

# Laporan Praktikum Kontrol Cerdas

Nama : Nefi Afif Sujatjana

NIM : 224308093

Kelas : TKA 7D

Akun GitHub (Tautan) : <https://github.com/nefiafif612-cmd>

## 1. Judul Percobaan :

Deep Reinforcement Learning untuk Kontrol Kompleks

## 2. Tujuan Percobaan

- Mahasiswa diharapkan dapat memahami konsep Deep Reinforcement Learning (DRL) dalam kontrol sistem kompleks.
- Mahasiswa diharapkan dapat mengimplementasikan Deep Q-Network (DQN) untuk kontrol otomatis.
- Mahasiswa diharapkan dapat menganalisis performa DRL dibandingkan dengan metode kontrol konvensional.

## 3. Landasan Teori

Reinforcement Learning (RL) adalah metode pembelajaran mesin di mana agen belajar mengambil tindakan optimal melalui interaksi dengan lingkungan untuk memaksimalkan reward. Proses ini biasanya dimodelkan dengan Markov Decision Process (MDP), yang mencakup state, aksi, transisi, reward, dan faktor diskonto. Tujuan utama agen adalah menemukan policy terbaik yang memetakan state ke aksi demi meraih reward jangka panjang maksimal. Formulasi matematis MDP beserta strategi optimasi kebijakan dijelaskan secara komprehensif oleh Sutton dan Barto (2018).

Q-Learning adalah algoritma RL populer yang memperkirakan nilai aksi  $Q(s, a)$ , yaitu ekspektasi reward saat mengambil aksi tertentu di suatu state. Namun, versi tabularnya terbatas pada ruang state kecil, karena tidak dapat menggeneralisasi state baru dalam ruang yang besar atau kontinu. Untuk mengatasi permasalahan tersebut, dikembangkanlah Deep Q-Network (DQN) yang menggunakan jaringan saraf dalam sebagai fungsi aproksimasi nilai aksi (Mnih et al., 2015).

DQN memperkenalkan beberapa teknik untuk meningkatkan stabilitas dan efektivitas pelatihan. Pertama, experience replay menyimpan pengalaman agen dalam buffer dan mengambilnya secara acak saat belajar, guna mengurangi korelasi antar data dan menstabilkan pembaruan. Kedua, target network digunakan sebagai salinan jaringan utama

untuk menjaga kestabilan target selama pelatihan. Ketiga, strategi  $\epsilon$ -greedy memungkinkan agen memilih aksi secara acak dengan probabilitas  $\epsilon$  yang berkurang seiring waktu, sehingga tercapai keseimbangan antara eksplorasi dan eksploitasi. Pendekatan ini terbukti berhasil pada berbagai tugas pengendalian kompleks, termasuk permainan Atari dengan performa setara manusia (Mnih et al., 2015).

DQN dirancang untuk aksi diskrit, sehingga kurang cocok untuk kasus dengan aksi kontinu seperti kontrol motor atau optimasi energi. Untuk itu, digunakan metode actor-critic seperti DDPG, TD3, atau SAC yang lebih sesuai untuk ruang aksi kontinu. Hal ini sejalan dengan hasil kajian yang menunjukkan bahwa pendekatan actor-critic lebih sesuai untuk ruang aksi kontinu karena mampu menghasilkan distribusi aksi yang lebih halus (Li, 2017; François-Lavet et al., 2018).

Platform simulasi seperti OpenAI Gym atau Gymnasium menjadi sarana utama dalam pengembangan dan pengujian algoritma RL. Lingkungan simulasi standar seperti CartPole, MountainCar, Pendulum, LunarLander, hingga Reacher, banyak digunakan sebagai benchmark untuk mengevaluasi performa agen RL dengan variasi parameter dan metode. Keberadaan simulasi ini memungkinkan peneliti maupun praktisi untuk menguji strategi kontrol cerdas sebelum diimplementasikan pada sistem nyata (Brockman et al., 2016).

