Ứng dụng Soft Actor - Critic cho điều hướng UAV trong môi trường 2D/3D

[1]ThS. Tạ Chí Hiếu*
[1]Đại học Thủy Lợi, Hà Nội
Email: *hieutc@tlu.edu.vn

[2]Phan Thị Phương Anh
[2]Đại học Thủy Lợi, Hà Nội
Email: 2251061711@e.tlu.edu.vn

[3]Nguyễn Anh Huy
[3]Đại học Thủy Lợi, Hà Nội
Email: 2251061796@e.tlu.edu.vn

Ngày nhận bài: 16/9/2025 Ngày chấp nhận đăng: 29/9/2025

Tóm tắt **- Bài báo trình bày nghiên cứu ứng dụng thuật toán Soft Actor - Critic (SAC) trong việc điều hướng máy bay không người lái (UAV) trong môi trường mô phỏng phức tạp 2D và 3D. SAC, một phương pháp học tăng cường ngoài chính sách (off -policy), được triển khai với kiến trúc mạng nơ - ron đa lớp để tối ưu chính sách ngẫu nhiên và hàm phần thưởng tổng thể, giúp UAV tự động điều chỉnh quỹ đạo, tránh vật cản và đạt mục tiêu một cách an toàn và hiệu quả. Kết quả thực nghiệm cho thấy SAC đạt tỉ lệ thành công cao, quỹ đạo mượt mà trong môi trường 2D và 3D, đồng thời vượt trội hơn các thuật toán phổ biến như PPO và A2C. Bài báo cũng đề xuất hướng phát triển mở rộng với việc áp dụng các thuật toán off-policy khác và bổ sung cảm biến ảnh cho điều hướng UAV trong môi trường thực tế.**

Từ khóa - Soft Actor-Critic (SAC), UAV navigation, Reinforcement learning, Autonomous control

I. GIỚI THIỆU CHUNG

Điều hướng máy bay không người lái (UAV) trong môi trường phức tạp là bài toán quan trọng đối với nhiều ứng dụng như giám sát, cứu hộ hay vận chuyển. UAV cần tự điều chỉnh quỹ đạo để tránh vật cản và đạt tới mục tiêu một cách an toàn. Bài toán này có thể được biểu diễn dưới dạng học tăng cường (reinforcement learning - RL) mà trong đó agent học chính sách điều khiển tối ưu qua tương tác với môi trường. Những phương pháp phổ biến như DQN, PPO hoặc A2C thường yêu cầu chia hành trình thành các đoạn huấn luyện tách biệt hoặc phải đặt lại chính sách ở mỗi giai đoạn, dẫn tới chính sách không liên tục và thiếu tính kế thừa.

Soft Actor - Critic (SAC) là một thuật toán off - policy hiện đại tối ưu chính sách ngẫu nhiên trong không gian hành động liên tục. Nó kết hợp kỹ thuật clipped double - Q của DDPG nhằm giảm sai lệch trong ước lượng giá trị và bổ sung entropy regularization vào hàm mục tiêu [1]. Việc tối đa hóa entropy khuyến khích agent duy trì sự ngẫu nhiên trong chính sách để khám phá nhiều quỹ đạo và tránh hội tụ vào nghiệm kém [2]. SAC do đó phù hợp cho các bài toán điều hướng liên tục cần khả năng khám phá mạnh, và có thể áp dụng chung cho môi trường hai chiều (2D) và ba chiều (3D) mà không cần thay đổi kiến trúc mạng.

