Type-2 Fuzzy Formation Control for Collision-Free Multi-Robot Systems

Yeong-Hwa Chang, Chun-Lin Chen, Wei-Shou Chan, and Hung-Wei Lin

Abstract

The main goal of this paper is to investigate the leader-following formation control of multi-robot systems. The problem of collision avoidance is considered as well. According to the graph-theoretic concepts and locally distributed information, an interval type-2 neural fuzzy formation controller is designed with the capability of on-line learning. The learning rules of controller parameters can be derived from the gradient descent method. To avoid collisions between robots, a fuzzy separation controller is proposed such that the local minimum problem can be solved. Both static and dynamic leaders are discussed for performance validations. Simulation and experimental results indicate that the proposed fuzzy formation and separation control can provide better formation responses compared to conventional consensus formation and potential-based collision-avoidance algorithms.

Keywords: Multi-robot systems, formation control, interval type-2 fuzzy control, collision avoidance.

1. Introduction

Distributed multi-agent coordination has attracted great attention in recent years, where only the locally available information from neighbors is required [1]-[5]. There are many applications of multi-agent systems, such as autonomous unmanned aerial vehicles [6], autonomous formation flight [7], congestion-controlled communication network [8], wireless sensor network [9], and autonomous multi-vehicle formations [10]. In consensus studies, graph theory is popularly used to characterize network topologies. A general consensus problem solving is to find a distributed control strategy such that the states of agents converge to a common value. Average-consensus problem was investigated for distributed networks with fixed and switching topologies [11].

Relying on graph theory, matrix theory, and control theory, the analysis of consensus protocols was thoroughly discussed. A fuzzy sliding-mode controller was proposed to investigate the formation control problem in directed graphs [12]. In [13], an impulsive control protocol was presented for multi-agent linear dynamic systems with fixed topology based on the local information of agents. A general case of leader-following consensus problems under fixed and switching topologies was discussed in [14]. In the work of [15], linear consensus protocol and saturated consensus protocol were presented for the consensus problem of heterogeneous multi-agent system. The heterogeneous multi-agent system consists of first-order and second-order integrator agents.

Fuzzy logic control, consisting of linguistic control rules, is a technique to design controllers based on human expert knowledge and experience. This technique is an effective alternative to overcome the difficulties in the requirement of exact mathematics models for plants with unexpected complex dynamics and external disturbances. Lately, neural fuzzy control, combining with the capability of fuzzy reasoning to handle uncertain information and the capability of artificial neural networks to learn from processes, has been popularly addressed. A robust fuzzy neural network control (FNN) scheme including a parameter tuning algorithm was designed for a linear Maglev rail system to achieve the objective of model-free control [16]. In the study of [17], a neural fuzzy network was proposed for the gait control of a humanoid robot. With the noisy of the measurements and the environmental variances, the parameters of the antecedent and the consequent membership functions might become uncertain, and the type-1 fuzzy logic system is not sufficient to fully and unable to directly handle these uncertainties [18], [19].

Recently, studies on type-2 fuzzy logic systems (T2FLSs) have drawn much attention. The concept of type-2 fuzzy sets (T2FSs) was introduced by Zadeh [20] as an extension of the concept of ordinary fuzzy sets. The expanded type-2 fuzzy logic system is able to handle uncertainties because it can model them and minimize their effects. The resulting T2FLSs have the
potential to provide better performance than that of T1FLSs [21], [22]. T2FLCs have been successfully applied in many applications as well as in control processes due to their ability to model uncertainties. In [21], the problem of wall following control of wheeled mobile robots was discussed, where the T2FLC was adopted with a reinforcement learning. However, the T2FLCs are pointed out to have heavy computational load [23]. To simplify the computation complexity, the grades of the secondary membership functions can be set to 1. If the secondary membership functions are set to be 1, the T2FLSs become the so-called interval T2FLSs (IT2FLCs). Fuzzy neural networks (FNNs), combining the interval type-2 fuzzy neural network (IT2FNN) is superior to the type-1 FNN (T1FNN) in the control of complicated and nonlinear systems [24].

Collision avoidance is an interesting topic addressed in multi-robot networks. A cooperative control laws was proposed for general nonlinear dynamic models to guarantees collision-free conflict resolution [25]. In [26], a fuzzy logic was designed for potential functions to achieve collision avoidance with input constrains. In the work of [27], a flocking algorithm was presented for separation forces generated to avoid collisions with external obstacles. A modified avoidance function was proposed for nonlinear Lagrange systems to achieve collision avoidance with bounded disturbances [28]. In [29], a potential field method was discussed for mobile robots to solve the local minimum problem. In addition, Wang et al. [30] presented a fuzzy potential force for the separating potential function in flocking control. However, only few of existing results have been presented to solve the problem of local minima in multi-robot systems.

This paper aims to investigate the formation control of leader-follower multi-robot systems, where the problem of collision avoidance is also considered. The graph theory is used to model the communication topology between robots. To improve the control performance, a novel formation algorithm, neural-fuzzy formation controller, is proposed for multi-robot systems in directed graphs. The interval type-2 neural-fuzzy control (IT2FNC) parameter consist input Gaussian membership functions and output fuzzy singleton, where the parameters of input and output membership functions can be adaptively adjusted. The proposed interval type-2 neural-fuzzy formation controller has the capability of on-line learning, and the adaptive rules can be derived using the gradient descent method. Moreover, a hybrid fuzzy based separation control is proposed for collision avoidance and the local minimum problem of traditional potential-based separation control can be solved. The fuzzy based separation control consists of triangular input membership functions and singleton output membership functions, where the control output provides an alternative moving direction for robots to achieve collision-free tasks. Numerical simulations are provided to validate the collision-free formation responses.

This paper is organized as follows. In Sec. 2, some essential graph-theoretic concepts and a network of kinematic model are introduced. In Sec. 3, the framework of an interval type-2 neural-fuzzy formation controller is investigated, where the updating rules for controller parameters are derived. In Sec. 4, the conventional potential-based collision avoidance is introduced. Moreover, a hybrid fuzzy-oriented separation control is presented. In Sec. 5, simulation and experimental results are provided for performance validations. Some concluding remarks are given in Sec. 6.

