TRƯỜNG ĐẠI HỌC CÔNG NGHỆ, ĐẠI HỌC QUỐC GIA HÀ NỘI

KHOA ĐIỆN TỬ-VIỄN THÔNG



BÁO CÁO MÔN HỌC

**CÁC THUẬT TOÁN THÍCH NGHI**

Đề tài:

**ỨNG DỤNG HỌC SÂU TĂNG CƯỜNG VÀO BÀI TOÁN ĐIỀU KHIỂN TAY MÁY**

Nhóm thực hiện: Nhóm 7

Nguyễn Thế Hưởng. MSSV: 18020624

Trần Đức Sơn. MSSV: 18021099

Vũ Duy Hưng. MSSV: 18020617

Giảng viên hướng dẫn: Nguyễn Thế Hoàng Anh

Hà Nội, 12-2021

**MỤC LỤC**

[**I.** **Giới thiệu** 3](#_Toc91449740)

[**II.** **Nền tảng lý thuyết** 3](#_Toc91449741)

[**2.1** **Học tăng cường, học sâu tăng cường** 3](#_Toc91449742)

[**a)** **Deep Q Network** 5](#_Toc91449743)

[**b)** **Double Deep Q network** 5](#_Toc91449744)

[**III.** **Triển khai chương trình** 5](#_Toc91449745)

[**3.1** **Xây dựng mô hình học tăng cường** 5](#_Toc91449746)

[**3.2** **Xây dựng Double Deep Q network** 7](#_Toc91449747)

[**a)** **Giải thuật** 7](#_Toc91449748)

[**b)** **Tham số** 7](#_Toc91449749)

[**3.3** **Xây dụng các hàm đánh giá** 7](#_Toc91449750)

[**IV.** **Kết quả đánh giá** 8](#_Toc91449751)

[**V.** **Kết luận** 10](#_Toc91449752)

[**VI.** **Tài liệu tham khảo** 10](#_Toc91449753)

**DANH MỤC HÌNH ẢNH**

[Hình 1: Sơ đồ học tăng cường 3](#_Toc91450702)

[Hình 2: Mô hình tổng quan về Deep reinforcement 4](#_Toc91450703)

[Hình 3: Q learning 4](#_Toc91450704)

[Hình 4: Deep Q network 5](#_Toc91450705)

[Hình 5: Double deep Q network 5](#_Toc91450706)

[Hình 6: Các hành động robot 6](#_Toc91450707)

[Hình 7: Sơ đồ hệ thống 6](#_Toc91450708)

[Hình 8: Điểm thưởng trên episode 8](#_Toc91450709)

[Hình 9: Giá trị Q\_eval trên mỗi episode 9](#_Toc91450710)

[Hình 10: Số bước chạy trên mỗi episode 9](#_Toc91450711)

[Hình 11: So sánh DQN và DDQN 10](#_Toc91450712)

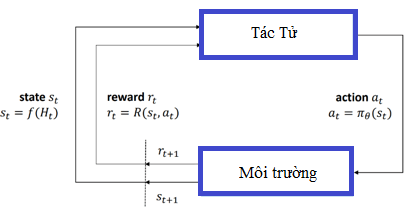
1. **Giới thiệu**

Tay máy đóng một vai trò quan trọng trong lĩnh vực tự động hóa công nghiệp. Tuy nhiên đa phần tay máy hiện nay chỉ thực hiện các quá trình lặp đi lặp lại các lệnh cố định dựa trên các phương pháp điều khiển truyền thống, điều này phù hợp với dây chuyền sản xuất với môi trường rõ ràng nhưng rất khó để điều khiển tay máy thích ứng được với thời đại thông minh chỉ dựa trên các trình điều khiển được viết sẵn. Với những tác vụ thông minh, môi trường hoạt động của tay máy thay đổi yêu cầu tay máy cũng phải linh hoạt hơn. Một thử thách được đặt ra cho việc điều khiển tay máy là làm sao điều khiển được quỹ đạo di chuyển của khâu công tác (thành phần của tay máy tiếp xúc trực tiếp với đối tượng) sao cho thực hiện được nhiệm vụ trong điều kiện hướng và vị trí đối tượng thay đổi.

