

NP Complete pada Game Ending

Ditari Salsabila E. 23514004
Program Magister Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
23514004@itb.ac.id

Abstrak—Salah satu dari permasalahan NP Complete yang paling populer, 3 SAT, merupakan permasalahan yang sering digunakan untuk reduksi dan membuktikan suatu permasalahan baru termasuk ke dalam kelas NP Hard. Beberapa jenis permainan telah dibuktikan termasuk ke dalam kelas NP hard dengan menggunakan metode ini. Apabila permainan tersebut juga termasuk ke dalam kelas NP, maka secara tidak langsung metode ini membuktikan bahwa permainan tersebut termasuk ke dalam kelas NP Complete.

Makalah ini akan membahas pembuktian NP Complete pada permainan flash, Ending dengan menggunakan reduksi 3 SAT dan dengan membuktikan terlebih dahulu bahwa permainan tersebut termasuk ke dalam kelas NP

Kata Kunci — NP Complete, NP Hard, Permainan flash Ending

I. PEMBUKAAN

“Jika suatu permasalahan dapat diverifikasi kebenarannya dengan cepat, apakah permasalahan tersebut juga dapat diselesaikan dengan cepat?” [1]. Hal ini merupakan dasar pertanyaan apakah P sama dengan NP, yang merupakan salah satu dari tujuh pertanyaan yang bernilai satu juta dolar (Millenium Prize Problem) [2].

Permasalahan P dan NP ini disebutkan pada surat Gödel kepada von Neumann (yang membahas mengenai suatu permasalahan dalam kategori NP Complete) pada tahun 1956 [1]. Konsep dan istilah NP Complete sendiri baru ditemukan oleh Cook pada tahun 1971 [3].

Contoh penerapan konsep NP pada kehidupan sehari-hari yaitu pada kriptografi. Kriptografi mengandalkan ‘kesulitan’ untuk memecahkan kunci, dengan tingkat kesulitan tertinggi berupa NP Complete. Selain kriptografi, masih terdapat banyak contoh permasalahan NP Complete lainnya yang mungkin tidak disadari oleh kebanyakan orang. Beberapa jenis permainan populer telah terbukti termasuk ke dalam kelas NP-hard atau NP Complete [4][5][6].

Untuk penjelasan lebih lanjut, berikut akan dijabarkan terlebih dahulu definisi dari beberapa istilah di atas.

II. ISTILAH DAN LANDASAN TEORI

A. P dan NP

Permasalahan keputusan atau decision problem merupakan permasalahan yang hanya memiliki dua jawaban, yaitu ya (positif) atau tidak (negatif) [7]. Berdasarkan kompleksitasnya, permasalahan ini dapat berupa P atau NP.

P merupakan singkatan dari polinomial dan dapat didefinisikan sebagai “permasalahan keputusan yang jika diselesaikan dengan mesin turing deterministik akan memiliki kompleksitas waktu polinomial” [8]. Yang dimaksud polinomial yaitu jika permasalahan P memiliki sebanyak n input, maka pergerakan mesin turing tidak akan lebih dari n . Jika permasalahan dianalogikan dengan suatu program dan mesin turing dengan suatu komputer, maka waktu runtime program tersebut polinomial.

NP merupakan singkatan dari Nondeterministik Polinomial. Definisi NP secara formal yaitu “suatu permasalahan keputusan yang apabila dijalankan dengan mesin turing non-deterministik akan memiliki kompleksitas waktu polinom” [8].

Pada mesin turing nondeterministik, suatu input dapat memiliki lebih dari satu perubahan fungsi state ke state lain. Bandingkan dengan mesin turing deterministik di mana suatu input pasti akan menyebabkan perubahan dari suatu state ke state lain yang tetap. Mesin turing non deterministik deterministik memiliki kemampuan ‘menerka’ state mana yang akan diambil dari state saat ini dengan input tertentu. Tahap ini disebut tahap menerka. Selanjutnya, sama seperti pada mesin turing deterministik, mesin turing akan memeriksa apakah input tersebut diterima atau ditolak. Tahap ini disebut tahap verifikasi. Sebuah permasalahan NP yang memiliki input sebanyak n akan membutuhkan waktu verifikasi $T(n)$. Dengan kata lain sebuah permasalahan termasuk NP jika pada tahap verifikasi membutuhkan waktu polinomial.

