SCIENCE CHINA
Information Sciences

• Supplementary File •

Robust adaptive time-varying region tracking control of multi-robot systems

Junjie Fu*, Yuezu Lv & Wenwu Yu

Jiangsu Key Laboratory of Networked Collective Intelligence, School of Mathematics, Southeast University, Nanjing 211189, *China*

Appendix A Examples of transformation matrices

The rotation and scaling transformation matrices which in 2D are given by

$$\bar{A}_1 = \begin{bmatrix} \cos\phi(t) & -\sin\phi(t) \\ \sin\phi(t) & \cos\phi(t) \end{bmatrix}, \bar{A}_2 = \begin{bmatrix} s_x(t) & 0 \\ 0 & s_y(t) \end{bmatrix},$$

respectively, where $\phi(t)$ is the angle of rotation, $s_x(t)$ and $s_y(t)$ are the scaling factors. A composite transformation can be obtained by multiplying every transformation matrix together. For example, a matrix for scaling and then rotation is given as $\bar{A}_3 = \bar{A}_1\bar{A}_2$.

Appendix B Proof of Theorem 1

Proof. From the definition of the region tracking potential function

$$P_G^i = \sum_{l=1}^{M} P_{Gl}^i = \sum_{l=1}^{M} k_g \ln\cosh\{\max[0, f_l(T(x_i) - c_l)]\}, \tag{B1}$$

the region tracking force $\Delta\xi_i$ takes the form of

$$\Delta\xi_i = \left(\frac{\partial P_G^i}{\partial T(x_i)}\right)^T = \sum_{l=1}^{M} k_g \max\{0, \tanh[f_l(T(x_i) - c_l)]\} \left(\frac{\partial f_l(T(x_i) - c_l)}{\partial T(x_i)}\right)^T.$$

Noting the collision avoidance potential function

$$U_R^i = \sum_{j\in\mathcal{N}_i^c} U_{Rij}, \tag{B2}$$

the collision avoidance force $\Delta\rho_i$ can be expressed as:

$$\begin{aligned}\Delta\rho_i =& \left(\frac{\partial U_R^i}{\partial\left(T(x_i) - T(x_j)\right)}\right)^T \\ =& \sum_{j\in\mathcal{N}_i^c} \frac{4k_p\left(R^2 - r^2\right)\min\left(0, \|T(x_i) - T(x_j)\|^2 - R^2\right)}{\left(\|T(x_i) - T(x_j)\|^2 - r^2\right)^3}\left(T(x_i) - T(x_j)\right) \\ =& \sum_{j\in\mathcal{N}_i^c} F_{ij}.\end{aligned}$$

For the sliding variable s_i, $i = 1, 2, \ldots, N$, it holds $s_i = \dot{x}_i - \dot{x}_{ri} = \Delta\dot{x}_i + A^{-1}\dot{A}\Delta x_i + A^{-1}\Delta\zeta_i = A^{-1}\left(\frac{d}{dt}(A\Delta x_i) + \Delta\zeta_i\right)$. Furthermore, from the agent dynamics (1), it is easy to obtain

$$M_i(x_i)\dot{s}_i + C_i(x_i, \dot{x}_i)s_i + D_i(x_i)s_i + M_i(x_i)\ddot{x}_{ri} + C_i(x_i, \dot{x}_i)\dot{x}_{ri} + D_i(x_i)\dot{x}_{ri} + g_i(x_i) + \tau_{di} = \tau_i. \tag{B3}$$

Substituting the proposed controller (3) into (B3), the closed-loop system can be obtained as

$$M_i(x_i)\dot{s}_i + C_i(x_i, \dot{x}_i)s_i + D_i(x_i)s_i = -k_{s_i}s_i - A^T\Delta\zeta_i + Y_i(x_i, \dot{x}_i, \dot{x}_{ri}, \ddot{x}_{ri})\tilde{\theta}_i + v_i - \tau_{di} \tag{B4}$$

where $\tilde{\theta}_i = \hat{\theta}_i - \theta_i$.

