HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



Bùi Hữu Thành

NGHIÊN CỨU PHƯƠNG PHÁP PHÂN PHỐI TÁC VỤ CHO XE TỰ HÀNH TRÊN NỀN TẢNG ĐIỆN TOÁN BIÊN VÀ HỌC TĂNG CƯỜNG SÂU ĐA TÁC TỬ

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN

MÃ SỐ: 8.48.01.04

TÓM TẮT ĐỀ ÁN TỐT NGHIỆP THẠC SĨ

HÀ NỘI - 2025

Đề án tốt nghiệp được hoàn thành tại:

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: TS. Đỗ Thị Liên

Phản biện 1: .....

Phản biện 2: .....

Đề án tốt nghiệp sẽ được bảo vệ trước Hội đồng chấm đề án tốt nghiệp thạc sĩ tại Học viện Công nghệ Bưu chính Viễn thông

Vào lúc: ..... giờ ..... ngày ..... tháng ..... năm .....

Có thể tìm hiểu đề án tốt nghiệp tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông

MỞ ĐẦU

Sự phát triển của xe tự hành đòi hỏi khả năng xử lý dữ liệu thời gian thực với độ trễ thấp, trong khi mô hình điện toán đám mây truyền thống còn nhiều hạn chế. Điện toán biên giúp đưa xử lý đến gần nguồn dữ liệu nhưng làm bài toán phân phối và lập lịch tác vụ trở nên phức tạp trong môi trường động và phân tán. Trong bối cảnh đó, Học tăng cường sâu đa tác tử (MADRL) là hướng tiếp cận tiềm năng nhờ khả năng học thích nghi và phối hợp ra quyết định. Đề tài nhằm nghiên cứu và xây dựng mô hình phân phối tác vụ thông minh cho xe tự hành trên nền tảng điện toán biên, hướng tới tối ưu hiệu suất và khả năng phản hồi thời gian thực.

Chương 1. Tổng quan về bài toán phân phối tác vụ cho xe tự hành trên nền tảng điện toán biên

Chương này trình bày tổng quan về các nền tảng công nghệ và xu hướng nghiên cứu liên quan đến việc phân phối tác vụ cho xe tự hành trong môi trường điện toán biên. Trong chương này, tôi sẽ giới thiệu khái quát về hệ thống xe tự hành và các hướng nghiên cứu hiện nay về

tối ưu hóa xử lý tác vụ, đảm bảo độ trễ thấp và nâng cao độ tin cậy cho xe tự hành sẽ được phân tích nhằm làm rõ bối cảnh khoa học của bài toán. Đồng thời, tôi trình bày khái quát về bài toán phân phối nhiệm vụ cho xe tự hành trên nền tảng điện toán biên. Bên cạnh đó, chương này cũng trình bày vai trò của điện toán biên và những ưu điểm nổi bật của nó so với điện toán đám mây truyền thống trong việc hỗ trợ xử lý thời gian thực cho xe tự hành. Trên cơ sở đó, tôi mô tả mô hình kiến trúc hệ thống phân phối tác vụ trên nền tảng điện toán biên đang được áp dụng và các thách thức của bài toán.

Trong mục 1.1, tôi sẽ trình bày về hệ thống xe tự hành và bài toán phân phối nhiệm vụ cho một nhóm xe tự hành trên nền tảng điện toán biên. Xe tự hành là công nghệ đột phá giúp nâng cao an toàn giao thông và tối ưu hoạt động đô thị nhờ khả năng tự nhận thức, tương tác và ra quyết định trong môi trường giao thông phức tạp. Do hạn chế về năng lượng và năng lực tính toán của phương tiện, việc dỡ tải sang điện toán biên là cần thiết nhưng đặt ra thách thức lớn trong quản lý và phân bổ tài nguyên trong môi trường giao thông động.

Nghiên cứu này tập trung giải quyết bài toán phân phối nhiệm vụ cho các xe tự hành. Có thể đơn giản hóa bài toán thành dạng IPO như sau

Input	Thông tin n nhiệm vụ Thông tin k phương tiện Thông tin các MEC Server trên bản đồ
Process	Một thuật toán lập lịch n nhiệm vụ cho k phương tiện
Out put	Kết quả phân chia n nhiệm vụ, mỗi nhiệm vụ được phân công cho 1 phương tiện kèm thứ tự thực hiện của phương tiện đó.

