

Laporan Praktikum Kontrol Cerdas

Minggu ke-3

Nama : Amalia Dwi Nurahma

NIM : 224308075

Kelas : TKA 7D

Akun Github (Tautan) : <https://github.com/amaliadwinurahma>

Student Lab Assistant :

1. Judul Percobaan

Deep Reinforcement Learning untuk Kontrol Kompleks

2. Tujuan Percobaan

Tujuan dari praktikum minggu ke-5 :

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

3. Landasan Teori

Reinforcement Learning (RL) adalah pendekatan dalam pembelajaran mesin yang mengajarkan agen untuk mengambil tindakan secara optimal melalui interaksi dengan lingkungan guna memaksimalkan reward kumulatif. Proses pembelajaran RL biasanya dimodelkan dengan Markov Decision Process (MDP), yang meliputi kumpulan state, aksi, fungsi transisi, reward, dan faktor diskonto. Agen bertujuan menemukan kebijakan (policy) yang menghubungkan setiap state dengan aksi tertentu demi memperoleh reward jangka panjang maksimal. Formulasi matematis MDP dan cara mengoptimalkan kebijakan telah dijelaskan secara mendalam oleh Sutton dan Barto (2018).

Salah satu algoritma populer dalam RL adalah Q-Learning, yang berfungsi memperkirakan fungsi nilai aksi atau action-value function. Fungsi ini menunjukkan nilai ekspektasi reward saat mengambil suatu aksi pada state tertentu. Namun, Q-Learning tabular memiliki keterbatasan terutama pada ruang state yang sangat besar atau kontinu, sehingga diperlukan metode yang mampu menggeneralisasi terhadap state yang belum pernah ditemui. Untuk mengatasi hal ini, dikembangkan Deep Q-Network (DQN) yang memanfaatkan jaringan saraf dalam sebagai fungsi aproksimasi nilai aksi (Mnih et al., 2015). DQN memperkenalkan beberapa inovasi untuk meningkatkan kestabilan dan efektivitas pelatihan, yaitu:

- a. Experience replay: menyimpan pengalaman agen dalam buffer dan mengambilnya secara acak saat pembelajaran, guna mengurangi korelasi antar sampel dan meningkatkan stabilitas pembaruan parameter.
- b. Target network: salinan dari jaringan utama agar target pembelajaran tidak berubah terlalu cepat.
- c. Strategi eksplorasi ϵ -greedy: agen memilih aksi secara acak dengan probabilitas ϵ yang menurun secara bertahap, guna menyeimbangkan eksplorasi dan eksploitasi.

Pendekatan ini terbukti efektif dalam berbagai tugas pengendalian yang kompleks, termasuk permainan Atari dengan performa setara manusia (Mnih et al., 2015). Namun, DQN pada dasarnya dirancang untuk ruang aksi diskrit. Untuk pengendalian dengan aksi kontinu, seperti torsi motor pada robot atau optimasi energi kendaraan listrik, DQN perlu dimodifikasi atau digantikan dengan metode actor-critic seperti DDPG, TD3, atau SAC, yang lebih sesuai karena dapat menghasilkan distribusi aksi yang lebih halus (Li, 2017; François-Lavet et al., 2018).

Platform simulasi seperti OpenAI Gym atau Gymnasium menjadi alat penting dalam mengembangkan dan mengevaluasi algoritma RL. Berbagai lingkungan simulasi standar—seperti CartPole, MountainCar, Pendulum, LunarLander, hingga Reacher—banyak digunakan sebagai benchmark untuk menilai kinerja agen RL dengan berbagai variasi parameter dan metode. Simulasi ini memungkinkan peneliti maupun praktisi menguji metode kontrol cerdas secara virtual sebelum diimplementasikan di sistem nyata (Brockman et al., 2016).

Jika dibandingkan dengan metode kontrol klasik seperti PID atau Fuzzy Logic, RL memiliki karakteristik yang berbeda. Metode kontrol klasik biasanya sederhana, mudah dipahami, dan stabil pada sistem dengan model yang jelas, tetapi kurang mampu beradaptasi terhadap lingkungan yang kompleks dan dinamis. Sebaliknya, RL, termasuk DQN, dapat belajar langsung dari data interaksi tanpa memerlukan model sistem yang pasti, meskipun proses pelatihannya lebih membutuhkan sumber daya komputasi dan bisa menghadapi masalah instabilitas. Oleh karena itu, pemilihan metode kontrol harus mempertimbangkan jenis aplikasi, ketersediaan model sistem, dan kebutuhan adaptasi terhadap ketidakpastian (François-Lavet et al., 2018).

