IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 2, 2020, accepted October 29, 2020, date of publication November 3, 2020, date of current version November 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3035348

A Human Decision-Making Behavior Model for Human-Robot Interaction in Multi-Robot Systems

JIE HUANG, (Member, IEEE), WENHUA WU, ZHENYI ZHANG, (Student Member, IEEE), AND YUTAO CHEN

College of Electrical Engineering and Automation, Fuzhou University, Fuzhou 350108, China

Corresponding author: Yutao Chen (ytchen_fzu@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61603094.

ABSTRACT In this paper, a novel human decision-making behavior model is proposed for human-robot interaction (HRI) for the control of multi-robot systems (MRS). The proposed human drift diffusion model (HDDM) combines the traditional drift diffusion model (DDM) and the null-space-based behavioral control (NSBC) method by introducing a data-processing station and a human cognitive system. In the HDDM, the evolution of human-decision information is computed. By using a threshold of such information to trigger human interaction, accurate human decision-making timing can be obtained. In addition, a cooperative controller is designed for robots to follow human instructions. Simulations under various scenarios show that by using the proposed HDDM and the controller, robots can complete human instructions more accurately comparing to traditional methods. An experiment using a group of quadrotors subject to external wind disturbances also demonstrate the effectiveness of the proposed HDDM in real-world uncertain environments.

INDEX TERMS Human intervention, multi-robot system, human drift diffusion model, human decision-making behavior model.

I. INTRODUCTION

In the past decade, multi-robot systems (MRS) have attracted much attention due to their loosely coupled network structure where robots can solve problems through interaction beyond the capability or knowledge of a single robot. In MRS, the originally large and complex system is divided into several smaller systems in which robots can communicate and cooperate. Robot formation is one of the control methods for robots to perform tasks collaboratively. Consensus problems in formation control have been widely studied, e.g. in [1]–[3].

Formation control methods in MRS include the leader-follower, the virtual-structure and the behavioral control. The leader-follower method defines robots as leader or followers, and the structure of formation is easy to implement by following the trajectory of the leader at a specified distance and relative position [4]. However, the leader-follower method relies too heavily on the trajectory of leader, which no longer has task execution capability if the leader fails [5]. The virtual-structure method forms the MRS into a rigid structure in which agent formation is easy to maintain, but this method is known to lack flexibility and adaptability [6]. The behavioral control method has a hierarchical structure that decomposes the formation task into several basic behaviors which are selected based on practical situations. Then the formation task is realized by behavior fusion [7]. However, the behavioral control has been found difficult to ensure the stability of formation control [8]. To tackle this problem, the null-space-based behavioral control (NSBC) has been proposed for generic robotic systems, in which the behaviors of robots are arranged in priority using null-space projection. In this way, tasks with higher priority can be completely executed and those with lower priority would only be executed if no conflicts occur. Moreover, the NSBC approach allows to elaborate stability properties of robot control due to its explicit usage of mathematical models [9], [10]. A disadvantage of the NSBC approach is that it cannot handle all task conflicts because it is not smart enough at the absence of human supervision and intervention [11].

Therefore, it is necessary to introduce human intervention to achieve HRI in MRS [12]. Results in constructing HRI structures can be found in [13]–[16]. In addition, human intervention methods in HRI have been studied in the leader-follower framework [17]–[19] and have been firstly

The associate editor coordinating the review of this manuscript and approving it for publication was Yilun Shang.

introduced into the behavioral control framework in [20]. Although successful in specific applications of interest, the methods aforementioned lack an accurate human model. Human models can help predict under which conditions a human being exhibits poor performance or instability, hence are necessary to improve the accuracy of human decision-making [21]. Some efforts have been reported for the purpose of developing human decision-making models. In [22], a Markov chain model is used in MRS to recognize human behaviors from sensory data. An extended decision field theory (EDFT) model has been proposed in [23] which is used for human behavior modeling in multiple sequential decisions in HRI. A drift diffusion model (DDM) has been proposed in [24] which is a cognitive process consists of simple decisions. Comparing to the Markov chain model and the EDFT model, the DDM does not employ factors such as probability or trust of human but instead uses actual sensory information in making simple decisions, e.g., two-alternative forced-choice tasks (TAFCTs). With just a few parameters and a conceptually simple process, the DDM can describe behavioral performance (human choice and reaction time) during the entire process of task execution. As a result, the DDM has been successfully applied to many human cognitive tasks in neurosciences, e.g., lexical decision [25]. In [26], a verified soft-max choice model, emerged from the DDM, is used for studying decision making in TAFCTs, and an extension of this model is proposed in social context. However, the decision making problem has not been studied in the human-robot context, hence is still an open problem involving a mixed group of human and robot decision makers [26].

This paper addresses the problem of human decision-making behavior modeling and controller design in MRS based on the behavioral control method. The specific contribution of this paper consists of the following three parts:

1) The behavioral control framework for HRI systems is extended. While earlier works usually define simple rules to decide when to trigger human intervention, we introduce a data-processing station and a human cognitive system into the behavioral control framework that allows for rigorous analysis of the human decision-making process in a system control perspective.
2) A human drift diffusion model (HDDM) is proposed by embedding the DDM into the NSBC method. This method offers a feasible solution to extend DDM to HRI systems. The HDDM models human decision-making process in HRS by computing the evolution of human decision-making information using sensor data in the NSBC method. By minimizing a human supervision cost function and setting a decision-making threshold for the non-zero initial human decision-making information, the HDDM can achieve a trade-off between speed and accuracy of human intervention. To complete the control loop, a cooperative controller is also designed to track the reference trajectory of the human intervention task generated by the HDDM.

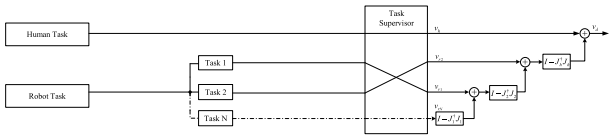

FIGURE 1. Sketch of the NSBC method in the HRI example. Human intervention is designed as a robot task just as other tasks. The human task is given the highest priority in order to be executed in any cases.

3) The theoretical results are verified in a simulation under various scenarios and also in an experiment using a group of quadrotors subject to external wind disturbances. The experimental results show that the HDDM can obtain accurate time for human decision-making in practical applications.

The remainder of the paper is organized as follows: In Section 2, the research problem and the preliminary and the robot task is designed. In Section 3, the human decision-making behavior is modeled, human intervention task and controller are designed. Section 4 and 5 are the simulation and experiment, respectively. Conclusion is drawn in Section 6. The stability of the system and the tasks are proved in Section 7.

II. UNITS

Consider a group of $n\,(n \geq 2)$ robots with the following model:

$$\dot{p}_j = v_j, \tag{1}$$

where $p_j \in \mathbb{R}^2$ is the position of the robot $j(j = 1, \ldots, n)$, and $v_j \in \mathbb{R}^2$ is the velocity vector. The formation control objective is to make robots follow human instructions and perform tasks, which are defined by different behaviors of robots, e.g., movement towards targets and obstacles avoidance. To decide when to introduce human intervention, we make the following assumption:

Assumption 1: Assuming that robots are completely autonomously controlled, the human intervention can only happen when the controller of the robots fail to finish tasks.