2. Preliminaries

A. Graph theory

Considering a multi-robot system of $n$ robots, let $G = (V, \Xi)$ be a directed graph (digraph), consisting of a vertex set $V = \{v_1, v_2, \ldots, v_n\}$ and an edge set $\Xi \subseteq V \times V$. The vertexes $v_i$ and $v_j$ represent the $i$th and $j$th robots, respectively. In digraphs, an edge of $G$ is an ordered pair of distinct nodes $(v_j, v_i) \in \Xi$, in which $v_i$ and $v_j$ are the head and tail of the edge, respectively [31]. The weighted adjacency matrix of a digraph $G$ is denoted as

$$A = \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1} & a_{n2} & \cdots & a_{nn}
\end{bmatrix} \in \mathbb{R}^{n \times n},$$

where $a_{ij}$ is the link weight, $a_{ij}$, if $(v_j, v_i) \in \Xi$, and $a_{ij} = 0$, if $(v_j, v_i) \notin \Xi$.

In this paper, a leader-follower problem will be dealt with, where the multi-robot system consists of $n$ robots, one leader and $n-1$ followers. In notations, the robots indexed by $1, 2, \ldots, n-1$ are followers and the item $n$ is the leader. Assume that the leader robot has only transmitting capability, i.e. the leader acquire no information from followers $a_{nj} = 0$, $j = 1, \ldots, n$. In this case, let the topology relationship of follower robots be denoted as $\mathcal{G}$, a subgraph of $G$. Then, the associated adjacency matrix of $\mathcal{G}$ is represented as

$$\mathbf{\tilde{A}} = \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1(n-1)} \\
a_{21} & a_{22} & \cdots & a_{2(n-1)} \\
\vdots & \vdots & \ddots & \vdots \\
a_{(n-1)1} & a_{(n-1)2} & \cdots & a_{(n-1)(n-1)}
\end{bmatrix} \in \mathbb{R}^{(n-1) \times (n-1)}.$$
Consequently, the connection relationship between the leader and followers can be described as \( B = \text{diag}[\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_{n-1}] \), where \( \bar{b}_i = a_{i,n}, \ i = 1,2,\ldots,n-1 \).

B. Kinematic model of multi-robot system

The scheme diagram of a wheeled robot is shown in Figure 1, where the kinematic equations for the \( i \)th robot are described as

\[
\begin{align*}
\dot{x}_i(t) &= v_i(t) \cos(\theta_i(t)), \\
\dot{y}_i(t) &= v_i(t) \sin(\theta_i(t)), \\
\dot{\theta}_i(t) &= \omega_i(t),
\end{align*}
\]

where \((x_i(t), y_i(t))\) is the center position, \(v_i(t)\) is the linear velocity, \(\omega_i(t)\) is the angular velocity, \(\theta_i(t)\) is the rotation angle, \(i = 1,2,\ldots,n\). It is denoted \((x_{hi}(t), y_{hi}(t))\) as the head position of the \( i \)th robot, such that

\[
\begin{align*}
x_{hi}(t) &= x_i(t) + l_h \cos(\theta_i(t)), \\
y_{hi}(t) &= y_i(t) + l_h \sin(\theta_i(t)),
\end{align*}
\]

where \(l_h\) is the length between the center position and the head position. Taking the derivative of (4), the governing equations of the head position for the \( i \)th robot can be determined as follows

\[
\begin{align*}
\dot{x}_{hi}(t) &= \cos(\theta_i(t)) \dot{x}_i(t) - l_h \sin(\theta_i(t)) \dot{\theta}_i(t), \\
\dot{y}_{hi}(t) &= \sin(\theta_i(t)) \dot{x}_i(t) + l_h \cos(\theta_i(t)) \dot{\theta}_i(t).
\end{align*}
\]

According to the configuration in Figure 1, the motion equations of \(v_i(t)\) and \(\omega_i(t)\) corresponding to the left- and right-wheel angular velocities, \(\omega_{l_i}(t)\) and \(\omega_{r_i}(t)\), can be obtained as

\[
\begin{align*}
\begin{bmatrix} v_i(t) \\ \omega_i(t) \end{bmatrix} &= \begin{bmatrix} 0.5r \\ -0.5r \end{bmatrix} \begin{bmatrix} \omega_{l_i}(t) \\ \omega_{r_i}(t) \end{bmatrix},
\end{align*}
\]

where \(r\) is the radius of wheel, and \(l_{hi}\) is the distance between the center of robot and the wheel. Substituting (6) into (5) and defining \(l_{ri} = l_{hi}/(2r_{hi})\), it leads to the follow equivalent model of a wheeled robot

\[
\begin{align*}
\begin{bmatrix} \dot{x}_{hi}(t) \\ \dot{y}_{hi}(t) \end{bmatrix} &= \begin{bmatrix} T_{11}(t) & T_{12}(t) \\ T_{21}(t) & T_{22}(t) \end{bmatrix} \begin{bmatrix} \omega_{l_i}(t) \\ \omega_{r_i}(t) \end{bmatrix},
\end{align*}
\]

where

\[
\begin{align*}
T_{11}(t) &= 0.5r \cos(\theta_i(t)) + r_{hi} \sin(\theta_i(t)), \\
T_{12}(t) &= 0.5r \cos(\theta_i(t)) - r_{hi} \sin(\theta_i(t)), \\
T_{21}(t) &= 0.5r \sin(\theta_i(t)) - r_{hi} \cos(\theta_i(t)), \\
T_{22}(t) &= 0.5r \sin(\theta_i(t)) + r_{hi} \cos(\theta_i(t)).
\end{align*}
\]

Let

\[
\begin{align*}
\begin{bmatrix} u_i'(t) \\ y_i'(t) \end{bmatrix} &= \begin{bmatrix} T_{11}(t) & T_{12}(t) \\ T_{21}(t) & T_{22}(t) \end{bmatrix} \begin{bmatrix} \omega_{l_i}(t) \\ \omega_{r_i}(t) \end{bmatrix},
\end{align*}
\]

where \(u_i'(t)\) and \(y_i'(t)\) are considered as the control actions to the head position. Therefore, the kinematic equations of a wheeled robot can be equivalently as follows,

\[
\begin{align*}
\begin{bmatrix} \dot{x}_{hi}(t) \\ \dot{y}_{hi}(t) \end{bmatrix} &= \begin{bmatrix} u_i'(t) \\ y_i'(t) \end{bmatrix}.
\end{align*}
\]

C. Definition of avoidance region

In this paper, all robots are assumed to have the same sensing capability. In addition, each robot is not allowed to collide with other robots during the formation process. The geometric relationship between robots is shown in Figure 2, where \(R_s\) is the sensing radius, and \(r_j = \left[ x_j(t), y_j(t) \right]^T \in \mathbb{R}^2\) is the position vector of the \( j \)th robot. The robot \( j \) is a neighbor of the \( i \)th robot if the Euclidean distance between two robots is less or equal to the sensing radius, \(d_j = \| r_i - r_j \| \leq R_s\). Let \(N_{ij}\) stand for the neighbor set of the \( i \)th robot. Once the \( j \)th robot moves into the sensing radius of the \( i \)th robot, the collision avoidance mechanism starts to work. In Figure 2, the notation \(r_{hi} = l_{hi}\) denotes the avoidance radius of which the minimum distance allowed between two robot is \(2r_{hi}\). In practice, it is reasonable to assume that \(R_s \geq 2r_{hi}\).