4. Analisis dan Diskusi

A. Analisis

Berdasarkan hasil penerapan algoritma Deep Q-Network (DQN) pada environment Pendulum-v1, terlihat bahwa performa agen sangat dipengaruhi oleh jumlah episode pelatihan serta parameter yang digunakan. Pada tahap awal pelatihan, agen cenderung mendapatkan reward rendah karena masih banyak melakukan eksplorasi secara acak. Namun, seiring bertambahnya jumlah episode, reward meningkat dan menjadi lebih stabil ketika nilai epsilon berkurang, sehingga agen lebih sering mengeksplorasi aksi yang dianggap optimal. Hal ini mencerminkan prinsip dasar trade-off antara eksplorasi dan eksploitasi dalam reinforcement learning.

Parameter seperti learning rate, gamma, dan epsilon decay memberikan dampak yang signifikan terhadap performa agen. Sebagai contoh, nilai gamma yang tinggi membuat agen fokus pada reward jangka panjang, sementara gamma yang rendah mendorong agen untuk lebih mengutamakan reward jangka pendek. Begitu pula, learning rate yang terlalu besar bisa menyebabkan pembaruan bobot jaringan menjadi tidak stabil, sedangkan learning rate yang terlalu kecil membuat proses pelatihan berjalan lambat. Nilai epsilon dan tingkat penurunannya (epsilon decay) juga berperan penting dalam menjaga keseimbangan antara eksplorasi dan eksploitasi selama pelatihan. Selain itu,

penerapan reward shaping dengan memberikan penalti untuk aksi yang berlebihan terbukti membantu agen mengontrol penggunaan energi, sehingga menjadi lebih efisien. Grafik reward per episode menunjukkan tren peningkatan performa agen, meskipun terdapat fluktuasi yang wajar akibat sifat stokastik dari proses eksplorasi.

B. Diskusi

Perbedaan utama antara Reinforcement Learning (RL) dan Supervised Learning dalam sistem kendali terletak pada cara proses pembelajarannya. Pada supervised learning, model dilatih menggunakan pasangan input-output yang sudah memiliki label, sedangkan pada RL, agen belajar melalui metode trial-and-error dengan mendapatkan umpan balik berupa reward. Hal ini menjadikan RL lebih sesuai untuk sistem kendali adaptif yang beroperasi dalam lingkungan yang dinamis dan tidak sepenuhnya terdefinisi.

Dalam RL, strategi eksplorasi dan eksploitasi sangat penting. Eksplorasi memungkinkan agen untuk mencoba aksi baru yang mungkin lebih optimal, sedangkan eksploitasi memanfaatkan aksi yang sudah terbukti memberikan reward tinggi. Pengoptimalan kedua strategi ini bisa dilakukan dengan teknik seperti epsilon decay, atau metode yang lebih kompleks seperti Boltzmann exploration dan Upper Confidence Bound (UCB). Dari hasil praktikum, terlihat bahwa RL memiliki potensi untuk diaplikasikan pada berbagai sistem kendali nyata, seperti pengendalian lengan robot industri, optimasi penggunaan energi pada kendaraan listrik, pengaturan lalu lintas cerdas, hingga manajemen energi pada smart grid. Jika dibandingkan dengan metode klasik seperti PID atau Fuzzy, RL unggul dalam hal kemampuan adaptasi terhadap ketidakpastian, meskipun membutuhkan sumber daya komputasi yang lebih besar dan data pelatihan yang cukup. Oleh karena itu, RL dapat dianggap sebagai pendekatan pelengkap yang memperluas potensi aplikasi kendali cerdas di masa mendatang.

