

Penerapan Algoritma Greedy dalam Permainan Brave Frontier

Akhmad Fakhoni Listiyan Dede / 13513601

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

akhmadfakhoni@gmail.com

Abstract—Brave and Frontier adalah permainan berbasis putaran. Permainan ini merupakan permainan strategi dimana pada awal permainan diberi monster dengan elemen tertentu, dan kemudian akan mendapatkan monster - monster yang lain setelah berhasil melewati satu stage. Dalam permainan ini ada atribut Petir, Air, Api, Pohon, Kegelapan, dan Cahaya. Masing - masing elemen memiliki keunggulan tersendiri terhadap elemen lain. Tiap monster juga punya atribut HP, Attack, Defence, dan Recovery. Dengan adanya atribut dan jenis elemen ini, pemain akan bingung dalam memilih monster mana yang akan dibawa untuk bertempur. Makalah ini akan membahas mengenai pemilihan monster dalam menyerang musuh, menggunakan algoritma greedy.

Index Terms—monster, atribut, elemen, attack, defence.

I. PENDAHULUAN

Brave Frontier adalah permainan strategi berbasis putaran, atau yang sering disebut turn based strategy. Pada awalnya pemain akan diberi satu buah monster sebagai modal untuk melawan musuh yang ditemui. Setelah monster dikalahkan, pemain akan mendapatkan monster tersebut. Sebelum berperang dengan musuh, pemain bisa menentukan monster mana saja yang akan dibawa. Ada 6 elemen monster, petir, air, api, pohon, cahaya, dan kegelapan. Monster dengan elemen petir memiliki keuntungan saat melawan air, air memiliki keuntungan saat melawan api, api memiliki keuntungan saat melawan pohon, pohon memiliki keuntungan saat melawan petir, cahaya memiliki keuntungan saat melawan kegelapan, dan kegelapan juga memiliki keuntungan saat menyerang cahaya.

Setiap monster juga memiliki empat jenis atribut, yaitu HP, Atk, Def, Rec. HP (Health Point) adalah banyaknya darah yang dimiliki oleh monster. Atk (Attack) mempengaruhi besarnya kerusakan saat menyerang musuh. Def (Defence) mempengaruhi besarnya kerusakan yang diterima saat musuh menyerang. Rec (Recovery) menentukan banyaknya HP yang bertambah untuk setiap hati yang diperoleh saat melawan musuh. Pada saat menyerang, selain besarnya Atk, ditentukan pula oleh besarnya Def monster musuh.



Gambar 1. Keterkaitan antar elemen

Pada saat awal permainan, pemain dapat mengatur monster mana saja yang akan dibawa untuk melawan musuh. Satu tim terdiri atas maksimal lima monster. Selain atribut dan elemen, monster juga memiliki cost. Saat pemain memilih monster, pemain juga dibatasi oleh cost, dan monster yang dia bawa tidak boleh melebihi cost yang telah ditentukan. Cost ini akan bertambah seiring dengan naiknya level pemain.

Dengan adanya berbagai macam atribut, elemen, dan cost dari permainan ini, maka penulis akan menguraikan beberapa strategi menggunakan algoritma greedy untuk memilih monster mana yang sebaiknya di bawa, dan urutan monster saat menyerang musuh.



Gambar 2. Monster yang memiliki HP, Atk, Def, Rec, dan cost

II. DASAR TEORI

Algoritma greedy merupakan metode yang paling populer untuk memecahkan persoalan optimasi. Ada dua macam persoalan optimasi, yaitu maksimasi dan minimasi.

1. Maksimasi

Greedy jenis maksimasi berusaha untuk mencari solusi maksimum dari sejumlah solusi yang mungkin. Contohnya adalah pada permasalahan knapsack. Permasalahan ini mencari benda sebanyak mungkin dengan cost beban yang sesedikit mungkin.