3. Interval Type-2 Neural Fuzzy Formation Control

A. Structure of interval type-2 neural fuzzy controller

In this section, an IT2NFC is proposed to deal with the leader-following formation problem, where the equivalent model of (10) is considered. Let the \(x\) - and \(y\) -axis error functions be respectively defined as

\[
\begin{align*}
e_i' &= \sum_{j=1}^{n} a_{ij} [(x_{hi}(t) - p_{ij}' - (x_{hi}(t) - p_{ij}')] + \tilde{b}_i [(x_{hi}(t) - p_{ij}' - (x_{hi}(t) - p_{ij})],
\end{align*}
\]

where...
follower are designated as follows
\[ z_{1i} = c_1 e_i \quad (13) \]
\[ z_{12} = c_2 \varepsilon_i \quad (14) \]
\[ u_i^I = c_3 h_i^5 \quad (15) \]

where \( c_1 \), \( c_2 \) and \( c_3 \) are positive constants, \( i = 1, 2, \ldots, n-1 \).

The fuzzy rules are given in Table 1, where the input and output spaces are fuzzily partitioned into five fuzzy sets, Negative Big (NB), Negative Small (NS), Zero (Z), Positive Small (PS), and Positive Big (PB). The input and output membership functions are depicted in Figure 3. The corresponding if-then fuzzy rules for the \( i \)th agent are expressed as
\[ R_i^k: \text{IF} \ z_{1i}^{(1)} \cdot \text{IS} \cdot \tilde{F}_{i1,k} \cdot \text{AND} \cdot z_{12}^{(1)} \cdot \text{IS} \cdot \tilde{F}_{i2,k} \]
\[ \text{THEN} \cdot h_i^5 \cdot \text{IS} \left[ W_i^{(4)} \right] \left[ W_i^{(4)} \right] \]

Figure 2. Geometric relationship between robots: (a) robot \( j \) is not a neighbor agent of robot \( i \), (b) robot \( j \) is a neighbor agent of robot \( i \).

In this study, the leader is maneuvered along a pre-specified trajectory, and the design of formation controller is focused on followers. The control signals of each follower are represented as
\[ u_i^I = \begin{bmatrix} u_{i1}^I & u_{i2}^I \end{bmatrix}^T \]

Moreover, from the decoupled characteristic of (10), it suffices to address only one subsystem, i.e. \( u_i^I \) or \( u_i^I \) of the \( i \)th follower. In the following, all superscript symbols of \( x \) and \( y \) will be omitted for conciseness.

The network structure of IT2NFC is shown in Figure 3, of which the control inputs and output of the \( i \)th follower are expressed as
\[ e_i^I(t) = \sum_{j \neq i} a_{ij} \left[ (y_{ij}(t) - p_j^I) - (y_{ji}(t) - p_i^I) \right] \]
\[ + b_i \left[ (y_{ii}(t) - p_i^I) - (y_{ii}(t) - p_i^I) \right], \]

where \( p_i^I \), \( p_j^I \), \( p_j^I \) and \( p_i^I \) are coordinate positions regarding to a desired formation pattern in \( x \)- and \( y \)-axis. It is noticed that \( a_{ij} = 1 \) means that the \( j \)th robot can send position information to the \( i \)th robot, and \( b_i = 1 \) means that the \( i \)th robot can receive position information from the leader.

Figure 3. Structure of IT2NFC.

### Table 1. IT2NFC rule base for formation control.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>( z_{1i}^{(1)} )</th>
<th>( z_{12}^{(1)} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z_{1i}^{(1)} )</td>
<td>NB</td>
<td>NS</td>
</tr>
<tr>
<td>NB</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>NS</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>Z</td>
<td>PB</td>
<td>PS</td>
</tr>
<tr>
<td>PS</td>
<td>PS</td>
<td>Z</td>
</tr>
<tr>
<td>PB</td>
<td>Z</td>
<td>NS</td>
</tr>
</tbody>
</table>

where \( i = 1, 2, \ldots, n-1 \) is the robot number and \( k = 1, 2, \ldots, n \) is the rule number, \( z_{1i}^{(1)} \) and \( z_{12}^{(1)} \) are the inputs of IT2NFC, \( \tilde{F}_{i1,k} \) and \( \tilde{F}_{i2,k} \) are the antecedent
fuzzy sets, and \([w_{ijd}^{(4)}, w_{ijr}^{(4)}]\) is a weighting interval set of the consequent part. The function of each IT2NFC layer is introduced as follows:

1) Input layer:
Choose \(z_{i}^{(1)}\) and \(z_{i}^{(2)}\) as (13) and (14), respectively.

2) Membership layer:
In this layer, each node performs as a Gaussian interval type-2 fuzzy membership function. The input membership functions are depicted in Figure 4. For the \(s\)th input, \(k\)th node, and \(i\)th robot, the associated Gaussian membership functions can be represented as

\[
\mu_{E_{i,k}}^{(2)}(z_{i}^{(1)}) = \exp \left[ -\frac{1}{2} \left( \frac{z_{i}^{(1)} - m_{i,k}^{(1)}}{\sigma_{i,k}^{(1)}} \right)^2 \right]
\]

where \(m_{i,k}^{(1)}\) and \(\sigma_{i,k}^{(1)}\) are given in ascending order, i.e.,

\[
w_{ijd}^{(4)}(w_{ijr}^{(4)}) \leq w_{ijd}^{(4)}(w_{ijr}^{(4)}) \leq \cdots \leq w_{ijd}^{(4)}(w_{ijr}^{(4)})
\]

The values of \(h_{li}^{(4)}\) and \(h_{ri}^{(4)}\) can be derived as

\[
h_{li}^{(4)} = \frac{\sum_{k=1}^{n} f_{ik}^{(3)} \overline{w}_{ik}^{(4)}}{\sum_{k=1}^{n} f_{ik}^{(3)}}
\]

(22)
5) Output layer

The output $h_i^{(3)}$ is determined as the average of $h_i^{(4)}$ and $h_{ir}^{(4)}$, i.e.