* Corresponding author (email: fujunjie@seu.edu.cn)

Consider the following Lyapunov function

$$V = \sum_{i=1}^{N} \frac{1}{2} s_i^T M_i(x_i) s_i + \sum_{i=1}^{N} \frac{1}{2k_{\theta_i}} \tilde{\theta}_i^T \tilde{\theta} + \sum_{i=1}^{N} \alpha_i P_G^i + \frac{1}{2} \sum_{i=1}^{N} U_R^i + \sum_{i=1}^{N} \frac{1}{2k_{\beta_i}} \tilde{\beta}_i^2 + \sum_{i=1}^{N} \frac{\delta_i}{k_{\delta_i}} \tag{B5}$$

where P_G^i is given in (B1), U_R^i is given in (B2) and $\tilde{\beta}_i = \hat{\beta}_i - \beta_i$.

The derivative of V along the solutions of (B4) is given as

$$\begin{aligned}
\dot{V} =& \frac{1}{2} \sum_{i=1}^{N} s_i^T \dot{M}_i(x_i) s_i + \sum_{i=1}^{N} s_i^T M_i(x_i) \dot{s}_i - \sum_{i=1}^{N} \delta_i(t) - \sum_{i=1}^{N} \tilde{\theta}_i^T Y_i^T s_i + \sum_{i=1}^{N} \alpha_i \frac{\partial P_G^i}{\partial T(x_i)} \dot{T}(x_i) + \sum_{i=1}^{N} \tilde{\beta}_i \|s_i\| \\
&+ \frac{1}{2} \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^c} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i) - T(x_j)\right) \\
=& - \sum_{i=1}^{N} k_{s_i} s_i^T s_i - \sum_{i=1}^{N} s_i^T D_i(x_i) s_i - \sum_{i=1}^{N} s_i^T A^T \Delta\zeta_i + \sum_{i=1}^{N} \alpha_i \Delta\xi_i^T \frac{d}{dt} \left(A \Delta x_i\right) + \sum_{i=1}^{N} s_i^T \left(v_i - \tau_{di}\right) \\
&+ \frac{1}{2} \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^c} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i) - T(x_j)\right) + \sum_{i=1}^{N} \tilde{\beta}_i \|s_i\| - \sum_{i=1}^{N} \delta_i(t),
\end{aligned} \tag{B6}$$

where we have used the fact that $\dot{M}_i(x_i) - 2C_i(x_i, \dot{x}_i)$ is skew symmetric.

Denote $a_{ij} = 1$ for $j \in \mathcal{N}_i^c$ and $a_{ij} = 0$ otherwise, then we have $a_{ij} = a_{ji}$ and it holds

$$\begin{aligned}
&\sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i^c} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i) - T(x_j)\right) \\
=& \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i)\right) - \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_j)\right) \\
=& \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i)\right) - \sum_{j=1}^{N} \sum_{i=1}^{N} a_{ji} \frac{\partial U_{Rji}}{\partial \left(T(x_j) - T(x_i)\right)} \frac{d}{dt} \left(T(x_i)\right) \\
=& \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i)\right) + \sum_{j=1}^{N} \sum_{i=1}^{N} a_{ij} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(T(x_i)\right) \\
=& 2 \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} \frac{d}{dt} \left(A \Delta x_i\right)
\end{aligned}$$

where we have used the fact $\frac{\partial U_{Rij}}{\partial \left(T(x_i) - T(x_j)\right)} = - \frac{\partial U_{Rji}}{\partial \left(T(x_j) - T(x_i)\right)}$.

Moreover, it holds

$$\begin{aligned}
&\sum_{i=1}^{N} s_i^T \left(v_i - \tau_{di}\right) + \sum_{i=1}^{N} \tilde{\beta}_i \|s_i\| - \sum_{i=1}^{N} \delta_i(t) \\
\leqslant& - \sum_{i=1}^{N} \left[\frac{\hat{\beta}_i^2 \|s_i\|^2}{\hat{\beta}_i \|s_i\| + \delta_i(t)} + \beta_i \|s_i\| + \tilde{\beta}_i \|s_i\| - \delta_i(t) \right] \\
=& \sum_{i=1}^{N} \frac{\delta_i(t) \hat{\beta}_i \|s_i\|}{\hat{\beta}_i \|s_i\| + \delta_i(t)} - \sum_{i=1}^{N} \delta_i(t) \\
\leqslant& 0.
\end{aligned}$$

Then, from (B6) we have

$$\dot{V} \leqslant - \sum_{i=1}^{N} k_{s_i} s_i^T s_i - \sum_{i=1}^{N} s_i^T D_i(x_i) s_i - \sum_{i=1}^{N} s_i^T A^T \Delta\zeta_i + \sum_{i=1}^{N} \alpha_i \Delta\xi_i^T \frac{d}{dt} \left(A \Delta x_i\right) + \sum_{i=1}^{N} \Delta\rho_i^T \frac{d}{dt} \left(A \Delta x_i\right).$$

