

Aplikasi Algoritma Greedy untuk Menyelesaikan Permainan Hedgewars

Diani Pavitri Rahasta¹, 13509021
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
¹13509021@std.stei.itb.ac.id, diani_pr@yahoo.com

Abstrak—Permainan Hedgewars merupakan permainan strategi dengan beberapa ketentuan senjata dan lawan. Dalam permainan ini, objektifnya adalah agar kita bisa menghancurkan tim lawan kita dengan senjata-senjata yang sudah disediakan. Dalam permainan ini ada beberapa mode permainan. Dalam makalah ini akan dibahas mengenai bagaimana cara memenangkan permainan ini dengan menggunakan algoritma *Greedy*. Algoritma *Greedy* dipilih karena dalam penerapan strategi untuk memenangkan permainan ini dibutuhkan teknik pemilihan lawan untuk diserang dan pemilihan senjata untuk digunakan. Dalam makalah ini akan dibahas mengenai permainan Hedgewars dalam mode *crazy*, yaitu mode permainan dengan senjata yang tidak terbatas. Pembahasan pemilihan strategi permainan dengan algoritma *Greedy* pada makalah ini dengan menggunakan asumsi bahwa faktor-faktor *random* dalam permainan Hedgewars, yaitu arah angin, sudut tembakan senjata, serta kekuatan penembakan senjata, diabaikan. Makalah ini hanya akan membahas strategi permainan Hedgewars secara konsep dalam melakukan pemilihan lawan dan senjata. Dalam makalah ini, yang berlaku sebagai subjek dari penerapan algoritma *Greedy* hanyalah pemain yang sedang aktif.

Index Terms—Algoritma *Greedy*, Hedgewars, *Health Point*, Lawan, Senjata, Strategi.

I. PENDAHULUAN

Permainan Hedgewars merupakan permainan bertipe strategi yang memiliki objektif menghancurkan semua lawan. Segi strategi dari permainan ini adalah dalam pemilihan lawan yang akan diserang dan senjata yang akan digunakan dalam melakukan penyerangan. Permainan ini merupakan permainan tim, di mana setiap pemain memiliki satu tim yang bisa dia jalankan dan beberapa tim yang menjadi lawannya.

Untuk menyelesaikan permainan ini dengan kemenangan, dibutuhkan strategi yang kuat dalam setiap giliran penyerangan. Dalam permainan ini, setiap orang dari setiap tim akan mendapat giliran untuk menyerang lawan secara bergantian. Dalam setiap giliran inilah algoritma *Greedy* akan diterapkan.

Pemilihan algoritma *Greedy* untuk menyusun strategi kemenangan diterapkan pada setiap giliran penyerangan. Algoritma *Greedy* akan digunakan untuk setiap giliran pada saat memilih lawan yang akan diserang pada saat itu dan senjata yang digunakan pada saat itu juga.

Algoritma *Greedy* dipilih untuk menyusun strategi kemenangan dalam permainan ini sebab permainan ini sesuai dengan sifat algoritma *Greedy* yaitu “*Take what you can get now*” [1]. Setiap giliran harus dibuat supaya memperoleh hasil maksimal untuk membantu perolehan hasil maksimal di akhir.

Dalam melakukan penyerangan terhadap lawan, yang harus dipertimbangkan adalah seberapa dekat lawan dengan kita, berapa banyak sisa *Health Point* yang dimiliki lawan, serta seberapa besar senjata yang kita pilih dapat mengurangi *health point* lawan. Dalam makalah ini, faktor-faktor *random* dalam permainan akan diabaikan. Faktor-faktor *random* tersebut antara lain kecepatan dan arah angin pada saat penyerangan, pengarahan senjata ke lawan, serta kekuatan penembakan senjata.

II. ALGORITMA GREEDY

Algoritma *Greedy* merupakan algoritma untuk penyelesaian persoalan optimasi. Algoritma ini memberikan langkah-langkah optimal pada saat penyelesaian persoalan. Pada setiap langkah penyelesaian masalah, algoritma ini akan menyeleksi pilihan langkah dan memilih langkah yang paling optimal pada saat itu. Dengan pemilihan langkah seperti itu, algoritma ini menganggap, dan berharap, setiap langkah optimal yang diambil akan berujung pada langkah optimal secara keseluruhan.

