

PENERAPAN ALGORITMA MINIMAX DENGAN OPTIMASI MEMORY ENHANCED TEST DRIVER WITH VALUE F PADA GAME CATUR

¹Safrizal, ²Deddy Rahmad Setiawan

¹safrizal.st.mm@gmail.com, ²deddyrah@gmail.com

Program studi teknik informatika Fakultas Teknik

Univesitas Satya Negara Indonesia

ABSTRAK

Algoritma Minimax adalah salah satu algoritma yang di gunakan untuk membuat sebuah kecerdasan buatan dalam aplikasi permainan strategi seperti catur karena algoritma minimax bisa menganalisa semua kemungkinan posisi bidak untuk menghasilkan keputusan yang terbaik. Penerapan Algoritma minimax dilakukan optimasi dengan algoritma MTD(f). Algoritma MTD (f) adalah sebuah algoritma optimasi Minimax baru yang lebih sederhana dan lebih optimal daripada beberapa pendahulunya. Proses Optimasi yang dilakukan pada Algoritma pada MTD(f) adalah dengan melakukan efisiensi berupa pemangkasan kemungkinan langkah langkah yang digambarkan dalam bentuk pohon algoritma MTD (f). Kecerdasan buatan (AI) memungkinkan mesin untuk belajar dari pengalaman, menyesuaikan input-input baru dan melaksanakan tugas seperti manusia. Sebagian besar contoh AI yang Anda dengar dewasa ini – mulai dari komputer yang bermain catur hingga mobil yang mengendarai sendiri – sangat mengandalkan pembelajaran mendalam dan pemrosesan bahasa alamiah. Dengan menggunakan teknologi ini, komputer dapat dilatih untuk menyelesaikan tugas-tugas tertentu dengan memproses sejumlah besar data dan mengenali pola dalam data. Pembuatan game catur ini dirancang menggunakan story board dengan Konsep GDLC (Game Development Life Cycle) yaitu pembuatan game yang dimulai dari konsep ide sampai menjadi game dalam bentuk aplikasi . Tahap tahap pada Game Development Life Cycle adalah sebagai berikut: Initiation, Pre- production, Testing, Beta, Release

Kata Kunci : Minimax, Catur, Artificial Intelegence, MTD(F).

ABSTRACT

The Minimax algorithm is an algorithm that is used to create an artificial intelligence in a strategy game application such as chess because the Minimax algorithm can analyze all possible pawn positions to produce the best decision. The application of the minimax algorithm is carried out with the optimization of the MTD (f) algorithm. The MTD (f) algorithm is a new Minimax optimization algorithm that is simpler and more optimal than some of its predecessors. The Optimization Process carried out on the Algorithm on MTD (f) is by doing efficiency in the form of trimming the possible steps described in the form of the MTD (f) algorithm tree. Artificial intelligence (AI) allows machines to learn from experience, adjust new inputs and carry out tasks like humans. Most AI examples that you hear today - from computers playing chess to self-driving cars - rely heavily on deep learning and natural language processing. By using this technology, computers can be trained to complete certain tasks by processing large amounts of data and recognizing patterns in the data. Making this chess game is designed using story boards with the Concept of GDLC (Game Development Life Cycle) which is making games that start from the concept of ideas until it becomes a game in the form of an application. Stage on Game Dev

Keywords: Minimax, Chess, Artificial Intelligence, MTD (F).

Pendahuluan

Latar Belakang

Semakin berkembangnya teknologi orang-orang saat ini lebih banyak menghabiskan waktu untuk bermain permainan pada alat elektroniknya seperti *smartphone* dan komputer yang dimainkan secara *personal* atau seorang diri saja. *Game* adalah sesuatu yang dapat dimainkan dengan aturan tertentu sehingga ada yang menang dan ada yang kalah.

Banyak jenis permainan yang ada di komputer, salah satunya adalah strategi yang memerlukan keahlian berpikir seperti catur. Catur adalah permainan papan yang di mainkan oleh dua orang pemain dengan tujuan mengalahkan lawan.

Kepopuleran game catur dapat dilihat dari banyaknya kejuaran yang diselenggarakan oleh berbagai negara di dunia. Hal ini yang mendorong banyak pemain catur mulai mendalami permainan catur secara profesional. Untuk menjadi pemain profesional dibutuhkan latihan yang intens dan optimal. Sedangkan dalam permainan catur dibutuhkan 2 pemain.

