

Penerapan Algoritma Greedy dalam Permainan MarketGlory

Erwin / 13511065

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13511065@stei.itb.ac.id

Abstract—Permainan MarketGlory adalah permainan berbasis *web-browser* bertipe strategi ekonomi. Pemain dapat bekerja, bertarung, mendirikan perusahaan, menjual barang dagangan, mencalonkan diri sebagai anggota legislatif, membuat organisasi, bahkan menyerang negara lain untuk menjajah negara tersebut. Aspek penting dalam permainan ini ada 2, yaitu *currency* dan *productivity*. *Currency* menunjukkan jumlah kekayaan peserta dalam 3 mata uang yaitu *Euro*, *Gold*, dan mata uang lokal. *Productivity* menyatakan jumlah produktivitas yang dimiliki oleh pemain. *Productivity* dihitung berdasarkan 3 aspek yaitu *Energy*, *Experience*, dan *Knowledge*. Jumlah uang yang didapat oleh pemain kebanyakan dihitung berdasarkan persentase *productivity* yang dimiliki oleh pemain. Terdapat banyak sekali strategi dalam memainkan permainan ini. Tanpa strategi yang jelas, pemain malah akan rugi. Makalah ini akan membahas dan membandingkan beberapa jenis *starter* dalam bermain permainan tersebut dengan algoritma Greedy.

Index Terms—MarketGlory, Fight, Energy

I. PENDAHULUAN

MarketGlory adalah sebuah permainan *browser* yang menekankan banyak aspek seperti ekonomi, politik, social, dan militer. Setiap pemain di MarketGlory mempunyai kesempatan untuk : bekerja, bertarung, membangun perusahaan, aktif di pemerintahan, mendapat peringkat militer dan membangun organisasi sendiri. Jalan untuk sukses di MarketGlory diperoleh melalui strategi pribadi oleh setiap pemain. Beberapa pemain akan mencapai tujuan lebih cepat dari yang lain tergantung taktik bermain. Sukses adalah hasil dari tindakan dan tekad bermain.

Pada saat awal permainan dimulai, *Energy*, *Experience*, dan *Knowledge* adalah bernilai 1.00. *Currency* yang ada juga semuanya bernilai 0. Pemain dapat melakukan *Fight* 10 kali/hari dan *Work* 1 kali/hari untuk mendapatkan uang (dalam hal ini IDR).

Perhitungan uang yang didapat dari *Fight* :

$$\text{Laba} = \text{Energy}/100 \times \text{FightValue},$$

Perhitungan uang yang didapat dari *Work* :

$$\text{Laba} = \text{Energy}/3000 \times \text{WorkValue}$$

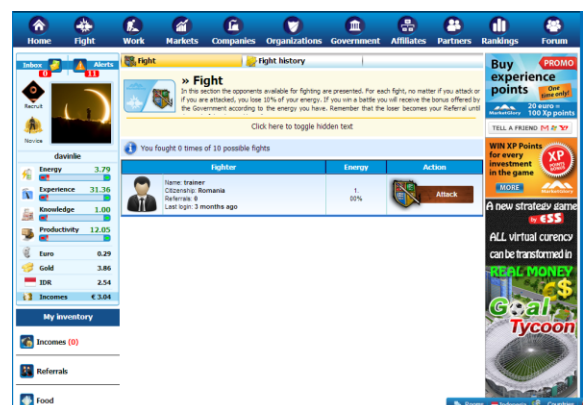
Fight akan menghabiskan 10% *Energy* sedangkan *Work* akan menghabiskan 50% *Energy*. *Energy* minimal adalah 1.00 dan tidak bisa lebih rendah dari nilai tersebut.



Gambar 1. Awal login

Energy didapat melalui pembelian koran. Setiap pembelian koran (@*ArticleValue*) akan mendapatkan 0.3 *Energy* dengan maksimal 10 kali pembelian koran per hari (3 *Energy*).

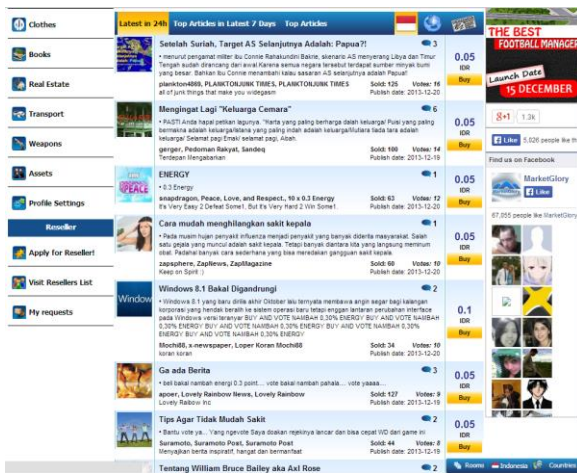
Pengelolaan *Energy* sangat penting dalam permainan tersebut karena nilai uang yang didapat dari *Fight* dan *Work* bergantung pada tingkat *Energy* pemain.



