Member

Professor Sung-Hyun Kim

Committee Member

Professor Hyun-Deok Kang

Committee Member

Department of Electrical Engineering

Ulsan, Korea

November 2018

Acknowledgments

During the doctor course, I experienced and learned a lot of things. I would like to thank many people who have contributed to this thesis.

First of all, I would like to take this opportunity to express my deepest gratitude to my advisor Professor Kang-Hyun Jo. His brilliant insight, patient guidance and countless encouragement helped me to complete the writing papers, as well as Ph.D degree successfully, and find a confidence. It is very honored for me to study with him. Without his consistent and illuminating instruction, the completion of this thesis would have never become possible.

I would like to thank all my colleagues of our laboratory, they are also my friends who always gave me a cheering and understanding.

I am very much grateful to University of Ulsan for giving me such a wonderful research environment and financial support. This university gave me the chance to conduct a number of useful courses with excellent instructors that will give me a way to think and work properly in the rest of my life.

Finally, I sincerely thank my parents for devotion, encouragement, and education. Without heir unconditional love and support, my dream of this thesis would never have become true. I would like to specially thank my wife for her endless love, understand, and support.

Abstract of Thesis

The vehicle platoon can significantly slow down the traffic congestion and improve the traffic efficiency, the driving safety and fuel economy. It is an important means to solve traffic safety problems. The monocular cameras can be used in the vehicle platoon systems for determining the distance to a front vehicle. The distance is used in the vehicle controller to control the vehicle platoon. Removing the background is an effective way to improve the accuracy and speed of detecting the vehicle or license plates. The cameras are installed on the vehicle, so the object detection algorithms for dynamic scenes is crucial. The vison-based front distance measurement method is effective in creating low-cost systems or as a failsafe method to provide system reliability under sensor failure. Since vehicle platoon control system has uncertain and nonlinear, it is significant to study the cooperative control problem of uncertain nonlinear vehicle platoon systems. The consensus and containment are two fundamental problems in cooperative control. The main research works of this dissertation are summarized as follows:

A fast and effective moving object detection method for a moving camera is proposed. The global motion is estimated through tracking the grid-based key points using optical flow. After the motion compensation, the background model, candidate background model and candidate age are used for the background modelling. Then the local pixel difference and the consistency of local changes between the current frame and the background model are used for the background subtraction. The lighting influence threshold and the local pixel difference between the current frame and two previous aligned frames are used to reduce the lighting influences. Finally, Gaussian filter, connected-components analysis, erosion and dilation are used to refine the results. The performance evaluation shows that this proposed method works very fast in real time and has competitive results compared with others in the public dataset.

A front vehicle distance measurement method is proposed. It locates the license plate of the front vehicle in the moving object area after removing the background. The license plate is positioned by extracting the texture in the vertical direction of the license plate area. The front vehicle distance is estimated from the observed license plate height by using the logarithmic equation with three fixed parameters.

An adaptive consensus control method of second-order nonlinear systems with output constraints is proposed. It guided by an active leader. Backstepping design combining with fuzzy approximate technique is employed in the consensus control design. A command governor is introduced to generate an optimal virtual control signal, which is able to balance the virtual control law and the actual velocity signal. The optimization problem is solved via a recurrent neural network. A barrier Lyapunov function is utilized to guarantee the uniformly ultimately bounded control of the closed-loop systems without violating the output constraints.

A containment control method for uncertain nonlinear multi-agent systems is proposed. The followers are governed by nonlinear systems with unknown dynamics while the multiple leaders are neighbors of a subset of the followers. Fuzzy logic systems (FLSs) are used to identify the unknown dynamics and a distributed state feedback containment control protocol is proposed. This result is extended to the output feedback case, where observers are designed to estimate the unmeasurable states. Then, an output feedback containment control scheme is presented. The developed state feedback and output feedback containment controllers guarantee that the states of all followers converge to the convex hull spanned by the dynamic leaders.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

GPS	Global Positioning System
SOBS	Self-Organization Background Subtraction
GMM	Gaussian Mixture Model
FLS	Fuzzy Logic System
NN	Neural Networks
RANSAC	Random Sampling Consistency
SUSAN	Smallest Univalue Segment Assimilating Nucleus
FAST	Features from Accelerated Segment Test
BRIEF	Binary Robust Independent Elementary Features
SURF	Speeded Up Robust Features
SIFT	Scale Invariant Feature Transform
RAM	Random Access Memory
PTZ	Pan Tilt Zoom
QVGA	Quarter Video Graphics Array
PAWCS	Pixel-based Adaptive Word Consensus Segmenter
MBS	Multimode Background Subtraction
IUTIS	In Unity There Is Strength
EFIC	Edge based Foreground background segmentation with Interior Classification
MoG	Mixture of Gaussians
RGB	Red Green Blue
LIDAR	Light Detection and Ranging
UUB	Uniformly Ultimately Bounded
LMI	Linear Matrix Inequality

Chapter 1 Introduction

1.1 Research Background and Objectives

The vehicles traveling on the road are not isolated individuals, but are coupled to other vehicles into a vehicle group system. In the vehicle group system, the driver, the vehicle, the road and other factors are mutually constrained to form a complex generalized dynamic system. The queuing of vehicles can significantly slow down the traffic congestion and improve the traffic efficiency, the driving safety and fuel economy. It is an important means to solve traffic safety problems. It has a good application prospect.

The study of vehicle platoon control began in the PATH project [1] in California in the 1980s. Then it is developed in the SARTRE project in Europe [2]. “queuing” is the formation of adjacent vehicles in a single lane. It automatically adjusts the longitudinal motion state of the vehicle based on the information of adjacent vehicles and finally achieves a consistent driving speed and desired formation [3].

Figure 1.1 The road test of the vehicle platoon in PATH project and SARTRE project.

The vehicle platoon can be regarded as a special multi-agent system which is a dynamic system composed of multiple single vehicle nodes. The individual vehicles are controlled through the information interaction between the nodes and then couples each other. An individual vehicle makes decisions based on the neighborhood information. But it can lead to the synchronization and consistency of the overall behavior. The relative state difference between any two agents tends to zero. The vehicle platoon control is considered a multi-agent cooperative control. The multi-agent system cooperative control designs the distributed control protocol to achieve a common global goal by using a limited local

communication and information exchange between the agents.

The consensus control is one of the basic problems of cooperative control of multi-agent systems. The consensus control uses the local communication between the agents. The state or output of the multi-agent system eventually tends to be consistent.

The containment control is another basic problem in cooperative control of multi-agent systems. The spatially distributed agents eventually gather in a desired target area through local information exchange. Multiple leaders in this system determine the desired target area.

Various vehicle sensing methods have been proposed, such as installing a magnetic reference track for the vehicles to follow, utilizing camera-based approaches, lidar rangefinders, and GPS or differential GPS to determine absolute position. A fixed lidar module is usually used to determine distance to the front vehicle. The monocular cameras also can be used in autonomous systems for determining the distance to a front vehicle. The distance is used in the vehicle controller to control the vehicle platoon. The monocular detection method relies on vehicle detection, specifically license plate detection, to measure the distance to the vehicle.

Although the video-based distance measurement method does not have better accuracy than the radar and lidar, but the cost problem must be considered in practical applications. With the development of cameras, its accuracy continues to increase and the price continues to decrease. The video-based distance measurement method is easy to operate and install. This method is effective in creating low-cost systems or as a failsafe method to provide system reliability under sensor failure.

For the front vehicle distance measurement, removing the background is an effective way to improve the accuracy and speed of detecting the vehicle or license plates. So the moving object detection is used to remove the background and extract the moving object area from the background image. The cameras are installed on the vehicle, so the object detection algorithms for dynamic scenes is crucial.

1.2 Related Works

In this section, some papers related to topics covered in the dissertation are reviewed.

1.2.1 Moving Objects Detection

The common object detection algorithms include inter-frame difference, optical flow and background subtraction. The inter-frame difference [4] is simple in calculation and is less sensitive to lighting. It can adapt to various dynamic environments and has good stability. But it can't extract the complete area of the object, only the boundaries can be extracted. The optical flow [5] uses each pixel vector in each frame to detect the motion area. The pixel vector magnitude and direction of foreground and background are different. It can effectively extract and track the moving object, but the calculation is more complex and the anti-noise performance is poor. If a few feature points in the image are selected to calculate the optical flow information, the complexity can be effectively reduced. The background subtraction [6] uses the differences between the current frame and a reference background model for foreground segmentation. It needs to establish an efficient background model and resolve the lighting change, noise and background update problems.

The mixed Gaussian model [7] can detect the object in complex scenes, but it is sensitive to the noise and has more calculation costs. Although the algorithm based on Codebook has good performance, it is sensitive to the change of lighting. Self-Organization Background Subtraction (SOBS) [8] has good robustness for the illumination. But if the input image is large, the calculation increases accordingly. For the unknown background prior model and foreground objects dynamics, the supervised object detection method [9][10] is not suitable for the unknown moving object segmentation. It is mainly used for known object detection. It has high computational costs and is not suitable for the mobile platforms with limited onboard resources.

For detecting moving objects for a moving camera, an adaptive motion model based on the camera motion pre-information can effectively compensate the camera motion and more accurately detect moving objects. But it has more processing time. In [11], the existence of a dominant plane and used various constraints to estimate camera motion is assumed. In [12], multiple homography matrices for a multi plane representation of 3D scene is used. It is used for freely camera moving in any directions and depth. But its computation is very complex. In [13], an adaptive disparity-based multi-homography

algorithm is proposed. It compensates for the global background motion over a wide range of backgrounds. It generates adaptive multi-homography matrices by motion grouping. In [14], a moving object extraction structure based on the motion compensation and hysteresis threshold is proposed, It constitutes space and time compensation.

Some background subtraction methods need a background model with the scene provided in advance, but it is not suitable for the tasks that process a new scene. Some background subtraction methods with high accuracy also do not work in real time. In [15], an adaptive background model registration algorithm by estimating homography motion is proposed. It can detect objects in moving cameras with complex backgrounds. In [16], a planar homography is used to compensate the camera motions. It can create a panorama background for whole video frames. In [17], a motion-compensated local to global background model is proposed. It detected the foreground regions using the moving cameras. GMM is used for superpixel-level background to refine the foreground area. In [18], a scene conditional background updating method is proposed, it adaptively established the background according to the scene change.

1.2.2 Vehicle Distance Measurement

The front vehicle distance measurement based on the monocular camera usually has two methods. The first method is based on the angle and trigonometric function to calculate the distance. It needs to detect the angle between the vertical line from the camera to the ground and the line from the camera to the light or bottom of the front vehicle. Then the distance from the front vehicle is detected according to the height from the camera to the ground and the internal parameter of the camera. This method requires the vehicle to be in the same horizontal plane. The measurement of the angle usually has error, so the detection result is not ideal. The second method is based on the principle of camera imaging. It calculates the relation function between the actual front vehicle distance with the pixel distance of the front vehicle distance, the pixel distance between the front vehicle lights [19], the pixel width of the license plate on the front vehicle [20]. This method has less computational complexity, because this method only needs calculate the fixed function parameters in advance.

In this dissertation, the front vehicle distance measurement is achieved by locating the license plate of the front vehicle in the moving object area after removing the background. The license plate locating is usually based on unique license plate features, such as edges, shapes, textures, colors, etc. The shape of the license plate is usually a rectangle with a fixed aspect ratio. To find the rectangle in the image, the edge in the image is first extracted. In [21], the Sobel filter is used to detect the edge. The adjacent pixels in the horizontal and vertical direction are subtracted. The edge contour is highlighted. The texture of other regions is weakened. At this time, the license plate area presents a rectangle. Then the rectangle with a specific aspect ratio is found to complete the locating of the license plate. In [22], the connected component analysis method based on the global information is used. After scanning the binary image of the vehicle, each pixel is marked in different regions based on the connectivity of the pixel. Then the aspect ratio of the license plate is used to filter the connected area and locating the license plate. This method has some limitations. It is usually used in the low-resolution image.

In [23], the texture feature of the license plate area is used for license plate locating. Since there are characters in the license plate area, the color difference between the character color and the license plate background color is obvious. After the scan line is set, the pixel value change in the binary image passing through the scan line is recorded. It used to determine whether the number of changes matches the number of characters in the license plate area. Then the license plate is located. In [24], a method with sliding concentric window is proposed. The license plate area is regarded as an irregular texture area. When the scanning window is located in an area where the texture changes significantly, this area is regarded as a license plate area. Each country's license plate has some fixed colors. In [25], all the pixels of the image are classified into 13 categories according to the HLS spatial features. The pixels that match the license plate color are classified into the license plate area pixels.

The locating method of the license plate should be determined according to the different environments and different image resolutions. For example, when the image is blurred, the connected component analysis has a good result. When the image is clear, the method of using the texture to locate the license plate has a better result. In actual license plate locating, most methods combine several features to improve the accuracy of locating.

1.2.3 Consensus Control

Consensus is a fundamental problem in coordination of multi-agent systems, whose aim is to drive all agents to reach an agreement using the local information exchanged among individuals. Most of the consensus approaches can be categorized into leaderless consensus and leader-following consensus. As a pioneering work, the reference [26] proposed an average consensus algorithm for networked first-order integrators using graph theory and frequency domain analysis method. Since then, some seminal works on consensus of multi-agent systems have been established [27][28][29].

The above-mentioned results mainly focus on the consensus control of networked first-order systems. In reality, many mechanical systems are governed by both position and velocity states, such as robots, autonomous surface vehicles and aircrafts. Therefore, it is significant to take the consensus control of networked second-order systems into consideration. From the existing results on consensus, it can be seen that the research of multi-agent systems with second-order dynamics has gained an extensive attention. In [30], leader-following consensus algorithms based on distributed observers were presented under a switching interconnection topology. In [31], some necessary and sufficient conditions, which were associated with the eigenvalues of the Laplacian matrix, were provided for consensus of networked second-order systems under a directed topology containing a spanning tree. In [32], the quantized consensus problem was studied for networked second-order systems via a sampled data method. In [33], a finite-time leader-following consensus control was designed for multi-agent systems with second-order dynamics. In [34], robust leader-following consensus control schemes were designed for second-order systems with non-identical nonlinearities in the presence of external disturbances. Furthermore, adaptive control technique was utilized to deal with the consensus control problem of second-order systems, which was used to identify the nonlinear dynamics of agents or control gains [35][36].

In particular, fuzzy logic systems (FLSs) or neural networks (NNs) based adaptive control technique has been widely applied to handle the nonlinear dynamics of the control systems [37][38][39]. The function approximation based adaptive control technique has been successfully extended to the consensus control of multi-agent systems with unknown dynamics [40][41]. An NN based adaptive

leaderless consensus approach and an NN based adaptive leader-following consensus approach were respectively, proposed for first-order systems in the presence of unknown dynamics and external disturbances [27][28]. Then, some adaptive FLS or NN based consensus control methods were developed for various classes of nonlinear multi-agent systems with unknown dynamics, such as second-order systems, high-order systems and strict-feedback systems [42][43]. It is worth pointing out that the existing consensus control approaches were developed without considering the constraints of the multi-agent systems.

Note the fact that many practical control systems are subject to constraints in the form of physical stoppages, saturation, or performance and safety specifications. The control performance might be influenced if the effect of the constraints is omitted [44]. Consensus control for multi-agent systems subject to constraints was addressed in [45]-[50]. In [45], the model predictive control was applied to deal with the consensus control problem of linear multi-agent systems subject to input constraints. In [46], consensus control based on the low gain feedback technique was developed for linear multi-agent systems with input constraints. In [47], finite-time consensus algorithms were proposed for multi-agent systems with double-integrator dynamics in the presence of input constraints. Furthermore, some results were extended to consensus control of multi-agent systems considering input constraints as well as actuator faults [48]. However, these approaches cannot solve the consensus control problem of multi-agent systems with output constraints. Output constraints exist in some practical systems, such as robotic system, marine surface vessel, flexible crane system, and so on [51][52][53]. In [49], a discarded consensus algorithm was designed for consensus of linear systems which can ensure the state estimate of each agent within a convex set. In [50], an adaptive consensus control algorithm was designed for first-order nonlinear systems with output constraints. Therefore, the consensus control problem for nonlinear multi-agent systems considering output constraints should be further investigated.

1.2.4 Containment Control

In many practical systems, multiple leaders are universal and have extensive applications. In the multi-agent systems, some agents are denoted as leaders, and the others are denoted as followers.

Moreover, the leaders only send information to some of the followers, and the followers can get at least one leader's information from a directed path. Containment control aims at guiding the states or outputs of the followers to converge to a convex hull formed by the multiple leaders using a distributed control protocol. Containment control problem has many applications. For example, a large amount of robots have a movement to a target while only a few of them equipped with necessary sensors are able to detect the obstacle. In this scenario, these robots are labeled as leaders, whereas others are marked as followers. Therefore, the followers can enter into the safe region which is spanned by the leaders and safely reach a target. Recently, distributed containment control problem has been investigated and obtained numerous research results [54]-[62]. Containment control strategies were proposed for multi-agent systems with single-integrator [54][55], double-integrator [56][57] or general linear dynamics [58]. However, the reported methods can only deal with the containment control problem of linear multi-agent systems. By now, there have been some results on containment control for nonlinear multi-agent systems in [59]-[62]. It should be noted that the proposed containment controllers required each agent satisfying Lagrangian dynamics with known nonlinearities [59][60], linearly parameterized nonlinearities [61] or unknown nonlinearities [62]. Therefore, containment control problem for uncertain nonlinear multi-agent systems needs to be further investigated.

1.3 Thesis Contributions

The dissertation offers the following contributions:

1. A background modelling method is proposed based on the background model, candidate background model and candidate age. The adaptive threshold is used for the age of the candidate background. This adaptive background model can remove more noise and extract more complete objects for the moving camera. A background subtraction method is proposed based on the local pixel difference and the consistency of local changes between the current frame and the background model. It can effectively reduce the influences of the hole and shadow.

2. A front vehicle distance measurement method is proposed. It locates the license plate of the front vehicle in the moving object area after removing the background. The license plate is positioned by

extracting the texture in the vertical direction of the license plate area. The front vehicle distance is estimated from the observed license plate height by using the logarithmic equation with three fixed parameters.

3. A novel adaptive fuzzy leader following consensus control method for output-constrained second-order nonlinear systems is proposed. It focusses on the multi-agent systems subject to output constraints. A barrier Lyapunov function is utilized in the control design, which grows rapidly when system outputs approach the boundaries of the constrained region. A neurodynamic optimization method is applied to consensus control of multi-agent systems.

4. A containment control method for the nonlinear multiagent systems is proposed to drive all followers to converge to a convex hull formed by the multiple leaders. A state feedback containment control scheme for more general nonlinear multi-agent systems with unknown dynamics is designed. Besides, considering that some states in the systems are unmeasurable in practice, an output feedback containment control scheme is proposed.

1.4 Main Contents and Organization

The remainder of this dissertation is organized as follows.

Chapter 2 proposes a moving camera based moving object detection method. It is based on global motion compensation and adaptive background model. After removing the background, the moving object areas are used for the vehicle distance measurement.

Chapter 3 proposes a front vehicle distance measurement method. The license plate is positioned by extracting the texture in the vertical direction of the license plate area. The front vehicle distance is estimated from the observed license plate height by using the logarithmic equation with three fixed parameters.

Chapter 4 describes the dynamics modelling of the vehicle platoon. It includes the vehicle longitudinal dynamics modelling, the communication information flow, the vehicle platoon geometry and the distributed controller.

Chapter 5 proposes a novel adaptive fuzzy leader following consensus control method for output-

constrained second-order nonlinear systems.

Chapter 6 proposes a containment control method for the nonlinear multiagent systems to drive all followers to converge to a convex hull formed by the multiple leaders.

Chapter 7 concludes our works and discuss several ideas and suggestion for future works.

Chapter 2 Moving Object Detection

A moving object detection method is proposed in this chapter. It is based on global motion compensation and adaptive background model. The results are the candidate vehicle areas. In these areas, the license plate can be detected and be used to calculate the inter-vehicle distance in chaper 3. This distance is the longitudinal control inputs in vehicle platoon system.

The background model is defined by two consecutive frames. First the grid-based key points are selected in the previous frame. Then the optical flow of each key point is calculated to obtain the corresponding point in the current frame. Using the corresponding key points in the current frame and the previous frame, the homography matrix (global motion) between two consecutive frames is calculated by Random Sampling Consistency (RANSAC [63]) which estimate the best transformation of two corresponding key points. It is used to align two previous frames and the background models with the current frame. Then the background model, candidate background model and candidate age are used for the background modelling. The local pixel difference and the consistency of local changes between the current frame and the background model are used for the background subtraction. Finally, some object segmentation methods are used to refine the results. The framework of the moving object detection method is shown in Figure 2.1.

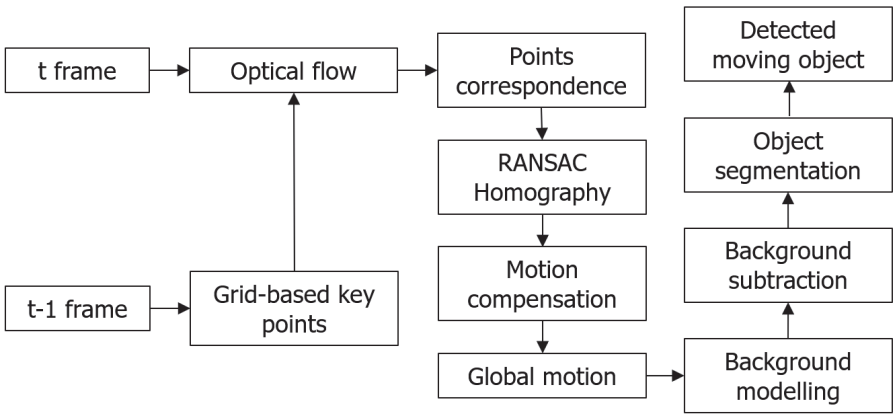


Figure 2.1 The framework of the moving object detection method.