Học tăng cường được lấy cảm hứng từ tự nhiên. Việc áp dụng học tăng cường vào điều khiển tay máy không cần các lệnh cố định mà học bằng cách tương tác với môi trường. Trong quá trình này tay máy sẽ nhận được phản hồi có thể đánh giá được hoạt động của tay máy và giúp tay máy có thể tìm được chiến lược tối ưu để hoàn thành nhiệm vụ.

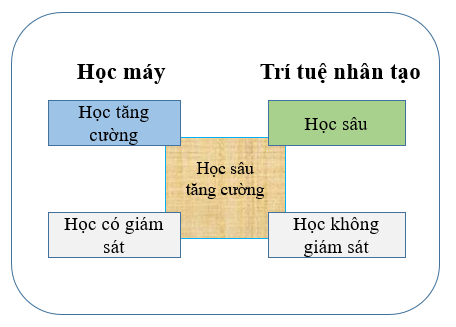
1. **Nền tảng lý thuyết**
   1. **Học tăng cường, học sâu tăng cường**

Học tăng cường [1] là một nhánh con của học máy, nó lấy cảm hứng từ suy nghĩ tự nhiên, khi con người liên tục lấy các thông tin từ môi trường và thực hiện các hành động để thích ứng với môi trường. Học tăng cường dựa trên quy trình quyết định Markov (Markov Decision Process), thường có năm giá trị là (S,A,R,P,) , trong đó S (state) là trạng thái, A (action) là hành động có thể lựa chọn, R (reward) là giá trị phần thưởng sau khi thực hiện hành động, P đại diện cho xác xuất chuyển trạng thái, là tham số tính toán phần thưởng và chiến lược biểu diễn ánh xạ từ trạng thái sang hành động.



Hình 1: Sơ đồ học tăng cường

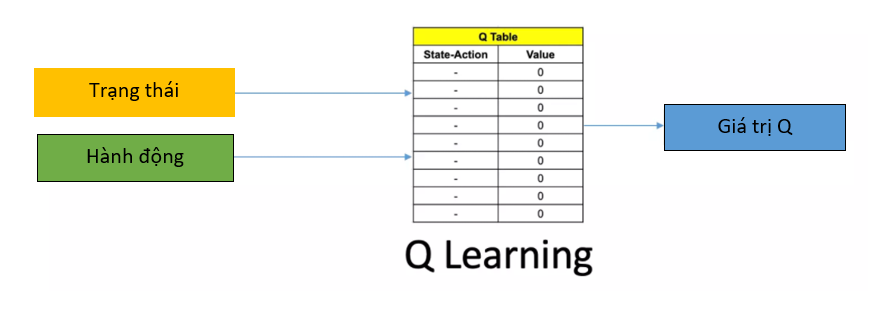
Học sâu tăng cường là kết hợp của học sâu và học tăng cường và có thể được coi là cầu nối giữa học máy và trí tuệ nhân tạo.



Hình 2: Mô hình tổng quan về Deep reinforcement

Q learning là một thuật toán học tăng cường off policy. **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.

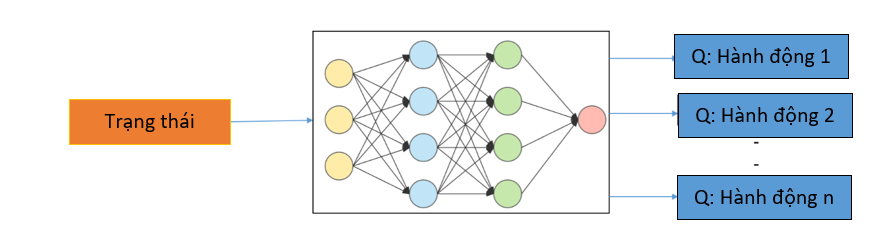
Để thực hiện thuật toán q learning, việc đầu tiên ta cần làm là tạo một bảng q (q - table). Q table là một ma trận có chiều là [trạng thái, hành động] ([state, action]). Chúng ta khởi tạo giá trị 0 cho ma trận này. Sau đó, chúng ta cập nhật và lưu trữ các **giá trị q** vào q-table sau từng **tập (episode** - 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).



