

# Permainan Papan Strategi Menggunakan Algoritma Minimax

**Sandy Kosasi**

STMIK Pontianak

Jalan Merdeka No. 372 Pontianak

e-mail: [sandykosasi@yahoo.co.id](mailto:sandykosasi@yahoo.co.id)&[sandykosasi@stmikpontianak.ac.id](mailto:sandykosasi@stmikpontianak.ac.id)

## Abstrak

Algoritma minimax merupakan algoritma yang digunakan untuk menentukan pilihan dengan tujuan memperkecil kemungkinan kehilangan nilai maksimal komputer. Prinsip dasar pada algoritma minimax ini adalah jalur yang akan dipilih merupakan jalur maksimum (max node) yang akan menghasilkan nilai maksimum di jalur tersebut, dan saat lawan yang akan bermain akan meminimalkan (min node) nilai komputer. Tujuannya untuk memaksimalkan kemungkinan nilai paling rendah yang akan diperoleh komputer menggunakan teknik depth-first search. Melalui teknik ini penelusuran pohon permainan akan dilakukan hingga Depth-First Search menemukan goal state atau telah selesai menelusuri keseluruhan pohon permainan. Hasil penelitian menunjukkan algoritma minimax ini akan berganti giliran memainkan giliran pemain dan kemudian memilih nilai yang paling kecil (min) yang merupakan gerakan terbaik yang dilakukan pemain. Hasil pengujian memperlihatkan algoritma minimax mampu menganalisis segala kemungkinan posisi permainan untuk menghasilkan keputusan terbaik. Ketika algoritma minimax digunakan untuk memroses data dengan ukuran masukan yang besar, karena proses untuk membangun pohon pencarian dengan algoritma ini memiliki kompleksitas algoritma eksponensial.

**Kata kunci:** Permainan Papan Strategi, Algoritma Minimax, Depth-First Search.

## 1. Pendahuluan

Permainan papan (*board game*) dapat berupa permainan catur, othello, igo dan lain-lain. Permainan papan ada yang murni berbasis strategi, kesempatan ataupun gabungan dari kedua hal tersebut [1]. Jenis-jenis permainan papan memiliki berbagai ukuran papan dan jumlah pemain yang bervariasi. Permainan papan biasanya dapat dimainkan seorang diri (*single player*) maupun dimainkan berdua atau lebih (*multiplayer*). Pemain yang memainkan permainan papan harus menyusun siasat dan rencana yang tepat sebelum menggerakkan pion (bidak) dalam mengalahkan lawannya. Kekeliruan menyusun rencana strategi dalam menentukan langkah permainan dapat mengalami kekalahan [2]. Memilih langkah yang tepat dalam menyusun papan strategi dapat menggunakan algoritma minimax agar dapat menghindari kekalahan [3]. Algoritma minimax bekerja dengan cara melakukan ekspansi untuk seluruh kemungkinan langkah yang ada, menghitung bobot setiap langkah, kemudian memilih langkah terbaik dari seluruh langkah yang telah di-expand. Algoritma minimax merupakan algoritma yang digunakan untuk menentukan pilihan dengan tujuan memperkecil kemungkinan kehilangan nilai maksimal. Algoritma minimax sangat bagus untuk menyelesaikan permainan berbasis giliran (*turn-based*) [3,4].

Permainan papan strategi ini menyerupai permainan permainan *last man standing* atau *survival*. Pemain akan menghadapi satu lawan dan mencoba memilih lokasi platform yang aman dari serangan poin (bidak) lawan. Kondisi kemenangan ditentukan ketika salah satu pemain tidak dapat bergerak lagi karena kehabisan langkah. Permainan papan strategi dilengkapi kecerdasan buatan yang menjadi lawan bagi pemain ketika pemain bermain seorang diri. Kecerdasan buatan akan berpikir layaknya seorang manusia [5]. Kecerdasan Buatan yang digunakan dalam permainan papan strategi adalah algoritma minimax dengan teknik *depth-first search*. *Depth-First Search* akan melakukan pencarian pada pohon dengan cara menelusuri suatu jalur tertentu hingga mencapai kedalaman yang paling akhir (*Leaf nodes*) terlebih dahulu sebelum ke jalur selanjutnya [6]. *Leaf nodes* pada pohon permainan merepresentasikan keadaan akhir pada permainan, di mana permainan tersebut dimenangkan, dikalahkan ataupun seri. Penelusuran pohon permainan akan dilakukan hingga *Depth-First Search* menemukan *goal state* atau telah selesai menelusuri keseluruhan pohon permainan [3,6].