Hal yang harus diingat mesin turing non deterministik merupakan model matematis murni. Model komputer yang ada saat ini mengacu kepada model von Neumann, yang merupakan mesin turing deterministik (atau setidaknya mendekati mesin turing deterministik, mengingat adanya batasan memori pada komputer yang tidak dapat memodelkan mesin turing dengan pita yang tidak terbatas). Sampai saat ini belum dapat dibuat mesin

yang memiliki kemampuan ‘menebak’. Pada kenyataannya permasalahan NP dalam bentuk program yang dijalankan pada komputer akan mengecek semua state yang mungkin dicapai untuk suatu fungsi, sehingga dapat menghasilkan kompleksitas waktu melebihi polinomial.

B. NP Hard dan NP Complete

NP Hard didefinisikan sebagai “persoalan yang memiliki tingkat kesulitan setidaknya sesulit NP” [9]. Definisi lainnya yaitu “suatu bahasa L dikategorikan sebagai ke dalam bahasa NP Hard jika semua bahasa yang berelasi dengan permasalahan NP dapat direduksi dalam waktu polinomial ke dalam bahasa L” [8]. Persoalan yang berelasi dengan bahasa L termasuk ke dalam NP Hard. Tidak semua persoalan yang termasuk ke dalam NP Hard berada dalam kelas NP, contohnya yaitu *halting problem*.

Suatu persoalan termasuk dalam kategori NP Complete jika persoalan tersebut termasuk ke dalam kelas NP dan juga termasuk ke dalam NP Hard [8]. Definisi NP Complete pertama kali dimunculkan oleh Cook. Cook membuktikan bahwa Satisfiability Problem (SAT) termasuk ke dalam NP Complete [3].

Untuk membuktikan bahwa suatu persoalan merupakan NP Complete dilakukan melalui dua tahap. Pertama, membuktikan bahwa persoalan tersebut termasuk ke dalam NP. Kedua, membuktikan bahwa persoalan tersebut termasuk ke dalam NP Hard. Tahap kedua dapat dilakukan melalui reduksi dari permasalahan NP Complete yang sudah diketahui ke dalam persoalan baru tersebut dalam waktu polinomial.

SAT dan salah satu hasil reduksinya yaitu 3 SAT merupakan permasalahan yang banyak digunakan untuk membuktikan bahwa suatu permasalahan termasuk ke dalam NP Complete. Penjelasan mengenai sebagai berikut.

Dimisalkan terdapat sebuah ekspresi boolean yang dapat terdiri dari beberapa klausa dengan tiap variabel bernilai true atau false dan dengan operator dapat berupa AND (\wedge), OR (\vee) dan NOT atau negasi (\neg). Satisfiability problem menanyakan “apakah terdapat pengisian nilai kebenaran untuk tiap variabel sedemikian sehingga ekspresi boolean tersebut bernilai benar” [8]. Contoh sebuah ekspresi boolean $a \wedge b$ akan menghasilkan nilai SAT positif (satisfiable), yaitu nilainya benar untuk a bernilai true dan b bernilai true. Sebaliknya ekspresi boolean $a \wedge \neg a$ memiliki jawaban SAT negatif (not satisfiable), karena ekspresi boolean tersebut tidak akan bernilai benar untuk semua nilai a yang mungkin.

