

Penerapan Algoritma Greedy pada Permainan *Fire Emblem*

Chairul Ichsan - 13508082

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
chairul_ichsan@gmail.com

Abstrak—*Fire Emblem* adalah salah satu game strategi yang menggunakan basis giliran (*turn-based*). Salah satu bagian dari permainan ini adalah melakukan penyerangan terhadap musuh saat sedang perang. Permainan ini berbasis giliran, oleh karena itu setiap karakter akan mempunyai giliran tertentu untuk melakukan aksi. Setiap karakter dibatasi hanya bisa melakukan satu kali aksi setiap giliran. Dengan demikian, aksi yang dilakukan setiap gilirannya menjadi kunci untuk mengalahkan tim lawan. Algoritma *Greedy* adalah algoritma yang menghasilkan solusi lokal terbaik dari suatu permasalahan, dengan harapan bahwa solusi tersebut akan menjadi solusi paling optimal bagi permasalahan tersebut secara keseluruhan. Pada makalah ini akan dibahas bagaimana penerapan algoritma *greedy* pada permainan *fire emblem* untuk menentukan aksi penyerangan yang paling optimal dalam suatu giliran.

Kata Kunci—*fire emblem*, algoritma, *greedy*.

I. PENDAHULUAN

Fire Emblem adalah permainan strategi yang menggunakan basis giliran. Permainan ini diproduksi untuk digunakan pada *console GameBoy Advance*. Inti dari permainan ini adalah melewati pertempuran – pertempuran untuk menyelesaikan permainan. Selama permainan pemain akan mempunyai sebuah grup yang berisi karakter – karakter yang bisa digunakan untuk bertempur. Setiap pertempurannya pemain akan bertempur dengan grup lawan. Ada dua kemungkinan kondisi yang harus dipenuhi untuk memenangkan pertempuran, yaitu membunuh pemimpin grup lawan atau membunuh semua anggota grup lawan. Permainan berakhir (*gameover*) jika pemimpin grup pemain terbunuh.

Permainan ini tidak jauh berbeda dengan permainan strategi berbasis giliran pada umumnya. Dalam pertempuran grup pemain dan grup lawan akan melakukan aksi secara bergantian. Pada setiap gilirannya, setiap karakter dalam satu grup juga akan melakukan aksinya secara bergiliran. Karakter hanya bisa melakukan satu kali aksi dalam satu giliran. Aksi yang bisa dilakukan oleh suatu karakter antara lain menyerang pemain lawan, menggunakan barang persediaan, dan menunggu. Setiap karakter mempunyai status yang menggambarkan kemampuan karakter tersebut.



Gambar 1 Contoh tampilan saat pertempuran

Tapi permainan ini juga mempunyai perbedaan dengan permainan strategi berbasis giliran pada umumnya. Pada permainan ini, setiap karakter hanya bisa melakukan aksi menyerang jika menggunakan senjata. Senjata tersebut mempunyai jumlah penggunaan yang terbatas. Setiap kali digunakan untuk menyerang, penggunaan senjata tersebut akan berkurang. Batas penggunaan suatu senjata berbeda-beda tergantung jenisnya. Jika penggunaan senjata sudah mencapai batasnya maka senjata tersebut akan hancur dan tidak dapat digunakan lagi. Dengan demikian, menggunakan senjata secara optimal menjadi faktor penting untuk memenangkan permainan ini. Penerapan algoritma *greedy* yang dibahas dalam makalah ini bertujuan untuk mencari kemungkinan penyerangan yang paling optimal, sehingga tidak terjadi penggunaan senjata secara sia – sia dan bisa memanfaatkan giliran secara optimal.

II. ALGORITMA GREEDY

Algoritma *greedy* adalah algoritma yang mencari solusi lokal paling optimal dengan harapan solusi tersebut adalah solusi paling optimal secara keseluruhan. Prinsip dari algoritma *greedy* adalah : “*take what you can get now*”.