2.1 Motion Compensation

The motion compensation estimates the homography matrix between two consecutive frames. The key points in the previous frame are obtained through manual selection, which uses less processing time. The grid-based key points instead of the key point detector, such as SUSAN [64], FAST [65], or Harris angle [66]. For the example in Figure 2.2, 16×16 key points are selected with evenly distributed in each row and each column. The number of key points is based on the accuracy of the homography matrix and the speed of calculation.

Figure 2.2 An example of optical flow result.

The optical flow of each key point in the previous frame x_{t-1} is calculated to obtain the corresponding point in the current frame x_t. Some feature similarity methods such as BRIEF [67], SURF [68], and SIFT [69] can be used to instead of the optical flow from x_{t-1} to x_t. But the pyramidal Lucas-Kanade optical flow [70] has less processing time than the feature-based approach in this case, so it is used to solve the key points matching problem.

This algorithm estimates the key points displacement by using the neighborhood of key points to solve an over-constrained equations system. It can detect larger motions by neighborhood window at larger scales of the image Gaussian pyramid. It tracks the key points at larger scales of the pyramid, and refines the motion displacement at its lower levels iteratively based on least square minimization. So, it can minimize the violations of assumptions and track faster and larger motions.

As shown in Figure 2.2, the green lines are the optical flow between two consecutive frames. All key point pixel in the previous frame are transformed into the location in current frame by the homography matrix. The magnitude and direction of the optical flow are indicated by the length and direction of the line respectively.

$$x_t = H_{t-1} x_{t-1} \tag{2.1}$$

The homography matrix H_{t-1} is the transformation matrix from x_{t-1} to x_t. It is defined by (2.1) and can be solved by a minimum of 6 pairs of key points. Since the point may be selected by mistake, the global motion may not be represented by the calculated homography matrix. The wrong estimation of H_{t-1} can be avoided by RANSAC. A few key-points are used to calculate a candidate homography matrix $\hat{H}_{t-1}$. Then $\hat{x}_t$ is obtained by using $\hat{H}_{t-1}$ to transform the key points. If the error in (2.2) is sufficiently small, $\hat{H}_{t-1}$ can be set as the best matrix H_{t-1}. Otherwise, the process is repeated using another set of key points.

$$e = \sum_{x \in x_t, \hat{x} \in \hat{x}_t} \left\| x - \hat{x} \right\|_2 \tag{2.2}$$

The image in previous frame I_{t-1} can be aligned with I_t by the homography matrix H_{t-1}. This transformation is shown in (2.3).

$$I_t(x_t) = I_{t-1}(H_{t-1} x_{t-1}) \tag{2.3}$$

The combined image with previous frame and current frame is shown in Fig. 3, the green box indicates the current frame, the opaque pixel indicates the overlapping area between the current frame and the compensated image in the previous frame. The semi-transparent area located outside the green box area indicates the non-overlapped area of the previous frame. After the frames are aligned, a part of the moving object in the scene can be shown by the frame difference between the two frames. The background models are transferred from the previous frame to the current frame by this motion compensation process.

Figure 2.3 The combined image with previous frame and current frame.

2.2 Background Modeling

Two background models are presented. The true background model M is used for the background subtraction. The candidate background model C is used to define a sufficiently stable pixel as a background model by the candidate age α. The background modeling strategy is shown in Figure 2.4.

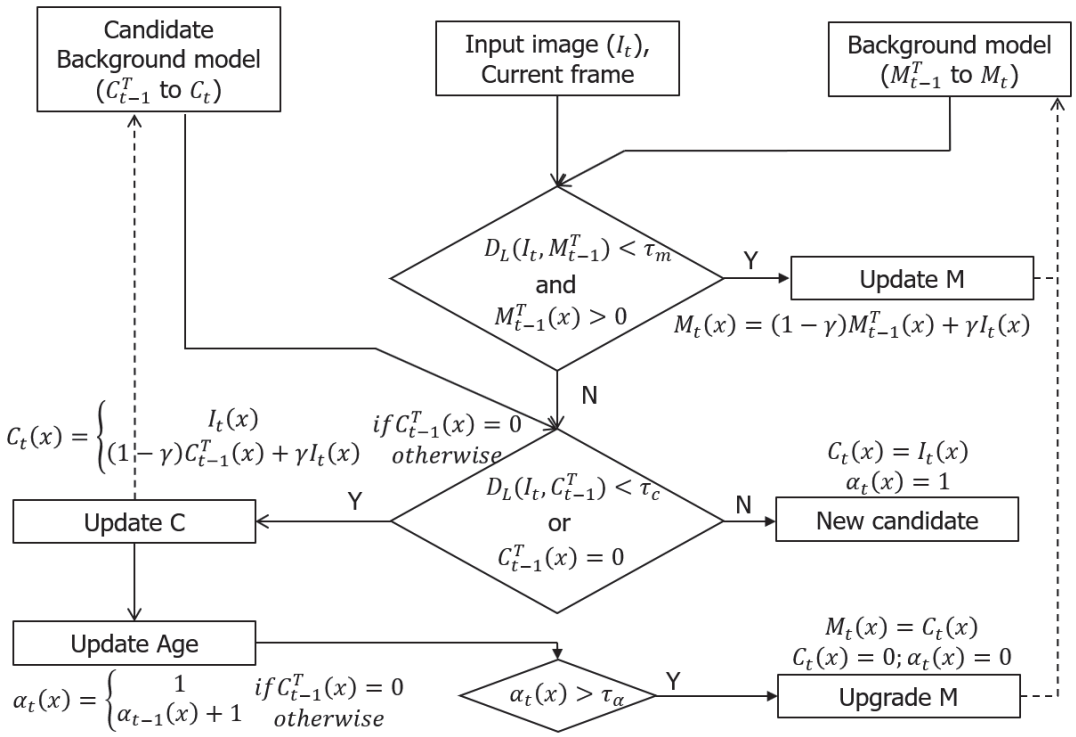


Figure 2.4 The background modelling strategy.

The background pixels in the candidate background model and the background model are set to uninitialized. The homography matrix H_{t-1} can be calculated by pyramidal Lucas-Kanade optical flow based on the key point of the current frame and the previous frame. Then the background model and candidate background model are transferred from the previous frame to the current frame and are updated by the background modeling strategy.

$$M_{t-1}^{T} = H_{t-1} M_{t-1} \tag{2.4}$$

$$C_{t-1}^{T} = H_{t-1} C_{t-1} \tag{2.5}$$

The pixel similarity between the input frame and two background models are defined by the local difference (D_L). It is used for defining which model is updated. The local difference between two frames is calculated by the difference average of 9-neighborhood of the corresponding pixels (x, y). It can effectively reduce the noise in object detection process without using much additional time.

$$D_L\left(I(x,y), M(x,y)\right) = \frac{1}{9}\sum_{i=x-1}^{i=x+1}\sum_{j=y-1}^{j=y+1}\left|I(i,j) - M(i,j)\right| \tag{2.6}$$

When the local difference between $I_t(x)$ and $M_{t-1}^{T}(x)$ is less than a given threshold τ_m , and $M_{t-1}^{T}(x)$ is in an initialized state, the background model $M(x)$ will updated with the learning rate γ .

$$M_t(x) = (1-\gamma) M_{t-1}^{T}(x) + \gamma I_t(x) \tag{2.7}$$

Otherwise, the second condition is checked. When the local difference between $I_t(x)$ and $C_{t-1}^{T}(x)$ is less than a given threshold τ_c or $C_{t-1}^{T}(x)$ is in an uninitialized state, $C(x)$ and $\alpha(x)$ will be updated. The candidate background and the age are used to check whether the pixel is sufficiently stable. If the pixel is sufficiently stable, the pixel has a high chance to be a background pixel. When the candidate background age is greater than a given threshold τ_α, the background model is updated by the candidate background model. $\alpha(x)$ is also set to zero and the candidate background model can accept a new candidate background. If neither of the above conditions are satisfied, $C(x)$ is updated by $I_t(x)$, so the pixel is treated as a fluctuating point and a candidate background.

The adaptive threshold is used for the age of the candidate background. The threshold is defined as the exponentially decayed value according to the average magnitude of optical flow. This process needs

to accommodate the movement speed of the camera. The faster update rate provides many noises since the background model is not stable. However, if the update rate is slow, it will not be able to deal with the fast movement of the camera.

$$\tau_\alpha = \begin{cases} \mu e^{-\sigma \bar{f}} & if \quad \mu e^{-\sigma \bar{f}} > 2 \\ 2 & otherwise \end{cases} \tag{2.8}$$

μ is the maximum value, σ is the exponentially decaying speed, $\bar{f}$ is the average optical flow. The average optical flow is shown in Figure 2.5.

Figure 2.5 Illustration of the average optical flow.

The example of the comparison for without adaptive threshold and using adaptive threshold are shown in Figure 2.6 and Figure 2.7. The original frame and the corresponding segmentation results are shown in (a) and (b). The background model and the detected moving object are shown in (c) and (d). In this scene, the camera movement is quite fast. Without adaptive threshold, there are many parts of the moving object that cannot be detected. But using the adaptive threshold, most of the parts of the vehicle are correctly detected.

(a)

(b)

(c) (d)

Figure 2.6 The example results without adaptive threshold.

(a) (b)

(c) (d)

Figure 2.7 The example results using adaptive threshold.

2.3 Background Subtraction

Using two previous homography matrixes (H_{t-1} and H_{t-2}), the previous two frames are aligned with the current frame. Then the local difference is calculated between them. The local difference between previous aligned frame and the current frame is also calculated. The frame difference can effectively adapt to the changes of lighting and other environments. The registration error in motion compensation can be minimized by aligning the previous two frames with the independent current frame.

$$I_{t-1}^{T} = H_{t-1} I_{t-1} \tag{2.9}$$

$$I_{t-2}^{T} = H_{t-1} H_{t-2} I_{t-2} \tag{2.10}$$

The local difference between the current frame and the background model is calculated. The background difference method does not exist "double shadow" and "hole" phenomenon. So, these two

kinds of local differences can make up for each other and can get more accurate and complete results.

The shadow area lacks illumination, its pixel change values between the current frame and the background model are generally consistent locally. But the pixel change values of the moving object are inconsistent locally. So, S_L is used to define the consistency of local changes between the current frame and background model. It is used to eliminate the shadow areas during the background subtraction.

$$S_L = \frac{1}{9}\sum_{i=x-1}^{i=x+1}\sum_{j=y-1}^{j=y+1}\left(I(i,j) - M(i,j) - D_L\left(I(x,y), M(x,y)\right)\right)^2 \tag{2.11}$$

The mask of foreground F is calculated by the given thresholds τ_s and the lighting influence threshold τ_l in (2.12).

$$F(x,y) = \begin{cases} 1 & L \geq \tau_{s1} + \tau_l \\ 0.5 & \tau_{s1} + \tau_l > L \geq \tau_{s2} + \tau_l \\ 0 & otherwise \end{cases} \tag{2.12}$$

$$L = D_L\left(I_t, M_t\right) + \lambda_1 S_L + \lambda_2 D_L\left(I_{t-1}^T, I_{t-2}^T\right) + \lambda_3 D_L\left(I_t, I_{t-1}^T\right) \tag{2.13}$$

$$\tau_l = \rho\frac{1}{N_A}\sum_{(x,y)\in A}\left|I_t(x,y) - I_{t-1}^T(x,y)\right| \tag{2.14}$$

N_A is the total number of pixels in the area A. ρ is the inhibition coefficient of lighting. The lighting influence threshold τ_l expresses the change of lighting in the entire frame. If the lighting in the scene changes significantly, this threshold will increase significantly. This effectively suppresses the influence of light changes on the detection results of moving objects.

Two kinds of thresholds are used to define the foreground with high confidence which means the difference between the background model and current frame is high, and foreground with lower confidence which means the difference is not too big. Usually, the foreground with low confidence is detected at the border and the shadow regions. It is shown with gray color in the results. It can be effectively refined by the subsequent methods. The foreground with high confidence is shown with white color in the results. The background subtraction strategy is shown in Figure 2.8.

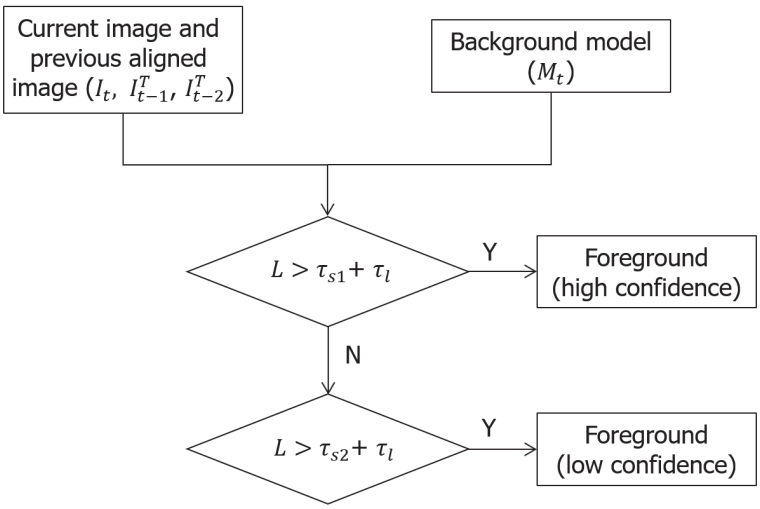
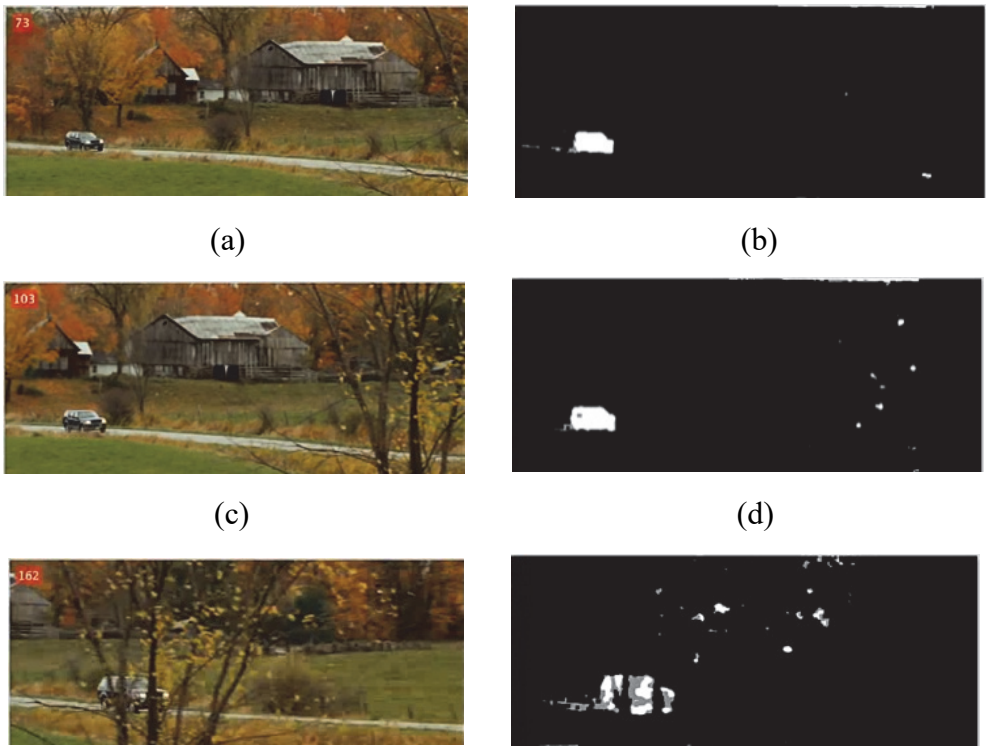


Figure 2.8 The background subtraction strategy.

The background subtraction results in the moving camera is shown in Fig. 9. The vehicle is moving from the left to the right in the scene and the camera is tracking its movements. Three captured frames are shown in Figure 2.9. (a), (c) and (e). The corresponding segmentation results are shown in Fig. Figure 2.9. (b), (d) and (f). The background model and the detected moving object are shown in (g) and (h). The lower threshold is used for the weak foreground. The higher threshold is used for the strong foreground.



(e) (f)

(g) (h)

Figure 2.9 The background subtraction results in the moving camera.

The background model in the last frame is shown in Figure 2.9. (g). The result shows that the right area of the background model is black, this area is not available because of the moving camera. When the candidate background is sufficiently stable, the background model will be updated by the candidate background. The detected moving object is shown in Figure 2.9. (h). The result shows that the non-occluded part of the vehicle is detected.

Multiple independently moving objects are segmented by 3×3 median filter and a 5×5 Gaussian filter. The edge can be refined to improve algorithm performance. The kernel size must be large enough to cover most of the segmented blobs and small enough to avoid multiple blobs overlap. The connected-components analysis is used to improve the background elimination and cluster the closely moving foreground regions to a unified object. Then the erosion and dilation are used to reduce the separate noises and fill the holes in the segmented blobs, respectively.

2.4 Experiment Results

The experiment uses Matlab on Intel i7 processor with 8G RAM. The grid size used for key point is set as 32×32. The input image is post-processed by a 5×5 Gaussian filter and a 3×3 median filter. The public dataset from ChangeDetection [71] website is used for the performance evaluation of this method. The following 4 sequences in the category PTZ dataset are used for the moving camera, *zoomInZoomOut*, *intermittentPan*, *continuousPan* and *twoPositionPTZCam*. The detection results on the PTZ dataset are shown in Figure 2.10. The scene from continuousPan is shown in the first row, the scene from intermittentPan is shown in the second row, and the scene from twoPositionPTZCam is shown in

the third row.

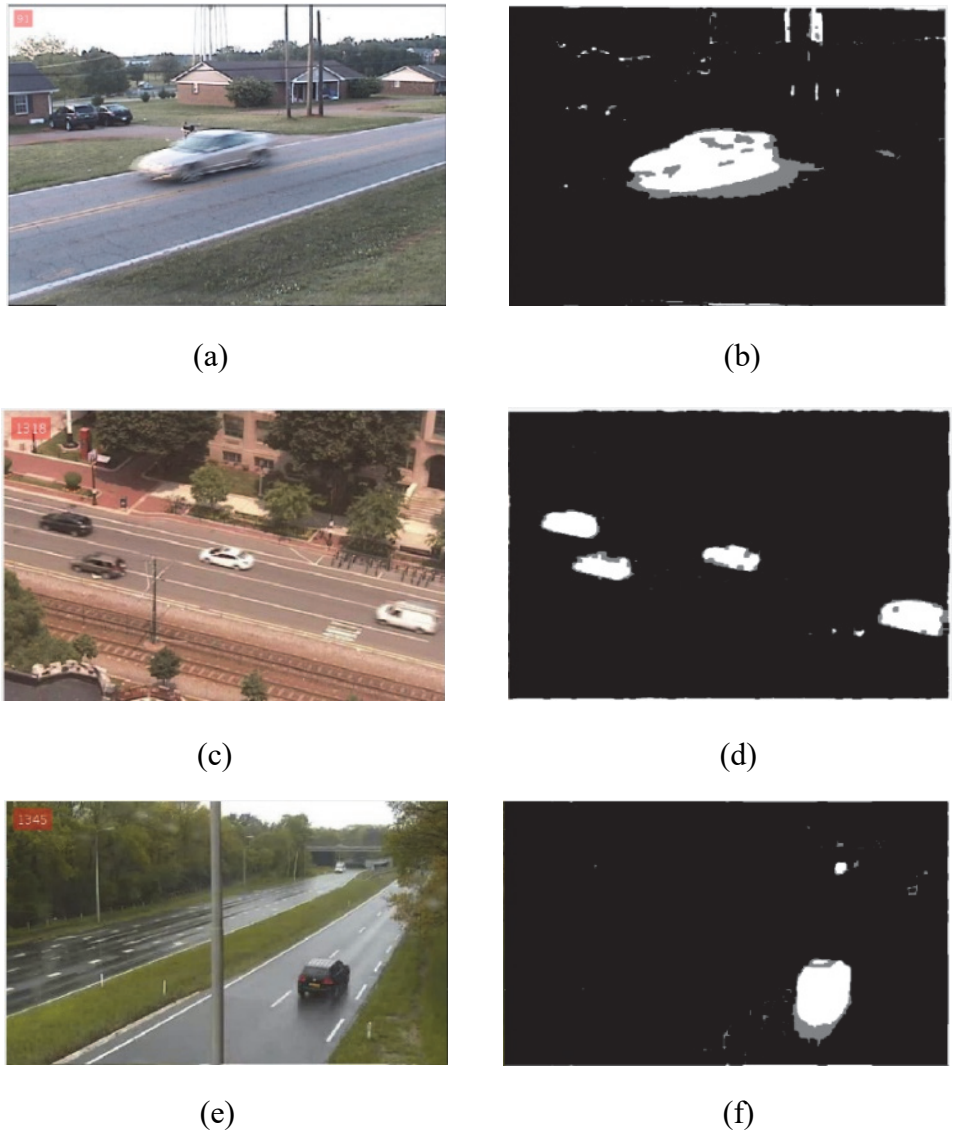


Figure 2.10 The detection results on the PTZ dataset.

