

PENERAPAN ALGORITMA *ITERATIVE-DEEPENING SEARCH* (IDS) DALAM PENYELESAIAN PERMAINAN TEKA-TEKI KAKURO

Jackri Hendrik

Dosen Tetap Program Studi Teknik Informatika STMIK - TIME, Medan, Indonesia
Jl. Merbabu No.32 AA-BB Medan, Indonesia

ABSTRAK

Kakuro merupakan permainan teka-teki logika yang sangat populer di Jepang dan muncul di hampir 100 majalah dan koran Jepang. Nikoli, pencipta Kakuro, bahkan berhasil menjual sekitar 1 juta buku Kakuro. Cara bermain Kakuro adalah pemain harus mengisi setiap kotak yang kosong dengan angka dari 1 hingga 9, dengan syarat bahwa total dari angka per baris dan kolom harus sesuai dengan nilai pada kotak petunjuk. Angka tidak boleh berulang pada baris dan kolom yang sama. Algoritma pencarian *Iterative-Deepening Search* (IDS) dapat digunakan untuk mencari penyelesaian dari permainan Kakuro. Pencarian akan dimulai dari kotak jawaban pertama dan diberikan angka 1 dilanjutkan dengan penelusuran pada kotak-kotak berikutnya. Apabila tidak terdapat angka yang dapat dimasukkan pada satu kotak, maka IDS akan kembali ke kotak sebelumnya dan mengganti angka pada kotak tersebut. Penelusuran dilakukan hingga semua kotak jawaban terisi. Aplikasi dapat digunakan untuk mencari jawaban dari permainan teka-teki Kakuro dengan menggunakan algoritma IDS.

Kata kunci: Permainan, Teka teki, Kakuro, IDS

I. PENDAHULUAN

Artificial Intelligence (AI) atau kecerdasan buatan adalah salah satu bidang ilmu komputer yang sangat penting untuk mempelajari kecerdasan manusia. Dalam era teknologi informasi ini, AI berkembang sangat pesat. Salah satu peranan AI adalah mencari penyelesaian yang dapat dilakukan oleh manusia. Namun, dengan kelebihan dalam kecepatan perhitungannya, AI dapat menemukan solusi lebih cepat dibandingkan dengan manusia. Salah satu penyelesaian dari permainan yang dapat ditemukan dengan menggunakan ilmu AI adalah Kakuro.

Kakuro adalah permainan teka-teki logika yang diproduksi oleh Nikoli. Kakuro merupakan permainan teka-teki logika yang sangat populer di Jepang dan muncul di hampir 100 majalah dan koran Jepang. Nikoli bahkan berhasil menjual sekitar 1 juta buku Kakuro. Cara bermain Kakuro adalah pemain harus mengisi setiap kotak yang kosong dengan angka bulat dari 1 hingga 9, dengan syarat bahwa total dari angka per baris dan kolom harus sesuai dengan nilai *input* pada setiap baris dan kolom. Angka tidak boleh berulang pada angka dan baris yang sama.

Salah satu teknik pencarian yang dapat digunakan untuk mencari penyelesaian teka-teki Kakuro adalah algoritma pencarian *Iterative-Deepening Search* (IDS). Metode IDS merupakan metode yang menggabungkan kelebihan dari pencarian *Breadth First Search* (BFS) yang *complete* dan *optimal*, dengan kelebihan *Depth Limited Search* (DFS) yang memiliki *space complexity* rendah atau membutuhkan sedikit memori. IDS melakukan pencarian secara iteratif menggunakan pencarian mendalam yang dimulai dengan batasan nol. Jika belum ditemukan solusi, maka dilakukan iterasi ke-2

dengan batasan level 1. Demikian seterusnya hingga ditemukan solusi. Teknik pencarian IDS akan dilakukan sedikit modifikasi sehingga dalam pencarian solusi, tidak melanggar aturan dari Kakuro.