Sulitnya mencari pemain lain sebagai lawan yang sesuai untuk berlatih tanding menjadi kendala bagi seseorang yang ingin menjadi pemain catur profesional. Untuk itu perlu adanya suatu sistem yang dapat menjadi lawan tanding yang sepadan yang dapat dimainkan kapan saja dan dimana saja. maka di butuhkan sebuah kecerdasan buatan yang dapat menjadi lawan bermain dalam permainan catur.

Artificial Intelligence (AI) atau kecerdasan buatan biasanya digunakan sebagai teknik untuk menggerakkan komputer sebagai lawan bermain dalam aplikasi permainan.

Algoritma yang dapat digunakan untuk membuat sebuah kecerdasan buatan dalam permainan catur adalah Algoritma *Minimax* karena algoritma ini dapat mencari seluruh kemungkinan langkah yang ada bahkan dapat dilakukan sampai akhir permainan, dioptimasi dengan algoritma MTD(f) yang membatasi kemungkinan langkah yang di hasilkan Algoritma *Minimax* agar pencarian langkah terbaik dapat di cari dengan lebih cepat.

Berdasarkan permasalahan di atas peneliti berinisiatif membuat penerapan Algoritma *Minimax* dan MTD(f) pada game catur. Penelitian ini akan menghasilkan sebuah aplikasi AI game berbasis *platform* unity.

Rumusan Masalah

Dalam penelitian ini dapat dirumuskan permasalahan, yaitu “ Bagaimana Menerapkan Algoritma *Minimax* dan MTD Pada Permainan catur di komputer?”

Batasan Masalah

Perlu adanya batasan pada pembahasan sehingga masalah tidak melebar. Fokus penelitian ini adalah sebagai berikut:

1. Permainan ini dilakukan oleh satu orang pemain melawan komputer.
2. Permainan ini offline dan tidak bisa bermain secara multiplayer.
3. Algoritma yang digunakan adalah algoritma *Minimax* dan MTD(f).

Tujuan

Tujuan dari penelitian ini untuk membuat game catur menggunakan Algoritma *Minimax* dengan optimasi algoritma MTD(f).

Manfaat

Manfaat dari penelitian game catur ini antara lain:

1. Bisa menjadi alat untuk berlatih catur.
2. Permainan catur dapat dimainkan kapan saja.
3. Game catur dapat dimainkan oleh siapa saja.

Landasan Teori

Definisi Permainan (Game)

Permainan (*games*) adalah setiap kontes antara pemain yang berinteraksi satu sama lain dengan mengikuti aturan-aturan tertentu untuk mencapai tujuan tertentu pula (Sadiman, 1993:75). Jadi permainan adalah cara bermain dengan mengikuti aturan-aturan tertentu yang dapat dilakukan secara individu maupun berkelompok untuk mencapai kemenangan.

GDLC (Game Development Life Cycle)

GDLC merupakan sebuah metode yang menangani pengembangan game dimulai dari titik awal hingga paling akhir. Dimulai dari tahap pembuatan ide dan konsep mengenai game yang akan dibuat. tahap tahapnya sebagai berikut : *Initiation, Pre- production, Testing, Beta, Release*

Catur

Catur adalah permainan strategi yang di mainkan oleh 2 orang pemain. Pecatur adalah orang yang memainkan catur, baik dalam pertandingan satu lawan satu maupun satu melawan banyak orang (dalam keadaan informal). Sebelum bertanding, pecatur memilih biji catur yang akan ia mainkan. Terdapat dua warna yang membedakan bidak atau biji catur, yaitu hitam dan putih. Pemegang buah putih memulai langkah pertama, yang selanjutnya diikuti oleh pemegang buah hitam secara bergantian sampai permainan selesai.

Unity 3D

Unity 3D adalah game engine merupakan sebuah software pengolah gambar, grafik, suara, input, dan lain-lain yang ditujukan untuk membuat suatu game. Kelebihan dari game engine ini adalah bisa membuat game berbasis 3D maupun 2D, dan sangat mudah digunakan. Unity merupakan game engine yang ber-multiplatform. Unity mampu di publish menjadi Standalone (.exe), berbasis web, berbasis web, Android, iOS Iphone, XBOX, dan PS3. Walau bisa dipublish ke berbagai platform, Unity perlu lisensi untuk dapat dipublish ke platform tertentu. Tetapi Unity menyediakan untuk free user dan bisa di publish dalam bentuk Standalone (.exe) dan web

Algoritma

Algoritma Minimax

Algoritma minimax merupakan salah satu algoritma yang digunakan pada permainan dua player yang memiliki AI games seperti catur. Pada algoritma minimax, pengecekan akan dilakukan untuk mencari seluruh kemungkinan yang ada bahkan dapat dilakukan sampai akhir permainan.