Hình 3: Q learning

1. **Deep Q Network**

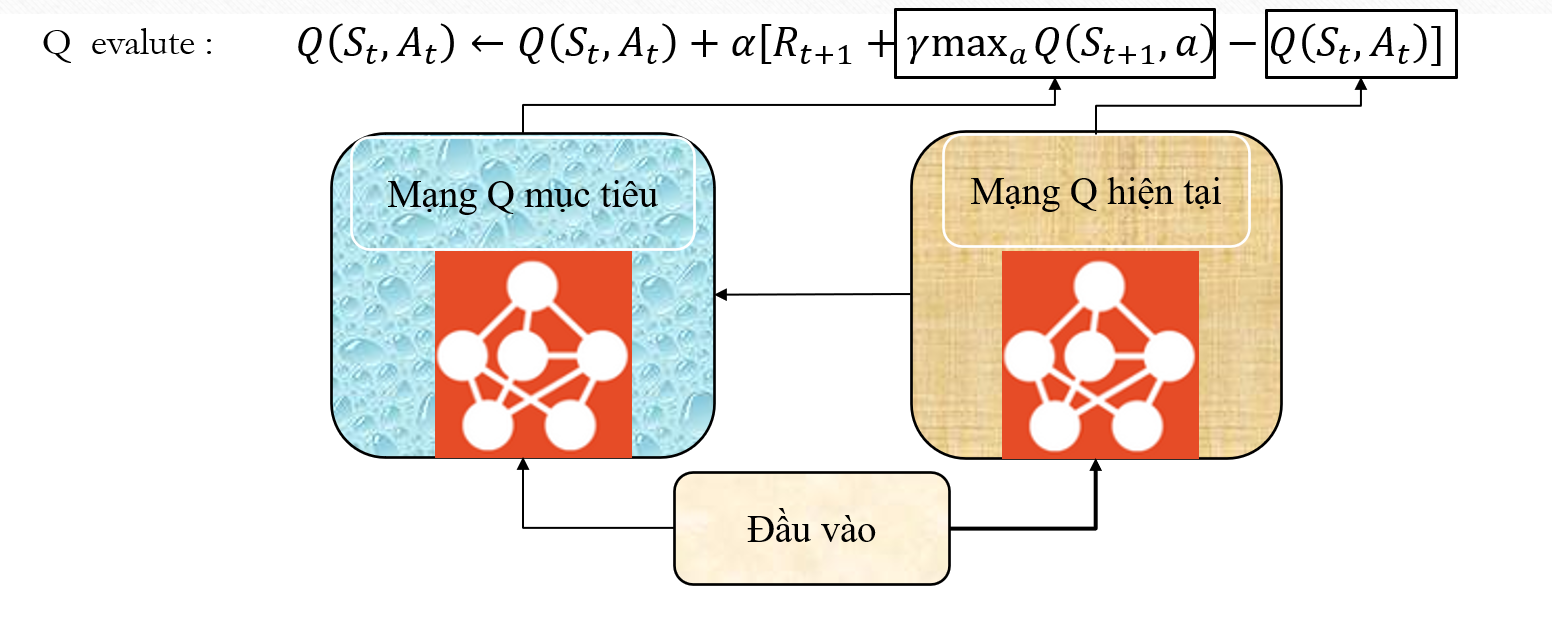
Deep Q Network là sự kết hợp của Q learning và deep learning. Sử dụng một mạng học sâu thay cho Q table.