Tiếp theo, trong mục 1.2 tôi đã nếu được khái quát về Điện toán biên và điện toán đám mây. Từ đó cũng liệt kê được những điểm giống và khác nhau giữa hai mô hình này. Cuối cùng, tôi cũng đã phân tích những điểm mạnh mà điện toán biên mang lại cùng khả năng phối hợp giữa hai mô hình trong các nghiên cứu tương lai.

Mục 1.3 trình bày Mô hình kiến trúc hệ thống phân phối tác vụ cho xe tự hành trên nền tảng điện toán biên. Kiến trúc dịch vụ trong mạng phương tiện hỗ trợ MEC

gồm tầng End, Edge và Cloud. Tầng End bao gồm các xe tự hành hoặc thiết bị người dùng, nơi sinh ra dữ liệu và thực hiện các tác vụ nhẹ, yêu cầu phản hồi tức thời. Tầng Edge đặt các máy chủ MEC gần người dùng, chịu trách nhiệm xử lý các tác vụ có độ trễ thấp, tính toán cục bộ và giảm băng thông trước khi gửi dữ liệu lên cloud. Tầng Cloud đảm nhận lưu trữ dữ liệu lớn, xử lý các tác vụ phức tạp và huấn luyện mô hình, sau đó nạp kết quả xuống Edge và End để hỗ trợ hoạt động hiệu quả của hệ thống.

Mục 1.4 tiếp theo trình bày các hướng tiếp cận hiện nay cho bài toán phân phối tác vụ cho xe tự hành. Đây là một bài toán tối ưu hóa phức tạp. Các nghiên cứu trong lĩnh vực này tập trung vào ba hướng chính bao gồm nghiên cứu về mục tiêu tối ưu, phương pháp giải quyết bài toán tối ưu hóa và hệ thống mô hình phân tải tác vụ. Chúng ta có thể nghiên cứu đơn mục tiêu như tổng số nhiệm vụ hoàn thành, độ trễ, lợi nhuận, chi phí, ... hoặc kết hợp các mục tiêu trên. Về phương pháp, chúng ta có thể sử dụng các thuật toán tìm kiếm nhưng chủ yếu hiện nay các nghiên cứu đều sử dụng sức mạnh của học tăng cường sâu và các thuật toán tối ưu hóa phức tạp để giải quyết vấn đề

năng động và không chắc chắn mà xe tự hành mang lại.

Trong nghiên cứu này, tôi tập trung vào nghiên cứu phương pháp giải quyết bài toán tối ưu hóa và chọn Học tăng cường sâu đa tác tử (MADRL) bởi khả năng xử lý hiệu quả tính động, tính phân tán và sự phức tạp của hệ thống giao thông thông minh.

Phần cuối chương, tôi trình bày về những thách thức và vấn đề còn tồn tại trong triển khai hệ thống phân phối tác vụ cho xe tự hành trên nền tảng điện toán biên. Dù các phương pháp tối ưu phân phối tác vụ hiện tại đã đạt được nhiều thành tựu đáng kể nhưng đa phần các nghiên cứu đều chưa xét đến yếu tố thời gian hoàn thành nhiệm vụ thay đổi. Đồng thời, các phương pháp cũng chưa mô hình được sát với thực tế môi trường phức tạp trong bài toán phân phối tác vụ cho xe tự hành. Do đó, cần thiết có những mô hình tối ưu hóa mới trong phân phối tác vụ cho xe tự hành, có khả năng đánh giá giá trị thời gian của tác vụ để đưa ra quyết định phân phối hợp lý và thích ứng hơn trong môi trường động và phân tán như VEC.

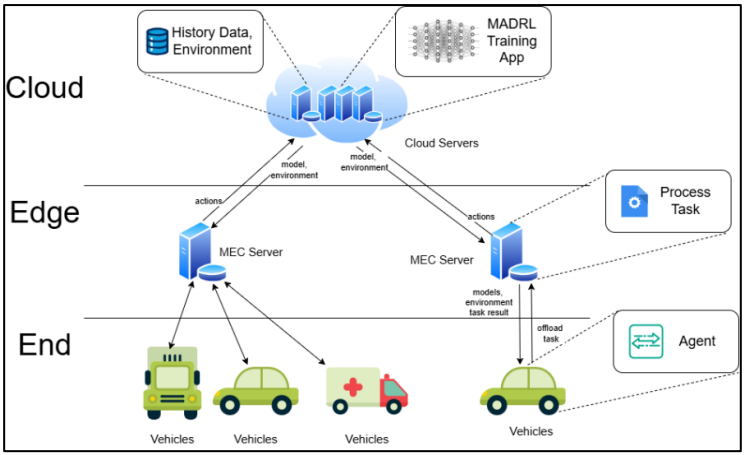
Chương 2. Đề xuất phương pháp phân phối tác vụ