2. Minimasi

Greedy jenis minimasi berusaha memilih solusi minimum dari semua kemungkinan solusi yang mungkin. Contohnya pada saat mencari jarak minimum dari titik awal ke akhir, maka akan dicari jarak yang terpendek dari titik awal ke titik berikutnya hingga mencapai titik akhir

Algoritma greedy adalah algoritma yang memecahkan masalah langkah per langkah. Pada setiap langkah, dia akan mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu tanpa memperhatikan konsekuensi akibat dari pilihan tersebut untuk langkah - langkah berikutnya. Pilihan terbaik pada setiap langkah ini dinamakan pilihan optimum lokal. Optimum lokal ini diharapkan bisa mengarah ke solusi optimum global.

Salah satu kelemahan dari algoritma greedy adalah tidak selalu menghasilkan solusi yang optimal. Hal ini dikarenakan greedy memilih untuk mencari optimum lokal, dan tidak memperhitungkan akibatnya kepada optimum global. Apalagi greedy juga tidak memperbolehkan kembali ke titik dimana kemungkinan dia melakukan kesalahan jika ternyata hasil yang didapat bukanlah hasil yang maksimum.

Greedy memiliki beberapa elemen:

1. Himpunan kandidat C
Berisi elemen - elemen pembentuk solusi.
2. Himpunan solusi S
Berisi kandidat - kandidat yang terpilih sebagai solusi persoalan.
3. Fungsi seleksi
Fungsi untuk memilih kandidat yang paling memungkinkan mencari solusi optimal. Kandidat yang sudah dipilih pada suatu langkah tidak pernah dipertimbangkan lagi pada langkah selanjutnya.
4. Fungsi kelayakan
Memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak, yaitu kandidat tersebut bersama dengan himpunan solusi yang sudah terbentuk tidak melanggar constraint.
5. Fungsi obyektif
Yaitu fungsi yang memaksimalkan atau meminimumkan nilai solusi

Permasalahan Knapsack

Persoalan knapsack adalah persoalan optimasi. Dalam permasalahan ini, diberikan sebuah cost maksimum yang bisa dipakai. Kemudian ada sejumlah item yang mempunyai cost tertentu dan profit tertentu. Algoritma Knapsack akan menentukan item mana yang akan diambil agar bisa mendapatkan profit yang sebesar - besarnya.

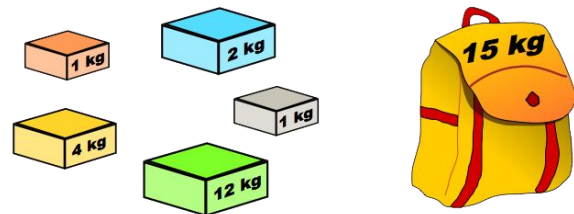
Untuk menyelesaikan persoalan tersebut, ada beberapa alternatif algoritma greedy yang bisa dipakai.

1. Greedy by Profit

Greedy by profit akan memilih barang dengan profit yang paling tinggi terlebih dahulu.

2. Greedy by Cost

Greedy by cost akan memilih item dengan cost yang paling kecil terlebih dahulu, sehingga diharapkan akan semakin banyak item yang bisa diambil, dan semakin banyak pula profit yang bisa diperoleh.



Gambar 2. Gambaran knapsack

III. ANALISIS PERMASALAHAN

Ada beberapa permasalahan dalam permainan Brave Frontier yang dapat diselesaikan dengan greedy. Penulis mengambil 2 jenis permasalahan, yaitu permasalahan saat menentukan monster mana yang akan dibawa untuk menyerang musuh, dan menentukan kombinasi serangan yang dilakukan oleh pemain untuk menyerang musuh.

1. Pemilihan monster yang dibawa

Permasalahan yang pertama adalah saat menentukan lima monster yang akan dibawa untuk melawan musuh. Dengan adanya pembatasan cost, maka dalam pemilihan monster juga harus selektif. Dalam pemilihan dengan algoritma greedy ini, dilakukan greedy by cost, greedy by Atk, dan greedy by Def, greedy by Rec. Dengan greedy by cost akan dicari monster dengan cost yang minimum. Sedangkan greedy by Atk akan dipilih monster dengan ATk maksimum. Greedy by Def mengambil monster dengan defence yang paling tinggi terlebih dahulu, dan Greedy by Rec memilih monster dengan Rec yang paling tinggi.