By noticing the definition of $\Delta\rho_i$ and the fact that

$$A s_i = \frac{d}{dt} (A \Delta x_i) + \Delta\zeta_i,$$

we can obtain

$$\begin{aligned}
\dot{V} \leqslant& - \sum_{i=1}^{N} k_{s_i} s_i^T s_i - \sum_{i=1}^{N} s_i^T D_i(x_i) s_i - \sum_{i=1}^{N} \Delta\zeta_i^T \Delta\zeta_i \\
\leqslant& 0.
\end{aligned}$$

It ensues that V is bounded. Then, Barbalat's Lemma can be used to show that $s_i \to 0$ and $\Delta\zeta_i \to 0$.

Noting that

$$\begin{aligned}
s_i =& A^{-1} \left(\frac{d}{dt} (A \Delta x_i) + \Delta\zeta_i \right) \\
=& A^{-1} (\dot{T}(x_i(t)) + \Delta\zeta_i),
\end{aligned}$$

it follows that $\dot{T}(x_i(t)) \to 0$ and $T(x_i(t))$ converge to some constant vectors where $i = 1, \ldots, N$.

Appendix C Simulation results

For the considered multi-robot system, we have

$$M_i \dot{s}_i + \eta_i s_i + Y_i \theta_i + \tau_{di} = \tau_i$$

where $Y_i = [\ddot{x}_{ri}, \dot{x}_{ri}]$ and $\theta_i = [M_i, \eta_i]^T$.

For the 2-D case, the basic region shape is specified as a circle with radius $D = 4m$ which can be described by

$$f(x) = x_x^2 + x_y^2 - D^2 \leqslant 0$$

where $x = [x_x, x_y]^T$. The time-varying region is obtained from $f\left[A(t)(x - x_o(t))\right] \leqslant 0$ where

$$A(t) = \begin{bmatrix} 1/s_x(t) & 0 \\ 0 & 1/s_y(t) \end{bmatrix} \begin{bmatrix} \cos\phi(t) & \sin\phi(t) \\ -\sin\phi(t) & \cos\phi(t) \end{bmatrix}$$

with the translation $x_o(t) = [t - 4, 0]^T$. The time-varying scaling functions and the rotation angle are set as

$$\begin{array}{ll} t \leqslant T_1: & s_x(t) = 1 + t/T; s_y(t) = 1 - t/(2T_1); \phi(t) = 0 \\ T_1 < t \leqslant T_2: & s_x(t) = 2; s_y(t) = 0.5; \phi(t) = \frac{\pi(t - T_1)}{2(T_2 - T_1)} \\ T_2 < t: & s_x(t) = 2; s_y(t) = 0.5; \phi(t) = \pi/2 \end{array}$$

with $T_1 = 15s$, $T_2 = 30s$. The center of this time-varying desired region is $x_o(t)$ and the scaling and rotation of the desired region are commanded during the motion.

The initial positions of the agents are distributed on a grid centered at $[-4, 0]$ as shown in Fig. C1. The initial velocities are set as zero. The region tracking potential function parameter $k_g = 0.2$. The collision avoidance potential function parameter $k_p = 0.5$. The sensing range $R = 1$ and the minimum distance parameter $r = 0.4$. The controller (3) is designed with parameters $k_{s_i} = 1, k_{\beta_i} = 0.1, k_{\delta_i} = 0.5, k_{\theta_i} = 0.1$ and $\alpha_i = 1$. The initial states of $\hat{\beta}_i$ and δ_i are set to 0 and 10, respectively.

The time-varying region tracking result is shown in Fig. C1 where the snap shots at several time instants are shown. It can be seen that all the robots track the desired time-varying region successfully. The time evolution of the variables $\Delta\zeta_i$ is plotted in Fig. C2. It can be seen that $\Delta\zeta_i$ approaches to zero. The control inputs are shown in Fig. C3. The estimation of the mass and damping parameter of the robots are shown in Figs. C4 and C5, which converge asymptotically to some finite steady state values. The estimation of the upperbound of the input disturbance is shown in Fig. C6, demonstrating that each adaptive gain $\hat{\beta}_i$ also converges to a finite steady state value.

