Letter

Supplementary Material for "Dynamic Robust Pursuit of Multiple Evaders Under State Measurement Uncertainty in Obstacle Environments"

Kai Rao, Huaicheng Yan, Zhihao Huang, Tiantian Xu, Penghui Yang, and Yunkai Lv

S1. Overall Framework Diagram:

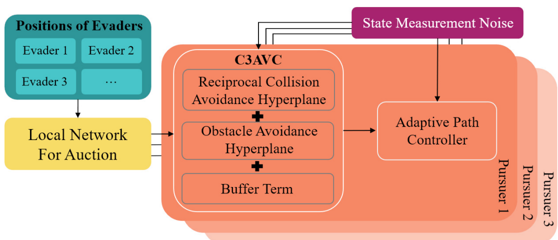

Fig. S1. The diagram of the framework architecture.

S2. Optimization of Obstacle Separation Hyperplane:

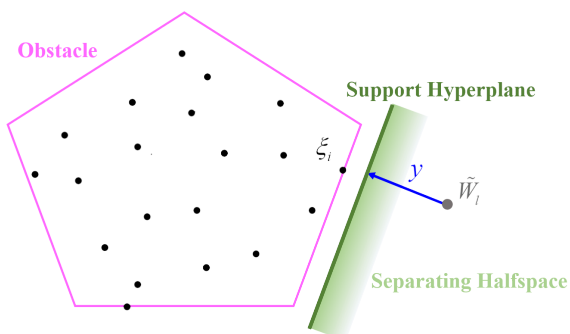

Fig. S2. Obstacle Separation Hyperplane.

The purpose of this step is to find an optimal separating hyperplane that can separate the obstacle from the robot. Here, we construct this optimization problem by using the collision vector. The collision vector refers to the shortest vector in the set of vectors obtained by pointing from the robot's position to the points on the obstacle, and the collision vector is the normal vector of its support hyperplane. Denote the collision vector as y and the position of the robot as $\tilde{W}_l$, then the separating halfspace can be written as:

$$\{W \in \mathbb{R}^d | y^T \left(W - \tilde{W}_l\right) \leq y^T y\}, \tag{1}$$

where d is the dimension of the halfspace. As shown in Fig. S2, since the obstacles are represented as a set of points $(\xi_1, \xi_2, \ldots, \xi_q)$, in order to obtain the largest possible feasible region, the solution for the normal vector of the obstacle separating hyperplane can be formulated as maximizing the square of the norm of the collision vector:

$$\begin{aligned} \max \quad & y^T y \\ s.t. \quad & (\xi_i - \tilde{W}_l)^T y \geq y^T y, \quad \forall i \in 1, \ldots, q, \end{aligned} \tag{2}$$

However, optimization problem (2) is non-convex. By defining $z = \frac{y}{y^T y}$, then the optimization problem (2) is transformed into

$$\begin{aligned} \min \quad & z^T z \\ s.t. \quad & (\xi_i - \tilde{W}_l)^T z \geq 1, \quad \forall i \in 1, \ldots, q, \end{aligned} \tag{3}$$

which is convex and can be solved with linear complexity[1]. Finally, for the pursuer l and the obstacle O_k, the separating hyperplane can be expressed as $a_{lk}^T W \leq b_{lk}$, where $a_{lk} = y = \frac{z}{z^T z}$, and $b_{lk} = y^T(y + \tilde{W}_l)$.

S3. Probabilistic Collision Avoidance Proof: The C3AVC of pursuer l can ultimately be expressed as:

$$C_l = \{W \in E | a_{lg}^T W \leq b_{lg} - \varphi_l^\sigma - \varphi_l^f, \forall g \in \mathcal{G}, g \neq l\}, \tag{4}$$

where $\mathcal{G} \triangleq \{1, \cdots, S, \cdots, S+K\}$, $g \neq l$ indicates all the robots and obstacles except pursuer l.

Before proceeding with the probability collision avoidance proof for C3AVC, it is necessary to introduce the following lemma:

Lemma 1 [1]: Given a multivariate random variable $z \sim \mathcal{N}(\tilde{z}, \Sigma_z)$, we have:

$$pb(a^T z \leq b) = \frac{1 + erf(\frac{b - a^T \tilde{z}}{\sqrt{2 a^T \Sigma_z a}})}{2}. \tag{5}$$

Theorem 1: During the pursuit of the evader, if $\tilde{W}_l \in C_l$ and $\tilde{W}_p \in C_p$, it can be guaranteed that the pursuers maintain probability reciprocal avoidance.

