ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN VĂN ĐỨC

XẤP XỈ ĐẠO HÀM CHIẾN LƯỢC CHO BÀI TOÁN ĐIỀU KHIỂN MÁY BAY CHIẾN ĐẤU

BÁO CÁO MÔN HỌC THỰC HÀNH NGHIÊN CỨU 2

Ngành: Công nghệ thông tin

HÀ NỘI - 2024

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN VĂN ĐỨC

XẤP XỈ ĐẠO HÀM CHIẾN LƯỢC CHO BÀI TOÁN ĐIỀU KHIỂN MÁY BAY CHIẾN ĐẤU

BÁO CÁO MÔN HỌC THỰC HÀNH NGHIÊN CỨU 2
Ngành: Công nghệ thông tin

Cán bộ hướng dẫn:
PGS. TS. LÊ THANH HÀ
TS. TẠ VIỆT CƯỜNG

HÀ NỘI - 2024

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG BIỂU

MỞ ĐẦU

Lý do chọn đề tài

Ngày nay, căng thẳng địa chính trị diễn ra ở khắp nơi trên toàn thế giới, thúc đẩy các quốc gia không ngừng phát triển và nâng cấp các loại vũ khí hiện đại. Máy bay chiến đấu là một trong những công cụ quân sự quan trọng nhất, đóng vai trò không thể thiếu trong việc bảo vệ không phận, hỗ trợ lực lượng mặt đất và thực hiện các nhiệm vụ đặc biệt, bao gồm: ngăn chặn các cuộc tấn công và thiết lập ưu thế trên không, tấn công các mục tiêu mặt đất, yểm trợ cho lực lượng bộ bình, trinh sát và giám sát. Một máy bay chiến đấu hiệu quả là sự kết hợp hoàn hảo giữa công nghệ và thiết kế. Ngoài động cơ mạnh mẽ và khả năng bay tốc độ cao, một chiến đấu cơ cần phải sở hữu nhiều yếu tố quan trọng khác. Trong đó, phi công là yếu tố con người không thể thiếu. Kỹ năng, kinh nghiệm và sự dũng cảm của phi công quyết định đến hiệu quả chiến đấu của máy bay. Tuy vậy, với sự phát triển của trí tuệ nhân tạo (AI), vai trò của phi công đang dần thay đổi. Các hệ thống AI tiên tiến có khả năng điều khiển máy bay không người lái, thực hiện các nhiệm vụ phức tạp và phản ứng nhanh chóng mà không bị ảnh hưởng bởi yếu tố tâm lý hay mệt mỏi. Điều này mở ra hướng đi mới, trong đó AI có thể thay thế hoặc hỗ trợ phi công, giảm thiểu rủi ro cho con người và tăng cường tính hiệu quả trong các cuộc chiến hiện đại.

Phương pháp nghiên cứu

Mục tiêu của nghiên cứu hướng đến ứng dụng các thuật toán học tăng cường, cụ thể là phương pháp Xấp xỉ Đạo hàm Chiến lược để giải quyết bài toán Điều khiển Máy bay chiến đấu, với các tác vụ cơ bản, bao gồm: dành lợi thế tư thế, tức là bay về phía đuôi của đối thủ và duy trì khoảng cách phù hợp, bắn hạ đối thủ và né tránh tên lửa. Các thuật toán được triển khai trên môi trường giả lập.

CHƯƠNG 1
LÝ THUYẾT ĐẠO HÀM CHIẾN LƯỢC

1.1. Giới thiệu về Học tăng cường

Học tăng cường là một lĩnh vực trong trí tuệ nhân tạo, cụ thể là học máy. Một mô hình Học tăng cường được huấn luyện để tìm hiểu cách thực hiện các hành động để tối đa hóa một mục tiêu hoặc phần thưởng trong môi trường cụ thể. Trong Học tăng cường, mô hình không được cung cấp các cặp đầu vào-đầu ra được đánh giá như các phương pháp Học có giám sát, mà thay vào đó, nó phải tự tìm hiểu thông qua thực nghiệm và tương tác với môi trường.

Các thuật toán Học tăng cường thường xuyên được áp dụng trong các bài toán mà các quyết định phải được đưa ra theo thời gian, và môi trường có thể thay đổi hoặc không biết trước. Chính vì thế, Học tăng cường đã có ứng dụng rộng rãi trong nhiều lĩnh vực, bao gồm điều khiển Rô-bốt, Trò chơi điển tử, Quản lý tài chính, và thậm chí trong Y học. Các nghiên cứu và tiến bộ trong lĩnh vực này đang mở ra nhiều cơ hội mới để xử lý các vấn đề phức tạp và đa dạng thông qua việc học từ trải nghiệm.

1.2. Tối ưu hóa Chiến lược

1.2.1. Phương pháp Đạo hàm Chiến lược

Trong học tăng cường, một chiến lược $\pi_\theta(a|s)$ là một hàm xác suất xác định hành động a tại trạng thái s, với tham số θ. Thay vì biểu diễn hàm giá trị dưới dạng bảng, thì các phương pháp Đạo hàm Chiến lược biểu diễn chiến lược dưới dạng tham số hóa θ, và cố gắng điều chỉnh tham số này để tối đa phần thưởng tích lũy.

Quá trình tác tử tương tác với môi trường được lưu lại dưới dạng một quỹ đạo, bao gồm liên tiếp các bộ trạng thái, hành động, phần thưởng và trạng thái tiếp theo. Gọi $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, ...)$ là quỹ đạo được sinh ra khi tác tử tương tác với môi trường tuân theo chiến lược π_θ. Định nghĩa Hàm mục tiêu $J(\theta)$ là kỳ vọng của tổng phần thưởng trong một quỹ đạo:

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta} \sum_{t=0}^{T} r_t \tag{1.1}$$

Thuật toán sẽ nhắm đến tối đa hóa phần thưởng bằng cách tối đa hóa $J(\theta)$. Đạo hàm của Hàm mục tiêu $J(\theta)$ được viết dưới dạng:

$$\nabla_\theta J(\theta) = \mathbb{E}_{s\sim d^{\pi_\theta}, a\sim\pi_\theta} \nabla_\theta \log \pi_\theta(a|s) Q^{\pi_\theta}(s,a) \tag{1.2}$$

trong đó, $Q^{\pi_\theta(s,a)}$ là hàm Giá trị hành động, đo kỳ vọng tổng thưởng khi chọn hành động a tại trạng thái s và từ đó tuân theo chiến lược π_θ và $d^{\pi_\theta}(s)$ là phân phối trạng thái dưới chiến lược π_θ.

Do hàm Giá trị hành động thường không được biết trước, nên không thể tính chính xác đạo hàm của $J(\theta)$, tuy nhiên có thể ước lược giá trị này:

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i|s_t^i) G_t^i \tag{1.3}$$

trong đó $G_t^i = \sum_{k=t}^{T} r_k^i$ là tổng phần thưởng từ thời điểm t trở đi trong tập quỹ đạo thứ i.

Thuật toán quy hoạch động Stochastic Gradient Ascent được áp dụng để tìm cực đại của Hàm mục tiêu bằng cách cập nhật tham số θ với tốc độ học là α theo công thức:

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta) \tag{1.4}$$

hay cụ thể là

$$\theta \leftarrow \theta + \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) G_t \tag{1.5}$$

Các thuật toán on-policy sử dụng chiến lược mục tiêu để lấy mẫu hành động, và cùng chiến lược này được sử dụng để tối ưu hóa. Ví dụ về các phương pháp on-policy bao gồm REINFORCE và các thuật toán actor-critic cơ bản (vanilla actor-critic). Trong các thuật toán off-policy, hành động được lấy mẫu bằng chiến lược hành vi (behaviour policy), trong khi một chiến lược mục tiêu riêng biệt được sử dụng để tối ưu hóa. Q-learning là một ví dụ về thuật toán off-policy, trong đó ϵ-greedy được sử dụng làm chiến lược hành vi, còn chiến lược mục tiêu hoặc chiến lược cập nhật là chiến lược tham lam tuyệt đối (absolute greedy), bất kể chiến lược hành vi. Ưu điểm chính của thuật toán off-policy so với on-policy:

- Lấy mẫu hiệu quả: Không yêu cầu toàn bộ quỹ đạo vì sử dụng phương pháp học sự khác biệt tạm thời (temporal difference learning). Đồng thời, có thể tái sử dụng các tập dữ liệu từ các tập quỹ đạo trước nhờ bộ nhớ phát lại kinh nghiệm (experience replay buffer).