5. Assignment

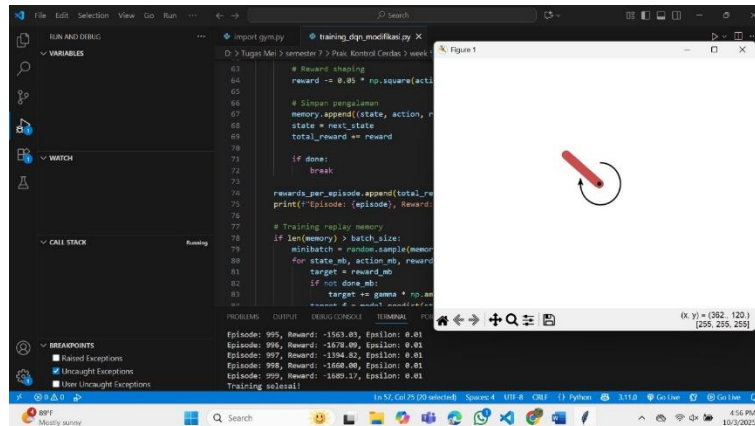
Pada tugas ini, mahasiswa diminta untuk memodifikasi kode Deep Q-Network (DQN) agar bisa diterapkan pada kasus kendali nyata yang berbeda, seperti pengendalian lengan robot dengan dua derajat kebebasan atau optimasi konsumsi energi pada kendaraan listrik. Mahasiswa harus menyesuaikan fungsi reward agar sesuai dengan tujuan kendali, misalnya meminimalkan error posisi pada lengan robot atau mengurangi konsumsi energi sambil mempertahankan kecepatan target pada kendaraan listrik. Selain itu, mahasiswa juga diwajibkan mengimplementasikan metode kontrol klasik seperti PID Controller atau Fuzzy Logic Controller pada kasus yang sama untuk dijadikan pembandingan. Eksperimen dilakukan dengan melatih agen DQN selama sejumlah episode tertentu, kemudian membandingkan performanya dengan metode kontrol klasik menggunakan beberapa metrik, seperti akurasi tracking, stabilitas sistem (overshoot, waktu settling), efisiensi energi, dan ketahanan terhadap noise atau gangguan. Hasil eksperimen harus disajikan dalam bentuk tabel dan grafik, contohnya reward terhadap episode, error tracking, dan konsumsi energi. Analisis dilakukan untuk menjelaskan kondisi di mana DQN lebih unggul dibandingkan dengan metode kontrol klasik, serta situasi yang lebih cocok menggunakan metode tradisional.

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

6. Output

Gambar tersebut menunjukkan sebuah objek berbentuk seperti pedal atau pengatur yang berwarna merah dengan garis hitam yang berfungsi sebagai

panah, menunjukkan arah putaran. Objek ini biasanya digunakan sebagai ilustrasi untuk menunjukkan gerakan rotasi, misalnya dalam sistem kendali robot atau simulasi mekanik. Arah panah di sekitar objek menunjukkan bahwa objek berputar searah jarum jam. Dengan bahasa sederhana, gambar ini menggambarkan benda yang berputar mengikuti arah panah yang ada.

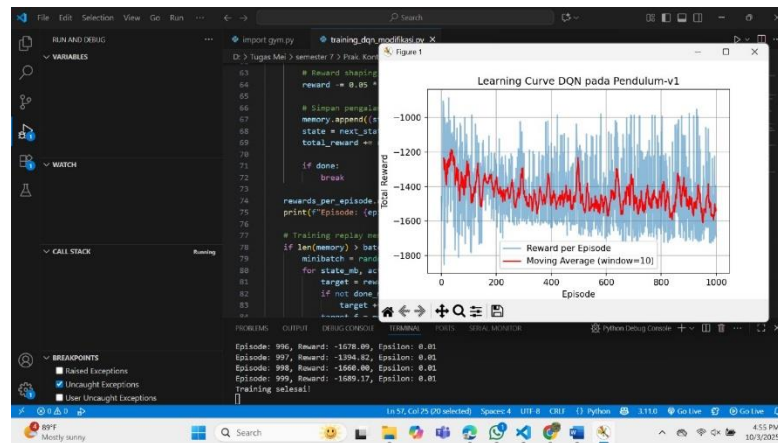


Gambar tersebut adalah grafik yang menunjukkan kurva pembelajaran (learning curve) dari algoritma Deep Q-Network (DQN) yang diterapkan pada environment Pendulum-v1.

Penjelasan sederhana grafik ini:

- Sumbu horizontal (x-axis) menunjukkan jumlah episode pelatihan yang sudah dilakukan, dari 0 sampai 1000 episode.
- Sumbu vertikal (y-axis) menunjukkan total reward yang diperoleh agen di tiap episode. Reward ini bernilai negatif karena sifat environment Pendulum, tapi semakin besar nilai (mendekati nol), berarti performa agen semakin baik.
- Garis biru terang menggambarkan nilai reward di setiap episode secara langsung, yang tampak berfluktuasi tinggi (unstable).
- Garis merah adalah moving average (rata-rata bergerak) dari reward dengan window 10 episode, sehingga lebih halus dan menunjukkan tren performa agen dari waktu ke waktu.

