

Penerapan Graf dalam Menentukan Strategi Permainan

Nugroho Satrijandi - 13510032
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
nugroho.s@students.itb.ac.id

Abstrak —Permainan merupakan salah satu sarana dalam menghilangkan kejenuhan permainnya. Jika kita berbicara tentang permainan, banyak jenis-jenis permainan, di antaranya permainan yang bertipe pertualangan, *arcade*, strategi, dan lainnya. Dalam suatu permainan komputer, banyak permainan yang membutuhkan strategi dalam memainkannya, sebagai contoh dalam menentukan *skill build* yang akan diambil, sampai pada *quest system* yang mengharuskan pemainnya untuk memiliki strategi dalam bermain. Untuk itu penulis mencoba untuk menerapkan pohon keputusan (*decision tree*) yang telah dipelajari dalam *game strategy*. Algoritma Topological Sort dalam suatu graf dapat membantu pemain dalam mengambil keputusan. Dalam hal ini, keputusan yang diambil adalah menentukan *game strategy* pada beberapa *game*. Pada makalah ini penulis akan menyajikan berbagai macam penerapan Topological Sort suatu graf dalam membuat keputusan dalam beberapa *game*. Selain itu penulis juga akan menjelaskan tentang kelebihan dan kekurangan dari teori graf pada penyusunan strategi juga peranannya dalam bidang keinformatikaan.

Kata Kunci —Topological Sort, Game Strategy, Graf, Skill Build, Quest system

I. PENDAHULUAN

Dewasa ini, hampir setiap orang bermain *game*, baik permainan komputer maupun permainan lainnya. Perkembangan dari suatu permainan yang ada sangatlah cepat, khususnya, permainan dengan media elektronik, sehingga menghasilkan banyak jenis-jenis permainan. Para pemain *game* memiliki banyak alasan untuk bermain *game*. Ada yang bermain untuk menghilangkan stress atau kejenuhan. Ada juga yang bermain untuk melatih strategi dan kemampuan berpikir, dan masih banyak lagi alasan-alasan lain.

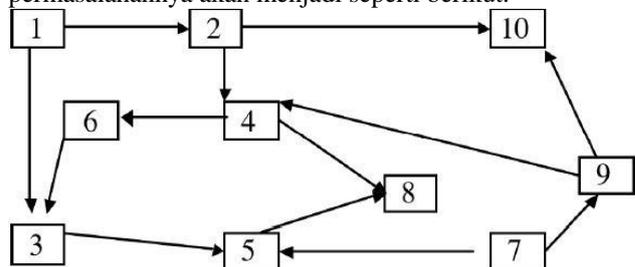
Dalam setiap *game*, pemain yang baik akan membuat suatu strategi untuk dapat memenangkan atau dapat memainkan *game* tersebut dengan baik. Suatu permainan membutuhkan suatu strategi permainan tertentu, bahkan permainan yang sederhana apapun memerlukan strategi. Sebagai contohnya dalam *game* buatan Rovio yang dalam permainannya hanya memerlukan satu buah jari dalam memainkannya, memerlukan strategi. Contoh lain pada *game* yang ber-genre pertualangan, dibutuhkan strategi untuk mengambil *quest* yang diperlukan sebagai prasyarat dalam mengambil *quest* lain. Juga dalam menentukan *skill* apa yang diperlukan sebagai prasyarat dari *skill* lain.

Jika kita memiliki strategi permainan yang baik kemungkinan kita akan dapat memainkan permainan tersebut dengan baik bahkan jika diadu dengan pemain lain, kita dapat memenangkan permainan tersebut. Untuk itu penulis akan mencoba menguraikan sedikit tentang algoritma *topological sort* pada suatu graf dan penerapannya dalam strategi suatu permainan.

II. PEMBAHASAN

A. Graf

Suatu graf digunakan untuk merepresentasikan objek-objek disrit dan hubungan antara objek-objek tersebut. Suatu graf dapat direpresentasikan dengan menyatakan objek sebagai noktah, bulatan, atau titik, sedangkan hubungan antar objek dinyatakan dengan garis. Graf didefinisikan msebagai pasangan himpunan simpul yang tidak kosong dan himpunan sisi. Suatu graf dapat digunakan untuk menyelesaikan berbagai macam masalah. Pada makalah ini akan dibahas penerapan graf dalam menentukan strategi permainan dengan menggunakan algoritma *topological sort*. Permasalahan yang muncul dari suatu *topological sort* ini yaitu diberikan suatu urutan partial dari elemen suatu himpunan dan dikehendaki agar elemen tersebut memiliki keterurutan linier. Jika direpresentasikan dalam suatu graf berarah, permasalahannya akan menjadi seperti berikut.