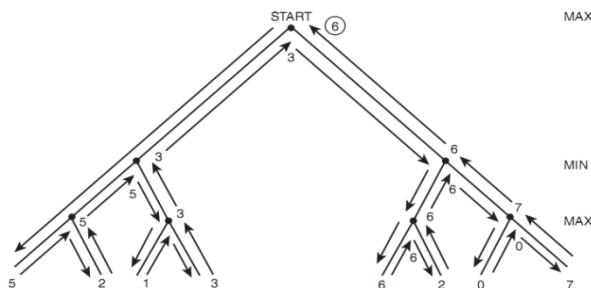
Penelitian sejenis yang sudah pernah dipublikasikan menyatakan penerapan algoritma minimax dalam sebuah permainan dapat memberikan suatu pendekatan untuk solusi-solusi dalam mencapai pilihan yang paling baik [7]. Bentuk permainan yang melibatkan 2 (dua) pemain atau lebih juga memiliki kelebihan menggunakan algoritma minimax dalam menentukan langkah yang paling tepat untuk memenangkan permainan [8,9]. Merujuk kepada penelitian

yang sudah dilakukan sebelumnya, dari segi antarmuka, permainan papan strategi memberikan tampilan yang lebih segar dan menarik dibanding tampilan yang hanya berwarna polos. Fitur yang membuat permainan papan strategi ini lebih unik dibandingkan dengan permainan papan lainnya adalah, adanya *multiplayer game mode* dimana 2 orang pemain dapat bermain bersamaan, konfigurasi permainan dapat dilakukan sebelum memulai permainan, pengarahan mengenai tata cara permainan berupa *game guide* dan fitur paling utama yang disediakan adalah level papan permainan dapat ditingkatkan hingga 3 tingkat untuk meningkatkan variasi permainan.

## 2. Metode Penelitian

Algoritma minimax mulai dikembangkan dari teori game yang berbasis zero-sum. Teori ini mendeskripsikan situasi di mana jika terdapat pemain yang mengalami pendapatan, pemain lain akan mengalami kehilangan dengan nilai yang sama dari pendapatan tersebut, dan sebaliknya [9]. Teori minimax menyatakan setiap dua orang pemain dalam permainan berbasis zero-sum, terdapat nilai  $V$  dari strategi yang dimiliki pemain. Strategi yang ditentukan pemain kedua akan menghasilkan konsekuensi kemungkinan untuk pemain pertama,  $V$ . Strategi yang ditentukan pemain pertama akan menghasilkan konsekuensi kemungkinan untuk pemain kedua,  $-V$ . Secara setara, strategi pemain pertama akan memastikan suatu nilai  $V$  tanpa memperdulikan strategi pemain kedua, dan bersamaan pemain kedua akan memastikan dirinya kehilangan nilai sebesar  $-V$  [5,6].

Algoritma minimax dapat menghasilkan pilihan langkah yang baik dengan mengasumsikan bahwa pemain lawan akan selalu memilih langkah terbaik untuk dirinya dan langkah terburuk bagi komputer. Prinsip dasar pada algoritma minimax ini adalah jalur yang akan dipilih oleh komputer merupakan jalur maksimum (*max node*) yang akan menghasilkan nilai maksimum di jalur tersebut, dan saat lawan yang akan bermain akan meminimalkan (*min node*) nilai komputer. Jadi, komputer bertujuan untuk memaksimalkan kemungkinan nilai paling rendah yang akan diperoleh komputer [5].



