

# BAB 1

## PENDAHULUAN

### Latar Belakang Masalah

Auto Chess merupakan sebuah permainan bergenre *strategy* dan berjenis *board game* yang dapat dimainkan pada *platform* android. *Game* ini merupakan *costume game* dan mode untuk game Dota Auto Chess yang merupakan *mod* pada permainan Dota 2. Pada *game* ini melibatkan beberapa pemain, para pemain akan melakukan fase-fase dasar *game* Auto Chess, yaitu membeli bidak, melakukan *upgrade* dan juga melakukan pengaturan *line up* hingga melakukan fase battle dengan pemain lain secara acak. Pemenang permainan ditentukan oleh pemain yang berhasil bertahan hingga semua pemain lain kehabisan *hit point*[1].

Pada game strategi Auto Chess terdapat 65 karakter dan karakteristik yang berbeda setiap karakternya. Dengan jumlah karakter yang cukup banyak, dan juga keterhubungan setiap fase terhadap fase lain hingga berpengaruh terhadap kemenangan yang diraih, maka tidaklah mudah untuk memenangkan suatu pertandingan dalam game ini. Perlu adanya pengambilan keputusan yang tepat untuk setiap fase sehingga pemain dapat memenangkan suatu pertandingan dan memaksimalkan point yang diperoleh.

Beberapa penelitian sebelumnya dalam bidang *game* strategi pernah dilakukan, yaitu pada DOTA 2 yang menggunakan Neural Network Backproagation menghasilkan akurasi kemenangan sebesar 50,97% pada penelitian ini terdapat beberapa permasalahan yaitu dibutuhkannya data pertandingan untuk proses training yang sangat besar, sehingga banyaknya data sangat berpengaruh terhadap akurasi yang didapatkan[2]. Kemudian penelitian pada *game* manajemen sumber daya dengan menggunakan metode SHOP menghasilkan akurasi sebesar 80%, namun pada penelitian ini terdapat beberapa permasalahan, salah satunya akurasi yang dihasilkan dipengaruhi oleh kompleksitas serta besarnya ruang lingkup *environment*, sehingga akurasi yang didapatkan akan menurun seiring meningkatnya kompleksitas serta besarnya ruang lingkup sebuah *environment* [3].

*Reinforcement Learning* adalah metode pembelajaran untuk memetakan setiap state terhadap *action* yang dipilih untuk memaksimalkan *reward* yang diterima, [4]. Beberapa penelitian lain tentang *Reinforcement Learning* dalam *game* bergenre *strategy* pernah dilakukan sebelumnya. Pada penelitian penerapan *Reinforcement Learning* dalam *game strategy* yaitu Start Craft berkesimpulan bahwa *Reinforcement Learning* dapat menyesuaikan dengan kondisi *environment* yang luas dan kompleks, tingkat akurasi yang dihasilkan yaitu sebesar 65% dalam waktu 15 menit untuk sebuah aksi(kondisi) dan terus meningkat setiap waktunya[5]. pada penelitian penerapan *Reinforcement Learning* yang menggunakan QLearning dalam *game strategy* Collection berkesimpulan bahwa QLearning dapat menghasilkan *Mean Square Deviation* (MSD) untuk Qlearn sebesar 1630,13 lebih unggul dari pada *Basic Reinforcement Learning* yaitu sebesar 198,88 dengan waktu learning yang relatif lebih singkat dan skor yang diperoleh relatif lebih konstan dibandingkan *Basic Reinforcement Learning*[6].

Berdasarkan beberapa penelitian tersebut, maka dapat disimpulkan bahwa metode *Reinforcement Learning* yang menggunakan QLearning dapat digunakan untuk memaksimalkan *reward* yang diterima dengan melakukan keputusan yang tepat. Pada penelitian ini akan dibangun sebuah sistem simulasi agen pada *game* Auto chess menggunakan *Reinforcement Learning* dan algoritma QLearning, yang diharapkan dapat berfungsi untuk menggantikan peran manusia dalam memainkan pemain pada sebuah *game* khususnya pada *game* strategi Auto Chess[7].

### **Rumusan Masalah**

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka yang menjadi pokok permasalahan adalah bagaimana cara mengimplementasikan *Reinforcement Learning* yang menggunakan algoritma Qlearning dalam *game* Auto Chess untuk dapat memenangkan suatu pertandingan.

### **Maksud dan Tujuan**

Maksud dari penelitian ini adalah untuk membangun sebuah sistem simulasi implementasi *Reinforcement Learning* yang menggunakan QLearning pada *game* Auto Chess.

Sedangkan tujuan dari penelitian ini adalah mengetahui performa *Reinforcement Learning* yang menggunakan algoritma Qlearn dalam memainkan game Auto Chess.