Remark 1: In this paper, we consider the decision-making problem in HRI systems, i.e. when and how human could intervene robot tasks. The controller design presented in later sections is to close the loop and verify the feasibility of the proposed HDDM. Therefore, a simple first-order dynamic model (1) is employed.

A sketch of the NSBC method in the HRI framework including N robot tasks is shown in Fig. 1. The human intervention task is set to the highest priority. A task supervisor is employed to adjust the priority of tasks. A task with a lower priority is projected to the null-space of a higher priority task. In the following, we briefly introduce the concept of general tasks, and then introduce two specific elementary tasks, i.e. the move-to-target task and the obstacle-avoidance task. Details of the human intervention task is given in Section III.

A general task function can be designed as:

$$\rho = f(p), \tag{2}$$

where p is the set of $p_j (1 \leqslant j \leqslant n)$, $p = [p_1, \cdots, p_n]^T$, f is a mapping of robots position to task function. The derivative of task function (2) is:

$$\dot{\rho} = \frac{\partial f(p)}{\partial p}\dot{p}_j = J(p)\,\dot{p}, \tag{3}$$

where $J(p) \in \mathbb{R}^{m\times n}$ is the Jacobian matrix which maps task function ρ to robot trajectory p, and m is the dimension of task function ρ.

The desired trajectory function ρ_d is the target trajectory that required for a certain control purposes, e.g. formation, avoiding obstacles and tracking, while the general task function ρ in (2) is the actual trajectory of robots. In the process of task execution, the desired task function $\rho_d(t)$ corresponds to the desired trajectory $p_d(t)$, which can be obtained by integrating robot velocity over time t. However, an integration of velocity may result in numerical drifts. This problem can be counteracted by the so-called Closed Loop Inverse Kinematics version algorithm with an output written as:

$$v_d = J_d^{\dagger}\left(\dot{\rho}_d + \Lambda\tilde{\rho}\right), \tag{4}$$

where v_d is the task velocity output, $\dot{\rho}_d$ is the derivative of the desired task function and $\tilde{\rho} = \rho_d - \rho$ is the task error. Matrix Λ is a positive constant gain and $J^{\dagger} = J^T\left(JJ^T\right)^{-1}$ is the pseudo-inverse of the Jacobian matrix. As shown in the Fig. 1, if there exist multiple tasks, each task is assigned with a priority level and then all composed tasks are fused based on the NSBC method, leading to

$$v_d = v_{1d} + N_1\left(v_{2d} + (N_2 v_{3d} + \cdots)\right), \tag{5}$$

where N_1 and N_2 are the null-space of the task 1 and 2. We have $N_{\vartheta} = I - J_{\vartheta}^{\dagger}J_{\vartheta}$, where the index ϑ indicates the priority order of the task. I is the identity matrix of appropriate dimension. v_{1d}, v_{2d} and v_{3d} are the task output of task 1, task 2 and task 3, respectively.

Remark 2: In NSBC, each task is assigned with a priority, and all composed task can be fused as (5) in a unique priority order using null-space projection. More details can be found in [27]–[29].

A. MOVE-TO-TARGET TASK FUNCTION DESIGN

The move-to-target task is defined by the movements of the robots team towards the target points. Once every robot arrived at the target point, the individual robot would stop. Thus, the behavior corresponding to the move-to-target task of the jth robot can be encoded by the function ρ_{mj} as:

$$\rho_{mj} = p_j, \tag{6}$$

where $p_j = [\,x_j \;\; y_j\,]^T$ is the position coordinates of the jth robot, and the desired function ρ_{mdj} is:

$$\rho_{mdj} = p_{gj}, \tag{7}$$

where the p_{gj} is the desired position of the jth robot. The output of ρ_{mdj} can be encoded by the NSBC method as:

$$v_{mj} = J_{mj}^{\dagger}\left(\dot{\rho}_{mdj} + \Lambda_{mj}\tilde{\rho}_{mj}\right), \tag{8}$$

where Λ_{mj} is a positive constant of gains of the move-to-target task, $\tilde{\rho}_{mj} = \rho_{mdj} - \rho_{mj}$ is the task error, and $J_{mj}^{\dagger} = J_{mj}^T\left(J_{mj}J_{mj}^T\right)^{-1}$ is the pseudo-inverse of J_{mj}.

B. OBSTACLES-AVOIDANCE TASK FUNCTION DESIGN

At the presence of an obstacle in the advancing direction, the aim of obstacles-avoidance task is to keep the robot on a safe distance from the obstacle, which means obstacles-avoidance task is set higher priority in robot tasks. Define D as the safety distance or the radius of the circular safety region. Once the obstacles are within the safety region of robots, the robots must execute the obstacles-avoidance behavior, which is defined as follows:

$$\rho_{aj} = \left\|p_j - p_{oq}\right\|, \quad \rho_{adj} = D, \tag{9}$$

where ρ_{aj} is the function of obstacles-avoidance task and p_{oq} is the obstacle position of the qth obstacle, $q = 1, 2, \cdots Q$. Define ρ_{adj} as the desired function. Then the velocity output function is given by:

$$v_{aj} = J_{aj}^{\dagger}\left(\dot{\rho}_{adj} + \Lambda_{aj}\tilde{\rho}_{aj}\right), \tag{10}$$

where Λ_{aj} is a positive constant of gains, and $\tilde{\rho}_{aj} = \rho_{adj} - \rho_{aj}$ is the task error, $J_{aj}^{\dagger}$ is the pseudo-inverse of the Jacobian matrix J_{aj}.

In this paper, the obstacles-avoidance task is set to the highest priority. As a result, the move-to-target task (v_{mj}) is projected onto the null-space of obstacles-avoidance task (v_{aj}), giving the final output of the jth robot as:

$$v_{rj} = v_{aj} + \left(I - J_{aj}^{\dagger}J_{aj}\right)v_{mj}, \tag{11}$$

where v_{rj} is the velocity output, and $I - J_{aj}^{\dagger}J_{aj}$ is the null-space of the obstacle avoidance task. If J_{aj} has full rank, the null-space of the obstacles-avoidance task is empty and the whole task output v_{mj} would be filtered out.

III. HUMAN DECISION-MAKING BEHAVIOR MODELING AND INTERVENTION TASK DESIGN

In this section, we develop the HDDM by embedding the DDM into the NSBC method. This is not trivial since there exist various sensor information but only one of which should be taken as the actual human sensory information. We show how to select one specific sensor information as human sensory information to model the human decision-making behavior. In addition, the human decision-making threshold in DDM is set based on the condition that the initial human decision-making information is zero. However, this is no longer the case in the HDDM. Therefore, we show how to compute the decision-making threshold with non-zero initial decision-making information in the HDDM.