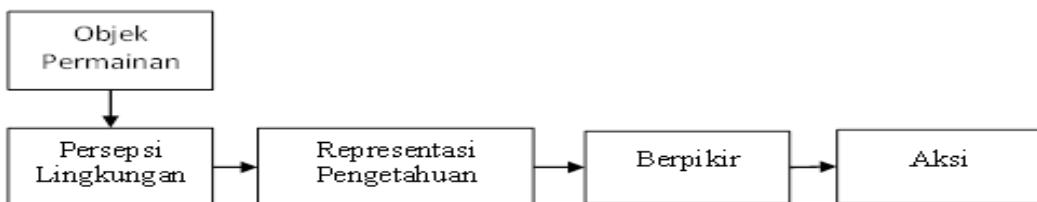
Gambar 1. Ilustrasi algoritma Minimax

Proses pencarian dalam algoritma minimax dimulai dari jalur paling kiri terlebih dahulu, sehingga *Depth First Search* akan menelusuri simpul paling kiri bawah yaitu 5. Nilai 5 disimpan sebagai nilai maksimum sementara karena berada di tingkat max. Kemudian *Depth First Search* melakukan *backtrack* dan menelusuri simpul yang bertetangga dengan simpul 5 yaitu simpul 2. Sehubungan nilai 5 lebih besar dari nilai 2, maka nilai 2 tidak disimpan. *Depth First Search* akan melakukan *backtrack* ke tingkat min sehingga nilai 5 yang diperoleh akan disimpan sebagai nilai minimum sementara. Untuk simpul 1 dan 3, nilai 3 yang akan disimpan karena merupakan nilai maksimum di tingkat max. Saat mencapai tingkat min, sudah ada nilai minimum sementara yaitu 5, namun karena nilai 3 lebih kecil daripada nilai 5, maka nilai 5 akan digantikan dengan nilai 3. Nilai 3 akan disimpan sebagai nilai maksimum sementara di tingkat paling atas karena merupakan tingkat max. Penelusuran jalur kanan akan dilakukan dengan cara yang sama seperti penelusuran jalur kiri sehingga diperoleh nilai 6. Namun karena nilai maksimum sementara pada tingkat paling atas adalah nilai 3, maka nilai 3 akan digantikan dengan nilai 6 karena nilai 6 lebih besar daripada nilai 3. Dengan demikian, jalur yang akan dipilih menggunakan algoritma Minimax adalah jalur sebelah kanan karena untuk kondisi terburuknya, pemain akan mendapatkan nilai 6 sedangkan jika pemain memilih jalur kiri, pemain hanya akan mendapatkan nilai 3 (Gambar 1).

## 3. Hasil Penelitian

Permainan Papan Strategi adalah jenis permainan papan yang dikembangkan dengan mengimplementasikan algoritma minimax. Minimax merupakan suatu algoritma pencarian yang terdapat dalam bidang ilmu kecerdasan buatan. Algoritma bekerja dengan metode penelusuran seperti *Depth First Search* dan kemudian mengambil nilai min/max sesuai dengan giliran pemain dalam permainan. Pada giliran komputer, akan dipilih nilai dari *leaf* yang paling besar (*max*) yang merupakan gerakan terbaik yang dilakukan oleh komputer. Algoritma ini akan berganti giliran memainkan giliran pemain dan kemudian memilih nilai *leaf* yang paling kecil (*min*) yang merupakan gerakan terbaik yang dilakukan pemain.

Dalam membangun sebuah sistem sangat dibutuhkan rancangan dari arsitektur sistem tersebut yang menjadi dasar dalam pengembangan sistem. Bagian inilah yang menjadi dasar dalam mengarahkan perkembangan permainan papan strategi. Berikut ini adalah skema kerja kecerdasan buatan dalam permainan papan strategi (Gambar 2).