3 SAT merupakan bentuk khusus dari SAT di mana ekspresi boolean terdiri dari beberapa klausa. Tiap klausa terdiri dari tepat tiga variabel (atau negasinya) yang dihubungkan dengan operator OR dan tiap klausa dihubungkan dengan operator AND. 3 SAT termasuk ke dalam NP Complete [8][10]. Contoh 3 SAT dengan 3 variabel x, y , dan z yaitu $(x \vee y \vee \neg z) \wedge (\neg x \vee y \vee \neg z) \wedge (\neg x \vee \neg y \vee z)$. Klausa pertama yaitu $(x \vee y \vee \neg z)$, $(\neg x$

$\vee \neg z)$ merupakan klausa kedua dan klausa ketiga yaitu $(\neg x \vee \neg y \vee z)$.

Contoh permasalahan lainnya yang termasuk NP Complete yaitu permasalahan keputusan untuk pewarnaan graph, lintasan Hamilton, pohon merentang minimum (minimum spanning tree), sum of subset, vertex cover, dan lain-lain [10].

III. PEMBAHASAN

A. NP Hard dan NP Complete pada Game

Selain permasalahan yang sudah disebutkan di atas, masih banyak permasalahan di dunia nyata yang termasuk ke dalam NP Hard bahkan NP Complete. Permainan seperti Minesweeper dan Lemmings memiliki kompleksitas NP Complete [4][6]. Permainan puzzle di mana pemain dapat menukar item dan harus menyusun posisi tiga item yang sama seperti Candy Crush memiliki kompleksitas NP Hard [12]. Contoh permainan lainnya yang memiliki kompleksitas serupa yaitu permainan Nintendo populer seperti Super Mario Bros, Zelda, Metroid dan Pokemon [5].

Kebanyakan pembuktian pada game ini menggunakan reduksi permasalahan 3 SAT untuk membuktikan kompleksitas NP Hard. Karena 3 SAT dapat dimodelkan dalam rangkaian logika, diperlukan beberapa *gadget* berikut :

1. *Variabel gadget*. *Variabel gadget* merupakan representasi variabel dalam 3 SAT ke dalam bentuk variabel dalam game, di mana player hanya memiliki dua pilihan. Salah satu pilihan dianggap true sedangkan pilihan lain dianggap false. Permainan seperti Sokoban atau *push game* menggunakan istilah *fork gadget* [11].

2. *Clause gadget*. *Clause gadget* merupakan representasi klausa pada 3 SAT, yaitu gabungan dari tiga *variabel gadget* dengan operator boolean OR.

3. *Wire gadget*. Jika 3 SAT dapat diibaratkan dengan rangkaian logika, maka *wire gadget* merupakan ‘kabel’ yang menghubungkan antara tiap *variabel gadget* dengan *clause gadget* yang berelasi. Contoh jika terdapat variabel x, y , dan z dan terdapat suatu klausa $(x \vee y \vee z)$ maka akan terdapat kabel yang menghubungkan masing-masing *variabel gadget* ke *clause gadget* tersebut. ‘Kabel’ ini juga menghubungkan tiap klausa, dari klausa pertama sampai terakhir untuk pengecekan kebenaran pernyataan 3 SAT. *Wire gadget* sangat bervariasi tergantung dari jenis permainannya.

Pada permainan 2D nintendo seperti Mario dan Zelda digunakan *crossover gadget*, yaitu gadget yang memungkinkan pemain hanya memilih satu arah (*unidirectional*) ketika terjadi persimpangan [5]. Hal ini untuk mencegah pemain menyimpang ke jalur lain yang tidak diambil, karena pada permainan 2D memungkinkan player bergerak ke empat arah. Jika diibaratkan pada rangkaian logika, crossover terjadi ketika dua ‘kabel’ saling berpotongan satu sama lain. Sama seperti pada rangkaian logika tentu saja nilai kebenaran tidak

boleh bertukar.