$$h_i^{(5)} = \frac{h_i^{(4)} + h_{ir}^{(4)}}{2}. \quad (26)$$

B. **Parameter-learning algorithms**

In this section, the method of gradient descent will be adopted to derive on-line learning algorithms to update IT2NFC parameters. First, an energy function $E_i$ is defined as [36]

$$E_i = \frac{1}{2}(e_i)^2, \quad (27)$$

where $i = 1,2,\ldots,n-1$. Then, the update laws of layer parameters are derived in the following:

1) Output layer:

The error term to be propagated is computed as

$$\delta_i^{(3)} = -\frac{\partial E_i}{\partial h_i^{(5)}} = -\frac{\delta h_i^{(3)}}{\delta e_i} \cdot \frac{\partial h_i^{(5)}}{\partial h_i^{(3)}}. \quad (28)$$

It is noticed that $\frac{\partial e_i}{\partial h_i^{(5)}}$ cannot be analytically determined. To overcome this problem, the adaptive law proposed in [37] is adopted,

$$\delta_i^{(3)} = e_i. \quad (29)$$

2) Type-reduction layer:

Form (26) and (29), the error terms to be propagated is computed as

$$\delta_i^{(4)} = \begin{cases} \frac{-\partial E_i}{\partial h_i^{(4)}} = -\frac{\partial E_i}{\partial h_i^{(5)}} \cdot \frac{\partial h_i^{(4)}}{\partial h_i^{(5)}} = \frac{1}{2} \delta_i^{(3)} & \text{if } \frac{\partial h_i^{(5)}}{\partial h_i^{(3)}} > 0 \\ \frac{-\partial E_i}{\partial h_i^{(4)}} = -\frac{\partial E_i}{\partial h_i^{(5)}} \cdot \frac{\partial h_i^{(4)}}{\partial h_i^{(5)}} = \frac{1}{2} \delta_i^{(3)} & \text{otherwise} \end{cases}. \quad (30)$$

It can be obtained that

$$\frac{-\partial E_i}{\partial w_{ik}^{(4)}} = \frac{-\partial E_i}{\partial h_i^{(5)}} \cdot \frac{\partial h_i^{(4)}}{\partial w_{ik}^{(4)}} = \delta_i^{(4)} \frac{f_k^{(3)}}{\sum_{k=1}^n f_k^{(3)}}, \quad (31)$$

$$\frac{-\partial E_i}{\partial w_{il}^{(4)}} = \frac{-\partial E_i}{\partial h_i^{(5)}} \cdot \frac{\partial h_i^{(4)}}{\partial w_{il}^{(4)}} = \delta_i^{(4)} \frac{f_k^{(3)}}{\sum_{k=1}^n f_k^{(3)}} \quad (32)$$

The weighting interval can be updated according to the following adaptive rules

$$w_{ik}^{(4)}(t+1) = w_{ik}^{(4)}(t) - \eta_{w_{ik}} \cdot \frac{\partial E_i}{\partial w_{ik}^{(4)}},$$

$$w_{il}^{(4)}(t+1) = w_{il}^{(4)}(t) - \eta_{w_{il}} \cdot \frac{\partial E_i}{\partial w_{il}^{(4)}}, \quad (33)$$

where $\eta_{w_{ik}} > 0$ and $\eta_{w_{il}} > 0$ are learning rates.

3) Rule layer:

Form (24), (25) and (30), the error term is computed as follows

$$\delta_i^{(3)} = \begin{cases} \frac{-\partial E_i}{\partial h_i^{(3)}} = \delta_i^{(4)} \frac{\partial h_i^{(4)}}{\partial h_i^{(3)}} = \delta_i^{(4)} \left( \frac{w_{ik}^{(4)} - h_{irk}^{(4)}}{2} \right) = \delta_i^{(4)} & \text{if } \frac{\partial h_i^{(5)}}{\partial h_i^{(3)}} > 0 \\ \frac{-\partial E_i}{\partial h_i^{(3)}} = \delta_i^{(4)} \frac{\partial h_i^{(4)}}{\partial h_i^{(3)}} = \delta_i^{(4)} \left( \frac{w_{il}^{(4)} - h_{iir}^{(4)}}{2} \right) = \delta_i^{(4)} & \text{otherwise} \end{cases}. \quad (34)$$

4) Membership layer:

Form (20) and (34), the error term is computed as follows

$$\delta_i^{(2)} = \begin{cases} \frac{-\partial E_i}{\partial \mu_{ik}^{(2)}} = \delta_i^{(3)} \frac{f_k^{(3)}}{\sum_{k=1}^n f_k^{(3)}} + \delta_i^{(2)} \frac{f_k^{(3)}}{\sum_{k=1}^n f_k^{(3)}} = \delta_i^{(2)} & \text{if } \frac{\partial \mu_{ik}^{(2)}}{\partial \mu_{ik}^{(3)}} > 0 \\ \frac{-\partial E_i}{\partial \mu_{ik}^{(2)}} = \delta_i^{(3)} \frac{f_k^{(3)}}{\sum_{k=1}^n f_k^{(3)}} + \delta_i^{(2)} \frac{f_k^{(3)}}{\sum_{k=1}^n f_k^{(3)}} = \delta_i^{(2)} & \text{otherwise} \end{cases}. \quad (35)$$

where (36) and (37) are calculated as follows

$$\frac{\partial \mu_{ik}^{(2)}(z_i^{(1)})}{\partial m_{ik,1}} = \begin{cases} 0, & \text{if } (m_{ik,1} + m_{ik,2}) < z_i^{(1)} \\ \frac{2}{m_{ik,1}}, & \text{otherwise} \end{cases}, \quad (36)$$

$$\frac{\partial \mu_{ik}^{(2)}(z_i^{(1)})}{\partial m_{ik,2}} = \begin{cases} 0, & \text{if } (m_{ik,1} + m_{ik,2}) > z_i^{(1)} \\ \frac{2}{m_{ik,2}}, & \text{otherwise} \end{cases}, \quad (39)$$

$$\frac{\partial \mu_{ik}^{(2)}(z_i^{(1)})}{\partial z_i^{(1)}} = \begin{cases} H_1, & \text{if } z_i^{(1)} < m_{ik,1} \\ 0, & \text{otherwise} \end{cases}, \quad (40)$$

$$\frac{\partial \mu_{ik}^{(2)}(z_i^{(1)})}{\partial z_i^{(1)}} = \begin{cases} H_2, & \text{if } z_i^{(1)} > m_{ik,2} \\ 0, & \text{otherwise} \end{cases}, \quad (41)$$

$$\frac{\partial \mu_{ik}^{(2)}(z_i^{(1)})}{\partial z_i^{(1)}} = \begin{cases} H_3, & \text{if } z_i^{(1)} < m_{ik,1} \\ 0, & \text{otherwise} \end{cases}, \quad (42)$$

$$\frac{\partial \mu_{ik}^{(2)}(z_i^{(1)})}{\partial z_i^{(1)}} = \begin{cases} H_4, & \text{if } z_i^{(1)} > m_{ik,2} \\ 0, & \text{otherwise} \end{cases}, \quad (43)$$
where

\[ H_i = \frac{(z_{i\mu} - m_{i,\mu})N(m_{i,\mu}, \sigma_{i,\mu}, z_{i\mu})}{(\sigma_{i,\mu})^2} \]