Proof: For pursuer l, based on (4) and Lemma 1, we have:

$$\begin{aligned} pb(a_{lp}^T W_l \leq b_{lp} - \varphi_l^f) &= \frac{1 + erf\left(\frac{b_{lp} - \varphi_l^f - a_{lp}^T \tilde{W}_l}{\sqrt{2 a_{lp}^T \Sigma_\Theta a_{lp}}}\right)}{2} \\ &\geq \frac{1 + erf\left(erf^{-1}(2\sqrt{1-\sigma} - 1)\right)}{2} \\ &= \sqrt{1-\sigma} \end{aligned} \tag{6}$$

Similarly for pursuer p, we also have:

$$pb(a_{pl}^T W_p \leq b_{pl} - \varphi_p^f) \geq \sqrt{1-\sigma}. \tag{7}$$

Based on (6) and (7) we get:

$$\begin{aligned} pb\left(dis\left(W_l, W_p\right) \geq 2R_f\right) \geq & pb(a_{lp}^T W_l \leq b_{lp} - \varphi_l^f) \\ & \cdot pb(a_{pl}^T W_p \leq b_{pl} - \varphi_p^f) \\ \geq & 1 - \sigma. \end{aligned} \tag{8}$$

Thus, (1a) of the main text is guaranteed. This completes the proof. ■

Theorem 2: During the pursuit of the evader, if $\tilde{W}_l \in C_l$, it can be guaranteed that the pursuers achieve probability obstacle avoidance.

Proof: We can get the following inequality for pursuer l similar to (6):

$$pb(a_{lk}^T W_l \leq b_{lk} - \varphi_l^f) \geq \sqrt{1-\sigma}. \tag{9}$$

Inequality (9) indicates that when $\tilde{W}_l \in C_l$, the probability of collision with the obstacle O_k is less than $\sqrt{1-\sigma}$. Since $0 \leq \sigma \leq 1$, the following inequality is satisfied:

$$pb(a_{lk}^T W_l \leq b_{lk} - \varphi_l^f) \geq \sqrt{1-\sigma} \geq 1 - \sigma. \tag{10}$$

Thus, (1c) of the main text is guaranteed. This completes the proof. ■

When evaders are within their C3AVCs, the proofs for (1b) and (1d) of the main text can be established using the same line of reasoning as for (1a) and (1b) of the main text.

S4. Computational Complexity of Constructing C3AVC: The C3AVC is composed of separating hyperplanes between robots as well as between robots and obstacles. For the separating hyperplanes among robots, since variance of the state measurement uncertainty of the Gaussian distribution is equal for each robot, it can be solved

[1] https://github.com/ZJU-FAST-Lab/SDQP

in closed form with a complexity of $O(S)$ [2] considering the other $S-1$ robots in the environment. Moreover, for the separating planes between the pursuer and all obstacles, benefiting from the low-dimensional nature of the following optimization problem:

$$\begin{aligned} &\min \quad z^T z \\ &s.t. \quad (\xi_i - \tilde{W}_l)^T z \geq 1, \quad \forall i \in 1, \ldots, q, \end{aligned} \tag{11}$$

it can be solved with the complexity of $O(K)$ considering K obstacles in the environment. Therefore, computing C3AVC in total requires $O(S+K)$ time.

S5. Path Controller Optimization: As introduced in the manuscript, our objective is to find a feedback path controller $u = \kappa d = \kappa D^{\vee} + \kappa_w \tilde{W}_l$ that can compute the control input based on the deviation $d = (D - \tilde{W}_l \mathbf{1}^T)^{\vee}$ between the current mean position of pursuer l and the vertex position of the C3AVC, where $\kappa_w = -\kappa(\mathbf{1}_{nl} \otimes I_2)$, $(\cdot)^{\vee}$ is the vectorized version.

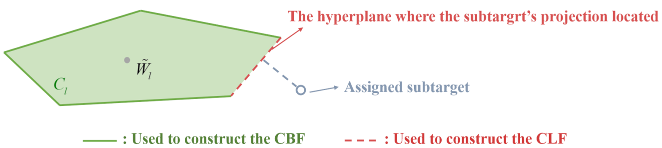

Fig. S3. Construction of CBF and CLF.