The detection results on the zoomInZoomOut sequence are shown in Figure 2.11. The scene when the camera zooms out is shown in the first row. The scene when the camera zooms in is shown in the second row.

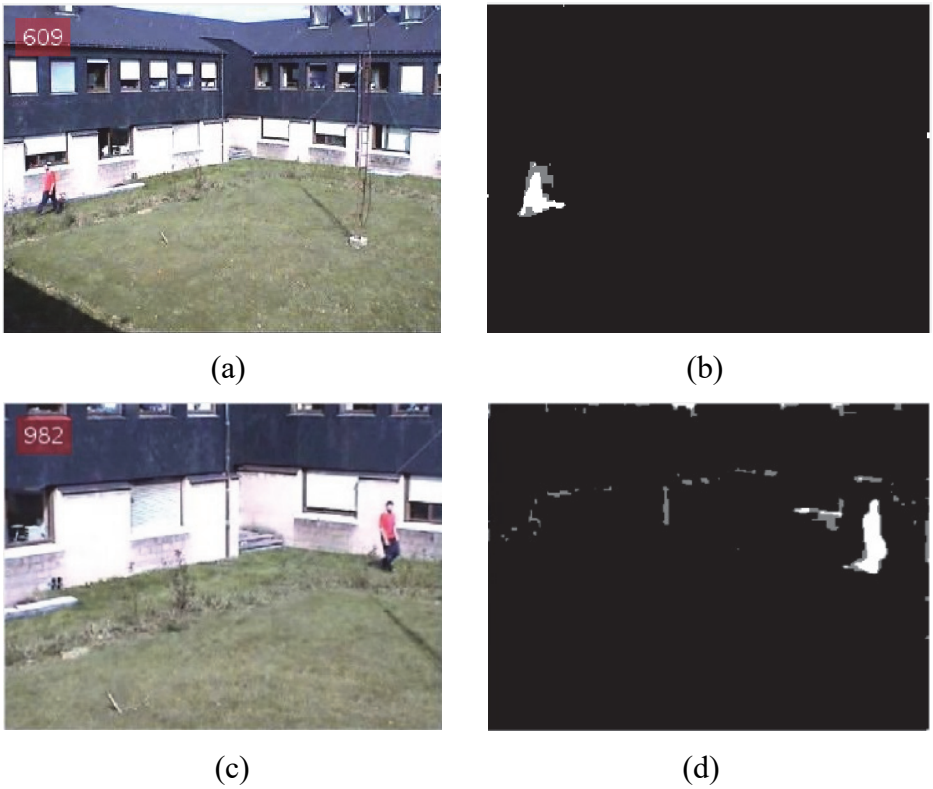


Figure 2.11 The detection results on the zoomInZoomOut sequence.

The *continuousPan* sequence has some scenes from consecutive pan cameras on one side of the road. The camera alternately turns left and right at low speed. The *intermittentPan* sequence has some scenes from a camera mounted at the top of a road. The camera is still for a while and then pans in a slow motion to capture another road view. The *twoPositionPTZCam* sequence has some scenes from a camera that monitors the traffic of road. The camera is moving in one direction and then suddenly changes the direction to the other. The *zoomInZoomOut* sequence has some scenes from a camera that captures images while zooming in and zooming out.

This proposed method works well on this PTZ dataset. It can adapt to the zooming in and zooming out of the camera. It can adapt to the situation with flaring effect from multiple moving objects and the sun. It also can deal with the sudden change of the camera angle.

The processing speeds on this PTZ dataset are shown in Table 2.1. According to these experimental results, the proposed method can work well in real time and works at 51 fps for the video with QVGA resolution.

Table 2.1 The processing speeds on the PTZ dataset.

Sequence name	Image size	Fps
continuosPan	480 × 704	14.8
intermittentPan	368 × 580	22.7
twoPositionPTZCam	340 × 570	24.6
zoomInZoomOut	240 × 320	51.2

The performance measures considered in this work include Precision (Pr), Recall (Re), Specificity (Sp), False positive rate (FPR), False negative rate (FNR), Percentage of wrong classification (PWC) and F-measure (F1), which are defined as follows:

$$\mathrm{Pr}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}},\quad \mathrm{Re}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}},\quad \mathrm{Sp}=\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} \tag{2.15}$$

$$\mathrm{FPR}=\frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}},\quad \mathrm{FNR}=\frac{\mathrm{FN}}{\mathrm{TP}+\mathrm{FN}} \tag{2.16}$$

$$\mathrm{PWC}=\frac{100\times(\mathrm{FN}+\mathrm{FP})}{\mathrm{TP}+\mathrm{FN}+\mathrm{FP}+\mathrm{TN}},\quad \mathrm{F1}=\frac{2\times\mathrm{Pr}\times\mathrm{Re}}{\mathrm{Pr}+\mathrm{Re}} \tag{2.17}$$

TP (true-positive) is the number of pixels correctly detected as foreground, FP (false-positive) is the number of pixels incorrectly detected as foreground, and FN (false-negative) is the number of pixels incorrectly detected as background. TN (true-negative) is the number of pixels correctly detected as background.

The quantitative performance comparison is shown in Table 2.2. The data of other comparison methods are from the ChangeDetection website. The results show that the proposed method has better results on F1 than other methods except EFIC. But PWC in EFIC is too high and not better than other methods. Other performance measures on the proposed method do not differ much from other results. So this method can detect better complete information of the object. The better Pr value indicates that it is insensitive to noise and it is not easy to locate the noise point as a moving pixel. The better F1 value indicates that fewer object information are discarded during detection. The lower PWC value indicates that it has less double shadow and hole.

Table 2.2 Performance comparison.

Algorithm	Pr	Re	Sp	FPR	FNR	PWC	F1
PAWCS [72]	0.473	0.698	0.991	0.009	0.302	1.116	0.462
MBS [73]	0.540	0.597	0.996	0.004	0.403	0.585	0.552
IUTIS-5 [74]	0.383	0.675	0.990	0.010	0.325	1.217	0.428
CwisarDRP [75]	0.320	0.754	0.988	0.012	0.246	1.298	0.429
EFIC [76]	0.528	0.918	0.922	0.078	0.082	7.871	0.584
MBSV0[77]	0.499	0.577	0.995	0.006	0.423	0.782	0.512
SharedM [78]	0.312	0.797	0.979	0.021	0.203	2.217	0.386
SuBSENSE [79]	0.284	0.831	0.963	0.037	0.169	3.816	0.348
CwisarDH [80]	0.482	0.336	0.998	0.002	0.664	0.685	0.322
RMoG[81]	0.221	0.641	0.928	0.072	0.359	7.476	0.247
Proposed	0.512	0.798	0.991	0.009	0.202	1.208	0.569

Chapter 3 Vehicle Distance Measurement

The vehicles are required to mount a rear-facing license plate for law enforcement purposes. This is an easily identifiable feature on every vehicle, which can be used for vehicle distance measurement. Some drawbacks to this method are the fact that some recently bought vehicles do not have a license plate. Also, some vehicles do not mount the license plate in the center of the vehicle. Despite these exceptions, this method provides a reliable method to detect the front vehicle.

The efficacy of the license plate recognition algorithm relies on the quality of the acquired images. The main methods of license plate detection are done using boundary or edge information, connected component analysis, texture features, color features, or character features. The fastest methods are the ones relying on boundary or edge features. The biggest drawback to these methods is the sensitivity to unwanted edges in the image.

3.1 License Plate Detection

Due to the regular distribution of characters, there are rich and regular vertical direction textures in the license plate area. These textures are different from other edge information of the vehicle and have important significance for the license plate location. When the front vehicle is closer to the camera, the resolution of the vehicle image is higher. The license plate text information is obvious. So the license plate can be positioned by extracting the texture in the vertical direction of the license plate area.

First, the color image of the detected vehicle is converted to the grayscale image. Since only the texture information in the image is needed, it is time consuming to process directly on the color image. The color value of a point in an RGB image can be represented by three channel values R, G, B. The pixel gray value of the corresponding gray image is Y.

$$Y = 0.3R + 0.59G + 0.11B \tag{3.1}$$

Then, median filtering is performed on the grayscale image by using $3*3$ window. The pixel value of a certain point in the grayscale image is replaced by the middle value of the nine-pixel values in the window. The median filtering can better eliminate the salt and pepper noise.

$$I_m(i,j) = med\left\{I(i-1, j-k), (l,k \in [1,3])\right\} \tag{3.2}$$

Where I_m is the image after median filter, $I(i,j)$ is the pixel value of the grayscale image, and med is the function that takes median value.

The Sobel operator is used to calculate the vertical texture of the vehicle grayscale image. $3*3$ kernel is used for the convolution of the image.

$$I_s = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I_m \tag{3.3}$$

Where I_s is the obtained vertical direction texture image. Because the Sobel operator subtracts the adjacent pixels of the image, only the texture information is retained. But the pixel in obtained image has a small value range and the texture information is not obvious. The texture image pixel values are mapped to [0, 255]. The obvious vehicle vertical texture image I_o can be obtained

$$I_o = \left(I_s(i,j) - \min(I_s)\right) \times 255 / \left(\max(I_s) - \min(I_s)\right) \tag{3.4}$$

After projection processing for each pixel gray value of the texture information image, the range of the overall gray value increases and the edge texture is more obvious. So a grayscale image that only preserves the texture in the vertical direction of the vehicle is obtained.

Figure 3.1 Original image and grayscale image of the vehicle image.

Figure 3.2 Texture image after Sobel operation and pixel mapping.

After obtaining the grayscale image, the redundant information is further eliminated by the binarization operation. Due to the influence of illumination, the brightness between the license plates is very different. In order to apply to various illuminations, the adaptive local binarization method is used to process the image. It divides the image into $n*m$ blocks, and perform binarization operations in each image block separately. The size of this image block is set to $(2w+1)*(2w+1)$, then the threshold $T(x,y)$ of each point in the image is calculated.

$$T(x,y)=0.5\times\left(\max_{\substack{-w\le m\le w\\-w\le n\le w}} I_o(x+m,y+n)+\min_{\substack{-w\le m\le w\\-w\le n\le w}} I_o(x+m,y+n)\right) \tag{3.5}$$

Then a binarized image can be obtained by using the calculated threshold.

$$I_b(x,y)=\begin{cases}0, & if\ I_o(x,y)<T(x,y)\\ 255, & else\end{cases} \tag{3.6}$$

The binarization of the threshold instead of manually selecting a fixed threshold. This method can ignore the shadows on the license plate and finally retain the texture in the vertical direction of the vehicle.

The binary image with vertical texture is shown in Figure 3.3.

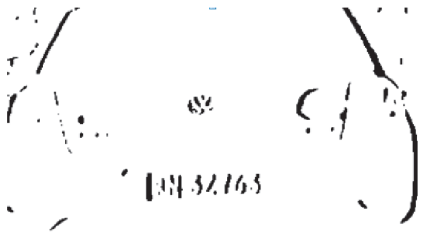


Figure 3.3 Binary image with vertical texture

The texture are some incoherent connected areas. Then morphological processing is used on these areas. In order to find the license plate contour, all the textures need to be connected to each other in the license plate area. The closing operation is iterated 8 times on the image by using 3*1 kernel. It can eliminate the small holes and fill the cracks. After the closing operation, the white textures of the license plate area are connected to each other to form a white area of the license plate. For the license plate location, the vehicle contour is redundant information. The opening operation is used to eliminate this redundant information by the 5*1 kernel. It can make the texture contour in the image smoother and eliminate the independent and fine texture. The license plate area has some features. Some filter conditions are set to find the contour of the white area and locate the license plate. The license plate has a fixed aspect ratio and area. In the vehicle image, the license plate should be in the center position area of the image.

(a) Closing operation (b) Opening operation (c) License plate positioning

Figure 3.4 Morphological algorithm in license plate detection

This method locates the license plate according to the texture of the license plate area. When the current vehicle distance is close, the resolution of detected vehicle image is high. The license plate area has obvious texture. This method is used to locate the license plate with higher accuracy. For higher resolution images, the algorithm complexity of this method is still small. The input image is a vehicle image rather than a global image, it can still meet the real-time requirements.

3.2 Vehicle Distance Estimator

The longitudinal control inputs required by each vehicle are the inter-vehicle distance and inter-

vehicle relative velocity. These parameters are generally obtained via GPS with inter-vehicle communication, a 360◦ LIDAR unit, or stereo vision camera.

The monocular camera lacks the depth information, but it is very effective at detecting an object with low computational complexity. If an object has a known size, a monocular camera can be calibrated to determine the distance to the object.

This method would not be effective under a variety of conditions, such as the size of the license plate on the vehicle is changed, the license plate is removed, or the license plate is placed in an atypical orientation.

The height and width of the plate in pixels in addition to the pixel position of the license plate can be used to estimate the distance to the front vehicle. The license plate is expected to be observable and within the range of the camera.

As the distance to an object increases, the observed size of the object decreases exponentially. So in this method, the observed license plate height H in the image decreases exponentially with the inter-vehicle distance L. The tunable parameters do not correspond directly to any physical measurement. The observed license plate height is used because the observed license plate width can vary depending on the heading of the front vehicle, but the observed height remains relatively constant with variations in the front vehicle heading.

$$H = ae^{-bL} + c \tag{3.7}$$

Therefore, the front vehicle distance is estimated from the observed license plate height by using the logarithmic equation with three fixed parameters.

$$L = \left| -\frac{1}{b} \log \frac{H - c}{a} \right| \tag{3.8}$$

The license plate dimension of small car in China is 440 mm by 140 mm. The license plate was detected and the distances were recorded at various angles (0 rad, π/2 rad) and distances (0.3 m, 3 m) from the camera. The least squares fitting is used to calculate the parameters of the distance estimators. The following parameters can be obtained. $a = 76.32$, $b = 1.513$, $c = 6.223$. The resulting curve and sample data points are shown in Figures 4.5. The accuracy based on the mean squared error is 0.09.

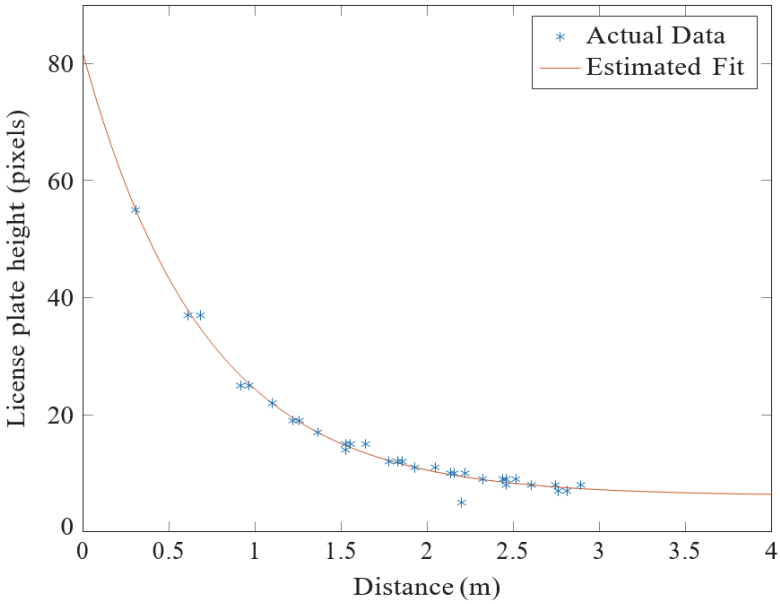


Figure 3.5 A sample fitting of the license plate distance estimator to sample points.

This method to determine the front vehicle distance can be implemented inexpensively as a redundant sensor for the LIDAR distance finder. This redundancy increases the vehicle reliability.

Chapter 4 Vehicle Platoon Modeling

The vehicle platoon is actually a dynamic system composed of multiple vehicles which control the nodes through information exchange. A single vehicle uses only the information of the vehicles in the neighborhood to give control decisions, and finally achieves the coordination goal on the overall level of the vehicle platoon. Assume a vehicle platoon runs on a straight road. It has $n+1$ vehicles. The leader vehicle number is 0, and the following vehicle numbers are $1 \sim n$. The goal of the vehicle platoon is that the following vehicle speed is consistent with the leader vehicle speed, and the distance between adjacent vehicles remains at the desired following distance.

The vehicle platoon has the following characteristics:

(1) Each vehicle node is an agent in the system. The dynamic characteristics of the individual vehicles are relatively independent, but the communication structure allows multiple vehicles to be coupled into a vehicle platoon.

(2) Due to the limitation of sensors and communication range, the controllers of the vehicle platoon are distributed. Only the state information of the neighboring vehicles can be obtained, but the requirements of the system for the performance of the vehicle platoon are global.

In this chapter, the longitudinal dynamics of the vehicle nodes are modeled. The communication information flow is described as the graph structure by the graph theory. Then, the mathematical expression of the vehicle platoon geometry is given. Finally, the distributed controller is modeled.

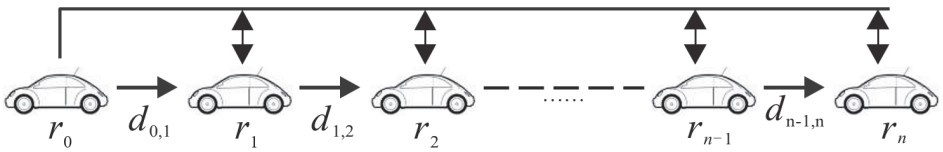


Figure 4.1 Vehicle platoon network topology.

4.1 Node dynamic

Node dynamics is used to describe the longitudinal dynamic of a single vehicle. There are strong nonlinear processes such as engine, transmission, windage resistance and braking system in the vehicle

node, so the longitudinal dynamics of the vehicle is strong nonlinear. Considering the complexity and accuracy of the modeling, the longitudinal dynamics of the vehicle is simplified, and the following models can be obtained.

The single integrator model uses the vehicle speed as the control input. The displacement is the only state in the vehicle dynamics. It greatly simplifies the theoretical analysis difficulty of the controller design.

Considering the vehicle as a mass point, the dual integrator model is used to describe the node dynamics. It ignores the time delay factor in the dynamic system [82][83]. The control variable is the acceleration of the vehicle.

$$\begin{cases} \dot{p}_i(t) = v_i(t) \\ \dot{v}_i(t) = u_i(t) \end{cases} \tag{4.1}$$

If the vehicle platoon is considered as a mass-spring-damping system [84][85], a linear second-order model description is obtained.

$$\begin{cases} \dot{p}_i(t) = v_i(t) \\ \dot{v}_i(t) + \eta v_i(t) = u_i(t) \end{cases} \tag{4.2}$$

Where η is the damping coefficient.

For theoretical analysis, a single integrator model is used to approximate the dynamics of a vehicle node [86][87]. The control variable is the vehicle speed.

$$\dot{p}_i(t) = u_i(t) \tag{4.3}$$

A state variable is further added to form a third-order state space model to approximate the power system dynamic in the vehicle node. In this model, the control input has been equivalently converted to the engine torque or braking torque.

A single-input single-output model is often used in the study of queue stability in the frequency domain.

4.2 Information Flow Topology

The information flow topology represents the transfer of information between the vehicles. It can

be intuitively abstracted into the graph structure. Then it can be described by the corresponding matrix. This technique is algebraic graph theory.

The graph theory plays an important role in describing the information communication among the vehicle platoon multi-agent systems. The vehicle platoon includes a leader and N followers. The communication among the N followers is specified by a directed graph $\mathcal{G}$. The graph consists of a node set and an edge set, which represent each agent and each directed information flow, respectively.

The adjacency matrix $\mathcal{A} = [a_{ij}] \in R^{N\times N}$ is defined as $a_{ij} = 1$ if there is a directed path from agent i to agent j; otherwise, $a_{ij} = 0$. a_{ij} indicates whether the vehicle i can obtain the status information of the vehicle j.

$\mathcal{N}_i$ denotes the neighbor set of the node i. It indicates a vehicle set which information can be detected by node i in the following vehicle. The degree matrix is defined as $\mathcal{D} = \mathrm{diag}[d_1, \cdots, d_N]$, where $d_i = \sum_{j\in\mathcal{N}_i} a_{ij}$. The Laplacian matrix is defined as $\mathcal{L} = \mathcal{D} - \mathcal{A}$.

In this paper, it is assumed that the graph contains a directed spanning tree, and the root agent can obtain the information of the leader. The adjacency matrix associated with the leader is defined by $\mathcal{A}_L = \mathrm{diag}[a_{1L}, \cdots, a_{NL}]$, where $a_{iL} = 1$ if and only if follower i can receive the leader's information; otherwise, $a_{iL} = 0$.

4.3 Vehicle Platoon Geometry Modeling

The goal of vehicle platoon control is that the following vehicle speed is consistent with the leader vehicle speed, and the distance between adjacent vehicles remains at the expected distance.

$$\begin{cases} \lim\limits_{t\to\infty} \left\| v_i(t) - v_0(t) \right\| = 0 \\ \lim\limits_{t\to\infty} \left\| p_{i-1}(t) - p_i(t) - d_{i-1,i} \right\| = 0 \end{cases} \tag{4.4}$$

Where $d_{i-1,i} > 0$ is the expected distance between node $i-1$ and node i. It determines the vehicle platoon geometry.

