



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang Masalah**

Pada umumnya, game dan simulasi mengambil bentuk dunia tiruan dengan lingkungan tertutup dan terkontrol yang memungkinkan dilakukannya pengujian suatu teknik yang hasilnya dapat diterapkan di dunia nyata (Salem, dkk, 2019). Game dan simulasi dibutuhkan untuk mengembangkan pemahaman mengenai sebuah sistem dengan mengamati bagaimana sistem tiruan tersebut beroperasi. Penelitian oleh Salem, dkk (2019) dan Sazaki, dkk (2017) menunjukkan bahwa penerapan kecerdasan buatan dalam pengembangan game terbukti berguna. Hal ini dimungkinkan ketika sebuah agen yang cerdas belajar untuk melakukan sebuah tugas dengan metode *trial and error* tanpa dioperasikan langsung oleh manusia (Goodfellow, dkk, 2017).

Russell dan Norvig dalam buku “Artificial Intelligence: A Modern Approach” mengatakan bahwa agen adalah apapun yang dapat mengamati lingkungan melalui sensor dan merespon lingkungan melalui aktuator. Russell dan Norvig mengadopsi pandangan bahwa kecerdasan sangat berkaitan dengan tindakan rasional, sehingga idealnya agen yang cerdas mengambil tindakan terbaik berdasarkan situasi yang ada (2010). Sehingga game dan simulasi dapat dimanfaatkan untuk memahami dan memvisualisasikan bagaimana perkembangan kecerdasan sebuah agen dalam pembelajaran mesin.

Unity, sebagai salah mesin pengembang game, berada di antara perkembangan pembelajaran mesin dan perkembangan game. Menurut Juliani (2017), Unity mengambil peran untuk membantu peneliti pembelajaran mesin dengan menyediakan skenario untuk training dan membantu pengembang game untuk memanfaatkan teknologi pembelajaran mesin terkini. Pada tahun 2017, Unity memperkenalkan Unity Machine Learning Agents Toolkit (Unity ML-Agents Toolkit), proyek *open-source* yang menjadikan game dan simulasi sebagai lingkungan untuk melatih kecerdasan agen. Pada tahun 2020, Unity ML-Agents Toolkit v1.0 dirilis bersamaan dengan beberapa implementasinya pada proyek game dan simulasi yang disediakan oleh Unity (Mattar, dkk, 2020).

Salah satu implementasi Unity ML-Agents Toolkit diterapkan pada Unity Karting Microgame, sebuah *3D kart racing game* yang dapat dimodifikasi dan dikustomisasi oleh pengguna Unity (Unity Learn, 2020). Unity Karting Microgame mengambil konsep simulasi *racing game* dimana *kart* dikendalikan untuk menyelesaikan lintasan. Menurut Sazaki, dkk (2017), *racing game* adalah game berbentuk simulasi yang membuat tiruan balapan untuk mendapatkan pemenang yang dapat mencapai garis akhir dengan waktu terbaik. Implementasi Unity ML-Agents Toolkit pada Unity Karting Microgame menghasilkan *self-driving kart*, agen cerdas dalam bentuk *kart* yang dapat mengemudikan dirinya sendiri tanpa bantuan manusia untuk dapat menyelesaikan lintasan. Sehingga Unity Karting Microgame termasuk dalam *robot path planning problem* yang menurut Choueiry, dkk (2019) diartikan sebagai permasalahan yang meminta solusi *collision-free path* (lintasan bebas tabrakan) dalam suatu lingkungan.

Penelitian oleh Arora, dkk (2014) dan Choueiry, dkk (2019) tentang penerapan kecerdasan buatan dalam menyelesaikan *robot path planning problem* menyimpulkan bahwa penggunaan algoritma genetik menunjukkan hasil yang sangat efektif dan efisien. Algoritma genetik adalah algoritma berbasis pencarian dengan mekanisme seperti seleksi alam. Martiana, dkk (2014) menjelaskan bahwa pencarian dimulai dengan membuat populasi awal yang terdiri dari sejumlah individu yang ditempatkan di suatu lingkungan. Setiap individu dievaluasi kemampuannya untuk bertahan hidup (*fitness*) untuk selanjutnya diseleksi berdasarkan nilai *fitness*-nya. Individu yang berhasil lolos seleksi akan menjadi induk untuk populasi berikutnya dan populasi baru yang terbentuk akan masuk kembali ke proses evaluasi *fitness*. Siklus ini akan terus berulang hingga didapatkan solusi (individu) dengan *fitness* yang terbaik.