2. Menentukan kombinasi serangan yang dipakai untuk menyerang musuh

Pada saat bertemu dengan musuh, maka ada berbagai kombinasi serangan yang memungkinkan. Pemain bisa melakukan serangan secara acak, maupun dengan prinsip - prinsip tertentu. Dalam makalah ini penulis akan menjabarkan tiga prinsip yang bisa digunakan untuk memilih kombinasi serangan. Yang pertama adalah memakai monster untuk melawan monster musuh dengan elemen yang lebih lemah. Dalam pembahasan selanjutnya, hal ini disebut sebagai greedy serangan by elemen. Yang kedua, bisa juga dengan memilih monster musuh dengan Atk yang paling tinggi terlebih dahulu, yang selanjutnya disebut sebagai greedy serangan by Atk. Cara yang ketiga yang bisa dilakukan adalah memilih monster musuh yang memiliki Def paling rendah terlebih dahulu, dengan harapan monster tersebut lebih cepat dikalahkan. Ini disebut sebagai greedy serangan by Def.



Gambar 3. Pemilihan Monster dalam 5 slot

IV. IMPLEMENTASI

Dalam penerapan greedy untuk memilih monster, penulis akan mengambil contoh 10 monster, dan kemudian diambil maksimal 5 monster dengan algoritma greedy yang akan dipakai. Permasalahan ini mirip dengan permasalahan knapsack. Kesepuluh monster tersebut ada di dalam tabel berikut.

| Nama | Atribut |
|--------------------|---|
| Ice Goddess Selena | Cost = 10 Atk = 1490 Def = 1239 Rec = 1417 |
| Bolt Goddess Amy | Cost = 15 Atk = 1432 Def = 1364 Rec = 1237 |

| | |
|-------------------|--|
| Earth Knight Edea | Cost = 10 Atk = 1215 Def = 1239 Rec = 951 |
| Earth Pike Lance | Cost = 8 Atk = 1306 Def = 1162 Rec = 784 |
| Great Thief Leon | Cost = 9 Atk = 1096 Def = 998 Rec = 1030 |
| Polar Angel Tiara | Cost = 8 Atk = 1010 Def = 1112 Rec = 1816 |
| Bishop Merith | Cost = 6 Atk = 667 Def = 1131 Rec = 1157 |
| Hell King Hades | Cost = 8 Atk = 1100 Def = 833 Rec = 542 |
| God Blade Mifune | Cost = 9 Atk = 1316 Def = 511 Rec = 638 |
| Bahamut | Cost = 8 Atk = 1138 Def = 970 Rec = 647 |

Tabel 1. Sampel tabel monster

A. Pemilihan Monster menggunakan greedy by cost

Pada greedy by cost ini, akan dipilih monster dengan cost yang paling murah, dan maksimal cost yang paling mahal. Untuk pengukuran hasil, diukur dengan cara melawan musuh dengan mode auto sehingga saling serang akan terjadi dengan sendirinya, dan waktu dihitung dari awal hingga pertempuran selesai. Pemilihan monster dilakukan pada 4 monster, dan cost yang tersisa akan dialokasikan pada monster dengan cost yang paling mahal.

Himpunan Kandidat:

Kesepuluh monster dalam tabel 1.

Fungsi seleksi:

Mengambil monster dengan cost yang paling kecil.

Fungsi Kelayakan:

Memeriksa apakah cost dari monster yang diambil melebihi cost total yang dimiliki pemain.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang dipilih dapat memenangkan satu stage permainan dengan waktu yang tercepat.

Himpunan Solusi

| Nama | Cost |
|-------------------|------|
| Bishop Merith | 6 |
| Earth Pike Lance | 8 |
| Polar Angel Tiara | 8 |
| Hell King Hades | 8 |
| Bahamut | 8 |

Tabel 2. Hasil Monster dari greedy dan costnya

Hasil:

Waktu untuk menyelesaikan 1 stage adalah 14 menit 5 detik.

B. Pemilihan Monster menggunakan greedy by Atk

Greedy by Atk akan dipilih monster dengan Atk yang paling tinggi terlebih dahulu, dengan maksimum monster yang bisa diambil adalah lima monster. Dengan diambilnya monster yang memiliki Atk paling besar, harapannya adalah monster musuh lebih cepat untuk di kalahkan.

