**Laporan Praktikum Kontrol Cerdas**

**Minggu ke-4**

|  |  |
| --- | --- |
| Nama  NIM  Kelas  Akun Github (Tautan)  Student Lab Assistant | : Kresensia Meita Indar Mayaningsih  : 224308087  : TKA 7D  : <https://github.com/kzmeita>  : |

1. **Judul Percobaan :**

Reinforcement Learning for Autonomous Control

1. **Tujuan Percobaan**

Tujuan dari praktikum minggu ke-4 :

1. Mahasiswa dapat memahami konsep dasar Reinforcement Learning (RL) dalam sistem kendali.
2. Mahasiswa dapat mengimplementasikan agen RL menggunakan algoritma Deep Q-Network (DQN).
3. Mahasiswa dapat menggunakan OpenAI Gym sebagai simulasi lingkungan untuk pelatihan RL.
4. Mahasiswa dapat melatih dan menguji agen RL untuk mengontrol lingkungan secara otonom.
5. Mahasiswa dapat menggunakan GitHub untuk version control dan dokumentasi praktikum.
6. **Landasan Teori**
7. **Analisis dan Diskusi**
8. **Analisis**
9. **Diskusi**
10. **Assignment**

Pada praktikum kali ini kami mempelajari penerapan **Deep Q-Network (DQN)** sebagai salah satu metode pada **Reinforcement Learning (RL)**. DQN merupakan pengembangan dari algoritma Q-Learning yang menggabungkannya dengan **Deep Neural Network** agar mampu menangani permasalahan state yang besar maupun kontinu. Pada Q-Learning klasik, nilai dari setiap aksi di setiap state disimpan dalam sebuah tabel yang disebut **Q-table**. Namun, pendekatan ini memiliki keterbatasan karena ketika jumlah state sangat besar (misalnya data sensor yang kompleks atau gambar dari kamera), ukuran Q-table akan menjadi sangat besar dan tidak efisien. Untuk mengatasi hal ini, DQN menggunakan jaringan saraf dalam memperkirakan fungsi Q, sehingga pembelajaran dapat berjalan lebih efektif. Dalam praktikum ini, kami melakukan beberapa modifikasi terhadap algoritma DQN, salah satunya dengan menambahkan **target network** agar proses pelatihan lebih stabil. Target network bekerja dengan cara menyalin parameter dari model utama secara periodik, sehingga update bobot tidak terlalu cepat berubah dan mengurangi osilasi saat proses training. Selain itu, kami juga menguji agen pada beberapa environment berbeda seperti **LunarLander-v2** dan **MountainCar-v0**, lalu membandingkan kinerja agen pada masing-masing environment tersebut. Untuk memperdalam pemahaman, dilakukan pula eksperimen dengan mengubah beberapa parameter penting, seperti **learning rate**, **discount factor (γ)**, dan **epsilon decay**, sehingga dapat diamati pengaruhnya terhadap kinerja agen baik dari segi kecepatan konvergensi maupun nilai reward yang diperoleh. Semua hasil eksperimen kemudian divisualisasikan dalam bentuk grafik reward, jumlah langkah, serta perubahan nilai epsilon pada setiap episode. Terakhir, seluruh kode program, hasil eksperimen, serta laporan singkat dipublikasikan ke dalam repository GitHub sebagai dokumentasi. Dengan praktikum ini, kami memahami bahwa penerapan DQN memberikan solusi terhadap keterbatasan Q-Learning tradisional, meskipun tetap memerlukan teknik tambahan seperti target network dan tuning parameter agar agen dapat belajar secara optimal pada environment yang berbeda.

1. **Data dan Output Hasil Pengamatan**

Data yang diperoleh pada praktikum minggu ke-1 disajikan dalam tabel dibawah ini:

|  |  |  |
| --- | --- | --- |
| No | Variabel | Hasil Pengamatan |
| 1. | Hasil program saat melakukan training dengan 200 episode |  |
| 2. | Hasil program sebelum dimodifikasi dengan 200 episode (test\_agent.py) |  |
| 3. |  |  |
| 4. |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |

1. **Kesimpulan**
2. **Saran**
3. **Daftar Pustaka**