Penelitian oleh Immanuel (2019) tentang implementasi evolusi jaringan saraf tiruan (*neuroevolution*) pada *autonomous mobile robot navigation* juga menunjukkan keberhasilan penerapan kecerdasan buatan dalam menyelesaikan *robot path planning problem*. Jaringan saraf tiruan adalah model logika yang menyerupai unit pemrosesan informasi (neuron) dalam otak manusia (Sher, 2012). Menurut Negnevitsky (2011), dalam kecerdasan buatan, jaringan saraf tiruan memiliki kemampuan untuk belajar dan dapat digunakan sebagai otak pada agen cerdas. *Neuroevolution* mengintegrasikan algoritma genetik dengan jaringan saraf tiruan untuk mencari jaringan saraf tiruan terbaik dengan proses seleksi alam (Iba, 2018). Jaringan saraf tiruan terbaik ini kemudian dijadikan sebagai otak dari

individu (agen cerdas). Sehingga pendekatan menggunakan algoritma genetik dan *neuroevolution* untuk menyelesaikan *robot path planning problem* dinilai efektif.

Dalam penelitian oleh Immanuel (2019) juga disimpulkan bahwa desain arsitektur (topologi) jaringan saraf tiruan berpengaruh pada tingkat keberhasilan simulasi. Topologi jaringan saraf tiruan disusun oleh *input nodes* sebagai penerima sinyal masuk, *hidden nodes* sebagai tempat sinyal diproses, *output nodes* sebagai pemberi sinyal keluar, dan *connections* sebagai penghubung antar *node* (Iba, 2018). Menurut Stork, dkk (2019) salah satu masalah *neuroevolution* terjadi pada evaluasi *fitness*. Heidenreich (2019) berpendapat semakin banyak jumlah *nodes* dan *connections* pada topologi maka jaringan saraf tiruan menjadi semakin kompleks dan komputasi yang dibutuhkan saat evaluasi *fitness* semakin tinggi. Menurutnya, untuk menyelesaikan masalah ini diterapkan optimisasi, yaitu penyusutan kompleksitas dengan mengurangi jumlah *nodes* dan *connections* agar ditemukan topologi jaringan saraf tiruan yang optimal.

Penelitian oleh Kenneth O. Stanley dan Risto Miikkulainen (2002), dengan judul *Evolving Neural Networks through Augmenting Topologies*, mengajukan metode *neuroevolution* yang berbeda dengan nama *NeuroEvolution of Augmenting Topologies* (NEAT). Menurut Heidenreich (2019), pada umumnya, jaringan saraf tiruan dibangun dari kompleksitas tinggi lalu disederhanakan (optimisasi), sedangkan Stanley dan Miikkulainen mengajukan teknik *augmenting topologies* atau topologi jaringan saraf tiruan yang arsitekturnya bertambah seiring evolusi terjadi. Ide dasar NEAT adalah membangun jaringan saraf tiruan yang dimulai dari jumlah *nodes* dan *connection* yang paling kecil atau sederhana lalu mengevolusikan

arsitekturanya seiring berjalan waktu jika dan hanya jika evolusi tersebut berguna dan diperlukan. Penelitian lain oleh Stanley dan Miikkulainen yang masuk dalam *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2002)* berhasil mendapatkan penghargaan *Best Paper Award in Genetic Algorithms* dan menunjukkan bahwa performa NEAT sangat baik dibanding dengan *Cellular Encoding* atau *Enforced Subpopulations (ESP)*. Performa NEAT dalam menyelesaikan *pole balancing tasks* 25 kali lebih cepat dibanding dengan *Cellular Encoding* serta 5 kali lebih cepat dibandingkan dengan ESP. NEAT juga diterapkan dalam penelitian oleh Yuksel (2018) untuk mencari agen cerdas yang dapat mengevakuasi diri dalam suatu lingkungan berbahaya dan melalui penelitiannya didapatkan kesimpulan bahwa NEAT terbukti efektif.

Perkembangan kecerdasan buatan dan perkembangan game saling terkait. Game dan simulasi dapat dimanfaatkan untuk memahami dan memvisualisasikan bagaimana perkembangan kecerdasan sebuah agen dalam pembelajaran mesin. Unity ML-Agents Toolkit sebagai proyek *open-source* untuk melatih kecerdasan agen sudah diimplementasikan pada berbagai game yang dikeluarkan Unity, salah satunya adalah Unity Karting Microgame. Unity Karting Microgame mengambil konsep simulasi *racing game* dimana *kart* dikendalikan untuk menyelesaikan lintasan. Implementasi Unity ML-Agents Toolkit pada Unity Karting Microgame menghasilkan *self-driving kart*, agen cerdas dalam bentuk kart yang dapat mengemudikan dirinya sendiri tanpa bantuan manusia untuk dapat menyelesaikan lintasan, sehingga Unity Karting Microgame termasuk dalam *robot path planning problem*. Penelitian menurut Arora, dkk (2014), Choueiry, dkk (2019), dan

Immanuel (2019) menunjukkan bahwa algoritma genetik, khususnya *neuroevolution*, terbukti berguna untuk menyelesaikan *robot path planning problem*. Penelitian oleh Stanley dan Miikkulainen tentang NEAT sebagai metode *neuroevolution* dihargai *Best Paper Award in Genetic Algorithms* dalam GECCO-2002 dan performa NEAT yang terbukti efektif didukung oleh kesimpulan dalam penelitian oleh Yuksel (2018). Dalam implementasi Unity ML-Agents Toolkit pada Unity Karting Microgame, NEAT belum diterapkan pada sebagai metode pengembangan kecerdasan untuk *self-driving kart*, maka penelitian ini akan mengimplementasi *Neuroevolution of Augmenting Topologies* untuk *self-driving kart* pada Unity Karting Microgame.