Gambar 2. Fight



Gambar 3. Work



Gambar 4. Koran

Asumsi-asumsi yang digunakan :

Jenis	Nilai
FightValue	8
WorkValue	2400
ArticleValue	0.05

II. DASAR TEORI

Algoritma Greedy adalah salah satu algoritma yang banyak digunakan untuk memecahkan persoalan optimasi. Persoalan optimasi terbagi menjadi 2, yaitu minimasi dan maksimasi. Minimasi berkaitan dengan cara agar solusi yang dihasilkan seminimal mungkin (misalnya dalam persoalan pemilihan uang untuk membayar). Maksimasi berkaitan dengan cara agar hasil yang didapat semaksimal mungkin (misalnya dalam pengerjaan barang dengan profit dan batas waktu tertentu).

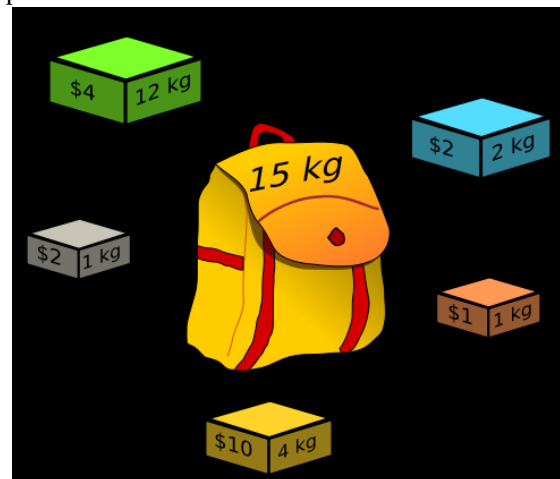
Prinsip dari algoritma Greedy adalah "take what you can get now" yang berarti ambillah apa pun yang dapat kamu ambil sekarang. Algoritma greedy akan mengambil solusi tahap per tahap berdasarkan solusi optimum tahap itu. Dengan solusi optimum per tahap, algoritma ini diharapkan dapat memperoleh solusi optimum secara

keseluruhan.

Ada 5 elemen yang penting dalam algoritma Greedy, yaitu :

1. Himpunan Kandidat (C)
Himpunan kandidat berisi semua entitas yang dapat menjadi solusi dari permasalahan
2. Himpunan Solusi (S)
Himpunan solusi adalah kumpulan entitas yang akan menjadi solusi dari permasalahan.
3. Fungsi Seleksi
Fungsi seleksi adalah fungsi yang digunakan untuk memilih sebuah solusi optimum.
4. Fungsi Kelayakan
Fungsi ini menjamin agar setiap solusi yang dipilih adalah sah. Sah mempunyai arti bahwa solusi tersebut tidak melanggar konstrain yang telah ditetapkan dalam persoalan.
5. Fungsi Objektif
Fungsi objektif adalah fungsi yang mengatur kondisi agar solusi tersebut dapat dikatakan optimum

Contoh aplikasi algoritma Greedy : Knapsack Problem. Persoalan knapsack dapat diselesaikan dengan menggunakan algoritma Greedy. Persoalannya adalah terdapat beberapa objek dengan berat dan keuntungan tertentu dan sebuah knapsack yang dapat menampung benda dengan berat tertentu. Yang menjadi persoalan adalah bagaimana cara kita dalam memilih objek yang akan dimasukkan ke dalam knapsack agar hasil yang diperoleh maksimum.



Gambar 5. Knapsack Problem

Cara menyelesaikan persoalan tersebut yaitu dengan greedy berdasarkan berat, greedy berdasarkan keuntungan, dan greedy berdasarkan *density* (perbandingan keuntungan/berat)

1. Greedy berdasarkan keuntungan

Pada setiap tahap, dipilih objek dengan keuntungan terbesar. Knapsack akan diisi dengan objek dengan keuntungan terbesar setiap tahap hingga knapsack tidak dapat diisi lagi.

2. Greedy berdasarkan berat
 Pada setiap tahap, dipilih objek dengan berat paling ringan. Knapsack akan diisi dengan objek dengan berat paling ringan setiap tahap hingga knapsack tidak dapat diisi lagi.
3. Greedy berdasarkan *density*
 Pada setiap tahap, dipilih objek dengan rasio keuntungan per berat paling besar. Knapsack akan diisi dengan objek dengan rasio keuntungan per berat paling besar setiap tahap hingga knapsack tidak dapat diisi lagi.

III. ANALISIS PERMASALAHAN

Persoalan dalam MarketGlory sebenarnya mirip dengan persoalan knapsack di atas. Pada persoalan pemilihan *starter* dalam permainan MarketGlory, kita akan mencari kombinasi pembelian dan waktu *Fight* serta *Work* yang tepat agar diperoleh keuntungan secara maksimal.