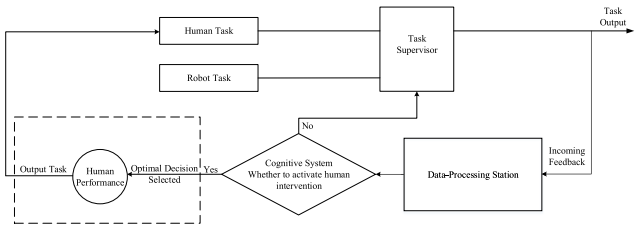

FIGURE 2. The decision making schematic diagram under human intervention during task execution. The task supervisor fuses tasks as in (5). The data-processing station computes and monitors the human decision-making information to decide whether to activate human intervention.

The decision making schematic diagram under human intervention is shown in Fig.2. Comparing to [20] where the human decision-making is a simple process to evaluate whether human intervention behavior is needed, we analyze the human decision-making process by introducing a data-processing station and a human cognitive system. The data-processing station receives incoming feedback information from robots sensors, and the human decision-making process is modeled if decision-making information has been received from the incoming feedback. In the human cognitive system, human judges whether to intervene robots based on the human decision-making model.

A. HUMAN DECISION-MAKING PROCESS MODELING

The HDDM can be written as

$$dZ = Vdt + \sigma dW, \tag{12}$$

where Z is the human decision-making information. Here, the HDDM contains two parts: the first is Vdt, which represents the amount of change in human decision-making information per unit time. The second part, σdW, represents white noise with Gaussian distribution, where W is a Wiener process, and σ is the standard deviation of the noise. The mechanism of the HDDM is as follows. Human decision-making information is first obtained from sensors on the robots, which is then translated to an incoming feedback to a data-processing station. In the HDDM, a noisy process that accumulates the human decision-making information over time is monitored. Once the accumulated information has reached a pre-defined threshold, human intervention is activated. The rate of accumulation of information is defined as the drift rate.

A key issue is to select a signal as the human decision-making information. As shown in Fig.3, various signals can be selected as the incoming feedback, e.g., p_j, v_j, velocity error $\tilde{v}_j$, position error $\tilde{p}_j$, ρ_{aj}, p_{oq} and p_{gj}. Some can reflect the progress of task execution directly (e.g., p_j, v_j, $\tilde{v}_j$, and $\tilde{p}_j$), but others cannot (e.g., ρ_{aj}, p_{oq} and p_{gj}). The criterion for selecting one specific signal is based on whether a signal can reflect accurately the task executing process. In the NSBC method, robots execute tasks according to a preset task function. As a result, the position error $\tilde{p}_j$, which is the difference between the preset and the actual position, is selected as the

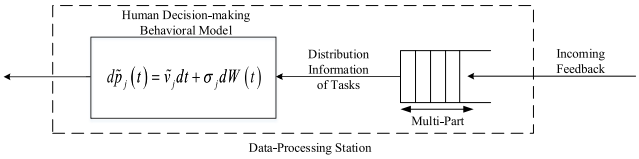

FIGURE 3. An illustration of the data-processing station in the HRI control framework.

decision-making information Z. As a result, we have

$$d\tilde{p}_j(t) = \tilde{v}_j dt + \sigma_j dW(t), \tag{13}$$

where $\tilde{p}_j = p_{rdj} - p_j$, and $\tilde{p}_j$ is used for supporting human decision-making on the jth robot, p_{rdj} and p_j are the preset and the actual position at time t, respectively. The change in $\tilde{p}_j$ over a unit time dt is denoted as $d\tilde{p}_j$. Note that the drift rate, which is a key parameter in the traditional DDM, is denoted as the time-varying velocity error $\tilde{v}_j = v_{rdj} - v_j$, in which v_{rdj} is the preset velocity required to reach the preset position and v_j is the actual velocity.

B. HUMAN DECISION-MAKING THRESHOLD

The HDDM is used in the free response paradigm [30], where human makes decisions once the amount of human decision-making information is greater than a preset threshold for the first time during the task execution process [31]. In this paper, the threshold is determined based on the Bayes Risk (BR) criteria [32], which is a common practice to achieve human decision-making speed-accuracy trade-off. The BR criteria assumes that human seeks to minimize a cost function B which is defined by the weighted sum of decision time and error rate, given as:

$$B = c_{1j}T_j + c_{2j}E_j, \tag{14}$$

where c_{1j} is the observing cost per unit time in the decision process and c_{2j} is the cost of human decision-making error. The two variables T_j and E_j are the human decision-making timing and the decision-making error rate, respectively, which can be computed by [31]:

$$E_j = \frac{1}{1 + e^{2\bar{\varsigma}_j\bar{a}_j}} - \left(\frac{1 - e^{-2\tilde{p}_{0j}\bar{a}_j}}{e^{2\bar{\varsigma}_j\bar{a}_j} - e^{-2\bar{\varsigma}_j\bar{a}_j}}\right) \tag{15}$$

$$T_j = \bar{\varsigma}_j \tanh\left(\bar{\varsigma}_j\bar{a}_j\right) + \left(\frac{2\bar{\varsigma}_j\left(1 - e^{-2\tilde{p}_{0j}\bar{a}_j}\right)}{e^{2\bar{\varsigma}_j\bar{a}_j} - e^{-2\bar{\varsigma}_j\bar{a}_j}} - \tilde{p}_{0j}\right), \tag{16}$$

where $\bar{\varsigma}_j = \frac{\varsigma_j}{\tilde{v}_j}$ and $\bar{a}_j = \left(\frac{\tilde{v}_j}{\sigma_j}\right)^2$. In (15) and (16), ς_j is the human decision-making threshold for the proposed HDDM, and $\tilde{p}_{0j}$ is the initial position error in the robot task execution. The error rate E_j decreases exponentially with $\bar{\varsigma}_j$ and the decision time T_j grows super linearly with ς_j. As a result, B has a minimum [31]. The threshold for human decision-making which minimizes B for given parameters ($\bar{a}_j$, c_{1j}, c_{2j}) can be computed by substituting (15) and (16) with the initial decision-making information value $\tilde{p}_{0j}$ in (14), and solving

the equation $\frac{dB}{d\varsigma_j} = 0$, resulting in:

$$C_j\left[\frac{-2\bar{\varsigma}_j e^{2\vartheta}}{\left(1+e^{2\vartheta}\right)^2} - \frac{\left(e^{-2\tilde{p}_{0j}\bar{a}_j}-1\right)\left(2\bar{a}_j e^{2\vartheta}+2\bar{a}_j e^{-2\vartheta}\right)}{\left(e^{2\vartheta}-e^{-2\vartheta}\right)^2}\right] + \left[\left(\tanh\left(\vartheta\right)+\bar{a}_j\right) + \frac{2\left(1-e^{-2\tilde{p}_{0j}\bar{a}_j}\right)\left(e^{2\vartheta}-e^{-2\vartheta}\right)}{\left(e^{2\vartheta}-e^{-2\vartheta}\right)^2}\right] + \frac{(1-2\vartheta)}{\left(e^{2\vartheta}-e^{-2\vartheta}\right)^2} = 0 \tag{17}$$

where $\vartheta = \bar{\varsigma}_j\bar{a}_j$, and $C_j = \frac{c_{2j}}{c_{1j}}$ is a positive constant ratio. Since the threshold ς_j is preset and constant, the drift rate $\tilde{v}_j$ in (17) is equal to the initial velocity error $\tilde{v}_{0j}$. It should be noted that, the threshold ς_j is unknown but $(\bar{a}_j, c_{1j}, c_{2j}, P_{0j})$ are known parameters. After determining the threshold ς_j, once $\tilde{p}_j$ is greater than the threshold for the first time, human should select a behavior among the set of human behaviors.