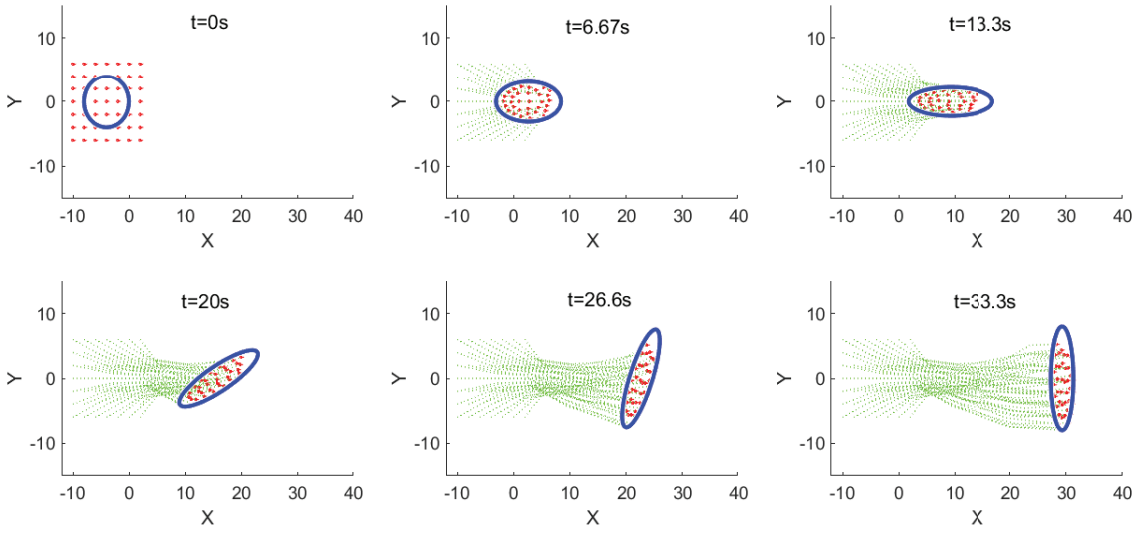

Figure C1 Trajectories of the robots. The blue solid line denotes the boundary of the desired time-varying region. The red circles represent the robots. The dotted green lines denote the trajectories of the robots.

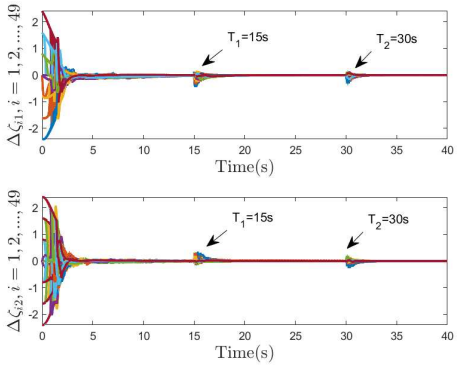

Figure C2 Trajectories of $\Delta\zeta_i$ which converge to zero asymptotically.

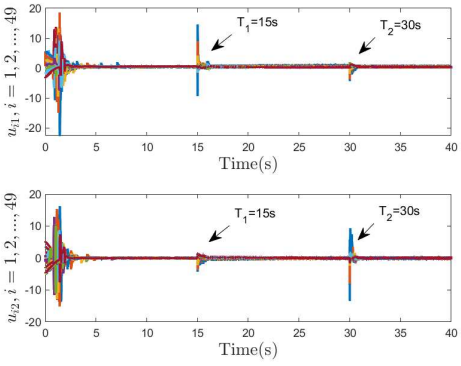

Figure C3 Trajectories of the control inputs which are continuous.

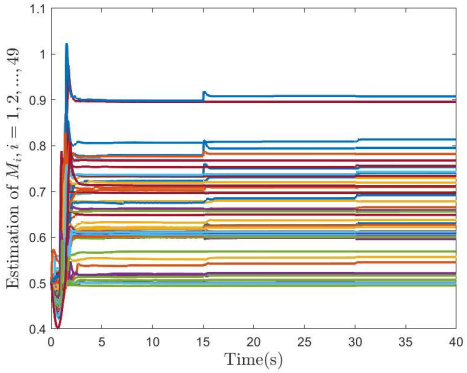

Figure C4 Estimation of the mass where the initial estimation is 0.5 and the true value is 1.