(44)

\[ H_2 = \frac{(z_{i\mu} - m_{i,\mu})N(m_{i,\mu+1}, \sigma_{i,\mu}, z_{i\mu})}{(\sigma_{i,\mu})^2} \]

(45)

\[ H_3 = \frac{(z_{i\mu} - m_{i,\mu})N(m_{i,\mu+2}, \sigma_{i,\mu}, z_{i\mu})}{(\sigma_{i,\mu})^2} \]

(46)

\[ H_4 = \frac{(z_{i\mu} - m_{i,\mu})N(m_{i,\mu+2}, \sigma_{i,\mu}, z_{i\mu})}{(\sigma_{i,\mu})^2} \]

(47)

Finally, the means and standard deviations of the membership functions are updated as follows

\[ m_{i,\mu}(t+1) = m_{i,\mu}(t) - \eta_{ma} \frac{\partial E_i}{\partial m_{i,\mu}}, q = 1, 2 \]

(48)

\[ \sigma_{i,\mu}(t+1) = \sigma_{i,\mu}(t) - \eta_{ma} \frac{\partial E_i}{\partial \sigma_{i,\mu}}, \]

(49)

where \( \eta_{ma} > 0 \) and \( \eta_{ma} > 0 \) are learning-rate parameters of the means and standard deviations, respectively.

### 4. Separation Control for Collision Avoidance

#### A. Potential-based separation control

Consider \( g_{ij} \) be the potential force between the \( i \)-th and \( j \)-th robots. Then the integrated separation force from all its neighboring robots can be formulated as

\[ u_i = -\sum_{j \in N_i} h_{ij}(r_i, r_j). \]

(50)

To derive a proper separation force between two connected robots, a smooth potential function \( V_{ij} \) is considered,

\[ V_{ij}(d_{ij}) = \min \left\{ 0, \frac{d_{ij}^2 - R_i^2}{d_{ij}^2 - R_i^2} \right\}^2, \]

(51)

such that

\[ h_{ij} = \nabla V_{ij}. \]

(52)

In a two-dimensional case, \( r_i = [x_i, y_i]^T \), the gradient computations of \( V_{ij} \) can be obtained as follows,

\[ \frac{\partial V_{ij}}{\partial x_i} = \begin{cases} 0, & \text{if } R_i < d_{ij}, \\ V_{ij}(x_i - x_j), & \text{if } 2r_i \leq d_{ij} \leq R_i \\ V_{max}, & \text{if } d_{ij} < 2r_i, \end{cases} \]

(53)

\[ \frac{\partial V_{ij}}{\partial y_i} = \begin{cases} 0, & \text{if } R_i < d_{ij}, \\ V_{ij}(y_i - y_j), & \text{if } 2r_i \leq d_{ij} \leq R_i \\ V_{max}, & \text{if } d_{ij} < 2r_i, \end{cases} \]

(54)

\[ V_{max} = \frac{4(R_i^2 - r_i^2)(d_{ij}^2 - R_i^2)}{(d_{ij}^2 - R_i^2)^3}, \]

(55)

where \( V_{max} \) is the maximum allowable separation force. Integrating the formation and separation forces, the net control action to an robot can be obtained as

\[ u_i = u_i^f + u_i^s. \]

(56)

#### B. Fuzzy separation control (FSC)

In multi-robot systems, the popularly used artificial potential function generates the repulsive force, which is inversely proportional to the distance. It cannot constrain the magnitude of control inputs as the magnitude of control inputs becomes very large when the distance between neighbor robots becomes small. To overcome this problem, a fuzzy logic based approach is used here to construct the potential force function. The reason for using fuzzy logic is due to the fact that fuzzy logic can generate a smooth nonlinear function with the flexibility of changing its outputs by adjusting fuzzy membership functions and fuzzy logic rules. The fuzzy based potential force from all its neighboring robots can be formulated as

\[ u_i^{\mu/\nu} = -\sum_{j \in N_i} h_{ij}^{\mu/\nu}(r_i, r_j). \]

(57)

where \( h_{ij}^{\mu/\nu} \) is the \( i \)-th robot fuzzy based potential function output form all its neighboring robots. Let the fuzzy inputs of the \( i \)-th robot be

\[ z_{ij}^{\mu/\nu} = (d_{ij})^2 - (R_i)^2, \]

(58)

The fuzzy rules are given in Table 2, where the input \( z_{ij}^{\mu/\nu} \) and output \( h_{ij}^{\mu/\nu} \) spaces are fuzzily partitioned into five fuzzy sets, Large (L), Big (B), Medium (M), Small (S), Zero (Z). The input and output membership functions are depicted in Figure 5. The corresponding if-then fuzzy rules for \( i \)-th robot are expressed as

\[ R_k^{\mu/\nu} : \text{IF} \; z_{ij}^{\mu/\nu} \; \text{is} \; M_k^{\mu/\nu} \; \text{THEN} \; h_{ij}^{\mu/\nu} \; \text{is} \; G_k^{\mu/\nu}, \]

(59)

where \( M_k^{\mu/\nu} \) and \( G_k^{\mu/\nu} \) are the fuzzy sets of antecedent and consequence parts, respectively, \( k = 1, 2, \ldots, 5 \). In Figure 5, the Gaussian function given below is adopted as the membership function:

\[ \mu_{u_{ij}} = \exp \left\{ -\frac{1}{2} \left( \frac{z_{ij}^{\mu/\nu} - m_k^{\mu/\nu}}{\sigma_k^{\mu/\nu}} \right)^2 \right\} \]

(60)

\[ \sigma_k^{\mu/\nu} = \left[ \begin{array}{cccc} \sigma_1^{\mu/\nu} & \sigma_1^{\mu/\nu} & \sigma_2^{\mu/\nu} & \sigma_3^{\mu/\nu} \end{array} \right]^T \]

(61)
\[ m^k_i = \begin{bmatrix} m_i^{k_1} & m_i^{k_2} & m_i^{k_3} & m_i^{k_4} & m_i^{k_5} \end{bmatrix} = \begin{bmatrix} -(R_i^2 - 2r_f^2) \\ -(R_i^2 - 2r_f^2) \\ -(R_i^2 - 2r_f^2) \\ 0 \end{bmatrix} \] (62)

Table 2. Fuzzy potential function rule base.