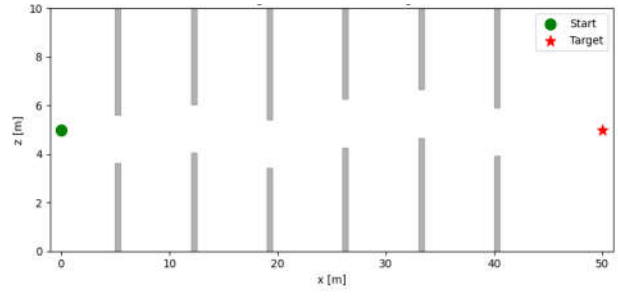
Báo cáo này tổng kết mô hình hóa điều hướng UAV 2D và 3D sử dụng cùng một thuật toán SAC. Nội dung bao gồm mô tả mô hình vật lý và môi trường, thiết kế hàm phần thưởng, mô tả thuật toán SAC và quy trình huấn luyện, cùng phân tích kết quả thực nghiệm ở cả hai môi trường.

II. PHƯƠNG PHÁP NGHIÊN CỨU

A. Mô hình vật lý UAV và môi trường mô phỏng

(1) Mô hình 2D: Trong môi trường hai chiều, UAV bay trên mặt phẳng x - z và chịu tác động của lực đẩy chính (thay đổi độ cao) và moment quay (điều chỉnh góc nghiêng). Trạng thái được biểu diễn bằng 6 biến $(x, z, \theta, \dot{x}, \dot{z}, \dot{\theta})$. Lực đẩy T được điều khiển bởi bộ điều khiển PID nhằm giữ UAV ở độ cao mong muốn, trong khi moment quay τ được ánh xạ tuyến tính từ hành động $a \in [-1,1]$. Mô hình động học sử dụng các phương trình cơ bản về lực và momen tương tự như trong [1].

Môi trường 2D có chiều dài cố định (ví dụ 100m) với các cột thẳng đứng sinh ngẫu nhiên tạo nên nhiều khe hẹp. Mỗi cột gồm hai phần: một phần phía dưới và một phần phía trên, cách nhau bởi khe rộng 2m; vị trí trung tâm khe được chọn ngẫu nhiên trong dải 4 - 6m theo trục z. Agent nhận vector quan sát gồm 9 thành phần: trạng thái hiện tại và khoảng cách tương đối tới mục tiêu và chướng ngại vật gần nhất. Hình minh họa một môi trường 2D điển hình.



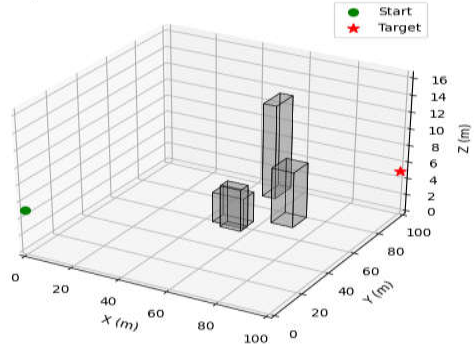
Hình 1. Môi trường UAV mô phỏng trong mô hình 2D với vật cản ngẫu nhiên.

(2) Mô hình 3D: Bước sang không gian ba chiều, UAV chuyển động trên không gian x, y, z . Trạng thái gồm vị trí (x, y, z) và vận tốc (v_x, v_y, v_z) . Lực điều khiển là vector 3 chiều $(u_x, u_y, u_z) \in [-1,1]^3$ được nhân với lực tối đa khác nhau ở trục ngang và trục dọc, sau đó trừ trọng lực cho thành phần z. Mô hình vật lý sử dụng hệ số cản khí động k_d, khối lượng 1kg, lực tối đa 8N theo trục x, y và 12N theo trục z, và giới hạn độ cao $[z_{min}, z_{max}]$.

Chướng ngại vật là các hộp chữ nhật (cuboid) sinh ngẫu nhiên trong không gian, với độ dài phương x và y được lấy ngẫu nhiên từ các khoảng khác nhau (5-15m, 20-40m hoặc 50-100m) và chiều cao lấy từ 3-12m, 15-50m hoặc 50-120m theo một phân bố tam giác. Số lượng

chướng ngại vật tăng lên theo khoảng cách tới mục tiêu; ví dụ 100m có 4 - 6 hộp, 200m có 6 - 9 hộp và 500m có 15-17 hộp. Các tham số trên được lựa chọn nhằm phản ánh mức độ phức tạp tăng dần theo khoảng cách bay. Số lượng và kích thước cuboid được điều chỉnh để tạo sử đa dạng môi trường nhưng vẫn đảm bảo UAV có thể tìm được quỹ đạo khả thi. Cách thiết kế này vừa đảm bảo môi trường đủ phức tạp để kiểm chứng, vừa duy trì khả năng học ổn định của mô hình.