For a constant distance in the vehicle platoon, $d_{i-1,i}$ is a given constant which is greater than zero.

$$d_{i-1,i} = d_0 \tag{4.5}$$

The expected distance between two adjacent vehicles is independent of the vehicle speed, which can achieve higher traffic flow density.

For a constant time headway in the vehicle platoon, $d_{i-1,i}$ is a linear function of the vehicle speed.

$$d_{i-1,i} = t_h v_i + d_0 \tag{4.6}$$

Where t_h is the time headway. The expected distance between two adjacent vehicles is linear with the vehicle speed. It matches the characteristics of the driver, but limits the achievable traffic flow density.

For nonlinear distance in the vehicle platoon, $d_{i-1,i}$ is a nonlinear function of the vehicle speed.

$$d_{i-1,i} = d_0 + Tv_i + Gv_i^2 \tag{4.7}$$

Where T and G are the parameters to be optimized, which are determined by optimizing traffic flow density, ensuring queue stability and traffic flow stability. The expected distance between two adjacent vehicles is a non-linear function of the vehicle speed. It improves traffic flow stability and increases traffic flow density.

Finally, the local controller at node i uses its neighbor information to control the node to achieve the desired target of the vehicle platoon.

Chapter 5 Adaptive Consensus Control

In this chapter, a novel adaptive leader-following consensus algorithm for second order nonlinear systems subject to output constraints is proposed. The nonlinear dynamics of the followers is assumed to be unknown, which is identified online by FLSs. A barrier Lyapunov function is adopted to handle the violations of output constraints. An adaptive fuzzy backstepping design based on barrier Lyapunov function is performed to construct the virtual control law. To minimize the actual control law, an optimal virtual control signal is obtained by a neurodynamic optimization approach, which is utilized to design the actual control law. It is proved that the uniformly ultimately bounded control is achieved for leader-following consensus of output-constrained second-order nonlinear systems.

5.1 Problem Statement

The leader-following consensus control problem of output-constrained second-order nonlinear systems is considered. The systems consist of N followers guided by a leader. The dynamics of the i th follower is described by a second-order system with unmatched nonlinearities:

$$\begin{cases} \dot{p}_i = v_i + f_{i1}(p_i) \\ \dot{v}_i = u_i + f_{i2}(p_i, v_i) \\ y_i = p_i \end{cases} \tag{5.1}$$

where $p_i, v_i \in R$ are the position and velocity of the ith follower, respectively. $u_i, y_i \in R$ are the control input and the output. $f_{i1}(p_i)$ and $f_{i2}(p_i, v_i)$ are smooth functions denoting the unknown dynamics. Let $y_i = [y_1, \cdots, y_N]^T$ be the output vector of the followers, which is required to remain in a compact set; i.e., $\|y\| \le K_c$, where K_c is a positive constant.

The leader is given by the following dynamics:

$$\begin{cases} \dot{p}_l = v_l \\ \dot{v}_l = f_l(p_l, v_l, t) \\ y_l = p_l \end{cases} \tag{5.2}$$

where $p_l, v_l \in R$ are the state variables. $y_l \in R$ is the output. $f_l(p_l, v_l, t)$ is a piecewise continuous in

time t . The leader information cannot be available to all followers.

Assumption 5.1. For any $K_c > 0$, there exist positive constants Y_{b1}, Y_{b2}, Y_b satisfying $\max\left[Y_{b1}, Y_{b2}\right] \le Y_b < K_c$, such that the leader's output satisfies $-Y_{b1} \le \overline{y}_l < Y_{b2}$, where $\overline{y}_l = [y_l, \cdots, y_l]^T \in R^N$.

Graph theory plays an important role in describing the information communication among multi-agent systems. The communication among the N followers is specified by a directed graph $\mathcal{G}$. The graph consists of a node set and an edge set, which represent each agent and each directed information flow, respectively. The adjacency matrix $\mathcal{A} = [a_{ij}] \in R^{N\times N}$ is defined as $a_{ij} = 1$ if there is a directed path from agent i to agent j; otherwise, $a_{ij} = 0$. $\mathcal{N}_i$ denotes the neighbor set of the node i. The degree matrix is defined as $\mathcal{D} = \text{diag}[d_1, \cdots, d_N]$, where $d_i = \sum_{j\in\mathcal{N}_i} a_{ij}$. The Laplacian matrix is defined as $\mathcal{L} = \mathcal{D} - \mathcal{A}$. In this paper, it is assumed that the graph contains a directed spanning tree, and the root agent can obtain the information of the leader. The adjacency matrix associatd with the leader is defined by $\mathcal{A}_L = \text{diag}[a_{1L}, \cdots, a_{NL}]$, where $a_{iL} = 1$ if and only if follower i can receive the leader's information; otherwise, $a_{iL} = 0$.

The control objective of this brief is to construct an adaptive consensus control law u_i, such that all followers can track that of the leader without violating the output constraints.

5.2 Consensus Control Design

The following lemmas are introduced, which are used in the consensus control design.

Lemma 5.1 [88]: For any positive constant λ, the following inequality holds for all z in the interval $|z| < \lambda$:

$$\log\frac{\lambda^2}{\lambda^2 - z^2} < \frac{z^2}{\lambda^2 - z^2}, \tag{5.3}$$

where $\log(*)$ denotes the natural logarithm of $*$.

Proof: Let $F(z) = \log\frac{\lambda^2}{\lambda^2 - z^2} - \frac{z^2}{\lambda^2 - z^2}$, for $|z| < \lambda$. Then, the derivative of $F(z)$ along with the variable z yields

$$\dot{F}(z) = -\frac{2z^3}{(\lambda^2 - z^2)^2}. \tag{5.4}$$

Note that $\dot{F}(z) < 0$, if $0 < z < \lambda$, and $\dot{F}(z) > 0$, if $-\lambda < z < 0$. Thus, in the interval $0 < |z| < \lambda$, $F(z)$ gets the maximum value as $z \to 0$, which is given as:

$$\lim_{z\to 0} F(z) = \lim_{z\to 0}[\log\frac{\lambda^2}{\lambda^2 - z^2} - \frac{z^2}{\lambda^2 - z^2}] = 0. \tag{5.5}$$

Therefore, it is obtained that $F(z) < 0$; i.e., $\log\frac{\lambda^2}{\lambda^2 - z^2} < \frac{z^2}{\lambda^2 - z^2}$.

The proof is completed.

Lemma 5.2 [89]: Let $F(X)$ be a continuous function defined on a compact set Ω. Then, for a given desired level of accuracy $\varepsilon > 0$, there exists an FLS $\theta^T\phi(X)$ such that

$$\sup_{X\in\Omega} |F(X) - \theta^T\phi(X)| \leq \varepsilon, \tag{5.6}$$

where $\theta \in R^m$ is an ideal constant weight vector, and m is the number of the fuzzy rules. $\phi(x) \in R^m$ is a fuzzy basis function vector.

Let θ^* be the optimal parameter vector defined as

$$\theta^* = \underset{\theta\in U}{\operatorname{argmin}}\{\sup_{X\in\Omega} |F(X) - \theta^T\phi(X)|\}. \tag{5.7}$$

Then

$$F(X) = \theta^{*T}\phi(X) + \varepsilon, \tag{5.8}$$

where ε is the minimum fuzzy approximation error.

The design procedure contains two steps, and each step is based on an error surface. The detailed adaptive consensus control design is presented as follows.

Step 1: The synchronization error is defined as

$$e_{i1} = \sum_{j\in\mathcal{N}_i} a_{ij}\,(p_i - p_j) + a_{iL}(p_i - p_l). \tag{5.9}$$

From (5.1), we get

$$\dot{e}_{i1} = (d_i + a_{iL})(v_i + f_{ij1}(p_i)) - \sum_{j\in\mathcal{N}_i} a_{ij}\,v_j - a_{iL}v_l, \tag{5.10}$$

where $f_{ij1}(p_i, p_j) = f_{i1}(p_i) - \frac{1}{d_i + a_{iL}}\sum_{j\in\mathcal{N}_i} a_{ij}\,f_{j1}(p_j)$.

By Lemma 5.2, the nonlinear function can be approximated by an FLS as

$$f_{ij1}(p_i, p_j) = \theta_{i1}^{*T}\phi_{i1}(p_i, p_j) + \varepsilon_{i1}, \tag{5.11}$$

where θ_{i1}^* is the optimal fuzzy parameter vector, $\phi_{i1}(p_i, p_j)$ is the fuzzy basis function vector, and ε_{i1} is

the minimum fuzzy approximate error.

To stabilize the error surface, the virtual control v_i^c is designed as

$$\begin{aligned} v_i^c = \; & -k_1 e_{i1} - \theta_{i1}^T \phi_{i1}(p_i, p_j) - \frac{e_{i1}}{\lambda^2 - e_{i1}^2} + \sum_{j \in \mathcal{N}_i} a_{ij} v_j - \\ & \hat{\varepsilon}_{i1m} \tanh\left(\frac{e_{i1}}{(\lambda^2 - e_{i1}^2)\epsilon_1}\right), \end{aligned} \tag{5.12}$$

where $k_1, \epsilon_1 > 0$ are design parameters. θ_{i1} is the estimate of θ_{i1}^*. $\hat{\varepsilon}_{i1m}$ is the estimate of ε_{i1m}, where $|\varepsilon_{i1} - \frac{a_{iL}}{d_i + a_{iL}} v_l| \leq \varepsilon_{i1m}$.

In the designed virtual control law, the first term $k_1 e_{i1}$ is a linear feedback control utilized to drive the output of the ith follower to consensus. The second term $\theta_{i1}^T \phi_{i1}(p_i, p_j)$ is an FLS, which is used to compensate the unknown dynamics of the ith follower and its neighbors. The term $\frac{e_{i1}}{\lambda^2 - e_{i1}^2}$ is to eliminate the effect of output constraints, which is derived from a barrier Lyapunov function defined later. The term $\hat{\varepsilon}_{i1m} \tanh\left(\frac{e_{i1}}{(\lambda^2 - e_{i1}^2)\epsilon_1}\right)$ is a robust signal, which is used to counteract the approximation error ε_{i1} and the leader's velocity obtained by the root agent $\frac{a_{iL}}{d_i + a_{iL}} v_l$.

A Lyapunov function is defined as:

$$V_{i1} = \frac{1}{2(d_i + a_{iL})} \log \frac{\lambda^2}{\lambda^2 - e_{i1}^2} + \frac{1}{2\gamma_1} \tilde{\theta}_{i1}^T \tilde{\theta}_{i1} + \frac{1}{2\tau_1} \tilde{\varepsilon}_{i1m}^2, \tag{5.13}$$

where $\gamma_1, \tau_1 > 0$ are design parameters. $\tilde{\theta}_{i1} = \theta_{i1}^* - \theta_{i1}$. $\tilde{\varepsilon}_{i1m} = \varepsilon_{i1m} - \hat{\varepsilon}_{i1m}$. Note that the barrier Lyapunov candidate $\log \frac{\lambda^2}{\lambda^2 - e_{i1}^2}$ grows to infinity at $|e_{i1}| = \lambda$. Therefore, it is positive definite and continuously differentiable in the set $|e_{i1}| < \lambda$; i.e., it is a valid Lyapunov candidate in the set $|e_{i1}| < \lambda$. By ensuring the boundedness of the barrier Lyapunov candidate in the closed-loop multi-agent systems, we can guarantee that the output constraints are not transgressed.

The time derivative of V_{i1} along with (5.10)-(5.12) yields

$$\begin{aligned} \dot{V}_{i1} = \; & \frac{e_{i1}}{\lambda^2 - e_{i1}^2} \left(v_i + \theta_{i1}^{*T} \phi_{i1}(p_i, p_j) + \varepsilon_{i1} - \frac{a_{iL}}{d_i + a_{iL}} v_l\right) + \frac{1}{\gamma_1} \tilde{\theta}_{i1}^T \dot{\tilde{\theta}}_{i1} + \frac{1}{\tau_1} \tilde{\varepsilon}_{1m} \dot{\tilde{\varepsilon}}_{1m}, \\ = \; & \frac{e_{i1}}{\lambda^2 - e_{i1}^2} \left[e_{i2} + \tilde{v}_i - k_1 e_{i1} + \tilde{\theta}_{i1}^T \phi_{i1}(p_i, p_j) - \frac{e_{i1}}{\lambda^2 - e_{i1}^2} - \right. \\ & \left. \hat{\varepsilon}_{i1m} \tanh\left(\frac{e_{i1}}{(\lambda^2 - e_{i1}^2)\epsilon_1}\right) + \varepsilon_{i1} - \frac{a_{iL}}{d_i + a_{iL}} v_l\right] + \\ & \frac{1}{\gamma_1} \tilde{\theta}_{i1}^T \dot{\tilde{\theta}}_{i1} + \frac{1}{\tau_1} \tilde{\varepsilon}_{i1m} \dot{\tilde{\varepsilon}}_{i1m}, \end{aligned} \tag{5.14}$$

where $e_{i2} = v_i - v_i^o$. $\tilde{v}_i = v_i^o - v_i^c$. v^o is a signal to be optimized.

Using the inequality $|g| - g \tanh(g/h) \leq 0.2785h \; (\forall h > 0)$, the inequality (5.14) can be further

expressed as

$$\begin{aligned}\dot{V}_{i1} \leq\ & \frac{e_{i1}}{\lambda^2-e_{i1}^2}[e_{i2}+\tilde{v}_i-k_1e_{i1}-\frac{e_{i1}}{\lambda^2-e_{i1}^2}+\tilde{\theta}_{i1}^T\phi_{i1}(p_i,p_j)+\\ & \tilde{\varepsilon}_{i1m}\tanh(\frac{e_{i1}}{(\lambda^2-e_{i1}^2)\epsilon_1})]+0.2785\epsilon_1\varepsilon_{i1m}-\\ & \frac{1}{\gamma_1}\tilde{\theta}_{i1}^T\dot{\theta}_{i1}-\frac{1}{\tau_1}\tilde{\varepsilon}_{i1m}\dot{\hat{\varepsilon}}_{i1m}.\end{aligned} \tag{5.15}$$

Using the Young's inequality, we have

$$\begin{cases}\frac{e_{i1}}{\lambda^2-e_{i1}^2}e_{i2}\leq\frac{e_{i1}^2}{2(\lambda^2-e_{i1}^2)^2}+\frac{1}{2}e_{i2}^2,\\ \frac{e_{i1}}{\lambda^2-e_{i1}^2}\tilde{v}_i\leq\frac{e_{i1}^2}{2(\lambda^2-e_{i1}^2)^2}+\frac{1}{2}\tilde{v}_i^2.\end{cases} \tag{5.16}$$

Then

$$\begin{aligned}\dot{V}_{i1} \leq\ & \frac{e_{i1}}{\lambda^2-e_{i1}^2}[-k_1e_{i1}+\tilde{\theta}_{i1}^T\phi_{i1}(p_i,p_j)+\tilde{\varepsilon}_{i1m}\tanh(\frac{e_{i1}}{(\lambda^2-e_{i1}^2)\epsilon_1})]+\\ & \frac{1}{2}e_{i2}^2+\frac{1}{2}\tilde{v}_i^2+0.2785\epsilon_1\varepsilon_{i1m}-\frac{1}{\gamma_1}\tilde{\theta}_{i1}^T\dot{\theta}_{i1}-\frac{1}{\tau_1}\tilde{\varepsilon}_{i1m}\dot{\hat{\varepsilon}}_{i1m}.\end{aligned} \tag{5.17}$$

The adaptive laws for θ_{i1} and $\hat{\varepsilon}_{i1m}$ are designed as

$$\begin{cases}\dot{\theta}_{i1}=&\gamma_1\text{Proj}(\theta_{i1},\phi_{i1}(p_i,p_j)\frac{e_{i1}}{\lambda^2-e_{i1}^2}),\\ \dot{\hat{\varepsilon}}_{i1m}=&\tau_1\text{Proj}[\hat{\varepsilon}_{i1m},\frac{e_{i1}}{\lambda^2-e_{i1}^2}\tanh(\frac{e_{i1}}{(\lambda^2-e_{i1}^2)\epsilon_1})],\end{cases} \tag{5.18}$$

where $\text{Proj}(*)$ is the projection operator.

Substituting (5.18) into (5.17), we have

$$\begin{aligned}\dot{V}_{i1} \leq\ & -k_1\frac{e_{i1}^2}{\lambda^2-e_{i1}^2}+\frac{1}{2}e_{i2}^2+\frac{1}{2}\tilde{v}_i^2-\sigma_1\tilde{\theta}_{i1}^T\tilde{\theta}_{i1}-\\ & \varsigma_1\tilde{\varepsilon}_{i1m}^2+0.2785\epsilon_1\varepsilon_{i1m}+\sigma_1\theta_{1m}^2+\varsigma_1b_{1m}^2,\end{aligned} \tag{5.19}$$

where $\sigma_1,\varsigma_1>0$ are design parameters. $||\tilde{\theta}_{i1}||\leq\theta_{1m},|\tilde{\varepsilon}_{i1m}|\leq b_{1m}$.

A command governor is introduced to generate an optimal virtual control signal v_i^o, which is able to balance the virtual control law v_i^c and the actual velocity signal v_i, such that control input can be reduced. Define a quadratic cost function as follows, which can generate the optimal virtual control signal v_i^o

$$\begin{aligned}\min\quad & q_{i1}(v_i^o-v_i^c)^2+\quad q_{i2}(v_i^o-v_i)^2\\ \text{s.t.}\quad & \underline{v}_i^*\leq v_i^o\quad\leq\overline{v}_i^*,\end{aligned} \tag{5.20}$$

where $\underline{v}_i^*\geq\underline{v}_i$ and $\overline{v}_i^*\leq\overline{v}_i$ are the allowed minimal and maximal velocity signals with $\underline{v}_i\leq|v_i|\leq\overline{v}_i$.

The objective function is to minimize the two errors including $(v_i^o-v_i^c)$ and $(v_i^o-v_i)$. Therefore, the command governor is able to prescribe a mediate signal between v_i^c and v_i.

The optimization problem (5.20) can be reexpressed as

$$\begin{aligned} \min \quad & q_i(v_i^o)^2/2 + c_i v_i^o \\ \text{s.t.} \quad & \underline{v}_i^* \le v_i^o \le \overline{v}_i^*, \end{aligned} \tag{5.21}$$

where $q_i = 2(q_{i1} + q_{i2})$. $c_i = -2q_{i1}v_i^c - 2q_{i2}v_i$.

A recurrent neural network is utilized to solve the above optimization problem:

$$\iota \dot{v}_i^o = \Upsilon(v_i^o - (q_i v_i^o + c_i)) - v_i^o, \tag{5.22}$$

where ι is a time constant. $\Upsilon(\vartheta_i)$ is a piecewise continuous activation function defined as

$$\Upsilon(\vartheta_i) = \begin{cases} \underline{v}_i^*, & \vartheta_i < \underline{v}_i^*, \\ \vartheta_i, & \underline{v}_i^* < \vartheta_i < \overline{v}_i^*, \\ \overline{v}_i^*, & \vartheta_i > \overline{v}_i^* \end{cases} \tag{5.23}$$

In [90], it has been proven that the recurrent neural network exponentially converges to the optimal solution of (5.21). Since the virtual control law v_i^c is bounded, $\tilde{v}_i$ is bounded by $\tilde{v}_m$ with $\tilde{v}_m > 0$.

Step 2: The error surface is chosen as $e_{i2} = v_i - v_i^o$. From (5.1), the dynamics of e_{i2} is given as

$$\dot{e}_{i2} = \quad u_i + f_{i2}(p_i, v_i) - \dot{v}_i^o. \tag{5.24}$$

An FLS is employed to approximate the nonlinear function $f_{i2}(p_i, v_i)$ as

$$f_{i2}(p_i, v_i) = \theta_{i2}^{*T}\phi_{i2}(p_i, v_i) + \varepsilon_{i2}, \tag{5.25}$$

where θ_{i2}^* is the optimal parameter vector, $\phi_{i2}(p_i, v_i)$ is the basis function vector, and ε_{i2} is the minimum approximate error bounded by ε_{2m}.

The actual control law u_i is designed as

$$u_i = -k_2 e_{i2} - \theta_{i2}^T\phi_{i2}(p_i, v_i) - \frac{1}{2}e_{i2} - \hat{\varepsilon}_{i2m}\tanh(\frac{e_{i2}}{\epsilon_2}), \tag{5.26}$$

where $k_2, \epsilon_2 > 0$ are design parameters. θ_{i2} is the estimate of θ_{i2}^*. $\hat{\varepsilon}_{2m}$ is the estimate of ε_{2m}.

A Lyapunov function is defined as:

$$V_{i2} = \frac{1}{2}e_{i2}^2 + \frac{1}{2\gamma_2}\tilde{\theta}_{i2}^T\tilde{\theta}_{i2} + \frac{1}{2\tau_2}\tilde{\varepsilon}_{i2m}^2, \tag{5.27}$$

where $\gamma_2, \tau_2 > 0$ are design parameters. $\tilde{\theta}_{i2} = \theta_{i2}^* - \theta_{i2}$. $\tilde{\varepsilon}_{i2m} = \varepsilon_{i2m} - \hat{\varepsilon}_{i2m}$.

