BAB 1
PENDAHULUAN

1.1 Latar Belakang Masalah

NPC atau *Non Player Character* adalah salah satu karakter dalam *game* yang tidak dikontrol oleh pemain. NPC dikontrol oleh kecerdasan buatan. Ada banyak algoritma optimasi yang dapat diterapkan untuk menentukan gerak NPC diantaranya algoritma *particle swarm optimization* (PSO) dan *differential evolution Algorithm* (DEA). Berdasarkan penelitian yang dilakukan Yoga Baskara [8], implementasi PSO pada *game action*-RPG berjalan dengan baik dalam menentukan perilaku NPC (*non-player character*) mendekati *player* dan menetukan NPC yang menjadi pemimpin. Dari hasil penelitian tersebut dari 10 NPC membutuhkan 15-18 kali proses iterasi untuk dapat menentukan NPC yang menjadi pemimpin pada koordinat *player* yang diam.

Berdasarkan penelitian dari Madehan Anis [3] telah dilakukan perbandingan PSO dan DEA dengan mengoptimalkan index performa, hasil dari perbandingan tersebut menghasilkan kesimpulan algoritma DEA mempunyai kecepatan komputasi yang lebih tinggi dibandingkan dengan PSO. Dalam kasus ini, secara umum DEA mempunyai respon waktu yang lebih baik dibanding PSO dimana sistem akan lebih cepat menuju keadaan *steady state*. Algoritma PSO menghasilkan indeks performansi optimum sebesar 6.579 dengan membutuhkan waktu kinerja rata-rata CPU (CPU *time*) dalam dua puluh kali *running* sebesar 21.14403 detik dan DEA menghasilkan indeks performansi optimum sebesar 6.417 dengan membutuhkan waktu kinerja rata-rata CPU time dalam dua puluh kali *running* sebesar 10.60157 detik. Perbedaan komputasi algoritma PSO dan DEA dipengaruhi oleh parameter-parameter yang terlibat didalamnya. Namun, tidak menutup kemungkinan untuk studi kasus yang berbeda dan penggunaan nilai parameter yang berbeda akan menunjukan pola hasil yang berbeda pula [3]. Maka dari itu algoritma PSO dan DEA perlu diuji kembali untuk menentukan performansi kedua algoritma pada pergerakan NPC.

Berdasarkan latar belakang tersebut, algoritma PSO dan DEA akan diimplementasikan pada *simulasi perilaku NPC secara berkelompok* untuk mengetahui kemungkinan algoritma DEA tetap lebih unggul dibandingkan algoritma PSO dalam mengoptimalkan waktu proses dan iterasi. Perbandingan PSO dan DEA pada simulasi action-RPG dipilih karena kedua algoritma merupakan algoritma yang dapat digunakan untuk optimasi pergerakan beberapa objek baik pada gerak lateral pesawat maupun pergerakan NPC pada *game*.

1.2 Identifikasi Masalah

Dengan latar belakang yang telah diuraikan diatas, adapun identifikasi masalah yang ada adalah bagaimana setiap NPC dalam suatu kelompok NPC dapat memilih NPC mana yang memiliki posisi terbaik untuk dijadikan pemimpin dan diikuti NPC lain dalam mengejar karakter *player* menggunakan DEA lalu membandingkan waktu proses dan iterasi untuk menuju keadaan steady state atau koordinat *player* antara penggunaan algoritma PSO dan DEA.

1.3 Maksud dan Tujuan

Berdasarkan permasalahan yang diteliti, maka maksud dan tujuan dari penelitian:

1.3.1 Maksud

Membandingkan antara algoritma PSO dan DEA yang lebih mengoptimalkan pergerakan NPC pada *simulasi perilaku NPC secara berkelompok* dengan mengoptimalkan waktu proses dan iterasi.

1.3.2 Tujuan

Mengetahui performansi dari perbandingan kedua algoritma dari segi waktu proses dan iterasi paling optimal untuk perilaku NPC dan langkah optimal menuju keadaan *steady state* dimana NPC dapat mencapai posisi atau koordinat *player*.