Himpunan Kandidat:

Kesepuluh monster dalam tabel 1.

Fungsi Seleksi:

Mengambil monster dengan Atk yang paling besar.

Fungsi Kelayakan:

Memeriksa apakah cost dari monster yang diambil melebihi cost total yang dimiliki pemain.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang dipilih dapat memenangkan satu stage permainan dengan waktu yang tercepat.

Himpunan Solusi

| Nama | Atk |
|--------------------|------|
| Ice Goddess Selena | 1490 |
| Bolt Goddess Amy | 1432 |
| God Blade Mifune | 1316 |
| Earth Pike Lance | 1306 |
| Earth Knight Edea | 1239 |

Tabel 3. Hasil monster dari greedy by Atk dan besarnya Atk

Hasil yang diperoleh

Waktu untuk menyelesaikan 1 stage adalah 8 menit 46 detik.

C. Pemilihan Monster menggunakan greedy by Def

Greedy by Def akan meningkatkan kemungkinan lamanya monster akan bertahan terhadap serangan musuh. Semakin tinggi Def, semakin kecil damage yang akan diterima.

Himpunan Kandidat:

Kesepuluh monster dalam tabel 1.

Fungsi Seleksi:

Mengambil monster dengan Def yang paling besar.

Fungsi Kelayakan:

Memeriksa apakah cost dari monster yang diambil melebihi cost total yang dimiliki pemain.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang dipilih dapat memenangkan satu stage permainan dengan waktu yang tercepat.

Himpunan Solusi

| Nama | Def |
|--------------------|------|
| Bolt Goddess Amy | 1364 |
| Ice Goddess Selena | 1239 |
| Earth Pike Lance | 1162 |
| Bishop Merith | 1131 |
| Polar Angel Tiara | 1112 |

Tabel 4. Hasil Monster dari greedy dan Defnya

Hasil yang diperoleh

Waktu untuk menyelesaikan 1 stage adalah 8 menit 1 detik.

D. Pemilihan Monster menggunakan greedy by Rec

Greedy by Rec akan memilih monster dengan Rec yang paling tinggi terlebih dahulu. Rec dipilih dengan harapan monster ini akan lebih sulit untuk dikalahkan, karena untuk setiap heart yang didapatkan si monster saat melawan musuh, bisa merecovery health point dengan cukup signifikan.

Himpunan Kandidat:

Kesepuluh monster dalam tabel 1.

Fungsi Seleksi:

Mengambil monster dengan Rec yang paling besar.

Fungsi Kelayakan:

Memeriksa apakah cost dari monster yang diambil melebihi cost total yang dimiliki pemain.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang

dipilih dapat memenangkan satu stage permainan dengan waktu yang tercepat.

Himpunan Solusi

| Nama | Rec |
|--------------------|------|
| Polar Angel Tiara | 1816 |
| Ice Goddess Selena | 1417 |
| Bolt Goddess Amy | 1237 |
| Bishop Merith | 1157 |
| Great Thief Leon | 1030 |

Tabel 5. Hasil Monster dari greedy dan Recnya

Hasil yang diperoleh

Waktu untuk menyelesaikan 1 stage adalah 8 menit 36 detik.

Apabila dirangkum ke dalam tabel, maka hasil dari waktu yang diperoleh adalah sebagai berikut.

| Greedy by | Waktu |
|-----------|-------------------|
| Cost | 14 menit 5 detik. |
| Atk | 8 menit 46 detik. |
| Def | 8 menit 1 detik. |
| Rec | 8 menit 36 detik. |

Tabel 6. Hasil berbagai macam algoritma pemilihan monster dalam 1 tabel

Terlihat bahwa ternyata dengan algoritma pemilihan greedy by Def, maka menghasilkan waktu tercepat dalam melawan musuh.

E. Pemilihan Menyerang menggunakan greedy serangan by elemen

Untuk memilih monster mana yang akan menyerang terlebih dahulu, maka ukuran waktu untuk memenangkan sebuah stage tidak bisa lagi digunakan. Yang akan digunakan adalah banyaknya turn yang dibutuhkan untuk menyelesaikan satu stage.