Prosedur minimax:

1. Tandai masing-masing level dari ruang pencarian sesuai dengan langkahnya pada level itu.
2. Mulai pada node daun (node paling bawah), dengan menggunakan fungsi nilai, berikan nilai pada masing-masing node.
3. Arah menjalar ke atas : jika node orang tua adalah MAX, pilihlah nilai terbesar yang terdapat pada node anak dan berikan nilai pada MAX (node orang tua)
4. Arah menjalar ke atas : jika node orangtua adalah MIN, pilihlah nilai terkecil yang terdapat pada node anak dan berikan nilai MIN (node orang tua).

Algoritma MTD (f)

Algoritma MTD (f) adalah sebuah algoritma optimasi Minimax baru yang lebih sederhana dan lebih optimal daripada beberapa pendahulunya. Nama dari algoritma ini adalah kependekan dari MTD(f), yang disingkat dari Memory-enhanced Test Driver with node n and value f (Setiadi, 2012).

Prosedur Algoritma MTD(f)

1. Memanggil fungsi Alpha Beta with memory berkali kali dengan metode Zero window pencarian Alpha
2. Pemanggilan Alpha Beta With memory mengembalikan batas dari nilai evaluasi minimax
3. Batas dari nilai itu kemudian disimpan dalam *Upperbound (batas atas)* dan *Lowerbound (batas bawah)*
4. Membentuk sebuah interval yang melingkupi nilai minimax yang sebenarnya pada pencarian dengan kedalaman tertentu
5. Positif dan negatif tak hingga adalah kependekan dari nilai diluar interval pada daun pohon
6. Ketika batas atas dan batas bawahnya bernilai sama atau batas bawah telah melampaui nilai batas atas maka nilai minimax ditemukan
7. Efisiensi dari MTD(f) berasal dari Pencarian alpha Beta dengan Zero window, dan menggunakan sebuah nilai batas yang baik (variabel beta) untuk melakukan pencarian zero window tsb
8. Alpha beta dipanggil dengan pencarian wide window seperti alpha beta, (root, infinity+infinity, depth yang memastikan nilai kembaliannya berada dalam interval alpha beta

9. Sedangkan dalam algoritma MTD (f) digunakan pencarian zerowindow, jadi untuk setiap pemanggilan Alpha beta akan mengembalikan batas bawah dan batas atas dari nilai berturut turut
10. Pencarian dengan zero window memberikan lebih banyak jalan pintas , tetapi lebih sedikit informasi hanya batas dan minimax saja
11. Untuk menyisiati itu (MTD) (f) perlu memanggil Alpha beta beberapa kali, untuk mendekati nilai itu. Imbas dari pemanggilan Alpha beta secara berulang ulang dapat dihilangkan dengan menggunakan versi dari Alpha beta yang menyimpan nilai simpul dalam memori.
12. Agar algoritma MTD (f) dapat berjalan dengan baik, maka diperlukan “Tebakan pertama” sebagai pengarah untuk menemukan nilai minimax yang baik.
13. Jika kita memberikan tebakan nilai minimax pada MTD (f) dia hanya akan melakukan dua pengulangan , yang satu untuk menentukan batas atas dari nilai X , dan yang satunya lagi untuk menemukan batas bawah dari nilai tersebut.

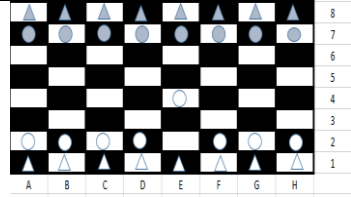
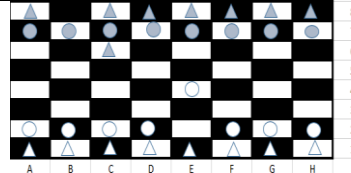
Tabel 1. Nilai Catur

No	Nama	Memakan	Di makan
1	Pion	1	-1
2	Gajah	2	-2
3	Kuda	3	-3
4	Benteng	4	-4
5	Mentri	5	-5
6	Raja	10	-10

Keterangan :

