

Week 4

Laporan Praktikum Kontrol Cerdas

Nama : Vivi Aulia Husna Wahidah
NIM : 224308023
Kelas : TKA-6A
Akun Github (Tautan) : <https://github.com/vivi-aulia>
Student Lab Assistant : Muhammad Mahirul Faiq

1. Judul Percobaan

Week 4: Reinforcement Learning for Autonomous Control

2. Tujuan Percobaan

Tujuan dari praktikum “Reinforcement Learning for Autonomous Control”, mahasiswa diharapkan mampu:

- Memahami konsep dasar Reinforcement Learning (RL) dalam sistem kendali.
- Mengimplementasikan agen RL menggunakan algoritma Deep Q-Network (DQN).
- Menggunakan OpenAI Gym sebagai simulasi lingkungan untuk pelatihan RL.
- Melatih dan menguji agen RL untuk mengontrol lingkungan secara otonom.
- Menggunakan GitHub untuk version control dan dokumentasi praktikum.

3. Landasan Teori

1. Reinforcement Learning (RL)

1.1 Definisi RL

Reinforcement Learning (RL) adalah cabang dari pembelajaran mesin yang berfokus pada bagaimana agen dapat mengambil tindakan dalam suatu lingkungan untuk memaksimalkan suatu fungsi penghargaan kumulatif. Dalam RL, agen belajar melalui interaksi dengan lingkungan dan menerima umpan balik dalam bentuk reward atau punishment untuk meningkatkan performanya seiring waktu.

1.2 Prinsip Dasar RL

Prinsip utama dalam RL melibatkan:

- Agan:** Entitas yang mengambil tindakan dalam lingkungan.
- Lingkungan:** Dunia tempat agen beroperasi dan mendapatkan umpan balik.
- State (S):** Keadaan sistem pada suatu waktu tertentu.
- Action (A):** Serangkaian tindakan yang dapat dilakukan oleh agen.
- Reward (R):** Umpan balik dari lingkungan sebagai konsekuensi dari tindakan yang dilakukan oleh agen.
- Policy (π):** Strategi yang digunakan agen untuk memilih tindakan berdasarkan state.
- Value Function (V) dan Q-Function (Q):** Estimasi nilai suatu state atau pasangan state-action dalam jangka panjang.

1.3 Hukum yang Mendasari RL

- Hukum Bellman:** Menggunakan relasi rekursif untuk mendefinisikan nilai dari suatu state berdasarkan nilai dari state berikutnya.
- Markov Decision Process (MDP):** Model matematis yang mendasari sebagian besar pendekatan RL, yang mencerminkan sifat probabilistik transisi state dalam suatu lingkungan.

2. Deep Q-Network (DQN)

2.1 Definisi DQN

Deep Q-Network (DQN) adalah algoritma RL yang menggunakan jaringan saraf tiruan (deep learning) untuk mengaproksimasi fungsi nilai tindakan (Q-value). Algoritma ini dikembangkan oleh Google DeepMind dan digunakan untuk mengatasi permasalahan RL dengan ruang state yang besar.

2.2 Prinsip Dasar DQN

DQN bekerja berdasarkan prinsip Q-learning, dengan perbaikan sebagai berikut:

- **Experience Replay:** Penyimpanan pengalaman dalam buffer untuk mengurangi korelasi antar sampel.
- **Target Network:** Penggunaan jaringan target yang diperbarui secara berkala untuk meningkatkan stabilitas pembelajaran.
- **Loss Function:** Perhitungan selisih antara nilai target dan nilai prediksi yang dioptimalkan menggunakan algoritma pembelajaran deep learning.

3. OpenAI Gym sebagai Simulasi Lingkungan

3.1 Definisi OpenAI Gym

OpenAI Gym adalah toolkit yang menyediakan berbagai lingkungan simulasi untuk pengembangan dan pengujian algoritma RL. Toolkit ini memungkinkan pengembang untuk membandingkan berbagai metode RL dalam berbagai lingkungan, termasuk permainan, robotika, dan optimasi.

3.2 Prinsip Dasar OpenAI Gym

- **Step Function:** Fungsi yang digunakan untuk memperbarui state berdasarkan tindakan agen.
- **Reward Function:** Menentukan nilai penghargaan yang diberikan kepada agen setelah melakukan suatu tindakan.
- **Reset Function:** Mengembalikan lingkungan ke kondisi awal setelah episode selesai.

4. Pelatihan dan Pengujian Agen RL

4.1 Proses Pelatihan Agen RL

Proses pelatihan agen RL melibatkan:

1. **Inisialisasi Parameter:** Menentukan arsitektur jaringan dan nilai awal.
2. **Interaksi dengan Lingkungan:** Agen mengambil tindakan dan menerima umpan balik.
3. **Pembaruan Model:** Menggunakan algoritma pembelajaran seperti gradient descent untuk memperbaiki model.
4. **Evaluasi Performa:** Mengukur efektivitas agen berdasarkan reward kumulatif yang diperoleh.

4.2 Pengujian Agen RL

Setelah pelatihan selesai, agen diuji untuk memastikan kemampuannya dalam mengontrol lingkungan secara otonom. Proses pengujian ini dilakukan dengan menerapkan policy yang telah dipelajari pada lingkungan tanpa pembaruan parameter.

4. Analisis dan Diskusi

Analisis Hasil:

1. Performa Agen di CartPole:

Agen DQN bertugas menjaga keseimbangan tiang di atas kereta. Jika pelatihan optimal, agen dapat bertahan lama. Jika tidak, performanya kurang baik.