cho xe tự hành trên nền tảng điện toán biên và học tăng cường sâu đa tác tử

Trong chương này, tôi sẽ trình bày phương pháp đề xuất nhằm giải quyết bài toán phân phối tác vụ cho xe tự hành trên nền tảng điện toán biên bằng cách ứng dụng học tăng cường sâu đa tác tử. Trước hết, tôi sẽ mô tả kiến trúc hệ thống đề xuất và mô hình hóa bài toán tối ưu hóa. Tiếp theo là môi trường học tăng cường với các đối tượng như phương tiện, MEC Server, bản đồ giao thông, nhiệm vụ, tác vụ,... Đồng thời, môi trường cũng đề cập đến các hành động của các tác tử, các trạng thái trong môi trường và đặc biệt là hàm phần thưởng để đánh giá hiệu quả của các chiến lược học tăng cường. Cuối chương tôi sẽ trình bày kiến trúc mạng nơ-ron, quy trình huấn luyện mô hình MADRL và cách tích hợp chúng vào cơ chế lập lịch để tối ưu hiệu năng phân phối tác vụ.

2.1 Thiết kế kiến trúc hệ thống đề xuất

Kiến trúc đề xuất cho hệ thống phân phối tác vụ của xe tự hành được thiết kế theo mô hình ba lớp, bao gồm Cloud, Edge và End.



Môi trường trong bài toán phân tải tác vụ cho nhóm xe tự hành được xây dựng dựa trên bốn thành phần cơ bản bao quát các yếu tố quyết định hoạt động của hệ thống. Thứ nhất là các nhiệm vụ với giá trị lợi ích và giới hạn thời gian hoàn thành khác nhau. Kế đến là các xe tự hành đóng vai trò agent đưa ra quyết định xử lý cục bộ hoặc gửi tác vụ lên MEC server. Tiếp theo là các MEC server có năng lực tính toán hạn chế đảm nhận xử lý các tác vụ nhận được. Cuối cùng là các Cloud server chủ yếu phục vụ huấn luyện mô hình và phân phối chính sách tối ưu xuống hệ thống.

Trong mô hình này, lớp Cloud chịu trách nhiệm triển khai các ứng dụng non-RealTime, tức là những tác

vụ không yêu cầu phản hồi tức thời như phân tích dữ liệu lớn, huấn luyện mô hình trí tuệ nhân tạo. Lớp Edge được triển khai gần với người dùng hơn chịu trách nhiệm xử lý các tác vụ Near-RealTime với độ trễ từ 10ms đến 1 giây. Lớp End bao gồm các xe tự hành và cơ sở hạ tầng mạng trực tiếp phục vụ xe tự hành.

Tôi thiết kế môi trường mô phỏng có 5 xe tự hành ở 5 vị trí khác nhau trên bản đồ. Có 25 nhiệm vụ cần các xe tự hành xử lý. Mỗi nhiệm vụ có giá trị lợi ích và giới hạn thời gian riêng, đồng thời có thể phụ thuộc lẫn nhau. Các xe tự hành lần lượt chọn các nhiệm vụ cho đến khi tất cả các nhiệm vụ được phân công đều cho các xe. Sau khi chọn nhiệm vụ, xe sẽ tìm đường đi ngắn nhất tới vị trí nhiệm vụ, kết hợp với các điều kiện giao thông được mô phỏng trên bản đồ. Trên đường đi, các xe sẽ xử lý các tác vụ nhỏ phát sinh (ví dụ: kiểm tra tình trạng cảm biến, thu thập dữ liệu môi trường, cập nhật bản đồ địa phương hoặc xử lý cảnh báo giao thông) bằng cách xử lý tại xe hoặc dỡ tải tác vụ lên các MEC Server xung quanh.