### **Batasan Masalah**

Agar penelitian yang dilakukan lebih terarah sesuai dengan tujuan yang ingin dicapai, maka batasan masalah dalam penelitian ini yaitu sebagai berikut.

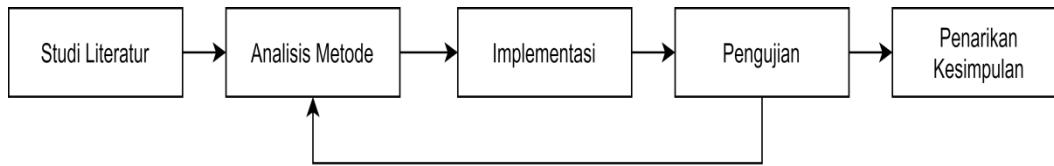
1. *Game* Auto Chess yang digunakan adalah *prototype game* Auto Chess untuk desktop.
2. Mode pertandingan pada *game* auto chess yang digunakan yaitu mode pemula.
3. Algoritma yang digunakan sebagai agen lawan yaitu algoritma *Basic Reinforcement Learning*.
4. Masukan

Data masukan dan data uji yang digunakan adalah sebagai berikut.

- a) Data input latih merupakan data hasil konversi kondisi *state* pada *game* yang diambil secara *real time*.
  - b) Data input latih dibagi menjadi 3 sesuai dengan 3 fase pada permainan yaitu data input fase pembelian, data input fase i, data input fase *upgrade*.
5. Proses
    - a) Banyaknya pertandingan untuk proses pelatihan adalah maksimal 30 pertandingan.

### **Metode Penelitian**

Metode penelitian yang digunakan dalam penelitian ini adalah metode kuantitatif [8]. Metode ini digunakan karena data yang digunakan dalam penelitian ini berbentuk angka dan bersifat fakta serta bisa diukur secara akurat dengan alat yang objektif.



**Gambar 1. 1** Metode Penelitian

### 1.5.1 Studi Literatur

Studi ini dilakukan dengan cara mempelajari, meneliti dan menelaah berbagai literatur-literatur yang bersumber dari buku-buku, teks, jurnal ilmiah, situs-situs di internet, dan bacaan-bacaan yang terkait dengan topik *action game*, *game Auto Chess*, dan metode *Reinforcement Learning* yang menggunakan QLearning.

### 1.5.2 Analisis Metode

Analisis data dan metode pada penelitian ini yaitu:

#### 1. Analisis Data Masukan

Analisis ini merupakan proses analisis data masukan berupa fase-fase dalam *game* yang akan digunakan dalam penelitian. Data masukan yang sudah ditentukan kemudian dianalisis kembali untuk menentukan fitur-fitur yang digunakan sehingga hasil akhir yang didapatkan berupa *state* untuk setiap fase yang telah ditentukan.

#### 2. Analisis Proses *Reinforcement Learning*

Analisis ini berisikan analisis terhadap metode *Reinforcement Learning* dan juga Algoritma QLearning dalam menentukan aksi untuk setiap fase yang telah ditentukan, berdasarkan data masukan (*state*).

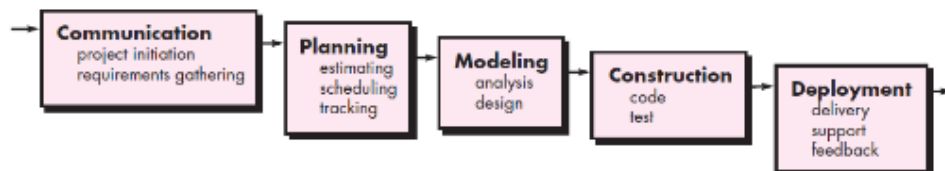
#### 3. Analisis Proses Pengujian

Analisis ini merupakan proses analisis terhadap parameter yang terdapat dalam metode *Reinforcement Learning* dan juga algoritma QLearning. Kemudian, menentukan nilai untuk parameter yang akan diuji, menganalisis proses pengujian, serta melakukan analisis terhadap setiap hasil pengujian, sehingga didapatkan kesimpulan akhir dari penelitian yang telah dilakukan.

### 1.5.3 Implementasi

Mengimplementasikan hasil analisis ke dalam program agar selanjutnya bisa masuk ketahap pengujian. Metode yang digunakan dalam pembuatan perangkat

lunak ini adalah metode *waterfall*, metode ini dipilih dikarenakan kebutuhan dan juga gambaran sistem yang akan dibangun sudah jelas sehingga kecil kemungkinan terjadinya perubahan. Metode *waterfall* meliputi beberapa proses sebagai berikut[9].