- Khả năng khám phá tốt hơn: Mẫu được thu thập bằng chính sách hành vi, chính sách này khác với chính sách mục tiêu, giúp tăng khả năng khám phá môi trường.

Tóm lại, Đạo hàm Chiến lược là tên gọi chung cho một tập hợp các thuật toán tối ưu hóa Chiến lược bằng phương pháp off-policy gradient. Các phương pháp Đạo hàm Chiến lược hoạt động bằng cách tính một bộ ước lượng của Đạo hàm Chiến lược và sử dụng nó trong một thuật toán Stochastic Gradient Ascent (SGA), để thực hiện cập nhật tham số để tối đa hóa một Hàm mục tiêu, mục đích cuối cùng là tối đa phần thưởng nhận được.

1.3. Proximal Policy Optimization

1.3.1. Giới thiệu về thuật toán PPO

Proximal Policy Optimization (PPO) là một thuật toán Học tăng cường, giúp đào tạo hàm quyết định của tác nhân máy tính để giải quyết các nhiệm vụ phức tạp. PPO được John Schulman phát triển vào năm 2017 và đã trở thành thuật toán học tăng cường mặc định tại OpenAI, công ty trí tuệ nhân tạo nổi tiếng của Mỹ. Nhiều chuyên gia coi PPO là công nghệ tiên tiến nhất vì nó tạo ra sự cân bằng giữa hiệu suất và khả năng hiểu, giúp giải quyết một số vấn đề trong học tăng cường mà các thuật toán khác gặp khó khăn. So với các thuật toán khác, ba ưu điểm chính của PPO là tính đơn giản, tính ổn định và hiệu quả mẫu.

PPO được phân loại là phương pháp Đạo hàm Chiến lược để đào tạo mạng chiến lược của tác nhân. Mạng chiến lược là hàm mà tác nhân sử dụng để đưa ra quyết định tại mỗi trạng thái của môi trường. Mục tiêu của PPO là đào tạo chiến lược sao cho nó có thể tối ưu hóa hiệu suất của tác nhân trong các nhiệm vụ phức tạp. Theo thuật ngữ Học tăng cường, chiến lược là một phép ánh xạ từ không gian hành động sang không gian trạng thái. Có thể hình dung chiến lược là hướng dẫn cho tác nhân RL, về mặt hành động mà nó nên thực hiện dựa trên trạng thái của môi trường mà nó đang ở.

Khi chúng ta nói về việc đánh giá một tác nhân trong học tăng cường, thường có nghĩa là đánh giá hàm chiến lược của tác nhân để xác định mức độ hoạt động tốt của nó

theo chiến lược đã cho. Đây là nơi các phương pháp Đạo hàm Chiến lược đóng vai trò quan trọng. Khi tác nhân "học" và không biết chắc hành động nào sẽ mang lại kết quả tốt nhất cho các trạng thái tương ứng, nó thực hiện việc này thông qua việc tính toán các gradient chiến lược. Quá trình này tương tự như cách hoạt động của một kiến trúc mạng nơ-ron, trong đó gradient của đầu ra — tức là logarit của xác suất các hành động trong một trạng thái cụ thể — được tính toán theo các tham số của môi trường, và sự thay đổi này sẽ được phản ánh trong chiến lược, dựa trên các gradient đó. Mặc dù phương pháp này đã được thử nghiệm và chứng minh là hiệu quả, nhưng một nhược điểm chính của các phương pháp Đạo hàm Chiến lược là chúng rất nhạy cảm với việc điều chỉnh siêu tham số như kích thước bước, tốc độ học, và các tham số khác. Hơn nữa, hiệu quả mẫu của các phương pháp này không cao. So với học có giám sát, trong đó lộ trình thành công hoặc hội tụ có thể đảm bảo với ít điều chỉnh siêu tham số hơn, học tăng cường phức tạp hơn nhiều. Quá trình học trong học tăng cường đụng phải nhiều yếu tố cần phải được cân nhắc và điều chỉnh đồng thời.

PPO được phát triển để cân bằng giữa các yếu tố quan trọng, bao gồm: Tính dễ triển khai, tính dễ điều chỉnh vì hạn chế sự nhạy cảm với các siêu tham số và khả năng hỗ trợ việc tối ưu hóa hiệu quả mẫu mà không cần sử dụng quá nhiều tài nguyên tính toán. Trên thực tế, PPO là một phương pháp Đạo hàm Chiến lược có thể học từ dữ liệu trực tuyến. Điều quan trọng là nó đảm bảo chiến lược được cập nhật không quá khác biệt so với chiến lược trước đó, giúp giảm thiểu sự thay đổi quá mức trong chiến lược và giúp duy trì độ ổn định trong quá trình học. Điều này giúp giảm phương sai trong quá trình đào tạo, đảm bảo quá trình học ổn định và hiệu quả hơn.

1.3.2. Kiến trúc thuật toán PPO

Kiến trúc của Proximal Policy Optimization (PPO) chủ yếu liên quan đến các mạng nơ-ron được sử dụng để biểu diễn các hàm chính sách và hàm giá trị trong học tăng cường sâu (Deep Reinforcement Learning).

1.3.2.1. Mạng Chiến lược

Mạng Chiến lược là thành phần quan trọng của PPO, chịu trách nhiệm đưa ra phân phối xác suất cho các hành động có thể thực hiện trong một trạng thái cụ thể.

$$s \to \pi(a|s) \tag{1.6}$$

Kiến trúc của mạng Chiến lược có thể khác nhau tùy vào tính chất của không gian

trạng thái. Mạng nơ-ron truyền thẳng dùng khi không gian trạng thái có cấu trúc đơn giản; trong khi mạng nơ-ron hồi quy dùng cho các môi trường có tính tuần tự, chẳng hạn như dữ liệu chuỗi thời gian; hoặc mạng nơ-ron tích chập thường dùng cho các môi trường có cấu trúc không gian, như xử lý hình ảnh.

1.3.2.2. Mạng Giá trị

Mạng Giá trị ước tính phần thưởng tích lũy dự kiến (giá trị) khi ở trong một trạng thái nhất định. Mạng này được dùng để đánh giá việc đưa ra quyết định của mạng Chiến lược có thực sự tốt hay không, việc này gọi là Đánh giá Chiến lược. Đầu vào của mạng là trạng thái của môi trường, đầu ra là giá trị của trạng thái đó.

$$s \rightarrow V^{\pi}(s) \tag{1.7}$$

Mạng Giá trị có thể có kiến trúc tương tự như mạng Chiến lược, nhưng thay vì xuất ra một phân phối xác suất cho các hành động, nó đưa ra một giá trị duy nhất cho mỗi trạng thái.

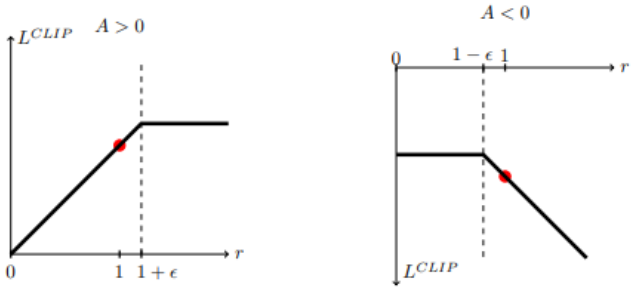
1.3.2.3. Hàm Mất mát

Tương ứng với hai mạng nơ-ron, thuật toán PPO đưa ra hai hàm Mất mát, cụ thể là hàm mục tiêu Surrogate sử dụng cho mạng Chiến lược và MSE sử dụng cho mạng Giá trị.