Graf 1. Graf berarah

Dalam suatu permainan, setiap simpul menyatakan suatu *skill* karakter atau *quest-quest* yang tersedia. Sebagai contoh simpul 4, jika dimisalkan sebagai sebuah *quest* dari suatu permainan. Maka 2 dan 9 menyatakan sebuah *quest* pre-requisit dari quest nomor 4. Sedangkan 4 merupakan *quest* pre-requisit dari quest nomor 6 dan nomor 8. Untuk menyelesaikan masalah tersebut dibutuhkan suatu algoritma yang digunakan untuk menyelesaikan permasalahan tersebut menjadi suatu

strategi dalam suatu permainan. Solusi dari graf berarah itu menjadi sebuah elemen yang memiliki graf yang memiliki keterurutan linier. Berikut beberapa solusi dari penyelesaian graf berarah tersebut $1 \rightarrow 2 \rightarrow 7 \rightarrow 9 \rightarrow 4 \rightarrow 6 \rightarrow 3 \rightarrow 5 \rightarrow 8 \rightarrow 10$ atau bisa juga $7 \rightarrow 9 \rightarrow 1 \rightarrow 2 \rightarrow 4 \rightarrow 8 \rightarrow 6 \rightarrow 3 \rightarrow 5 \rightarrow 10$.

Ini menyebabkan dalam pengambilan keputusan terdapat beberapa solusi yang ada. Sehingga untuk mengambil keputusan pemain dapat menggunakan berbagai macam strategi.

B. Algoritma Topological Sort

Topological sort merupakan suatu keterurutan dari semua simpul dalam DAG (*directed acyclic graph*). *Topological sort* ini berguna untuk mengurutkan sesuatu yang harus dikerjakan terlebih dahulu. Ide dari *topological sort* ini adalah membuat suatu list linier yang memiliki keterurutan dengan cara memilih salah satu simpul yang tidak memiliki pre-requisit atau dengan kata lain memilih simpul yang tidak memiliki predesesor. Jika tidak ada, maka graf tersebut tidak memiliki solusi. Selanjutnya penghapusan simpul yang tidak memiliki predesesor tersebut ke dalam suatu list, dan menghapus hubungan antara simpul tersebut dengan simpul yang lain. Dan proses tersebut terus berulang hingga semua simpul telah terhapus. Setelah semua elemen terhapus, urutan penghapusan dari elemen-elemennya akan menjadi suatu list linier yang memiliki keterurutan.

III. PENERAPAN ALGORITMA TOPOLOGICAL SORT DALAM STRATEGI PERMAINAN

A. Game Strategy

Suatu *game* biasanya memiliki suatu cara/strategi untuk dapat memenangkan atau untuk dapat memainkannya dengan baik. Biasanya game-game tersebut menyediakan *skill-skill* atau *quest-quest* yang dapat diambil. Ini membutuhkan kemampuan khusus dari pemain untuk dapat memilih strategi yang akan diambil dalam memainkannya.

Beberapa permainan menggunakan sistem quest pre-requisit dalam sistem permainannya. Untuk mengambil quest dibutuhkan suatu quest pre-requisit untuk dapat mengambilnya. Pemain yang mahir akan terlebih dahulu merencanakan strategi dalam permainan. Seperti contohnya, *skill* apa saja yang dibutuhkan atau *quest* apa saja yang dibutuhkan untuk mengambil *skill/quest/job* yang lain dan untuk menentukan *skill* apa saja yang tidak perlu diambil. Untuk itu perlu adanya strategi untuk menentukan/mengurutkan setiap *skill/quest/job* yang ada menjadi sebuah list linier dengan bantuan *topological sort*.