Hình 4: Deep Q network

1. **Double Deep Q network**

Double deep q network [2]: Sử dụng 2 mạng DNN để tính các giá trị Q hiện tại và Q mục tiêu.

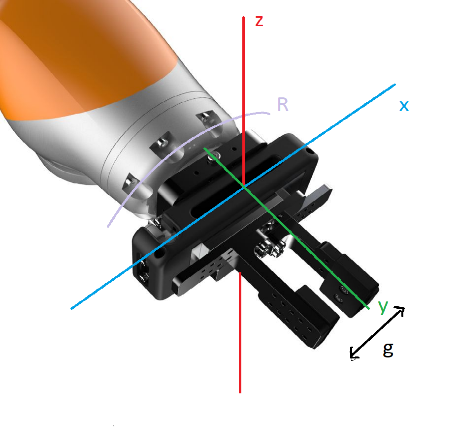


Hình 5: Double deep Q network

1. **Triển khai chương trình**
   1. **Xây dựng mô hình học tăng cường**

Từ yêu cầu thực tiễn và nền tảng lý thuyết, nhóm đã xây dựng chương trình mô phỏng điều khiển chuyển động của tay máy như sau:

* Mục tiêu: Điều khiển khâu công tác của tay máy di chuyển trong vùng làm việc tới đích và thực hiện hành động gắp đối tượng (viên phấn).
* Phần mềm mô phỏng: Robot Kuka trong thư viện Pybullet, thư viện OpenAi Gym kết nối giữa môi trường của robot Kuka.
* Môi trường hoạt động: Robot và đối tượng nằm ngẫu nhiên trong vùng hoạt động của robot giới hạn trong vùng (0.5x0.5x0.5).
* Trạng thái: Vị trí và hướng của khâu công tác
* Hành động: Gồm 8 hành động: Di chuyển theo trục x (2), di chuyển theo trục y (2), di chuyển theo trục z (2), xoay hướng khâu công tác, và kẹp. Hành động trong một tập tối đa 500 bước, mỗi bước tương ứng 0.01 đơn vị độ dài và 2 độ xoay.



Hình 6: Các hành động robot

Hàm điểm thưởng: Dựa vào khoảng cách Euclid giữa khâu công tác với đối tượng và kết quả gắp vật:

d = . (1)

