

# Tối Ưu Hóa Tín Hiệu Đèn Giao Thông Sử Dụng Thuật Toán Học Tăng Cường

Mai Đức Thuận<sup>[19522316]</sup> and Phạm Đức Thế<sup>[19522253]</sup>

Trường Đại Học Công Nghệ Thông Tin, Thành Phố Hồ Chí Minh, Việt Nam  
<https://www.uit.edu.vn/>

**Tóm tắt nội dung** Vấn đề ùn tắc tại các nút giao thông đang là một trong những vấn đề nan giải ở cả trong và ngoài nước. Bài báo này trình bày một phương pháp cải tiến điều khiển tín hiệu đèn giao thông nhằm đạt hiệu quả cao hơn. Phương pháp này sử dụng thuật toán học tăng cường để mô phỏng và tối ưu lưu lượng phương tiện tại một nút giao thông cụ thể.

**Keywords:** Tín hiệu giao thông · Tối ưu hóa · Máy học · SUMO Simulation · Học tăng cường.

## 1 Giới thiệu

Chúng ta biết rằng, vấn đề ùn tắc giao thông tại các điểm nút giao thông, giao cắt đồng mức trong các thành phố lớn của nước ta và trên thế giới ngày càng nghiêm trọng. Để giải quyết vấn đề này các nhà khoa học và các nhà quản lý đã nghiên cứu và áp dụng nhiều biện pháp điều khiển nhằm làm giảm thiểu sự ùn tắc tại các nút giao thông. Hiện tại, tại các điểm nút giao cắt đồng cấp được điều khiển bằng đèn tín hiệu hoặc cảnh sát giao thông. Điều khiển bởi cảnh sát mang lại hiệu quả cao, bởi nó có tính mềm dẻo và linh hoạt, thời gian của các pha được người cảnh sát xác định phụ thuộc vào mật độ phương tiện hiện có tại nút. Tuy nhiên với phương pháp này tốn nhiều nhân lực và chi phí cao. Điều khiển các nút giao thông bằng đèn tín hiệu cũng đã mang lại hiệu quả với những nút có mật độ phương tiện tham gia vừa phải. Các phương pháp điều khiển bằng đèn tín hiệu hiện tại còn nhiều nhược điểm. Phương pháp điều khiển đèn với chu kỳ cố định rất cứng nhắc, có nhiều thời gian chết. Đèn xanh vẫn được bật ngay cả khi trên pha đó không có phương tiện, trong khi pha khác có thể có nhiều phương tiện thì vẫn phải chờ.

Một số nhà khoa học nghiên cứu và ứng dụng công nghệ thông tin vào việc điều khiển, thực hiện điều khiển chu kỳ đèn động dựa vào mật độ phương tiện hiện tại xung quanh nút. Giải pháp này đã đem lại hiệu quả cao hơn giải pháp điều khiển đèn theo chu kỳ cố định. Tuy nhiên giải pháp này mới chỉ sử dụng các thông tin mang tính cục bộ về mật độ các phương tiện hiện tại xung quanh nút, thông tin này chỉ có tác dụng đối với trạng thái điều khiển hiện tại. Trong thực tế mật độ tại các nút phụ thuộc nhiều vào lưu lượng các phương tiện sẽ đi

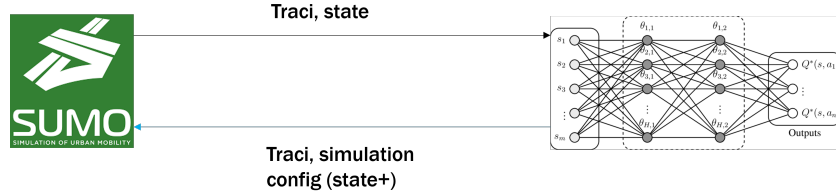
tới nút, đây là một yếu tố ảnh hưởng tới mật độ phương tiện tại nút trong các trạng thái tiếp theo.