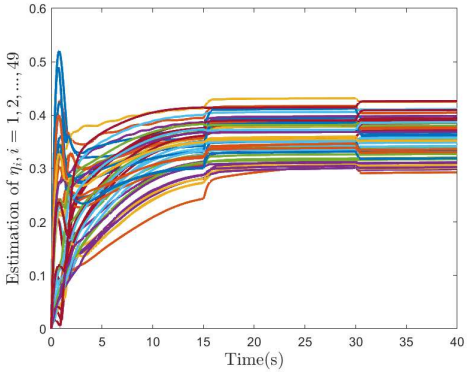

Figure C5 Estimation of the damping parameter where the initial estimation is 0 and the true value is 0.5.

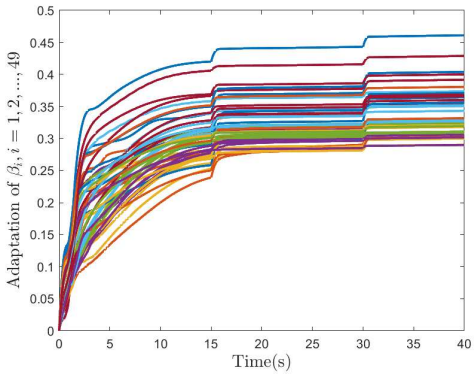

Figure C6 Adaptation of β_i where the initial estimation is 0.

A comparison with a controller designed with a conventional region tracking potential function such as in [1–3] is performed to show the advantage of reduced control input with the proposed region tracking potential function (B1). The following quadratic potential function

$$P_G^i = \sum_{l=1}^{M} k_g \left(\max\left[0, f_l(T(x_i) - c_l)\right]\right)^2 \tag{C1}$$

is used to obtain the region tracking force $\Delta\xi_i$. A smaller potential function gain $k_g = 0.02$ is used to mitigate the large control inputs while all the other parameters and initial conditions remain the same as in the previous example. The time-varying region tracking result is shown in Fig. C7 which shows that time-varying region tracking can still be achieved. The time evolution of the variables $\Delta\zeta_i$ is plotted in Fig. C8. It can be seen that $\Delta\zeta_i$ approaches to zero. The estimations of the mass and damping parameters of the robots are shown in Figs. C10 and C11 which converge asymptotically to some finite steady state values. The estimation of the upperbound of the input disturbance is shown in Fig. C12. However, comparing the controller input in this case as shown in Fig. C9 with that in Fig. C3, it can be seen that even with a smaller potential function gain, much larger control efforts are needed in the initial phase with the potential function (C1).

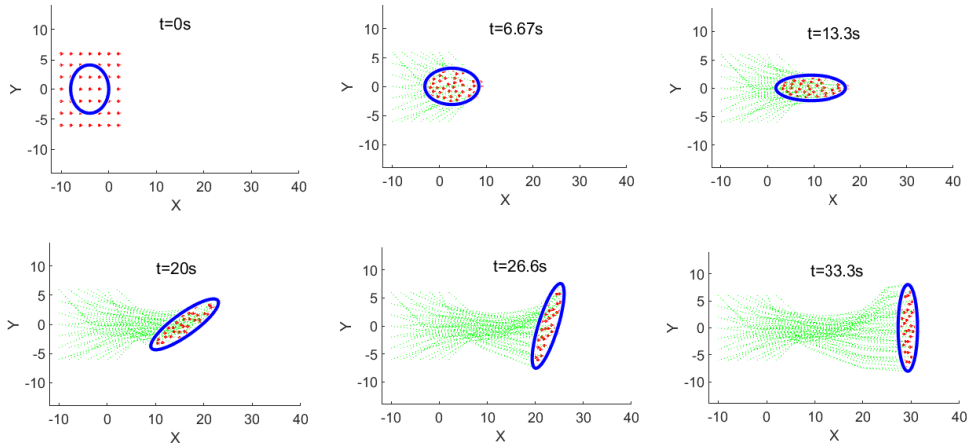

Figure C7 Trajectories of the robots with the conventional region tracking potential (C1). The blue solid line denotes the boundary of the desired time-varying region. The red circles represent the robots. The dotted green lines denote the trajectories of the robots.

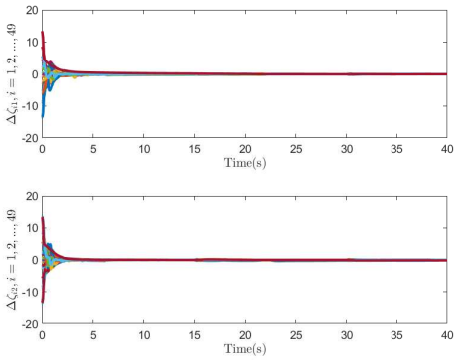

Figure C8 Trajectories of $\Delta\zeta_i$ with the conventional region tracking potential (C1).

