

Penyelesaian Game Puzzle Hashiwokakero dengan Teknik Solving Hashi dan Depth First Search

Eriska Amrina Pratiwi, Rusdi Efendi, dan Reza Firsandaya Malik
Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Sriwijaya
Email: rezafm@unsri.ac.id

Abstract— *Hashi* merupakan game *puzzle* logika yang mengasah otak dan sangat populer di Jepang. Makalah ini membahas mengenai cara mencari penyelesaian terhadap game *puzzle Hashi* dengan menggunakan empat teknik *Solving Hashi*, seperti: *Just Enough Neighbours*, *One Unsolved Neighbour*, *Few Neighbours*, *Leftovers* dan *Isolation* serta *Depth First Search* (DFS). Teknik *Solving Hashi* digunakan untuk mencari dan membangun jembatan-jembatan yang sudah pasti dapat dibangun antar pulau, sedangkan DFS mencari dan membangun sisa jembatan yang tidak ditemukan oleh teknik *Solving Hashi*. Hasilnya teknik *Solving Hashi* dan DFS dapat menyelesaikan setiap skenario *puzzle Hashi*.

Kata Kunci — *Hashiwokakero*, Teknik *Solving Hashi*, *Depth First Search*

I. PENDAHULUAN

Hashi adalah permainan *puzzle* logika yang diciptakan oleh Nikoli, sebuah perusahaan *game* di Jepang yang spesialis dalam pembuatan permainan logika. *Hashiwokakero* pertama sekali muncul di majalah *puzzle* Nikoli pada bulan September 1990 (Nikoli, 2011). Selain mengasyikkan, permainan *Hashi* juga menguji logika dan kecerdasan orang yang memainkannya. Di Inggris, permainan *Hashi* dikenal dengan sebutan "*Bridges or Chopsticks*", sedangkan di sebagian besar Eropa, permainan *Hashi* dikenal dengan sebutan "*Ai-Ki-Ai*" (Nikoli, 2011).

Struktur tulisan ini dari bagian I mengenai pendahuluan yang membahas perkembangan permainan *puzzle Hashiwokakero*, bagian II membahas metodologi, bagian III membahas hasil dan pembahasan dan bagian IV kesimpulan.

II. METODOLOGI

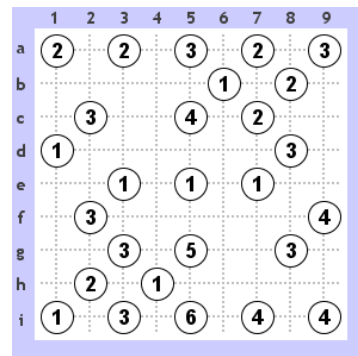
A. Peraturan Permainan

Permainan *Hashi* dimainkan pada sebuah kotak. Di dalam kotak ini, terdapat beberapa lingkaran (sel) yang terletak secara horizontal dan vertikal. Sel ini dikenal juga dengan sebutan pulau. Setiap pulau, terdapat angka dari 1 sampai 8. Tujuan dari permainan *Hashi* adalah menghubungkan semua pulau dengan

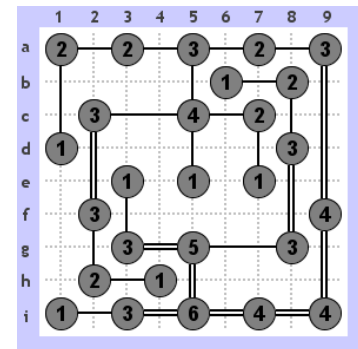
membuat jembatan antar pulau dan tidak ada pulau yang terisolir (atau tidak terhubung satu sama lain).

Aturan pembuatan jembatan pada kotak permainan *Hashi* adalah sebagai berikut:

1. Satu jembatan dapat menghubungkan dua pulau yang terletak pada koordinat x yang sama (horizontal) atau koordinat y yang sama (vertikal),
2. Jembatan tidak boleh saling bertimpaan dengan jembatan lainnya,
3. Maksimum hanya boleh terdapat 2 jembatan di antara 2 pulau yang sama,
4. Banyaknya jembatan yang terhubung ke pulau harus sama dengan angka yang tertulis pada pulau.