Trong bài báo này đề xuất phương pháp điều khiển động đèn tín hiệu bằng cách kết hợp các thông tin về mật độ phương tiện hiện tại của nút và thông tin dự báo lưu lượng phương tiện sẽ tới nút để đưa ra quyết định điều khiển. Phương pháp sử dụng SUMO simulation [1] để mô phỏng nút giao thông và sử dụng thuật toán Deep Q-learning để đưa ra các hành động tối ưu tiếp theo.

## 2 Phương pháp tiếp cận

### 2.1 Tổng quan

Phương pháp sử dụng SUMO simulation để mô phỏng một ngã tư cùng với thông tin của các phương tiện giao thông. Nó cũng hỗ trợ giao diện điều khiển lưu lượng (Traci), là một API Python xử lý mô phỏng SUMO như một máy chủ và cho phép người dùng thu thập dữ liệu từ mô phỏng lưu lượng hoặc sửa đổi mô phỏng. Sử dụng Traci để xuất các thông tin cần thiết (lưu lượng hiện tại, số phương tiện) dùng làm dữ liệu đầu vào của mô hình thuật toán Deep Q-learning. Sau khi tính toán hoàn tất, mô hình sẽ đưa ra các giá trị đại diện cho hành động tối ưu kế tiếp. Cuối cùng đưa nó vào SUMO bằng Traci để tiếp tục mô phỏng lưu thông và trực quan kết quả. Mô hình được biểu diễn như Hình 1.



Hình 1. Tổng quan kiến trúc của mô hình.

### 2.2 SUMO Simulation

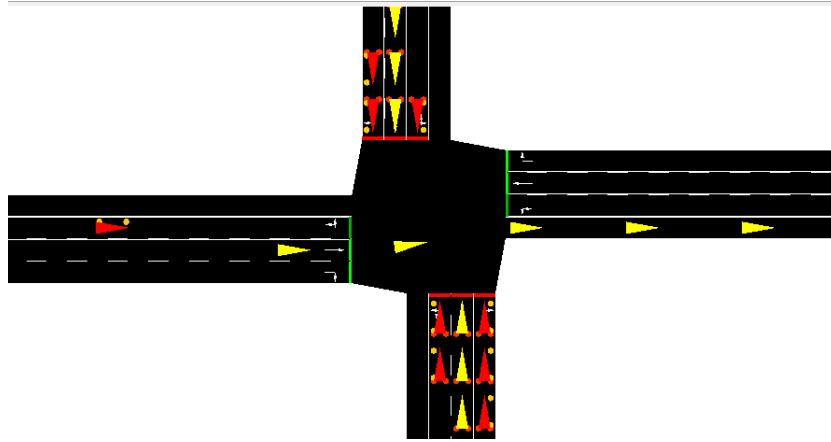
Simulation of Urban Mobility (hay gọi tắt là SUMO) là một trình mô phỏng lưu lượng đa phương thức, cực nhỏ, mã nguồn mở. Nó cho phép người dùng mô phỏng cách một nhu cầu giao thông cụ thể thực hiện trên một mạng lưới đường nhất định. SUMO là một lựa chọn tuyệt vời để mô phỏng giao thông đô thị vì nó có thể thực hiện một phương pháp phân phối giao thông được tối ưu hóa dựa trên loại phương tiện hoặc hành vi của người lái xe. Nó cũng có thể cập nhật lộ trình của phương tiện nếu có bất kỳ tắc nghẽn nào hoặc nếu xảy ra tai nạn, tìm ra phương án định tuyến tốt nhất để nâng cao tính bền vững và khả năng phục hồi của đô thị. Nó cũng hỗ trợ Giao diện điều khiển lưu lượng (viết

tất là TraCI), là một API Python xử lý mô phỏng SUMO như một máy chủ và cho phép người dùng thu thập dữ liệu từ mô phỏng lưu lượng hoặc sửa đổi mô phỏng. Điều này làm cho TraCI trở thành một công cụ hoàn hảo để thực hiện Học tăng cường trong trình mô phỏng SUMO.