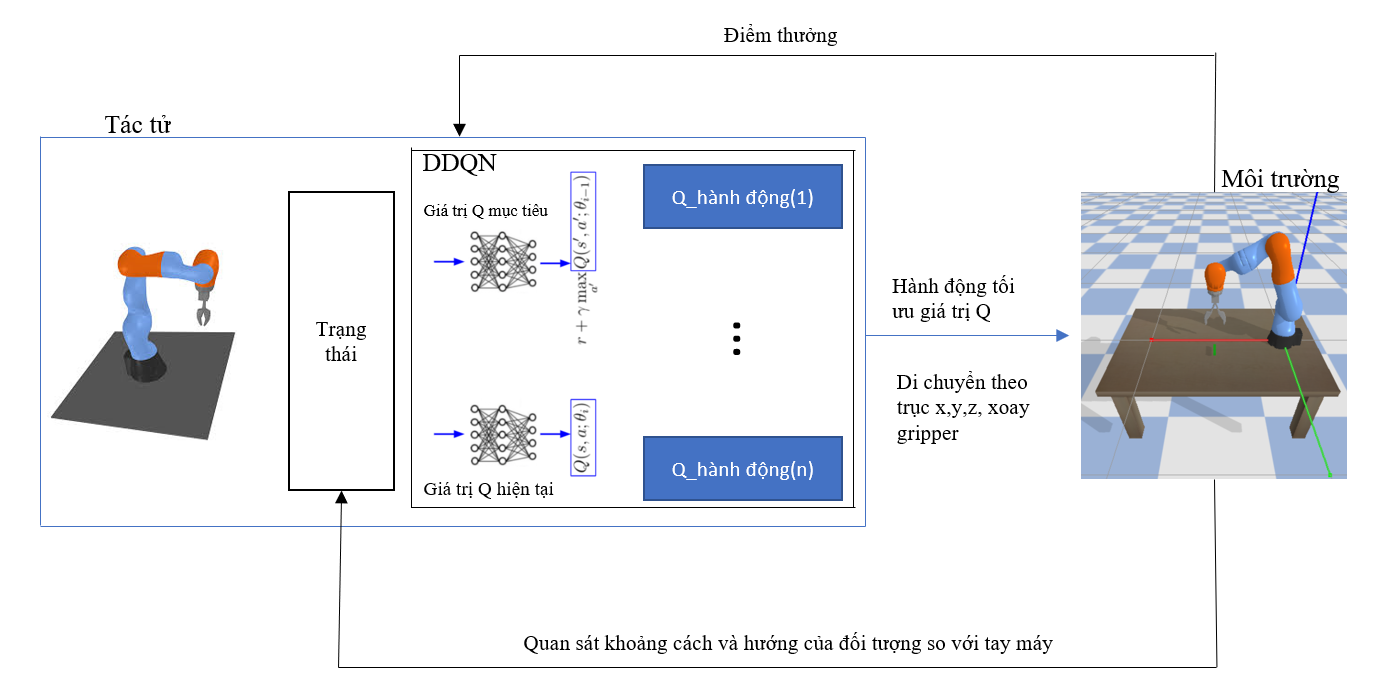
Với ( , ) là tọa độ của vật gắp.

( , ) là tọa độ tay kẹp của robot.

Nếu d > 0.05: R = -2\*d. (2)

d < 0.05: R = 3 - 2\*d. (3)

Robot kẹp được vật: R = R + 1. (4)



Hình 7: Sơ đồ hệ thống

* 1. **Xây dựng Double Deep Q network**

1. **Giải thuật**

Khởi tạo mạng chính , mạng mục tiêu , bộ đệm ,   
Với mỗi vòng lặp:  
Với mỗi bước hành động trong môi trường:  
Quan sát trạng thái và lựa chọn hành động   
Thực hiện và quan sát trạng thái tiếp theo và điểm thưởng   
Lưu trữ vào bộ nhớ đệm   
Mới mỗi bước cập nhật:  
Tính toán giá trị

(5)

Thực hiện bước hiệu chỉnh dựa trên giá trị (6)  
Cập nhật thông số mạng mục tiêu:

1. **Tham số**

Một số tham số đầu vào của mạng được điều chỉnh để tăng khả năng hội tụ của bài toán. Nhóm có tham khảo bài viết [3] và thực nghiệm thu được:

*Bảng 1: Giá trị và ý nghĩa các tham số*

|  |  |  |
| --- | --- | --- |
| Tham số | Giá tri | Ý nghĩa |
| Learning\_rate (tỉ lệ học) |  | Mức độ chấp nhận giá trị mới so với giá trị cũ. |
| (Hệ số suy giảm) |  | Mức độ quan trọng của phần thưởng tiếp theo. |
| Epsilon\_greedy |  | Tỉ lệ chọn giá trị hành động cao nhất. |

Các giá trị này ảnh hưởng tới việc xác định giá trị Q của hành động và lựa chọn hành động tốt nhất theo:

(7)

Hành động thực hiện = Hành động(maxQ) với xác suất .