2.2 Mô hình toán học cho phân phối tác vụ trong kiến trúc đề xuất

2.2.1 Phát biểu bài toán phân phối tác vụ

Một nhiệm vụ có thể được biểu diễn bởi một bộ 5 thành phần như sau

$$M_i \triangleq \langle s_i, e_i, T_i, P_i, M_i^- \rangle \quad (2.1)$$

Trong đó M_i là nhiệm vụ thứ i trong danh sách $M(\tau)$; s_i là điểm bắt đầu nhận nhiệm vụ, e_i là điểm kết thúc nhiệm vụ; T_i là thời hạn hoàn thành của nhiệm vụ; P_i là lợi nhuận cơ bản của phương tiện khi hoàn thành nhiệm vụ và $M_i^- \in M(\tau)$ là tập hợp các nhiệm vụ tiền nhiệm mà phải hoàn thành trước M_i. Các nhiệm vụ có $M_i^- = \emptyset$ có thể thực hiện ngay lập tức mà không cần chờ đợi. Trên đường thực hiện nhiệm vụ M_i, các xe tự hành cần thực hiện các tác vụ $t_{i,j} \in K_i$. Đầu ra của bài toán là xác định nghiệm bao gồm danh sách chỉ số id của xe thực hiện và thứ tự thực hiện trên xe. Như vậy, tập nghiệm này cần thỏa mãn các điều kiện

- Mỗi nhiệm vụ chỉ được gán cho duy nhất một phương tiện (2.5).
- Danh sách nhiệm vụ của một phương tiện không có hai nhiệm vụ nào có cùng thứ tự ưu tiên (2.6)

- Trong danh sách các nhiệm vụ, các nhiệm vụ tiền nhiệm phải có thứ tự ưu tiên nhỏ hơn các nhiệm vụ phụ thuộc vào nó (2.7)

Chiến lược dỡ tải tác vụ của phương tiện được xây dựng dựa trên việc cân nhắc giữa khả năng xử lý cục bộ và điều kiện của MEC server tùy thuộc chiến lược nào tốn ít thời gian hơn. Thêm vào đó, mỗi lần thực hiện dỡ tải tác vụ lên MEC Server sẽ gây tiêu tốn thêm chi phí cho cả quá trình thực hiện nhiệm vụ. Thời gian thực hiện một tác vụ sẽ được tính dựa vào thời gian dỡ tải tác vụ tổng cộng trên MEC Server t_off và thời gian tự xử lý trên phương tiện t_veh

$$\text{t_proc}_j = \min\left(, \text{t_veh}_j \right) \qquad (2.11)$$

Thời hạn hoàn thành một nhiệm vụ M_i đơn lẻ được tính bằng tổng thời gian hoàn thành các tác vụ khi thực hiện nhiệm vụ

$$\delta_i = \sum_{j|t_{i,j} \in K_i} \text{t_proc}_j \qquad (2.12)$$

Từ việc tính toán này, ta sẽ thu được công thức tính

tổng thời gian để hoàn thành nhiệm vụ M_i là tổng thời gian hoàn thành các nhiệm vụ M_i và nhiệm vụ tiền nhiệm của M_i đồng thời phải xử lý các nhiệm vụ có độ ưu tiên cao hơn trên các phương tiện thực hiện các nhiệm vụ trên. Để có lợi nhuận sau khi thực hiện các nhiệm vụ, điều kiện cần thiết trong trường hợp này là

$$\delta_i(\tau) \leq T \qquad (2.13b)$$

Ngoài ra, Lợi nhuận R_i còn lại sau khi phương tiện hoàn thành nhiệm vụ trừ đi chi phí phát sinh chi phí để thực hiện nhiệm vụ cũng cần nhỏ hơn tổng lợi ích mà nhiệm vụ thu được. Từ đó ta có điều kiện

$$R_i \geq 0 \qquad (2.18)$$

Cuối cùng, bài toán tối ưu hóa việc gán nhiệm vụ và dỡ tải tác vụ cho xe tự hành với mục tiêu tối đa hóa số lượng nhiệm vụ hoàn thành trước thời gian T có thể được mô hình hóa như sau

$$\mathcal{P}_1 : \max_{N(\tau)} \sum\nolimits_{M_i \in M(\tau)} \mathbf{1}_{\{\delta_i(\tau) < T\}} \qquad (2.19a)$$

$$s.t. (2.5), (2.6), (2.7), (2.13b), (2.18) \qquad (2.19b)$$

Hay có thể phát biểu ngắn gọn thành bài toán tối đa hóa số lượng nhiệm vụ được hoàn thành trong thời gian τ thỏa mãn các điều kiện trong các công thức (2.5),(2.6),(2.7),(2.13b),(2.18b).