Elemen – elemen algoritma *greedy* :

1. Himpunan Kandidat

Himpunan yang berisi kemungkinan – kemungkinan yang bisa menjadi solusi dari

permasalahan

2. Himpunan Solusi

Himpunan yang berisi kandidat yang telah terpilih sebagai solusi.

3. Fungsi Seleksi

Fungsi untuk melakukan seleksi terhadap kandidat agar menghasilkan solusi yang diharapkan

4. Fungsi Kelayakan

Fungsi untuk memastikan bahwa solusi yang dipilih memenuhi syarat

5. Fungsi Obyektif

Memilih solusi paling optimal dari himpunan solusi.

Algoritma *greedy* akan membentuk solusi langkah per langkah. Pada setiap langkah algoritma mengambil pilihan terbaik yang dapat diperoleh pada saat itu tanpa memperhatikan konsekuensi ke depannya, sebagaimana prinsip algoritma *greedy*. Dengan demikian, secara umum algoritma *greedy* bekerja dengan cara melakukan pencarian himpunan solusi pada himpunan kandidat dengan fungsi seleksi dan diverifikasi dengan fungsi kelayakan kemudian memilih solusi paling optimal dengan fungsi obyektif.

Penerapan algoritma *greedy* yang dibahas dalam makalah ini pun untuk mencari solusi lokal, yaitu solusi penyerangan yang paling optimal dalam satu giliran. Solusi tersebut diharapkan juga merupakan solusi paling optimal untuk memenangkan permainan, walaupun masih ada faktor – faktor lain yang bisa mempengaruhi pertempuran.

III. PENERAPAN ALGORITMA *GREEDY*

A. Parameter permainan yang digunakan dalam penerapan

Penerapan algoritma *greedy* ini akan menggunakan parameter – parameter yang ada dalam permainan. Secara umum, parameter yang digunakan ada empat :

- Jenis serangan
- Senjata
- Status karakter
- Lokasi karakter lawan

Dalam permainan ini ada dua jenis serangan, yaitu serangan fisik dan serangan sihir. Jenis serangan ini tergantung dari karakter, ada karakter yang bisa melakukan serangan fisik dan ada karakter yang bisa melakukan serangan sihir. Perbedaan jenis serangan ini akan berpengaruh terhadap senjata yang digunakan dan status karakter digunakan untuk perhitungan total serangan.

Sebagaimana telah dijelaskan sebelumnya, setiap karakter hanya dapat melakukan serangan jika menggunakan senjata. Senjata dalam permainan ini terdiri

dari beberapa kategori, seperti kategori pedang, panah, tombak, dsb. Kategori senjata akan mempengaruhi serangan suatu karakter sebab pada permainan ini senjata kategori tertentu akan lebih efektif jika digunakan terhadap kategori tertentu lainnya. Berikut ini tabel yang menjelaskan kategori senjata yang lebih efektif terhadap kategori tertentu lainnya :

Tabel 1 Tabel efektifitas senjata

No.	Kategori Senjata	Efektif terhadap
1	Pedang	Kampak
2	Tombak	Pedang
3	Kampak	Tombak
4	Sihir <i>dark</i>	Sihir <i>anima</i>
5	Sihir <i>light</i>	Sihir <i>dark</i>
6	Sihir <i>anima</i>	Sihir <i>light</i>

Selain itu, kategori dan jenis senjata tertentu juga akan lebih efektif terhadap karakter tertentu, misalnya panah akan lebih efektif jika digunakan terhadap karakter yang bergerak dengan cara terbang, salah satu jenis pedang akan lebih efektif jika digunakan terhadap karakter yang menggunakan baju pelindung yang berat, dsb.