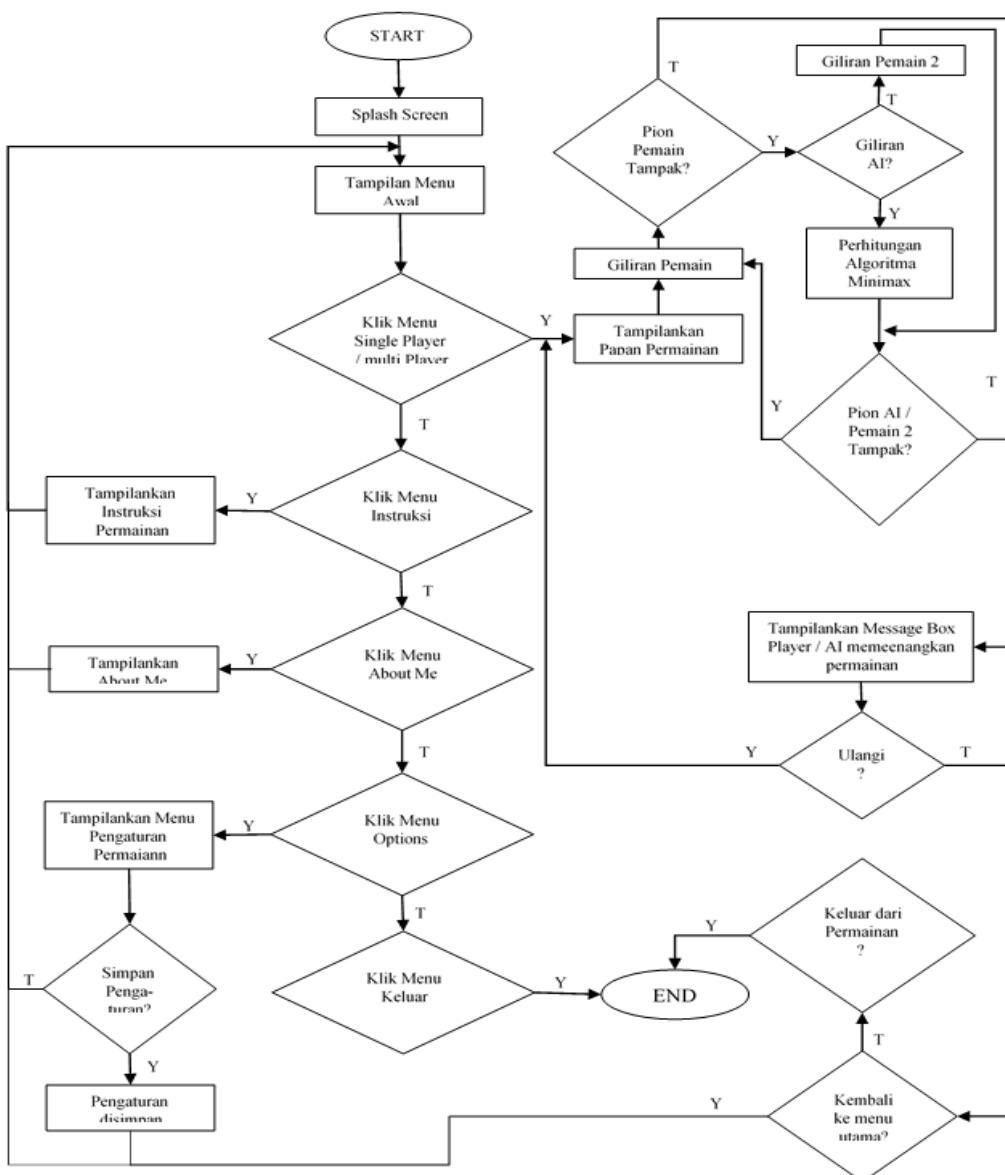


Gambar 2. Skema Proses Kecerdasan Buatan

Kecerdasan buatan mengawali variabel proses mulai dari objek permainan adalah merupakan seluruh komponen maupun objek yang berada dalam papan permainan, persepsi lingkungan adalah proses memperoleh informasi dari lingkungan mengenai objek dan kondisi lingkungan permainan, representasi pengetahuan adalah merepresentasikan pemahamannya mengenai objek dan lingkungannya, berpikir artinya menyimpulkan implikasi akan apa yang diketahui dan pilihan-pilihan apa yang ia miliki untuk mendapatkan hasil seoptimal mungkin, dan aksi artinya pilihan atas apa yang harus dilakukan dalam mencapai tujuan dari permainan. Berikut adalah komponen-komponen dalam permainan papan strategi:

- a. Sebuah papan permainan standar yang berukuran 3x3 dan 4x4. Papan yang digunakan dibuat sedemikian rupa sehingga memberikan kesan papan permainan yang elegan dan menarik dipandang oleh pemain;
- b. Pion dalam permainan yang akan digerakkan oleh pemain dan juga komputer yang mengimplementasikan algoritma minimax;
- c. Pijakan/platform yang memiliki 3 variasi ketahanan.
- d. Cara bergerak pion masing-masing pemain dapat bergerak ke segala arah dengan syarat tidak melewati batas papan permainan dan juga jangkauan dari setiap pion hanyalah satu petak pijakan dari posisi pion. Setiap pion tidak dapat saling menindih/berada pada satu petak pijakan secara bersamaan.
- e. Posisiawal masing-masing pemain dan juga komputer memiliki posisi awal yang sudah ditetapkan, dengan pion pemain berada di bagian sudut kiri atas dan pion lawan berada di bagian diagonal kanan bawah pion pemain/bagian sudut kanan bawah.
- f. Permainan akan berakhir ketika salah satu pemain megalami mati langkah. Pemenang akan diberitahukan melalui kotak pesan dalam permainan.

Perangkat lunak permainan papan strategi tersusun dari enam tampilan, antara lain tampilan *splashscreen*, tampilan main menu, tampilan permainan papan strategi, tampilan instruksi permainan, tampilan konfigurasi permainan dan tampilan about. Adanya kolaborasi dari setiap tampilan tersebut, akan memberikan *interface* yang *user-friendly*. Perancangan ini dibuat sedemikian rupa untuk memperoleh hasil yang optimal dengan berbagai aturan-aturan tertentu yang berlaku pada permainan papan strategi ini agar dapat menghasilkan antarmuka antara perangkat lunak dan pemakai yang baik dan benar (Gambar 3)..



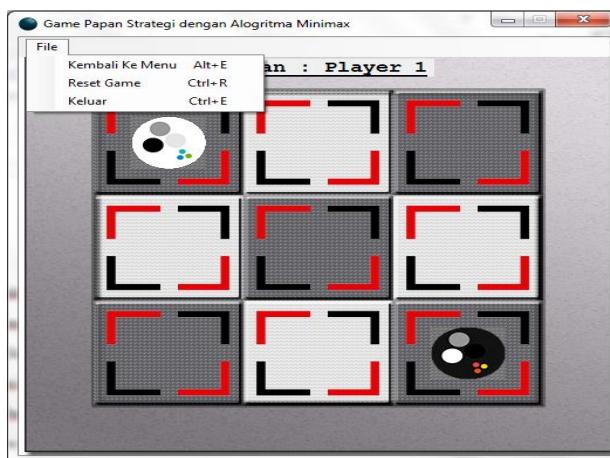
Gambar 3. Bagan Alir Permainan Papan Strategi

Form MainMenu berfungsi sebagai form utama dalam perangkat lunak dan memiliki beberapa menu: Menu Single Game, untuk memulai permainan dengan melawan komputer sebagai lawan tanding. Menu Multi Player Game, juga untuk memulai permainan akan tetapi lawan tandingnya adalah sesama pemain. Menu Instruksi, merupakan fitur yang diberikan kepada pengguna yang baru pertama kali menggunakan perangkat lunak. Menu Option, merupakan fitur dalam permainan papan strategi yang memberikan pengguna pengalaman lebih dalam memainkan permainan dengan memberikan fitur pengaturan warna pion, pengaturan ukuran papan permainan yang akan digunakan dan juga pengaturan level ketahanan papan permainan. Menu About Me, untuk menampilkan identitas programmer dari aplikasi perangkat lunak. Menu Exit, untuk keluar dari aplikasi perangkat lunak (Gambar 4).