C. HUMAN INTERVENTION BEHAVIOR TASK DESIGN

Human decision-making behavior is a cognitive process, in which one behavior is selected from a set of human behaviors to help robots team finish tasks successfully. In this paper, we consider two human behaviors, namely the supervision behavior and the intervention behavior. The supervision behavior is defined by human monitoring the robot task execution process without intervention until failure is detected. The intervention behavior is defined by human partially intervening robots. Other human behaviors, such as planning, recording, can also be taken into account, leading to slightly different task design processes. For the human supervision behavior, there is no task input to the robots. Instead, for the human intervention behavior, task input to the robots should be provided. Thus, the task corresponding to the human intervention behavior should be designed.

As shown in Fig. 2, human does not directly control the robot team, but rather sends a task as input to the robot team through the supervisor. In this paper, the human intervention behavior task is set to the highest priority and the original robot task is assigned with a lower priority. In the NSBC method, a supervisor acts as an administrator module in the process of executing tasks when robots react to the surroundings (e.g. obstacles). As a result, each robot must be equipped with a supervisor to adjust the priority of behaviors based on the internal states of robot or the local information stimuli.

Define the human intervention task function as:

$$\rho_h = f\left(p_h\right), \tag{18}$$

where $\rho_h \in \mathbb{R}^m$ and p_h is the real-time position of the robot that the human would intervene. The derivative of the human intervention task is given by:

$$\dot{\rho}_h = \frac{\partial f\left(p_h\right)}{\partial p_h}\dot{p_h} = J_h v_h, \tag{19}$$

where $J\left(h\right) \in \mathbb{R}^{m\times 1}$ is the Jacobian matrix, and $J\left(h\right)$ is the mapping between ρ_h and p_h. Through inverting the Jacobian matrix J_h, a motion reference trajectory of ρ_h can be generated:

$$v_h = J_h^{\dagger}\left(\dot{\rho}_{hd} + \Lambda_h\tilde{\rho}_h\right), \tag{20}$$

where Λ_h is a positive definite constant in human intervention task and $\tilde{\rho}_h = \rho_{hd} - \rho_h$ is the human intervention task error. $J_h^{\dagger}$ is the pseudo-inverse of J_h.

After the human intervention task is designed, the fusion of human intervention tasks and robot tasks is needed. v_r is the final velocity signal of robot tasks and v_h is the velocity output of human intervention task. Given the preset priority order, according to Fig. 1, v_r is projected onto the null-space of human intervention task. As a result, the final desired output of the behavior control method under human intervention is obtained as:

$$v_d = v_h + \left(I - J_h^{\dagger}J_h\right)v_r. \tag{21}$$

It is worth noting that if $I - J_h^{\dagger}J_h = 0$, robot tasks cannot be executed simultaneously with the human task.

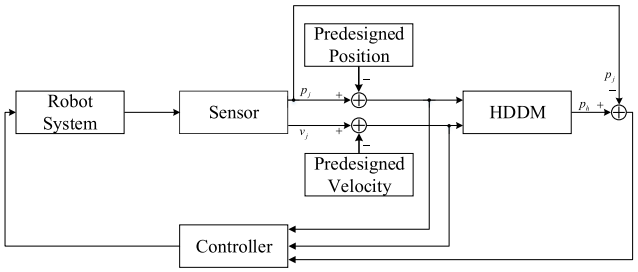

FIGURE 4. The diagram of the cooperative controller with HDDM in the loop. The HDDM provides trajectories to be tracked for the controller.

D. CONTROLLER DESIGN

In this section, a controller is designed for HRI systems based on the behavioral control. The behavioral control method is compatible with any types of controller, since it only provides robot motion reference trajectory [33]. We develop a cooperative controller for robots motion and human intervention. The diagram of the controller feedback loop is shown in Fig. 4. The tracking error e_j is defined as the distance between the reference position p_{dj} from the NSBC method and the actual position p_j as:

$$e_j = p_{dj} - p_j \tag{22}$$

The first order derivative of e_j is:

$$\dot{e}_j = \dot{p}_{dj} - \dot{p}_j = v_{dj} - u_j \tag{23}$$

where v_{dj} is the reference velocity from the NSBC method, and u_j is the actual velocity. The Lyapunov function candidate can be defined as:

$$V_c = \frac{1}{2}{e_j}^T e_j \tag{24}$$

Through the first order derivative of V_c, the following formula can be obtained:

$$\dot{V}_c = {e_j}^T\dot{e}_j \tag{25}$$

TABLE 1. Parameter values used in the simulation.

Parameter	Value
initial position p_{0j}	$\begin{bmatrix} 0 & 8 \end{bmatrix}^T \begin{bmatrix} 4 & 6 \end{bmatrix}^T \begin{bmatrix} 4 & 0 \end{bmatrix}^T$
obstacles p_{oq}	$\begin{bmatrix} 8 & 11 \end{bmatrix} \begin{bmatrix} 13 & 10.5 \end{bmatrix} \begin{bmatrix} 10 & 4 \end{bmatrix}$
task function ρ_{rd1}	$\begin{bmatrix} 1.4t+2 & 0.8t+8 \end{bmatrix}^T$
task function ρ_{rd2}	$\begin{bmatrix} 1.4t+6 & 0.8t+6 \end{bmatrix}^T$
task function ρ_{rd3}	$\begin{bmatrix} 1.4t+5 & 0.8t+1 \end{bmatrix}^T$
$\Lambda_{mj}, \Lambda_{aj}, \Lambda_h$	15, 0.8, 1.5
safe distance D	$1.5m$
$c_{1j}, c_{2j}, C_j, \sigma_j$	0.5, 10, 20, 1
threshold ς_j	$3m$

To guarantee the error e_j converge to 0, the condition $\dot{V}_c \leqslant 0$ should be satisfied, resulting in:

$$\dot{e_j} = -\kappa e_j \tag{26}$$

where $\kappa \in \mathbb{R}^{2\times 2}$ is a positive definite gain matrix. Substituting (26) into (23) we obtain the control law:

$$u_j = v_{dj} + \kappa e_j, \tag{27}$$

According to (27), we have the following theorem.

Theorem 1: Using the control law (27) for the first-order robot system (1), the position error $\tilde{p}_j$ and the velocity error $\tilde{v}_j$ approach to zero asymptotically.

Proof: See the Appendix.

Remark 3: During the robot task execution, autonomous robots encounter problems that cannot be solved by their local controller, i.e., the classical local minima problem [34]. This particular situation would cause the robots stop somewhere. In this paper, the human intervention behavior (21) can be employed to solve this problem. In addition, the controller is designed to enable the robots to follow the reference trajectory of the human intervention task under the NSBC framework.