Agent quan sát trạng thái hiện tại và vectơ khoảng cách tới mục tiêu (x_d, y_d, z_d). Hình 2 minh hoạ môi trường 3D điển hình.



Hình 2. Môi trường UAV mô phỏng trong mô hình 3D với vật cản ngẫu nhiên.

B. Hàm phần thưởng

Môi trường 2D - Trong môi trường 2D, hàm phần thưởng nhằm hướng dẫn UAV vừa tiến về phía mục tiêu vừa giữ ổn định độ cao và tránh va chạm. Tổng phần thưởng tại thời điểm t có dạng sau:

$$r_t = w_h\, r_h + w_p\, r_p + w_v\, r_v + w_z\, r_z + r_{\text{term}} \qquad (1)$$

Với r_h thưởng cho giữ độ cao đúng, r_p thưởng theo tiến độ dọc trục x, r_v phạt vận tốc quá lớn, r_z thưởng khi tới gần đích và r_{term} thưởng hoặc phạt ở cuối tập (thưởng lớn nếu chạm đích, phạt nặng nếu va chạm). Các trọng số w_h, w_p, w_v, w_z được điều chỉnh thực nghiệm. Khác với PPO, SAC áp dụng hàm thưởng này cho toàn bộ hành trình mà không chia đoạn huấn luyện.

Môi trường 3D - Hàm phần thưởng cho môi trường 3D phản ánh quãng đường 3 chiều, tránh va chạm, duy trì độ cao an toàn và hãm tốc khi gần đích. Tổng phần thưởng tại thời điểm t có dạng sau:

$$\begin{aligned} r_t = {} & c_1\,(d_{t-1} - d_t) - c_2\,\text{speed}_t\,\mathbb{1}_{(d_t<r_{\text{slow}})} \\ & + c_3\,\mathbb{1}_{(d_t<r_{\text{goal}}\wedge \text{speed}_t<v_{\text{goal}})} \\ & + c_4\,\mathbb{1}_{\text{collision}} - c_5\,\Delta h_t \\ & + c_6\,g(d_{\min}) + c_7 \end{aligned} \qquad (2)$$

Trong đó d_t là khoảng cách tới mục tiêu tại bước t, speed_t là tốc độ tổng hợp, $\mathbb{1}_{(\cdot)}$ là hàm chỉ báo, $d_{\min}$ là khoảng cách tới chướng ngại vật gần nhất, và $g(\cdot)$ là hàm thưởng nhỏ khi duy trì khoảng cách an toàn. Thành phần Δh_t phạt khi UAV vượt ra khỏi dải độ cao cho phép. Các hệ số c_i và ngưỡng $r_{\text{slow}}, r_{\text{goal}}, v_{\text{goal}}$ được lựa chọn qua thực nghiệm để cân bằng giữa tốc độ và an toàn. Phần thưởng đến đích được giảm dần theo số bước để khuyến khích hoàn thành nhanh.

C. Thuật toán SAC

Nghiên cứu sử dụng thuật toán Soft Actor - Critic (SAC), một phương pháp học tăng cường ngoài chính sách (off - policy) theo cấu trúc actor - critic, được triển khai thông qua thư viện Stable - Baselines3. SAC tối ưu chính sách bằng cách cực đại hóa tổng phần thưởng kỳ vọng cùng với entropy của chính sách, qua đó tăng cường khả năng khám phá và hạn chế hội tụ sớm.