Gambar 1. Keadaan Awal Permainan *Hashi* (Indigopuzzle, 2011)



Gambar 2. Keadaan Akhir Permainan *Hashi* (Indigopuzzle, 2011)

Pada Gambar 1 menunjukkan keadaan awal permainan Hashi yang menjadi soal dan akan berubah setiap permainan puzzle ini. Untuk Gambar 1 maka keadaan akhir permainan Hashi diselesaikan dengan aturan pembuatan jembatan yang telah dipaparkan di atas.

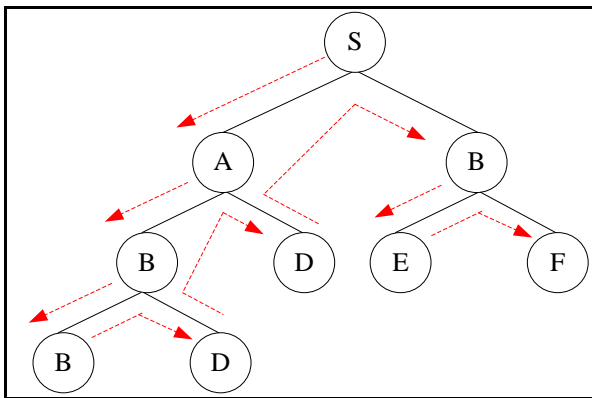
B. Teknik Solving Hashi

Teknik Solving Hashi merupakan suatu teknik yang khusus digunakan dalam mencari solusi dalam permainan Hashi. Menurut Indigopuzzles, cara terbaik untuk mencari jawaban permainan Hashi adalah dengan berusaha menyelesaikan satu per satu masalah dalam setiap soalnya. Beberapa teknik solving Hashi yang dipublikasikan oleh Indigopuzzles (Indigopuzzles, 2011) dan dapat digunakan untuk mencari penyelesaian permainan Hashi, yaitu:

1. Teknik *Just Enough Neighbours*
2. Teknik *One Unsolved Neighbour*
3. Teknik *Few Neighbours*
4. Teknik *Leftovers*
5. Teknik *Isolation*

C. Depth First Search

Pencarian dengan metode *Depth First Search* (DFS) dilakukan dari *node* awal secara mendalam hingga yang paling akhir (*dead-end*) atau sampai ditemukan. Dengan kata lain, simpul cabang atau anak yang terlebih dahulu dikunjungi. Sebagai ilustrasinya dapat dilihat pada Gambar 3. (Desiani dan Arhami, 2005).



Gambar 3. Depth First Search (Desiani dan Arhami, 2005)

Berdasarkan gambar, proses pencarian dilakukan dengan mengunjungi cabang terlebih dahulu hingga tiba di simpul terakhir. Jika tujuan yang diinginkan belum tercapai maka pencarian dilanjutkan ke cabang sebelumnya, turun ke bawah jika memang masih ada cabangnya. Begitu seterusnya hingga diperoleh tujuan (*goal*). Operasi semacam ini dikenal dengan sebutan *backtracking*.

DFS juga memiliki kelebihan di antaranya adalah cepat mencapai kedalaman ruang pencarian. Jika diketahui bahwa lintasan solusi permasalahan akan panjang maka DFS tidak akan memboros waktu untuk melakukan sejumlah besar keadaan ‘dangkal’ dalam permasalahan *graph* / pohon. DFS jauh lebih efisien untuk ruang pencarian dengan banyak cabang

karena tak perlu mengevaluasi semua simpul pada suatu level tertentu pada daftar *open*. Selain itu, DFS memerlukan memori yang relatif kecil karena hanya *node-node* pada lintasan yang aktif saja yang disimpan.

III. HASIL DAN PEMBAHASAN

A. Pencarian Solusi dengan Teknik Solving Hashi

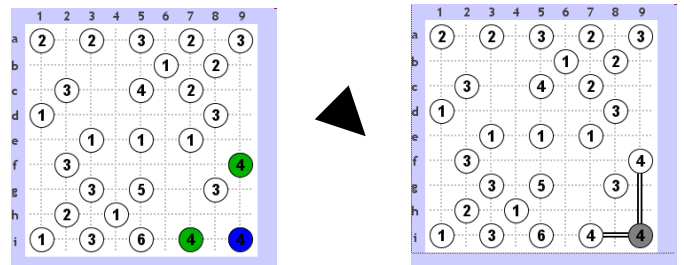
Teknik *solving Hashi* yang digunakan dalam mencari solusi permainan Hashi, mencakup beberapa teknik berikut:

1. Teknik *Just Enough Neighbours*
2. Teknik *One Unsolved Neighbour*
3. Teknik *Few Neighbours*
4. Teknik *Leftovers*
5. Teknik *Isolation*

Semua teknik ini akan diimplementasikan secara berurutan untuk mencari jawaban dari soal permainan Hashi.