Hàm mục tiêu Surrogate

$$L^{CLIP}(\theta) = E_t[\min(r_t(\theta)A_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t)] \tag{1.8}$$

trong đó $r_t(\theta) = e^{\pi_\theta(a|s) - \pi_{\theta_{old}}(a|s)}$ là hàm tỷ lệ, đo đạc sự khác biệt giữa chiến lược mới và chiến lược cũ.



Hình 1.1. Hàm mục tiêu Clip Surrogate

MSE

$$L_t^{VF} = \frac{1}{N}\sum_{t=1}^{N}(V(s_t) - G_t)^2 \tag{1.9}$$

1.3.3. Mô tả thuật toán PPO

Hàm mục tiêu Clipped Surrogate cuối cùng cho PPO dựa theo Actor-Critic trông như sau, đây là sự kết hợp giữa hàm mục tiêu Clipped Surrogate, hàm mất mát giá trị và entropy của phần thưởng:

$$L_t^{CLIP+VF+S}(\theta) = \mathbb{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S[\pi_\theta](s_t)] \tag{1.10}$$

trong đó, c_1, c_2 là hệ số, và S ký hiệu cho entropy của phần thưởng, và L_t^{VF} là hàm mất mát giá trị.

Một cách triển khai Đạo hàm Chiến lược khá phổ biến và rất phù hợp để sử dụng với mạng nơ-ron hồi quy, đó là chạy chiến lược trong T bước thời gian (với T nhỏ hơn nhiều so với số lượng tập), và sử dụng các mẫu thu thập được để cập nhật. Cách này yêu cầu một bộ ước lượng lợi thế không nhìn xa hơn bước thời gian T. Bộ ước lượng được sử dụng là:

$$A_t = -V(s_t) + r_t + \gamma r_{t+1} + ... + \gamma^{T-t+1} r_{T-1} + \gamma^{T-t} V(s_T) \tag{1.11}$$

trong đó t là bước thời gian, thuộc khoảng $[0, T]$, nằm trong một đoạn quỹ đạo có độ dài T. Tổng quát hóa lựa chọn này, chúng ta có thể sử dụng một phiên bản rút gọn của ước lượng lợi thế tổng quát (Generalized Advantage Estimation), mà sẽ trở thành phương trình (1.11) khi $\lambda = 1$:

$$A_t = \delta_t + (\gamma\lambda)\delta_{t+1} + ... + (\gamma\lambda)^{T-t+1}\delta_{T-1} \tag{1.12}$$

trong đó $\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$

Thuật toán Proximal Policy Optimization (PPO) sử dụng các đoạn quỹ đạo có độ dài cố định được mô tả như sau. Trong mỗi vòng lặp, mỗi trong số N tác nhân (chạy song song) thu thập dữ liệu trong T bước thời gian. Sau đó, chúng ta xây dựng hàm mất mát thay thế dựa trên NT bước thời gian này và tối ưu hóa nó bằng phương pháp SGD theo minibatch (hoặc thường sử dụng Adam để có hiệu suất tốt hơn) trong K epoch.

Algorithm 1 PPO, Actor-Critic Style

for iteration=$1, 2, \ldots$ **do**
  for actor=$1, 2, \ldots, N$ **do**
    Run policy $\pi_{\theta_{\text{old}}}$ in environment for T timesteps
    Compute advantage estimates $\hat{A}_1, \ldots, \hat{A}_T$
  end for
  Optimize surrogate L wrt θ, with K epochs and minibatch size $M \leq NT$
  $\theta_{\text{old}} \leftarrow \theta$
end for

Hình 1.2. Mã giả cho thuật toán PPO dạng Actor-Critic

CHƯƠNG 2
BÀI TOÁN ĐIỀU KHIỂN MÁY BAY CHIẾN ĐẤU

2.1. Giới thiệu bài toán

Trong những năm gần đây, với sự phát triển mạnh mẽ của công nghệ mô phỏng và trí tuệ nhân tạo, các bài toán điều khiển tự động hóa trong lĩnh vực hàng không đã trở thành một trong những trọng tâm nghiên cứu. Đặc biệt, các ứng dụng điều khiển máy bay chiến đấu không người lái (Unmanned Combat Aerial Vehicles - UCAV) đang ngày càng được chú trọng, khi nhu cầu về hiệu quả chiến đấu và an toàn phi công ngày càng tăng cao.

Các bài toán điều khiển máy bay chiến đấu đặt ra nhiều thách thức. Thứ nhất là đặc tính phi tuyến của động lực học máy bay. Máy bay chiến đấu thường hoạt động ở những môi trường khắc nghiệt, đòi hỏi hệ thống điều khiển phải phản ứng nhanh, chính xác trong các điều kiện biến đổi liên tục. Tiếp đó là tối ưu hóa nhiều mục tiêu: Các tác vụ chiến đấu thường liên quan đến việc cân bằng giữa việc tấn công mục tiêu, tránh tên lửa, duy trì độ cao an toàn và tối ưu hóa hành trình. Bên cạnh đó là tương tác đa tác tử: Trong các tình huống chiến đấu, máy bay phải đối đầu với nhiều đối thủ, yêu cầu khả năng ra quyết định chiến lược, hợp tác hoặc cạnh tranh trong thời gian thực. Cuối cùng là yêu cầu thời gian thực: Các quyết định phải được đưa ra trong khoảng thời gian rất ngắn, đảm bảo tính hiệu quả và an toàn. Để giải quyết các vấn đề này, mô hình hóa các bài toán điều khiển máy bay chiến đấu trong môi trường mô phỏng là bước đầu quan trọng. Những môi trường như vậy cho phép kiểm tra và huấn luyện các tác tử AI thông qua các chiến lược học tăng cường (Reinforcement Learning - RL). Điển hình là các thuật toán như Proximal Policy Optimization (PPO), mang lại hiệu quả cao trong việc tối ưu hóa chiến lược điều khiển.

Bài toán điều khiển máy bay chiến đấu có thể được phân loại thành:

- **Điều khiển cơ bản**. Bao gồm điều chỉnh hướng bay, độ cao, tốc độ để thực hiện các nhiệm vụ cơ bản.
- **Chiến đấu không vũ khí**. Tối ưu hóa vị trí và tư thế để tránh địch và tạo lợi thế

chiến lược.

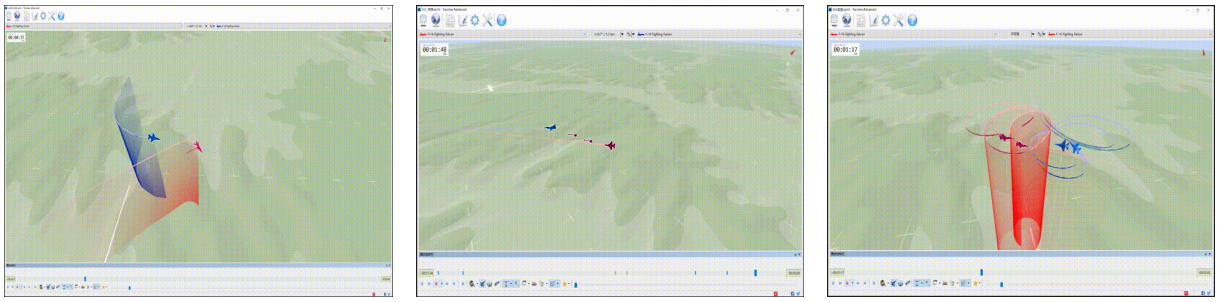
- **Chiến đấu sử dụng vũ khí**. Phối hợp giữa tấn công và phòng thủ, bao gồm việc né tên lửa và tấn công đối thủ hiệu quả.

Nhờ vào các khuôn khổ mô phỏng và các thuật toán học máy hiện đại, bài toán này không chỉ giúp nâng cao khả năng điều khiển tự động hóa trong hàng không mà còn mở ra nhiều hướng nghiên cứu quan trọng trong trí tuệ nhân tạo, động lực học và tối ưu hóa chiến lược.