Pada permainan minesweeper, nilai boolean *true* atau *false* dibentuk berdasarkan apakah ada atau tidaknya ranjau pada suatu lokasi. Kabel kemudian dibentuk dengan menyusun lokasi ranjau sedemikian rupa sehingga ‘kabel’ menghantarkan nilai boolean tersebut. Pada minesweeper ‘kabel’ juga digunakan untuk membuat gerbang logika AND, OR, *inverter* atau negasi, dan *crossover*. Selain itu juga terdapat *phase shifter* dan *curve* (belokan di bagian ujung) [6].

Pada permainan Sokoban atau push game, di mana pemain harus mendorong balok ke tempat tertentu, digunakan *lock gadget* di mana pada lokasi persimpangan pemain hanya bisa mengakses suatu jalan keluar jika pemain masuk dari posisi tertentu (variabel tertentu) [11]. Selain itu juga digunakan gerbang logika seperti XOR dan *one-way gadget* (jalur searah selama jalur tersebut belum pernah ditempuh).

Pada permainan Lemmings, *wire gadget* terdiri dari *wire/kabel* (untuk jalur lurus), *junction/persimpangan* (sama seperti *crossover*), *corner* (sudut tepi, kurang lebih sama seperti *curve* pada minesweeper).

Tergantung jenis permainan, dapat dibutuhkan gadget tambahan selain yang disebutkan di atas, misal start dan finish gadget, seperti yang digunakan pada pembuktian permainan Super Mario Bros [5].

B. Permainan Ending

Ending merupakan permainan puzzle langkah strategi yang dibuat Aaron Steed pada tahun 2013. Permainan ini dibuat menggunakan Flash dan dapat dimainkan pada browser ataupun smartphone. Tampilan permainan minimalis dengan area menyerupai papan catur dua dimensi, tiap kotak pada papan dapat berisi simbol pemain, simbol kotak atau simbol musuh. Style permainan merupakan gabungan dari permainan seperti catur dan *roguelike*. Pemain dapat memilih dari beberapa level permainan dan terdapat opsi membuat level permainan sendiri.

Permainan berjalan sebagai berikut. Pemain bertugas menggerakkan simbol pemain hingga mencapai exit yang berada di tepi papan. Pemain dapat bergerak ke kanan, kiri, atas, dan bawah (kecuali jika ada simbol tertentu yang menghalangi). Untuk tiap satu langkah pergerakan pemain, maka simbol musuh juga akan dapat bergerak satu langkah pada saat bersamaan. Simbol pemain memiliki jumlah langkah yang terbatas (48 langkah pada awal permainan) dan akan berkurang setiap kali pergerakan. Jika pemain menghancurkan simbol musuh atau simbol kotak putih maka jumlah langkah akan bertambah. Pemain akan memenangkan suatu level jika mencapai exit dan pemain akan kalah jika dihancurkan oleh musuh, terperangkap (tidak ada pergerakan yang mungkin), atau kehabisan jumlah langkah.

Gambar 1 beberapa simbol yang digunakan pada permainan. Dari kiri ke kanan secara berurutan yaitu simbol pemain, simbol exit, simbol persegi abu-abu dan

putih. Simbol persegi sendiri merupakan item statis yang tidak bergerak pada permainan. Simbol persegi abu-abu tidak dapat disentuh, dilewati atau dihancurkan pemain, sedangkan simbol putih dapat dihancurkan pemain.



Gambar 1. Simbol utama

Gambar 2 menunjukkan salah satu simbol musuh. Untuk memudahkan, bagian yang berbentuk persegi merupakan bagian belakang musuh dan bagian garis merupakan bagian depan. Secara berurutan pada gambar yaitu simbol musuh yang menghadap ke atas, ke kanan, ke bawah dan ke kiri. Musuh ini hanya bergerak maju jika pemain berada tepat di depannya dan menghancurkannya. Musuh ini dapat dihancurkan dari sisi lain selain sisi depan.