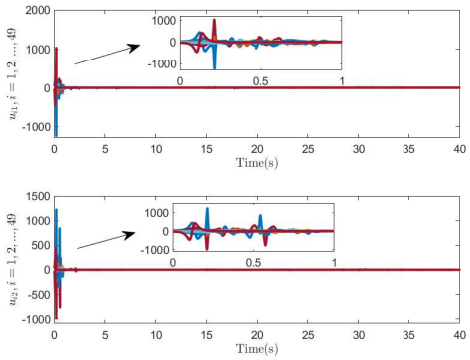

Figure C9 Trajectories of the control inputs with the conventional region tracking potential (C1).

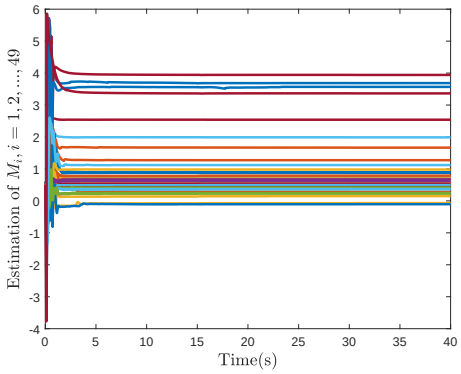

Figure C10 Estimation of the mass with the conventional region tracking potential (C1) where the initial estimation is 0.5 and the true value is 1.

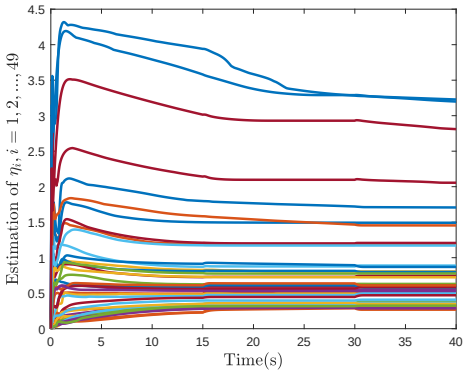

Figure C11 Estimation of the damping parameter with the conventional region tracking potential (C1) where the initial estimation is 0 and the true value is 0.5.

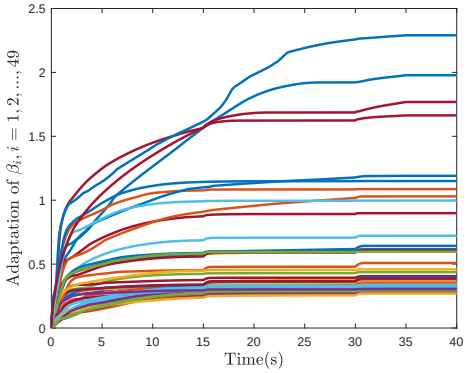

Figure C12 Adaptation of β_i with the conventional region tracking potential (C1) where the initial estimation is 0.

To show the convergence speed and the controller input magnitude trade-off in choosing the controller parameters for the proposed controller (3), we simulate the time-varying region tracking with a different set of parameters $k_{s_i} = 0.1, k_{\beta_i} = 0.05, k_{\delta_i} = 0.2, k_{\theta_i} = 0.05$. All the other parameters are the same as in the previous case. The region tracking results are shown in Figs. C13 and Fig. C14. Compared with Figs. C2 and C3 it can be seen that with smaller controller parameters, the control input gets smaller while the convergence speed becomes slower at the same time.

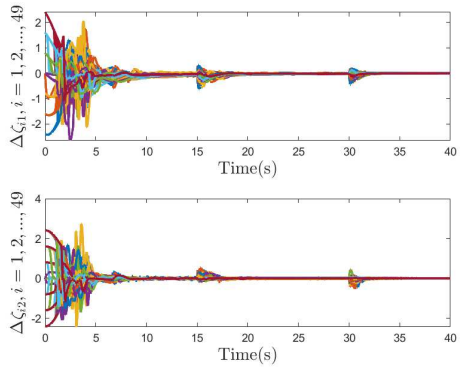

Figure C13 Trajectories of $\Delta\zeta_i$ which converge to zero asymptotically.