2.2. Môi trường mô phỏng

Nghiên cứu này sử dụng môi trường mô phỏng có tên là Light Aircraft Game. Đây là một môi trường mô phỏng nhẹ, linh hoạt và có khả năng mở rộng, được đóng gói dưới dạng giao diện tương thích với Gym để tạo điều kiện cho việc triển khai các thuật toán Học tăng cường. Môi trường có tích hợp sẵn động lực lực học bay sử dụng JSBSim để tái hiện chuyển động chính xác của máy bay; và động lực học tên lửa, được triển khai dựa trên thuật toán dẫn đường tỷ lệ, giúp mô phỏng các tình huống truy đuổi và tấn công thực tế.

Các môi trường cơ bản trong bộ mô phỏng Light Aircraft Game bao gồm: Single Control, Single Combat và Multiple Combat.



((a)) Single Control ((b)) Single Combat ((c)) Multiple Combat

Hình 2.1. Các môi trường cơ bản trong Light Aircraft Game được hiển thị trên TactView

2.3. Các tham số môi trường

2.3.1. Trạng thái dừng

Môi trường cung cấp nhiều trạng thái dừng khác nhau.

Trạng thái Cực Đoan được xác định thông qua một số tiêu chí cụ thể, bao gồm vận tốc dài quá lớn, vận tốc góc quá lớn, độ cao quá lớn và gia tốc quá mức cho phép.

Khi một trong các chỉ số này vượt quá ngưỡng an toàn, mô phỏng sẽ kết thúc vì tình trạng của máy bay không còn trong phạm vi hoạt động bình thường, điều này có thể dẫn đến những nguy cơ nghiêm trọng cho máy bay và nhiệm vụ.

Trạng thái Độ Cao Thấp xảy ra khi máy bay hạ thấp xuống mức độ cao quá thấp, gây nguy hiểm cho an toàn bay. Khi máy bay bay dưới độ cao an toàn hoặc vượt quá giới hạn tối thiểu cho phép, mô phỏng sẽ kết thúc. Điều này đảm bảo rằng máy bay không rơi vào trạng thái nguy hiểm, có thể dẫn đến tai nạn hoặc thất bại trong nhiệm vụ.

Trạng thái Overload diễn ra khi gia tốc của máy bay vượt quá giới hạn tối đa mà hệ thống có thể chịu đựng. Nếu gia tốc máy bay lớn hơn ngưỡng cho phép, mô phỏng sẽ kết thúc để tránh việc gây hại cho máy bay hoặc các hệ thống của nó. Điều này nhằm duy trì sự an toàn trong suốt quá trình mô phỏng, đảm bảo máy bay hoạt động trong phạm vi chịu tải cho phép.

Trạng thái Safe Return được kích hoạt khi máy bay bị bắn hạ hoặc khi tất cả các máy bay đối thủ đã bị phá hủy trước khi máy bay của mình bị hạ gục. Trong tình huống này, mô phỏng sẽ kết thúc, xác định kết quả của cuộc đối đầu, với chiến thắng thuộc về bên còn lại nếu máy bay của mình bị bắn hạ trước.

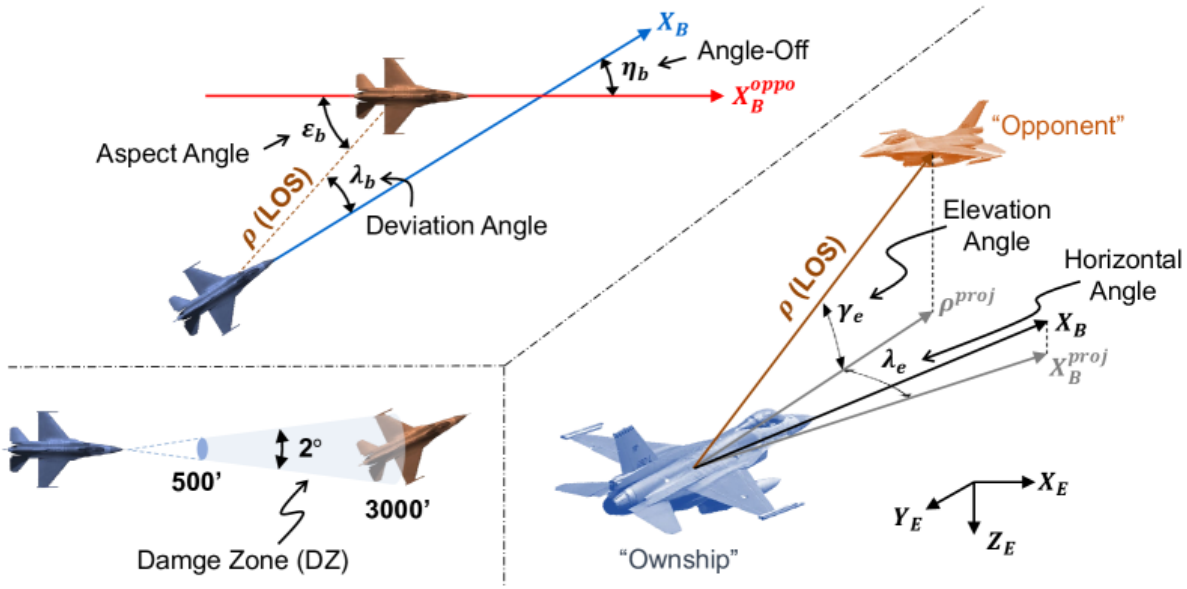
Trạng thái Timeout xảy ra khi thời gian trải nghiệm vượt quá giới hạn cho phép trong mô phỏng. Nếu quá trình mô phỏng kéo dài hơn thời gian dự định, bất kể kết quả của các hành động trong trò chơi, mô phỏng sẽ kết thúc và một kết quả sẽ được ghi nhận. Điều này nhằm kiểm soát và đảm bảo rằng quá trình mô phỏng không kéo dài ngoài mức cần thiết.

Trạng thái Unreach Heading sẽ kết thúc mô phỏng nếu máy bay không thể đạt được hướng mục tiêu (target heading) trong khoảng thời gian giới hạn. Đây là một tình huống phạt khi máy bay không thể hoàn thành nhiệm vụ điều hướng, do sai sót trong điều khiển hoặc gặp phải vấn đề về khí động học, khiến việc duy trì hoặc thay đổi hướng trở nên không thể thực hiện.

2.3.2. Các trạng thái

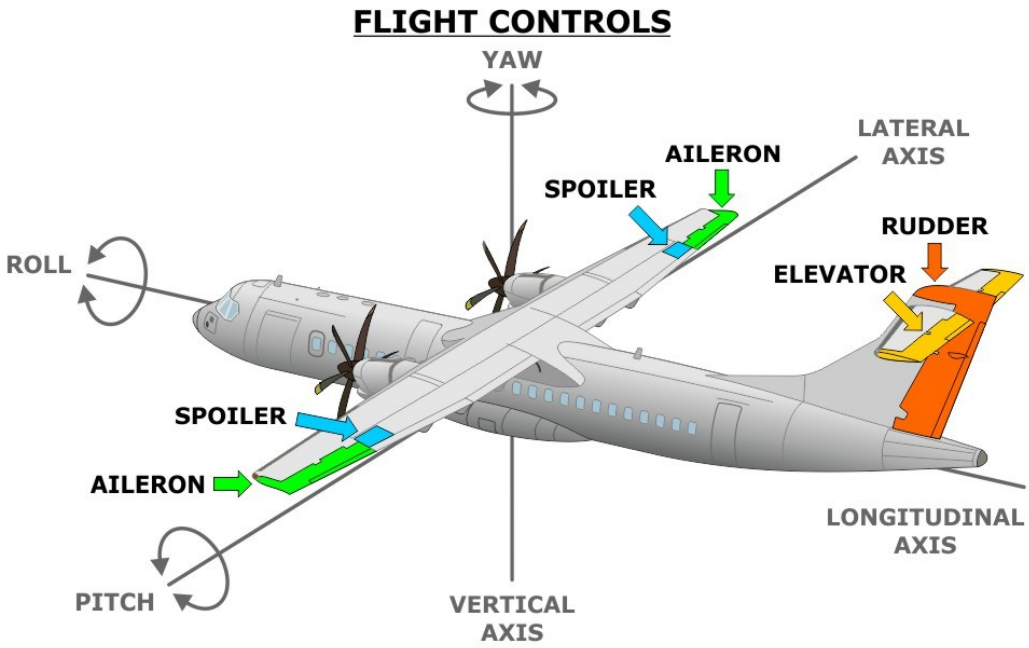
Môi trường cung cấp các thông số liên quan đến trạng thái máy bay.