Gambar 2. Simbol musuh

Pada permainan, simbol musuh ini dapat digabungkan dengan simbol lainnya dan membuat simbol musuh yang baru dengan sifat gabungan. Pada gambar 3, simbol musuh baru pertama dibuat dengan menggabungkan dua simbol musuh di atas, dan akan menghancurkan pemain yang berada di kanan dan bawahnya. Musuh ini hanya dapat dihancurkan dari sisi kiri dan atas.

Simbol musuh baru kedua dengan menggabungkan dua simbol musuh dan simbol persegi abu-abu. Karena persegi abu-abu tidak dapat disentuh oleh pemain, simbol musuh ini juga tidak dapat menghancurkan simbol pemain namun dapat mendorong pemain jika pemain berada di sebelah kiri atau kanan simbol. Simbol ketiga mirip dengan simbol kedua hanya saja hanya mendorong pemain yang berada di bawahnya. Untuk memudahkan, ketiga simbol baru ini diberi nama simbol 1, 2, dan 3.



Gambar 3. Variasi simbol musuh 1, 2, dan 3

Terdapat simbol musuh lainnya seperti pada gambar 4. Simbol ini hanya dapat bergerak dua arah, simbol di sebelah kiri bergerak vertikal sedangkan simbol di sebelah kanan bergerak horizontal. Simbol ini akan bergerak jika simbol pemain berada dalam jarak dekat (6 kotak) dan berusaha untuk mendekati dan menghancurkan pemain. Simbol ini juga dapat dikombinasikan dengan simbol lain seperti simbol persegi abu-abu. Hasilnya yaitu simbol musuh baru yang tidak dapat dihancurkan pemain, namun dapat bergerak vertikal atau horizontal jika pemain mendekat. Simbol baru ini disebut simbol 4 dan simbol 5 dan ditunjukkan pada gambar 5.



Gambar 4. Simbol musuh lainnya



Gambar 5. Variasi simbol musuh 4 dan 5

Masih terdapat beberapa simbol lain dan variasi gabungannya. Namun, untuk membuktikan kompleksitas pada permainan cukup dengan menggunakan simbol-simbol di atas.

IV. PEMBUKTIAN DAN IMPLEMENTASI

Untuk pembuktian, asumsikan bahwa papan/ruang pada permainan Ending dapat berukuran sangat besar (meskipun desain *custom level* papan dibatasi berukuran 11x11). Permasalahan apakah ada urutan strategi langkah yang memungkinkan pemain dapat mencapai exit pada permainan Ending merupakan NP Complete.

A. Permasalahan berada di kelas NP

Langkah pertama yaitu membuktikan bahwa permasalahan ini termasuk ke dalam kelas NP. Jika exit dapat dicapai maka diperlukan sejumlah n langkah yang dilakukan pemain untuk mencapai exit. Namun jika exit sama sekali tidak dapat dicapai oleh pemain, karena adanya batasan jumlah langkah maka pada akhirnya player akan mati. Tidak peduli apakah exit dicapai atau tidak, jumlah maksimal langkah yang ditempuh dapat diverifikasi dalam waktu polinomial. Pembuktian ini mirip dengan pembuktian pada permainan Lemming [4].

B. Permasalahan termasuk NP Hard

Untuk membuktikan bahwa permasalahan termasuk ke dalam NP hard digunakan reduksi 3 SAT ke dalam permainan Ending. Dibutuhkan beberapa gadget sebagai berikut.

Variable gadget digunakan untuk memodelkan variabel. Namun, bukan suatu variabel yang diberikan nilai true atau false, melainkan diberikan dua pilihan yaitu suatu variabel dan negasinya. Jika suatu variabel dipilih oleh pemain, maka variabel tersebut akan bernilai true dan negasi variabel tersebut tidak bisa dipilih. Demikian sebaliknya jika negasi suatu variabel yang dipilih, maka negasi variabel tersebut bernilai true dan variabel tidak bisa dipilih.