Pada awal permainan, jumlah *Energy* dan uang yang kita miliki adalah 0 sehingga kita tidak bisa membeli koran untuk menambah *Energy*. Oleh karena itu, kita harus *Work* atau *Fight* terlebih dahulu untuk mengumpulkan modal.

Energy berperan penting dalam menentukan keuntungan optimum karena keuntungan yang didapat dari *Work* dan *Fight* bergantung dari persentase tertentu dari *Energy*. Untuk itu, akan dibahas beberapa pendekatan yang mengelola *Energy* dan uang yang didapat agar keuntungan yang diperoleh menjadi maksimum.

Ada 3 pendekatan yang akan dilakukan dalam hal memilih *starter* bermain MarketGlory yaitu pertama dengan memaksimalkan keuntungan *Work*, kedua dengan memaksimalkan keuntungan *Fight*, dan yang terakhir dengan memaksimalkan kesempatan.

1. Memaksimalkan *Work*
 Cara memaksimalkan hasil *Work* adalah dengan melakukan *Fight* sebanyak 10 kali terlebih dahulu. *Energy* kita awalnya berada pada tingkat 1.00 (tingkat minimal) sehingga tidak ada pengurangan *Energy* sebesar 10%. Setelah melakukan *Fight* sebanyak 10 kali, uang yang didapat akan digunakan untuk membeli koran yang kemudian menghasilkan *Energy* yang akan digunakan untuk *Work*. Pendekatan ini akan kita sebut dengan *Greedy by Work*.
2. Memaksimalkan *Fight*
 Cara memaksimalkan hasil *Fight* adalah dengan melakukan *Work* sebanyak 1 kali terlebih dahulu.

Energy kita awalnya berada pada tingkat 1.00 (tingkat minimal) sehingga tidak ada pengurangan *Energy* sebesar 50%. Setelah melakukan *Work* sebanyak 1 kali, uang yang didapat akan digunakan untuk membeli koran yang kemudian menghasilkan *Energy* yang akan digunakan untuk *Fight* 10 kali. Pendekatan ini akan kita sebut dengan *Greedy by Fight*.

3. Memaksimalkan Kesempatan
 Cara memaksimalkan kesempatan adalah dilakukan dengan membeli koran setiap kali selesai *Fight*. Harapannya adalah dengan *Energy* yang bertambah, hasil yang didapat dari *Fight* yang berikutnya juga akan bertambah. Setelah 10 kali *Fight*, dilanjutkan dengan *Work*. Pendekatan ini akan kita sebut dengan *Greedy by Chance*.

IV. IMPLEMENTASI

Pada bagian ini, akan diterapkan strategi greedy dengan pendekatan-pendekatan yang telah disebutkan dalam Bab 3. Hasilnya kemudian akan dibandingkan untuk menentukan pendekatan yang akan memberikan hasil yang paling optimum.

1. Greedy by Work
 Pada greedy by work, kita memaksimalkan hasil yang didapat dari *Work* dengan mengumpulkan uang sebanyak-banyaknya dari *Fight*. Uang tersebut kemudian akan digunakan untuk membeli koran semaksimal mungkin (10). Tujuannya adalah agar kita bisa *Work* dengan menggunakan *Energy* yang maksimal.

Himpunan kandidat :
 Seluruh tindakan yang menyebabkan *Energy* maksimal pada saat akan dilakukan *Work*

Fungsi Seleksi :
 Pilihlah tindakan *Fight*.

Fungsi Kelayakan :
 Periksa apakah *Fight* sudah melebihi 10 kali atau belum. Jika sudah 10 kali, maka tidak bisa *Fight* lagi karena 1 hari hanya maksimum 10 kali *Fight*

Fungsi Objektif :
 Periksa apakah tindakan tersebut memberikan laba optimum

Himpunan solusi :

No.	Action	Energy	Laba	Uang (IDR)
1.	Fight	1.00	0.08	0.08
2.	Fight	1.00	0.08	0.16

3.	Fight	1.00	0.08	0.24
4.	Fight	1.00	0.08	0.32
5.	Fight	1.00	0.08	0.40
6.	Fight	1.00	0.08	0.48
7.	Fight	1.00	0.08	0.56
8.	Fight	1.00	0.08	0.64
9.	Fight	1.00	0.08	0.72
10.	Fight	1.00	0.08	0.80
11.	Beli 10 koran	4.00	-0.5	0.30
12.	Work	2.00	3.2	3.50

Total penghasilan sebesar 3.50 IDR

2. Greedy by Fight

Pada greedy by fight, kita memaksimalkan hasil yang didapat dari *Fight* dengan mengumpulkan uang dari *Work*. Uang tersebut kemudian akan digunakan untuk membeli koran semaksimal mungkin (10). Tujuannya adalah agar kita bisa *Fight* dengan menggunakan *Energy* yang maksimal.