II. TEORITIS

A. Permainan

Salah satu tugas perkembangan yang harus dipenuhi pada usia 6-13 tahun adalah seorang anak mampu untuk menyesuaikan diri dengan teman sebayanya. Untuk dapat menyesuaikan diri dengan lingkungan anak harus berkomunikasi dan berinteraksi dengan teman sebaya, sebagian besar interaksi teman sebaya selama masa anak-anak adalah melibatkan permainan. Tidak ada bidang lain yang lebih benar kecuali belajar menjadi seseorang yang sosial. Karena belajar sosial bergantung pada kesempatan berhubungan dengan anggota kelompok teman sebaya dan arena hal ini terutama terjadi dalam kegiatan bermain, maka bermain dianggap sebagai alat yang penting bagi sosialisasi[3].

Permainan (*play*) adalah suatu kegiatan menyenangkan yang dilaksanakan untuk kepentingan kegiatan itu sendiri[3].

Permainan adalah suatu bentuk penyesuaian diri manusia yang sangat berguna, menolong anak menguasai kecemasan dan konflik. Karena tekanan tekanan terlepas dalam permainan, anak dapat mengatasi masalah-masalah kehidupan. Permainan memungkinkan anak melepaskan energi fisik yang berlebihan dan membebaskan perasaan-perasaan terpendam. [3]

B. Kecerdasan Buatan

Kecerdasan buatan berasal dari bahasa Inggris "*Artificial Intelligence*" (atau disingkat AI), yaitu *intelligence* adalah kata sifat yang berarti cerdas,

sedangkan *artificial* artinya buatan. Kecerdasan buatan di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia[10].

Para ahli mendefinisikan definisi AI secara berbeda-beda tergantung pada sudut pandang mereka masing-masing. Ada yang fokus pada logika berpikir manusia saja, tetapi ada juga yang mendefinisikan AI secara lebih luas pada tingkah laku manusia. Stuart Russel dan Peter Norvig mengelompokkan AI ke dalam empat kategori, yaitu: [11]

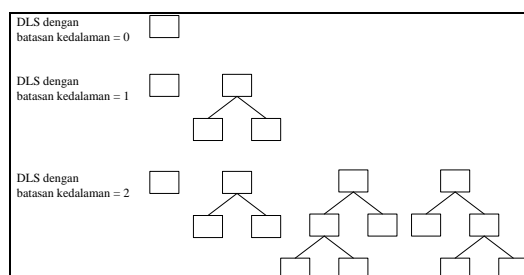
1. Sistem yang dapat berpikir seperti manusia "*Thinking humanly*"
2. Sistem yang dapat bertingkah laku seperti manusia "*Acting humanly*"
3. Sistem yang dapat berpikir secara rasional "*Thinking rationally*"
4. Sistem yang dapat bertingkah laku secara rasional "*Acting rationally*"

Thinking humanly dan *acting humanly* adalah dua definisi dalam arti yang sangat luas. Sampai saat ini, pemikiran manusia yang di luar rasio, yakni refleks dan intuitif (berhubungan dengan perasaan), belum dapat ditirukan sepenuhnya oleh komputer. Dengan demikian, kedua definisi ini dirasa kurang tepat untuk saat ini. Definisi *thinking rationally* terasa lebih sempit daripada *acting rationally*. Oleh karena itu, definisi AI yang paling tepat untuk saat ini adalah *acting rationally*, dengan pendekatan *rational agent*. Hal ini berdasarkan pemikiran bahwa komputer bisa melakukan penalaran secara logis dan juga bisa melakukan aksi secara rasional berdasarkan hasil penalaran tersebut. [11]

C. Iterative Deepening Search

Iterative Deepening Search (IDS) merupakan metode yang menggabungkan kelebihan BFS (*complete* dan *optimal*) dengan kelebihan DFS (*space complexity* rendah atau membutuhkan sedikit memori). Tetapi konsekuensinya adalah *time complexity*-nya menjadi tinggi.