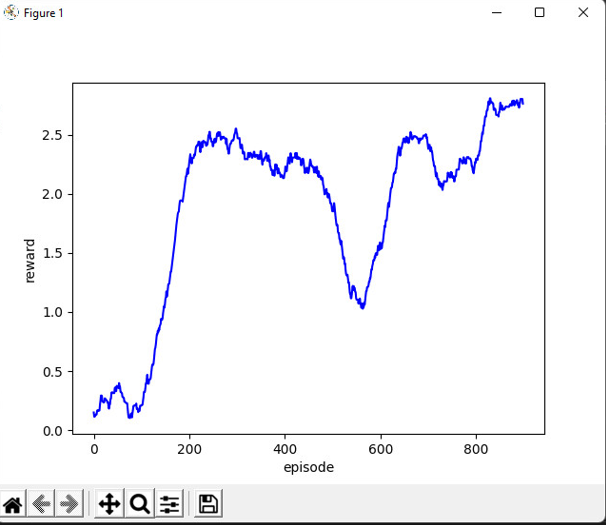
* 1. **Xây dụng các hàm đánh giá**

Nhóm đã xây dựng một số hàm nhằm đánh giá kết quả mô phỏng dựa trên các tiêu chí:

* Đánh giá điểm thưởng trên các episode.
* Đánh giá giá trị Q\_eval trên các episode.
* Đánh giá số bước chạy trong các episode.
* So sánh kết quả sử dụng DQN và DDQN.

1. **Kết quả đánh giá**

* Đánh giá điểm thưởng trên các episode.



Hình 8: Điểm thưởng trên episode

Ban đầu các giá trị thấp do robot di chuyển ngẫu nhiên và tăng dần đến 200 eps di chuyển đến vật bắt đầu ổn định.

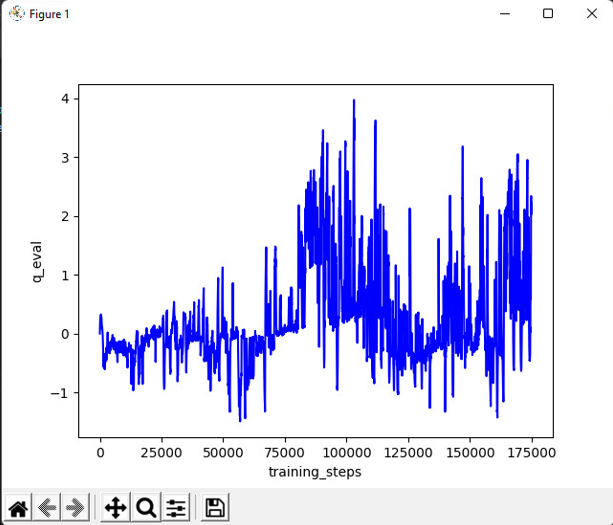
Từ 200 – 500 eps robot nhiệm vụ di chuyển đến vật đạt kết quả ổn định.

Từ 500 - 600 eps sau bắt đầu thử nhiệm vụ gắp vật và bị lỗi dẫn đến kết quả bị tụt điểm thưởng do vật bị văng ra xa.

Từ 600 -800 robot bắt đầu lại quá trình.

Từ 800 eps robot bắt đầu thực hiện gắp được nhiều hơn và điểm tăng vượt mốc 2.5 và dần hội tụ về 3.

* Đánh giá giá trị Q\_eval trên các episode.

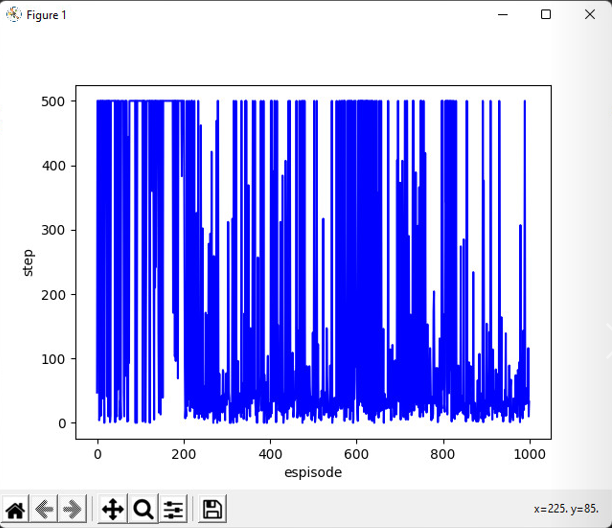


Hình 9: Giá trị Q\_eval trên mỗi episode

Càng hội tụ thì Q\_eval càng lớn.