Để xây dựng một mạng lưới giao thông trong SUMO thì các nodes và edges là các yếu tố cần thiết. Mỗi mạng SUMO chứa các thông tin liên quan đến lưu lượng truy cập như sau:

- Mỗi edge là một tập hợp các làn đường.
- Vị trí, hình dạng và tốc độ trên mỗi làn.
- Quy định hướng di chuyển.
- Các kết nối giữa các làn tại các nút giao.
- Vị trí của đèn giao thông.

Phương pháp được mô phỏng như trên Hình 2



**Hình 2.** Nút giao thông mô phỏng.

### 2.3 Thuật toán học tăng cường

Học tăng cường (Reinforcement learning) là một lĩnh vực con của học máy, nghiên cứu cách thức một agent trong một môi trường nên chọn thực hiện các hành động nào để cực đại hóa một khoản thưởng (reward) nào đó về lâu dài. Các thuật toán học tăng cường cố gắng tìm một chiến lược ánh xạ các trạng thái của thế giới tới các hành động mà agent nên chọn trong các trạng thái đó.

Môi trường thường được biểu diễn dưới dạng một quá trình quyết định Markov trạng thái hữu hạn (Markov decision process - MDP), và các thuật toán học tăng cường cho ngữ cảnh này có liên quan nhiều đến các kỹ thuật quy hoạch động. Các xác suất chuyển trạng thái và các xác suất thu lợi trong MDP thường là ngẫu nhiên nhưng lại tĩnh trong quá trình của bài toán (stationary over the course of the problem).

Thuật toán học tăng cường gồm 7 khái niệm chính:

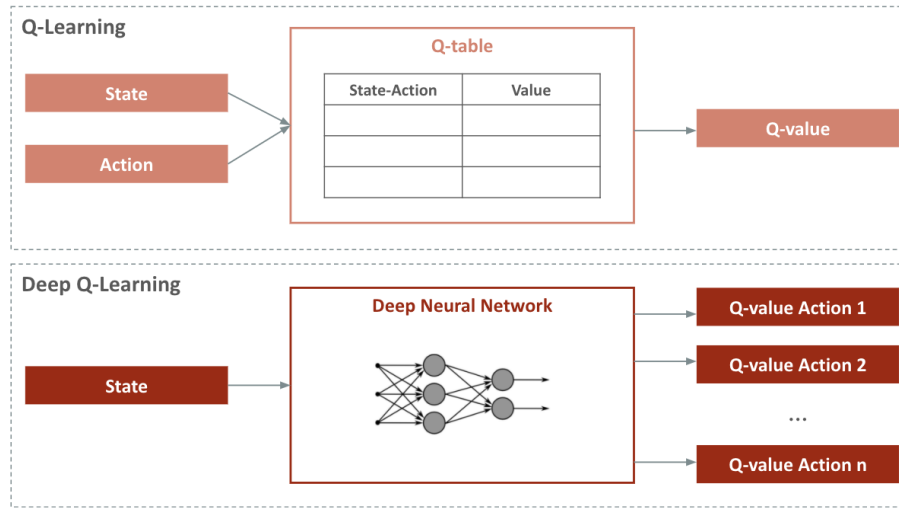
- Agent: Thực thể thực hiện các action ảnh hưởng tới environment.
- Environment: Môi trường đang xét.
- State: Trạng thái hiện tại của environment.
- Action: Hoạt động tác động tới trạng thái của environment.
- Reward: Phần thưởng nhận được khi thực thi action.
- Episode: Một quá trình mô phỏng.
- Policy: Thuật toán sử dụng.