Gambar 4. Tampilan Form MainMenu

Form Game merupakan form yang akan dijalankan ketika menu 'Single Game' atau menu 'Multi Player Game' dipilih. Form ini menawarkan beberapa menu, yaitu : menu 'Kembali Ke Menu' untuk menutup form Game dan kembali ke form MainMenu, menu 'Reset Game' untuk mengembalikan kondisi permainan, menu 'Keluar' untuk keluar dari aplikasi perangkat lunak. Algoritma minimax dalam permainan papan strategi berasumsi bahwa pemain pertama memberikan langkah penyelesaian yang bernilai  $V$  terhadap permainan pemain kedua, dan sebaliknya, pemain kedua memberikan langkah penyelesaian bernilai  $-V$ . Oleh karena itu, algoritma minimax beranggapan bahwa dimana pemain yang satu berjuang untuk mendapat nilai maksimum, sedangkan lawannya berjuang untuk mendapat nilai minimum(Gambar 5).



Gambar 5. Tampilan Form MainMenu

Selanjutnya berikut ini merupakan pseudo-code Fungsi PencarianMinimax yang akan digunakan:

```
FUNCTION PencarianMinimax(level)
    IF levelPencarian = 4 THEN
        RETURN 0
    END IF

    LOOP sejumlah baris papan
    LOOP sejumlah kolom papan
        IF kondisiPapanAI[baris][kolom] >= 3 THEN
            IF posisiAI[baris][kolom] = 'AI' THEN
                RETURN level - 1000
            ELSE
                RETURN 1000 - level
            END IF
        END IF
```

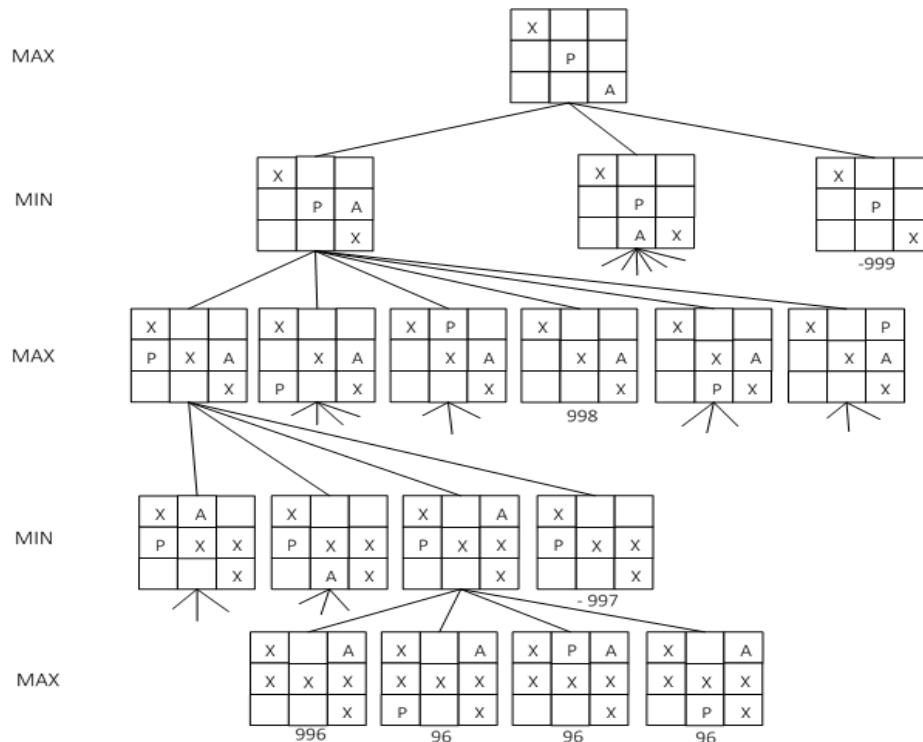
```
        END IF
    END LOOP
END LOOP

SET pilihan = -1000
SET temp = -1000
IF giliranAI = 'Pemain' THEN
    LOOP sejumlah baris langkah giliranAI = 'Pemain'
        LOOP sejumlah kolom langkah giliranAI = 'Pemain'
            SET posisiAI[[baris]][[kolom]] = 'Pemain'
            SET kondisiPapanAI[[baris]][[kolom]] += 1
            SET giliranAI = 'AI'
            SET temp = PencarianMinimax(level + 1)
            SET giliranAI = 'Pemain'
            SET kondisiPapanAI[[baris]][[kolom]] -= 1
            SET posisiAI[[baris]][[kolom]] = 'AI'
            IF pilihan = -1000 THEN
                SET pilihan = temp
            END IF
            IF temp < pilihan THEN
                SET pilihan = temp
            END IF
        END LOOP
    END LOOP
ELSE
    LOOP sejumlah baris langkah giliranAI = 'AI'
        LOOP sejumlah kolom langkah giliranAI = 'AI'
            SET posisiAI[[baris]][[kolom]] = 'AI'
            SET kondisiPapanAI[[baris]][[kolom]] += 1
            SET giliranAI = 'Pemain'
            SET temp = PencarianMinimax(level + 1)
            SET giliranAI = 'AI'
            SET kondisiPapanAI[[baris]][[kolom]] -= 1
            SET posisiAI[[baris]][[kolom]] = 'Pemain'
            IF pilihan = -1000 THEN
                SET pilihan = temp
            END IF
            IF temp > pilihan THEN
                SET pilihan = temp
            END IF
        END LOOP
    END LOOP
END IF
RETURN pilihan
End FUNCTION
```