Dengan greedy by elemen, pemilihan monster yang akan menyerang monster musuh didasarkan pada monster yang memiliki elemen lebih kuat akan menyerang monster dengan elemen yang lebih lemah. Hal ini dilakukan dengan harapan memaksimalkan damage yang akan diperoleh oleh musuh karena elemen yang lebih kuat dapat menyerang dengan Atk yang lebih tinggi daripada saat dia menyerang elemen yang biasa saja.

Jika monster milik musuh tidak ada yang memiliki elemen lebih lemah, maka penyerangan akan dilakukan secara random dan akan menyerang 1 monster sampai monster tersebut mati sebelum menyerang monster lainnya

Himpunan Kandidat:

Setiap kombinasi serangan yang mungkin untuk

menyerang monster musuh.

Fungsi Seleksi:

Memilih monster dengan elemen yang lebih kuat menyerang monster dengan elemen yang lebih lemah.

Fungsi Kelayakan:

Memeriksa apakah dengan greedy serangan by elemen, satu stage bisa diselesaikan dengan lancar.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang dipilih dapat memenangkan satu stage permainan dengan turn yang paling sedikit.

Himpunan Solusi:

Semua kombinasi serangan yang dilakukan oleh pemain dengan kondisi monster dengan elemen yang lebih kuat menyerang monster musuh dengan elemen yang lebih kuat. Jika tidak ada elemen yang lebih lemah, maka monster akan membantu serangan monster lain. Monster musuh dengan elemen yang lebih kuat akan diserang dengan urutan yang paling akhir.

Hasil:

Hasil yang didapatkan dengan algoritma greedy serangan by elemen adalah 24 turn.

F. Pemilihan Menyerang menggunakan greedy serangan by Atk

Dengan greedy by Atk, maka setiap monster dari pemain akan menyerang monster musuh dengan nilai Atk yang paling tinggi terlebih dahulu. Setelah monster tersebut berhasil dikalahkan, kemudian beranjak ke monster berikutnya dengan Atk yang paling tinggi pula. Dengan penyerangan model seperti ini, diharapkan monster musuh dengan Atk yang paling tinggi memiliki kesempatan menyerang yang lebih sedikit, sehingga meminimalkan peluang monster milik pemain dikalahkan oleh monster musuh.

Himpunan Kandidat:

Setiap kombinasi serangan yang mungkin untuk menyerang monster musuh.

Fungsi Seleksi:

Memilih menyerang monster lawan yang memiliki Atk paling besar terlebih dahulu.

Fungsi Kelayakan:

Memeriksa apakah dengan greedy serangan by Atk, satu stage bisa diselesaikan dengan lancar.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang dipilih dapat memenangkan satu stage permainan dengan turn yang paling sedikit.

Himpunan Solusi:

Semua kombinasi serangan yang dilakukan oleh pemain dengan kondisi monster musuh yang memiliki Atk paling tinggi akan diserang terlebih dahulu hingga mati, baru berpindah ke monster musuh lain. Monster musuh dengan Atk paling rendah akan diserang terakhir.

Hasil:

Hasil yang didapatkan dengan algoritma greedy serangan by Atk adalah 26 turn.

G. Pemilihan Menyerang menggunakan greedy serangan by Def

Dengan greedy by Def, maka setiap monster dari pemain akan menyerang monster musuh dengan nilai Def yang paling rendah terlebih dahulu. Dengan algoritma ini, diharapkan lebih banyak monster musuh yang kalah terlebih dahulu, sehingga monster musuh yang menyerang monster pemain semakin sedikit, dan peluang monster pemain untuk bertahan lebih tinggi.

Himpunan Kandidat:

Setiap kombinasi serangan yang mungkin untuk menyerang monster musuh.

Fungsi Seleksi:

Memilih menyerang monster lawan yang memiliki Def paling kecil terlebih dahulu.

Fungsi Kelayakan:

Memeriksa apakah dengan greedy serangan by Atk, satu stage bisa diselesaikan dengan lancar.