Selain karakteristik tersebut senjata juga memiliki status tersendiri yang bisa mempengaruhi serangan yang dihasilkan, yaitu :

1. Level senjata
2. Mt, mempengaruhi nilai serangan senjata tersebut
3. Hit, mempengaruhi kemungkinan serangan dengan senjata tersebut mengenai sasaran
4. Crit, mempengaruhi kemungkinan serangan dengan senjata tersebut mengeluarkan serangan dengan nilai tiga kali lipat serangan biasa

Status dari karakter akan mempengaruhi serangan yang dilakukannya. Namun, tidak semua status berkaitan dengan serangan. Berikut ini adalah tabel yang menjelaskan kaitan status dengan serangan :

Tabel 2 Tabel fungsi status

No.	Status	Kegunaan
1	Str	Meningkatkan serangan fisik
2	Mag	Meningkatkan serangan sihir
3	Skill	Meningkatkan kemungkinan serangan mengenai sasaran (mempengaruhi penggunaan senjata)
4	Spd	Meningkatkan kemampuan menghindar dan memungkinkan karakter menyerang dua kali dalam satu giliran
5	Luck	Meningkatkan kemungkinan melakukan serangan yang

		bernilai tiga kali lipat serangan biasa
6	Def	Mengurangi serangan fisik yang diterima
7	Res	Mengurangi serangan sihir yang diterima

Selain status seperti yang disebutkan dalam tabel di atas, karakter juga mempunyai status keahlian penggunaan senjata. Tingkatan status ini ditandai dengan huruf, yaitu : $S > A > B > C > D > E$. Perbedaan keahlian penggunaan senjata akan mempengaruhi kemungkinan keberhasilan suatu serangan.

Dalam permainan ini medan pertempuran mempunyai pengaruh yang menentukan keberhasilan suatu serangan. Misalnya, suatu karakter mempunyai kemungkinan menghindari serangan lawan lebih tinggi saat berada di dalam hutan daripada saat berada di lapangan terbuka. Setiap jenis medan mempunyai dua karakteristik atau status yang mempengaruhi serangan, yaitu DEF dan AVO. DEF memberikan karakter tambahan kemampuan untuk bertahan dari serangan AVO memberikan karakter tambahan kemungkinan untuk menghindari serangan yang diterima. Oleh karena itu, lokasi karakter lawan juga harus diperhitungkan untuk menentukan serangan yang optimal.

B. Elemen – elemen yang digunakan dalam penerapan

Sebagaimana telah dijelaskan bahwa algoritma *greedy* memiliki elemen – elemen yang digunakan untuk menemukan solusi. Penerapan dari elemen – elemen tersebut adalah sebagai berikut :

1. Himpunan Kandidat
Himpunan kandidat yang digunakan berupa kemungkinan penyerangan dari tiap karakter. Misalkan grup pemain terdiri dari tiga karakter, A, B, C dan grup lawan terdiri dari dua karakter D, E. Himpunan kandidat dari kondisi tersebut adalah :
{A menyerang D}, {A menyerang E},
{B menyerang D}, {B menyerang E},
{C menyerang D}, dan {C menyerang E}.
2. Himpunan Solusi
Himpunan solusi berisi kemungkinan penyerangan yang merupakan penyerangan paling optimal pada saat itu.
3. Fungsi Seleksi
Fungsi seleksi yang digunakan akan memilih kemungkinan penyerangan yang paling besar. Fungsi ini memanggil beberapa fungsi lain untuk memperhitungkan pengaruh dari faktor – faktor yang telah dijelaskan sebelumnya. Fungsi – fungsi tersebut akan memberikan nilai tingkat optimalitas pada setiap kemungkinan penyerangan. Fungsi seleksi akan memilih kemungkinan penyerangan dengan nilai tingkat optimalitas paling besar.
4. Fungsi Kelayakan

Fungsi kelayakan yang digunakan adalah fungsi untuk memeriksa apakah suatu penyerangan bisa terjadi atau tidak. Dalam hal ini, penyerangan dinilai tidak bisa terjadi jika langkah karakter tidak cukup untuk mencapai karakter lawan atau karakter tidak memiliki senjata yang bisa digunakan.