<table>
<thead>
<tr>
<th>( h^p_i )</th>
<th>Z</th>
<th>S</th>
<th>M</th>
<th>B</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>( z^p_i )</td>
<td>Z</td>
<td>S</td>
<td>M</td>
<td>B</td>
<td>L</td>
</tr>
</tbody>
</table>

In (60), \( m_i^k \) and \( \sigma_k^p \) are means and standard deviations, respectively, \( k = 1, 2, \ldots, 5 \). By using the centroid defuzzification technique, the fuzzy based potential function output can be calculated as follows:

\[ h^p_i = \frac{\sum_{k=1}^{5} \mu_{u^p_k} g_k^p}{\sum_{k=1}^{5} \mu_{u^p_k}}, \] (63)

where \( g_k^p \) is the value corresponding to singleton outputs,

\[ g_k^p = [0 \ 1 \ 4 \ 8 \ 20]^T. \] (64)

In Figure 6, the case of robot \( i \) with two neighbor robots is considered, where \( (x_{i1}, y_{i1}) \) and \( (x_{i2}, y_{i2}) \) are neighboring robots, and the related target is located at \( (x_i', y_i') \). The notation \( u_i' \) is denoted as the attractive force to the target, and \( h_i^p \) and \( h_i^s \) are separation forces corresponding to neighboring robots. Then, the integrated separation force to the \( i \)th robot is the vector sum of \( h_i^p \) and \( h_i^s \), where \( \phi_i \) is the related direction of \( u_i' \) to the \( x \)-axis. In addition, \( u_i \) is the net force of the \( i \)th robot, and \( \alpha_i \) is angle between the robot and its target. In case \( \phi_i = \alpha_i \), the direction of \( u_i^s, i \) is opposite to the attractive action \( u_i' \). Moreover, if the magnitude of \( u_i' \) is less or equal to the magnitude of \( u_i^s, i \), the \( i \)th agent will be stuck in the local minimum.

To solve this problem, a fuzzy separation control method will be presented, and the key idea is depicted in Figure 7, where an extra angle \( \phi_i^{pec} \) is added to the original separation force, \( \phi_i^{new} = \phi_i + \phi_i^{pec} \). In Figure 7, \( u_i^{s, pec} \) is the modified separation force, of which the magnitude keeps unchanged but the direction is changed because of \( \phi_i^{pec} \). Consequently, the integrated net force of attractive force and separation force can be represented as

\[ u_i = u_i' + u_i^{s, pec} (u_i^p, \phi_i^{pec}). \] (65)
For those follower robots, communication-connected to the leader, their respective targets can be obtained according to the leader position and designated formation pattern, however, a substitute solution is required for other followers. Alternatively, from (11) and (12), targets can be equivalently viewed as the propagated errors from other followers,

$$x'_i = x_i - e'_i, \quad (66)$$
$$y'_i = y_i - e'_i, \quad (67)$$

where follower $i$ is not communicated to the leader, $b_i = 0, i = 1, 2, \ldots, n - 1$. The design of fuzzy separation controller will be illustrated in the following. First, let the fuzzy inputs of the $i$th robot be

$$z_{i}^{\text{fisc.A}} = \phi_i - \alpha_i, \quad (68)$$
$$z_{i}^{\text{fisc.D}} = ||r_i - r_j||, \quad (69)$$

where $r_j = [x_j \ y_j]^T \in R^2$ is the position of the center gravity of neighboring robots,

$$x_j = \sum_{i \in N_i} \frac{x_i}{|N_i|}, \quad (70)$$
$$y_j = \sum_{i \in N_i} \frac{y_i}{|N_i|}, \quad (71)$$

where $|\cdot|$ is the cardinality of a set, i.e. $|N_i|$ is the number of neighboring robots of the $i$th robot.

The fuzzy rules are given in Table 3, where the input and output spaces are fuzzily partitioned into ten fuzzy sets, Negative Big (NB), Negative Small (NS), Zero (ZO), Positive Small (PS), Positive Big (PB), Very Close (VC), Close (C), Medium (M), Far (F), and Very Far (VF). The input and output membership functions are depicted in Figures 8-9, respectively. The corresponding if-then fuzzy rules for the $i$th robot are expressed as

$$R_{ik}^{\text{fisc.A}} \cdot \text{IF} \cdot z_{i}^{\text{fisc.A}} \cdot \text{is} \cdot M_{i}^{\text{fisc.A, AND} \cdot z_{i}^{\text{fisc.D}} \cdot \text{is} \cdot M_{k}^{\text{fisc.D}} \text{THEN} \cdot \phi_{i}^{\text{fisc}} \cdot \text{is} \cdot G_{ik}^{\text{fisc}} \quad (72)$$

where $M_{i}^{\text{fisc.A}}$ and $M_{k}^{\text{fisc.D}}$ are the fuzzy sets of the antecedent part, and $G_{ik}^{\text{fisc}}$ is the fuzzy set of the consequent part, $i = 1, 2, \ldots, n - 1$, and $k = 1, 2, \ldots, 5$.

By using the centroid defuzzification technique, the defuzzified fuzzy output is calculated as

$$\phi_{i}^{\text{fisc}} = \frac{\sum_{i=1}^{5} \sum_{k=1}^{5} \mu_{ik} \cdot G_{ik}^{\text{fisc}}}{\sum_{i=1}^{5} \sum_{k=1}^{5} \mu_{ik}}, \quad (73)$$

where $G_{ik}^{\text{fisc}} \in \{G_{ik}^{\text{fisc}} | l, k = 1, 2, \ldots, 5\}$ and a min-max operation is performed over all rules mapping to the same output fuzzy set.

$$b_{ik}^{\text{fisc}} \in \{G_{ik}^{\text{fisc}} | l, k = 1, 2, \ldots, 5\}$$
$$\forall_{l} \in \{l, k = 1, 2, \ldots, 5\}$$

$$= \max_{k \in G_{ik}^{\text{fisc}}} \left\{ \min\left(\mu_{il}^{\text{fisc}} \cdot \mu_{lk}^{\text{fisc}}\right) \right\}.$$

Table 3. Fuzzy rule base.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>$z_{i}^{\text{fisc.A}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
</tr>
<tr>
<td>VC</td>
<td>PS</td>
</tr>
<tr>
<td>C</td>
<td>PS</td>
</tr>
<tr>
<td>M</td>
<td>PS</td>
</tr>
<tr>
<td>F</td>
<td>Z</td>
</tr>
<tr>
<td>VF</td>
<td>Z</td>
</tr>
</tbody>
</table>

Figure 8. The input membership functions.