Berikut adalah beberapa contoh dari graf berarah dalam

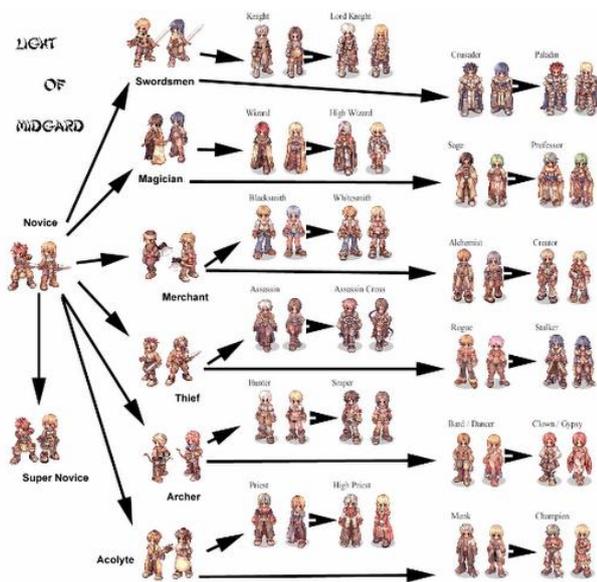
suatu permainan yang membutuhkan penyelesaian dengan algoritma *topological sort*. Berikut adalah gambar *skill tree* yang digunakan pada *game Dragon Nest Online*.



Gambar 1. Skill Tree pada Dragon Nest Online

Pada contoh game diatas, user telah diperhadapkan pada suatu *skill simulator*, untuk dapat memilih *skill* apa saja yang ingin diambil. Namun ada beberapa *game* yang tidak menyediakan dalam bentuk graf, melainkan hanya menggunakan penjelasan di deskripsi pada *skill/quest/job* yang ada. Sehingga untuk mempermudah pemain dapat membuat graf sendiri untuk memudahkan dalam penyusunan strategi.

Strategi permainan lainnya yang digunakan, kebanyakan pada *game* yang memiliki banyak *job* dalam permainannya. Seorang pemain harus memilih salah satu *job* yang dapat berubah ke *job* lainnya. *Job tree* dimasukkan dalam strategi permainan, karena awalnya pemain harus menentukan dahulu *job* yang diinginkannya nanti (baik 2nd *job* ataupun 3rd *job*) sebelum memutuskan. Karena kesalahan *job* awal akan membuat pemain mengulangi permainan yang ada, karena tidak memutuskan dengan matang. Untuk lebih jelasnya, berikut akan disajikan gambar *job tree* dari sebuah *game online*.



Gambar 2. Job Tree dalam Ragnarok Online

B. Penerapan Graf dalam Game

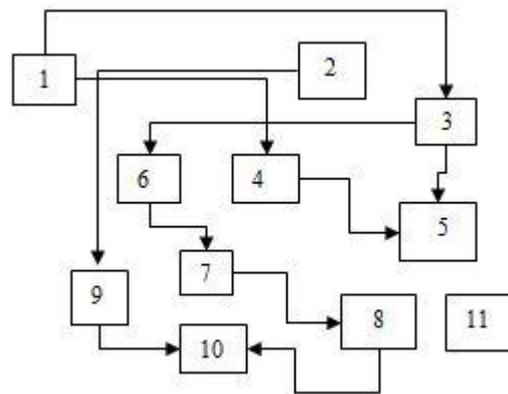
Graf memiliki banyak peranan dalam suatu permainan, seperti menentukan jalur terdekat dari suatu tempat ke tempat lain pada permainan yang bertipe pertualangan, agar pemain tidak bolak-balik. Namun pada makalah ini penulis akan mengkhususkan penerapan graf tersebut dalam penentuan strategi yang ada. Misalnya dalam suatu game, tidak tersedia fitur seperti pada Gambar 1 sehingga membuat pemain kebingungan dalam menentukan quest/skill apa yang perlu untuk diambil terlebih dahulu sebelum mengambil quest/skill yang diinginkan. Untuk itu pemain dapat membuat terlebih dahulu graf berarah, dengan simpul-simpul menandakan skill/quest/job yang tersedia. Dan setiap simpul dapat terhubung dengan simpul lain atau juga bisa simpul tersebut tidak memiliki sisi. Jika memiliki sisi, maka sisi yang berarah itu menunjukkan bahwa ada ketergantungan antara simpul yang satu dengan simpul yang lain. Misalnya, 1→2, berarti 1 merupakan pre-requisit dari 2. Jika tidak memiliki sisi, maka itu menandakan simpul tersebut tidak membutuhkan syarat apapun dan bukan merupakan syarat bagi simpul lain.

Sebagai contoh, tersedia berbagai macam quest sebagai berikut dilengkapi dengan deskripsinya masing-masing.

- Quest 1
Pre-Requisit : -
- Quest 2
Pre-Requisit : -
- Quest 3
Pre-Requisit : Quest 1
- Quest 4
Pre-Requisit : Quest 1
- Quest 5
Pre-Requisit : Quest 3 dan Quest 4
- Quest 6
Pre-Requisit : Quest 3
- Quest 7

- Pre-Requisit : Quest 6
- Quest 8
Pre-Requisit : Quest 7
- Quest 9
Pre-Requisit : Quest 2
- Quest 10
Pre-Requisit : Quest 8 dan Quest 9
- Quest 11
Pre-Requisit : -

Dari quest data tersebut kita dapat menggambarkan grafnya sebagai berikut.