Menurut alur algoritma minimax permainan papan strategi, pada giliran AI, algoritma minimax dijalankan dan akan mengevaluasi seluruh node yang ada dalam pohon pencarian satu persatu hingga selesai dengan memanggil fungsi PencarianMinimax secara rekursif hingga tingkat kedalaman tertentu. Dalam permainan papan strategi tingkat kedalaman dalam permainan adalah sedalam 4 tingkat. Hal tersebut dilakukan karena jumlah node yang akan dievaluasi tidaklah sedikit sehingga memerlukan waktu yang cukup besar dalam pencarian solusi. Melalui pemanggilan fungsi PencarianMinimax dalam permainan papan strategi, AI akan mencoba memainkan pemain juga secara langsung dengan membentuk sebuah papan permainan semu. Fungsi PencarianMinimax akan dilakukan terus sejumlah langkah yang mungkin dilakukan dan sedalam 4 tingkat. Penelusuran algoritma minimax ini dilakukan dengan metode *depth first-search*. Selanjutnya nilai setiap *leaf* akan dikembalikan ke tingkat diatasnya. Setiap tingkat percabangan genap akan diambil nilai maksimum yang memberikan nilai gerakan terbaik bagi AI sedangkan setiap tingkat percabangan ganjil akan diambil nilai minimum yang

memberikan nilai gerakan terbaik bagi pemain. Setelah rekursi fungsi PencarianMinimax selesai dilakukan, pion AI akan digerakkan ke posisi paling menguntungkan bagi AI.

Untuk penelusuran pohon permainan pada permainan papan strategi menggunakan algoritma *Depth First Search*, maka tingkat kedalaman yang akan ditelusuri adalah jumlah kotak yang belum diisi dikurangkan dengan jumlah kotak yang sudah terisi. Diketahui bahwa dari awal permainan, sudah ada 2 kotak yang terisi oleh tiap pion, dan jumlah kotak keseluruhan dalam permainan papan strategi adalah 7 kotak dengan ukuran papan permainan 3x3 dan tingkat ketahanan papan adalah level 1, maka tingkat kedalaman pada permainan papan strategi dari awal permainan adalah 7 tingkat. Untuk faktor percabangan dalam permainan papan strategi mempunyai jumlah yang bervariasi dari tingkat ke tingkat. Oleh karena itu, memisalkan saja hanya ada 5 faktor percabangan dari tingkat ke tingkat walaupun pada kenyataannya bisa lebih maupun kurang. Dengan demikian, telah memperoleh nilai untuk faktor percabangan dan juga nilai untuk faktor kedalaman yaitu  $O(5^7)$ . Berikut adalah pengujian dari algoritma minimax yang diimplementasikan dalam permainan papan strategi :



Gambar 6. Pohon permainan papan strategi

Prosesnya berawal dari sebuah kotak besar yang didalamnya terdapat 9 buah kotak yang merupakan papan permainan dengan "P" yang merupakan posisi pemain, "A" yang merupakan posisi AI dan "X" yang merupakan posisi pijakan yang telah hancur. Selanjutnya untuk bagian MAX merupakan giliran dari komputer yang mengambil nilai terbesar dalam setiap cabang pada tingkat yang sama, sedangkan pada bagian MIN merupakan giliran dari pemain yang mengambil nilai terkecil dalam setiap cabang pada tingkat yang sama (Gambar 6).

