



### **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

Catur adalah permainan dengan struktur aturan yang kompleks dan tingkat individual permainannya tergantung pada representasi sistem peraturannya, serta antisipasi dan kreatifitas yang digunakan selama permainan (Scholz dkk., 2008). Selain itu, Scholz dkk. (2008) menganggap catur sebagai model untuk proses dan kemampuan kognitif seperti persepsi, manajemen informasi, atensi, memori, pemikiran logis, dan pemecahan masalah. Menurut Setiadi (2012), salah satu jenis *two player strategy board game* ini pada umumnya terdiri dari delapan macam bidak dengan ukuran papan delapan kali delapan persegi dengan masing-masing pemain memiliki 16 bidak yang dimainkan.

David dkk. (2014) mengatakan, bahwa kecerdasan buatan pada permainan catur merupakan salah satu bidang yang paling sulit dalam menghasilkan performa yang sebanding dengan performa yang dimiliki oleh manusia. Menurut Heath dan Allum (1997), Minimax merupakan algoritma pertama yang digunakan pada permainan catur. Kemudian pada tahun 1956, McCarthy melakukan modifikasi pada Minimax dengan cara memformulasikan Alpha-Beta Pruning. Setelah itu, algoritma Alpha-Beta Pruning dianalisis dan terbukti bahwa algoritma ini mengembalikan pergerakan dengan nilai yang sama seperti yang dihasilkan oleh algoritma Minimax.

Menurut Marckel (2017), algoritma Alpha-Beta Pruning adalah *adversarial search algorithm* yang menerapkan pemangkasan cabang pada struktur data *tree* sebagai upaya untuk mempercepat proses pencarian pada algoritma Minimax.

Selain itu, Alpha-Beta Pruning juga memungkinkan kecerdasan buatan untuk mencari secara mendalam semua opsi yang berpotensi untuk bergerak dan memilih yang terbaik. Menurut Xia (2018), pemangkasan ini tidak memengaruhi hasil akhir di *root tree*, Hanya saja, algoritma ini dapat meningkatkan pemecahan kedalaman hingga dua kali lipat dibandingkan dengan algoritma Minimax.

Penggunaan algoritma Alpha-Beta Pruning itu sendiri telah terbukti dapat mengalahkan *grandmaster* catur, yaitu AI “Deep Blue” menang melawan Garry Kasparov (Hsu, 1999) dan AI “Hydra” menang melawan Michael Adams (Ertel, 2017). Ertel (2017) mengatakan, bahwa Deep Blue saat itu dapat melakukan pencarian hingga *depth* 12 dan Hydra dapat melakukan pencarian hingga *depth* 18.

Kelemahan dari metode Alpha-Beta Pruning menurut Schaeffer dan Plaat (1996) adalah sulitnya untuk mengukur dampak kedalaman pencarian variabel. Hal ini disebabkan oleh pencarian yang seharusnya bukan hanya mempertimbangkan ukuran *tree* saja, tetapi juga kualitas jawaban yang diberikan. Selain itu menurut Carolus (2006), kompleksitas algoritma Alpha-Beta Pruning bergantung pada urutan dari pergerakannya, yaitu pemangkasan terjadi ketika pergerakan yang baik diperiksa terlebih dahulu. Sehingga, kasus terburuknya dapat menghasilkan kompleksitas yang sama dengan algoritma Minimax.

Schaeffer (1989) menyebutkan beberapa optimasi untuk algoritma Alpha-Beta Pruning, diantaranya adalah Iterative Deepening, Transposition Tables, Refutation Tables, Minimal Window Search, Aspiration Search, dan Killer Heuristic. Menurutnya, optimasi Alpha-Beta Pruning yang cukup populer adalah Killer Heuristic. Selain itu, Polk dan Oommen (2016) mengatakan bahwa optimasi Alpha-Beta Pruning dengan cara mengurutkan langkah (*move ordering*) yang populer

adalah History Heuristic dan Killer Heuristic. Merekapun telah membuktikan bahwa Killer Heuristic lebih baik dibandingkan History Heuristic.

Namun, Schaeffer (1989) mengatakan bahwa belum ada yang benar-benar mengetahui seberapa efektif optimasi yang dihasilkan oleh metode optimasi Killer Heuristic dibandingkan dengan Alpha-Beta Pruning tanpa optimasi, khususnya efektifitas pada jumlah *node* yang dibuat. Selain itu, AI Deep Blue dan Hydra menggunakan *hardware* yang sangat mahal yang mengakibatkan kedua AI tersebut dapat menelusuri dua ratus juta *node* hanya dalam waktu satu detik (Ertel, 2017).