B. Teknik Just Enough Neighbours

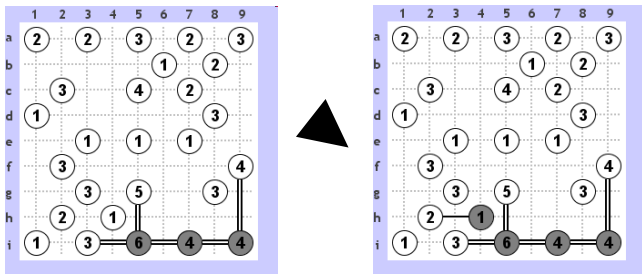
Teknik ini adalah teknik yang digunakan untuk menghubungkan suatu pulau dengan tetangganya, bila jumlah jembatan yang dimiliki sama dengan jumlah jembatan yg dapat dibangun ke tetangga. Perhatikan pada Gambar 4 bahwa pulau (4) pada i9 membutuhkan 4 jembatan. Dan oleh karena berada di pojok, pulau tersebut tidak mempunyai pilihan jembatan lain selain menghubungkan diri dengan pulau f9 dan i7, dengan masing-masing 2 jembatan.



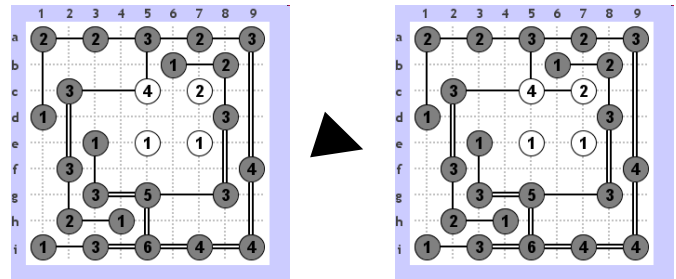
Gambar 4. Teknik *Just Enough Neighbours* (Indigopuzzle, 2011)

C. Teknik One Unsolved Neighbours

Teknik ini adalah teknik yang digunakan untuk menghubungkan suatu pulau dengan tetangganya, bila hanya ada satu pulau tetangga didekat pulau tersebut dengan sisa bobot satu. Perhatikan pada Gambar 5 bahwa pulau (1) pada h4 hanya membutuhkan 1 jembatan. Dan oleh karena tidak ada pulau lain yang dapat diraih, maka pulau h4 hanya bisa menghubungkan dirinya dengan pulau (2) pada h2.



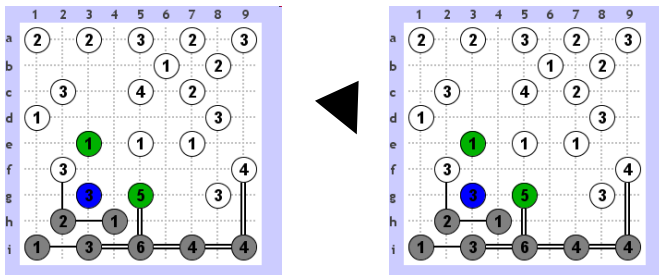
Gambar 5. Teknik *One Unsolved Neighbours* Indigopuzzle, 2011)



Gambar 7. Teknik *Leftovers* (Indigopuzzle, 2011)

D. Teknik Few Neighbours

Teknik ini berdasarkan pada peraturan bahwa “hanya boleh ada 2 jembatan antar pulau”. Perhatikan pada Gambar 6 bahwa pulau (3) pada g3 hanya bisa meraih 2 pulau lainnya. Dan oleh karena pulau e3 hanya mempunyai angka 1, maka pulau g3 menghubungkan 1 jembatan ke pulau e3 dan 2 jembatan ke pulau g5.