Ada beberapa elemen yang harus diperhatikan dalam penggunaan algoritma *Greedy* untuk menyelesaikan masalah. Berdasarkan referensi [1], elemen tersebut antara lain adalah himpunan kandidat, himpunan solusi, fungsi seleksi, fungsi kelayakan, dan fungsi obyektif. Himpunan kandidat merupakan seluruh elemen-elemen pembentuk solusi. Himpunan solusi merupakan kumpulan solusi dari setiap langkah dalam penyelesaian masalah. Fungsi seleksi merupakan fungsi yang digunakan untuk memilih isi dari himpunan solusi melalui seleksi pada himpunan kandidat. Fungsi kelayakan merupakan fungsi pembatas yang memeriksa kelayakan setiap elemen pada himpunan kandidat yang dikenai fungsi seleksi. Fungsi obyektif merupakan fungsi yang merupakan batasan optimum dari masalah yang akan diselesaikan.

Secara umum, skema algoritma *Greedy* adalah sebagai berikut. Tersedia n elemen dalam himpunan kandidat

sebagai objek seleksi dari algoritma *Greedy*. Elemen-elemen ini kemudian diseleksi dengan fungsi seleksi. Untuk setiap elemen yang lolos fungsi seleksi, himpunan kandidat dikurangi satu anggota. Elemen terpilih kemudian dikenai fungsi kelayakan, apakah elemen tersebut, dengan elemen-elemen di himpunan solusi, akan membentuk solusi optimal. Jika ya, elemen tersebut dimasukkan ke himpunan solusi, jika tidak, maka berakhirilah perjalanan elemen tersebut dalam penyelesaian masalah.

III. PENERAPAN ALGORITMA PADA PERMAINAN

A. Algoritma Greedy pada Permainan Hedgewars

Permainan Hedgewars merupakan permainan pengembangan dari permainan klasik Worms. Cara bermainnya hampir seluruhnya sama. Perbedaan permainan ini terletak pada tokoh utama permainan. Pada permainan Worms, tokoh utama permainan adalah sekumpulan cacing sedangkan pada permainan Hedgewars landak. Permainan Hedgewars memiliki tampilan antarmuka dan pilihan karakter yang lebih menarik. Selain itu, senjata-senjata yang digunakan dalam permainan Hedgewars berbeda dengan pada permainan Worms, meskipun beberapa senjata secara teknis penyerangan dan kekuatannya masih sama.

Peraturan permainan adalah sebagai berikut. Ada beberapa tim dalam permainan, dan tim yang menang adalah tim yang paling lama bertahan hidup sementara tim lainnya mati. Untuk mengalahkan tim lawan, harus dilakukan penembakan, atau penggunaan senjata, pada tim lawan. Permainan ini berakhir apabila tinggal satu tim yang bertahan hidup dan semua tim sisanya kehabisan pemain karena mati.

Dalam permainan ini, setiap tim akan diberi kesempatan sekali dalam setiap giliran. Pada setiap kesempatan tersebut, tim diberi kesempatan menyerang dengan satu orang dari tim tersebut diaktifkan untuk menyerang. Satu anggota tim yang diaktifkan tersebut diambil secara acak. Pada setiap giliran, setiap tim diberi kesempatan menyerang dengan urutan yang tidak berubah dari awal sampai akhir, tim A kemudian tim B, dan seterusnya.

Ada beberapa mode permainan dalam permainan Hedgewars. Setiap mode menentukan alur permainan dan senjata apa saja yang bisa dipakai. Dalam makalah ini, mode permainan yang digunakan adalah mode normal dengan mode senjata *crazy*. Pada mode ini, senjata yang bisa digunakan tidak terbatas. Pemain bisa menggunakan senjata manapun yang dia inginkan dan senjata tersebut tidak akan habis. Dengan menggunakan mode ini, pemain memang lebih mudah untuk menang. Namun jika permainan dilakukan dengan melawan komputer, akan sulit karena komputer biasanya jauh lebih menguasai penggunaan senjata dibanding pemain.