Pada tingkat 4 terlihat bahwa nilai MAX yang akan dipilih adalah 996 yang merupakan nilai terbaik yang akan diambil oleh komputer. Setelah nilai dikembalikan pada tingkat 3, nilai MIN yang akan dipilih adalah nilai paling minimal yang terdapat pada setiap cabang dengan tingkat yang sama yaitu -997. Selanjutnya dilakukan pengembalian nilai pada tingkat 2 yang merupakan giliran komputer. Pada tingkat ini, nilai MAX yang akan dipilih adalah 998. Nilai MAX yang didapat akan dikembalikan pada tingkat 1 yang mengambil Nilai MIN yaitu -999. Terakhir, pada tingkat 0 akan dipilih nilai MAX yang merupakan langkah yang benar-benar akan dijalankan oleh komputer.

#### 4. Kesimpulan

Proses pembangunan pohon pencarian minimax dilakukan dengan metode *depth first search*. Algoritma Minimax mampu menganalisis segala kemungkinan posisi permainan untuk menghasilkan keputusan terbaik. Ketika algoritma minimax digunakan untuk memroses data

dengan ukuran masukan yang besar, karena proses untuk membangun pohon pencarian dengan algoritma ini memiliki kompleksitas algoritma eksponensial. Untuk itu memerlukan waktu yang lebih lama untuk menemukan solusi yang terbaik. Untuk mengurangi waktu yang diperlukan oleh algoritma minimax dalam penelusurannya, dapat dilakukan pemberian optimasi dalam algoritma ini agar tidak semua simpul dibangkitkan. Algoritma optimasi yang diberikan dapat berupa optimasi MTD(f), alpha-beta.

### References

- [1] Crawford, C. Chris Crawford on Game Design, Peach Pit, United State.2003.
- [2] McGonigal, J. Reality is Broken: Why Games Make Us Better and How They Can Change the World, Penguin Books, New York.2011.
- [3] Heaton, Jeff. Artificial Intelligence for Humans, Volume 1: Fundamental Algorithms. CreateSpace Independent Publishing Platform. First Edition. 2013.
- [4] Rahmat, R.F., Muchtar, M.A., dan Arisandi, D. Implementasi Permainan Reversi Menggunakan Penelusuran BFS dengan Konsep Algoritma Minimax. Skripsi. Program Studi Teknik Informatika. Jurusan Teknik Informatika. Universitas Sumatera Utara. Medan; 2012.
- [5] Coppin, B. Artificial Intelligence Illuminated, Jones & Bartlett Learning, United Kingdom. 2004.
- [6] Poole, David L. and Mackworth, Alan K. Artificial Intelligence: Foundations of Computational Agents. Cambridge University Press. 2010.
- [7] Gapur, Abdul. Aplikasi Struktur Data The Minimax Game Tree Pada Permainan Catur. Tugas Mata Kuliah IF2251 Strategi Algoritmik. Departemen Teknik Informatika. Institute Teknologi Bandung. Bandung; 2007.
- [8] Firmansyah, D.H. Implementasi Algoritman Minimax pada Permainan Tic-Tac-Toe skala 9 x 9. Skripsi. Program Studi Teknik Informatika. Jurusan Teknik Informatika. Universitas Komputer Indonesia. Bandung; 2008.
- [9] Ilman, A. Penerapan Algoritma Minimax Dengan Optimasi MTD(f) pada Permainan Catur. Makalah IF2251 Strategi Algoritmik. Departemen Teknik Informatika. Institute Teknologi Bandung. Bandung; 2008.