Wire gadget merupakan gadget yang menghubungkan antara variabel dengan klausa, juga antara satu klausa dengan lainnya. Termasuk di dalamnya yaitu *crossover gadget*, yang memastikan bahwa tidak akan terjadi pertukaran jalur logika jika terjadi perseberangan antara kabel. *Crossover gadget* dirancang sedemikian rupa sehingga jalur hanya dapat ditempuh vertikal saja atau horizontal saja (kecuali jika jalur tersebut sudah pernah ditempuh sebelumnya, dalam hal ini, berarti variabel atau negasi variabel yang terkait dengan jalur tersebut bernilai true).

Clause gadget digunakan untuk memodelkan klausa. Tiap *clause gadget* terhubung dengan tiga *variable gadget*, dan akan bernilai true jika salah satu dari ketiganya bernilai true. Terdapat *wire* yang menghubungkan *clause gadget* ke *finish gadget* yang bisa

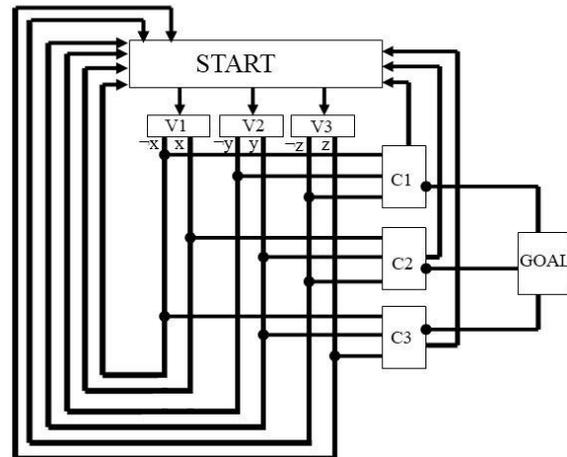
diakses hanya jika *clause gadget* tersebut bernilai true.

Finish gadget memiliki simbol exit yang harus dicapai pemain. *Finish gadget* dimodifikasi sedemikian rupa sehingga exit hanya mungkin dicapai jika semua *clause gadget* bernilai true.

Contoh implementasi dengan tiga variabel (dan negasinya) serta tiga klausa dapat dilihat pada gambar 6. V1, V2, dan V3 merupakan *variable gadget*. C1, C2, C3 merupakan *clause gadget*. Goal merupakan *finish gadget*. *Wire gadget* ditunjukkan dengan garis, sedangkan *crossover gadget* merupakan perempatan pada wire. Gambar tanda panah menunjukkan jalur searah, namun jalur yang sudah dikunjungi dapat dikunjungi kembali, pengecualian untuk *variable gadget* hanya dapat dikunjungi sekali.

Pemain mulai dari start dan memilih pilihan pada *variable gadget*. Pemain kemudian akan menempuh semua jalur ke klausa yang mungkin jika ada. Jika tidak ada jalur, maka pemain harus kembali ke start. Dari klausa pemain dapat menuju ke *finish gadget* (berarti klausa tersebut bernilai true) atau kembali ke start untuk memilih variabel berikutnya. Meskipun pemain sudah mencapai *finish gadget*, exit hanya dapat dicapai setelah jalur menuju ke semua klausa dikunjungi dan bernilai true.

Ekspresi boolean untuk 3 SAT yaitu $(\neg x \vee \neg y \vee \neg z) \wedge (x \vee y \vee \neg z) \wedge (\neg x \vee y \vee z)$. Jika terdapat langkah strategi sehingga exit dapat dicapai pada permainan Ending, maka 3 SAT bernilai positif (*satisfiable*). Jika tidak maka 3 SAT tidak *satisfiable*.