C. Cara kerja algoritma yang diterapkan

Pertama – tama program akan melakukan pendataan terhadap anggota grup pemain dan grup lawan. Kemudian program akan membentuk himpunan kandidat yang berisi semua kemungkinan penyerangan karakter pemain terhadap karakter lawan.

Kemudian fungsi seleksi akan melakukan pemilihan terhadap himpunan kandidat tersebut. Pertama – tama fungsi seleksi akan memanggil fungsi untuk menentukan tingkat optimalitas pada setiap anggota himpunan kandidat. Cara kerja fungsi untuk menentukan tingkat optimalitas adalah sebagai berikut :

- a) Menghitung pengaruh dari senjata yang digunakan oleh karakter

Pada bagian ini, pertama – tama fungsi akan melakukan perbandingan antara senjata yang digunakan oleh karakter pemain dengan senjata yang digunakan oleh karakter lawan sebagai berikut :

- Jika senjata karakter pemain lebih unggul, tingkat optimalitas ditambah 1
- Jika senjata karakter lawan lebih unggul tingkat optimalitas dikurangi 1
- Jika tidak ada senjata yang lebih unggul tingkat optimalitas tidak ditambah atau dikurangi

Kemudian fungsi akan melihat apakah senjata yang digunakan efektif terhadap karakter lawan atau tidak. Jika ya, tingkat optimalitas ditambah 1.

Setelah itu fungsi memperhitungkan efek dari status senjata tersebut. Tingkat optimalitas ditambahkan dengan level senjata, status Mt, status Crit/10, dan status Hit/10.

- b) Menghitung pengaruh dari status karakter

Pada bagian ini pertama – tama fungsi akan menambahkan tingkat optimalitas dengan selisih antara status Str karakter pemain dengan status Def karakter lawan (atau selisih status Mag dan status Res jika serangan nya merupakan serangan sihir). Kemudian tingkat optimalitas ditambahkan dengan selisih antara status Spd karakter pemain dan karakter lawan. Begitu juga dengan selisih status Skill dan Luck.

Setelah itu fungsi akan menambahkan tingkat optimalitas dengan selisih dari kemampuan penggunaan senjata karakter pemain dengan karakter lawan. Tingkat kemampuan penggunaan senjata diubah menjadi angka dengan urutan $E = 1, D = 2, C = 3, B = 4, A = 5, S = 6$.

- c) Menghitung pengaruh dari medan tempat karakter lawan berada

Pada bagian ini fungsi akan mengurangi tingkat optimalitas dengan nilai DEF dan AVO dari medan tempat karakter lawan berada.

Dengan demikian nilai tingkat optimalitas menggambarkan bagaimana keunggulan senjata dan status karakter pemain dan kondisi medan karakter lawan. Setelah itu fungsi seleksi akan memilih kemungkinan penyerangan dengan nilai tingkat optimalitas yang paling tinggi.

Setelah itu fungsi kelayakan akan menilai apakah penyerangan tersebut bisa dilakukan atau tidak, jika tidak fungsi seleksi akan dipanggil lagi untuk mencari kemungkinan penyerangan lain, jika ya kemungkinan tersebut akan dikembalikan sebagai solusi.

D. Contoh Penerapan

Berikut ini adalah contoh kasus yang dihadapi langsung dalam permainan :



Gambar 2 Contoh kasus

Dari contoh tersebut terlihat bahwa grup pemain mempunyai dua karakter, sebut saja karakter P1 dan P2. Grup lawan juga mempunyai dua karakter sebut saja karakter L1 dan L2. Status keempat karakter tersebut adalah sebagai berikut :

Tabel 3 Tabel status karakter

Status	P1	P2	L1	L2
Str/Mag	19	14	6	15
Skil	15	13	2	7
Spd	15	13	2	8
Luck	18	15	2	14
Def	15	9	1	4
Res	7	11	1	4
Senjata	Kampak	Sihir <i>anima</i>	Cakar	Cakar
Keahlian penggunaan senjata	3	5	-	-
Level	1	1	-	-

Senjata				
Mt	7	5	-	-
Hit	60	90	-	-
Crit	0	0	-	-

Keempat karakter tersebut berada pada medan terbuka, sehingga DEF dan AVO nya bernilai 0.