2.2.2 Tối ưu hóa dựa trên mô hình học tăng cường sâu đa tác tử

Trong mục này, tôi đã so sánh Học tăng cường sâu đơn tác tử và đa tác tử từ đó rút ra kết luận mô hình đa tác tử phù hợp với bài toán phân phối nhiệm vụ cho xe tự hành. Trong các hệ thống điện toán biên cho phương tiện, tính phân tán, tính động cao và sự tương tác phức tạp giữa các phương tiện khiến mô hình tác tử đơn lẻ không còn phù hợp. Ngược lại, học tăng cường sâu đa tác tử nổi lên như một giải pháp hiệu quả nhờ khả năng học chính sách hợp tác hoặc cạnh tranh giữa nhiều tác tử đồng thời, giúp mô phỏng sát thực tế hành vi của các phương tiện trong mạng.

Với việc sử dụng Học Tăng cường Đa Tác tử làm hướng tiếp cận chính cho bài toán, bước quan trọng tiếp theo là thiết kế hàm phần thưởng – thành phần cốt lõi quyết định hành vi học của từng tác tử. Hàm phần thưởng

tôi thiết kế phụ thuộc vào các thành phần

- Phần thưởng từ giá trị nhiệm vụ và hoàn thành sớm TCR
- Chi phí dỡ tải tác vụ lên MEC server OC
- Chi phí di chuyển để hoàn thành nhiệm vụ TC
- Lợi ích từ việc hoàn thành các nhiệm vụ phụ thuộc DR

Hàm phần thưởng tổng quát

Từ 4 công thức trên, ta có được công thức tính hàm phần thưởng R_i cho một nhiệm vụ m_i như sau

$$R_i = \mathbf{1}_{\{\text{complete}_i\}} \begin{bmatrix} \text{profit}_i + r_1 \cdot TCR_i - \\ r_2 \cdot OC_i - r_3 \cdot TC_i + r_4 \cdot DR_i \end{bmatrix} \quad (2.24)$$

Các hệ số r_1, r_2, r_3, r_4 là các hệ số điều chỉnh hàm trong phần thưởng.

Tổng phần thưởng R_{total} được tính bằng tổng các phần thưởng R_i của từng nhiệm vụ i mà agent hoặc hệ thống đã hoàn thành. Việc này giúp đánh giá hiệu quả tổng thể của chiến lược phân phối và xử lý tác vụ, đồng thời làm cơ sở để huấn luyện agent tối ưu hóa lựa chọn nhiệm

vụ và phương án xử lý.

2.3 Xây dựng mô hình học tăng cường sâu đa tác tử cho bài toán tối ưu hóa phân phối tác vụ trong xe tự hành

2.3.1 Kiến trúc mạng nơ-ron trong mô hình

Trong phần này, tôi đã so sánh 3 thuật toán phổ biến trong học tăng cường là DQN, PPO và A2C để đưa ra kết luận DQN phù hợp với bài toán phân phối tác vụ biểu nhờ khả năng học nhanh và hiệu quả trong những môi trường có phần thưởng được phản hồi ngay sau mỗi hành động.

Trong nghiên cứu này, tôi sử dụng một mạng Nơ-ron được thiết kế theo cấu trúc nhiều lớp nhằm học và xấp xỉ hàm giá trị hành động $Q(s, a)$ một cách hiệu quả trong không gian trạng thái có số chiều lớn. Kiến trúc mạng bao gồm một tầng đầu vào, ba tầng ẩn và một tầng đầu ra, sử dụng các hàm kích hoạt phi tuyến SELU và ELU để tăng tính ổn định và khả năng biểu diễn của mô hình.

Cụ thể, tầng đầu vào nhận vector trạng thái có kích thước bằng số chiều của trạng thái. Ba tầng ẩn được xây

dựng với kích thước giảm dần, lần lượt bằng 130%, 60% và 20% số chiều của trạng thái đầu vào. Kiến trúc này cho phép tác tử Double DQN học biểu diễn đặc trưng tốt hơn, đồng thời giảm sai lệch ước lượng Q-value, hỗ trợ tác tử đưa ra quyết định tối ưu.

Trong mô hình nhiều tác tử, mỗi xe tự hành được trang bị một mạng nơ-ron riêng biệt để ước lượng giá trị hành động, và do đó cần duy trì một bộ nhớ kinh nghiệm độc lập và bộ nhớ kinh nghiệm chung ở mức toàn cục để tích hợp thông tin thu thập từ tất cả các tác tử.