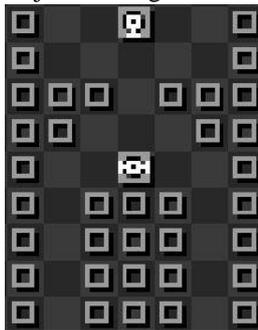
Gambar 6. Framework

Implementasi dilakukan sebagai berikut :

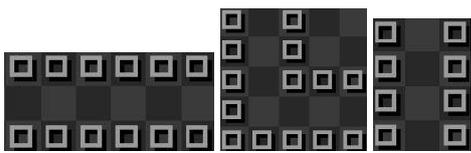
Variabel gadget ditunjukkan pada gambar 7. Pemain berjalan dari atas dan simbol 3 akan mendorong pemain sehingga jalur terblokir dan tidak bisa dikunjungi kembali. Berikutnya, pemain harus memilih salah satu dari dua jalur di bawah. Kedua jalan merupakan representasi suatu variabel dan negasi variabel. Jika Pemain sudah memilih salah satu jalan, maka simbol 2 akan mendorong pemain dan memblokir jalur, sehingga pemain tidak dapat memilih negasi atau variabel yang tidak dipilih.

Wire gadget untuk jalur horizontal, vertikal, dan belokan hanyalah jalan yang dibatasi dengan simbol

persegi abu-abu yang tidak bisa dilewati pemain seperti pada gambar 8. Karena pada permainan ending jumlah langkah dibatasi, untuk mencegah pemain kehabisan langkah dapat ditaruh simbol kotak putih di tiap jarak tertentu pada jalur. Pemain dapat menghancurkan kotak putih dan menambah jumlah langkah.



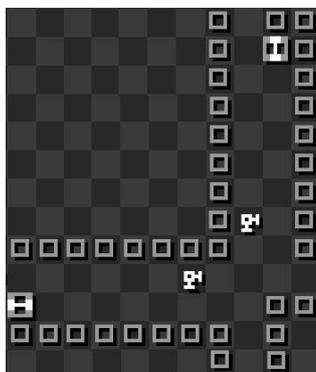
Gambar 7. Variable gadget



Gambar 8. Wire (a) horizontal (b)belokan (c)vertikal

Untuk *crossover gadget* ditunjukkan pada gambar 9. Pemain dapat menempuh jalur dari atas atau dari kiri. Namun, pemain dari jalur atas hanya dapat bergerak vertikal ke bawah setelah menghancurkan simbol 1 dari atas. Pemain tidak bisa bergerak horizontal ke kanan karena dihalangi oleh simbol 4 yang bergerak mengikuti pemain, atau bergerak ke kiri karena terdapat simbol 1.

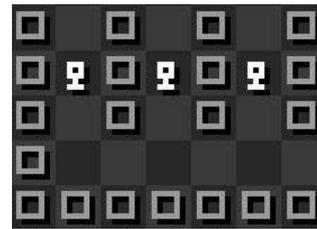
Demikian sebaliknya, pemain yang menempuh jalur dari kiri hanya dapat bergerak ke kanan setelah menghancurkan simbol 1 dari kiri. Simbol 5 menghalangi pemain bergerak ke bawah dan simbol 1 di atas mencegah pemain bergerak ke atas. Dengan kata lain, jalur vertikal hanya dapat ditempuh vertikal, demikian juga dengan jalur horizontal kecuali jika jalur tersebut sudah pernah dikunjungi sebelumnya (variabel atau klausa bernilai true).



Gambar 9. Crossover gadget

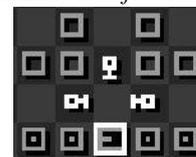
Clause gadget ditunjukkan pada gambar 10. Terdapat tiga jalur untuk mencapai *clause gadget*. Masing-masing merupakan jalur dari *variable gadget* yang berbeda. Jalur manapun yang ditempuh akan memungkinkan clause

gadget bernilai true. Hal ini merupakan representasi klausa pada 3 SAT yang akan bernilai true jika salah satu dari variabel klausa bernilai true. Agar clause gadget bernilai true, terdapat jalur menuju *finish gadget* yang dihalangi oleh simbol musuh. Pemain harus menghancurkan simbol musuh tersebut.