Berbeda dari kontrol klasik seperti PID atau Fuzzy Logic yang sederhana dan stabil pada sistem dengan model jelas, RL seperti DQN lebih fleksibel karena belajar langsung dari interaksi tanpa memerlukan model. Namun, RL cenderung lebih mahal secara komputasi dan rentan terhadap instabilitas. Oleh karena itu, pemilihan metode kontrol harus mempertimbangkan konteks aplikasi, ketersediaan model sistem, serta kebutuhan adaptasi terhadap ketidakpastian (François-Lavet et al., 2018).

#### **4. Analisis dan Diskusi**

Berdasarkan hasil implementasi algoritma Deep Q-Network (DQN) pada environment Pendulum-v1, dapat diamati bahwa performa agen sangat dipengaruhi oleh jumlah episode pelatihan dan parameter yang digunakan. Pada awal pelatihan, agen sering kali menghasilkan reward yang rendah karena masih banyak melakukan eksplorasi acak. Namun, seiring bertambahnya episode, reward cenderung meningkat dan stabil ketika nilai epsilon menurun, sehingga agen lebih banyak mengeksplorasi aksi yang dianggap optimal. Hal ini sesuai dengan prinsip dasar *exploration-exploitation trade-off* pada reinforcement learning.

Penggunaan parameter seperti learning rate, gamma, dan epsilon decay terbukti memberikan pengaruh signifikan terhadap kinerja agen. Misalnya, nilai gamma yang tinggi membuat agen lebih fokus pada reward jangka panjang, sedangkan gamma

rendah mendorong agen untuk mengejar reward instan. Demikian juga, nilai learning rate yang terlalu besar dapat menyebabkan pembaruan bobot jaringan saraf menjadi tidak stabil, sementara learning rate yang terlalu kecil membuat proses pelatihan lambat. Nilai epsilon dan epsilon decay juga penting untuk menjaga keseimbangan antara eksplorasi dan eksploitasi selama pelatihan.

Selain itu, penerapan reward shaping dengan memberikan penalti terhadap penggunaan aksi yang berlebihan terbukti membantu agen dalam mengendalikan energi yang digunakan, sehingga lebih efisien. Grafik reward per episode menunjukkan adanya tren peningkatan performa agen, meskipun fluktuasi tetap terjadi akibat sifat stokastik dari proses eksplorasi.

## **B. Diskusi**

Perbedaan mendasar antara Reinforcement Learning dan Supervised Learning pada sistem kendali terletak pada mekanisme pembelajaran. Pada supervised learning, model belajar dari pasangan input-output yang sudah diketahui labelnya, sedangkan pada reinforcement learning agen belajar melalui trial-and-error dengan menerima reward sebagai umpan balik. Hal ini membuat RL lebih cocok untuk sistem kendali adaptif yang beroperasi di lingkungan dinamis dan tidak sepenuhnya terdefinisi.

Strategi eksplorasi dan eksploitasi merupakan aspek penting dalam RL. Eksplorasi diperlukan agar agen dapat menemukan aksi baru yang mungkin lebih optimal, sementara eksploitasi digunakan untuk memanfaatkan aksi yang sudah terbukti menghasilkan reward tinggi. Optimasi kedua strategi ini dapat dilakukan dengan menggunakan skema epsilon decay atau dengan pendekatan yang lebih maju seperti Boltzmann exploration dan Upper Confidence Bound (UCB).

Dari hasil praktikum, dapat dilihat bahwa RL berpotensi diaplikasikan pada berbagai sistem kendali nyata. Misalnya, pengendalian lengan robot industri, optimasi penggunaan energi pada kendaraan listrik, pengaturan lalu lintas cerdas, hingga sistem manajemen energi pada smart grid. Dibandingkan dengan metode klasik seperti PID atau Fuzzy, RL memiliki keunggulan dalam adaptasi terhadap ketidakpastian, namun memerlukan biaya komputasi yang lebih tinggi serta data pelatihan yang memadai. Oleh karena itu, RL dapat dipandang sebagai pendekatan komplementer yang memperluas cakupan aplikasi kendali cerdas di masa depan.