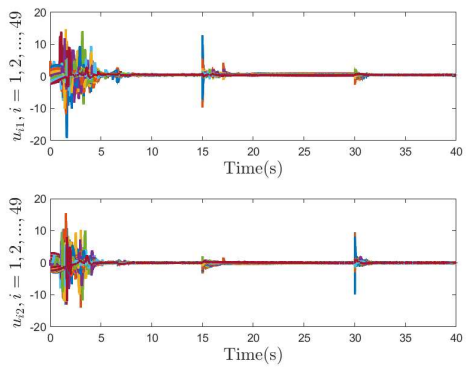

Figure C14 Trajectories of the control inputs which are continuous.

Furthermore, a 3-D case is also simulated to demonstrate the performance of the region tracking controller in high-dimensional

space. In this case, $x_i \in \mathbb{R}^3$ and $\tau_{di} = 0.1\sin(i \times t) * \mathbf{1}_3$ where $\mathbf{1}_3 = [1,1,1]^T$. The basic shape is now specified by a ball with radius $D = 4m$ which can be described by

$$f(x) = x_x^2 + x_y^2 + x_z^2 - D^2 \leqslant 0$$

where $x = [x_x, x_y, x_z]^T$. The time-varying region is obtained from $f[A(t)(x - x_o(t))] \leqslant 0$ where

$$A(t) = \begin{bmatrix} 1/s_x(t) & 0 & 0 \\ 0 & 1/s_y(t) & 0 \\ 0 & 0 & 1/s_y(t) \end{bmatrix} \begin{bmatrix} \cos\phi(t) & \sin\phi(t) & 0 \\ -\sin\phi(t) & \cos\phi(t) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

with the translation $x_o(t) = [t - 4, 0, 0]^T$. The time-varying scaling functions and the rotational angle are the same as in the 2D case. The initial positions of the robots are distributed on a grid centered at $[-4, 0, 0]^T$ as shown in Fig. C15. The initial velocities are set to zero. All the other controller parameters are the same as in the 2-D case.

The time-varying region tracking result is shown in Fig. C15 where the snap shots at several time instants are shown. It can be seen that all the robots track the desired time-varying region successfully. The time evolution of the variables $\Delta\zeta_i$ is plotted in Fig. C16. It can be seen that $\Delta\zeta_i$ approaches to zero. The control inputs are shown in Fig. C17. The estimation of the mass and damping parameters of the robots are shown in Figs. C18 and C19, which converge asymptotically to some finite steady state values. The estimation of the upperbound of the input disturbance is shown in Fig. C20, demonstrating that each adaptive gain $\hat{\beta}_i$ also converges to a finite steady state value.

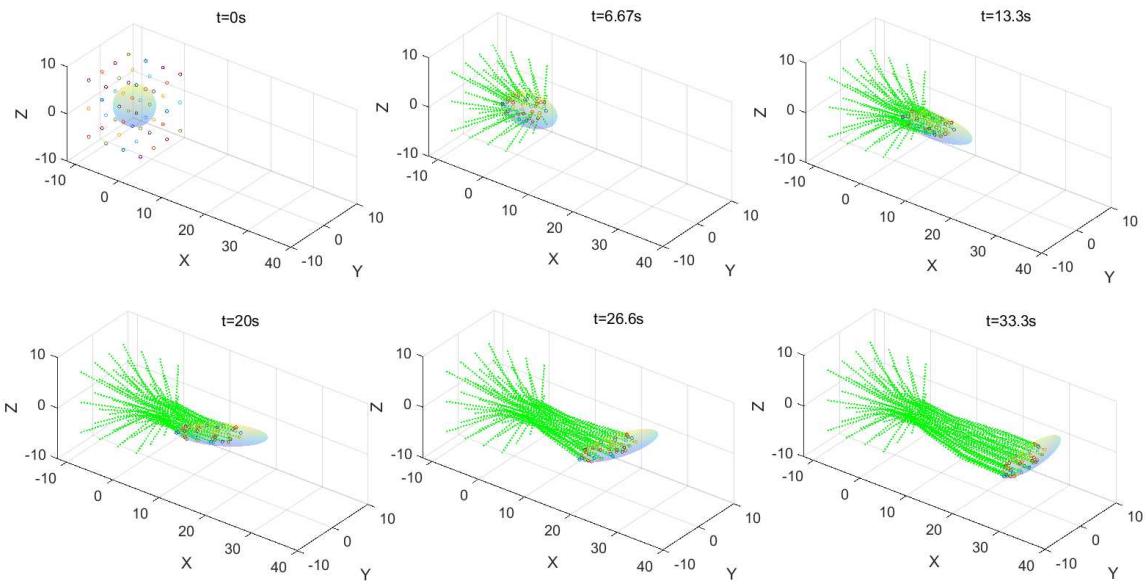

Figure C15 Trajectories of the robots in the 3-D case. The blue solid line denotes the boundary of the desired time-varying region. The red circles represent the robots. The dotted green lines denote the trajectories of the robots.