Considering the characteristic that the C3AVC is composed of the intersection of a series of hyperplane, as illustrated in Fig. S3, the CBF is chosen as multiple affine barrier functions $h_i(x) = A_{xi}x + b_{xi}$, where A_{xi} and b_{xi} are the union of rows and elements of the hyperplane coefficient matrices constituting C_l except for the hyperplane where the subtarget projection is located. In this letter, we construct the CLF $V(x) = s^T x + b_s$ based on the hyperplane where the subtargrt's projection on the C3AVC is located. Considering the actuator constraints of practical physical systems, we assume that the control input u is bounded as $A_u u \leq b_u$. Then the optimization problem can be formulated as:

$$\begin{aligned} &find \quad \kappa \\ &s.t. \quad L_f h_i(x) + L_g h_i(x) u + c_b h_i(x) \geq 0, \\ &\qquad L_f V(x) + L_g V(x) u + c_l V(x) \leq 0, \\ &\qquad A_u u \leq b_u, \end{aligned} \tag{12}$$

where L_f and L_g are Lie derivatives, c_b and c_l are positive scalars since h_i and V all have relative degree $r = 1$.

In practice, we aim to find a controller that satisfies the constraints in (12) with some margin. By introducing slack variables Ω_b and Ω_l [3], we have:

$$\begin{aligned} &\min_{\kappa, \Omega_b, \Omega_l} \quad \lambda_b^T \Omega_b + \lambda_l \Omega_l \\ &s.t. \quad -(L_f h_i(x) + L_g h_i(x) u + c_b h_i(x)) \leq \Omega_{bi}, \\ &\qquad L_f V(x) + L_g V(x) u + c_l V(x) \leq \Omega_l, \\ &\qquad A_u u \leq b_u, \end{aligned} \tag{13}$$

where λ_b and λ_l are user-defined constants defining the trade-off between the CBF and CLF constraints, and Ω_{bi} is the element of Ω_b.

Then, when the mean position of the pursuer l is $\tilde{W}_l$, by substituting $h_i(x) = A_{xi}x + b_{xi}$ and $V(x) = s^T x + b_s$, the optimization problem (13) can be formulated as :

$$\begin{aligned} &\min_{\kappa, \Omega_b, \Omega_l} \quad \lambda_b^T \Omega_b + \lambda_l \Omega_l \\ &s.t. \quad -(A_{xi}\kappa_w + c_b A_{xi}) \tilde{W}_l \leq \Omega_{bi} + c_b b_{xi} + A_{xi}\kappa D^{\vee}, \\ &\qquad \left(s^T \kappa_w + c_l s^T\right) \tilde{W}_l \leq \Omega_l - c_l b_s - s^T \kappa D^{\vee}, \\ &\qquad A_u \kappa_w \tilde{W}_l \leq b_u - A_u \kappa D^{\vee}, \\ &\qquad \Omega_b, \Omega_l \leq 0, \end{aligned} \tag{14}$$

which is a linear programming, and we use Mosek[2] for optimization problem solving.

S6. Screenshots of Pursuit Process:

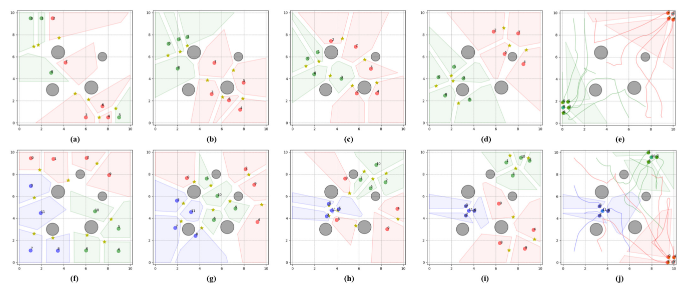

Fig. S4. Pursuit process. (a)-(e) Screenshots of the pursuit process in Scenario 1; (f)-(j) Screenshots of the pursuit process in Scenario 2.

References

[1] L. Blackmore, M. Ono and B. C. Williams, "Chance-constrained optimal path planning with obstacles," *IEEE Trans. Robot.*, vol. 27, no. 6, pp. 1080-1094, Dec. 2011.

[2] T. W. Anderson and R. R. Bahadur, "Classification into two multivariate normal distributions with different covariance matrices," *Ann. Math. Statist.*, vol. 33, pp. 420-431, 1962.

[3] M. Bahreinian, M. Kermanshah, and R. Tron, "Designing robust linear output feedback controller based on CLF-CBF framework via Linear-Programming(LP-CLF-CBF)," *IEEE Trans. Autom. Control.*, vol. 70, no. 4, pp. 2683-2689, Apr. 2025.

[2] https://www.mosek.com/