Figure 9. The output membership functions.

5. Simulation and Experimental Results

In the following, all the robots are assumed to be homogeneous with the same specifications, $r_a = 0.05$ (m), $R_1 = 0.2$ (m), $r = 0.029$ (m) and $l_a = 0.034$ (m). To verify the feasibility of proposed interval type-2 neural-fuzzy formation controller and fuzzy separation controller, both the collision avoidance and formation preservation are considered. In the leader-follower formation control, the case of five robots, one leader and four followers, is considered. The communication topology is shown in Figure 10, where the circles labelled 1 to 4 denote the follower robots and the circle...
\( L \) represents the leader robot. From (2), the information exchanges between leader and followers can be modelled as

\[
\begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}. \tag{74}
\]

Figure 10. Communication topology of multirobot systems.

The initial position and orientation of the leader is \((0.55, 0, 0)\), and the followers are initially placed at:

\((-0.335, 0.335, 135\degree), (-0.335, 0.335, 45\degree), (-0.335, -0.335, 45\degree),\) and \((0.335, -0.335, 135\degree)\).

The formation pattern is designated as:

\[(p_{x1}, p_{y1}) = (-0.85, -0.25),\]
\[(p_{x2}, p_{y2}) = (-0.35, -0.25),\]
\[(p_{x3}, p_{y3}) = (-0.35, 0.25),\]
\[(p_{x4}, p_{y4}) = (-0.85, 0.25),\]

and \((p_{x5}, p_{y5}) = (0, 0)\) (m).

According to the initial positions and the designated formation of robots, followers are required to exchange positions, i.e. follower 1 versus follower 3, and follower 2 versus follower 4. The leader keeps standstill until \(30\) (sec). After that, the leader moves along a sinusoidal trajectory at \(t = 30\) s. In addition, there is one cylinder of radius 0.08 (m), placed at the position \((1.18, 0.12)\) (m). This cylinder can be viewed as a standstill obstacle. The parameters of IT2NFC are originally chosen as \(c_1 = 1, c_2 = 1, c_3 = 1, \sigma_{
u,k} = 0.15, \eta_{\sigma} = \eta_{\omega} = 0.001, \eta_{ul} = \eta_{wr} = 0.003,\) and

\[
m_{ij,k1}, w_{ij,k1}^{(4)} = \begin{cases}
-1.05, & j = 1 - 5, \\
-0.55, & j = 6 - 10, \\
-0.05, & j = 11 - 15, \\
0.45, & j = 16 - 20, \\
0.95, & j = 21 - 25, \\
-0.95, & j = 1 - 5, \\
-0.45, & j = 6 - 10, \\
0.05, & j = 11 - 15, \\
0.55, & j = 16 - 20, \\
1.05, & j = 21 - 25.
\end{cases}
\]

For performance comparisons, the conventional consensus algorithm (CA) [38] and the potential-based separation control (PSC) are also considered.

A. Simulation results without measurement uncertainties

The formation responses are shown in Figures 11-13, where the methods of CA+PSC, CA+FSC, and IT2NFC+FSC are considered. The moving trajectories, position errors and the relative distances between two followers are depicted in Figures 11-13. It can be seen that even the collision avoidance can be achieved with CA+PSC and CA+FSC, however, these two strategies eventually fail to preserve the desired formation pattern. On the other hand, from Figure 13, collision-free formation can be obtained by using the proposed IT2NFC+FSC control scheme. In Figures 11-13, it seems that the control performance become worse between 30 s and 60 s. To avoid misunderstandings, we need to explain this situation further. First, the leader starts moving along a sinusoidal trajectory at \(t = 30\) s. The second follower will get close to the cylinder obstacle during the leader-following process. Then, the effort of collision avoidance is performed on the second follower that causes a detouring from its moving trajectory. Similar situations will happen in other simulation and experimental results. In particular, it can be observed that the second follower can successfully bypass the fixed obstacle and keep its way to form a designated pattern. The absolute sum of the formation errors of (11) and (12) are summarized in Table 4, where IAE is the integral absolute error, ISE is the integral square error, ITAE is the integral time absolute error, and ITAE stands for the integral time square error. In Table IV, it can be seen that the CA+FSC has much less formation error than the counterpart of CA+PSC in every index. Comparison results illustrate that the proposed fuzzy-based separation control can provide improved responses. In addition, the IT2NFC+FSC has even better performance than CA+FSC. It illustrates that the proposed neural-fuzzy formation controller can improve the formation responses effectively. It is obviously that the proposed interval type-2 neural-fuzzy formation and fuzzy separation combined controller can provide better responses than the counterparts of conventional consensus algorithm with potential-based separation control.

Table 4. Formation comparisons without uncertainties.

<table>
<thead>
<tr>
<th></th>
<th>IAE</th>
<th>ISE</th>
<th>ITAE</th>
<th>ITSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA+PSC</td>
<td>1536.9</td>
<td>575.27</td>
<td>41928.21</td>
<td>15611.02</td>
</tr>
<tr>
<td>CA+FSC</td>
<td>501.81</td>
<td>111.97</td>
<td>13203.06</td>
<td>1875.27</td>
</tr>
<tr>
<td>IT2NFC+FSC</td>
<td>281.66</td>
<td>70.58</td>
<td>5637.85</td>
<td>578.70</td>
</tr>
</tbody>
</table>
Figure 11. Simulation results without uncertainties (CA+PSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.

Figure 12. Simulation results without uncertainties (CA+FSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.

Figure 13. Simulation results without uncertainties (IT2NFC+FSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.
Figure 14. Simulation results with uncertainties (CA+PSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.

Figure 15. Simulation results with uncertainties (CA+FSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.