Perhatikan gambar 1. IDS melakukan pencarian secara iteratif menggunakan penelusuran *Depth-Limited-Search* (DLS) dimulai dengan batasan level 0. Jika belum ditemukan solusi, maka dilakukan iterasi ke-2 dengan batasan level 1. Demikian seterusnya sampai ditemukan solusi. [11]



Gambar 1. Penelusuran Metode IDS

Penelusuran di setiap proses pencariannya menggunakan cara kerja DFS, yaitu dengan menelusuri satu node dalam setiap level dari yang paling kiri. Jika pada level yang paling dalam, solusi belum ditemukan, maka pencarian dilanjutkan pada node sebelah kanan. Demikian seterusnya sampai ditemukan solusi. [1]

Dengan adanya kelebihan pada DFS, maka IDS hanya membutuhkan memori yang kecil karena hanya simpul dalam *path* saja yang disimpan. Sama halnya dengan BFS, metode IDS dapat menemukan solusi terbaik karena pencarian dilakukan secara merata terlebih dahulu hingga batas bertambah dan perulangan terjadi, ini mendukung IDS untuk menemukan solusi tanpa akan menemukan jalan buntu jika percabangan ke bawah pada pohon tidak terbatas [7].

Metode pencarian pada IDS menggunakan metode pencarian pada DFS, yaitu pencarian pada sebuah pohon dengan menelusuri satu cabang sebuah pohon sampai menemukan solusi. Pencarian dilakukan pada satu node dalam setiap level dari yang paling kiri. Jika pada level yang paling dalam, solusi belum ditemukan, maka pencarian dilanjutkan pada node sebelah kanan. Node yang kiri dapat dihapus dari memori. Demikian seterusnya sampai ditemukan solusi. Jika solusi ditemukan maka tidak diperlukan proses backtracking (penelusuran balik untuk mendapatkan jalur yang diinginkan). Akan tetapi, dalam metode pencarian IDS, pencarian mencoba menemukan batas kedalaman yang terbaik dengan terus menelusuri batas kedalaman satu per satu, pencarian dilakukan secara iteratif dimulai dari batasan level 0. Jika solusi belum ditemukan, maka dilakukan iterasi berikutnya dengan batasan level 1, demikian seterusnya sampai menemukan solusi[7].

IDS melakukan pencarian berulang-ulang dengan menambah batas kedalaman setiap mengulang pencarian, dan akhirnya menemukan solusi pada penelusuran dengan batas kedalaman tertentu. Solusi ditemukan pada percabangan kiri atau kanan. Pencarian ini akan menemukan solusi lebih cepat daripada penelusuran pada DFS, jika solusi tidak berada pada *node* yang dalam. [5]

D. Kakuro

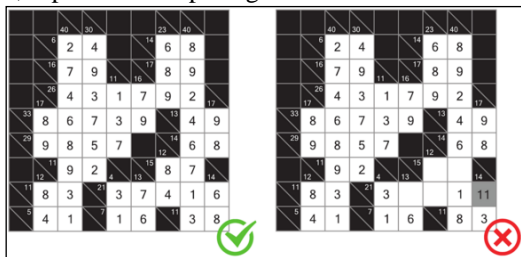
Kakuro muncul pertama sekali pada tahun 1966 di majalah teka teki Amerika Serikat pada tahun 1966. Pada saat itu, permainan teka-teki itu bernama "*Cross Sums*". Pada tahun 1980, seorang pria Jepang yang bernama Maki Kaji mengadakan kunjungan bisnis ke Amerika Serikat. Maki adalah seorang penggemar teka-teki, seperti orang Jepang pada umumnya. Maki sangat tertarik dengan permainan teka-teki ini, sehingga setelah kembali ke Jepang, Maki mulai membuat dan mempublikasi permainan ini di Jepang. Permainan ini dinamakan *kasan kurosu*, kombinasi dari kata Jepang untuk "tambah", dan kata Inggris "*cross*" dalam bahasa Jepang. Selanjutnya,

kasan kurosu berubah nama menjadi *kakuro*. Maki dan perusahaannya, Nikoli berhasil menjual sekitar 1 juta buku *Kakuro*. [12]

Kakuro dimainkan pada sebuah tabel kotak-kotak yang mirip dengan permainan teka teki silang. Di dalam kakuro, jumlah dari nilai *input* harus sama dengan petunjuk yang tertera di kiri atau di atas. Pemain hanya bisa memasukkan angka dari 1 hingga 9, sehingga jumlah angka sama dengan petunjuk. Tidak boleh terdapat angka yang sama pada baris dan kolom yang sama. Hanya akan ada satu solusi untuk setiap *puzzle* [12].