- **Độ cao tương đối và độ cao tuyệt đối**. Độ cao tương đối của máy bay so với máy bay khác hoặc so với một mục tiêu nhất định, độ cao tuyệt đối so với mực nước biển. Đơn vị (m).



Hình 2.2. Các thông số trạng thái của máy bay

- **Hướng tương đối**. Hướng hiện tại của máy bay so với hướng mục tiêu. Đơn vị (rad).
- **Vận tốc dài tương đối và vận tốc dài tuyệt đối**. Các vận tốc dài tương đối là vận tốc dài theo hệ tọa độ của máy bay, các vận tốc dài tuyệt đối là vận tốc dài theo hệ tọa độ quan sát. Đơn vị (m/s)



Hình 2.3. Các thông số trạng thái của máy bay

- **Tư thế của máy bay**. Các góc nghiêng của máy bay theo các trục dọc (roll), trục ngang (pitch) và trục thẳng đứng (yaw). Đơn vị (rad)
- **Vận tốc không khí**. Thông số liên quan đến vận tốc không khí. Đơn vị (m/s)

2.3.3. Các hành động

Giống như một vật rắn, một máy bay trong không gian cần 3 tham số để xác định tư thế (roll, pitch, yaw) và 1 tham số để xác định tọa độ trọng tâm. Vì vậy để điều khiển vị trí của máy bay thì cần điều khiển 4 tham số này.

- **Điều khiển góc roll**. Điều khiển góc của cánh liệng (aileron), lệnh được chuẩn hóa trong khoảng [-1, +1]. Trong đó -1 tương ứng với lệnh lệch tối đa cánh trái, +1 tương ứng với lệnh lệch tối đa cánh phải.
- **Điều khiển góc pitch**. Điều khiển góc của cánh nâng (elevator), lệnh được chuẩn hóa trong khoảng [-1, +1]. Trong đó -1 tương ứng với lệnh cúi xuống tối đa (pitch xuống) và +1 tương ứng với lệnh ngẩng lên tối đa (pitch lên).
- **Điều khiển góc yaw**. Điều khiển góc của cánh lái hướng (rudder), lệnh được chuẩn hóa trong khoảng [-1, +1]. Trong đó -1 tương ứng với lệnh quay tối đa sang trái, +1 tương ứng với lệnh quay tối đa sang phải.
- **Điều khiển tốc độ máy bay**. Điều khiển lực đẩy động cơ (throttle), lệnh được chuẩn hóa trong khoảng [0.4, 0.9]. Trong đó 0.4 tương ứng với lực đẩy tối thiểu (giới hạn dưới trong mô phỏng để đảm bảo máy bay không bị rơi), 0.9 tương ứng với lực đẩy tối đa (giới hạn trên cho động cơ).

2.3.4. Các hàm phần thưởng

Altitude Reward có tác dụng phạt nếu máy bay chiến đấu không đáp ứng các yêu cầu về độ cao. Khi máy bay bay thấp hơn mức độ cao an toàn, phần thưởng sẽ bị trừ và nằm trong phạm vi từ -1 đến 0. Tương tự, khi máy bay bay thấp hơn độ cao nguy hiểm, hàm phần thưởng cũng sẽ phạt trong phạm vi tương tự. Điều này nhằm đảm bảo rằng máy bay không rơi vào trạng thái nguy hiểm về mặt độ cao, đồng thời khuyến khích máy bay duy trì độ cao an toàn trong suốt nhiệm vụ.

Event Driven Reward được sử dụng để đánh giá các sự kiện quan trọng trong quá trình bay. Nếu máy bay bị bắn hạ bởi tên lửa, phần thưởng sẽ là -200, điều này phản ánh

sự thất bại trong việc bảo vệ máy bay khỏi mối đe dọa. Tương tự, nếu máy bay rơi do tai nạn, phần thưởng cũng bị trừ -200, vì đây là một tình huống không mong muốn. Ngược lại, khi máy bay bắn hạ được đối thủ, phần thưởng sẽ là +200, khuyến khích hành động tấn công và thành công trong chiến đấu.

Heading Reward đo lường sự khác biệt giữa hướng bay hiện tại của máy bay và hướng mục tiêu. Phần thưởng được tính dưới dạng trung bình nhân của các phần thưởng Gaussian đã được chuẩn hóa cho từng biến hướng có liên quan. Mục tiêu của hàm phần thưởng này là khuyến khích máy bay bay về đúng hướng mục tiêu, đồng thời giảm thiểu các sai lệch về hướng để tối ưu hóa hiệu suất của nhiệm vụ.

Missile Posture Reward dựa trên sự suy giảm vận tốc của tên lửa. Phần thưởng này phản ánh mức độ giảm vận tốc của tên lửa khi tiếp cận mục tiêu. Mục đích là đánh giá và tối ưu hóa hiệu suất của tên lửa trong quá trình bay và tác động đến kết quả chiến đấu, từ đó khuyến khích các chiến thuật phóng tên lửa hiệu quả.

Posture Reward được xây dựng dựa trên hai yếu tố chính: Hướng (Orientation) và Khoảng cách (Range). Về hướng, phần thưởng sẽ khuyến khích máy bay bay về phía đối thủ, đồng thời phạt nếu máy bay bị đối thủ nhắm vào. Về khoảng cách, phần thưởng khuyến khích máy bay tiếp cận gần đối thủ để tăng khả năng tấn công, nhưng cũng phạt nếu máy bay quá xa đối thủ. Phần thưởng này là sự kết hợp của các yếu tố như góc phương vị (AO), góc tới (TA), và khoảng cách (R) tại thời điểm cuối cùng của hành động.

Relative Altitude Reward đánh giá độ cao tương đối giữa máy bay chiến đấu hiện tại và máy bay đối thủ. Phần thưởng sẽ bị trừ nếu độ cao của máy bay hiện tại lớn hơn 1000m so với đối thủ, trong phạm vi từ -1 đến 0. Điều này nhằm duy trì sự ổn định và kiểm soát độ cao trong môi trường chiến đấu, khuyến khích máy bay duy trì sự kiểm soát trong suốt nhiệm vụ. Phần thưởng là tổng của tất cả các hình phạt được áp dụng.

Shoot Penalty Reward là một biện pháp phạt nhằm hạn chế việc phóng tên lửa một cách không hợp lý. Khi máy bay phóng tên lửa, một phần thưởng -10 sẽ được áp dụng để tránh việc phóng tất cả tên lửa cùng một lúc mà không có mục tiêu rõ ràng. Phần thưởng này là tổng của tất cả các sự kiện phóng tên lửa và khuyến khích hành động phóng tên lửa có tính toán và chiến lược.

2.3.5. Các thông số hiển thị

Đây là các thông số dùng để hiển thị quá trình mô phỏng.