Quá trình cập nhật dựa trên hai mạng nơ ron chính:

- Mạng Critic (Q - value): ước lượng giá trị hành động $Q(s,a)$ bằng cách tối thiểu hóa hàm mất mát Bellman:

$$\begin{aligned} L_Q = E_{(s,a,r,s')}\Big[\big(Q(s,a) \\ - (r + \gamma \\ \cdot E_{a'\sim\pi}[Q'(s',a') \\ - \alpha\, log\,\pi\,(a'|s')]\big)^2\Big] \end{aligned} \qquad (3)$$

Trong đó, Q' là mạng mục tiêu và α là hệ số entropy.

- Mạng Actor (Chính sách): Được cập nhật để tối đa hóa phần thưởng kỳ vọng và entropy:

$$J_\pi = E_{s\sim D}\big[E_{a\sim\pi}[\alpha\, log\,\pi\,(a|s) - Q(s,a)]\big] \qquad (4)$$

Mô hình được huấn luyện trong 1 triệu *timesteps*. Cả hai mạng Actor và Critic có cùng kiến trúc mạng MLP gồm hai lớp ẩn, mỗi lớp gồm 256 nơ ron, sử dụng hàm kích hoạt ReLU. Các siêu tham số chính bao gồm: tốc độ học 3×10^{-4}, hệ số chiết khấu γ=0.99, hệ số làm mượt mạng mục tiêu theo cơ chế Polyak τ=0.005, và kích thước lô (batch size) là 256. Ngoài ra, hệ số entropy α được điều chỉnh tự động, replay buffer có kích thước 10^6, số bước khởi động (learning starts) đặt là 5000, mỗi bước môi trường đi kèm một lần cập nhật gradient, và gradient được chuẩn hóa với $|g| \le 0.5$. Các mạng được tối ưu bằng Adam, với giá trị khởi tạo ngẫu nhiên mặc định của môi trường.

Trong quá trình huấn luyện, mô hình được đánh giá định kỳ sau mỗi 1.000 bước trên 100 tập kiểm thử với chướng ngại vật sinh ngẫu nhiên. Một tập được xem là thành công nếu UAV tiếp cận mục tiêu với sai số theo trục hoành nhỏ hơn 0.5 m. Ngược lại, tập bị xem là thất bại nếu UAV va chạm, vượt giới hạn an toàn về độ cao (z < 3.0 hoặc z > 8.0) hoặc góc nghiêng vượt quá ±90°. Mô hình có tổng phần thưởng trung bình cao nhất trong các lần đánh giá sẽ được chọn làm mô hình tốt nhất để sử dụng trong giai đoạn kiểm thử chính thức.

D. Quy trình huấn luyện

- Sinh môi trường: Với 2D, sinh ngẫu nhiên 6 - 10 cặp cột thẳng đứng tạo khe hẹp; với 3D, sinh ngẫu nhiên các cuboid với số lượng và kích thước theo khoảng cách tới mục tiêu. Mục tiêu được đặt tại tọa độ $(d, 0)$cho 2D hoặc $(d, d, 5)$ cho 3D.

- Khởi tạo SAC: Tạo hai mạng critic và một mạng actor với kiến trúc MLP, thiết lập buffer kinh nghiệm (1 triệu mẫu), và các siêu tham số như trên.

- Tương tác và cập nhật: Tại mỗi bước, agent quan sát trạng thái, lấy hành động từ actor, nhận phần thưởng và trạng thái kế tiếp, lưu vào buffer và cập nhật mạng theo thuật toán SAC.

Đánh giá định kỳ: Sau mỗi số bước cố định (ví dụ 1000 bước), dừng huấn luyện tạm thời và đánh giá mô hình trên 100 tập kiểm thử mới. Ghi lại tỉ lệ thành công, số bước trung bình, tỉ lệ va chạm và vi phạm an toàn. Mô hình có tổng phần thưởng trung bình cao nhất được lưu lại để đánh giá cuối.