The time derivative of (5.27) along with (5.24) and (5.26) is given by

$$\begin{aligned}\dot{V}_{i2} = \;& e_{i2}[-k_2 e_{i2} + \tilde{\theta}_{i2}^T \phi_{i2}(p_i, v_i) - \frac{1}{2} e_{i2} - \hat{\varepsilon}_{i2m} \tanh(\frac{e_{i2}}{\epsilon_2}) + \varepsilon_{i2}] - \\ & \frac{1}{\gamma_2} \tilde{\theta}_{i2}^T \dot{\theta}_{i2} - \frac{1}{\tau_2} \tilde{\varepsilon}_{i2m} \dot{\hat{\varepsilon}}_{i2m} \\ \leq \;& e_{i2}[-k_2 e_{i2} + \tilde{\theta}_{i2}^T \phi_{i2}(p_i, v_i) - \frac{1}{2} e_{i2} + \tilde{\varepsilon}_{i2m} \tanh(\frac{e_{i2}}{\epsilon_2})] + \\ & 0.2785 \varepsilon_{i2m} \epsilon_2 - \frac{1}{\gamma_2} \tilde{\theta}_{i2}^T \dot{\theta}_{i2} - \frac{1}{\tau_2} \tilde{\varepsilon}_{i2m} \dot{\hat{\varepsilon}}_{i2m}.\end{aligned} \tag{5.28}$$

The adaptive laws for θ_{i2} and $\hat{\varepsilon}_{2m}$ are designed as

$$\begin{cases} \dot{\theta}_{i2} = & \gamma_2 \text{Proj}(\theta_{i2}, \phi_{i2}(p_i, v_i) e_{i2}), \\ \dot{\hat{\varepsilon}}_{i2m} = & \tau_2 \text{Proj}[\hat{\varepsilon}_{i2m}, e_{i2} \tanh(\frac{e_{i2}}{\epsilon_2})]. \end{cases} \tag{5.29}$$

Then

$$\begin{aligned}\dot{V}_{i2} \leq \;& -k_2 e_{i2}^2 - \frac{1}{2} e_{i2}^2 - \sigma_2 \tilde{\theta}_{i2}^T \tilde{\theta}_{i2} - \varsigma_2 \tilde{\varepsilon}_{i2m}^2 + \\ & 0.2785 \varepsilon_{i2m} \epsilon_2 + \sigma_2 \theta_{2m}^2 + \varsigma_2 b_{2m}^2,\end{aligned} \tag{5.30}$$

where $\sigma_2, \varsigma_2 > 0$ are design parameters. $||\tilde{\theta}_{i2}|| \leq \theta_{2m}, |\tilde{\varepsilon}_{i2m}| \leq b_{2m}$.

Remark 5.1. Optimization-based command governors are widely suggested for enforcing constraint satisfaction [91]. The command governor in (5.20) plays the role of a pre-filter, which generates an optimal signal v_i^o by taking the velocity constraint into account. The first term is to minimize the error between the optimal signal v_i^o and the virtual control law v_i^c. The second term is to minimize the error between the optimal signal v_i^o and the actual velocity signal v_i. It means that the command governor is to prescribe a mediate signal between v_i and v_i^c. Besides, since a direct and aggressive tracking of the virtual control law is avoided in Step 2, it is able to minimize the control effort during transient phase.

5.3 Stability Analysis

The consensus control performance of second-order nonlinear systems consisting of (5.1) and (5.2) is summarized by the following theorem.

Theorem 5.1: Consider the nonlinear multi-agent systems consisting of N followers modeled by (5.1) guided by the leader (5.2), the virtual control law (5.12), the optimal virtual control law (5.22), the actual control law (5.26), and the adaptive laws (5.18), (5.29). Then, the proposed adaptive fuzzy consensus control scheme guarantees the following terms:

(1) Uniformly ultimately bounded control is achieved.

(2) All followers are able to synchronize to the leader with a bounded tracking error.

(3) The outputs of all followers remain in a compact set, and the output constraints are never violated.

Proof: The Lyapunov function is constructed as

$$V = \sum_{i=1}^{N}(V_{i1} + V_{i2}). \tag{5.31}$$

The time derivative of V along (5.19) and (5.30) yields

$$\begin{aligned}\dot{V} \leq\ & \sum_{i=1}^{N}(-k_1\frac{e_{i1}^2}{\lambda^2-e_{i1}^2} - k_2e_{i2}^2 - \sigma_1\tilde{\theta}_{i1}^T\tilde{\theta}_{i1} - \sigma_2\tilde{\theta}_{i2}^T\tilde{\theta}_{i2} - \varsigma_1\tilde{\varepsilon}_{i1m}^2 - \varsigma_2\tilde{\varepsilon}_{i2m}^2) + \\ & \sum_{i=1}^{N}(0.2785\varepsilon_{i1m}\epsilon_1 + 0.2785\varepsilon_{i2m}\epsilon_2) + N\sigma_1\theta_{1m}^2 + N\sigma_2\theta_{2m}^2 + \\ & N\varsigma_1b_{1m}^2 + N\varsigma_2b_{2m}^2 + \frac{N}{2}\tilde{v}_m^2.\end{aligned} \tag{5.32}$$

By Lemma 5.1, it is obtained that

$$\dot{V} \leq\ -\alpha V + \beta, \tag{5.33}$$

where

$$\begin{cases}\alpha = & \min[2k_1(d_i + a_{iL}),2k_2, 2\gamma_1\sigma_1, 2\gamma_2\sigma_2, 2\varsigma_1\iota_1, 2\varsigma_2\iota_2], \\ \beta = & \sum_{i=1}^{N}(0.2785\varepsilon_{i1m}\epsilon_1 + 0.2785\varepsilon_{i2m}\epsilon_2) + N\sigma_1\theta_{1m}^2 + \\ & N\sigma_2\theta_{2m}^2 + N\varsigma_1b_{1m}^2 + N\varsigma_2b_{2m}^2 + N\tilde{v}_m^2/2.\end{cases} \tag{5.34}$$

Note that $\dot{V} < 0$, if $\alpha V > \beta$. Then, by (5.31), we have

$$\begin{cases}|e_{i1}| \leq & \lambda\sqrt{1 - e^{-2\beta(d_i+a_{iL})/\alpha}}, \\ |e_{i2}| \leq & \sqrt{\frac{2\beta}{\alpha}}, \\ ||\tilde{\theta}_{i1}|| \leq & \sqrt{\frac{2\beta\gamma_1}{\alpha}}, \\ ||\tilde{\theta}_{i1}|| \leq & \sqrt{\frac{2\beta\gamma_2}{\alpha}}, \\ |\tilde{\varepsilon}_{1m}| \leq & \sqrt{\frac{2\beta\tau_1}{\alpha}}, \\ |\tilde{\varepsilon}_{2m}| \leq & \sqrt{\frac{2\beta\tau_2}{\alpha}}.\end{cases} \tag{5.35}$$

Thus, all error signals in the closed-loop systems are bounded. Form (5.9), we have

$$e_1 = (\mathcal{L} + \mathcal{A}_L)(y - \overline{y}_l), \tag{5.36}$$

where $e_1 = [e_{11}, \cdots, e_{N1}]^T$. Then, the tracking error is bounded by

$$||y - \overline{y}_l|| \leq \frac{||e_1||}{\sigma(\mathcal{L}+\mathcal{A}_L)}, \tag{5.37}$$

where $\sigma(*)$ denotes the minimum singular value of the matrix $*$. Furthermore, we can get that

$$||y|| \leq \frac{||e_1||}{\sigma(\mathcal{L}+\mathcal{A}_L)} + ||\bar{y}_l|| \leq \frac{\lambda\sqrt{N}}{2\sigma(\mathcal{L}+\mathcal{A}_L)} + Y_b \triangleq K_c. \tag{5.38}$$

From the above analysis, we can conclude that the output signals are remain in a compact set, whose bound is associated with the design parameter λ and the communication graph $\mathcal{G}$. Therefore, the proposed adaptive fuzzy consensus control method can achieve uniformly ultimately bounded control, the tracking error converges to a bounded compact set, and the output constraints are not violated.

Remark 5.2. As for a barrier Lyapunov function, the synchronization error e_{i1} satisfies $|e_{i1}(t)| < \lambda, \forall t > 0$ under the initial condition that $|e_{i1}(0)| < \lambda$. As for a quadratic Lyapunov function, consensus control can be realized without violating output constraints if the initial condition satisfies $||e_1(0)|| < \sqrt{N\lambda^2 - \Pi}$, where $\Pi = \sum_{i=1}^{N}((||\theta_{i1}|| + \theta_{1m})^2/\gamma_1 + (||\theta_{i2}|| + \theta_{2m})^2/\gamma_2 + (\varepsilon_{i1m} + \hat{\varepsilon}_{i1m})^2/\tau_1 + (\varepsilon_{i2m} + \hat{\varepsilon}_{i2m})^2/\tau_2)$. Note that an additional condition $N\lambda^2 > \Pi$ needs to be satisfied. Therefore, compared with a quadratic Lyapunov function, the initial conditions arising from the use of a barrier Lyapunov function are less restrictive.

5.4 Simulation Results

A vehicle platoon with one leader is considered as a second-order model. A simulation example will be given to verify the obtained theoretical results. It is the following follower dynamics [92].

$$\begin{cases} \dot{p}_i = & v_i, \\ \dot{v}_i = & u_i + f_{i2}(v_i), \\ y_i = & p_i, \end{cases} \tag{5.39}$$

where $f_{i2}(v_i) = -k/m\ v_i\ (i = 1, \cdots, 4)$. m is the mass. k is the coefficient parameter. The model parameters are chosen as $k = 0.1, m = 1$.

The leader is modeled by

$$\begin{cases} \dot{p}_l = & v_l, \\ \dot{v}_l = & -0.1v_l, \\ y_l = & p_l. \end{cases} \tag{5.40}$$

Fuzzy membership functions are given by

$$
\begin{aligned}
\mu_1(p_i) &= \frac{1}{1+\exp[-4(p_i/2-\pi/2)]},\\
\mu_2(p_i) &= \exp(-p_i^2),\\
\mu_3(p_i) &= \frac{1}{1+\exp[-4(p_i/2+\pi/2)]},\\
\mu_1(v_i) &= \frac{1}{1+\exp[-4(v_i/2-\pi/2)]},\\
\mu_2(v_i) &= \exp(-v_i^2),\\
\mu_3(v_i) &= \frac{1}{1+\exp[-4(v_i/2+\pi/2)]}.
\end{aligned}
$$

The communication graph is shown in Figure 5.1, where the leader is denoted by node 0, and the followers are denoted by nodes 1-4 with node 2 as the root node. From Figure 5.1, we can get the adjacency matrix $\mathcal{A}=\begin{bmatrix}0&1&0&0\\0&0&0&0\\0&1&0&0\\0&0&1&0\end{bmatrix}$ and the leader adjacency matrix $\mathcal{A}_L=\text{diag}[0,1,0,0]$.

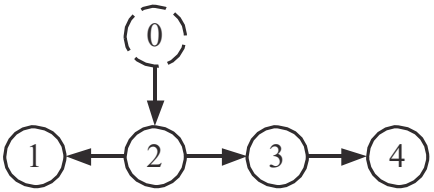


Figure 5.1 Communication graph.

In simulation, the initial value of the states is set as $p_1(0)=0.25, p_2(0)=0.35, p_3(0)=0.2, p_4(0)=0.1, v_1(0)=0, v_2(0)=0.01, v_3(0)=0.02, v_4(0)=0.03, p_l(0)=0.5, v_l(0)=0.05$. The fuzzy parameter vector is initialized to zero. The design parameters of the proposed consensus control algorithm are set as $k_1=0.5, k_2=5, \lambda=0.5, \gamma_2=300, \tau_1=\tau_2=10, \epsilon_1=\epsilon_2=1$. $q_{i1}=q_{i2}=1, \iota=0.08, \overline{v}_i^*=0.7, \underline{v}_i^*=-0.7$.

Simulation results of the proposed adaptive fuzzy consensus control method via neurodynamic optimization (Scheme I) are shown in Figures 5.2-5.4. Figures 5.2-5.4 show the simulation results using Scheme I. The consensus control performance is presented in Figure 5.2. We can see that the positions of all followers can synchronize to that of the leader. The synchronization errors are shown in Figure 5.3, and they are bounded by 0.5. The control inputs are shown in Figure 5.4.

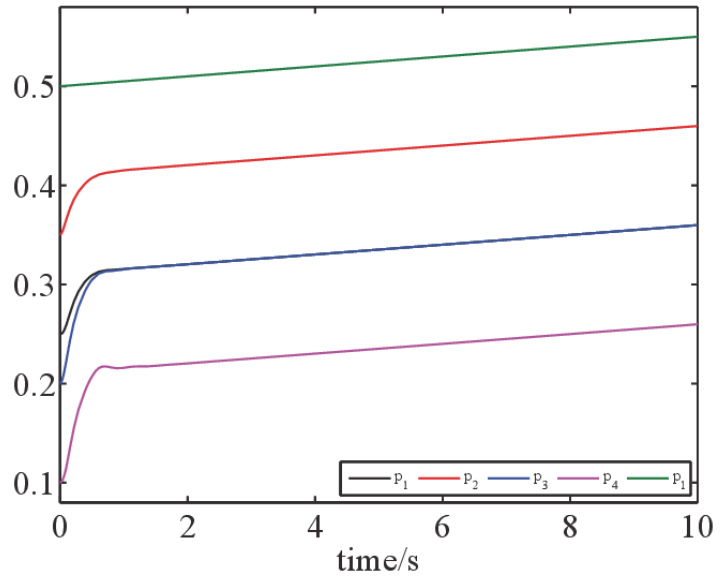


Figure 5.2 Consensus control performance using Scheme I.

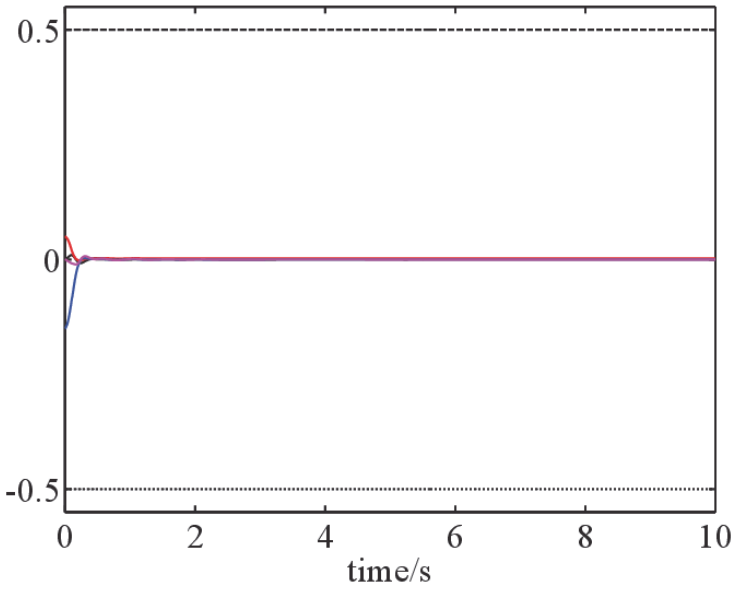


Figure 5.3 The trajectories of synchronization errors using Scheme I.

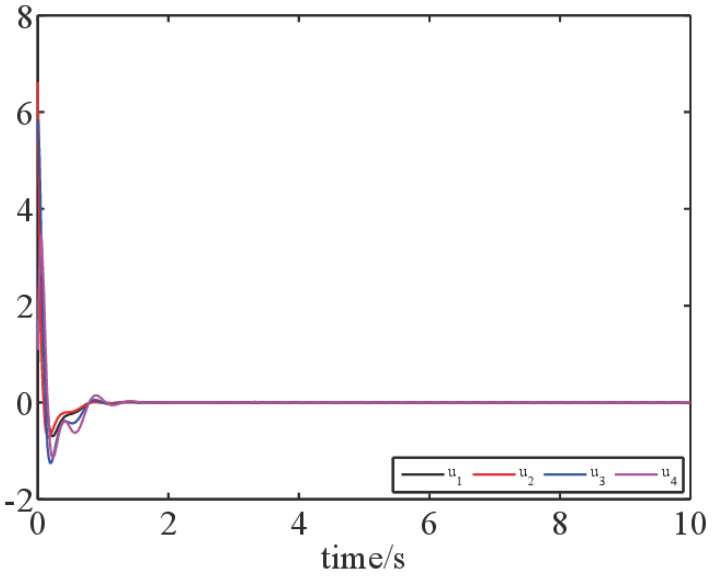


Figure 5.4 The trajectories of control inputs using Scheme I.

From the above simulation, we can conclude that the proposed adaptive fuzzy consensus control approach can realize the consensus control of output-constrained second-order nonlinear systems with unknown dynamics.

Chapter 6 Adaptive Containment Control

In this chapter, the containment control problem for multi-agent systems with more general nonlinear dynamics is studied. The nonlinear dynamics of each follower can be totally unknown. Using FLSs to identify the unknown nonlinear dynamics, distributed state feedback and output feedback containment control schemes are proposed to drive the states of all followers into the convex hull spanned by the leaders. It is proved that the containment control errors converge to a residual set.

Notations. Throughout this paper, R^+ is a set of positive real numbers. $R^{n\times m}$ is a set of $n \times m$ real matrices. I_N is an identity matrix with the dimension of N. $\|\cdot\|$ is the Euclidean norm of a vector. $\|\cdot\|_F$ is the Frobenius norm of a matrix. $tr(\cdot)$ is the trace of a matrix. $\overline{\sigma}(\cdot)$ and $\underline{\sigma}(\cdot)$ are the maximum and minimum singular values of a matrix, respectively. $\mathrm{diag}(\lambda_i)$ is a diagonal matrix with λ_i being the ith diagonal element. $\otimes$ is the Kronecker product.

6.1 Problem Statement

Consider a class of nonlinear multi-agent systems consisting of N followers and M leaders. The dynamics of follower i are described by

$$\begin{aligned} \dot{x}_i &= Ax_i + B[u_i + f_i(x_i)], \quad i = 1, \cdots, N, \\ y_i &= Cx_i, \end{aligned} \tag{6.1}$$

where $x_i = [x_{i1}, \cdots, x_{in}]^T \in R^n$, $u_i = [u_{i1}, \cdots, u_{im}]^T \in R^m$, $y_i \in R^p$ are the state vectors, inputs and outputs of the systems. $f_i(x_i) = [f_{i1}(x_i), \cdots, f_{im}(x_i)]^T \in R^m$ are unknown nonlinear functions. $A \in R^{n\times n}$, $B \in R^{n\times m}$, $C \in R^{p\times n}$ are known matrices.

The dynamics of leader k are given by

$$\dot{x}_k = Ax_k + Br_k(t), \quad k = N+1, \cdots, N+M, \tag{6.2}$$

where $x_k = [x_{k1}, \cdots, x_{kn}]^T \in R^n$ are the state vectors. $r_k(t) \in R^m$ are unknown bounded inputs.

The information flow among the agents can be described by a directed graph $G = (\nu, \varepsilon, \Lambda)$ consists of a vertex set $\nu = \{n_1, \cdots, n_{N+M}\}$, an edge set $\varepsilon = \{(n_i, n_j) \in \nu \times \nu\}$, and an adjacency matrix $\Lambda = [a_{ij}] \in R^{(N+M)\times(N+M)}$. n_i represents agent node i. $(n_i, n_j) \in \varepsilon$ means that there is a directed

information flow from agent i to agent j. The neighbor set of node i is denoted by $N_i = \{j|(n_i, n_j) \in \varepsilon\}$. Each element a_{ij} of Λ is defined as $a_{ij} > 0$, if $(n_i, n_j) \in \varepsilon$ and $a_{ij} = 0$, if $(n_i, n_j) \notin \varepsilon$. Throughout this paper, it is assumed that $a_{ii} = 0$. If $a_{ij} = a_{ji}, \forall i, j$, the graph G is undirected; otherwise the graph G is directed. A directed graph has a spanning tree if there is a root node, such that there is a directed path from the root node to every other node in the graph. The Laplacian matrix $L = [L_{ij}] \in R^{(N+M)\times(N+M)}$ is defined as

$$L_{ij} = \begin{cases} -a_{ij}, & i \neq j \\ \sum_{j \in N_i} a_{ij}, & i = j \end{cases}.$$

Then, the Laplacian matrix $L = D - \Lambda$, where $D = \mathrm{diag}(d_i)$ is the degree matrix with $d_i = \sum_{j=1}^{N} a_{ij}$ $(i = 1, \cdots, N)$.

An agent is called a follower if the agent has at least one neighbor. An agent is called a leader if the agent has no neighbor. Without loss of generality, we assume that the agents indexed by $1, \cdots, N$ are followers, whereas the agents indexed by $N + 1, \cdots, N + M$ are leaders. Then, the Laplacian matrix L can be partitioned as

$$L = \begin{bmatrix} L_1 & L_2 \\ 0_{M\times N} & 0_{M\times M} \end{bmatrix},$$

where $L_1 \in R^{N\times N}$ and $L_2 \in R^{N\times M}$.

Assumption 6.1. For each follower, there exists at least one leader that has a directed path to that follower.

Lemma 6.1 [24]. Under Assumption 1, all the eigenvalues of L_1 have positive real parts, each entry of $-L_1^{-1}L_2$ is nonnegative, and each row of $-L_1^{-1}L_2$ has a sum equal to 1.