Remark 4: In this paper, we focus on the decision-making problem in HRI systems. Model uncertainties and external disturbances are not taken into account.

IV. SIMULATION

In this section, a platoon of three robots moving in the $x-y$ plane is considered, and each robot is modeled as a first-order system. The control objective is to make the robot team move to the desired targets in an unknown environment without collision with obstacles. The robot team is equipped with sensors to detect obstacles. The values of parameters used in this simulation are shown in Table 1, where $j \in \{1, 2, 3\}$. The human decision-threshold for all the three robots are set as $3m$. In 1, the parameters (c_{1j}, c_{2j}, C_j and σ_j) are the same for the three robots. The initial velocity error $\tilde{v}_{0j} = 1\ m/s$.

A. ROBOT ENCOUNTERS LOCAL MINIMA PROBLEM

In the first simulation, robot 1 and robot 3 is able to finish tasks autonomously as shown in Fig. 5. However, when robot 2 is avoiding the first obstacle at $(13m, 10.5m)$, its sensor has detected another new obstacle at $(13m, 8m)$, which makes it stuck in the local minima point. Fig. 6 shows that the distance between the first robot 2 and the obstacle is equal to that between robot 2 and the new obstacle, from $5.6s$ to $6.22s$. In this case, the human decision-making information has reached the preset decision threshold at time $6.22s$, as shown in Fig. 7.

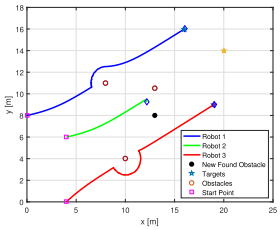

FIGURE 5. Trajectories of the three robots. The one in the middle encounters a local minima problem.

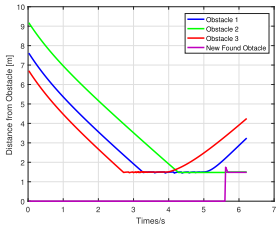

FIGURE 6. Distances between robots and their nearby obstacles in the local minima scenario.

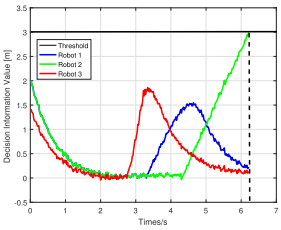

FIGURE 7. The evolution of decision-making information (position error) in the local minima problem.

B. ROBOTS FINISH TASKS WITH HUMAN INTERVENTION

In the second simulation, human intervention is triggered on to help robots escape from the local minima point. The human intervention task is designed to make robot 2 move to the new target at $(14m, 9m)$. After finishing the human intervention task, robot 2 execute its original task towards to the original target. The trajectory of robot 2 under human intervention is shown in Fig. 8.

Fig. 9 shows that when performing the human intervention task, robot 2 does not perform the original obstacles-avoidance task even if the distance between the robot and the obstacle is smaller than the preset safe distance $1.5m$. In Fig. 10, the human decision-making information value temporarily increases during the human intervention task, and then decreases as robot 2 moves away from the local minima point.

Fig. 11 shows the when the human intervention is triggered on. In this paper, robots position error is used as the human decision-making information, leading to an exact match of

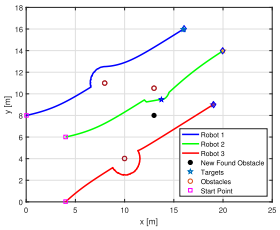

FIGURE 8. Trajectories of the three robots in case of human intervention.

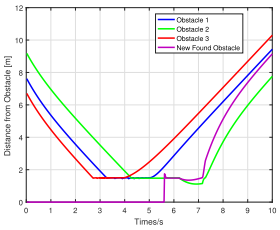

FIGURE 9. Distance between robots and obstacles in case of human intervention.

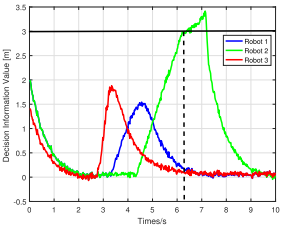

FIGURE 10. The evolution of decision-making information (position error) in case of human intervention.

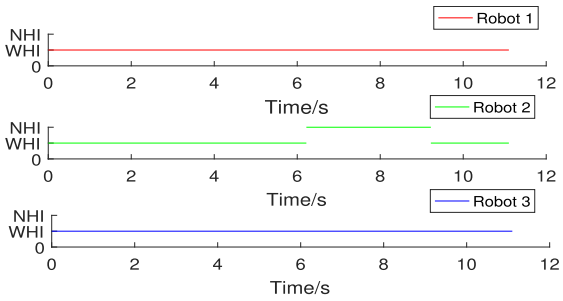

FIGURE 11. Indicator of the human intervention. NHI: human intervention triggered on; WHI: human intervention triggered off.

the human intervention timing during the local minima problem period, which is between 6.22*s* and 9.4*s*. This shows the advantage of the proposed HDDM when compared to tte traditional method in [20], where the timing of human intervention does not match that of the local minima problem period.

V. EXPERIMENT

In this section, the proposed HDDM and the cooperative controller is implemented in a group of quadrotors for an experiment of MRS with HRI.

A. EXPERIMENTAL CONFIGURATION

Three unmanned aerial vehicles (UAVs), which are quadrotors, are used in this experiment. The controller of each UAV

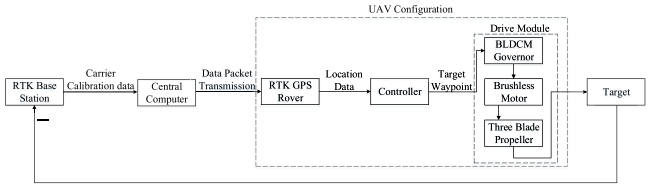

FIGURE 12. The experimental configuration of UAVs.

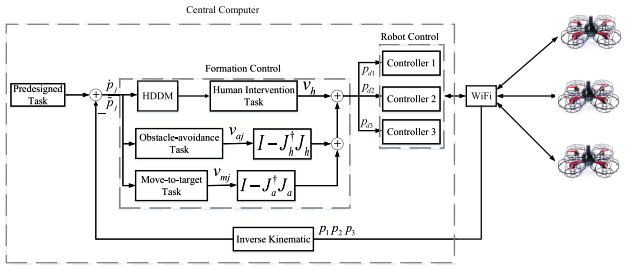

FIGURE 13. The control diagram used in the experiment.

is a PID controller. In the experiment, the UAVs may be subject to external wind disturbances. The schematic configuration of the experiment is shown in Fig. 12. Specifically, a real-time kinematic (RTK) based station obtains the UAVs position and sends carrier calibration data to a central computer. The calibration data is packed in the central computer and is sent to every UAV. The RTK global positioning system (GPS) rover in each UAV can provide the location data, and then sends it to the UAVs controller. Finally, the UAVs are driven to target according to target waypoint information by the drive module, which contains a brushless direct current motor (BLDCM) and three blade propellers. In this process, the velocity of UAVs is not adjustable. In total, the experiment employs a central computer, a WiFi Access Point (AP), a differential GPS base station and a wireless network router.