Pada awal permainan, pemain juga akan diminta menentukan berapa banyak tim yang bermain dan berapa banyak landak yang menjadi anggota dari suatu tim. Pemain juga memilih ukuran serta pola lokasi yang akan digunakan untuk bermain. Pada saat permainan dimulai,

landak-landak yang ikut permainan akan dimunculkan secara acak pada lokasi permainan yang dipilih. Lokasi landak pada lokasi permainan acak dan tidak bisa diubah kecuali landak tersebut diaktifkan untuk menyerang.



Gambar 1: Tampilan Pemilihan Mode Permainan dan Mode Senjata

Penerapan algoritma *Greedy* untuk penyusunan strategi kemenangan pada permainan ini dapat digunakan untuk tiga fungsi seleksi. Fungsi pertama adalah *Greedy by location*. Fungsi ini akan memperhitungkan lawan-lawan yang berada paling dekat dengan dirinya sekarang. Fungsi kedua adalah *Greedy by health point*. Fungsi ini akan mempertimbangkan lawan-lawan dengan *health point* yang paling sedikit. Fungsi ketiga adalah *Greedy by weapon choices*. Fungsi ini akan memperhitungkan senjata apa yang akan digunakan dalam giliran tersebut.

Ketiga fungsi seleksi untuk penerapan algoritma *Greedy* di atas hanya akan memberikan strategi secara konsep mengenai bagaimana seharusnya kita menyerang lawan. Namun, karena semua faktor *random* dari permainan ini diabaikan, maka penerapan algoritma *Greedy* belum bisa memberikan strategi teknis permainan mengenai bagaimana cara menembak dan lain sebagainya.

Secara umum, elemen-elemen algoritma *Greedy* dalam permainan Hedgewars adalah sebagai berikut.

Sebagai himpunan kandidat adalah seluruh landak dari seluruh tim selain pemain yang sedang aktif untuk menyerang. Himpunan solusi berisi landak-landak anggota tim lawan yang diserang. Fungsi seleksi, seperti sudah dijelaskan, akan ada tiga macam, seleksi berdasarkan lokasi landak berdiri, berdasarkan *health point* landak, serta berdasarkan senjata yang digunakan untuk menyerang. Fungsi kelayakan pada penerapan algoritma *Greedy* pada permainan ini adalah memeriksa apakah landak yang akan diserang merupakan landak anggota tim lawan atau bukan, selain itu juga dilihat apakah landak tersebut masih hidup, atau hanya tersisa mayatnya saja. Fungsi obyektif dari permainan ini jelas, untuk membunuh semua landak tim lawan.

Kompleksitas algoritma dari setiap algoritma juga akan diperhitungkan. Namun pada persoalan ini, kompleksitas algoritma hanya berpengaruh pada pengaplikasian algoritma pada program dan tidak terlalu berpengaruh pada penyusunan strategi untuk memenangkan permainan. Akan dibandingkan kompleksitas algoritma ketiga algoritma untuk mendapatkan algoritma yang paling mangkus.

B. Greedy by Location

Fungsi seleksi pertama yang dipilih untuk menyusun strategi kemenangan dalam permainan ini adalah *Greedy by location*. Seleksi landak yang akan diserang dilakukan berdasarkan seberapa dekat dia berada dengan landak yang sedang aktif menyerang. Fungsi seleksi ini dipilih sebab jika landak yang akan diserang berada pada jarak dekat, kekuatan senjata yang ditembakkan akan optimal dan tentunya landak sasaran akan mati lebih cepat.

Dalam permainan *Hedgewars*, kontur lokasi permainan tidak selalu rata. Bisa saja ada kasus di mana landak yang menjadi sasaran berada di lokasi yang dekat namun terhalang oleh dinding kontur dari lokasi yang digunakan. Hal ini bisa menjadi masalah namun bisa juga diabaikan. Untuk mengatasi hal ini, bisa digunakan senjata yang memang bisa menyerang menembus dinding kontur lokasi yang digunakan.