Definition 6.1. The set $\Xi \subseteq R^n$ is said to be convex if for any $x_1, x_2 \in \Xi$ and any $\alpha \in [0,1]$, the point $\alpha x_1 + (1 - \alpha)x_2$ is in Ξ. The convex hull $\mathrm{Co}(X)$ for a set of points $X = \{x_1, \cdots, x_n\}$ is the minimal convex set containing all points in X and is defined as $\mathrm{Co}(X) = \{\sum_{i=1}^{n} \alpha_i x_i | x_i \in X, \alpha_i > 0, \sum_{i=1}^{n} \alpha_i = 1\}$.

The *control objective* is to design containment controllers u_i, such that the states of all followers converge to the convex hull formed by the leaders $r_l(t)$, i.e., $\inf_{h(t)\in R(t)} \|x_i - h(t)\| < \epsilon, \forall \epsilon > 0$, where $i = 1, \cdots, N, R(t) = \mathrm{Co}\{x_{N+1}(t), \cdots, x_{N+M}(t)\}$.

Let $x_L = [x_{N+1}(t), \cdots, x_{N+M}(t)]^T$, and $x_{dL} = [x_{d1}(t), \cdots, x_{dN}(t)]^T = -L_1^{-1}L_2x_L$. From Lemma 5.1, we can obtain $\inf_{h(t)\in R(t)}\|x_{di} - h(t)\| < \epsilon$ with $i = 1, \cdots, N$. Therefore, the control objective can be transformed as

$$\inf_{h(t)\in R(t)}\|x_i - x_{di}\| < \bar{\epsilon}, \forall \bar{\epsilon} > 0,$$

where $i = 1, \cdots, N$. The containment control errors are defined as $\xi_i = x_i - x_{di},\ i = 1, \cdots, N$.

The singleton fuzzifier, product inference, and the center-defuzzifier are used to deduce the following fuzzy rules [93][94]:

R_{i}: IF x_1 is F_{i}^1, and $\cdots$ and x_{n} is $F_{\mathrm{i}}^{\mathrm{n}}$, THEN y is $B^{\mathrm{i}}(\mathrm{i} = 1, \cdots, \mathrm{r})$, where $\mathrm{x} = [\mathrm{x}_1, \cdots, \mathrm{x}_{\mathrm{n}}] \in R^{\mathrm{n}}$ and $\mathrm{y} \in R$ are the input and output of the fuzzy system, respectively. $F_{\mathrm{i}}^{\mathrm{j}}(\mathrm{j} = 1, \cdots, \mathrm{n})$ and B^{i} are fuzzy sets in R. The fuzzy inference engine performs a mapping from fuzzy sets in R^{n} to a fuzzy set in R based on the IF-THEN rules in the fuzzy rule base and the compositional rule of inference. The fuzzifier maps a crisp point **x** into a fuzzy set A_x in R. The defuzzifier maps a fuzzy set in R to a crisp point in R. Since the strategy of singleton fuzzification, center-average defuzzification and product inference is used, the output of the fuzzy system can be formulated as

$$\mathrm{y}(\mathrm{x}) = \frac{\sum_{j=1}^{\mathrm{N}} \theta_j \prod_{\mathrm{i}=1}^{r} \mu_{F_{\mathrm{i}}^j}(\mathrm{x_i})}{\sum_{j=1}^{\mathrm{N}} \prod_{\mathrm{i}=1}^{r} \mu_{F_{\mathrm{i}}^j}(\mathrm{x_i})}, \tag{6.3}$$

where θ_j is the point at which fuzzy membership function $\mu_{\mathrm{B}^j}(\theta_j)$ achieves its maximum value. It is assumed that $\sum_{j=1}^{\mathrm{N}} \mu_{\mathrm{B}^j}(\theta_j) = 1$. Let $\varphi_j = \frac{\prod_{\mathrm{i}=1}^{r} \mu_{F_{\mathrm{i}}^j}(\mathrm{x_i})}{\sum_{j=1}^{\mathrm{N}} \prod_{\mathrm{i}=1}^{r} \mu_{F_{\mathrm{i}}^j}(\mathrm{x_i})}$, $\varphi(\mathrm{x}) = [\varphi_1(\mathrm{x}), \cdots, \varphi_{\mathrm{n}}(\mathrm{x})]^T$, and $\theta = [\theta_1, \cdots, \theta_{\mathrm{n}}]^T$. Then the fuzzy logic system (6.3) can be rewritten as

$$\mathrm{y}(\mathrm{x}) = \theta^T\varphi(\mathrm{x}). \tag{6.4}$$

It has been proved in [95] that if Gaussian functions are used as membership functions, the following lemma holds.

Lemma 6.2. Let $f(x)$ be a continuous function defined on a compact set Ω. Then, for any constant $\varepsilon > 0$, there exists an FLS such as

$$\sup_{x\in\Omega} |f(x) - \theta^T\varphi(x)| \le \varepsilon,$$

where Ω is a compact region for x. $\theta = [\theta_1, \cdots, \theta_m]^T$ is an adjustable vector. $\varphi(x) =$

$[\varphi_1(x), \cdots, \varphi_m(x)]^T$ is a fuzzy basis function vector. Optimal parameter vector θ^* is defined as

$$\theta^* = \arg\min_{\theta \in U_\theta} [\sup|\hat{f}(x|\theta) - f(x)|],$$

where U_θ is the compact set of θ. Then

$$f(x) = \theta^{*T}\varphi(x) + \varepsilon,$$

where ε is the minimum fuzzy approximation error with an unknown bound.

6.2 Distributed State Feedback Containment Control

6.2.1 State Feedback Containment Controller Design

Distributed containment controllers are proposed as

$$u_i = u_{i1} - u_{i2}, \quad i = 1, \cdots, N, \tag{6.5}$$

where u_{i1} and u_{i2} are designed as follows.

$$u_{i1} = cK[\textstyle\sum_{j=1}^{N} a_{ij}\,(x_i - x_j) + \sum_{j=N+1}^{N+M} a_{ij}\,(x_i - x_j)], \tag{6.6}$$

where $c \in R^+$ is a coupling gain. $K \in R^{m\times n}$ is a controller gain with $K = -B^T P_1$, and P_1 is positive definite satisfying the following Riccati inequality.

$$A^T P_1 + P_1 A - P_1 B B^T P_1 + Q_1 \le 0, \tag{6.7}$$

where Q_1 is positive definite. By Lemma 6.2, the multiple-input multiple-output unknown dynamics $f_i(x_i)$ can be approximated by FLSs as [59][60].

$$f_i(x_i) = \theta_i^{*T}\varphi(x_i) + \varepsilon_i. \tag{6.8}$$

Then, u_{i2} are designed as

$$u_{i2} = \theta_i^T \varphi(x_i), \tag{6.9}$$

where θ_i are the estimations of θ_i^*. Let $x_F = [x_1^T, \cdots, x_N^T]^T$, $x_L = [x_{N+1}^T, \cdots, x_{N+M}^T]^T$. Then, one has

$$\begin{aligned} \dot{x}_F = \; & (I_N \otimes A + cL_1 \otimes BK)x_F + c(L_2 \otimes BK)x_L + \\ & (I_N \otimes B)\tilde{\theta}^T\varphi(x) + (I_N \otimes B)\overline{\varepsilon}, \end{aligned} \tag{6.10}$$

$$\dot{x}_L = (I_M \otimes A)x_L + (I_M \otimes B)\overline{r}, \tag{6.11}$$

where $\tilde{\theta} = \mathrm{diag}(\theta_i^* - \theta_i)$, $\varphi(x) = [\varphi^T(x_1), \cdots, \varphi^T(x_N)]^T$, $\overline{\varepsilon} = [\varepsilon_1^T, \cdots, \varepsilon_N^T]^T$, $\overline{r} = [r_{N+1}^T(t), \cdots, r_{N+M}^T(t)]^T$. Let $e_i = \sum_{j=1}^{N} a_{ij}\,(x_i - x_j) + \sum_{j=N+1}^{N+M} a_{ij}\,(x_i - x_j)$ and $e = [e_1^T, \cdots, e_N^T]^T$.

Then

$$e = (L_1 \otimes I_n)x_F + (L_2 \otimes I_n)x_L. \tag{6.12}$$

Substituting (6.10) and (6.11) into the derivative of (6.12), we have

$$\begin{aligned} \dot{e} = \; & (I_N \otimes A + cL_1 \otimes BK)e + (I_N \otimes B)\tilde{\theta}^T\varphi(x) + \\ & (L_1 \otimes B)\overline{\varepsilon} + (L_2 \otimes B)\overline{r}. \end{aligned} \tag{6.13}$$

6.2.2 Stability Analysis

Theorem 6.1 Consider the multi-agent systems given by (6.1), (6.2). Under Assumption 6.1, the communication graph is directed and has a spanning tree. Select the containment controllers (6.5), (6.6), (6.9) with the coupling gain c satisfying

$$c \geq \frac{1}{2\min\limits_{i=1,\cdots,N}(\lambda_i)}, \tag{6.14}$$

where λ_i are the eigenvalues of L_1. θ_i are updated by

$$\dot{\theta}_i = T_{\theta_i}[\varphi(x)e_i^T P_1 B - \sigma\theta_i], \tag{6.15}$$

where $T_{\theta_i} > 0$, $\sigma > 0$. Then, all the signals in the closed-loop ti-agent systems are UUB, and the containment control errors satisfy

$$\lim_{t\to\infty} \|\xi\| \leq \varrho_1, \tag{6.16}$$

where $\xi = [\xi_1^T, \cdots, \xi_N^T]^T$, $\varrho_1 \in R^+$.

Proof: Consider the Lyapunov function candidate

$$V_1 = \frac{1}{2}e^T(L_1^{-1} \otimes P_1)e + \frac{1}{2}tr(\tilde{\theta}^T T_\theta^{-1}\tilde{\theta}), \tag{6.17}$$

where $T_\theta = \mathrm{diag}(T_{\theta_i})$. Substituting $K = -B^T P_1$ and (6.13) into the derivative of (6.17), we have

$$\begin{aligned} \dot{V}_1 = \; & \frac{1}{2}e^T[L_1^{-1} \otimes (P_1A + A^T P_1) - 2cI_N \otimes P_1BB^T P_1]e + \\ & e^T[(I_N \otimes P_1B)\overline{\varepsilon} + (L_1^{-1}L_2 \otimes P_1B)\overline{r}] + \\ & e^T(I_N \otimes P_1B)\tilde{\theta}^T\varphi(x) - tr(\tilde{\theta}^T\Gamma_\theta^{-1}\dot{\theta}). \end{aligned} \tag{6.18}$$

It follows from (6.15) that

$$\begin{aligned} \dot{V}_1 = \; & \frac{1}{2}e^T[L_1^{-1} \otimes (P_1A + A^T P_1) - 2cI_N \otimes P_1BB^T P_1]e + \\ & e^T[(I_N \otimes P_1B)\overline{\varepsilon} + (L_1^{-1}L_2 \otimes P_1B)\overline{r}] + \sigma tr(\tilde{\theta}^T\theta). \end{aligned} \tag{6.19}$$

By Assumption 6.1 and Lemma 6.1, all the eigenvalues of L_1 have positive real parts. Thus, there

exists a unitary matrix $U \in R^{N\times N}$ such that $U^T L_1^{-1} U = \mathrm{diag}(\lambda_i^{-1}),\ i = 1,\cdots,N$. Let $e = (U \otimes I_n)\varsigma$, where $\varsigma = [{\varsigma_1}^T,\cdots,{\varsigma_N}^T]^T$. Then, it follows from (6.19) that

$$\begin{aligned}\dot{V}_1 = \ & \tfrac{1}{2}\textstyle\sum_{i=1}^{N} \lambda_i^{-1} \varsigma_i^T (P_1 A + A^T P_1 - 2c\lambda_i P_1 B B^T P_1)\varsigma_i + \\ & e^T[(I_N \otimes P_1 B)\overline{\varepsilon} + (L_1^{-1}L_2 \otimes P_1 B)\overline{r}] + \sigma tr(\tilde{\theta}^T\theta).\end{aligned} \tag{6.20}$$

Substituting (6.7) and (6.14) into (6.20), one has

$$\begin{aligned}\dot{V}_1 \le \ & -\tfrac{1}{2}\min_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(Q_1)\|e\|^2 + \\ & e^T[(I_N \otimes P_1 B)\overline{\varepsilon} + (L_1^{-1}L_2 \otimes P_1 B)\overline{r}] + \sigma tr(\tilde{\theta}^T\theta).\end{aligned} \tag{6.21}$$

By Lemma 6.2, $\theta_i^*,\ \epsilon_i$ are bounded and use the fact that $r_k(t)$ are bounded. Then, there exist positive constants $\theta_M,\ \varepsilon_M,\ r_M$, such that $\|\theta^*\|_F \le \theta_M,\ \|\overline{\varepsilon}\| \le \varepsilon_M,\ \|\overline{r}\| \le r_M$. It follows from (6.21) that

$$\begin{aligned}\dot{V}_1 \le \ & -\tfrac{1}{2}\min_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(Q_1)\|e\|^2 - \sigma||\tilde{\theta}||_F^2 + \\ & \|e\|\overline{\sigma}(P_1 B)[\varepsilon_M + \overline{\sigma}(L_1^{-1}L_2)r_M] + \sigma||\tilde{\theta}||_F\theta_M.\end{aligned} \tag{6.22}$$

Rewrite (6.22) in the following matrix form

$$\dot{V}_1 \le -z_1^T \Sigma_1 z_1 + h_1 z_1, \tag{6.23}$$

where

$$\begin{aligned} z_1 &= [\|e\|\ ||\tilde{\theta}||_F]^T, \\ h_1 &= [\overline{\sigma}(P_1 B)[\varepsilon_M + \overline{\sigma}(L_1^{-1}L_2)r_M]\ \sigma\theta_M], \\ \Sigma_1 &= \begin{bmatrix} -\dfrac{1}{2}\min\limits_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(Q_1) & 0 \\ 0 & \sigma \end{bmatrix}. \end{aligned}$$

Noting the fact that $\underline{\sigma}(Q_1) > 0$ and $\sigma > 0$, it follows that Σ_1 is positive definite. Then

$$\dot{V}_1 \le -\overline{\sigma}(\Sigma_1)\|z_1\|^2 + \|h_1\|\|z_1\|. \tag{6.24}$$

Let

$$\begin{aligned} \underline{R}_1 &= \min[\underline{\sigma}(L_1^{-1})\underline{\sigma}(P_1),\ T_{\theta i}^{-1}], \\ \overline{R}_1 &= \max[\overline{\sigma}(L_1^{-1})\overline{\sigma}(P_1),\ T_{\theta i}^{-1}]. \end{aligned}$$

Then

$$\tfrac{1}{2}\underline{R}_1\|z_1\|^2 \le V_1 \le \tfrac{1}{2}\overline{R}_1\|z_1\|^2. \tag{6.25}$$

From (6.24) and (6.25), we have

$$\dot{V}_1 \le \alpha V_1 + \beta\sqrt{V_1}, \tag{6.26}$$

where $\alpha = \frac{2\underline{\sigma}(\Sigma_1)}{\overline{R}_1}, \beta = \frac{\sqrt{2}\|h_1\|}{\sqrt{\underline{R}_1}}$. Then

$$\sqrt{V_1(t)} \leq \sqrt{V_1(0)}e^{-\frac{\alpha}{2}t} + \frac{\beta}{\alpha}(1 - e^{-\frac{\alpha}{2}t}). \tag{6.27}$$

Since $\lim_{t\to\infty} \sqrt{V_1(t)} = \frac{\beta}{\alpha}$, we obtain all signals in the closed-loop multi-agent systems are UUB. Then

$$\|e\| \leq \frac{\|h_1\|\overline{R}_1}{\underline{\sigma}(\Sigma_1)\sqrt{\underline{R}_1\,\underline{\sigma}(P_1)}}. \tag{6.28}$$

Then, it follows from (6.12) that

$$\|(L_1 \otimes I_n)x_F + (L_2 \otimes I_n)x_L\| \leq \frac{\|h\|\overline{R}}{\underline{\sigma}(\Sigma)\sqrt{\underline{R}\,\underline{\sigma}(P)}}. \tag{6.29}$$

Then, we get (6.16) with $\varrho_1 = \frac{\|h_1\|\overline{R}_1}{\underline{\sigma}(L_1^{-1})\underline{\sigma}(\Sigma_1)\sqrt{\underline{R}_1\,\underline{\sigma}(P_1)}}$. It means that the states of the followers converge to the convex hull formed by those of the leaders with the containment errors being UUB. The containment control problem is solved.

Remark 6.1. In [59]-[62], the distributed containment control approaches were proposed for nonlinear Lagrangian systems. However, the previous approaches cannot be applied to the nonlinear multi-agent systems (6.1), (6.2). Therefore, it is significant to investigate the distributed containment control problem for more general nonlinear multi-agent systems in the presence of unknown dynamics.

6.3 Distributed Output Feedback Containment Control

The containment control designed in Section 6.2 is based on the assumption that the states of the systems are directly measured. However, in practice, state variables are often unmeasured for many nonlinear systems. Therefore, the output feedback containment control will be designed for the multi-agent systems with unmeasured states consisting by (6.1) and (6.2) in this, which is very important in both theory and real-world applications. We assume here that the states of the leaders are measurable and $r_k(t) = 0$.

6.3.1 Output Feedback Containment Controller Design

Design distributed observers to estimate the unmeasurable states. Let $\hat{x}_i$ be the estimations of x_i. Similar to [96], the observers are designed in the following form

$$\begin{aligned} \dot{\hat{x}}_i &= A\hat{x}_i + B[u_i + \theta_i^T \varphi_i(\hat{x}_i)] + c_o F\tilde{y}_i, \\ \hat{y}_i &= C\hat{x}_i, \end{aligned} \tag{6.30}$$

where $c_o \in R^+$ is a coupling gain. $\tilde{y}_i = y_i - \hat{y}_i$. $F \in R^{n\times m}$ is an observer gain with $F = {P_2}^{-1}C^T$, and P_2 is positive definite satisfying the following linear matrix inequality (LMI)

$$\begin{bmatrix} P_2A + A^TP_2 - C^TC + Q_2 & M \\ M^T & -\frac{I_m}{\gamma} \end{bmatrix} \leq 0, \tag{6.31}$$

where γ is an adjustable parameter to guarantee the existence of P_2. $M = C^T - P_2B$ and Q_2 is positive definite.

Based on the developed observers, the output feedback containment controllers are designed in (6.5) with

$$u_{i1} = cK[\textstyle\sum_{j=1}^{N} a_{ij}\,(\hat{x}_i - \hat{x}_j) + \sum_{j=N+1}^{N+M} a_{ij}\,(\hat{x}_i - x_j)], \tag{6.32}$$

$$u_{i2} = \theta_i^T \varphi(\hat{x}_i), \tag{6.33}$$

Let $\hat{x}_F = [\hat{x}_1^T, \cdots, \hat{x}_N^T]^T$. Then, one has

$$\begin{aligned} \dot{\hat{x}}_F &= (I_N \otimes A + cL_1 \otimes BK)\hat{x}_F + c(L_2 \otimes BK)x_L + c_o(I_N \otimes FC)\tilde{x}, \\ \dot{x}_L &= (I_M \otimes A)x_L, \end{aligned} \tag{6.34}$$

where $\tilde{x} = [\tilde{x}_1^T, \cdots, \tilde{x}_N^T]^T$ with $\tilde{x}_i = x_i - \hat{x}_i$ being the state estimation errors. The state estimation error equation is described by

$$\dot{\tilde{x}} = (I_N \otimes A - c_o I_N \otimes FC)\tilde{x} + (I_N \otimes B)[\tilde{\theta}^T\varphi(\hat{x}) + \theta^{*T}(\varphi(x)) - \varphi(\hat{x})) + \bar{\varepsilon}]. \tag{6.35}$$

Let $\hat{e}_i = \sum_{j=1}^{N} a_{ij}\,(\hat{x}_i - \hat{x}_j) + \sum_{j=N+1}^{N+M} a_{ij}\,(\hat{x}_i - x_j)$ and $\hat{e} = [\hat{e}_1^T, \cdots, \hat{e}_N^T]^T$. Then

$$\hat{e} = (L_1 \otimes I_n)\hat{x}_F + (L_2 \otimes I_n)x_L. \tag{6.36}$$

Then

$$\dot{\hat{e}} = (I_N \otimes A + cL_1 \otimes BK)\hat{e} + c_o(L_1 \otimes FC)\tilde{x}. \tag{6.37}$$

6.3.2 Stability Analysis

Theorem 6.2 Consider the multi-agent systems given by (6.1), (6.2). Under Assumption 6.1, the

communication graph is directed and has a spanning tree. Select the output feedback containment controllers (6.5), (6.32) and (6.33) with the coupling gains c and c_o satisfying (6.14) and (6.38)

$$c_o \geq \frac{1}{2}. \tag{6.38}$$

θ_i are updated by

$$\dot{\theta}_i = T_{\theta_i}[\varphi_i(\hat{x}_i)\tilde{y}_i{}^T - (\sigma I_{nm} + \frac{1}{2\gamma}\varphi_i(\hat{x}_i)\varphi_i^T(\hat{x}_i))\theta_i]. \tag{6.39}$$

where $T_{\theta_i} > 0, \sigma > 0, \gamma > 0$. Then, all the signals in the closed-loop systems are UUB, and the containment control errors satisfy

$$\lim_{t\to\infty} \|\xi\| \leq \varrho_2, \tag{6.40}$$

where $\varrho_2 \in R^+$.