Gambar 9. Clause gadget

Pada *finish gadget* terdapat beberapa jalur menuju ke *exit* yang dihalangi oleh simbol musuh. Sebagai contoh pada gambar, terdapat tiga jalur yang dapat ditempuh dari tiga clause gadget berbeda. Exit hanya bisa dicapai jika semua musuh telah dihancurkan dan tiap-tiap musuh hanya dapat dihancurkan dari jalur *clause gadget* tertentu. Pemain hanya dapat mencapai exit jika semua *clause gadget* bernilai true. Hal ini merupakan representasi 3 SAT yang hanya akan bernilai true jika semua klausa bernilai true. Dengan kata lain, pemain hanya akan dapat mencapai exit jika dan hanya jika permasalahan 3 SAT yang berelasi bernilai true/ *satisfiable*.



Gambar 10. Finish gadget

V. KESIMPULAN

Dari pembuktian di atas, dapat disimpulkan bahwa permasalahan apakah terdapat langkah strategi yang memungkinkan pemain mencapai exit pada permainan Ending merupakan NP Complete. Karena strategi untuk mencapai exit merupakan objektif utama permainan, dapat dikatakan permainan ini memiliki tingkat kesulitan yang 'sulit' dan menantang bagi pemain.

Selain Ending, masih banyak permainan (dan permasalahan umum) lainnya di kehidupan sehari-hari yang memiliki kompleksitas NP hard atau bahkan NP complete. Permasalahan seperti ini termasuk dalam kategori 'sulit untuk diselesaikan'. Reduksi permasalahan 3 SAT merupakan salah satu cara yang dapat digunakan untuk membuktikan bahwa permasalahan termasuk ke dalam kelas NP hard (dan mungkin kelas NP Complete).

DAFTAR PUSTAKA

- [1] Juris Hartmanis. "Godel, Von Neumann, and the P = NP problem", berada di *Bulletin of the European Association for Theoretical Computer Science*, volume 38, 1989.
- [2] Carlson et al, "The Millenium Prize Problems", Providence, RI: American Mathematical Society dan Clay Mathematics Institute, 2006

- [3] Stephen A Cook, "The Complexity of Theorem-Proving Procedures", berada di prosiding *3rd Annual ACM Symposium on Theory of Computing*, 1971.
- [4] Cormode, Graham. (2004). *The hardness of the lemmings game, or Oh no, more NP-completeness proofs* [pdf]. Tersedia: <http://dimacs.rutgers.edu/~graham/pubs/papers/cormodelemmings.pdf>. File : cormodelemmings.pdf.
- [5] G. Aloupis et all. (2012). *Classic Nintendo Games are (NP-)Hard* [pdf]. Tersedia: <http://arxiv.org/abs/1203.1895>. File : 1203.1895v2.pdf.
- [6] Michiel de Bondt. (2012). *The computational complexity of Minesweeper* [pdf]. Sumber: <http://arxiv.org/abs/1204.4659>. File: 1204.4659v1.pdf
- [7] A. Church, *A Note on the Entscheidungsproblem*, J. Symb. Logic **1**, 1936.
- [8] Hopcroft et all, *Introduction to Automata Theory, Languages, and Computation (3rd Edition)*, Addison-Wesley, 2006.
- [9] Jan Van Leeuwen, *Handbook of Theoretical Computer Science, vol. A : Algorithms and Complexity*, Amsterdam : Elsevier, 1994.
- [10] Michael dan David S, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H.Freeman, 1979.
- [11] Demaine et all, "PushPush and Push-1 are NP-hard in 2D", berada di prosiding *12th Annual Canadian Conference on Computational Geometry*, 2000.
- [12] Gualà et all. (2014). *Bejeweled, Candy Crush and other Match-Three Games are (NP-) Hard* [pdf]. Tersedia: <http://arxiv.org/abs/1403.5830>. File: 1403.5830v1.pdf

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Desember 2014

Ditari Salsabila Esperanti
23514004