## 2.4 Q-learning và Deep Q-learning

Q-learning [2] là một thuật toán học-tăng-cường off policy ( off policy reinforcement learning algorithm). Nó được gọi là off policy vì thuật toán q learning học từ các hành động được thực hiện ngẫu nhiên, tức là không có trước một policy nào cả. Cụ thể, q-learning học để tìm ra được một chiến lược (policy) có thể tối đa hoá được tổng phần thưởng (maximizes the total reward). Chữ Q đại diện cho chất lượng (Quality). Chất lượng trong trường hợp này là sự biểu thị của độ tốt của việc thực hiện một hành động (action) tới việc làm tăng phần thưởng trong dài hạn.

Để thực hiện thuật toán q-learning, việc đầu tiên ta cần làm là tạo một bảng q (q - table). Q- table là một ma trận có chiều là [trạng thái, hành động] ([state, action]). Chúng ta khởi tạo giá trị 0 cho ma trận này. Sau đó, chúng ta cập nhật và lưu trữ các giá trị q vào q-table sau từng tập (episode - bao gồm tất cả các trạng thái được tính từ lúc tác nhân - agent (trong thuật toán Markov) bắt đầu học cho đến khi agent gặp điểm terminate). Bảng q này trở thành bảng tham chiếu để tác nhân (agent) chọn hành động tốt nhất (best action) dựa trên giá trị q (q value).

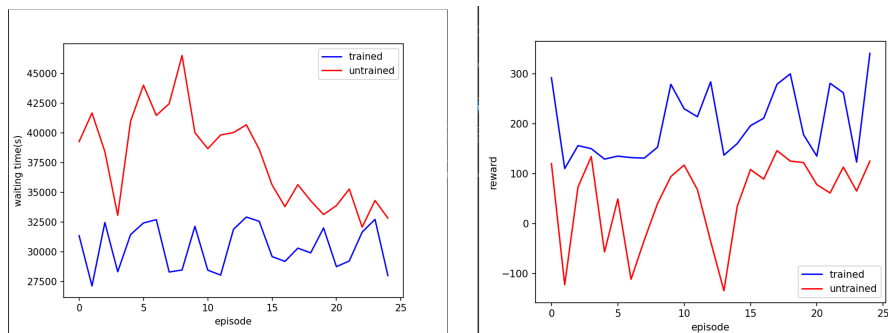
Đối với Deep Q-learning [3], thay vì sử dụng bảng Q-table, thuật toán sử dụng một mạng Neural Network.



Hình 3. Q-learning vs Deep Q-learning.

### 3 Tổng kết

Điều khiển đèn tín hiệu giao thông là một vấn đề phức tạp cần được nghiên cứu và đưa ra các giải pháp điều khiển hiệu quả hơn. Trong bài báo này nhóm đã đề xuất một phương pháp cải tiến để tối ưu lưu lượng giao thông tại các nút giao cắt giao thông. Phương pháp này sử dụng thuật toán học tăng cường để mô phỏng và tối ưu lưu lượng phương tiện tại một nút giao thông cụ thể. Kết quả giữa không sử dụng và sử dụng phương pháp như Hình 4.



Hình 4. Kết quả.

Trong tương lai, nhóm sẽ tiếp tục phân tích và có thể tìm ra các phương pháp đổi mới và hiện đại. Đồng thời thực nghiệm với loại hình nút giao cắt khác nhau, các mô hình khác nhau nhằm tối ưu hóa lưu lượng giao thông tốt hơn.

## Lời cảm ơn

Em muốn gửi lời cảm ơn đến các Thầy Cô Giảng Viên Trường Đại Học Công Nghệ Thông Tin và đặc biệt là thầy Đỗ Trọng Hợp, người đã tận tình hướng dẫn, chỉ bảo chúng em trong thời gian qua, đồng thời cũng là người đã giúp nhóm hoàn thiện bài báo cáo này.

Em xin chân thành cảm ơn!

## Tài liệu

1. SUMO User Documentation, 2021, <https://sumo.dlr.de/docs/index.html>
2. Q-learning, 2019, <https://en.wikipedia.org/wiki/Q-learning>
3. Deep Q-learning, 2020, <https://wiki.pathmind.com/deep-reinforcement-learning>