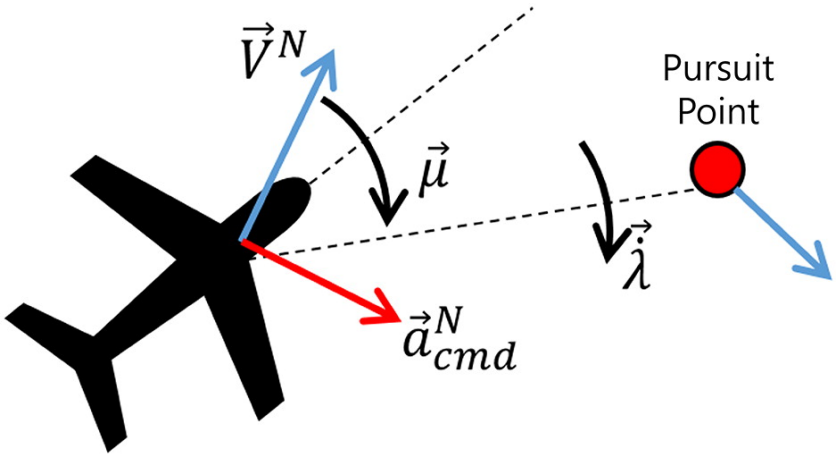
- Kinh độ (longitude) của máy bay, tính theo đơn vị độ (°). Vị trí của máy bay trên trục đông-tây của Trái Đất. Giá trị này được sử dụng để xác định vị trí chính xác trên bản đồ hoặc môi trường 3D.
- Vĩ độ (latitude) của máy bay, tính theo đơn vị độ (°). Vị trí của máy bay trên trục bắc-nam của Trái Đất. Tương tự kinh độ, thông số này được sử dụng để định vị máy bay trong không gian địa lý.
- Độ cao (altitude) của máy bay so với mực nước biển, đo bằng mét (m). Thông số này biểu thị độ cao hiện tại của máy bay, dùng để xác định vị trí trên trục thẳng đứng trong mô phỏng hoặc môi trường đồ họa.
- Góc lăn (roll) của máy bay, đo bằng radian (rad). Góc quay của máy bay quanh trục dọc (từ trước ra sau). Đây là trạng thái nghiêng sang trái hoặc phải, thường được dùng để biểu diễn chuyển động quay tròn trong không gian.
- Góc ngẩng/cúi (pitch) của máy bay, đo bằng radian (rad). Góc giữa thân máy bay và mặt phẳng ngang, biểu thị trạng thái máy bay hướng lên trên hoặc xuống dưới trong không gian.
- Hướng thực tế (true heading) của máy bay, đo bằng radian (rad). Hướng mà máy bay đang bay so với hướng bắc thực. Thông số này thường được hiển thị trong giao diện điều khiển hoặc radar để người dùng dễ dàng quan sát hướng bay.

2.4. Các tác vụ

2.4.1. Tác vụ Điều hướng

Tác vụ điều hướng tập trung vào việc giữ ổn định và điều chỉnh hướng bay của máy bay để đạt được hướng mục tiêu. Hàm phần thưởng cho tác vụ này bao gồm các hàm nhắm vào việc phạt các hành động điều hướng và kiểm soát độ cao, cụ thể là HeadingReward() và AltitudeReward(). Điều kiện dừng của nhiệm vụ được xác định thông qua các trạng thái như: UnreachHeading() (không đạt được hướng), ExtremeState() (trạng thái vượt giới hạn), Overload() (quá tải), LowAltitude() (độ cao quá thấp), và Timeout() (hết thời gian). Trạng thái của hệ thống được mô tả thông qua các biến như: độ chênh lệch độ cao (Delta Altitude), chênh lệch hướng bay (Delta Heading), vận tốc (Delta Velocities), và các thông số khác như độ cao tuyệt đối (Altitude), góc nghiêng dọc (Roll), góc nghiêng ngang (Pitch), và các vận tốc trong hệ tọa độ thân máy bay. Hành

động bao gồm các lệnh điều khiển như điều chỉnh cánh liệng (Aileron Command), cánh lái độ cao (Elevator Command), cánh đuôi đứng (Rudder Command), và lực đẩy động cơ (Throttle Command), với các giá trị đều được chuẩn hóa để đảm bảo tính linh hoạt. Các thông số hiển thị như kinh độ, vĩ độ, độ cao, và hướng bay giúp giám sát trực quan quá trình điều hướng.



Hình 2.4. Tác vụ Điều hướng máy bay

2.4.2. Tác vụ Chiến đấu không vũ khí

Tác vụ này mô phỏng các tình huống chiến đấu không sử dụng vũ khí, tập trung vào kiểm soát tư thế và vị trí máy bay. Trong tác vụ này, tác tử sẽ cố gắng điều khiển máy bay bay về phía sau đối thủ và giữ một khoảng cách nhất định, điều này sẽ giúp né tránh tên lửa và giành lợi thế tư thế trong chiến đấu. Hàm phần thưởng bao gồm AltitudeReward() để phạt khi độ cao không đạt yêu cầu, PostureReward() để phạt tư thế không phù hợp, và EventDrivenReward() để phạt các hành động không đáp ứng mục tiêu. Điều kiện dừng bao gồm các yếu tố như: độ cao thấp (LowAltitude()), trạng thái vượt giới hạn (ExtremeState()), quá tải (Overload()), đảm bảo an toàn trở về (SafeReturn()), và hết thời gian (Timeout()). Trạng thái của hệ thống được mô tả qua kinh độ, vĩ độ, độ cao, góc lăn (Roll angle), góc ngẩng (Pitch angle), góc phương vị (Yaw angle), cùng các vận tốc theo phương Bắc, Đông, thẳng đứng, và hệ tọa độ thân máy bay. Hành động bao gồm điều khiển các bề mặt khí động học như cánh liệng, cánh nâng, cánh lái hướng, và điều chỉnh lực đẩy động cơ, tất cả đều chuẩn hóa để đảm bảo độ chính xác. Các thông số hiển thị tương tự như kinh độ, vĩ độ, độ cao, và tư thế máy bay, cung cấp dữ liệu trực quan cho người quan sát.

2.4.3. Tác vụ Chiến đấu sử dụng tên lửa

Tác vụ này bổ sung yếu tố tấn công và phòng thủ bằng cách mô phỏng tình huống bắn và né tên lửa. Hàm phần thưởng bao gồm PostureReward() để duy trì tư thế, AltitudeReward() để kiểm soát độ cao, EventDrivenReward() để đánh giá hành động theo sự kiện, và các hàm đặc thù như MissilePostureReward() và ShootPenaltyReward(). Điều kiện dừng tương tự tác vụ chiến đấu không vũ khí, bao gồm các trạng thái như độ cao thấp, trạng thái vượt giới hạn, quá tải, và đảm bảo an toàn. Trạng thái trong tác vụ này phức tạp hơn với thông tin chi tiết về máy bay chính (độ cao, vận tốc, tư thế) và thông tin tương đối về máy bay địch (vị trí, khoảng cách, vận tốc) cũng như tên lửa (vị trí, hướng, khoảng cách). Hành động trong tác vụ này tương tự như các tác vụ trước nhưng tập trung vào phối hợp điều khiển để đối phó với mối đe dọa từ tên lửa. Các thông số hiển thị cũng bao gồm kinh độ, vĩ độ, độ cao, và tư thế, giúp theo dõi chiến thuật trong thời gian thực.

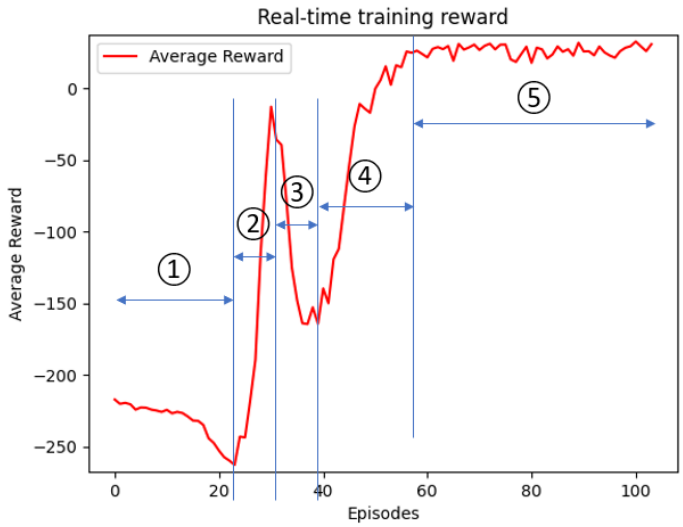
CHƯƠNG 3
KẾT QUẢ TRIỂN KHAI THUẬT TOÁN

Light Aircraft Game cung cấp mô phỏng 3 tác vụ chính, bao gồm: Single Control, Single Combat và Multiple Combat. Dưới đây là triển khai đào tạo tác tử điều khiển máy bay thực hiện cả 3 tác vụ này sử dụng thuật toán PPO cho Single Control và Single Combat, và MAPPO cho Multiple Combat.