In this experiment, the UAVs only use their own sensors to detect obstacles and other UAVs, without information exchange among themselves. As shown in the Fig. 13, this experiments contains the human intervention task, the move-to-target task and the obstacle-avoidance task. In addition, the human intervention task is set to the highest priority, and the priority of obstacle-avoidance task is always higher than the move-to-target task. The safe distance between two UAVs is set as 3*m*, and the human decision-making threshold is set as 3*m*. The world coordinate system in meters is used, which is with due north as the positive direction of x axis and due east as the positive direction of y axis.

B. EXPERIMENTAL RESULTS

Fig. 14 shows that a team of three UAVs is avoiding obstacles, which are represented as yellow columns. Fig. 15 shows that while UAV 2 is avoiding obstacle 2 at (13*m*, 10.5*m*), a new obstacle at (13*m*, 5*m*) is detected. When the distance d between UAV 2 and obstacle 2 is equal to that between UAV 2 and the new obstacle, UAV 2 encounters the local minima problem, which cannot be solved by its own autonomous control system. As shown in Fig. 16, UAV 2 alternatively

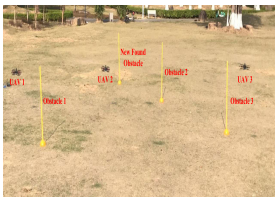

FIGURE 14. The UAVs are avoiding obstacles.

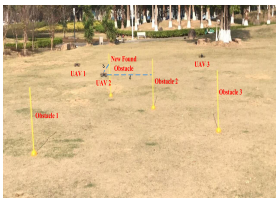

FIGURE 15. UAV 2 has encountered a local minima problem.

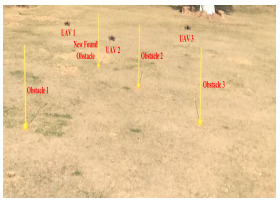

FIGURE 16. UAV 2 executes the human intervention task and escapes from the local minima point.

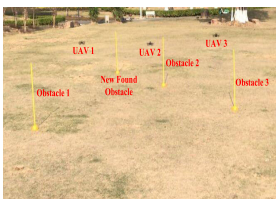

FIGURE 17. UAV 2 executes its original task and finally reaches the target.

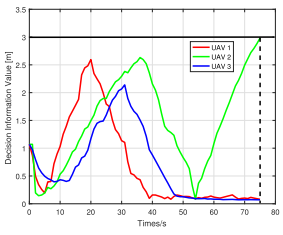

FIGURE 18. The evolution of human decision making information of all three UAVs.

executes the human intervention task, and moves away from the local minima point. Fig. 17 shows that after UAV 2 finishing the human intervention task, its original task is executed leading to the final target.

In Fig. 18, the evolution of human decision-making information value is given. Since UAV 2 has encountered a local minima problem, the human decision-making information of UAV 2 first reached the preset threshold. The accurate

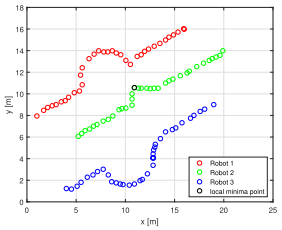

FIGURE 19. Trajectories of UAVs recorded by the UAV formation software.

timing of human decision-making (76s) is obtained. In order to help UAV 2 solve the local minima problem, the human intervention behavior has been chosen, giving the human intervention task as an input to UAV 2. The trajectories of all three UAVs is shown in Fig. 19, which are recorded by the UAV formation software. Each circle represents the position of UAVs at a given sampling time.

VI. CONCLUSION

This paper has proposed a novel HDDM by combining the traditional DDM and the NSBC method. The HDDM is used for modeling the human decision-making process in HRI systems. The behavioral control framework in HRI is extended by introducing a data station and a human decision cognitive process. By computing and monitoring the evolution of the human decision-making information, accurate human decision-making timing can be obtained. Morever, a cooperative controller has been designed using the Lyapunov direct method, which can realize human-robot interaction and cooperation. A simulation and an experiment has demonstrated the effectiveness of the proposed HDDM in real-world applications.

APPENDIX

PROOF OF THEOREM 1

Consider the Lyapunov candidate function V:

$$V = V_s + V_T, \tag{28}$$

where V_s is Lyapunov function used to prove the stability of the controller (25), which can be computed by

$$V_s = \sum_{j=1}^{n} \frac{1}{2} u_j^T u_j. \tag{29}$$

Similarly, the Lyapunov candidate function for the task is denoted as V_T.

First we prove the controller is stable. The derivative of V_s is

$$\begin{aligned} \dot{V}_s &= \sum_{j=1}^{n} u_j \dot{u}_j = \sum_{j=1}^{n} \left(v_{dj} + \kappa e_j\right) \left(\kappa \dot{e}_j\right) \\ &= \sum_{j=1}^{n} \left(v_{dj} + \kappa e_j\right) \left(-\kappa^2 e_j\right) \end{aligned} \tag{30}$$

where $\dot{e}_j$ is given in (26), and κ is a positive constant gain matrix satisfying $-\kappa^2 \leqslant 0$.

It is worth noting that the sign of the trajectory and velocity error is identical. If the sign of the trajectory error is positive, which means the practical position lags behind the reference position, a positive velocity error is needed to catch up with the reference position. If the sign of the trajectory error is negative, which means the practical position is ahead of the reference position, a negative velocity is needed to return to the reference position. As a result, $v_{dj} + \kappa e_j$ and $-\kappa^2 e_j$ always have opposite signs leading to $\dot{V}_s \leqslant 0$.

Next we prove the stability of the task. In the case that there is no obstacle in the environment, robots do not need to execute the obstacle-avoidance task hence the local minimum problem can be avoided. The equation of task stability can be written as the following:

$$V_T = \frac{1}{2}\tilde{\rho}_m^T \tilde{\rho}_m. \tag{31}$$

Taking the first derivative of (31) gives:

$$\dot{V}_T = -\tilde{\rho}_m^T J_m J_m^\dagger \Lambda_m \tilde{\rho}_m \leqslant 0, \tag{32}$$

where $J_m J_m^\dagger = I^{2\times 2}$. If there are obstacles in the environment, let us construct the Lyapunov function as:

$$V_T = \frac{1}{2}\tilde{\rho}_h^T \tilde{\rho}_h + \frac{1}{2}\tilde{\rho}_a^T \tilde{\rho}_a + \frac{1}{2}\tilde{\rho}_m^T \tilde{\rho}_m = \frac{1}{2}\tilde{\delta}^T \tilde{\delta}, \tag{33}$$

where $\tilde{\delta} = \begin{bmatrix} \tilde{\rho}_h^T & \tilde{\rho}_a^T & \tilde{\rho}_m^T \end{bmatrix}^T$. Taking the derivative of V_T gives:

$$\dot{V}_T = \tilde{\delta}^T \dot{\tilde{\delta}} = -\tilde{\delta}^T \begin{bmatrix} J_h & J_a & J_r \end{bmatrix}^T v_d. \tag{34}$$