Graf 2. Graf hasil dari data yang ada

Setelah kita membuat graf, maka kita dapat membuat sebuah list linier dari bentuk graf tersebut, menjadi seperti ini.

$$1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 2 \rightarrow 10 \rightarrow 11$$

Dari graf tersebut, Quest 11 merupakan simpul terpendek yang merupakan quest yang bukan merupakan syarat dari quest yang lain, dan juga tidak memiliki pre-requisit lain. Namun dalam sebuah permainan atau game yang memanfaatkan NPC sebagai media pemberi quest, quest yang diperoleh terkadang memiliki pre-requisit yang bukan merupakan suatu quest yang diperoleh dari NPC, melainkan pre-requisit lain, maka pre-requisit tersebut juga dapat didefinisikan sebagai suatu simpul. Misalnya untuk melanjutkan ke Quest 10, pemain harus mengumpulkan uang sebanyak Rp. 10.000,-. Sehingga pre-requisit tersebut dapat ditambahkan menjadi suatu simpul baru.

Akan tetapi pertanyaannya, jika hanya semudah itu apakah harus untuk mendesain sebuah graf untuk mendesain suatu game strategy? Hal tersebut merupakan contoh sederhananya, namun dalam kenyataannya, pembuat game membuat sistem game tidak sesederhana itu, khususnya game-game yang memiliki kompleksitas yang tinggi. Sehingga memiliki banyak sekali quest-quest yang bisa mencapai ratusan bahkan lebih. Permasalahan yang sesungguhnya adalah tergantung gaya permainan sang pemain. Jika pemain tersebut hanya untuk bermain tanpa menggunakan strategi, tentu pemain tersebut tidak perlu memanfaatkan teori graf dengan algoritma topological sort ini untuk bermain.

Namun jika, pemain tersebut ingin menyelesaikan permainan dengan cepat saat permainan berbasis waktu misalnya tanpa harus bolak-balik dari NPC yang satu ke NPC yang lain untuk mendapatkan *quest*, maka pemain tersebut harus menggunakan teori graf dengan *topological sort* ini sebagai basisnya.

Pada *game-game* yang sudah dimainkan oleh banyak orang, banyak orang telah menyediakan *walkthrough* game tersebut untuk membantu menyelesaikan *quest*, dan juga terkadang tersedia *Skill Tree Simulator* sehingga pemain tidak akan salah dalam memilih *skill* yang ada.

Contoh *game* lain yang menggunakan penerapan graf dalam penentuan *game strategy* yaitu Defence of the Ancient(DotA). Permainan yang cukup populer, yang mengandalkan strategi dalam *gameplay*-nya. *Gameplay* diambil dari Wikipedia, merupakan suatu cara spesifik di mana pemain berinteraksi dengan permainan, *Gameplay* juga merupakan suatu pola dari game mengenai peraturan-peraturan yang ada dan juga fitur-fitur yang tersedia dalam sebuah game. Dalam permainan DotA ini pemain dihadapkan untuk membentuk *hero* yang ada dengan *skill build* sesuai hati pemainnya. Namun jika pemilihan *skill*-nya tidak tepat, maka kemungkinan besar pemain tersebut akan kalah dalam permainan tersebut.

Ada juga game yang menggunakan *combo system* dalam permainan, seperti pada *game Seal Online*. Agar suatu *combo* dapat dihasilkan, maka pemain harus dapat melangsungkan suatu aksi yang memiliki keterurutan. Seperti misalnya menekan tombol pada keyboard yang memiliki keteraturan.

Berikut adalah beberapa contoh *combo* yang ada dalam *game* tersebut.

- 15 combo (A S D A S D A S D A S D A S D): Serangan Pedang Cahaya
- 16 combo (S S S A A A S S S D D D A S D AS): Rataan Tipareth Sang Kecantikan
- 18 combo (D A S D A S D A A S S D D A S D SD): Pedang Dewa Membelah Angkasa
- 19 combo (A A A S S S D D D A S D S A A A A): 7 Titah Penghancur Langit ke 9
- 21 combo (D S A S D S A S D S A S D S A S D D D D): 10 Elim Turun ke Shiltz

Pada *combo* tersebut digunakan suatu keterurutan list linier, dimana pemain harus menekan tombol tersebut secara berurutan, sehingga menghasilkan *combo* yang baik.