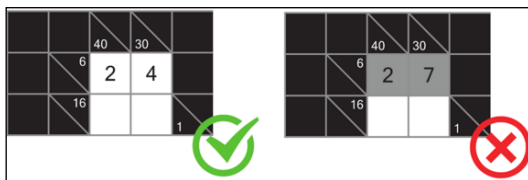
Secara ringkas, aturan permainan Kakuro adalah sebagai berikut: [9]

1. Angka yang dapat dimasukkan adalah 1 sampai 9, seperti terlihat pada gambar 2 berikut.



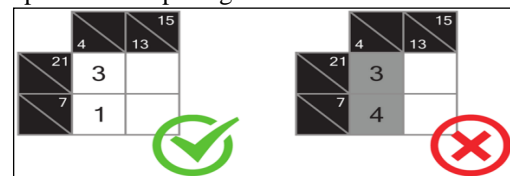
Gambar 2. Aturan-1

2. Jumlah dari angka secara horizontal harus sama dengan petunjuk atau nilai yang tertulis di samping, seperti terlihat pada gambar 3 berikut.



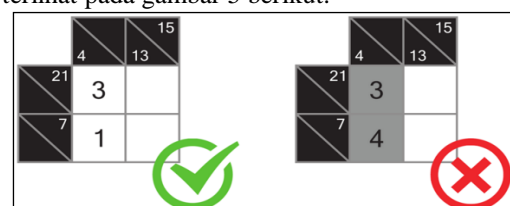
Gambar 3. Aturan-2

3. Jumlah dari angka secara vertikal harus sama dengan petunjuk atau nilai yang tertulis di atas, seperti terlihat pada gambar 4 berikut.



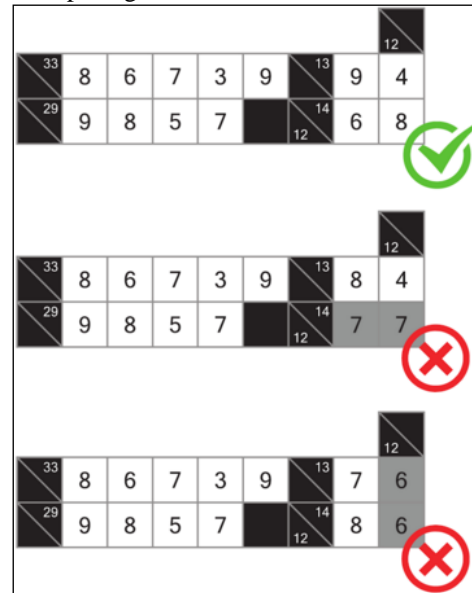
Gambar 4. Aturan-3

4. Hanya boleh terdapat satu angka pada satu baris yang sama dan satu kolom yang sama, seperti terlihat pada gambar 5 berikut.



Gambar 5. Aturan-4

5. Hanya boleh terdapat satu angka pada satu baris yang sama dan satu kolom yang sama, seperti terlihat pada gambar 6 berikut.

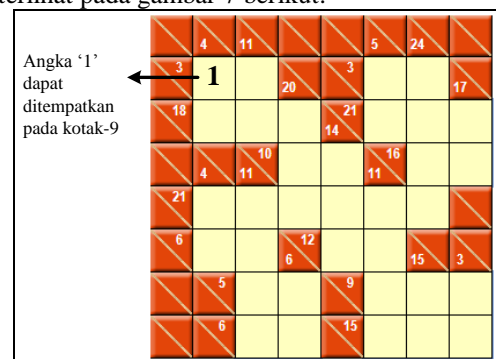


Gambar 6. Aturan-4

III. ANALISA DAN PEMBAHASAN

Pencarian jawaban Kakuro dilakukan dengan menggunakan algoritma pencarian *Iterative Deepening Search* (IDS). Pencarian jawaban Kakuro dengan algoritma pencarian IDS adalah sebagai berikut:

1. Pencarian dimulai dari kotak pertama, yaitu kotak-0. Apabila kotak-0 adalah kotak petunjuk, maka majukan indeks kotak dan lanjutkan pemeriksaan pada kotak-1. Kotak-1 hingga kotak-8 juga merupakan kotak petunjuk, sehingga pencarian berlanjut hingga kotak-9.
2. Pencarian dilanjutkan ke kotak-9. Kotak-9 kosong atau dianggap bernilai '0', tambahkan angka sekarang dengan satu = '1' dan uji angka '1' pada kotak-9. Ternyata angka '1' tidak menyalahi aturan, karena belum terdapat angka '1' pada baris pertama atau kolom pertama. Tempatkan angka '1' pada kotak-9, seperti terlihat pada gambar 7 berikut.