Substituting (11) into (21), we obtain the task output as:

$$v_d = J_h^\dagger \Lambda_h \tilde{\rho}_h + N_h \Lambda_a J_a^\dagger \tilde{\rho}_a + N_h N_a J_h^\dagger \Lambda_m \tilde{\rho}_m, \tag{35}$$

where $N_h = I - J_h^\dagger J_h$, $N_a = I - J_a^\dagger J_a$. Substituting (35) into (34) we obtain

$$\begin{aligned} \dot{V}_T &= -\tilde{\delta}^T \begin{bmatrix} \Lambda_h & O_{mh,ma} & O_{mh,mm} \\ J_h^\dagger J_a \Lambda_h & N_h J_a J_a^\dagger \Lambda_a & J_a \bar{N} J_m^\dagger \Lambda_m \\ J_m J_h^\dagger \Lambda_h & J_m N_h J_a^\dagger \Lambda_a & J_m \bar{N} J_m^\dagger \Lambda_m \end{bmatrix} \tilde{\delta} \\ &= -\tilde{\delta}^T \underbrace{\begin{bmatrix} A_{11} & O_{mh,ma} & O_{mh,mm} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}}_{A} \delta^T \end{aligned} \tag{36}$$

where $\bar{N} = N_h N_a$, m_h and m_a are the dimension of the human intervention task and the obstacle avoidance task, respectively. Here, $J_h N_h$ is equal to 0.

A necessary condition for $\dot{V}_T \leq 0$ is that the matrix A in (36) is positive definite. A sufficient condition of this is that all the sub-matrices of A are positive definite. It is obvious that A_{11} is positive definite as long as the gain $\Lambda_h > 0$. The sub-matrix A_{22} is positive definite if the human intervention task and the obstacle-avoidance task are independent [35] and the gain $\Lambda_a > 0$. The sub-matrix A_{33} is positive definite if the move-to-target task is independent to the augmented Jacobian matrix obtained by stacking the obstacles-avoidance task and the human intervention task, and the gain Λ_m is positive. Since the rank of the Jacobian matrices $J_h^\dagger$, $J_a^\dagger$ and $J_m^\dagger$ are all 2, which satisfies the task independent condition in [35], hence sub-matrices A_{11}, A_{22} and A_{33} are positive definite. However, sub-matrices A_{21}, A_{31} and A_{32} are positive definite only if the human intervention task, the obstacle-avoidance task and the move-to-target task are orthogonal, which cannot be guaranteed since the sign of $\bar{N} = N_h N_a$ is unclear.

We observe that (36) satisfies the following inequality:

$$\begin{aligned} \dot{V}_T \leqslant & -\Lambda_{11,m} \| \tilde{\rho}_h \|^2 - \Lambda_{22,m} \| \tilde{\rho}_a \|^2 - \Lambda_{33,m} \| \tilde{\rho}_m \|^2 \\ & + \Lambda_{21,M} \| \tilde{\rho}_h \| \, \| \tilde{\rho}_a \| + \Lambda_{31,M} \| \tilde{\rho}_h \| \, \| \tilde{\rho}_m \| \\ & + \Lambda_{32,M} \| \tilde{\rho}_a \| \, \| \tilde{\rho}_m \| \\ = & -\hat{\delta}^T \Delta \hat{\delta}, \end{aligned} \tag{37}$$

where $\Lambda_{11,m} = \min\{\Lambda_h\}$, $\Lambda_{22,m} = \min_j\{\Lambda_{aj}\}$ and $\Lambda_{33,m} = \min\{\Lambda_{mj}\}$ are the upper bound on the induced norm of A_{21}, A_{31} and A_{32}, respectively. Also

$$\hat{\delta} = \begin{bmatrix} \| \tilde{\rho}_h \| & \| \tilde{\rho}_a \| & \| \tilde{\rho}_m \| \end{bmatrix}^T,$$

$$\Delta = \begin{bmatrix} \Lambda_{11,m} & -\frac{\Lambda_{31,M}}{2} & -\frac{\Lambda_{31,M}}{2} \\ -\frac{\Lambda_{21,M}}{2} & \Lambda_{22,m} & -\frac{\Lambda_{32,M}}{2} \\ -\frac{\Lambda_{31,M}}{2} & -\frac{\Lambda_{32,M}}{2} & \Lambda_{33,m} \end{bmatrix},$$

where $\| \cdot \|$ is the Euclidean norm. The matrix Δ is positive definite and symmetric because its main diagonal elements are positive and its eigenvalues are positive. Thus, (37) can be rewritten as:

$$\dot{V}_T \leqslant -\lambda_{\min}\{\Delta\} \hat{\delta}^T \hat{\delta} \leqslant 0, \tag{38}$$

where $\lambda_{\min}\{\Delta\}$ is the minimum eigenvalue of Δ. This completes the proof. ■

REFERENCES

[1] J. Huang, N. Zhou, and M. Cao, "Adaptive fuzzy behavioral control of second-order autonomous agents with prioritized missions: Theory and experiments," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9612–9622, Dec. 2019.

[2] W. Ren and R. W. Beard, *Distributed Consensus in Multi-Vehicle Cooperative Control*. London, U.K.: Springer, 2008.

[3] W. Hu, L. Liu, and G. Feng, "Output consensus of heterogeneous linear multi-agent systems by distributed event-triggered/self-triggered strategy," *IEEE Trans. Cybern.*, vol. 47, no. 8, pp. 1914–1924, Aug. 2017.

[4] J. Cruz, "Leader-follower strategies for multilevel systems," *IEEE Trans. Autom. Control*, vol. 23, no. 2, pp. 244–255, Apr. 1978.

[5] S. He, M. Wang, S.-L. Dai, and F. Luo, "Leader–follower formation control of USVs with prescribed performance and collision avoidance," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 572–581, Jan. 2019.

[6] W. Ren and R. W. Beard, "Decentralized scheme for spacecraft formation flying via the virtual structure approach," *J. Guid., Control, Dyn.*, vol. 27, no. 1, pp. 73–82, Jan. 2004.

[7] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Trans. Robot. Autom.*, vol. 14, no. 6, pp. 926–939, Dec. 1998.

[8] J. Chen, M. Gan, J. Huang, L. Dou, and H. Fang, "Formation control of multiple euler-Lagrange systems via null-space-based behavioral control," *Sci. China Inf. Sci.*, vol. 59, no. 1, pp. 1–11, Jan. 2016.

[9] G. Antonelli and S. Chiaverini, "Kinematic control of platoons of autonomous vehicles," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1285–1292, Dec. 2006.

[10] Y. Chen, Z. Zhang, and J. Huang, "Dynamic task priority planning for null-space behavioral control of multi-agent systems," *IEEE Access*, vol. 8, pp. 149643–149651, 2020.