C. Kelebihan dan Kekurangan dari Penggunaan Teori Graf dalam Penentuan Strategi

Dalam penggunaan graf sebagai sarana untuk menentukan suatu strategi permainan, penggunaannya memiliki kelebihan dan kekurangan. Berikut disajikan kelebihan dan kekurangan dari penggunaan graf dalam penentuan strategi permainan.

<i>Kelebihan</i>	<i>Kekurangan</i>
Cocok untuk pemain yang menginginkan dalam permainan cepat.	Tidak cocok jika digunakan untuk game yang memiliki kompleksitas yang sangat rendah atau sangat tinggi.
Tidak membuat pemain untuk mondar-mandir dari suatu tempat ke tempat lain.	Harus mencari data/informasi permainan tersebut terlebih dahulu.
Membuat pemain tidak salah dalam mengambil keputusan dalam menentukan strategi permainan.	Karena menghasilkan banyak solusi, <i>topological sort</i> ini menghasilkan banyak strategi dalam suatu permainan, sehingga keunggulan strategi tergantung dari pemain.
Memudahkan pemain dalam melakukan suatu aksi yang memiliki keteraturan sehingga menghasilkan suatu <i>combo</i> .	

Tabel 1. Kelebihan dan Kekurangan Penerapan Teori Graf

D. Peranan dalam Bidang Keinformatikaan

Apakah hal itu begitu penting dan apa manfaatnya dengan bidang ilmu keinformatikaan? Ya, itu merupakan hal yang penting dalam pembuatan *game*. Dalam pembuatan *game*, graf merupakan salah satu teori yang paling banyak dipakai untuk merancang terlebih dahulu *gameplay* dari suatu game. Jika tidak dirancang terlebih dahulu dengan suatu graf, maka *gameplay* dari game tersebut akan kacau.

Dalam pembuatan *game*, hendaknya *game creator* membuat *skill tree* seperti pada Gambar 1, agar memudahkan pemain tidak salah dalam menentukan skill yang diinginkan.

Kebanyakan pemain *game*, khususnya *game online* merasa jenuh akan permainan yang membosankan, akan lebih baik jika *game creator* melakukan update sesekali, seperti menambah jumlah daun(*job*) pada pohon *job*. Hal ini akan membuat pemain akan penasaran dengan *job* baru ini, sehingga akan membuat pemain tidak bosan dan ingin mencoba *job* baru tersebut.

Selain itu, dalam pembuatan *game* juga perlu diperhatikan bahwa kebanyakan pemain *game* tidak suka dengan *game* yang *walkthrough*-nya yang terlalu bertele-tele, sehingga membuat pemain harus mondar-mandir dari satu tempat ke tempat yang lain misalnya dalam *game* ber-genre *MMORPG*.

VI. KESIMPULAN

Di akhir penulisan makalah ini, penulis mendapatkan

beberapa kesimpulan, di antaranya.

1. *Topological sort* pada *directed acyclic graph* dapat dimanfaatkan dalam menentukan suatu strategi permainan.
2. Salah satu penerapan teori graf pada penentuan strategi permainan dapat diterapkan pada semua permainan, khususnya permainan yang menyediakan fitur untuk menaikkan *skill* atau menyelesaikan *quest* yang ada dengan cara membuat terlebih dahulu DAG-nya, selanjutnya dibuat list linier yang telah terurut dengan *topological sort*.
3. Penerapan teori graf ini tergantung dari penggunaannya dalam permainan.
4. Dalam penerapannya dalam penentuan strategi permainan, teori graf ini memiliki kelebihan dan kekurangan.
5. Di bidang informatika, penerapan graf ini berguna bagi *game creator* dalam menentukan *gameplay* dari *game* yang akan dibuat.

REFERENSI

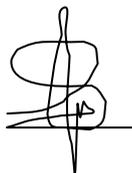
- [1] <http://homepages.ius.edu/rwisman/C455/html/notes/Chapter22/TopSort.htm>, diakses tanggal 10 Desember 2011
- [2] <http://dragonnestblog.com/2011/08/01/2nd-job-skill-guides-from-korea/>, diakses tanggal 10 Desember 2011
- [3] <http://gamergroundz.blogspot.com/2009/12/ragnarok-job-tree.html>, diakses tanggal 10 Desember 2011
- [4] Munir, Rinaldi. 2008. Struktur Diskrit. Bandung. Informatika

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, 29 April 2010

ttd



Nugroho Satrijandi - 13510032