III. KẾT QUẢ NGHIÊN CỨU

A. Kết quả định lượng

Trong môi trường 2D, mô hình SAC được huấn luyện trên 1 triệu bước với các tham số như trên và đánh giá trên 100 tập kiểm thử. Kết quả đạt tỉ lệ thành công 94,0 %, số bước trung bình 129,04 ± 28,16, tỉ lệ va chạm 4,0 % và tỉ lệ vi phạm an toàn 2,0 %. Bảng 1 tóm tắt kết quả đánh giá cho đoạn bay 100 m. So với PPO và A2C, SAC đạt tỉ lệ thành công cao hơn và số bước trung bình thấp hơn.

BẢNG 1. KẾT QUẢ ĐÁNH GIÁ SAC TRONG MÔI TRƯỜNG 2D

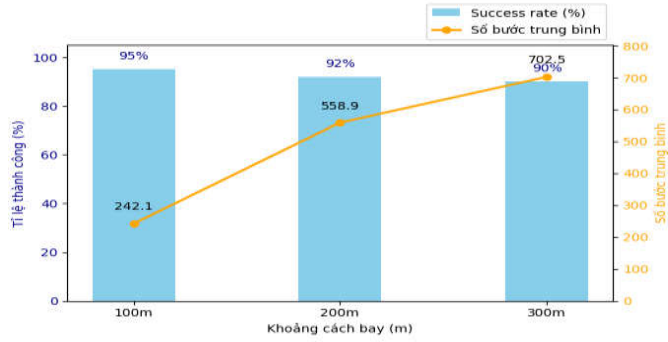
Đoạn bay	Tỉ lệ thành công [%]	Số bước trung bình	Tỉ lệ va chạm [%]	Tỉ lệ vi phạm an toàn [%]
0-100m	94.0	129.04 ± 28.16	4.0	2.0

Trong môi trường 3D, thuật toán SAC đạt hiệu quả điều hướng ổn định với tỉ lệ thành công cao trên các quãng đường 100m, 200m và 300m. Cụ thể, tỉ lệ thành công đạt 95,0% ở 100m, 92,0% ở 200m và 91,0% ở 300m. Các tỉ lệ va chạm, vi phạm độ cao và hết thời gian đều duy trì ở mức thấp, cho thấy SAC có khả năng tổng quát hóa tốt và đảm bảo an toàn trong không gian ba chiều phức tạp.

BẢNG 2. KẾT QUẢ ĐÁNH GIÁ SAC TRONG MÔI TRƯỜNG 3D

Khoảng cách [m]	Tỉ lệ thành công [%]	Số bước trung bình	Va chạm [%]	Vi phạm độ cao [%]	Hết thời gian [%]
100	95.0	242.1 ± 7.2	3	1	1
200	92.0	558.9 ± 8.7	5	2	1
300	90.0	702.5 ± 9.1	6	2	2

Để làm rõ hơn xu hướng thể hiện trong Bảng 2, Hình 3 trình bày trực quan tỉ lệ thành công và số bước trung bình của SAC theo khoảng cách. Có thể thấy tỉ lệ thành công duy trì ở mức cao trên 90% ở cả ba khoảng cách, chứng tỏ chính sách học được có khả năng điều hướng ổn định ngay cả khi khoảng cách tăng. Tuy nhiên, số bước trung bình tăng đáng kể từ khoảng 300 bước ở 100 m lên hơn 700 bước ở 300 m, cho thấy hành trình dài hơn đòi hỏi UAV nhiều thao tác điều chỉnh hơn để tránh chướng ngại và duy trì an toàn. Xu hướng này cho thấy SAC vừa đạt được độ tin cậy cao, vừa thể hiện sự thích ứng với mức độ phức tạp ngày càng tăng của môi trường.