[11] G. Antonelli, F. Arrichiello, S. Chakraborti, and S. Chiaverini, "Experiences of formation control of multi-robot systems with the null-space-based behavioral control," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 1068–1073.
[12] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: A survey," *Found. Trends Hum.-Comput. Interact.*, vol. 1, no. 3, pp. 203–275, 2007.
[13] K. Morioka, J.-H. Lee, and H. Hashimoto, "Human-following mobile robot in a distributed intelligent sensor network," *IEEE Trans. Ind. Electron.*, vol. 51, no. 1, pp. 229–237, Feb. 2004.
[14] A. Franchi, P. R. Giordano, C. Secchi, H. I. Son, and H. H. Bulthoff, "A passivity-based decentralized approach for the bilateral teleoperation of a group of UAVs with switching topology," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 898–905.
[15] B. Sadrfaridpour and Y. Wang, "Collaborative assembly in hybrid manufacturing cells: An integrated framework for human–robot interaction," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 1178–1192, Jul. 2018.
[16] H. Fang, C. Shang, and J. Chen, "An optimization-based shared control framework with applications in multi-robot systems," *Sci. China Inf. Sci.*, vol. 61, no. 1, pp. 201–203, Jan. 2018.
[17] S. Scheggi, F. Morbidi, and D. Prattichizzo, "Human-robot formation control via visual and vibrotactile haptic feedback," *IEEE Trans. Haptics*, vol. 7, no. 4, pp. 499–511, Oct. 2014.
[18] P. Walker, S. A. Amraii, N. Chakraborty, M. Lewis, and K. Sycara, "Human control of robot swarms with dynamic leaders," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 1108–1113.
[19] D. Herrera, F. Roberti, M. Toibero, and R. Carelli, "Human interaction dynamics for its use in mobile robotics: Impedance control for leader-follower formation," *IEEE/CAA J. Automatica Sinica*, vol. 4, no. 4, pp. 696–703, 2017.
[20] J. Huang, W. Wu, Y. Ning, N. Zhou, and Z. Xu, "A behavior control scheme for multi-robot systems under human intervention," in *Proc. Chin. Control Conf. (CCC)*, Jul. 2019, pp. 6189–6193.
[21] S. Musić and S. Hirche, "Control sharing in human-robot team interaction," *Annu. Rev. Control*, vol. 44, pp. 342–354, 2017.
[22] A. Pentland and A. Liu, "Modeling and prediction of human Behavior," *Neural Comput.*, vol. 11, no. 1, pp. 229–242, Jan. 1999.
[23] J. Gao and J. D. Lee, "Extending the decision field theory to model operators' reliance on automation in supervisory control situations," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 36, no. 5, pp. 943–959, Sep. 2006.
[24] R. Ratcliff and G. McKoon, "The diffusion decision model: Theory and data for two-choice decision tasks," *Neural Comput.*, vol. 20, no. 4, pp. 873–922, Apr. 2008.
[25] R. Ratcliff, P. Gomez, and G. McKoon, "A diffusion model account of the lexical decision task," *Psychol. Rev.*, vol. 111, no. 1, pp. 159–170, 2004.
[26] A. Stewart, M. Cao, A. Nedic, D. Tomlin, and N. Leonard, "Towards human–robot teams: Model-based analysis of human decision making in two-alternative choice tasks with social feedback," *Proc. IEEE*, vol. 100, no. 3, pp. 751–775, Mar. 2012.
[27] G. Antonelli, F. Arrichiello, and S. Chiaverini, "The null-space-based behavioral control for mobile robots," in *Proc. Int. Symp. Comput. Intell. Robot. Autom.*, Jun. 2005, pp. 15–20.
[28] F. Arrichiello, S. Chiaverini, G. Indiveri, and P. Pedone, "The null-space-based behavioral control for mobile robots with velocity actuator saturations," *Int. J. Robot. Res.*, vol. 29, no. 10, pp. 1317–1337, Sep. 2010.
[29] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Experiments of formation control with multirobot systems using the null-space-based behavioral control," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 5, pp. 1173–1182, Sep. 2009.
[30] J. R. Peters, V. Srivastava, G. S. Taylor, A. Surana, M. P. Eckstein, and F. Bullo, "Human supervisory control of robotic teams: Integrating cognitive modeling with engineering design," *IEEE Control Syst. Mag.*, vol. 35, no. 6, pp. 57–80, Nov. 2015.
[31] R. Bogacz, E. Brown, J. Moehlis, P. Holmes, and J. D. Cohen, "The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced-choice tasks.," *Psychol. Rev.*, vol. 113, no. 4, pp. 700–712, 2006.
[32] W. Edwards, "Optimal strategies for seeking information: Models for statistics, choice reaction times, and human information processing," *J. Math. Psychol.*, vol. 2, no. 2, pp. 312–329, Jul. 1965.
[33] K. Baizid, G. Giglio, F. Pierri, M. A. Trujillo, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero, "Behavioral control of unmanned aerial vehicle manipulator systems," *Auto. Robots*, vol. 41, no. 5, pp. 1203–1220, 2017.
[34] M. H. Mabrouk and C. R. McInnes, "Solving the potential field local minimum problem using internal agent states," *Robot. Auto. Syst.*, vol. 56, no. 12, pp. 1050–1060, Dec. 2008.
[35] G. Antonelli, "Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems," *IEEE Trans. Robot.*, vol. 25, no. 5, pp. 985–994, Oct. 2009.

JIE HUANG (Member, IEEE) received the B.E. degree in electrical engineering and automation and the M.E. degree in control engineering from Fuzhou University, China, in 2005 and 2010, respectively, and the Ph.D. degree in control science and engineering from the Beijing Institute of Technology, Beijing, China, in 2015. From 2005 to 2015, he was a Lecturer with the Fujian Institute of Education, Fuzhou, China. From 2014 to 2018, he held postdoctoral position and a lecturer appointment with the Faculty of Science and Engineering, University of Groningen, The Netherlands. He is currently a Full Professor of robotic and control with the College of Electrical Engineering and Automation, Fuzhou University. His research interests include autonomous robots, complex network dynamics, and multi-agent systems. He is the Vice-President of the Fujian Automation Association, Fujian, China.

WENHUA WU received the B.E. degree in automation from the Wuhan University of Science and Technology, China, in 2017. He is currently pursuing the M.E. degree with Fuzhou University, China. His research interests include human–robot interaction and multi-agent systems.

ZHENYI ZHANG (Student Member, IEEE) received the B.E. degree in mechanical and electronic engineering and the M.E. degree in mechanical engineering from Zhejiang Sci-Tech University, Hangzhou, China, in 2016 and 2019, respectively. He is currently pursuing the Ph.D. degree with Fuzhou University, China. His research interests include intelligent robot ethology and multi-agent systems.

YUTAO CHEN received the B.E. degree in automation from Hunan University, China, in 2012, the Ph.D. degree from the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, China, in 2014, and the Ph.D. degree from the Department of Information Engineering, University of Padova, Italy, in 2018. From 2019 to 2020, he was a Postdoctoral Researcher with the Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands. He is currently an Assistant Professor with the College of Electrical Engineering and Automation, Fuzhou University, China. His research interests include model predictive control algorithms, unmanned intelligent systems, and applications.