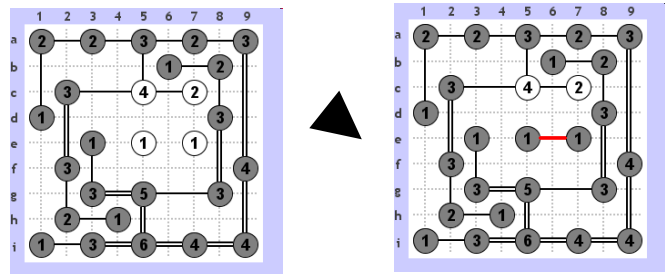
Gambar 6. Teknik *Few Neighbours* (Indigopuzzle, 2011)

E. Teknik Leftovers

Teknik ini adalah teknik yang digunakan untuk menghubungkan suatu pulau dengan tetangganya, bila pulau tersebut memiliki sisa bobot sebesar N dan banyak pulau tetangga sebanyak N, dengan (N-1) tetangga mempunyai bobot / sisa bobot = 1, maka bangun 1 jembatan dari pulau tsb dengan tetangganya yg tidak memiliki sisa bobot 1. Perhatikan pada Gambar 7 bahwa pulau (4) pada c5, membutuhkan 2 jembatan lainnya. Salah satu tetangganya adalah pulau (1) pada e5 yang hanya dapat menerima 1 jembatan. Oleh karena itu, setidaknya satu jembatan pada pulau c5 harus mengarah ke pulau lainnya, yaitu pulau (2) pada c7.

F. Teknik Isolation

Teknik ini adalah teknik yang paling penting untuk menyelesaikan *puzzle Hashi*. Teknik ini menggunakan aturan bahwa setiap pulau harus bisa menjangkau pulau lainnya. Perhatikan pulau (1) di e5 pada Gambar 8 yang hanya akan memiliki 1 jembatan berikut.



Gambar 8. Teknik *Isolation* (Indigopuzzle, 2011)

Pada gambar 8 jika pulau e5 dihubungkan ke pulau di samping kanannya, maka kedua pulau tersebut akan menjadi terisolasi. Oleh karenanya, pulau e5 harus dihubungkan dengan pulau di atasnya, c5.

G. Pencarian Solusi dengan Depth First Search

Teknik *Depth First Search* (DFS) digunakan untuk menyelesaikan sisa jembatan yang belum dibangun. Algoritma DFS akan mengembangkan hanya 1 kemungkinan arah jembatan yang dapat dibangun pada suatu pulau menjadi keadaan baru dari keadaan awal. Selanjutnya, algoritma DFS akan memajukan pencarian ke pulau berikutnya dan mengembangkan 1 kemungkinan arah jembatan pada pulau tersebut tanpa melanggar aturan permainan *Hashi*.

Bila tidak ada jembatan yang dapat dikembangkan pada suatu jalur karena semuanya melanggar aturan *Hashi*, maka metode DFS akan kembali ke jalur sebelumnya dan mengambil arah jembatan yang lain. Keadaan ini berlanjut hingga jembatan pada pulau terakhir dikembangkan.

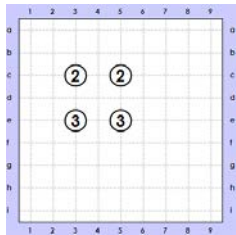
Algoritma proses DFS secara ringkas dapat dijabarkan sebagai berikut:

1. Penelusuran dimulai pada pulau-1.
2. Kembangkan satu kemungkinan arah jembatan ke kanan atau ke bawah dari pulau-1. Tempatkan kemungkinan arah

jembatan *valid* ini ke dalam satu *node* baru pada level-1, misalkan Node(1).