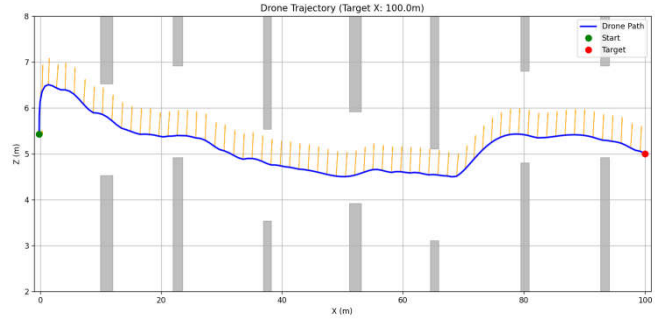
Hình 3. Tỉ lệ thành công và số bước trung bình của SAC theo khoảng cách trong môi trường 3D

B. Phân tích quỹ đạo bay

Trong môi trường 2D, mô hình SAC thể hiện khả năng kiểm soát vững vàng và tính tổng quát hóa tốt trong bài toán điều hướng UAV trên quãng đường 100m. Hình 4 dưới đây minh họa quỹ đạo thành công nhất trong số 100 lần thử nghiệm, trong đó UAV đã vượt qua toàn bộ chướng ngại vật với chuyển động mượt mà và ổn định.

Môi trường bay và quỹ đạo của UAV được biểu diễn trên không gian hai chiều, trong đó các thành phần bao gồm:

- Điểm màu xanh lá là vị trí xuất phát của UAV.
- Điểm màu đỏ là vị trí mục tiêu.
- Đường xanh lam là quỹ đạo bay thực tế của UAV trong quá trình điều hướng.
- Mũi tên cam biểu thị hướng bay (góc nghiêng) của UAV tại các thời điểm khác nhau.
- Hình chữ nhật xám là các chướng ngại vật.



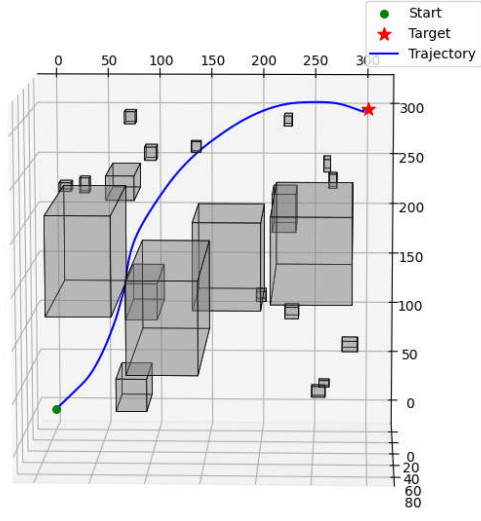
Hình 4. Kết quả điều hướng của máy bay không người lái ở khoảng cách 100m

Trong môi trường 3D, mô hình SAC được sử dụng để huấn luyện UAV điều hướng trên quãng đường 300m với các chướng ngại vật dạng khối hộp được phân bố ngẫu

nhiên. UAV được yêu cầu tiếp cận mục tiêu theo cả ba trục tọa độ (x, y, z), qua đó kiểm tra khả năng thích ứng và đảm bảo an toàn trong không gian ba chiều phức tạp. Hình 5 minh họa quỹ đạo bay thành công của UAV, trong đó UAV đã tránh được các chướng ngại vật và duy trì chuyển động ổn định cho đến khi đạt mục tiêu.

Môi trường bay và quỹ đạo của UAV được trực quan hóa trong không gian ba chiều, bao gồm:

- Điểm màu xanh lá là vị trí xuất phát của UAV.
- Ngôi sao màu đỏ là vị trí mục tiêu.
- Đường xanh lam là quỹ đạo bay thực tế của UAV trong quá trình điều hướng.
- Các khối hộp màu xám là chướng ngại vật trong không gian.