Langkah penentuan strategi yang digunakan adalah sebagai berikut. Pada saat landak sudah diaktifkan untuk menyerang pada giliran tersebut, periksa daerah sekeliling landak dengan radius 1 cm. Ulangi pemeriksaan hingga ditemukan landak lain di dekat lokasi tersebut. Jika landak tersebut adalah landak lawan, maka lakukan penyerangan terhadap landak tersebut. Jika landak tersebut adalah landak anggota tim yang sama dengan landak yang sedang aktif, lanjutkan pencarian landak pada radius 1 cm berikutnya. Jika lokasi landak terhalang dinding, gunakan senjata yang bisa menembus tembok atau bisa dilempar melewati tembok. Jika tidak, gunakan senjata apapun yang bisa digunakan untuk menyerang.

Pada fungsi seleksi ini, kompleksitas algoritma dihitung dari berapa kali pencarian dalam radius 1 cm dilakukan. Karena pencarian dilakukan terhadap radius daerah, maka pencarian dilakukan dengan mengunjungi setiap bagian dari lingkaran yang dibentuk dengan landak yang sedang aktif menyerang sebagai pusatnya.

Pencarian akan dilakukan setiap cm dengan luas pencarian adalah luas lingkaran dengan jari-jari dimulai dari 1 cm. Karena rumus luas lingkaran adalah $\pi \times (\text{jari-jari})^2$, maka kompleksitas algoritmanya adalah $O(n^2)$.

Algoritma ini akan menjadi kurang mangkus jika landak yang sedang aktif untuk menyerang berada pada posisi yang jauh dari manapun. Hal ini bisa terjadi jika pemain setiap tim sedikit dan lokasi yang digunakan besar. Namun akan menjadi sangat mangkus jika pemain yang digunakan sedikit dan lokasi yang digunakan kecil sehingga setiap landak tidak akan terpisah jauh.



Gambar 2: Pemilihan Landak yang Akan Diserang

C. Greedy by Health Point

Fungsi kedua yang dipilih untuk menerapkan algoritma ini adalah fungsi seleksi berdasarkan *health point* landak yang akan diserang. Pada saat permainan, setiap *health point* landak akan ditampilkan di atas kepala masing-masing landak di bawah namanya. *Health point* ini menunjukkan berapa lama lagi landak tersebut bisa bertahan untuk diserang. *Health point* maksimal seekor landak dalam permainan ini adalah 100.

Pada seleksi dengan fungsi *Greedy by health point* ini, pemilihan dilakukan berdasarkan landak mana yang memiliki *health point* paling kecil. Jika landak dengan *health point* minimal itu berada jauh dari landak yang sedang aktif menyerang, bisa digunakan senjata yang memang difungsikan untuk penyerangan jarak jauh. Pada fungsi ini, senjata yang digunakan juga dipertimbangkan, walaupun tidak spesifik. Namun sebaliknya, jarak landak yang akan diserang terhadap landak yang sedang aktif menyerang tidak akan diperhitungkan.

Langkah-langkah untuk menjalankan fungsi ini adalah sebagai berikut. Pada lokasi permainan, lakukan pencarian landak dan pengecekan nilai *health point*nya.

Simpan data lokasi landak dan *health point*nya ke dalam suatu tabel lalu lakukan *sorting* untuk mencari landak dengan *health point* minimal. Jika bertemu landak dengan *health point* minimal, periksa apakah landak tersebut adalah landak tim lawan. Jika landak tersebut adalah landak dari tim yang sama dengan landak yang sedang aktif menyerang, maka periksa landak dengan *health point* kedua minimal dan seterusnya sampai bertemu dengan landak yang merupakan landak dari tim lawan untuk diserang. Jika landak tersebut terhalang dinding dari landak yang sedang aktif menyerang, gunakan senjata yang bisa menyerang dengan menembus dinding. Jika landak tersebut berada sangat jauh dari landak yang sedang aktif menyerang, gunakan senjata yang bisa menyerang jarak jauh. Selain itu, gunakan senjata apapun yang bisa digunakan untuk menyerang.