Các thí nghiệm được triển khai trên phần cứng bao gồm CPU Intel(R) Core(TM) i5-14600K, GPU NVIDIA GeForce RTX 4070 12GB với các bước đào tạo từ 10 triệu đến 50 triệu bước.

3.1. Single Control

Huấn luyện tác tử điều khiển máy bay thực hiện tác vụ Điều hướng, huấn luyện trên 10 triệu bước chia thành 104 tập, với 1620 FPS.



Hình 3.1. Kết quả training trong Tác vụ Điều hướng máy bay (1)

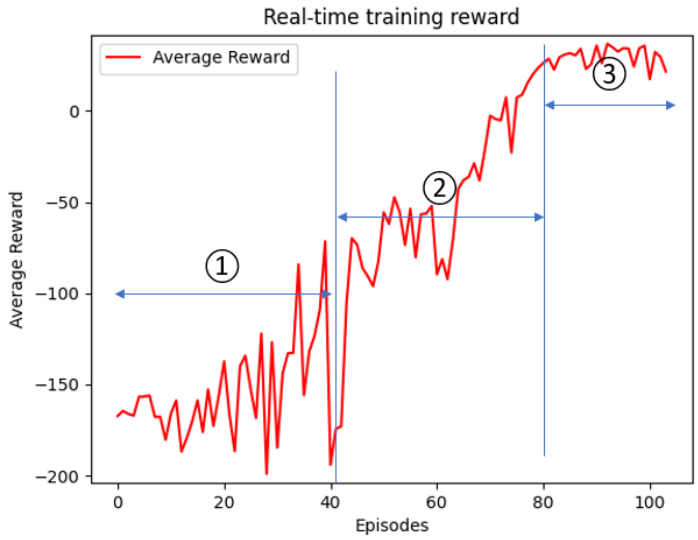
Trong quá trình huấn luyện, tác tử trải qua năm giai đoạn chính. Ở giai đoạn 1, tác tử thực hiện các hành động một cách ngẫu nhiên để khám phá môi trường, dẫn đến phần thưởng nhận được thường thấp. Đến giai đoạn 2, tác tử tìm ra một chiến lược tốt hơn, nhưng chiến lược này vẫn chưa phải là tối ưu. Sau đó, trong giai đoạn 3, phần thưởng có thể giảm xuống khi tác tử thử nghiệm các chiến lược mới hoặc khi môi trường có sự thay

đổi. Tiếp tục ở giai đoạn 4, tác tử phát hiện các chiến lược tốt hơn, giúp phần thưởng tăng dần một cách ổn định. Cuối cùng, tại giai đoạn 5, quá trình hội tụ xảy ra khi tác tử đạt được chiến lược tối ưu.

Hình 3.2. Kết quả training trong Tác vụ Điều hướng máy bay (2)

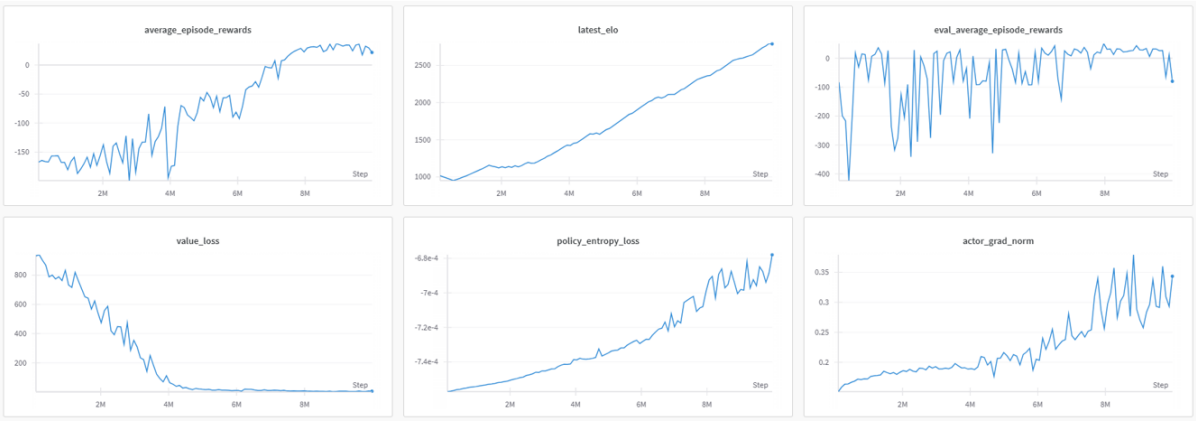
3.2. Single Combat - No Weapon

Huấn luyện tác tử điều khiển máy bay thực hiện tác vụ Chiến đấu không vũ khí. Trong tác vụ này, tác tử sẽ cố gắng điều khiển máy bay bay ra phía sau đối thủ và giữ khoảng cách nhất định, điều này giúp giành lợi thế về tư thế trong chiến đấu, giúp máy bay né được tầm nhìn của đối thủ và có được vị trí tốt để tấn công. Thử nghiệm này diễn ra trên 10 triệu bước chia thành 104 tập, với 1530 FPS.

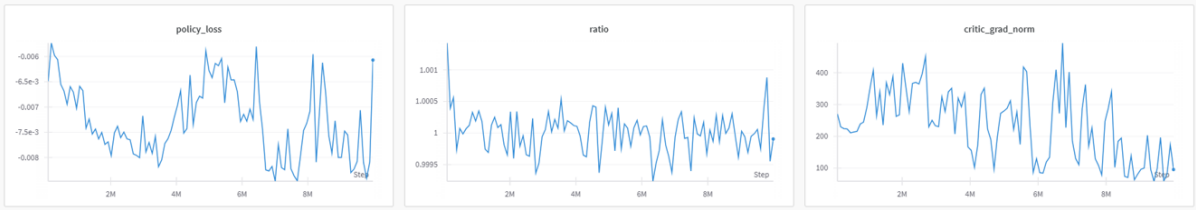


Hình 3.3. Kết quả training trong Tác vụ Chiến đấu không vũ khí (1)

Quá trình huấn luyện của tác tử diễn ra qua ba giai đoạn chính. Ở giai đoạn 1, phần thưởng nhận được có mức độ dao động lớn, cho thấy tác tử đang trong giai đoạn khám phá. Trong giai đoạn này, tác tử thử nghiệm nhiều chiến lược khác nhau để hiểu rõ hơn về môi trường và xác định các hành động tiềm năng. Đến giai đoạn 2, tác tử bắt đầu tìm ra những chiến lược tốt hơn, dẫn đến phần thưởng cải thiện dần. Tuy nhiên, quá trình khám phá vẫn tiếp tục, khiến phần thưởng chưa ổn định và vẫn dao động nhiều. Cuối cùng, trong giai đoạn 3, tác tử tìm ra chiến lược tối ưu, dẫn đến quá trình huấn luyện hội tụ. Phần thưởng trở nên ổn định hơn khi tác tử áp dụng chiến lược này một cách nhất quán và hiệu quả



Hình 3.4. Kết quả training trong Tác vụ Chiến đấu không vũ khí (2)



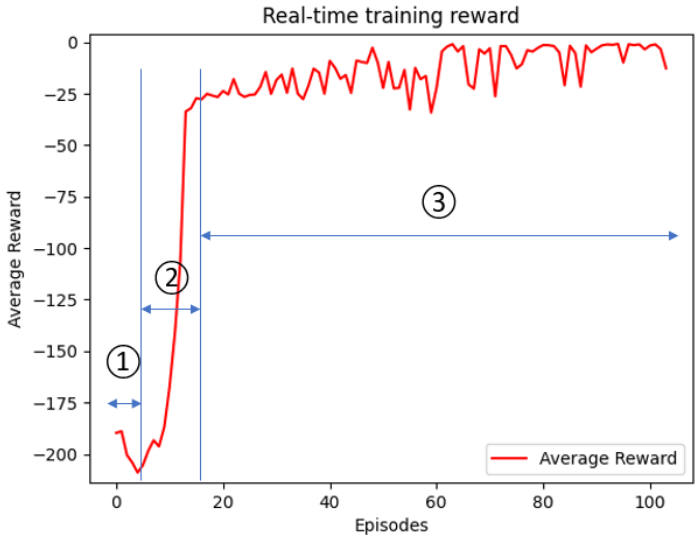
Hình 3.5. Kết quả training trong Tác vụ Chiến đấu không vũ khí (3)

3.3. Single Combat - Shoot

Huấn luyện tác tử điều khiển máy bay thực hiện tác vụ Chiến đấu có sử dụng tên lửa. Trong tác vụ này, tác tử sẽ cố gắng điều khiển máy bay giành lợi thế mục tiêu, né tránh tên lửa và tấn công máy bay địch bằng tên lửa. Thử nghiệm này diễn ra trên 10 triệu bước chia thành 104 tập, với 640 FPS.