Fungsi Objektif:

Memeriksa apakah kombinasi monster yang dipilih dapat memenangkan satu stage permainan dengan turn yang paling sedikit.

Himpunan Solusi:

Semua kombinasi serangan yang dilakukan oleh pemain dengan kondisi monster musuh yang memiliki Def paling kecil akan diserang terlebih dahulu hingga mati, baru berpindah ke monster musuh lain. Monster musuh dengan Def paling besar akan diserang paling akhir.

Hasil:

Hasil yang didapatkan dengan algoritma greedy serangan by elemen adalah 21 turn.

Apabila dimasukkan ke dalam tabel, maka hasilnya akan menjadi seperti berikut:

| Greedy serangan by | Banyaknya putaran |
|--------------------|-------------------|
| Elemen | 24 |
| Atk | 26 |
| Def | 21 |

Tabel 7. Tabel hasil algoritma pemilihan monster mana yang terlebih dahulu menyerang

Terlihat di tabel 7 bahwa algoritma greedy serangan by Def memiliki putaran yang paling sedikit. Hal ini dikarenakan monster musuh dengan Def yang paling kecil akan lebih cepat mati saat diserang secara bersamaan oleh monster pemain.

IV. KESALAHAN YANG MUNGKIN MUNCUL

Dalam pengujian untuk setiap algoritma greedy dalam masalah ini ada hal yang menyebabkan hasil pengujian kurang valid. Salah satunya adalah saat menguji lamanya waktu yang dibutuhkan untuk menyelesaikan satu stage untuk pengujian greedy by cost, greedy by Atk, greedy by Def, dan greedy by Rec. Hal ini dikarenakan pemain tidak bisa memilih musuh yang sama untuk setiap algoritma greedy. Musuhnya akan berbeda - beda walaupun stage permainan sama. Begitu juga untuk algoritma greedy serangan by elemen, greedy serangan by Atk, dan greedy serangan by Def. Musuh yang dihadapi juga tidak bisa sama persis untuk setiap algoritma. Tapi untuk meminimalkan kesalahan yang muncul, penulis berusaha untuk mengukur dengan memainkan monster untuk semua algoritma di stage yang sama, sehingga paling tidak level kesulitan yang dihadapi tiap algoritma juga sama.

Kemungkinan kesalahan yang kedua adalah, pemilihan monster yang diambil sebagai sampel untuk atribut tertentu ada yang atributnya tidak dominan. Misalnya saja pada Tabel 6, greedy yang paling efektif justru adalah greedy by Def. Padahal secara logika seharusnya adalah greedy by Atk. Hal ini bisa terjadi karena monster yang terpilih dengan greedy by Def adalah monster dengan tipe elemen dominan Air, sedangkan monster musuh dominan dengan tipe api, sehingga serangan monster hasil greedy by Def ini bisa lebih efektif daripada monster hasil greedy by Atk.

V. KESIMPULAN

Algoritma greedy cukup efektif untuk menyelesaikan permasalahan dalam permainan Brave Frontier ini. Yang pertama adalah pemilihan monster pemain dengan greedy by Def memiliki efektifitas yang paling tinggi. Yang kedua adalah greedy serangan by Def yang bisa mendapatkan putaran paling sedikit untuk menyelesaikan 1 stage. Hal ini dikarenakan dengan menyerang monster musuh yang memiliki Def paling kecil, maka monster tersebut akan lebih cepat mati, sehingga tidak bisa melakukan serangan untuk putaran berikutnya. Hal ini akan mengurangi damage yang diderita oleh monster milik pemain untuk setiap putaran.

REFERENCES

- [1] Munir, Rinaldi, “Diktat Kuliah IF3051 Strategi Algoritma ”, Bandung : Institut Teknologi Bandung.
- [2] http://bravefrontierglobal.wikia.com/wiki/Brave_Frontier_Wiki
- [3] Laba, Gilang Laksana, “Penerapan Algoritma Greedy dalam permainan Front Mission Evolved”, Bandung : Institut Teknologi Bandung.

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, 19 Mei 2014



Akhmad Fakhoni L D - 13513601