**Gambar 1. 2** Waterfall SDLC[9]

*a. Communication*

Tahapan ini merupakan tahapan awal pengumpulan kebutuhan dari perangkat lunak yang akan dibangun seperti *game strategy*, notasi perhitungan dari *Reinforcement Learning* dan Algoritma Qlearn. Pada tahap ini data yang akan digunakan merupakan data hasil konversi tiga fase dasar dalam *game* yaitu, fase pembelian bidak, fase *line up* dan fase *upgrade* pada *game* Auto Chess yang diperoleh secara langsung (*real time*), serta mengumpulkan data-data pendukung lainnya melalui sumber kepustakaan.

*b. Planning*

Tahapan ini merupakan lanjutan dari tahapan Communication, dimana setelah dilakukan pengumpulan terhadap kebutuhan perangkat lunak, maka mulai dilakukan penjadwalan terhadap pembangunan perangkat lunak.

*c. Modeling*

Tahapan ini merupakan tahapan penerjemahan kebutuhan perangkat lunak yang didapatkan ke dalam bentuk analisis seperti Use Case Diagram, Class Diagram, Sequence Diagram, dan lainnya. lalu dilanjutkan ke dalam pemodelan perangkat lunak sebelum nantinya dilakukan pembangunan perangkat lunak.

*d. Construction*

Tahapan ini merupakan tahap pembangunan untuk sistem simulasi implementasi *Reinforcement Learning* yang menggunakan algoritma Qlearn

pada *game* Auto Chess. Proses pembangunan perangkat lunak yang dilakukan dibagi menjadi dua proses yaitu:

1. Pembangunan *prototype game* Auto Chess.
2. Pembangunan sistem simulasi (sistem utama).
3. Implementasi metode pada *prototype game* Auto Chess.

e. *Deployment*

Tahapan ini merupakan tahap akhir dalam pembangunan sistem dimana sistem ini dapat melakukan simulasi implementasi *Reinforcement Learning* yang menggunakan algoritma Qlearn dalam *game* Auto Chess.

#### **1.5.4 Pengujian**

Perangkat lunak yang telah dibuat dan diuji secara menyeluruh kemudian masuk ke tahap pengujian metode. Pengujian metode ini dilakukan dengan menguji hasil pelatihan serta menguji parameter algoritma seperti parameter *learning rate*, dan juga *discount rate*, serta melakukan pengujian penggunaan *resource* yaitu penggunaan *memory* baik saat proses pelatihan maupun saat proses pengujian.

#### **1.5.5 Penarikan Kesimpulan**

Setelah tahap pengujian selesai maka data yang dihasilkan dari proses pengujian dianalisa sehingga bisa ditarik kesimpulan berdasarkan tujuan penelitian yang telah ditentukan sebelumnya.

### **1.6 Sistematika Penulisan**

Sistematika penulisan disusun untuk memberikan gambaran secara umum mengenai permasalahan dan pemecahannya. Sistematika penulisan skripsi ini adalah sebagai berikut.

## **BAB 1 PENDAHULUAN**

Bab ini membahas mengenai latar belakang, rumusan masalah, maksud dan tujuan, batasan masalah, metode penelitian, serta sistematika penulisan untuk menjelaskan pokok – pokok pembahasannya.

## **BAB 2 LANDASAN TEORI**

Pada bab ini akan menjelaskan mengenai landasan teori dan teori yang digunakan dalam penelitian ini seperti, pengenalan *Vidio Game*, pengenalan

*Reinforcement Learning*, teori analisis sistem, dan bahasa pemrograman yang digunakan.

### **BAB 3 ANALISIS DAN PERANCANGAN SISTEM**

Bab ini berisi pemaparan tentang metode *Reinforcement Learning*, analisis aplikasi, analisis kebutuhan fungsional, analisis kebutuhan *non-fungsional*, analisis basis data, dan perancangan aplikasi.

### **BAB 4 IMPLEMENTASI DAN PENGUJIAN**

Berisi pembahasan implementasi yang dilakukan, mulai dari bahasa pemrograman, hardware dan software yang digunakan, serta hasil pengujian. Pengujian yang dilakukan antara lain pengujian parameter algoritma kondisi henti, *learning rate*, *discount rate*, serta pengujian penggunaan *resource* diantaranya waktu pembelajaran dan bobot yang disimpan.

### **BAB 5 KESIMPULAN DAN SARAN**

Berisi kesimpulan dari hasil pengujian yang dilakukan, apakah tujuan awal dalam penelitian telah tercapai atau tidak, serta berisi saran untuk penelitian selanjutnya.