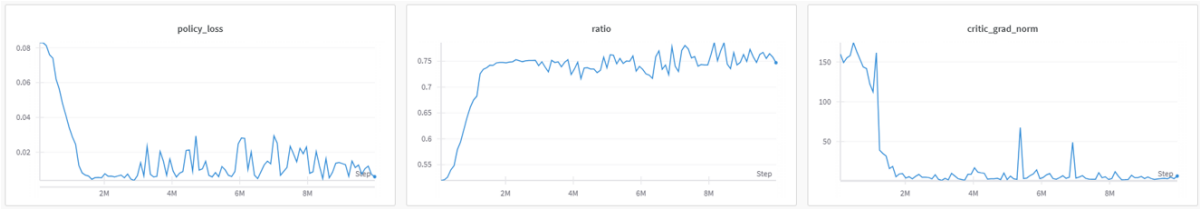
3.4. Multiple Combat

Trong tác vụ này, nhiều tác tử chia sẻ một chiến lược hoặc một mạng nơ-ron chung, nhưng vẫn cạnh tranh hoặc hợp tác với nhau trong quá trình huấn luyện. Mục đích của



Hình 3.6. Kết quả training trong Tác vụ Chiến đấu có sử dụng tên lửa (1)

Hình 3.7. Kết quả training trong Tác vụ Chiến đấu có sử dụng tên lửa (2)

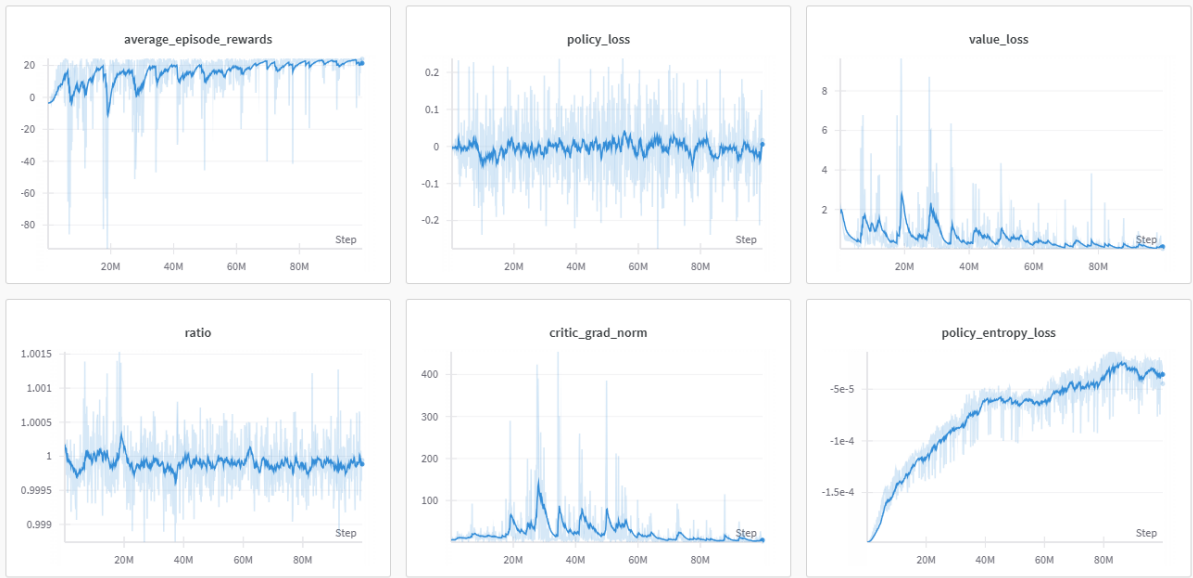


Hình 3.8. Kết quả training trong Tác vụ Chiến đấu có sử dụng tên lửa (3)

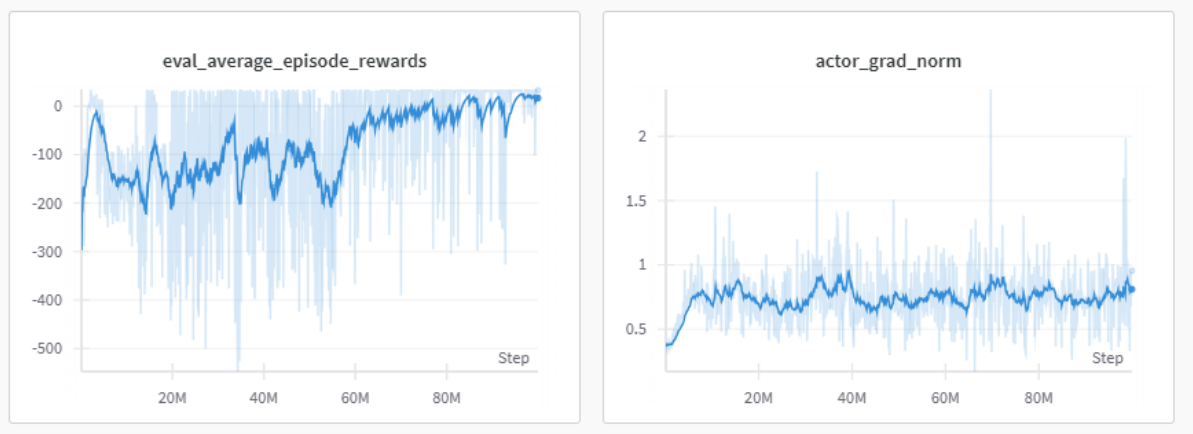
phương pháp này là cải thiện khả năng học tập của mỗi tác tử bằng cách cho chúng tiếp xúc với các tình huống đa dạng hơn và phức tạp hơn khi tương tác với các tác tử khác.

Trong quá trình học, các tác tử học được một chiến lược tốt trong từng giai đoạn nhưng lại không tốt trong giai đoạn tiếp theo, dần dần chiến lược được cải thiện. Khi học tới gần 100M steps, chiến lược học được có xu hướng hội tụ. Tuy nhiên vẫn có những dao động nhỏ, có thể các dao động này sẽ bị triệt tiêu khi training với số bước lớn hơn.

- Thuật toán: MAPPO
- Siêu tham số: 100M steps, 1041 episodes, 640 FPS
- Thời gian training: 48h



Hình 3.9. Kết quả training trong Tác vụ Chiến đấu đa tác tử có sử dụng tên lửa (1)



Hình 3.10. Kết quả training trong Tác vụ Chiến đấu đa tác tử có sử dụng tên lửa (2)

TÀI LIỆU THAM KHẢO

Tiếng Việt

Tiếng Anh

[1] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov, "Proximal Policy Optimization Algorithms", , 2017.

[2] Du, W., Ding, S., "A survey on multi-agent deep reinforcement learning: from the perspective of challenges and applications", *SArtif Intell Rev 54, 3215–3238*, 2021.

[3] Richard Sutton and Andrew G. Barto , "Reinforcement Learning: An Introduction", 2020.
[Online]. Available: $http://incompleteideas.net/book/RLbook2020.pdf$

[4] Long-Ji Lin, "Reinforcement learning for robots using neural networks.", *Technical report, DTIC Document*, 1993.