Gambar 7. Penempatan Angka '1' pada

Kotak-9

3. Majukan indeks dan lanjutkan pemeriksaan pada kotak-10. Kotak-10 kosong atau dianggap bernilai '0', tambahkan angka sekarang dengan satu = '1' dan uji angka '1' pada kotak-1. Angka '1' menyalahi aturan, karena telah terdapat angka '1' pada baris yang sama (sebelah kiri). Tambah angka dengan satu dan uji dengan angka '2'. Ternyata angka '2' tidak menyalahi aturan, karena belum terdapat angka '2' pada baris pertama atau kolom kedua, kemudian hasil penjumlahan baris juga sama dengan 3 (sesuai dengan nilai penjumlahan baris pada kotak-8 yang menjadi kotak petunjuk). Tempatkan angka '2' pada kotak-10, seperti terlihat pada gambar 8 berikut.

Angka '2' dapat ditempatkan pada kotak-10

	4	11			5	24	
3	1	2		20	3		17
18				14	21		
	4	11	10		16		
21					11		
6			12			15	3
	5			9			
	6			15			

Gambar 8. Penempatan Angka '2' pada Kotak-10

4. Pencarian dilanjutkan ke kotak-11. Kotak-11 merupakan kotak petunjuk, majukan indeks pencarian ke kotak-12. Kotak-12 merupakan kotak petunjuk, majukan indeks pencarian ke kotak-13. Kotak-13 kosong atau dianggap bernilai '0', tambahkan angka sekarang dengan satu = '1' dan uji angka '1' pada kotak-13. Ternyata angka '1' tidak menyalahi aturan, karena belum terdapat angka '1' pada baris pertama atau kolom yang sama. Tempatkan angka '1' pada kotak-13, seperti terlihat pada gambar 9 berikut.

Angka '1' dapat ditempatkan pada kotak-13

	4	11			5	24	
3	1	2		20	3	1	17
18				14	21		
	4	11	10		16		
21					11		
6			12			15	3
	5			9			
	6			15			

Gambar 9. Penempatan Angka '1' pada Kotak-13

5. Bila setelah diuji hingga angka '9' dan suatu kotak jawaban masih tidak dapat diisi angka, maka pencarian *backtrack* ke kotak jawaban sebelumnya, dan melakukan pengujian kembali dengan angka yang lain.
6. Demikian pencarian berlangsung, hingga semua kotak jawaban terisi dengan angka yang *valid*. Pada posisi ini, solusi soal Kakuro telah terpecahkan.
7. Apabila terjadi *backtrack* pada kotak-0, maka soal Kakuro tidak memiliki solusi atau penyelesaian.

Hasil

Sistem dapat digunakan untuk mencari jawaban dari permainan teka-teki Kakuro dengan ukuran kotak minimal 5 x 5 kotak dan maksimal 10 x 10 kotak. Adapun kelebihan dari sistem adalah sebagai berikut:

1. Sistem dapat digunakan untuk mencari jawaban dari permainan teka-teki Kakuro dengan menggunakan algoritma pencarian *Iterative Deepening Search* (IDS) sehingga pembuat soal Kakuro dapat menguji apakah soal yang telah dibuatnya mempunyai solusi atau tidak.
2. Sistem juga dapat digunakan untuk membantu penggemar teka-teki Kakuro yang membutuhkan waktu dan tenaga yang relatif besar untuk mencari jawaban Kakuro secara manual.
3. Bila soal Kakuro tidak memiliki jawaban, maka sistem akan menampilkan pesan kesalahan.
4. Sistem dapat menampilkan langkah-langkah pencarian IDS, sehingga *user* dapat melihat proses pencarian yang dilakukan oleh sistem.
5. Sistem dapat mencari jawaban lebih cepat apabila digunakan pencarian cepat (sistem tidak mencatat langkah-langkah pencarian).
6. *User* dapat memainkan Kakuro dengan langsung memasukkan angka pada kotak-kotak yang kosong. Apabila solusi Kakuro ditemukan, maka sistem akan menampilkan pesan.