Berdasarkan kondisi tersebut, himpunan kandidat nya adalah :

{P1 menyerang L1}, {P1 menyerang L2},

{P2 menyerang L1}, {P2 menyerang L2}.

Setelah itu fungsi seleksi akan mulai bekerja, diawali dengan memanggil fungsi untuk menghitung tingkat optimalitas. Berikut ini tabel perhitungannya :

Tabel 4 Tabel perhitungan tingkat optimalitas

Tahap perhitungan	P1 vs L1	P1 vs L2	P2 vs L1	P2 vs L2
a.) Efek Senjata				
Efektifitas	-	-	-	-
Level senjata	+1	+1	+1	+1
Mt	+7	+7	+5	+5
Hit	+6	+6	+9	+9
Crit	-	-	-	-
b.) Efek Status				
Selisih Str-Def / Mag-Res	+18	+15	+13	+10
Selisih Skill	+13	+8	+11	+6
Selisih Spd	+13	+7	+11	+5
Selisih Luck	+16	+4	+13	+1
Selisih kemampuan penggunaan senjata	+3	+3	+5	+5
c.) Efek Medan				
DEF	-	-	-	-
AVO	-	-	-	-
Total	77	51	68	42

Dari hasil perhitungan tingkat optimalitas di atas dapat disimpulkan bahwa fungsi seleksi akan memilih {P1 menyerang L1}, kemungkinan dengan tingkat optimalitas paling tinggi.

Setelah itu kemungkinan tersebut akan diverifikasi dengan fungsi kelayakan. Keempat kemungkinan tersebut mungkin terjadi, karena kedua karakter P1 dan P2 memiliki langkah yang cukup dan senjata untuk melakukan serangan.

Hasil tersebut penulis coba bandingkan dengan hasil pada permainan sebenarnya. Hasil nya adalah sebagai berikut :

Tabel 5 Tabel hasil pada permainan sebenarnya

Hasil	P1 vs L1	P1 vs L2	P2 vs L1	P2 vs L2
Nilai serangan	25	22	18	15
Kemungkinan	97%	73%	100	97%

mengenai sasaran				
Kemungkinan menghasilkan serangan tiga kali lipat	5%	0	4%	0

Kemungkinan penyerangan yang dipilih oleh fungsi seleksi memang merupakan penyerangan paling optimal pada saat itu.

IV. KESIMPULAN

1. Algoritma *greedy* digunakan untuk mencari solusi optimum lokal dengan harapan solusi tersebut juga merupakan optimum secara global.
2. Jika membandingkan hasil perhitungan algoritma *greedy* dengan hasil aktual dari permainan, algoritma *greedy* berhasil memberikan suatu solusi kemungkinan penyerangan yang paling optimal pada suatu saat. Optimalitas suatu serangan tidak hanya dilihat berdasarkan nilai serangan, tapi juga kemungkinan mengenai serangan dan kemungkinan melakukan serangan tiga kali lipat sebab pada permainan ini penggunaan senjata terbatas.
3. Penerapan algoritma *greedy* yang telah dibahas masih membutuhkan penyempurnaan, sebab perkiraan rasio pengaruh dari suatu hal (misalnya status karakter) terhadap tingkat optimalitas belum 100% akurat.

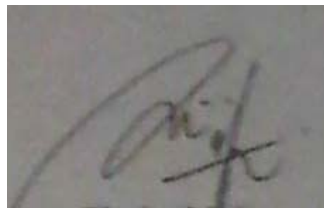
REFERENSI

- [1] Munir, Rinaldi. "Diktat Kuliah IF3051 Strategi Algoritma", Program Studi Teknik Informatika ITB, 2009.

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, 7 Desember 2010



Chairul Ichsan - 13508082