Giá trị Q\_eval thay đổi tương ứng với giá trị phần thưởng thu được.

* Đánh giá số bước chạy trong các episode.



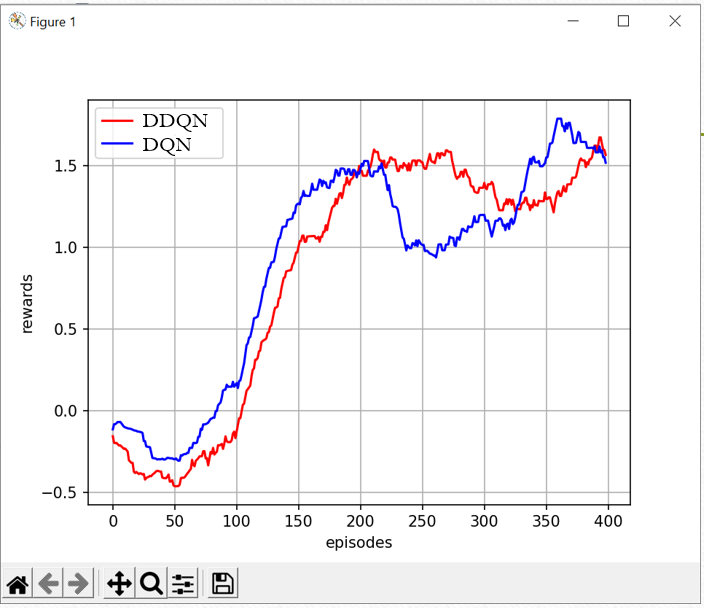
Hình 10: Số bước chạy trên mỗi episode

Khi chưa ổn định số bước thường tối đa (500 bước) sau đó giảm dần.

Khi gắp vật mất ổn định và số bước cũng tăng lên

Đến khi ổn định hơn từ 800 – 1000 eps số bước giảm trong khoảng 100 bước. Robot cần thực hiện ít hành động hơn để có thể thực hiện được mục tiêu.

* So sánh kết quả sử dụng DQN và DDQN.

****

Hình 11: So sánh DQN và DDQN

DQN cho tốc độ tăng điểm thưởng tốt hơn nhưng cũng biến động lớn hơn.

DDQN tốc độ tăng điểm chậm hơn nhưng ổn định hơn là do nó dùng thêm một mạng học sâu để đánh giá các giá trị Q mục tiêu nên hành động thay đổi chậm hơn nhưng tin cậy hơn.

1. **Kết luận**

Việc đưa RL vào việc training tay máy là có khả thi, tuy nhiên cần thời gian dài để training và tối ưu. Chưa xác định cần tối thiểu bao nhiêu episode để robot tay máy hoạt động chính xác 100%, hệ thống vẫn ẩn chứa rủi ro. Việc áp dụng vào thực tế còn ảnh hưởng bởi nhiều tác động bất ngờ khác mà môi trường mô phỏng chưa thể tái tạo được có thể ảnh hưởng đến quá trình hoạt động của robot ví dụ như điện yếu, ngoại lực … Tuy nhiên, mô hình trên là cơ sở để nhóm tiếp tục nghiên cứu và phát triển thêm trong tương lai.

1. **Tài liệu tham khảo**

[1] https://en.wikipedia.org/wiki/Reinforcement\_learning

[2]  [H. van Hasselt. , A. Guez, D. Silver. Deep Reinforcement Learning with Double Q-Learning, AAAI. 2016.](https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/download/12389/11847)

[3] Nathan Sprague. *Parameter Selection for the Deep Q-Learning Algorithm*