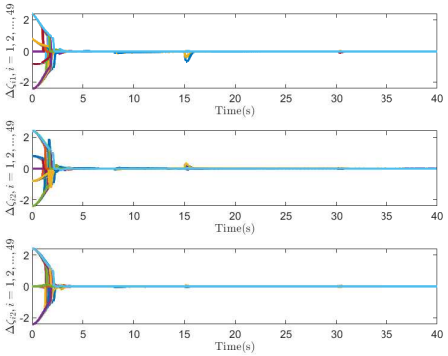

Figure C16 Trajectories of $\Delta\zeta_i$ in the 3-D case.

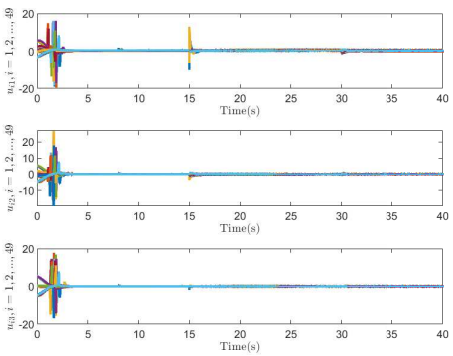

Figure C17 Trajectories of the control inputs in the 3-D case.

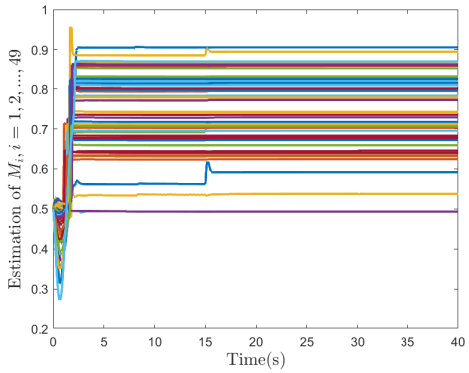

Figure C18 Estimation of the mass in the 3-D case where the initial estimation is 0.5 and the true value is 1.

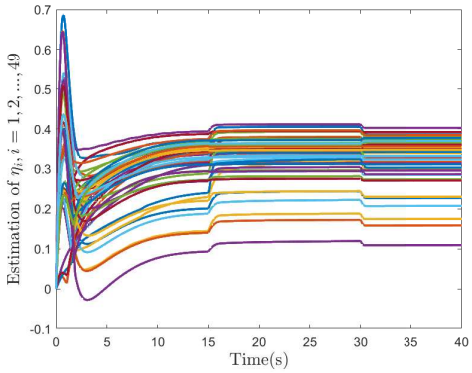

Figure C19 Estimation of the damping parameter in the 3-D case where the initial estimation is 0 and the true value is 0.5.

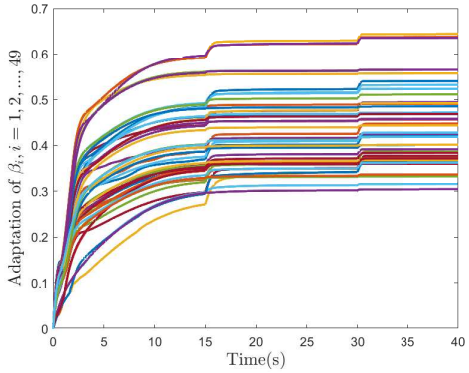

Figure C20 Adaptation of β_i in the 3-D case where the initial estimation is 0.

References

1 C. Cheah, S. Hou, J. Slotine. Region-based shape control for a swarm of robots. Automatica, 2009, 45: 2406–2411.
2 S. Hou, C. Cheah. Dynamic compound shape control of robot swarm. IET Control Theory and Applications, 2012, 6(3): 454–460.
3 X. Sun, S. Ge, J. Zhang, et al. Region tracking control for high-order multi-agent systems in restricted space. IET Control Theory and Applications, 2016, 10(4): 396–406.