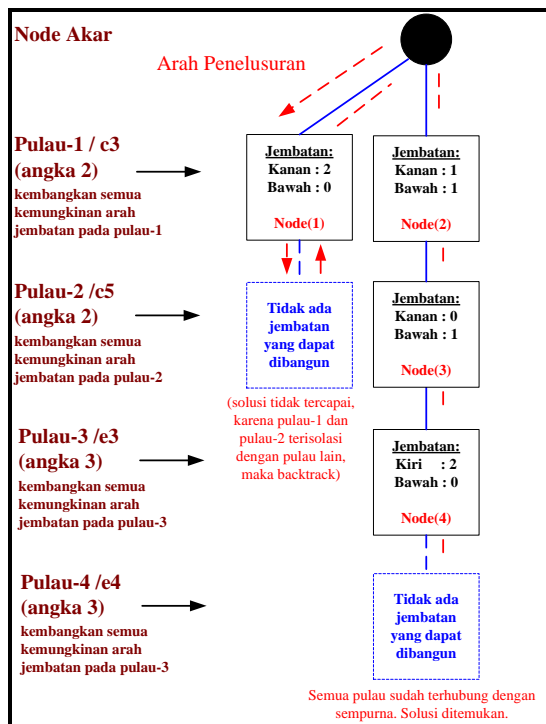
3. Penelusuran dilanjutkan pada Node(1). Kembangkan lagi satu kemungkinan arah jembatan ke kanan atau ke bawah dari pulau-2. Tempatkan kemungkinan arah jembatan *valid* ini ke dalam *node* baru pada level-2, misalkan Node(2).
4. Teruskan penelusuran yang sama pada *node* berikutnya, secara mendalam pada level-level berikutnya.
5. Apabila terdapat suatu *node* yang tidak dapat membangun jembatan karena menyalahi aturan Hashi, maka penelusuran kembali (*backtrack*) ke *node* induk dan bangun lagi jembatan lain yang berbeda dengan *node* anak yang sudah ada. Teruskan penelusuran ke *node* anak yang baru.
6. Penelusuran yang sama dilanjutkan pada *node-node* berikutnya, hingga didapatkan solusi.

Sebagai contoh sederhana, misalkan soal *Hashi* yang akan diselesaikan seperti terlihat pada gambar 9.



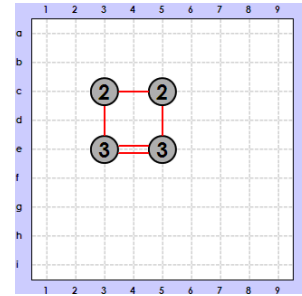
Gambar 9. Contoh Soal Hashi

Penelusuran dengan metode DFS pada Gambar 9 adalah sebagai berikut: (lihat Gambar 10)



Gambar 10. Penelusuran Metode DFS

Solusi Hashi yang ditemukan dapat dilihat pada gambar 10.



Gambar 11. Solusi Hashi dari Penelusuran DFS

IV. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut :

1. Untuk soal-soal permainan *Hashi* yang tergolong mudah, dapat diselesaikan dan ditemukan solusinya dengan hanya menggunakan teknik solving *Hashi* itu sendiri sedangkan untuk soal-soal permainan *Hashi* yang tergolong rumit digunakan teknik solving *Hashi* dan DFS.
2. DFS dapat membantu pencarian solusi permainan *Hashi* ketika terdapat jembatan yang tidak bisa ditemukan dengan menggunakan teknik solving *Hashi*.
3. Gabungan teknik solving *Hashi* dan DFS dapat membantu pencarian solusi permainan *Hashi* dengan lebih baik.

DAFTAR PUSTAKA

- [1] Anderson, Daniel. 2009. Hashiwokakero is Np-complete. Department of Computer Science, Aarhus University, Denmark, Volume 109, Issue 19, September, 2009, (<http://www.daimi.au.dk/~koda/hashiwokakero.pdf>, diakses 20 Februari 2011).
- [2] Desiani, Anita dan Muhammad Arhami. 2006. Konsep Kecerdasan Buatan. Edisi I. ANDI, Yogyakarta.
- [3] Kusumadewi, S., 2002, Artificial Intelligence (Teknik dan Aplikasinya), Edisi kedua, Graha Ilmu.
- [4] Morsink, T. 2009. Hashiwokakero, (<http://www.liacs.nl/assets/Bachelorscripties/2009-11TimoMorsink.pdf>, diakses 20 Februari 2011).
- [5] Nikoli. 2001. Hashiwokakero I. Nikoli Co., Ltd, Japan.
- [6] Yamamoto, Saito. 2008. Hashi. Terjemahan oleh : Drs. Suharsono. Mitra Media.
- [7] <http://www.conceptispuzzles.com/> (diakses 15 Februari 2011)
- [8] <http://www.indigopuzzles.com/ipuz/help.action?helpId=hashi/index> (diakses 15 Februari 2011)
- [9] <http://www-01.ibm.com/software/awdtools/rup/> (diakses 15 April 2011)
- [10] <http://www.nikoli.co.jp/> (diakses 15 Februari 2011)