Proof: Consider the Lyapunov function candidate

$$V_2 = \frac{1}{2}\tilde{x}^T(I_N \otimes P_2)\tilde{x} + \frac{1}{2}tr(\tilde{\theta}^T T_\theta^{-1}\tilde{\theta}), \tag{6.41}$$

Substituting $F = P_2{}^{-1}C^T$ and (6.35) into the derivative of (6.41), we have

$$\begin{aligned} \dot{V}_2 = \ & \frac{1}{2}\tilde{x}^T[I_N \otimes (P_2A + A^TP_2 - 2c_oC^TC)]\tilde{x} + \\ & \tilde{x}^T(I_N \otimes P_2B)[\tilde{\theta}^T\varphi(\hat{x}) + \theta^{*T}(\varphi(x)) - \varphi(\hat{x})) + \overline{\varepsilon}] + tr(\tilde{\theta}^T\Gamma_\theta^{-1}\dot{\tilde{\theta}}). \end{aligned} \tag{6.42}$$

By (6.39) and $M = C^T - P_2B$, we have

$$\begin{aligned} \dot{V}_2 = \ & \frac{1}{2}\tilde{x}^T[I_N \otimes (P_2A + A^TP_2 - 2c_oC^TC)]\tilde{x} + \\ & \tilde{x}^T(I_N \otimes P_1B)[\theta^{*T}(\varphi(x) - \varphi(\hat{x})) + \overline{\varepsilon}] - \\ & \sigma tr(\tilde{\theta}^T\theta) + \textstyle\sum_{i=1}^N \tilde{x}_i^T M\tilde{\theta}_i^T\varphi_i(\hat{x}_i) - \frac{1}{2\gamma}\sum_{i=1}^N \varphi_i^T(\hat{x}_i)\theta_i\,\tilde{\theta}_i^T\varphi_i(\hat{x}_i). \end{aligned} \tag{6.43}$$

Using the Young's inequality

$$\tilde{x}_i^T M\tilde{\theta}_i^T\varphi_i(\hat{x}_i) \leq \frac{\gamma}{2}\tilde{x}_i^T MM^T\tilde{x}_i + \frac{1}{2\gamma}\varphi_i{}^T(\hat{x}_i)\tilde{\theta}_i\tilde{\theta}_i^T\varphi_i(\hat{x}_i). \tag{6.44}$$

Then

$$\begin{aligned} \dot{V}_2 \leq \ & \frac{1}{2}\tilde{x}^T[I_N \otimes (P_2A + A^TP_2 + \gamma MM^T - 2c_oC^TC)]\tilde{x} + \\ & \tilde{x}^T(I_N \otimes P_1B)[\theta^{*T}(\varphi(x) - \varphi(\hat{x})) + \overline{\varepsilon}] - \sigma tr(\tilde{\theta}^T\theta) - \\ & \frac{1}{2\gamma}\textstyle\sum_{i=1}^N \varphi_i^T(\hat{x}_i)\theta_i^*\,\tilde{\theta}_i^T\varphi_i(\hat{x}_i). \end{aligned} \tag{6.45}$$

It follows from (31) and (38) that

$$\begin{aligned} \dot{V}_2 \leq \ & -\frac{1}{2}\tilde{x}^T(I_N \otimes Q_2)\tilde{x} + \tilde{x}^T(I_N \otimes P_1B)[\theta^{*T}(\varphi(x) - \varphi(\hat{x})) + \overline{\varepsilon}] + \\ & \sigma tr(\tilde{\theta}^T\theta) - \frac{1}{2\gamma}\textstyle\sum_{i=1}^N \varphi_i^T(\hat{x}_i)\theta_i^*\,\tilde{\theta}_i^T\varphi_i(\hat{x}_i). \end{aligned} \tag{6.46}$$

By Lemma 6.2, it follows that $\varphi_i(x_i)$ and $\varphi_i(\hat{x}_i)$ are bounded. Then, there exist positive constants ρ_M and φ_M, such that $\|\theta^{*T}(\varphi(x) - \varphi(\hat{x})) + \overline{\varepsilon}\| \leq \rho_M$, $\|\varphi_i(\hat{x}_i)\| \leq \varphi_M$. Then, (6.46) can be rewritten as

$$\dot{V}_2 \leq \quad -\frac{1}{2}\underline{\sigma}(Q_2)\|\tilde{x}\|^2 - \sigma||\tilde{\theta}||_F^2 + \|\tilde{x}\|\overline{\sigma}(P_1B)\rho_M + ||\tilde{\theta}||_F(\sigma\theta_M + \frac{1}{2\gamma}\varphi_M^2\theta_M). \tag{6.47}$$

Let

$$\begin{aligned} z_2 &= [\|\tilde{x}\| \ ||\tilde{\theta}||_F]^T, \\ h_2 &= [\overline{\sigma}(PB)\rho_M \ \sigma\theta_M + \frac{1}{2\gamma}\varphi_M^2\theta_M], \\ \Sigma_2 &= \begin{bmatrix} -\frac{1}{2}\underline{\sigma}(Q_2) & 0 \\ 0 & \sigma \end{bmatrix}. \end{aligned}$$

Rewrite (6.47) in the following form

$$\dot{V}_2 \leq -z_2^T\Sigma_2 z_2 + h_2 z_2, \tag{6.48}$$

Using the similar analysis process above, it follows that $\tilde{x}$ and $\tilde{\theta}$ are UUB and the bound of $\tilde{x}$ is given by

$$\|\tilde{x}\| \leq \varrho_{o1}, \tag{6.49}$$

where $\varrho_{o1} = \frac{\|h_2\|\overline{R}_2}{\underline{\sigma}(\Sigma_2)\sqrt{\underline{R}_2\,\underline{\sigma}(P_2)}}$ with $\underline{R}_2 = \min[\underline{\sigma}(P),\ T_{\theta i}^{-1}]$, $\overline{R}_2 = \max[\overline{\sigma}(P),\ T_{\theta i}^{-1}]$.

Consider another Lyapunov function candidate

$$V_3 = \frac{1}{2}\hat{e}^T(L_1^{-1} \otimes P_1)\hat{e}. \tag{6.50}$$

Substituting $K = -B^TP_1$ and (6.37) into the derivative of (6.50), we have

$$\dot{V}_3 = \quad \frac{1}{2}\hat{e}^T[L_1^{-1} \otimes (P_1A + A^TP_1) - 2cI_N \otimes P_1BB^TP_1]\hat{e} + \hat{e}^T(c_oI_N \otimes FC)\tilde{x}. \tag{6.51}$$

Then

$$\dot{V}_3 \leq -\frac{1}{2}\min_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(Q_1)\|\hat{e}\|^2 + \|\hat{e}\|\overline{\sigma}(FC)||\tilde{x}||. \tag{6.52}$$

Noting the fact that

$$\|\hat{e}\| \geq \frac{2\overline{\sigma}(FC)}{\min_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(Q_1)}||\tilde{x}||$$

indicates $\dot{V}_3 \leq 0$. Considering (6.36) and (6.49), it follows that

$$\|\hat{x}_F - x_{dL}\| \leq \frac{2\overline{\sigma}(FC)}{\min\limits_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(L_1^{-1})\underline{\sigma}(Q_1)}\varrho_{o1}. \tag{6.53}$$

Note the fact that $\|\xi\| \leq \|\hat{x}_F - x_{dL}\| + \|\tilde{x}\|$. Then, (6.40) is obtained with

$$\varrho_2 = \frac{2\overline{\sigma}(FC)}{\min\limits_{i=1,\cdots,N}(\lambda_i^{-1})\underline{\sigma}(L_1^{-1})\underline{\sigma}(Q_1)}\varrho_{o1} + \varrho_{o1}. \tag{6.54}$$

Then, the containment control problem is solved.

Remark 6.2. In [97], consensus scheme was developed for nonlinear multi-agent systems (6.1), i.e., the proposed method can guarantee all states of the followers synchronize to that of a single leader. In this paper, containment control approach is designed to guarantee all states of the followers stay in a dynamic convex hull formed by multiple leaders.

6.4 Simulation Results

A simulation example is provided to show the effectiveness of the proposed distributed output feedback containment controllers. A vehicle platoon with multiple leaders is considered as a second-order model. It is the following follower dynamics [98].

$$A = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}, B = \begin{bmatrix}0\\1\end{bmatrix}, C = [1 \quad 0], f_{i2}(v_i) = -k/m\, v_i \ (i = 1,\cdots,4).,$$

Choose fuzzy membership functions as

$$\begin{aligned}\mu_1(v_i) &= \frac{1}{1+\exp[-4(v_i/2-\pi/2)]},\\ \mu_2(v_i) &= \exp(-v_i^2),\\ \mu_3(v_i) &= \frac{1}{1+\exp[-4(v_i/2+\pi/2)]}.\end{aligned}$$

The communication graph is described as Figure 6.7.

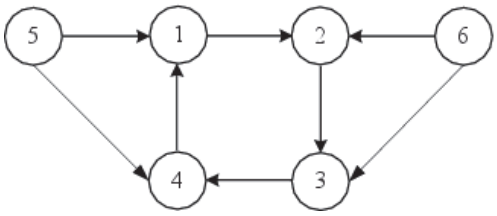


Figure 6.1 Communication graph.

In simulation, the initial value of the states is set as

$$x_1(0)=[0.13;0];\ x_2(0)=[0.35;0.01];\ x_3(0)=[0.4;0.02];\ x_4(0)=[0.22;0.03];$$
$$x_5(0)=[0.2;0.01];\ x_6(0)=[0.5;0.01];$$

$$\hat{x}_1(0)=[0.12;0];\ \hat{x}_2(0)=[0.35;0.01];\ \hat{x}_3(0)=[0.4;0.02];\ \hat{x}_4(0)=[0.22;0.03;];$$

$$P_1=\begin{bmatrix}1.6170 & 1.6170\\ 1.6170 & 3.2340\end{bmatrix}, P_2=\begin{bmatrix}0.8586 & -0.8706\\ -0.8706 & 1.7041\end{bmatrix},$$

$$K=[-1.6170 \quad -3.2340], F=\begin{bmatrix}2.4165\\ 1.2345\end{bmatrix}$$

In simulation, $c_o = 10$, $c = 2$, $T_{\theta i} = 10$, $\sigma = 0.01$. The containment results and containment errors using the output feedback containment controllers are shown in Figures 6.2, 6.3. It can be observed that the proposed containment scheme can realize that the states of followers converge to the convex hull formed by those of the leaders, i.e. the states of all followers stay in the area formed by the leaders. Figure 6.4 shows the states of developed observer, from which we can see that the designed observer can estimate unmeasurable states with the estimation errors in a small neighborhood of the origin. The profiles of the designed distributed output feedback containment controllers are shown in Figure 6.5. It can be observed that the designed containment controllers guarantee both the stability and good containment performance of the closed-loop multi-agent systems with unknown dynamics.

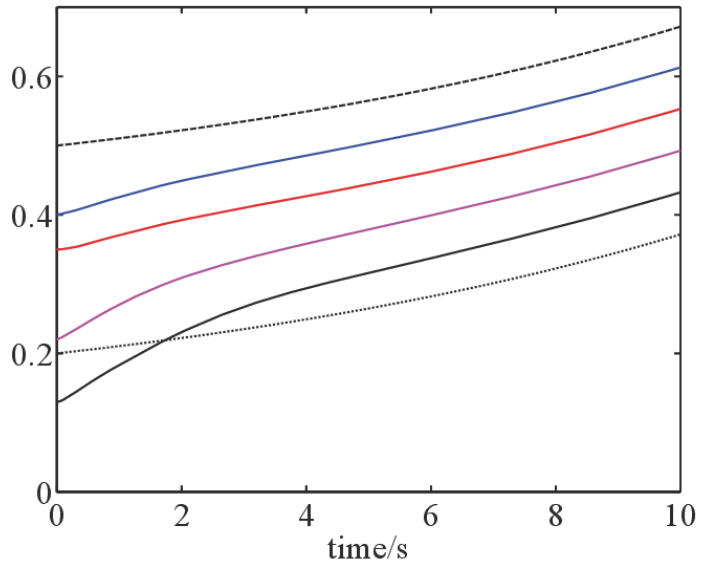


Figure 6.2 Follower states x_{i1} (solid line), leader states x_{k1} (dotted line).

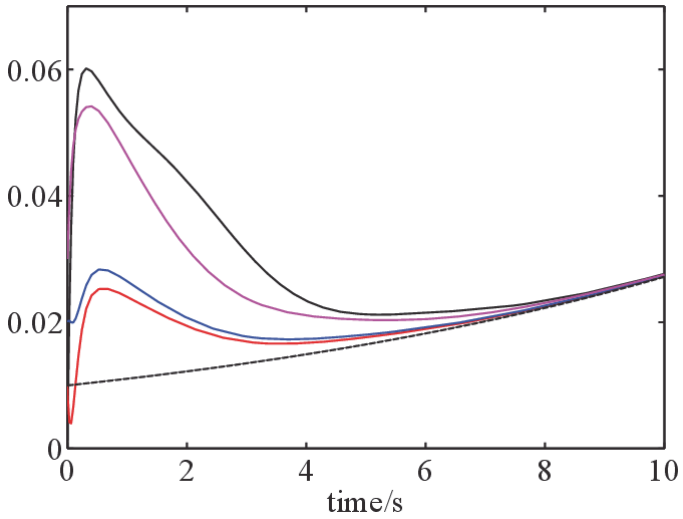


Figure 6.3 Follower states x_{i2} (solid line), leader states x_{k2} (dotted line).

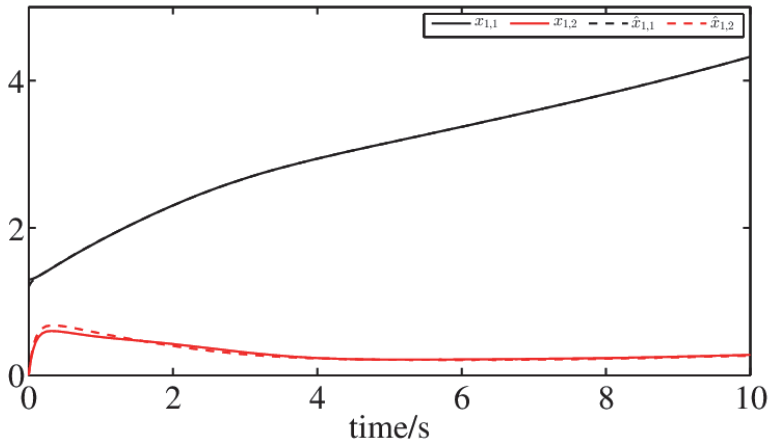


Figure 6.4 The estimation effect of observers.

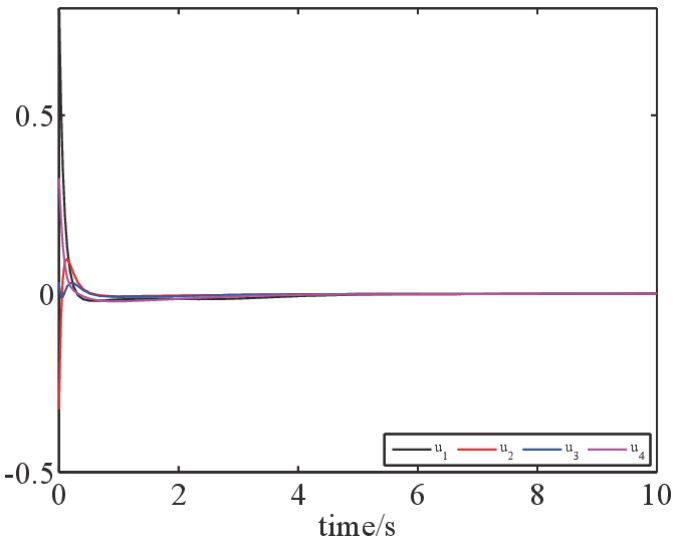


Figure 6.5 Profiles of control inputs.

Chapter 7 Conclusion and Future Works

7.1 Conclusion

This dissertation mainly studies several problems of the vison-based front vehicle distance measurement system and the coordinated control of uncertain nonlinear vehicle platoon systems. The specific research work and related research results obtained are as follows:

A fast and effective moving object detection method for a moving camera is proposed. The global motion is estimated through tracking the grid-based key points using optical flow. It maintains less processing time. The adaptive background model can remove more noise and extract more complete objects. The improved background subtraction can effectively reduce the influences of the hole, shadow and lighting change. The experimental results show that the proposed method has fast speed and high accuracy for detecting moving objects for a moving camera, and it can overcome the influences of the noise and lighting change.

A front vehicle distance measurement method is proposed. It locates the license plate of the front vehicle in the moving object area after removing the background. The license plate is positioned by extracting the texture in the vertical direction of the license plate area. The front vehicle distance is estimated from the observed license plate height by using the logarithmic equation with three fixed parameters. The experimental results show that the proposed method has better performance and application value.

A novel adaptive fuzzy consensus control method via neurodynamic optimization for networked output-constrained second-order nonlinear systems with unknown dynamics is proposed. The unknown dynamics was dealt with FLSs, and the approximation error was counteracted by a robust term. Backstepping design combining with a barrier Lyapunov function was utilized to construct the virtual control law and the actual control law. In addition, a neurodynamics-based governor was introduced to generate an optimal virtual control signal, which can further limit the velocity signal in a valid range and minimize the control input. It was proved that all closed-loop signals remain bounded without violating the output constraints.

A containment control method is proposed for uncertain nonlinear multi-agent systems with measurable and unmeasurable states. Based on FLSs identifying the unknown dynamics of the followers, distributed state feedback containment controllers were designed first. Then, adaptive fuzzy observers were designed to estimate the unmeasurable states. Based on the developed observers, distributed output feedback containment controllers were designed. Both of the developed containment controllers ensure that the states of the followers converge to the convex hull formed by those of the leaders with the containment control errors in a small residual set.

7.2 Future Works

The research on the vison-based front vehicle distance measurement and the coordination control problem of vehicle platoon systems has achieved fruitful results, but there are still many problems to be further studied and discussed. The next work will be studied in the following aspects:

More complex features (shape, texture, etc) are used to improve the moving object detection performance.

Different algorithms are used for vehicle image positioning for different resolution vehicle images. The accuracy of vehicle detection and distance estimation still needs to be improved for various road environments.

The consensus control via neurodynamic optimization for more complex nonlinear systems in the presence output constraints.

The containment control problem of uncertain nonlinear multi-agent systems with time-delay.

Publications

Journal Publications:

1. Yang Yu and Kang-Hyun Jo, "Adaptive Fuzzy Containment Control for Uncertain Nonlinear Multiagent Systems." Mathematical Problems in Engineering, Vol.14, doi:10.1155/2014/840517, 2014, IF 1.145, Published.

2. Yang Yu and Kang-Hyun Jo, "Output Feedback Fault-tolerant Control for a Class of Discrete-Time Fuzzy Bilinear Systems." International Journal of Control, Automation and Systems, Vol.14, No.2, pp. 486-494, 2016, IF. 2.173, Published.

3. Yang Yu, Wei Wang and Kang-Hyun Jo, "Adaptive Consensus Control of Output-Constrained Second-order Nonlinear Systems via Neurodynamic Optimization." Neurocomputing, Vol.295, pp.1-7, 2018, IF. 3.241, Published.

4. Yang Yu, Laksono Kurnianggoro and Kang-Hyun Jo, "Moving Object Detection for a Moving Camera Based on Global Motion Compensation and Adaptive Background Model." International Journal of Control, Automation and Systems, 2018, IF. 2.173, Under review.

Conference Publications:

1. Yang Yu, Laksono Kurnianggoro, and Kang-Hyun Jo, "Design of Automatic Water Cannon System based on 2D Laser Scanner." Institute of Control, Robotics and Systems 2015 (ICROS 2015), vol. 5, pp. 37-38, 2015.

2. Yang Yu, Laksono Kurnianggoro, and Kang-Hyun Jo, "Design of intelligent car washing system." SICE Annual Conference 2015 (SICE 2015), pp. 1447-1450, 2015.

3. Yang Yu, Laksono Kurnianggoro, and Kang-Hyun Jo, "The Car Washing Control Method using

3D Contour Segmentation." 15th International Conference on Control, Automation and Systems (ICCAS 2015). pp. 935-937, 2015.

4. Yang Yu, Laksono Kurnianggoro, Wahyono, and Kang-Hyun Jo, "Online Programming Design of Distributed System based on Multi-level Storage." Intelligent Computing Methodologies, International Conference on Intelligent Computation 2016 (ICIC 2016), vol 9773, pp 745-752, 2016.

5. Yang Yu, Danilo Caceres Hernandez, and Kang-Hyun Jo, "Lane Line Detection based on Inverse Perspective Mapping and Fuzzy C-Means." International Conference on ICT Robotics 2016 (ICT-ROBOT 2016).

6. Yang Yu, and Kang-Hyun Jo, "Real-Time Gesture Recognition and Tracking based on Gabor Feature and Localized Contour Sequence." Proceedings of the SICE Annual Conference 2016 (SICE 2016), pp. 1686-1689, 2016.

7. Yang Yu, and Kang-Hyun Jo, "Lane Detection based on Color Probability Model and Fuzzy Clustering." Proc. SPIE 10615, Ninth International Conference on Graphic and Image Processing (ICGIP 2017).