Pada fungsi seleksi ini, kompleksitas algoritma dihitung dari fungsi *sorting* yang dilakukan ditambah dengan pencarian pada tabel. Jika *sorting* dilakukan dengan *merge sort*, kompleksitas algoritmanya adalah $O(n \log n)$.

Algoritma ini bisa dibilang mangkus untuk segala situasi. Hanya saja, harus diberi kondisi untuk senjata yang digunakan pada saat landak yang akan diserang sudah dipilih.



Gambar 3: Tampilan Health Point di Atas Kepala Landak

D. Greedy by Weapon Choices

Fungsi ketiga dari aplikasi algoritma *Greedy* pada permainan Hedgewars adalah fungsi seleksi berdasarkan senjata yang digunakan. Dalam permainan Hedgewars, senjata-senjata yang tersedia diklasifikasikan dalam

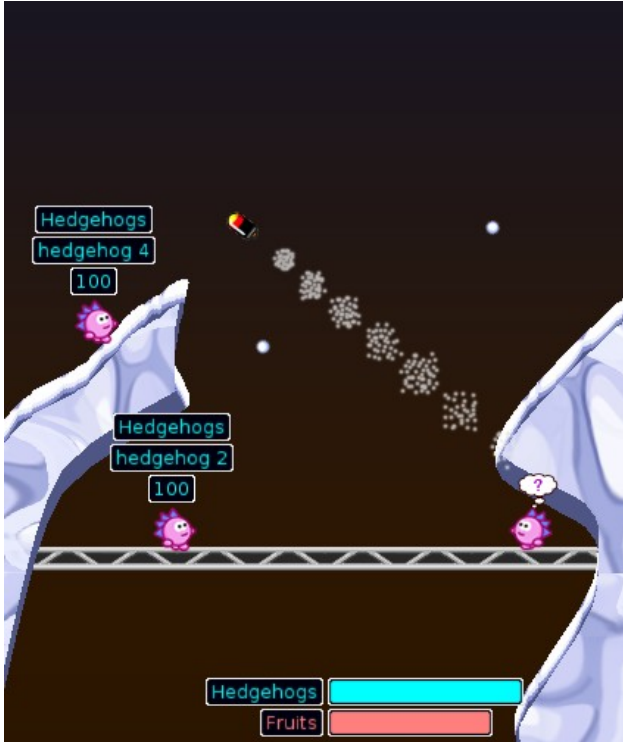
tujuh kategori. Ada tiga kategori lain yang merupakan pelengkap, yaitu alat untuk berpindah tempat, alat untuk mendapatkan keuntungan lebih dari penyerangan, dan alat untuk membantu memudahkan penyerangan. Kategori senjata antara lain adalah kategori senjata tembak jarak jauh, kategori senjata granat dan bom yang dilempar, kategori senjata pistol, kategori senjata jarak dekat yang tidak meledak, kategori senjata jarak dekat dan meledak, kategori senjata serangan udara, dan kategori senjata menembus dinding. Masing-masing senjata memiliki nilai perusakan yang berbeda-beda.



Gambar 4: Kategori Senjata

Seleksi fungsi *Greedy by weapon choices* ini dilakukan dengan cara memilih senjata apa yang paling cocok digunakan pada situasi tertentu. Situasi tertentu yang dimaksud adalah lokasi landak yang akan diserang dan sisa *health point* yang dimiliki landak tersebut. Pemilihan dilakukan sesuai dengan kategori senjata yang ada. Penyerangan jarak jauh yang tidak dihalangi apapun bisa dilakukan dengan senjata kategori senjata tembak jarak jauh. Penyerangan jarak dekat yang tidak dihalangi apapun bisa dilakukan dengan senjata kategori senjata pistol, jarak dekat meledak maupun tidak meledak. Jika landak yang sedang aktif menyerang bisa leluasa untuk pergi setelah menyerang jarak dekat, maka lebih baik menggunakan kategori senjata jarak dekat dan meledak. Untuk serangan jarak dekat dengan halangan dinding tipis, kategori pistol bisa digunakan. Untuk serangan jarak dekat dengan dinding tebal bisa digunakan kategori senjata menembus tembok. Jika landak terpisah kontur ketinggian, bisa digunakan senjata kategori granat dan bom yang dilempar. Jika landak yang akan diserang berada sangat jauh dari landak yang sedang aktif menyerang, dapat digunakan senjata serangan udara.