2.3.2 Quy trình huấn luyện mô hình

Quy trình huấn luyện tác tử Double DQN được thực hiện theo vòng lặp tương tác liên tục giữa tác tử và môi trường. Mỗi tác tử quan sát trạng thái, lựa chọn hành động, nhận phần thưởng rồi cập nhật mạng Q thông qua cơ chế Double Q-Learning nhằm giảm sai lệch ước lượng giá trị Q. Trong mỗi episode, trainer sẽ kích hoạt hàm thực hiện một episode, tại đó toàn bộ quá trình tương tác giữa các tác tử và môi trường được thực hiện: tác tử quan sát trạng thái, lựa chọn hành động, nhận phần thưởng và cập nhật dữ liệu phục vụ học tập. Sau khi một episode kết thúc,

hệ thống ghi nhận điểm thưởng của episode đó vào lịch sử và tính toán giá trị trung bình dựa trên một cửa sổ trượt gồm các episode gần nhất. Cách tiếp cận này giúp đánh giá hiệu năng của mô hình một cách ổn định và giảm thiểu tác động của các dao động ngẫu nhiên trong phần thưởng.

Song song với quá trình đánh giá, mô hình cũng được lưu định kỳ sau một số lượng episode xác định nhằm đảm bảo khả năng theo dõi tiến độ và phục hồi khi cần thiết. Các số liệu thống kê và biểu đồ theo dõi kết quả huấn luyện được cập nhật đồng thời, giúp giám sát quá trình hội tụ của chính sách.

Chương 3. Kiểm nghiệm và đánh giá mô hình đề xuất trong tối ưu hóa phân phối tác vụ cho xe tự hành trên nền tảng điện toán biên

Chương này tôi sẽ trình bày toàn bộ quá trình thực nghiệm nhằm đánh giá hiệu quả của phương pháp phân phối tác vụ dựa trên học tăng cường sâu đa tác tử. Trước tiên, chương này mô tả cách xây dựng dữ liệu mô phỏng và thiết lập các tham số huấn luyện cho mô hình, bao gồm cấu hình môi trường, đặc tính tác vụ và các tham số tối ưu

hóa. Tiếp theo, kết quả huấn luyện mô hình sẽ được phân tích chi tiết, cùng với việc nạp mô hình vào từng tác tử và tiến hành chạy thử nghiệm so sánh với các phương pháp lập lịch truyền thống để làm nổi bật ưu thế của MADRL. Cuối cùng, tôi đề cập đến khả năng triển khai thực tế của hệ thống, bao gồm các yêu cầu, thách thức và định hướng ứng dụng.

3.1 Dữ liệu thực nghiệm

3.1.1 Nhiệm vụ và tác vụ

Trong mô hình thực nghiệm, hệ thống được khởi tạo với 25 nhiệm vụ độc lập, mỗi nhiệm vụ đại diện cho một yêu cầu vận chuyển hoặc xử lý mà xe tự hành cần thực hiện. Các nhiệm vụ này được phân phối cho 5 xe tự hành, và bài toán đặt ra là xác định chiến lược lập lịch tối ưu để đảm bảo các nhiệm vụ được hoàn thành đúng thứ tự, đúng thời gian và đạt lợi nhuận cao nhất. Mỗi nhiệm vụ được mô tả dưới dạng một cấu trúc dữ liệu bao gồm các thành phần chính:

- Điểm bắt đầu (start_point): tọa độ ví trí bắt đầu nhiệm vụ.

- Điểm kết thúc (end_point): tọa độ vị trí đích đến để hoàn thành nhiệm vụ.
- Danh sách nhiệm vụ tiền nhiệm (dependencies): tập các nhiệm vụ bắt buộc phải hoàn thành trước khi nhiệm vụ hiện tại có thể được thực thi.
- Lợi nhuận (profit): Lợi ích nhận được sau khi nhiệm vụ hoàn tất

3.1.2 Xe tự hành và tác tử

Để mô phỏng hành vi của xe tự hành trong hệ thống giao thông thông minh, tôi xây dựng một lớp đối tượng đại diện cho mỗi phương tiện. Các phương tiện này được xem như một tác tử thông minh có khả năng ra quyết định trong môi trường động. Phương tiện có thể lựa chọn nhiệm vụ cần thực hiện, xử lý các tác vụ nhận được trên đường tại chính thiết bị (local), hoặc dỡ tải các tác vụ lên các máy chủ MEC nhằm giảm độ trễ và tối ưu thời gian xử lý.