Hình 5. Kết quả điều hướng của UAV trong môi trường 3D ở khoảng cách 300m

Kết quả trong môi trường 2D cho thấy SAC học được chính sách điều khiển toàn cục mà không cần chia đoạn, đạt tỉ lệ thành công cao và duy trì quỹ đạo mượt mà. Thành phần entropy trong hàm mục tiêu giúp chính sách duy trì tính ngẫu nhiên cao ở đầu quá trình huấn luyện, tăng khả năng khám phá và tránh bị mắc kẹt ở cực trị cục bộ[1]. So với PPO hoặc A2C, SAC có ưu điểm là off-policy nên tận dụng tốt bộ nhớ kinh nghiệm và tương thích với batch lớn.

Trong môi trường 3D, hàm phần thưởng được mở rộng để xử lý ba trục, tránh va chạm với cuboid và yêu cầu UAV hãm tốc trước khi tới đích. Việc sử dụng cùng một kiến trúc mạng cho thấy SAC có khả năng tổng quát hóa cao; tuy nhiên các hệ số thưởng/phạt cần được điều chỉnh cẩn thận để cân bằng giữa tốc độ và an toàn. Các thí nghiệm cần đánh giá tỉ lệ thành công trên nhiều khoảng cách khác nhau để xác định mức độ mở rộng của thuật toán. Nhờ khả năng sử dụng buffer lớn và cập nhật off-policy, SAC có tiềm năng vượt trội khi áp dụng cho không gian lớn hơn.

Kết quả này khẳng định tính hiệu quả vượt trội của SAC trong việc điều hướng UAV 2D/3D, đồng thời mở ra hướng nghiên cứu chuyên sâu hơn. Cụ thể, cần thực hiện ablation study để phân tích định lượng đóng góp của từng thành phần trong hàm phần thưởng (ổn định độ cao, tiến độ, kiểm soát vận tốc, thưởng gần đích, phần thưởng/phạt cuối tập). Việc này sẽ giúp làm rõ mức độ ảnh hưởng của từng hệ số tới sự cân bằng giữa tốc độ và an toàn.

IV. KẾT LUẬN

Nghiên cứu đã triển khai hiệu quả thuật toán Soft Actor-Critic cho điều hướng UAV trong môi trường 2D/3D, đạt tỉ lệ thành công và độ an toàn cao hơn các thuật toán so sánh nhờ thiết kế hàm phần thưởng kết hợp đa thành phần. Kết quả đã xác thực tính khả thi của việc tối ưu off-policy và entropy regularization với các bài toán điều hướng phức tạp.

Bên cạnh việc kiểm chứng hiệu quả trên bộ tham số tối ưu hiện tại, nhóm nhận thấy cần thực hiện thêm ablation study, phân tích định lượng từng hệ số c_i trong hàm phần thưởng để làm rõ vai trò và mức ảnh hưởng của từng thành phần đối với chỉ số tốc độ - an toàn. Hướng nghiên cứu này không chỉ giúp giải thích cơ sở khoa học cho việc lựa chọn tham số mà còn đảm bảo quá trình tối ưu đạt cân bằng tốt nhất trong các bài toán thực tiễn.

Kết quả nghiên cứu mở ra triển vọng phát triển cho UAV điều hướng tự động trong môi trường ngày càng phức tạp, đồng thời tạo nền tảng bài bản cho các hướng mở rộng về perception, phối hợp đa agent và so sánh sâu hơn với các thuật toán off-policy hiện đại như TD3, DDPG cải tiến.

TÀI LIỆU THAM KHẢO

[1] J. Amendola et al., “Drone landing and reinforcement learning: State-of-art, challenges and opportunities,” IEEE Open Journal of Intelligent Transportation Systems, vol. 5, 2024.

[2] G. Miera et al., “LiDAR-based drone navigation with reinforcement learning,” Proc. IEEE ICRA, London, 2023.

[3] C. Wang et al., “Vision-based deep reinforcement learning of UAV,” Proc. IEEE/RSJ IROS, Detroit, 2023.

[4] Tạ Chí Hiếu & Phạm Văn Cường, “Segmented PPO-Based Transfer Learning for 2D UAV Navigation,” NSA 2025 Proceedings.

[5] Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor.” International Conference on Machine Learning (ICML).

[6] Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). MIT Press.