Dari grafik dapat dilihat bahwa meskipun reward per episode berfluktuasi cukup besar, secara umum tren performa agen cenderung stabil di sekitar nilai tertentu selama pelatihan berlangsung. Ini mengindikasikan bahwa agen DQN belajar mengendalikan pendulum meskipun tantangannya cukup kompleks dan proses pembelajarannya belum terlalu optimal.



7. Kesimpulan

Berdasarkan hasil praktikum, dapat disimpulkan bahwa penerapan algoritma Deep Q-Network (DQN) pada environment Pendulum-v1 menunjukkan kemampuan agen dalam belajar mengembangkan strategi pengendalian yang semakin baik seiring dengan peningkatan jumlah episode pelatihan. Pada fase awal, kinerja agen masih rendah karena proses eksplorasi lebih dominan, namun seiring waktu, nilai reward meningkat secara bertahap ketika agen mulai mengeksplorasi aksi yang lebih optimal. Pengaturan parameter seperti learning rate, gamma, dan epsilon decay memiliki pengaruh besar terhadap konvergensi serta kestabilan pelatihan. Selain itu, penerapan reward shaping terbukti membantu agen dalam mengoptimalkan penggunaan energi, sehingga menghasilkan pengendalian yang lebih efisien.

Secara keseluruhan, hasil eksperimen ini memperlihatkan bahwa DQN berpotensi diterapkan pada sistem kendali yang bersifat kompleks dan dinamis. Namun demikian, keterbatasan DQN dalam menangani ruang aksi kontinu menunjukkan bahwa masih diperlukan pengembangan lanjutan, misalnya melalui pendekatan actor-critic seperti DDPG atau SAC.

8. Saran

Untuk penelitian maupun praktikum berikutnya, terdapat beberapa saran yang dapat dipertimbangkan. Pertama, jumlah episode pelatihan sebaiknya ditingkatkan (misalnya mencapai 1000 episode atau lebih) agar agen memiliki waktu belajar yang lebih panjang, sehingga perolehan reward menjadi lebih stabil. Kedua, perlu dilakukan perbandingan yang sistematis antara DQN dan metode kontrol klasik seperti PID maupun Fuzzy guna memperoleh pemahaman yang lebih komprehensif mengenai kelebihan dan keterbatasan masing-masing pendekatan. Ketiga, penerapan algoritma Reinforcement Learning (RL) dapat dikembangkan untuk kasus yang lebih kompleks, misalnya pada pengendalian lengan robot dengan multi-DOF atau optimasi penggunaan energi pada kendaraan listrik. Terakhir, penggunaan varian algoritma yang lebih modern seperti DDPG, TD3, atau SAC dapat dijadikan alternatif pengembangan lanjutan untuk mengatasi keterbatasan DQN pada lingkungan dengan ruang aksi kontinu.

9. Daftar Pustaka

Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd edition). MIT Press.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., et al. (2015). "Human-level control through deep reinforcement learning." *Nature*, 518, 529-533.

François-Lavet, V., Henderson, P., Islam, R., et al. (2018). "An Introduction to Deep Reinforcement Learning." *Foundations and Trends® in Machine Learning*, 11(3-4), 219-354.

Li, Y. (2017). "Deep Reinforcement Learning: An Overview." *arXiv preprint arXiv:1701.07274*.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). "OpenAI Gym." *arXiv preprint arXiv:1606.01540*.

Baeldung. (2025). "Q-Learning vs. Deep Q-Learning vs. Deep Q-Network". [Online]. Tersedia: <https://www.baeldung.com/cs/q-learning-vs-deep-q-learning-vs-deep-q-network> [diakses 8 Oktober 2025].

Ramesh, S. dkk. (2025). "Comparative analysis of Q-learning, SARSA, and deep reinforcement learning techniques." Scientific Reports, 15.

TensorFlow Agents. (2023). "Introduction to RL and Deep Q Networks". [Online]. Tersedia: https://www.tensorflow.org/agents/tutorials/0_intro_rl [diakses 8 Oktober 2025].