MEC Server được mô phỏng bằng một lớp có vai trò cung cấp tài nguyên xử lý bổ sung cho các phương tiện tự hành khi thực hiện dỡ tải tác vụ. Mỗi đối tượng MEC được gắn với một vị trí cố định trên bản đồ. Mỗi MEC có

một tham số quan trọng là tần số CPU, biểu thị năng lực xử lý tính toán của máy chủ. Tần số CPU (đơn vị Hz) quyết định tốc độ thực hiện các tác vụ được dỡ tải từ phương tiện.

3.2 Cài đặt thực nghiệm

Mỗi phương tiện tự hành được mô hình hóa như một tác tử trong môi trường học tăng cường đa tác tử. Mỗi episode mô phỏng tương ứng với một khoảng thời gian xác định trong môi trường giao thông, trong đó các phương tiện thực hiện các nhiệm vụ di chuyển và xử lý tác vụ. Mỗi phương tiện có giới hạn số lượng nhiệm vụ tối đa trong một episode, đảm bảo cân bằng tải và phản ánh khả năng xử lý thực tế. Quy tắc dừng huấn luyện được thiết lập dựa trên điểm thưởng trung bình, tức là khi các phương tiện đạt hiệu năng tối ưu theo chỉ số này, quá trình huấn luyện sẽ được kết thúc sớm để tiết kiệm tài nguyên tính toán.

Hệ thống được mô phỏng trên server chạy hệ điều hành CentOS 7.9 với CPU Intel Xeon Gold 6226R 64 lõi (2 socket × 16 lõi, hỗ trợ Hyper-Threading) và bộ nhớ đệm L1/L2/L3 lần lượt 1 MiB/32 MiB/44 MiB. Hệ thống trang

bị 2 GPU NVIDIA Quadro RTX 6000 với 24 GB VRAM mỗi card, driver 550.144.03 và CUDA 12.4, sẵn sàng tăng tốc các tác vụ tính toán. Cấu hình này đủ mạnh để thực thi các mô hình học tăng cường sâu đa tác tử cho xe tự hành.

3.3 Đánh giá thực nghiệm

3.3.1 Các chỉ số đánh giá

Trong nghiên cứu này, với kết quả từ R lần mô phỏng, tại lần thứ i ta thu được tổng lợi nhuận của các tác tử và tổng số nhiệm vụ hoàn thành lần lượt là P_i và C_i. Công thức để đánh giá bao gồm

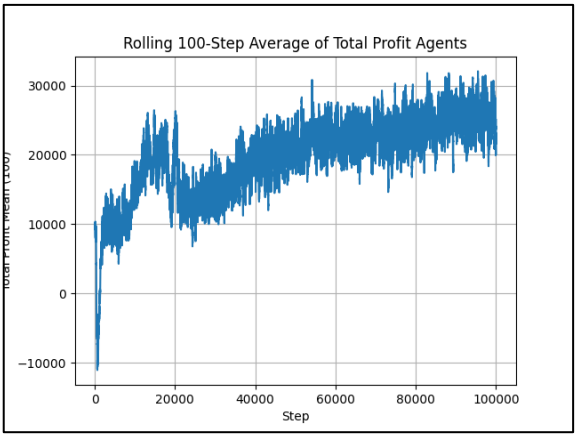
$$\bar{P} = \frac{1}{R}\sum_{1}^{R} P_i \, , \bar{C} = \frac{1}{R}\sum_{1}^{R} C_i \qquad (3.1)$$

Trong đó $\bar{P}$ và $\bar{C}$ lần lượt là các giá trị trung bình tổng lợi nhuận của các tác tử và trung bình tổng số nhiệm vụ hoàn thành. Ta sẽ so sánh các giá trị này với nhau

3.3.2 Kết quả thực nghiệm

Hình ảnh dưới đây trình bày kết quả huấn luyện của mô hình MA-DDQN, cho thấy sự hội tụ và ổn định của các agent cùng với xu hướng tăng của phần thưởng tối

đa trong suốt 100,000 episode. Đường màu xanh thể hiện giá trị lợi nhuận tổng thể đạt được tại mỗi bước huấn luyện. Ta có thể quan sát rằng sau giai đoạn đầu với sự dao động lớn, đường tổng lợi nhuận trung bình (Total Profit) liên tục tăng và đạt trạng thái ổn định minh chứng cho khả năng phối hợp giữa các agent trong việc tối ưu hóa tổng lợi nhuận của hệ thống.