Adapun kelemahan dari sistem pencarian jawaban permainan teka-teki Kakuro adalah sebagai berikut:

1. Sistem tidak mendukung soal Kakuro yang memiliki ukuran di bawah 5 x 5 kotak dan di atas 10 x 10 kotak.
2. Sistem tidak memiliki suara dan animasi serta tidak mendukung efek 3 dimensi.
3. Sistem tidak menyediakan permainan Kakuro dari level paling mudah hingga paling rumit.
4. Sistem tidak dapat dijalankan secara *mobile* melalui *web* dan *smartphone*.

IV. KESIMPULAN

Setelah menyelesaikan perancangan aplikasi penyelesaian permainan teka-teki Kakuro menggunakan algoritma pencarian *Iterative-*

Deepening Search (IDS), beberapa hal yang dapat disimpulkan adalah sebagai berikut:

1. Aplikasi dapat digunakan untuk mencari jawaban dari permainan teka-teki Kakuro dengan menggunakan algoritma pencarian *Iterative Deepening Search* (IDS) sehingga pembuat soal Kakuro dapat menguji apakah soal yang telah dibuatnya mempunyai solusi atau tidak.
2. Aplikasi juga dapat digunakan untuk membantu penggemar teka-teki Kakuro yang membutuhkan waktu dan tenaga yang relatif besar untuk mencari jawaban Kakuro secara manual.
3. Bila soal Kakuro tidak memiliki jawaban, maka aplikasi akan menampilkan pesan kesalahan.
4. Aplikasi dapat menampilkan langkah-langkah pencarian IDS, sehingga *user* dapat melihat proses pencarian yang dilakukan oleh sistem.
5. Aplikasi dapat mencari jawaban lebih cepat apabila digunakan pencarian cepat (sistem tidak mencatat langkah-langkah pencarian).
6. *User* dapat memainkan Kakuro dengan langsung memasukkan angka pada kotak-kotak yang kosong. Apabila solusi Kakuro ditemukan, maka sistem akan menampilkan pesan.
7. Aplikasi merupakan implementasi nyata dari kecerdasan buatan (*artificial intelligence*), terutama algoritma pencarian IDS untuk mencari penyelesaian dari suatu permasalahan, dalam hal ini soal Kakuro.

DAFTAR PUSTAKA

- [1] Cormen, et.al, 2009, *Introduction to Algorithms*, Third Edition. MIT Press, Massachusetts.
- [2] Freeman Joan, dan Utami Munandar, 2001, **Cerdas dan Cemerlang**, PT. Gramedia Pustaka Indonesia, Jakarta
- [3] Hurlock, 2010, **Psikologi Perkembangan**, Penerbit Erlangga, Jakarta.
- [4] Le Compte, D., 2010, *200 Crazy Clever Kakuro Puzzles*, Lulu.com.
- [5] Prasetyo, R. Muhammad Khalil, 2012, **Penyelesaian Permainan Checkers Pada Mobile Device Berbasis Android Menggunakan Algoritma Iterative Deepening Search**, USU, Medan.
- [6] Romlah, 2001, **Teori dan Praktek Bimbingan dan Konseling Kelompok**, Universitas Negeri Malang Press, Malang.
- [7] Russell, S. J. dan Norvig, P, 2010, *Artificial Intelligence: A Modern Approach, Third Edition*, Pearson Education, New Jersey.
- [8] Santrock, 2006, *Human Adjustment*, University Of Texas at Dallas. Mc Graw, Hill Companies.
- [9] Snels, N., 2014, **King-Sized Kakuro Volume 3**, *PuzzleBooks.net*.
- [10] Sutojo, T., dkk, 2011, **Kecerdasan Buatan**, Penerbit Andi, Yogyakarta.
- [11] Suyanto, 2011, *Artificial Intelligence*, Penerbit Informatika, Bandung.
- [12] Timmerman, C., 2006, *The Everything Easy Kakuro Book*, Adams Media, Avon, Massachusetts