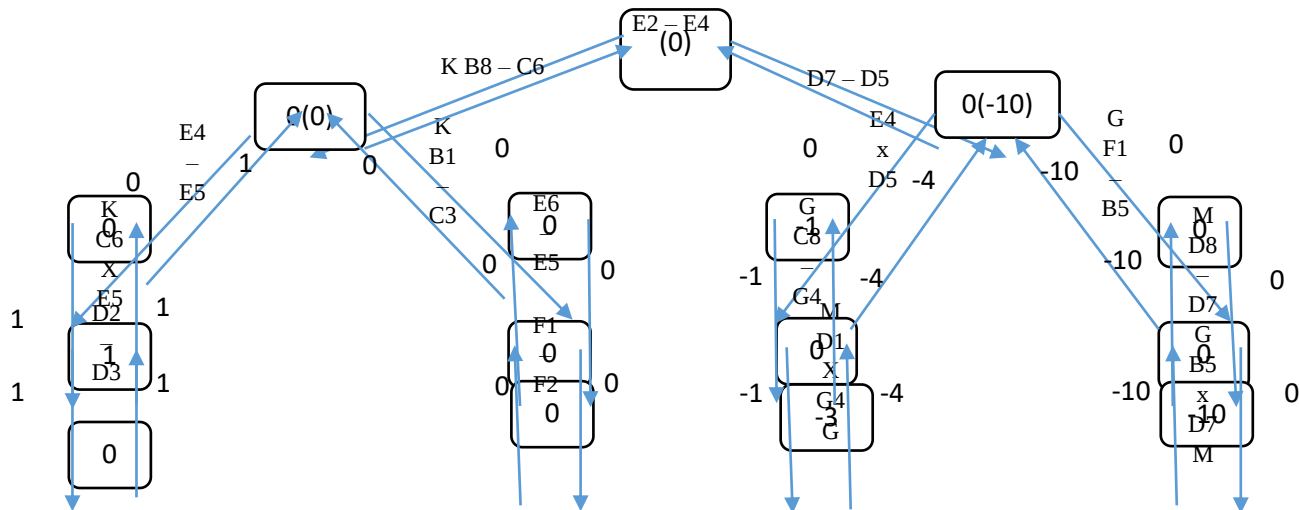
1. **A1 – B1** = Langkah pada notasi catur
2. **1** = Nilai Langkah
3.  = Nilai Langkah yang di akumulasikan
4.  = Nilai kemudian di naikan
5. **2/1** = Pada kedalaman ke 1 memiliki 2 nilai yaitu nilai langkah dan nilai
6. **(1)** = nilai pada kedalaman 0

Metodologi Penelitian**A. Storyboard Game***Tabel 1 Storyboard Game*

1		Permainan dimulai, yaitu kita menjalankan pion E2 – E4. untuk menjawab langkah ini komputer sudah menggunakan algortima minimax yang di optimasi algoritma mtd
2		Komputer menjawab dengan mengerakan pion K B8 - C6. Karena akan di jawab oleh algoritma minimax dengan optimasi mtd yang akan di jelaskan di bawah ini.

3		Lalu putih menjalankan M D1 – H5 dan hitam menjalankan M D7 – F5. Sampai di tentukan siapa yang menang dan yang kalah. Permainan selesai.
---	--	---

Penerapan pohon Algoritma optimasi MTD dalam bentuk penerapan langkah catur



Analisa Kebutuhan

1. Perangkat Keras

Perangkat keras yang digunakan untuk membuat game yaitu *laptop* dengan spesifikasi :

- Prosesor intel core i5 – 6200U @2,8GHz
- VGA Nvidia Geforce 920M
- Ram 4 GB
- Hardisk 500 GB

2. Perangkat Lunak

Perangkat lunak yang digunakan yaitu :

- Windows 10
- Unity3D
- Notepad

Metode Pengumpulan Data

Metode pengumpulan data merupakan teknik atau cara yang dilakukan untuk mengumpulkan data, adapun metode pengumpulan data yang digunakan dalam penelitian ini, antara lain.

1. Studi Literatur

Studi literatur yang digunakan yaitu buku-buku, jurnal, prosiding dan internet yang menyajikan informasi tentang, algoritma minimax dan unity.

2. Observasi

Observasi di lakukan dengan mengamati langsung game catur yang ada.

Analisa game

Game yang dibangun adalah game 1 pemain dengan komputer sebagai musuhnya. Game ini bergenre strategi dan tingkat kesulitannya adalah sulit. Karena algoritma minimax akan selalu bergerak berdasarkan tingginya keuntungan dan tingginya kerugian musuh.

Analisa Hasil Dan Pembahasan

A. Proses pembuatan

B. Game

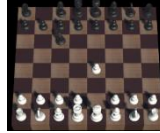
- Langkah Pertama E2 – E4



Gambar 1. Langkah Pertama E2 – E4

Player mengerjakan pion E2 – E4 menghasilkan pohon pencarian yang terdapat pada gambar Langkah Pertama.