Selain dari penelitian yang telah dibahas diatas, tidak ditemukan penelitian lain terkait pengukuran *resource* dan performa yang dihasilkan oleh algoritma Alpha-Beta Pruning dibandingkan dengan Alpha-Beta Pruning dengan optimasi Killer Heuristic. Oleh sebab itu, belum diketahui dengan pasti performa algoritma Alpha-Beta Pruning tanpa optimasi dibandingkan performa algoritma Alpha-Beta Pruning yang dioptimasi oleh Killer Heuristic terutama pada jumlah *node* dan waktunya dengan menggunakan *resource* komputer yang terbatas.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah dibahas, terdapat dua masalah yang dirumuskan dalam penelitian ini, yaitu:

- a. Bagaimana mengimplementasi algoritma Alpha-Beta Pruning dengan optimasi Killer Heuristic pada kecerdasan buatan permainan catur?
- b. Bagaimana perbandingan ukuran pohon yang terbentuk dan waktu yang dibutuhkan oleh algoritma Alpha-Beta Pruning tanpa optimasi dan algoritma

Alpha-Beta Pruning yang dioptimasi oleh Killer Heuristic dalam membuat suatu keputusan langkah pada permainan catur?

### 1.3 Batasan Masalah

Berdasarkan rumusan masalah, batasan masalah pada penelitian ini adalah sebagai berikut.

- a. Regulasi permainan catur yang digunakan mengikuti regulasi catur internasional yang berasal dari *The Fédération Internationale des Échecs* atau *World Chess Federation (FIDE)*.
- b. Tidak ada limitasi waktu yang diberikan pada setiap entitas dalam melakukan setiap langkah maupun keseluruhan permainannya.
- c. Nilai evaluasi untuk setiap algoritma diambil berdasarkan jumlah nilai *material* dan nilai posisi setiap bidak pada papan.
- d. Jumlah maksimum *killer move* yang tersimpan adalah dua *move*.
- e. Perbandingan yang diukur adalah nilai rata-rata yang berasal dari ukuran *tree* yang terbentuk dan waktu yang dibutuhkan algoritma dari uji coba yang dilakukan.
- f. Waktu yang digunakan algoritma dihitung berdasarkan lamanya fungsi algoritma berjalan pada setiap gilirannya dalam satuan detik.
- g. Algoritma yang digunakan hanya dijalankan pada *single thread*.
- h. Tingkat kedalaman algoritma yang digunakan pada penelitian ini dimulai dari *depth* dua hingga lima.

#### **1.4 Tujuan Penelitian**

Berdasarkan rumusan masalah, tujuan dari dilakukannya penelitian ini adalah sebagai berikut.

- a. Mengimplementasi algoritma Alpha-Beta Pruning dengan optimasi Killer Heuristic pada kecerdasan buatan agar mampu menghasilkan keputusan langkah pada permainan catur.
- b. Mengetahui perbandingan antara algoritma Alpha-Beta Pruning tanpa optimasi dengan algoritma Alpha-Beta Pruning yang dioptimasi oleh Killer Heuristic yang diukur melalui ukuran pohon yang terbentuk dan waktu yang dibutuhkan kedua algoritma tersebut dalam membuat keputusan langkah permainan catur.

#### **1.5 Manfaat Penelitian**

Penelitian ini dapat memiliki beberapa manfaat, diantaranya adalah sebagai berikut.

- a. Dapat digunakan sebagai pengembangan atau penambahan wawasan kecerdasan buatan di bidang catur.
- b. Dapat menambah bahan pertimbangan para pengembang permainan catur dalam menentukan algoritma apa yang sesuai untuk digunakan pada aplikasinya.
- c. Dapat menambah bahan pertimbangan para pengembang permainan catur dalam membuat dan menjalankan kecerdasan buatan catur pada komputer dengan *resource* komputasi yang terbatas.

## 1.6 Sistematika Penulisan

Sistematika penulisan laporan Skripsi ini terdiri dari lima bab, yaitu sebagai berikut.

### 1. BAB I PENDAHULUAN

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

### 2. BAB II LANDASAN TEORI

Bab ini menjelaskan landasan teori dari penelitian yang dilakukan, seperti catur, kecerdasan buatan, Minimax, Alpha-Beta Pruning, dan Killer Heuristic.

### 3. BAB III METODOLOGI PENELITIAN DAN PERANCANGAN

Bab ini menjelaskan metodologi penelitian yang digunakan serta perancangannya. Adapun perancangan yang dimaksud yaitu *flowchart* algoritma Alpha-Beta Pruning, *flowchart* algoritma Alpha-Beta Pruning dengan optimasi Killer Heuristic, *flowchart* alur aplikasi, dan rancangan tampilan antarmuka aplikasi yang digunakan untuk melakukan penelitian.

### 4. BAB IV IMPLEMENTASI DAN ANALISIS

Bab ini membahas tentang proses implementasi algoritma Alpha-Beta Pruning dengan atau tanpa optimasi Killer Heuristic pada aplikasi yang dibuat serta hasil analisis kedua algoritma.

### 5. BAB V SIMPULAN DAN SARAN

Bab ini terdiri atas kesimpulan dari hasil pengujian kedua algoritma dan saran untuk pengembangan penelitian selanjutnya.