8. Yang Yu, and Kang-Hyun Jo, "Lane Detection and Vehicle Localization based on Particle Filter." 17th International Conference on Control, Automation and Systems (ICCAS 2017), pp. 1046-1048, 2017.

9. Yang Yu, Laksono Kurnianggoro, and Kang-Hyun Jo, "Vehicle Contour Segmentation from 3D Point Cloud." Changwon International Conference on Intelligent Robot and Convergence Industry (CICIRO 2017), pp. 1-3, 2017.

10. Yang Yu, and Kang-Hyun Jo, "Contour Segmentation Based on Density Gradient and Region Growing." 18th International Conference on Control, Automation and Systems (ICCAS 2018),

2018.

References

[1] S. E. Shladover, C. A. Desoer, and J. K. Hedrick, “Automated vehicle control developments in the PATH program,” IEEE Transactions on Vehicular Technology, vol. 40, no. 1, pp. 114-130, 1991.

[2] E. Coelingh and S. Solyom, “All aboard the robotic road train,” IEEE Spectrum, vol. 49, no. 11, pp. 34-39, 2012.

[3] D. Swaroop, J. K. Hedrick, and C. C. Chien, “A comparision of spacing and headway control laws for automatically controlled vehicles,” Vehicle System Dynamics, vol. 23, no. 1, pp. 597-625, 1994.

[4] S. Minaeian, J. Liu, and Y. J. Son, “Effective and efficient detection of moving targets from a UAV’s camera,” IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 2, pp. 497-506, 2018.

[5] L. Kurnianggoro, Wahyono, Y. Yu, D. C. Hernandez, and K. H. Jo, “Online background-subtraction with motion compensation for freely moving Camera,” International Conference on Intelligent Computing, vol. 9772, pp. 569-578, 2016.

[6] Y. Lin, Y. Tong, Y. Cao, Y. Zhou, and S. Wang, “Visual-attention-based background modeling for detecting infrequently moving objects,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 27, no. 6, pp. 1208-1221, 2017.

[7] Y. Wu, X. He, and T. Q. Nguyen, “Moving object detection with a freely moving camera via background motion subtraction,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 27, no. 2, pp. 236-248, 2017.

[8] L. Maddalena, A. Petrosino, “The SOBS algorithm: what are the limits?” IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 21-26, 2012.

[9] H. H. Kim, J. K. Park, J. H. Oh, and D. J. Kang, “Multi-task convolutional neural network system for license plate recognition,” International Journal of Control Automation and Systems, vol. 15, no. 6, pp. 2942-2949, 2017.

[10] Y. Wang, Z. M. Luo, and P. M. Jodoin, “Interactive deep learning method for segmenting

moving objects," Pattern Recognition Letters, vol. 96, pp. 66-75, 2017.

[11] D. Zhou, V. Frémont, B. Quost, Y. Dai, and H. Li, "Moving object detection and segmentation in urban environments from a moving platform," Image and Vision Computing, vol. 68, pp. 76-87, 2017.

[12] T. Chen and S. Lu, "Object-level motion detection from moving cameras," IEEE Transactions on Circuits and Systems for Video Technology, vol. 27, no. 11, pp. 2333-2343, 2017.

[13] S. Kim, D. W. Yang, and H. W. Park, "A disparity-based adaptive multihomography method for moving target detection based on global motion compensation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 8, pp. 1407-1420, 2016.

[14] C. H. Yeh, C. Y. Lin, K. Muchtar, H. E. Lai, and M. T. Sun, "Three-pronged compensation and hysteresis thresholding for moving object detection in real-Time video surveillance," IEEE Transactions on Industrial Electronics, vol. 64, no. 6, pp. 4945-4955, 2017.

[15] T. Minematsu, H. Uchiyama, A. Shimada, H. Nagahara, and R. I. Taniguchi, "Adaptive background model registration for moving cameras," Pattern Recognition Letters, vol. 96, pp. 86-95, 2017.

[16] D. Avola, L. Cinque, G.L. Foresti, C. Massaroni, and D. Pannone, "A keypoint-based method for background modeling and foreground detection using a PTZ camera," Pattern Recognition Letters, vol. 96, pp. 96-105, 2017.

[17] A. Zheng, L. Zhang, W. Zhang, C. Li, J. Tang, and B. Luo, "Local-to-global background modeling for moving object detection from non-static cameras," Multimedia Tools and Applications, vol. 76, no. 8, pp. 11003-11019, 2017.

[18] K. Yun, J. Lim, and J. Y. Choi, "Scene conditional background update for moving object detection in a moving camera," Pattern Recognition Letters, vol. 88, no. 1, pp. 57-63, 2017.

[19] W. Y. Wang, M. C. Lu, and H. L. Kao, "Nighttime vehicle distance measuring systems," IEEE Transactions on Circuits and Systems, vol. 54, no. 1, pp. 81-85, 2007.

[20] A. M. Ibarra, O. J. M. Perez, and C. Torre-Ferrero, "Monovision-based vehicle detection, distance and relative speed measurement in urban traffic," Intelligent Transport Systems, IET, vol.

8, no. 8, pp. 655-664, 2014.

[21] M. Sarfraz, M. J. Ahmed, and S. A. Ghazi, “Saudi Arabian license plate recognition system,” International Conference on Geometric Modeling and Graphics, pp. 36-41, 2003.

[22] Z. Qin, S. Shi, and J. Xu, “Method of license plate location based on comer feature,” World Congress on Intelligent Control and Automation, vol. 2, pp. 8645-8649, 2006.

[23] Y. S. Soh, B. T. Chun, and H. S. Yoon, “Design of real time vehicle identification system,” International Conference on Systems, Man, and Cybernetics, vol. 3, pp. 2147-2152, 1994.

[24] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, and V. Loumos, “A license plate-recognition algorithm for intelligent transportation system applications,” IEEE Transactions on Intelligent Transportation Systems, vol. 7, no. 3, pp. 377-392, 2006.

[25] X. Shi, W. Zhao, and Y. Shen, “Automatic License Plate Recognition System Based on Color Image Processing,” Lecture Notes in Computer Science, vol. 3483, no. 4, pp. 1159-1168, 2005.

[26] J. A. Fax and R. M. Murray, “Information flow and cooperative control of vehicle formations,” IEEE Transactions on Automatic Control, vol. 49, no. 9, pp. 1465-1476, 2004.

[27] Z. G. Hou, L. Cheng, and M. Tan, “Decentralized robust adaptive control for the multiagent system consensus problem using neural networks,” IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 39, no. 3, pp. 636-647, 2009.

[28] A. Das and F. L. Lewis, “Distributed adaptive control for synchronization of unknown nonlinear networked systems,” Automatica, vol. 46, no. 12, pp. 2014-2021, 2010.

[29] J. H. Qin, Q. C. Ma, Y. Shi, and Long Wang, “Recent advances in consensus of multi-agent aystems: A brief survey,” IEEE Transactions on Industrial Electronics, vol. 64, no. 6, pp. 4972-4983, 2017.

[30] Y. G. Hong, G. R. Chen, and L. Bushnell, “Distributed observers design for leader-following control of multi-agent networks,” Automatica, vol. 44, no. 3, pp. 846-850, 2008.

[31] W. W. Yu, G. R. Chen, and Cao M, “Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems,” Automatica, vol. 46, no. 6, pp. 1089-1095, 2010.

[32] W. S. Chen, X. B. Li, and L. C. Jiao, “Quantized consensus of secondorder continuous-time

multi-agent systems with a directed topology via sampled data," Automatica, vol. 49, no. 7, pp. 2236-2242, 2013.

[33] Y. Zhao, Z. D. Duan, G. H. Wen, and Y. J. Zhang, "Distributed finitetime tracking control for multi-agent systems: An observer-based approach," Systems & Control Letters, vol. 62, no. 1, pp. 22-28, 2013.

[34] Z. Y. Meng, Z. L. Lin, and W. Rend, "Robust cooperative tracking for multiple non-identical second-order nonlinear systems," Automatica, vol. 49, no. 8, pp. 2363-2372, 2013.

[35] J. Mei, W. Ren, and J. Chen, "Distributed consensus of second-order multi-agent systems with heterogeneous unknown inertias and control gains under a directed graph," IEEE Transactions on Automatic Control, vol. 61, no. 8, pp. 2019-2034, 2016.

[36] C. C. Hua, X. You, and X. P. Guan, "Adaptive leader-following consensus for second-order time-varying nonlinear multiagent systems," IEEE Transactions on Cybernetics, vol. 47, no. 6, pp. 1532-1539, 2017.

[37] H. Y. Li, L. J. Wang, H. P. Du, and A. Boulkroune, "Adaptive fuzzy tracking control for strict-feedback systems with input delay," IEEE Transactions on Fuzzy Systems, vol. 25, no. 3, pp. 642-652, 2017.

[38] Y. J. Liu and S. C. Tong, "Barrier Lyapunov functions for Nussbaum gain adaptive control of full state constrained nonlinear systems," Automatica, vol. 76, pp. 143-152, 2017.

[39] Z. H. Peng, D. Wang, and J. Wang, "Predictor-based neural dynamic surface control for uncertain nonlinear systems in strict-feedback form," IEEE Transactions on Neural Networks and Learning Systems, vol. 28, no. 9, pp. 2156-2167, 2017.

[40] L. Cheng, Z. G. Hou, M. Tan, Y. Z. Lin, and W. J. Zhang, "Neuralnetwork-based adaptive leader-following control for multiagent systems with uncertainties," IEEE Transactions on Neural Networks and Learning Systems, vol. 21, no. 8, pp. 1351-1358, 2010.

[41] A. Das and F. L. Lewis, "Cooperative adaptive control for synchronization of second-order systems with unknown nonlinearities," International Journal of Robust and Nonlinear Control, vol. 21, no. 3, pp. 1509-1524, 2011.

[42] S. J. Yoo, “Distributed consensus tracking for multiple uncertain nonlinear strict-feedback systems under a directed graph,” IEEE Transactions on Neural Networks and Learning Systems, vol. 24, no. 4, pp. 666-672, 2013.

[43] C. L. Philip Chen, C. E. Ren, and T. Du, “Fuzzy observed-based adaptive consensus tracking control for second-order multiagent systems With heterogeneous nonlinear dynamics,” IEEE Transactions on Fuzzy Systems, vol. 24, no. 4, pp. 906-915, 2016.

[44] Y. J. Liu and S. C. Tong, “Barrier Lyapunov functions-based adaptive control for a class of nonlinear pure-feedback systems with full state constraints,” Automatica, vol. 64, pp. 70-75, 2016.

[45] G. Ferrari-Trecate, L. Galbusera, M. P. E. Marciandi, and R. Scattolini, “Model predictive control schemes for consensus in multi-agent systems with single-and double-integrator dynamics,” IEEE Transactions on Automatic Control, vol. 54, no. 11, pp. 2560-2572, 2009.

[46] J. H. Qin, W. M. Fu, W. X. Zheng, H. J. Gao, “On the bipartite consensus for generic linear multiagent systems with input saturation,” IEEE Transactions on Cybernetics, vol. 47, no. 8, pp. 1948-1958, 2017.

[47] J. T. Lyu, J. H. Qin, D. Gao, and Q. Q. Liu, “Consensus for constrained multi-agent systems with input saturation,” International Journal of Robust and Nonlinear Control, vol. 26, no. 14, pp. 2977-2993, 2016.

[48] X. Z. Jin, Y. G. He, and Y. G. He, “Finite-time robust fault-tolerant control against actuator faults and saturations,” IET Control Theory & Applications, vol. 11, no. 4, pp. 550-556, 2017.

[49] Z. X. Liu and Z. Q. Chen, “Discarded consensus of network of agents with state constraint,” IEEE Transactions on Automatic Control, vol. 57, no. 11, pp. 2869-2874, 2012.

[50] W. C. Meng, Q. M. Yang, J. N. Si, and Y. X. Sun, “Consensus control of nonlinear multiagent systems with time-varying state constraints,” IEEE Transactions on Cybernetics, doi: 10.1109/TCYB.2016.2629268.

[51] Z. Zhao, W. He, and S. S. Ge, “Adaptive neural network control of a fully actuated marine surface vessel with multiple output constraints,” IEEE Transactions on Control Systems Technology, vol. 19, no. 1, pp. 1536-1543, 2011.

[52] W. He, S. Zhang, and S. S. Ge, “Adaptive control of a flexible crane system with the boundary output constraint,” IEEE Transactions on Industrial Electronics, vol. 61, no. 8, pp. 4126-4133, 2014.

[53] W. He, A. O. David, Z. Yin, and C. Y. Sun, “Neural network control of a robotic manipulator with input deadzone and output constraint,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 6, pp. 759-770, 2016.

[54] M. Ji, G. Ferrari-Trecate, M. Egerstedt and A. Buffa, “Containment control mobile networks,” IEEE Transactions on Automatic Control, vol. 53, no. 8, pp. 1972-1975, 2008.

[55] Y. C. Cao, W. Ren and M. Egerstedt, “Distributed containment control with multiple stationary or dynamic leaders in fixed and switching directed networks,” Automatica, vol. 48, no. 8, pp. 1586-1597, 2012.

[56] Y. C. Cao, D. Stuart, W. Ren and Z. Y. Meng, “Distributed containment control for multiple autonomous vehicles with double-integrator dynamics: algorithms and experiments,” IEEE Transactions on Control Systems Technology, vol. 19, no. 4, pp. 929-938, 2011.

[57] J. Z. Li, W. Ren and S. Y. Xu, “Distributed containment control with multiple dynamic leaders for double-integrator dynamic using only position measurements,” IEEE Transactions on Automatic Control, vol. 57, no. 6, pp. 1553-1559, 2012.

[58] Z. K. Li, W. Ren, X. D. Liu and M. Y. Fu, “Distributed containment control of multi-agent systems with general linear dynamics in the presence of multiple leaders,” International Journal of Robust and Nonlinear Control, vol. 23, no. 5, pp. 534-547, 2013.

[59] Z. Y. Meng, W. Ren and Z. You, “Distributed finite-time attitude containment control for multiple rigid bodies,” Automatica, vol. 46, no. 12, pp. 2092-2099, 2010.

[60] J. Mei, W. Ren and G. F. Ma, “Distributed containment control for Lagrangian networks with parametric uncertainties under a directed graph,” Automatica, vol. 48, no. 4, pp. 653-659, 2012.

[61] J. Mei, W. Ren and G. F. Ma, “Containment control for networked unknown Lagrangian systems with multiple dynamic leaders under a directed graph,” American control conference, pp. 522-527, 2013.

[62] D. Yu, L. J. Bai and W. J. Ren, “Finite time containment control of nonlinear multi-agent

networks," Chinese control and decision conference, pp. 707-712, 2013.

[63] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," Communications of the ACM, vol. 24, no. 6, pp. 381-395, 1981.

[64] S. M. Smith and J. M. Brady, "Susan–a new approach to low level image processing," International Journal of Computer Vision, vol. 23, no. 1, pp. 45-78, 1997.

[65] E. Rosten and T. W. Drummond, "Machine learning for high-speed corner detection," European Conference on Computer Vision, vol. 3951, pp. 430-443, 2006.

[66] C. Harris and M. Stephens, "A combined corner and edge detector," Alvey Vision Conference, vol. 15, pp. 147-151, 1988.

[67] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," European Conference on Computer Vision, vol. 6314, pp. 778-792, 2010.

[68] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," European Conference on Computer Vision, vol. 3951, pp. 404-417, 2006.

[69] D. G. Lowe, "Object recognition from local scale-invariant features," Proc. of the Seventh International Conference on Computer Vision, vol. 2, pp. 1150-1157, 1999.

[70] J. Y. Bouguet, "Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm," Intel Corporation, vol. 5, no. 4, pp. 1-10, 2001.

[71] http://jacarini.dinf.usherbrooke.ca/dataset2014/.

[72] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, "A self-adjusting approach to change detection based on background word consensus," Winter Conference on Applications of Computer Vision, pp. 990-997, 2015.

[73] Sajid, Hasan, and S. S. Cheung, "Universal multimode background subtraction," IEEE Transactions on Image Processing, vol. 26, no. 7, pp. 3249-3260, 2017.

[74] S. Bianco, G. Ciocca and R. Schettini, "Combination of video change detection algorithms by genetic programming," IEEE Transactions on Evolutionary Computation, vol. 21, no. 6, pp. 914-928, 2017.

[75] M. D. Gregorio and M. Giordano, “WiSARDRP for change detection in video sequences,” European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, pp. 453-458, 2017.

[76] G. Allebosch, F. Deboeverie, P. Veelaert, and W. Philips, “EFIC: edge based foreground background segmentation and interior classification for dynamic camera viewpoints,” International Conference on Advanced Concepts for Intelligent Vision Systems, vol. 9386, pp. 130-141, 2015.

[77] H. Sajid and S. C. S. Cheung, “Background subtraction for static & moving camera,” IEEE International Conference on Image Processing, pp. 4530-4534, 2015.

[78] Y. Chen, J. Wang, and H. Lu, “Learning sharable models for robust background subtraction,” International Conference on Multimedia and Expo, pp. 1-6, 2015.

[79] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, “Subsense: a universal change detection method with local adaptive sensitivity,” IEEE Transactions on Image Processing, vol. 24, no. 1, pp. 359-373, 2015.

[80] M. Gregorio and M. Giordano, “Change detection with weightless neural networks,” International Conference on Computer Vision and Pattern Recognition, pp. 403-407, 2014.

[81] S. Varadarajan, P. Miller, and H. Zhou, “Spatial mixture of gaussians for dynamic background modelling,” International Conference on Advanced Video and Signal Based Surveillance, pp. 63-68, 2013.

[82] H. Hao, P. Barooah, and P. G. Mehta, “Stability margin scaling laws for distributed formation control as a function of network structure,” IEEE Transactions on Automatic Control, vol. 56, no. 4, pp. 923-929, 2011.

[83] H. Hao and P. Barooah, “On achieving size-independent stability margin of vehicular lattice formations with distributed control,” IEEE Transactions on Automatic Control, vol. 57, no. 10, pp. 2688-2694, 2012.

[84] A. A. Peters, R. H. Middleton, and O. Mason, “Leader tracking in homogeneous vehicle platoons with broadcast delays,” Automatica, vol. 50, no. 1, pp. 64-74, 2014.

[85] S. Knorn, A. Donaire, and J. C. Agüero, “Passivity-based control for multi-vehicle systems

subject to string constraints," Automatica, vol. 50, no. 12, pp. 3224-3230, 2014.

[86] F. Lin, M. Fardad, and M. R. Jovanovic, "Optimal control of vehicular formations with nearest neighbor interactions," IEEE Transactions on Automatic Control, vol. 57, no. 9, pp. 2203-2218, 2012.

[87] F. Lin, M. Fardad, and M. R. Jovanovic, "Algorithms for leader selection in stochastically forced consensus networks," IEEE Transactions on Automatic Control, vol. 59, no. 7, pp. 1789-1802, 2013.

[88] B. B. Ren, S. S. Ge, K. P. Tee, and T. H. Lee, "Adaptive neural control for output feedback nonlinear systems using a barrier Lyapunov function," IEEE Transactions on Neural Networks, vol. 21, no. 8, pp. 1339-1345, 2010.

[89] L. X. Wang, "Adaptive Fuzzy Systems and Control," Englewood Cliffs, NJ: Prentice-Hall, 1994.

[90] Y. Xia and J. Wang, "A general methodology for designing globally convergent optimization neural networks," IEEE Transactions on Neural Networks and Learning Systems, vol. 9, no. 6, pp. 1331-1343, 1998.

[91] E. Garone, S. D. Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey on theory and applications," Automatica, vol. 75, pp. 306-328, 2017.

[92] W. Wang, D. Wang, and Z. H. Peng, "Cooperative learning neural network output feedback control of uncertain nonlinear multi-agent systems under directed topologies," International Journal of Systems Science, vol. 48, no. 12, pp. 2590-2598, 2017.

[93] S. C. Tong, C. L. Liu and Y. M. Li, "Fuzzy-adaptive decentralized outputfeedback control for large-scale nonlinear systems with dynamical uncertainties," IEEE Transactions on Fuzzy Systems, vol. 18, no. 5, pp. 845-861, 2010.

[94] S. C. Tong and Y. M. Li, "Adaptive fuzzy output feedback tracking backstepping control of strict-feedback nonlinear systems with unknown dead zones," IEEE Transactions on Fuzzy Systems, vol. 20, no. 1, pp. 168-180, 2012.

[95] L. X. Wang and J. M. Mendel, “Fuzzy basis function, universal approximation, and orthogonal least aquare learning,” IEEE Transactions on Neural Networks, vol. 3, pp. 807-814, 1992.

[96] S. C. Tong, B. Y. Huo and Y. M. Li, “Observer-based adaptive decentralized fuzzy fault-tolerant control of nonlinear large-scale systems with actuator failures,” IEEE Transactions on Fuzzy Systems, vol. 22, no. 1, pp. 1-15, 2014.

[97] W. Wang, D. Wang and Z. H. Peng, “Cooperative fuzzy adaptive output feedback control for synchronization of nonlinear multi-agent systems under directed graphs,” International Journal of Systems Science, vol. 48, no. 12, pp. 2590-2598, 2017.

[98] Z. K. Li, Z. S. Duan and G. R. Chen, “Dynamic consensus of linear multiagent systems,” IET Control Theory and Applications, vol. 5, no. 1, pp. 19-28, 2011.