## 5. Assignment

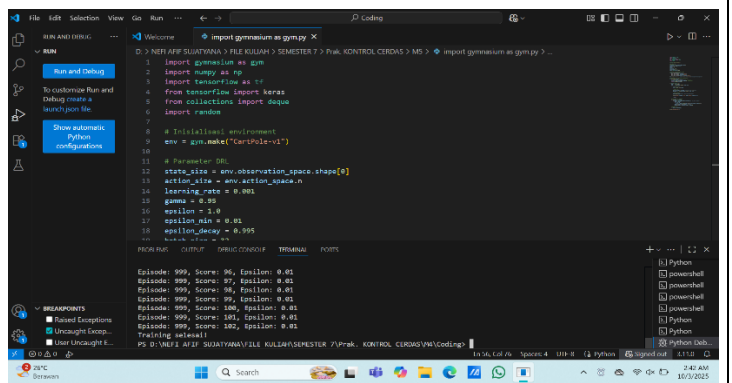
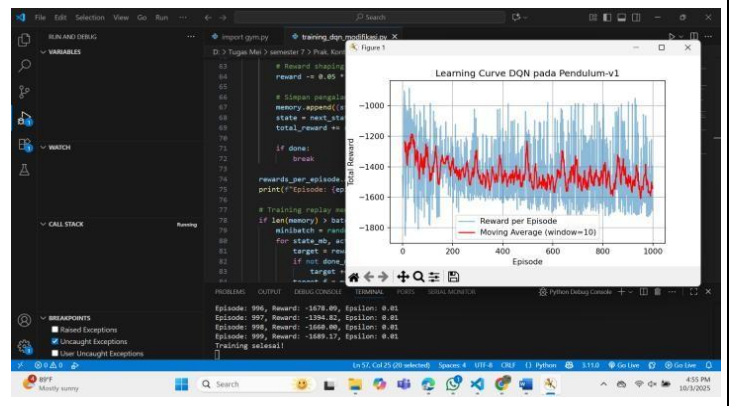
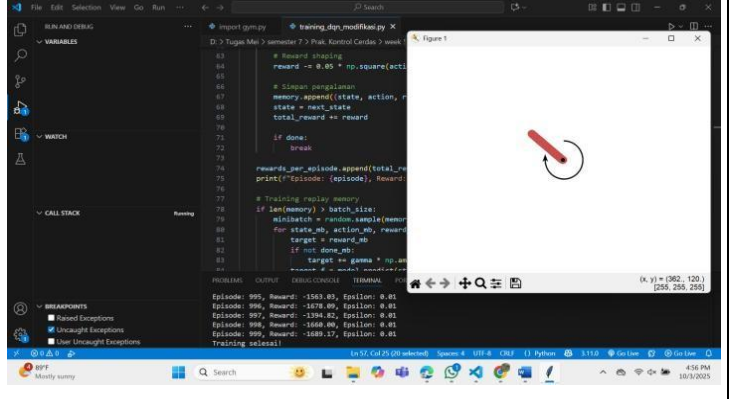
Pada tugas ini, mahasiswa diminta untuk memodifikasi kode Deep Q-Network (DQN) agar dapat diterapkan pada kasus kendali nyata yang berbeda, misalnya pada pengendalian lengan robot dua derajat kebebasan (robotic arm control) atau pada optimasi konsumsi energi kendaraan listrik. Mahasiswa harus menyesuaikan fungsi reward sehingga sesuai dengan tujuan kendali, seperti meminimalkan error posisi pada lengan robot atau mengurangi konsumsi energi sekaligus mempertahankan kecepatan target pada kendaraan listrik. Selain itu, mahasiswa juga diwajibkan mengimplementasikan metode kontrol klasik, yaitu PID Controller atau Fuzzy Logic Controller, pada kasus yang sama agar dapat dijadikan pembandingan.