Hình 3.1. Kết quả huấn luyện mô hình MA-DDQN

Trong các thực nghiệm, tôi so sánh mô hình MA-DDQN với hai thuật toán cơ bản.

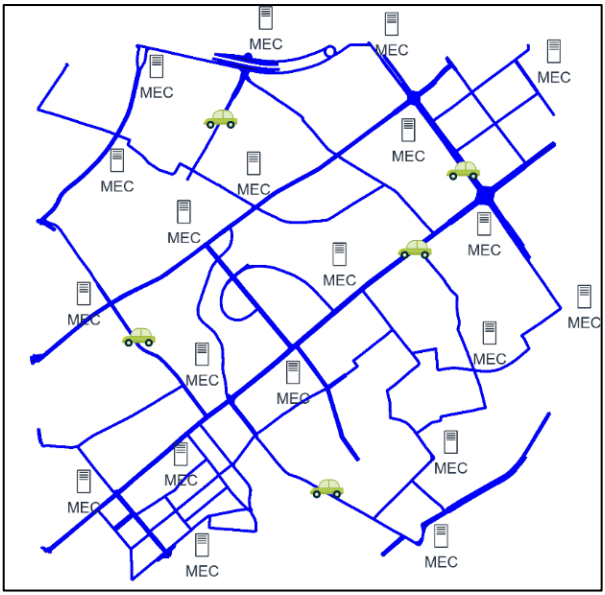
- Thuật toán chọn ngẫu nhiên (Random Selection) thực hiện phân phối ngẫu nhiên các nhiệm vụ cho

các xe mà không dựa trên bất kỳ tiêu chí tối ưu nào

- Thuật toán Max-Profit thực hiện theo chiến lược tham lam: trước tiên sắp xếp các nhiệm vụ theo thứ tự dựa trên các phụ thuộc giữa nhiệm vụ, sau đó phân chia lần lượt cho các xe.

Để đánh giá hiệu quả của các thuật toán, tôi thực hiện mô phỏng trên 100 môi trường ngẫu nhiên khác nhau. Từ kết quả thực nghiệm ta có thể thấy có thể thấy đa số các lần chạy thì MA-DDQN cho kết quả vượt trội so với các phương pháp truyền thống (87/100 lần). Kết quả trung bình cũng cho thấy số lượng nhiệm vụ hoàn thành trung bình cao hơn 26.17% so với Max-Profit; tổng lợi nhuận trung bình cũng cao hơn 26.17%. **Xây dựng thử nghiệm hệ thống phân phối tác vụ cho xe tự hành trên nền tảng điện toán biên**

Trong thử nghiệm, tôi xây dựng một mô hình hệ thống phân phối nhiệm vụ cho 5 xe tự hành và 25 nhiệm vụ khác nhau trên nền tảng điện toán biên (MEC). Hệ thống bản đồ được xây dựng từ dữ liệu thực tế lấy từ OSM với trung tâm là Học viện Công nghệ Bưu Chính Viễn thông, bán kính 2.5km.



KẾT LUẬN CHUNG

Trong đề án, tôi đã tiến hành nghiên cứu cơ sở lý thuyết về điện toán biên (MEC) và các nguyên tắc phân phối tác vụ cho xe tự hành. Quá trình nghiên cứu bao gồm việc phân tích các yếu tố liên quan như số lượng xe, số lượng nhiệm vụ, phụ thuộc giữa các nhiệm vụ, cũng như các chỉ số đánh giá hiệu quả phân phối. Những nghiên cứu này đã giúp làm rõ bài toán phân phối tác vụ trong môi trường đa tác vụ phức tạp, đồng thời làm cơ sở cho việc phát triển các giải pháp tối ưu hóa. Trên cơ sở đó, đề án

đã đề xuất và triển khai thuật toán MA-DDQN để phân phối nhiệm vụ giữa các xe tự hành trên nền tảng MEC. Thuật toán này cho phép các agent phối hợp chặt chẽ với nhau, tối ưu hóa cả số lượng nhiệm vụ hoàn thành và tổng lợi nhuận hệ thống. Kết quả thực nghiệm cho thấy MA-DDQN vượt trội hơn hẳn so với các phương pháp cơ bản như Max-Profit và Random Selection, chứng minh hiệu quả trong việc phân phối tác vụ thông minh và khả năng mở rộng cho môi trường nhiều xe, nhiều nhiệm vụ.