Himpunan kandidat :

Seluruh tindakan yang menyebabkan *Energy* maksimal pada saat akan dilakukan *Fight*

Fungsi Seleksi :

Pilihlah tindakan *Work*.

Fungsi Kelayakan :

Periksa apakah *Fight* sudah melebihi 1 kali atau belum. Jika sudah 1 kali, maka tidak bisa *Work* lagi karena 1 hari hanya maksimum 1 kali *Work*.

Fungsi Objektif :

Periksa apakah tindakan tersebut memberikan laba optimum

Himpunan solusi :

No.	Action	Energy	Laba	Uang (IDR)
1.	Work	1.00	0.8	0.8
2.	Beli 10 koran	4.00	-0.5	0.3
3.	Fight	3.60	0.32	0.62
4.	Fight	3.42	0.29	0.81
5.	Fight	3.08	0.27	1.08
6.	Fight	2.77	0.25	1.33
7.	Fight	2.49	0.22	1.55
8.	Fight	2.24	0.20	1.75
9.	Fight	2.02	0.18	1.93
10.	Fight	1.82	0.16	2.09

11.	Fight	1.64	0.15	2.24
12.	Fight	1.47	0.13	2.37

Total penghasilan sebesar 2.37 IDR

3. Greedy by Chance

Pada greedy by chance, kita akan memaksimalkan semua kesempatan yang ada. Laba yang didapat dari *Fight* akan digunakan untuk membeli koran untuk menambah *Energy*. Ini akan mengakibatkan laba *Fight* yang terus meningkat tetapi dibarengi dengan pengurangan 10% setiap kali *Fight* karena *Energy* lebih dari 1.00.

Himpunan kandidat :

Seluruh tindakan yang menyebabkan kesempatan membeli koran maksimal.

Fungsi Seleksi :

Pilihlah tindakan yang menyebabkan kesempatan membeli koran maksimal.

Fungsi Kelayakan :

Periksa apakah sudah *Fight* 10 kali dan *Work* 1 kali.

Fungsi Objektif :

Periksa apakah tindakan tersebut memberikan laba optimum

Himpunan solusi :

No.	Action	Energy	Laba	Uang (IDR)
1.	Fight	1.00	0.08	0.08
2.	Beli 1 koran	1.30	-0.05	0.03
3.	Fight	1.17	0.1	0.13
4.	Beli 2 koran	1.67	-0.1	0.03
5.	Fight	1,50	0.13	0.16
6.	Beli 3 koran	2.40	-0.15	0.01
7.	Fight	2.16	0.19	0.20
8.	Beli 4 koran	3.36	-0.20	0.00
9.	Fight	3.02	0.27	0.27
10.	Fight	2.72	0.24	0.51
11.	Fight	2.45	0.22	0.73
12.	Fight	2.21	0.20	0.93
13.	Fight	1.99	0.18	1.11
14.	Fight	1.79	0.16	1.27
15.	Work	0.90	1.43	2.70

Total penghasilan sebesar 2.70 IDR

Tabel perbandingan seluruh algoritma greedy :

Greedy by	Laba (IDR)
Work	3.50
Fight	2.37
Chance	2.70

V. KESIMPULAN

Dari data yang diperoleh, terlihat bahwa solusi yang paling optimum adalah algoritma greedy yang menggunakan pendekatan Greedy by Work. Pendapatan yang didapat sebesar 3.50 IDR jauh melebihi pendekatan yang lain (Greedy by Fight = 2.37 IDR dan Greedy by Chance 2.70 IDR). Algoritma Greedy by Work paling optimum karena dapat dilihat bahwa *Work* memberikan pendapatan yang paling menguntungkan per satuan tenaga meskipun memakan 50% tenaga. Algoritma Greedy by Fight paling tidak menguntungkan karena pada saat awal dilakukan *Work* dengan *Energy* yang minimal sehingga pendapatan dari *Work* sangat rendah. Kesimpulannya adalah lebih efisien menggunakan algoritma yang mengutamakan hasil yang dicapai oleh *Work* karena *Work* memberikan hasil yang paling optimum per satuan tenaga. Penulis mengakui bahwa kesimpulan tersebut tidak sepenuhnya akurat karena banyak faktor yang diabaikan seperti perhitungan pajak, *Productivity*, pengurangan energy per jam, dan sebagainya.

PUSTAKA

- [1] Munir, Rinaldi, "Diktat Kuliah IF2211 Strategi Algoritma", Bandung : Institut Teknologi Bandung
- [2] <http://www.marketglory.com/info/documentation>

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, 20 Desember 2013



Erwin/13511065