Eksperimen dilakukan dengan melatih agen DQN dalam sejumlah episode tertentu, kemudian performanya dibandingkan dengan metode kontrol klasik berdasarkan beberapa metrik, antara lain akurasi tracking, stabilitas sistem (overshoot, settling time), efisiensi energi, serta ketahanan terhadap noise atau gangguan. Hasil eksperimen harus disajikan dalam bentuk tabel dan grafik, seperti reward terhadap episode, error tracking, dan konsumsi energi. Analisis dilakukan untuk menjelaskan kapan DQN lebih unggul dibandingkan metode kontrol klasik, serta kondisi di mana metode klasik lebih sesuai digunakan.

Seluruh kode sumber, konfigurasi eksperimen, grafik hasil, serta laporan singkat wajib diunggah ke GitHub dengan dokumentasi yang jelas pada file README.md. Laporan harus memuat pendahuluan, metode yang digunakan (DQN dan PID/Fuzzy), hasil eksperimen, analisis perbandingan, serta kesimpulan dan saran. Melalui tugas ini, mahasiswa diharapkan mampu memahami kelebihan dan keterbatasan algoritma DQN jika dibandingkan dengan pendekatan kontrol tradisional, serta mendapatkan wawasan tentang potensi penerapan Reinforcement Learning pada sistem kendali nyata.

## 6. Data dan Output Hasil Pengamatan

Data yang diperoleh pada praktikum minggu ke-5 dengan tabel dibawah ini:

No	Variabel	Hasil Pengamatan
1.	Hasil training DQN di modif dengan 1000 episode, didapat Score dan Episode dan saat training membutuhkan waktu 9 jam	
2.	Hasil training DQN sesudah di modif menggunakan 1000 Episode dan didapatkan hasil reward -1689.17 dan Epsilon 0.01. Saat training membutuhkan waktu kisaran 17 jam.	
3.	Hasil training DQN sesudah dimodif dengan 1000 Episode dan mendapat hasil reward -1689.17 dan epsilon 0.01. Pada saat training dibutuhkan waktu kurang lebih 17 jam.	

## 7. Kesimpulan

Penerapan Deep Q-Network (DQN) pada environment Pendulum-v1 menunjukkan bahwa agen dapat mempelajari strategi pengendalian yang lebih baik seiring bertambahnya episode pelatihan. Pada awalnya, reward rendah akibat eksplorasi, namun meningkat saat agen mulai mengeksploitasi aksi optimal. Parameter seperti learning rate, gamma, dan epsilon decay sangat

berpengaruh pada konvergensi dan stabilitas pelatihan. Teknik reward shaping juga membantu meningkatkan efisiensi pengendalian dengan mengoptimalkan penggunaan energi.

Meski DQN efektif pada sistem kendali yang kompleks dan dinamis, keterbatasannya dalam ruang aksi kontinu menuntut pengembangan metode lain seperti actor–critic (DDPG, SAC) untuk hasil yang lebih optimal

## 8. Saran

Disarankan untuk menambah jumlah episode pelatihan agar reward lebih stabil, serta melakukan perbandingan DQN dengan metode kontrol klasik seperti PID dan Fuzzy Logic untuk memahami kelebihan masing-masing. Penerapan RL juga bisa diperluas ke kasus yang lebih kompleks, seperti pengendalian lengan robot multi-DOF atau optimasi energi kendaraan listrik. Selain itu, penggunaan algoritma modern seperti DDPG, TD3, atau SAC dianjurkan untuk mengatasi keterbatasan DQN pada ruang aksi kontinu.

## 9. 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.
- Li, Y. (2017). Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*.
- François-Lavet, V., Henderson, P., Islam, R., Bellemare, M. G., & Pineau, J. (2018). An introduction to deep reinforcement learning. *Foundations and Trends in Machine Learning*, 11(3–4), 219–354.
- Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). OpenAI Gym. *arXiv preprint arXiv:1606.01540*.