2. Dijawab B8 – C6



Gambar 2. Langkah B8 – C6

Untuk menjawab Langkah E2 – E4 dengan K B8 – C6 dan sampai permainan berakhir yang dapat ditentukan kalah dan menang

Kesimpulan Dan Saran

Kesimpulan

Berdasarkan Hasil dari pengujian sistem dan analisis hasil dapat diambil kesimpulan sebagai berikut :

1. Algoritma Minimax dan Algoritma Memory-Enhanced Test Driver with Value f dapat diterapkan pada permainan catur dan berjalan sesuai dengan logika algoritma.
2. Algoritma Memory-Enhanced Test Driver with Value f membuat komputer melakukan pencarian langkah terbaik dengan cepat.

Saran

Game ini dapat menggunakan Algoritma MTD dengan kedalaman lebih dalam tinggi lagi dari penulis buat.

Daftar Pustaka

- Andrian Chanhar, K. S. (2012). Penerapan Algoritma MTD(f) pada Permainan TIC TAC TOE Dengan Variasi Area. *888IJCCS, Vol.x, No.x, Julyxxxx, pp. 1~5*, 1 - 11.
- Andrian Chanhar, K. S. (2015). Penerapan Algoritma MTD Pada Permainan TIC TAC TOE dengan Variasi Area.
- Ardiansa*, S. A. (2012). Perbandingan Performa Algoritma Minimax dan Negascout pada Permainan Checkers Berbasis Android. 1-9.
- Christian Hadi, O. S. (2017). Penerapan Algoritma Minimax Dengan Memory Enhanced Test Driver Value F pada permainan Congklak.
- Dahwila Syapnika 1, E. R. (2015). PENERAPAN ALGORITMA MINIMAX PADA PERMAINAN CHECKERS . *Jurnal Riset Komputer (JURIKOM)*, Vol. 2 No. 6, 28 - 32.
- Erick Sanjaya, T. W. (2015). Penerapan Algoritma Minimax dan memory Enhanced test Driver with value f pada permainan Checkers.
- Ginonggom1, M. A. (t.thn.). ANALISIS PENGGUNAAN ALGORITMA GREEDY PADA PERMAINAN CATUR . 1-14.
- Ilman, A. (t.thn.). PENERAPAN ALGORITMA MINIMAX DENGAN OPTIMASI MTD(f) PADA PERMAINAN CATUR.
- Panji Novantara, N. S. (2017). Rancang Bangun Aplikasi Catur Menggunakan Algoritma Minimax Dengan Optimasi Alpha-Beta Pruning . *Jurnal Cloud Information, Volume 3 Nomor 2.*, 1-15.
- Pratama, W. (2014). GAME ADVANTURE MISTERI KOTAK PANDORA. *Jurnal Telematika Vol. 7 No.2*, 13 - 31.

- Rendy Adiwikarta¹), H. B. (2017). Pengembangan Permainan Video Endless Running Berbasis Android Menggunakan Framework Game Development Life Cycle . *Kalbiscientia*, Volume 4 Nomor 2 , 142-148.
- Setiadi, D. R. (Mei 2012). IMPLEMENTASI ALGORITMA MINIMAX UNTUK ARTIFICIAL INTELEGENCE PADA PERMAINAN CATUR SEDERHANA. *Techno.COM*, Vol.11, No. 2, 97 - 106.
- Aditama, Bidak. 16 Januari 2012. Manfaat Bermain Catur bagi Anak-Anak, (Online), (<http://bidakaditama.com/2012/01/manfaat-bermain-catur-bagi-anak-anak/> , diakses 10 desember 2012)
- Bowler, Dermot. 2007. *Autism Spectrum Disorder*. Chichester: John Wiley & Sons Ltd.
- Budiyanto, dkk, Tanpa Tahun. *Modul Training of Trainer Pendidikan inklusif*. ____: Departemen Pendidikan Nasional
- Christian, Julie. 2006. *Autism Spectrum Disorder Handbook*. South Dakota: Sanford School of Medicine
- Danuatmaja, Bonny. 2003. *Terapi Anak Autis di Rumah*. ____: Puspa Indah
- Ghanizadeh, Ahmad. 2010. Clinical Approach to Motor Stereotypies in Autistic Children. *Iran J Pediatr. (online)*, vol. 20, No.2, (<http://www.ijp.tums.ac.ir>, diakses 12 maret 2013).
- Handojo. 2006. *Autisma*. Jakarta: PT Bhuana Ilmu Populer
- Hartanti, Sasminta C.Y. dkk. 2012. *Permainan Kecil*. Malang: Wineka Media
- Harun, Undi. 1985. *Seri Teori Bermain Catur*. Klaten: PT. Intan