Figure 16. Simulation results with uncertainties (IT2NFC+FSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.
B. Simulation results with measurement uncertainties

Assume that there exist measurement uncertainties in the calculation of head position of (4). Without loss of generality, an artificial uncertainty $0.03 \cdot \text{rand}(\cdot) \ (\text{m})$ is added into the $x$- and $y$-axis head position, where $\text{rand}(\cdot) \in [-1, 1]$ is uniformly distributed. The formation responses are shown in Figures 14-16. Similar to the previous illustrations, the moving trajectories and the relative distances between two follower robots are depicted in Figures 14-16. It can be seen that even the collision avoidance can be achieved with CA+PSC and CA+FSC, however, these two strategies eventually fail to preserve the desired formation pattern. On the other hand, from Figure 16, collision-free formation can be obtained by using the proposed IT2NFC+FSC control scheme subject to uncertainties. Furthermore, it can be observed that the third follower can successfully bypass the fixed obstacle and keep its way to form a designated pattern. The formation errors of different control strategies are summarized in Table 5. It can be concluded that the combined IT2NFC and FSC controller can provide better formation responses than the counterparts of conventional consensus algorithm with potential-based separation control.

<table>
<thead>
<tr>
<th></th>
<th>IAE</th>
<th>ISE</th>
<th>ITAE</th>
<th>ITSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA+PSC</td>
<td>1960.1</td>
<td>1298.47</td>
<td>51603.11</td>
<td>34895.32</td>
</tr>
<tr>
<td>CA+FSC</td>
<td>504.36</td>
<td>112.19</td>
<td>13115.49</td>
<td>1827.67</td>
</tr>
<tr>
<td>IT2NFC+FSC</td>
<td>302.81</td>
<td>75.41</td>
<td>6048.96</td>
<td>658.98</td>
</tr>
</tbody>
</table>

C. Experimental results

As shown in Figure 17, the overall experimental system mainly consists of a vision control interface, a Bluetooth communication network and some differentially driven e-puck mobile robots. Note that the dimension of an e-puck mobile robot is the same as the parameters used in previous simulations. The sampling period is set as 100 (ms). The camera is utilized to obtain the position of each robot by the techniques of image processing. The measurement inaccuracy of robot positions can be considered as the uncertainties addressed in simulations.

Using different strategies, CA+PSC, CA+FSC and IT2NFC+FSC, the corresponding experimental results are shown in Figures 18-20, respectively. From Figures 18(a) and 18(b), it can be seen that there exits local minima in the beginning of position exchanges using the CA+PSC. After 30 seconds, the leader starts to move and the phenomenon of local minima gradually disappear, however, the formation errors are significant. From Figures 19(a) and 19(b), there are no local minima during position exchanges with the use of CA+FSC method. However, there still exist some formation errors in the phase of formation control, from 30 (sec) to 60 (sec). On the other hand, from Figures 20(a) and 20(b), followers can quickly swap positions with no local minima by using IT2NFC+FSC method. In addition, there exists no collision between followers during exchanging positions. Moreover, it can be seen that the formation errors can be significantly reduced, and the desired formation pattern can be achieved. Performance comparisons, summarized in Table 6, can highlight the superiority of the proposed IT2NFC+FSC method.

<table>
<thead>
<tr>
<th></th>
<th>IAE</th>
<th>ISE</th>
<th>ITAE</th>
<th>ITSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA+PSC</td>
<td>1460.40</td>
<td>608.40</td>
<td>39249.42</td>
<td>17246.28</td>
</tr>
<tr>
<td>CA+FSC</td>
<td>617.78</td>
<td>164.84</td>
<td>15282.07</td>
<td>2758.91</td>
</tr>
<tr>
<td>IT2NFC+FSC</td>
<td>396.80</td>
<td>108.25</td>
<td>7617.87</td>
<td>998.50</td>
</tr>
</tbody>
</table>

6. Conclusions

This paper presents an interval type-2 neural fuzzy formation controller for multi-robot systems. The learning rules for controller parameters can be derived from the gradient decent method. In addition, a fuzzy separation control is proposed to achieve collision avoidance such that the problem of local minima can be solved. Simulation and experimental results illustrate that the collision-free leader-following formation can be accomplished using the proposed control scheme. Performance comparisons indicate that the proposed interval type-2 fuzzy-based control scheme has better formation responses compared to the counterparts of conventional consensus algorithm and potential-based separation control.
Figure 18. Experiment results (CA+PSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.

Figure 19. Experiment results (CA+FSC): (a) Robots’ trajectories and formation errors, (b) The distances between robots.

Figure 20. Experiment results (IT2NFC+FSC): (a) Robots’ trajectories and formation errors, (b) The distance between robots.
Acknowledgment

This work is partially supported by Taiwan National Science Council under the grant NSC 101-2221-E-182-041-MY2. The authors also want to appreciate the High Speed Intelligent Communication Research Center, Chang Gung University, Taiwan, for continuing support in the study of multi-agent systems.

References


Yeong-Hwa Chang received the B.S. degree in electrical engineering from Chung Cheng Institute of Technology, Tao-Yuan, Taiwan, the M.S. degree in control engineering from National Chiao Tung University, Hsin-Chu, Taiwan, and the Ph.D. degree in electrical engineering from the University of Texas at Austin, U.S.A., in 1982, 1987, and 1995, respectively. He is currently a Professor with the Department of Electrical Engineering, Chang Gung University, Tao-Yuan, Taiwan. His research interests include intelligent systems, multi-robot systems, haptic control systems, and embedded control systems.

Chun-Lin Chen received the B.S. degree in electronic engineering from Lung-Hwa University of Science and Technology, Tao-Yuan, Taiwan, R.O.C., in 2002, and the M.S. degrees in electrical engineering from Chang Gung University, Tao-Yuan, Taiwan, R.O.C., in 2005, where he is currently pursuing the Ph.D. degree in intelligent control. His current research interests are on the multi-robot control, image processing, and bionic optimization algorithm.

Wei-Shou Chan received the B.S. degree in electrical engineering from Lung-Hwa University of Science and Technology, Tao-Yuan, Taiwan and the M.S. degree in electrical engineering from Chang Gung University, Tao-Yuan, Taiwan, in 2006 and 2008, respectively. He is working toward the Ph.D. degree in the Department of Electrical Engineering at Chang Gung University, Tao-Yuan, Taiwan.
R.O.C. His current research interests include intelligent systems, dynamic surface control, and multi-robot systems.

Hung-Wei Lin received the B.S. degree in electrical engineering from Feng Chia University, Taichung, Taiwan, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from National Defense University, Chung Cheng Institute of Technology, Tao-Yuan, Taiwan, in 2000 and 2004, respectively. He is currently an Assistant Professor with the Department of Electrical Engineering, Lee-Ming Institute of Technology, New Taipei City, Taiwan. His research interests include digital signal processors and embedded systems.