1.4 Batasan Masalah

Dalam perbandingan kedua algoritma, perlu diberikan beberapa batasan agar mempermudah saat melakukan perancangan. Berikut batasan masalahnya :

1. Simulasi dibuat untuk membandingkan performansi waktu proses dan iterasi yang dibutuhkan NPC menuju posisi atau koordinat *player*.
2. Algoritma yang digunakan adalah algoritma *particle swarm optimization* (PSO) dan *algoritma defferential evolution* (DEA).
3. Terdapat 10 NPC.
4. Posisi awal tiap partikel sudah ditentukan.
5. Simulasi hanya menunjukan bagaimana kelompok NPC mendekati karakter *player*.
6. Aplikasi simulasi berbasis desktop.

1.5 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam penulisan tugas akhir adalah metode penelitian kualitatif. Metode ini adalah salah satu jenis metode yang lebih menekankan pada aspek pemahaman secara mendalam terhadap suatu masalah daripada melihat permasalahan untuk penelitian generalisasi. Metode penelitian ini lebih suka menggunakan teknik analisis mendalam.Metodologi penelitian dalam skripsi ini meliputi metode pengumpulan data dan metode pembangunan perangkat lunak.

1. Tahap Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian adalah study literatur. Study literatur merupakan tahapan pengumpulan data dengan cara mengumpulkan literatur, jurnal, paper dan bacaan-bacaan yang ada kaitannya dengan judul penelitian yaitu perbandingan algoritma *particle swarm optimization* (PSO) dan *differential evolution algorithm* (DEA) untuk perilaku NPC menyerang secara berkelompok.

2. Tahap Pembangunan Perangkat Lunak

Model ini mengusulkan sebuah pendekatan kepada perkembangan aplikasi yang sistematik yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, dan pengujian.

1. Analisis Kebutuhan

   Tahap analisis kebutuhan ini yaitu untuk memahami sistem yang ada pada permainan *action*-RPG serta mendefinisikan permasalahan sistemnya dan selanjutnya untuk menentukan kebutuhan sistem secara garis besar.

2. Desain Sistem

   Tahap desain sistem ini yaitu untuk memberikan gambaran umum kepada pengguna mengenai permainan *action*-RPG yang akan dibangun dan rancang bangun yang lengkap tentang sistem permainan *action*-RPG yang akan dikembangkan.

3. Penulisan Kode Program

   Tahap penulisan kode program yaitu untuk merancang sistem dengan menggunakan bahasa pemrograman matlab. Tahap ini dilakukan pembuatan komponen-komponen sistem yang meliputi modul program dan antarmuka

4. Pengujian Program

   Tahap pengujian program ini dilakukan untuk mendapatkan serta memastikan bahwa NPC pada simulasi *action*-RPG yang dibangun sesuai dengan kebutuhan yang telah dideskripsikan.

1.6 Sistematika penulisan

Sistematika penulisan tugas akhir disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika penulisan tugas akhir ini adalah sebagai berikut :

BAB 1 PENDAHULUAN

Bab ini berisi penjelasan mengenai latar belakang masalah, identifikasi masalah, maksud dan tujuan, batasan masalah, metodologi penelitian serta sistematika penulisan.

BAB 2 LANDASAN TEORI

Membahas berbagai konsep dasar dan teori-teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis

permasalahan serta tinjauan terhadap penelitian-penelitian serupa yang telah pernah dilakukan sebelumnya termasuk sintesisnya.

BAB 3 ANALISIS DAN KEBUTUHAN ALGORITMA

Menganalisis masalah dari model penelitian, menggambarkan identifikasi masalah, analisis kebutuhan data, *hardware*, *software*, *brainware*, diagram pembuatan sistem dan UML (*Unified Modeling Language*).

BAB 4 IMPLEMENTASI DAN PENGUJIAN ALGORITMA

Merupakan tahapan yang dilakukan dalam penelitian dalam menjelaskan implementasi, tampilan antarmuka, menu yang tersedia pada sistem, dan pengujian terhadap sistem.

BAB 5 KESIMPULAN DAN SARAN

Bab ini berisi mengenai hal-hal penting yang telah dibahas dan kemudian dibuat kesimpulan. Bab ini juga berisi saran-saran yang diberikan untuk pengembang selanjutnya.