2. Pengaruh Parameter terhadap Kinerja Agen:

- **Gamma:** Menentukan fokus agen pada reward jangka pendek atau panjang.
- **Epsilon:** Mengontrol eksplorasi dan eksploitasi tindakan.
- **Learning Rate:** Memengaruhi kecepatan dan stabilitas pembelajaran.

3. Tantangan Pelatihan RL:

- Menyeimbangkan eksplorasi dan eksploitasi.
- Menghindari ketidakstabilan pelatihan.
- Menyesuaikan parameter dengan optimal.

Diskusi:

1. Perbedaan RL dan Supervised Learning dalam Kendali:
- RL: Belajar dari pengalaman dengan mekanisme reward.

Supervised Learning: Menggunakan dataset berlabel untuk belajar pola.

RL lebih cocok untuk lingkungan dinamis dan kompleks.
2. Optimasi Eksplorasi dan Eksploitasi RL:
- Epsilon-greedy decay: Eksplorasi tinggi di awal, menurun seiring waktu.

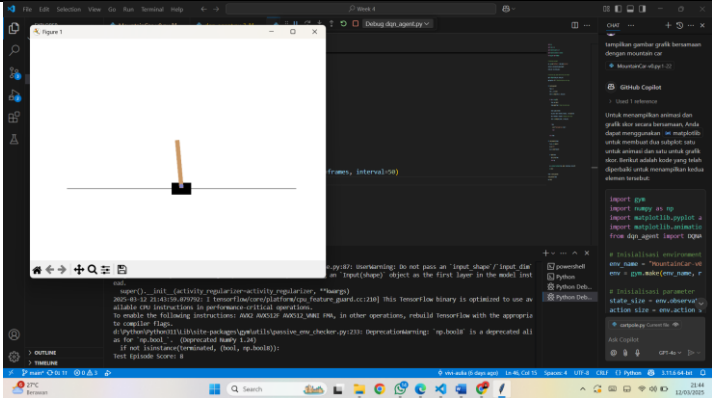
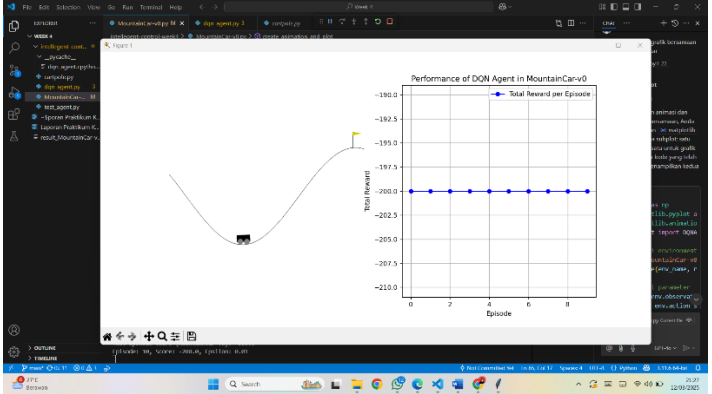
UCB & Boltzmann Exploration: Menyeimbangkan eksplorasi dan eksploitasi berdasarkan reward.
3. Aplikasi RL dalam Sistem Kendali Nyata:
- Robotika: Mengoptimalkan pergerakan.

Kendaraan Otonom: Mengambil keputusan adaptif.

Industri & Drone: Meningkatkan efisiensi dan navigasi otomatis.

5. Data dan Output Hasil Pengamatan

Data dan hasil yang diperoleh selama percobaan.

No	Variabel	Hasil Pengamatan
1	Cartpole	
2	Mountaincar-v0	

6. Kesimpulan

Berdasarkan hasil penelitian dan analisis yang dilakukan, dapat disimpulkan bahwa algoritma Deep Q-Network (DQN) mampu mengoptimalkan performa agen dalam mengontrol lingkungan secara otonom. Keberhasilan agen dalam menjalankan tugasnya sangat bergantung pada konfigurasi parameter pelatihan, seperti gamma, epsilon, dan learning rate. Tantangan utama yang dihadapi dalam implementasi RL adalah keseimbangan antara eksplorasi dan eksploitasi serta stabilitas pelatihan yang harus dijaga agar agen dapat mempelajari kebijakan yang optimal. Selain itu, perbandingan antara RL dan supervised learning menunjukkan bahwa RL lebih efektif dalam menangani sistem yang dinamis dan kompleks karena kemampuannya untuk belajar langsung dari interaksi dengan lingkungan tanpa bergantung pada dataset berlabel. Dengan berbagai keunggulan ini, RL berpotensi besar dalam penerapannya pada berbagai sistem kendali nyata, seperti robotika, kendaraan otonom, dan industri.

7. Saran

- Melakukan eksplorasi lebih lanjut terhadap konfigurasi parameter untuk menemukan kombinasi yang paling optimal dalam pelatihan agen RL.
- Menggunakan metode eksplorasi yang lebih adaptif, seperti Bayesian Optimization atau Meta-Learning, guna meningkatkan efisiensi pembelajaran agen.
- Menguji agen RL dalam lingkungan simulasi yang lebih kompleks untuk mengevaluasi kemampuan generalisasi terhadap berbagai skenario kendali.
- Mengimplementasikan RL pada sistem kendali nyata dengan validasi tambahan melalui pengujian pada perangkat keras dan lingkungan fisik untuk memastikan efektivitas dan keandalan.

8. Daftar Pustaka

- Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction (2nd ed.)*. MIT Press.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Hassabis, D. (2015). *Human-level control through deep reinforcement learning*. *Nature*, 518(7540), 529-533.
- Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., ... & Wierstra, D. (2016). *Continuous control with deep reinforcement learning*. arXiv preprint arXiv:1509.02971.
- OpenAI. (2016). *OpenAI Gym*. arXiv preprint arXiv:1606.01540.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.