Fungsi seleksi ini dapat dijalankan sebagai berikut. Pada saat giliran landak diaktifkan, pilih secara random landak yang akan diserang. Pastikan yang dipilih secara *random* hanya landak anggota tim lawan. Selanjutnya lihat lokasi dan sisa *health point* landak terpilih, lalu lakukan pemilihan senjata berdasarkan kategori yang sudah dipaparkan di atas.



Gambar 5: Tampilan Penggunaan Senjata Tembak Jarak Jauh

Pada fungsi seleksi ini, penghitungan kompleksitas algoritma dilakukan dengan memperhitungkan proses *random* dan pemilihan senjata. Proses *random* dijalankan sebanyak n kali, ditambah pemilihan senjata sebanyak m kali. Kompleksitas algoritma dari fungsi seleksi ini adalah $O(mn)$.

Algoritma ini bisa dibilang mangkus untuk segala situasi. Algoritma ini juga bisa digunakan untuk melengkapi algoritma-algoritma sebelumnya untuk fungsi pemilihan senjata.

IV. ANALISIS PENERAPAN ALGORITMA

Algoritma *Greedy* dapat diterapkan untuk menyusun strategi meskipun hanya secara konsep. Algoritma ini cukup fleksibel untuk bisa menangani masalah penyusunan strategi pada permainan HedgeWars. Setiap fungsi yang dipilih untuk mengaplikasikan algoritma ini bisa diaplikasikan dengan baik.

Berdasarkan hasil pembahasan dari setiap fungsi seleksi dari algoritma *Greedy* yang diterapkan, analisis yang dapat ditarik adalah sebagai berikut. Untuk fungsi seleksi berdasarkan lokasi, algoritma yang digunakan sudah cukup mangkus, namun masih harus ditambah dengan fungsi yang tepat untuk pemilihan senjata. Untuk

fungsi seleksi berdasarkan *health point*, algoritma yang digunakan sudah mangkus namun masih bisa dikembangkan lagi. Sementara untuk fungsi ketiga dari segi algoritma dan penerapannya sejauh ini tidak ada masalah yang cukup mengganggu.

Dari segi kompleksitas algoritma, dapat dilihat bahwa fungsi seleksi berdasarkan *health point* minimal memiliki kompleksitas algoritma paling rendah. Hal ini disebabkan karena proses perbandingan yang dilakukan pada fungsi ini memang hanya sedikit.

V. KESIMPULAN

Ketiga fungsi seleksi yang digunakan dalam pengaplikasian algoritma *Greedy* dalam permainan HedgeWars memiliki kelebihan dan kekurangan masing-masing. Dilihat dari segi kemangkusan algoritma, akan lebih baik jika ketiga algoritma itu digabungkan. Anggap algoritma utamanya adalah algoritma dengan fungsi *Greedy by health point*, namun pencarian dilakukan dengan fungsi seleksi *Greedy by location*. Data landak yang disimpan di tabel kemudian ditambahkan dengan data lokasi landak tersebut. Kemudian untuk menentukan penyerangan, digunakan fungsi seleksi *Greedy by weapon choices*.

Dari hasil pembahasan, dapat dilihat bahwa algoritma *Greedy* cocok digunakan untuk menyusun strategi kemenangan pada permainan HedgeWars. Namun seperti sudah dijelaskan di awal, masih banyak faktor lain yang seharusnya dipertimbangkan. Untuk pembuatan *artificial intelligence* permainan ini, masih banyak hal yang harus dikaji terkait dengan penerapan strategi secara teknis.

REFERENCES

- [1] Munir, Rinaldi. 2009. *Diktat Kuliah IF3051 Strategi Algoritma*, Program Studi Teknik Informatika STEI ITB.
- [2] <http://www.hedgeWars.org/node/1107>
- [3] <http://www.hedgeWars.org/screenshots.html>
- [4] <http://www.hedgeWars.org/node/2624>

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

Diani Pavitri Rahasta
13509021