## **1.2 Rumusan Masalah**

Berdasarkan topik utama penelitian yang diteliti dan telah dijabarkan sebelumnya, masalah yang dirumuskan adalah sebagai berikut..

1. Bagaimana cara mengimplementasi *Neuroevolution of Augmenting Topologies* untuk *self-driving kart* pada Unity Karting Microgame.
2. Bagaimana hubungan kompleksitas dan *fitness* dalam implementasi NEAT untuk *self-driving kart* pada Unity Karting Microgame.

## **1.3 Batasan Masalah**

Batasan-batasan masalah pada penelitian ini adalah sebagai berikut.

1. Simulasi dirancang berdasarkan *3D kart racing game template* dari Unity Karting Microgame.

2. Individu adalah kart dari Unity Karting Microgame dengan jaringan saraf tiruan yang dibangun menggunakan NEAT untuk menjadi *self-driving kart*.
3. Dalam simulasi, beberapa individu dijalankan bersamaan dan *collision* antar individu tidak mempengaruhi nilai *fitness* individu tersebut. Individu juga tidak dipengaruhi oleh kondisi lintasan dan/atau cuaca.
4. Objektif simulasi adalah mendapatkan individu terbaik yang memiliki nilai *fitness* paling tinggi dan dapat menyelesaikan lintasan dengan bebas tabrakan.
5. Jaringan saraf tiruan untuk setiap individu memiliki 5 *input node* dan 2 *output node*. Nilai untuk 5 *input node* didapatkan dari 5 buah sensor pada kart untuk membaca jarak dari kart ke tepi lintasan. Nilai dari 2 *output node* digunakan untuk pergerakan *self-driving kart* pada *steering* (kiri/kanan) dan *acceleration* (gas/rem).
6. Nilai *fitness* dihitung berdasarkan *checkpoint* pada lintasan dan waktu yang dibutuhkan kendaraan dari satu *checkpoint* ke *checkpoint* yang berikutnya.

#### **1.4 Tujuan Penelitian**

Tujuan dan fokus utama penelitian berdasarkan rumusan masalah adalah sebagai berikut.

1. Mengimplementasi *Neuroevolution of Augmenting Topologies* untuk *self-driving kart* pada Unity Karting Microgame.
2. Membandingkan hubungan kompleksitas dan *fitness* dalam implementasi NEAT untuk *self-driving kart* pada Unity Karting Microgame.



## 1.5 Manfaat Penelitian

Manfaat dari implementasi mengimplementasi *Neuroevolution of Augmenting Topologies* untuk *self-driving kart* pada Unity Karting Microgame adalah membantu peneliti memahami implementasi pembelajaran mesin dalam bentuk simulasi visual dan membandingkan hubungan kompleksitas dan *fitness* dalam implementasi NEAT untuk *self-driving kart* pada Unity Karting Microgame. Selain itu, hasil penelitian ini dapat dijadikan acuan untuk penelitian selanjutnya terkait NEAT dan Unity Karting Microgame.

## 1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan untuk penyajian laporan skripsi ini terdiri dari BAB I, BAB II, BAB III, BAB IV, dan BAB V, masing-masing akan diuraikan sebagai berikut.

### BAB I PENDAHULUAN

Bab ini terdiri dari enam bagian, yaitu latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan laporan.

### BAB II LANDASAN TEORI

Bab ini berisi landasan teori yang digunakan untuk penelitian, yaitu game dan simulasi, Unity Karting Microgame dan *self-driving kart*, algoritma genetik, jaringan saraf tiruan, dan *Neuroevolution of Augmenting Topologies*.

### BAB III METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

Bab ini berisi penjelasan metode penelitian yang digunakan dan perancangan sistem yang terdiri dari rancangan *use case diagram* untuk simulasi, flowchart, serta rancangan antarmuka pengguna.

### BAB IV IMPLEMENTASI DAN ANALISIS

Bab ini berisi pembahasan mengenai implementasi NEAT pada *karting microgame* dan analisis dari simulasi yang telah dibangun dengan melihat hubungan kompleksitas dan *fitness* yang dihasilkan.

### BAB V SIMPULAN DAN SARAN

Bab ini berisi simpulan dari hasil penelitian dan analisis yang dilakukan dalam penelitian, serta saran yang dapat digunakan untuk pengembangan aplikasi maupun penelitian lebih lanjut.