

Aplikasi Algoritma Greedy untuk Menyelesaikan Permainan Magic Wingdom

Muhammad Iqbal 13510064
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13510064@std.stei.itb.ac.id

Abstrak—Magic Wingdom adalah salah satu permainan *arcade puzzle* yang berplatform *mobile* dan dapat dimainkan di sistem operasi Android. Permainan ini cukup sederhana namun tidak mudah untuk dimainkan sehingga dapat membuat pemain memutar otak. Tujuan permainan ini adalah memasang telur yang ada di *screen* ke yang lainnya agar tidak saling bertabrakan. Makin lama permainan maka makin banyak telur yang ada sehingga meningkatkan kesulitan permainan. Dalam studi ini akan dibahas strategi untuk menyelesaikan permainan ini dengan pendekatan *greedy* yaitu *greedy by location* dan *greedy by point*. *Greedy by location* berprinsip mengambil rangkaian telur yang posisinya sudah hampir bertabrakan dengan telur lain. *Greedy by point* berprinsip mencocokkan telur yang paling banyak terdapat di layar sehingga akan menimbulkan poin yang banyak serta dapat membuat layar lebih kosong. Kedua strategi ini memberikan hasil yang mendekati hasil optimal.

Kata kunci— *greedy*, optimisasi, poin, Magic Wingdom.

I. PENDAHULUAN

Magic Wingdom adalah sebuah permainan *arcade puzzle visual-matching* yang dibuat oleh PlayCreek Games. *Arcade games* adalah permainan sederhana yang mudah dimengerti dan biasanya dimainkan untuk menghabiskan waktu, permainan Magic Wingdom ini cukup atraktif dengan grafis yang bagus dan gameplay yang cukup unik yaitu menggabungkan konsep *line-drawing* dan konsep *marble-matching*. Dalam permainan ini pemain akan mencocokkan telur dengan warna yang sama sehingga akan menghindarkan telur saling bertabrakan dengan warna yang berbeda. Cara mencocokkan telur ini adalah dengan menarik line dan telur akan bergerak sesuai line yang telah digambar.. Karena itulah permainan Magic Wingdom ini cukup unik karena melibatkan konsep *line-drawing* pada *marble-matching*.

Objektif dari permainan ini adalah bertahan dalam kurun waktu tertentu dengan menghindarkan telur saling bertabrakan. Beberapa tampilan permainan Magic Wingdom :



Gambar 1 Gambar telur telah ditarik dari satu tempat ke tempat lain untuk menghindari tabrakan



Gambar 2 Gambar pemindahan telur ke tempat lain dengan penggambaran line

Elemen dalam permainan ini adalah :

1. Level meter, menunjukkan seberapa jauh progress dari permainan. Jika level meter ini sudah penuh maka akan berlanjut ke level berikutnya dimana permainan akan dibuat lebih sulit dan cepat.
2. Life point, menunjukkan sisa nyawa yang ada untuk pemain. Jika life point ini habis maka permainan akan berakhir.
3. Telur, merupakan elemen utama dalam permainan. Dalam permainan ini terdapat warna-warni elemen telur yang jika dua telur dengan warna yang berbeda bertabrakan maka life point akan berkurang.
4. Elemen bonus, merupakan elemen tambahan yang jika telur bertabrakan dengannya akan menimbulkan efek tambahan tertentu. Contohnya akan meledakkan telur di area tertentu. Ada juga elemen yang cukup disentuh dengan tangan maka akan muncul efeknya.
5. Poin, menunjukkan seberapa banyak poin yang telah didapat. Poin didapat dari proses pencocokan telur. Poin ini akan meningkat dengan adanya *combo* yaitu tabrakan beruntun telur dengan warna yang sama.

Tujuan dari studi ini adalah untuk pencarian cara optimal dalam bertahan agar dapat melewati level dengan baik dengan pendekatan algoritma *Greedy*.

Algoritma *greedy* merupakan salah satu metode paling populer untuk menyelesaikan masalah persoalan optimasi. Yaitu permasalahan yang mencari solusi optimal, baik itu bersifat maksimalisasi atau minimalisasi.

Algoritma ini akan membentuk solusi langkah per langkah, dimana pada setiap langkah akan diputuskan kemungkinan mana yang paling baik. Pilihan ini nantinya akan disebut sebagai solusi optimal local. Namun tidak selalu kumpulan solusi lokal ini akan menghasilkan solusi optimal, karena bisa saja solusi optimal global terdiri dari solusi lain yang berbeda.

Dalam permainan Magic Wingdom ini, tiap langkah akan menghasilkan kondisi yang baru. Dengan demikian akan tercipta kemungkinan solusi yang baru.

II. DASAR TEORI

Dalam Algoritma Greedy terdapat elemen-elemen berikut:

1. Himpunan kandidat, C
2. Himpunan solusi, S
3. Fungsi seleksi, SELEKSI()
4. Fungsi kelayakan, LAYAK()
5. Fungsi obyektif, SOLUSI()

Himpunan kandidat merupakan himpunan kandidat solusi yang merepresentasikan opsi yang ditawarkan setiap langkahnya. Himpunan solusi merepresentasikan langkah apa saja yang telah diambil hingga akhir

algoritma.

Fungsi seleksi menyeleksi kumpulan pilihan yang ada dan mengembalikan pilihan yang akan menghasilkan solusi optimal. Fungsi kelayakan bertindak sebagai pembatas, apakah pilihan yang dipilih layak sesuai ketentuan dari persoalan. Kombinasi dari kedua fungsi ini akan menghasilkan solusi optimum untuk setiap langkahnya. Solusi-solusi optimum lokal tersebut akan menjadi hasil dari sebuah fungsi objektif, yaitu persoalan utama yang ingin diselesaikan.

Berikut adalah skema umum dari algoritma Greedy:

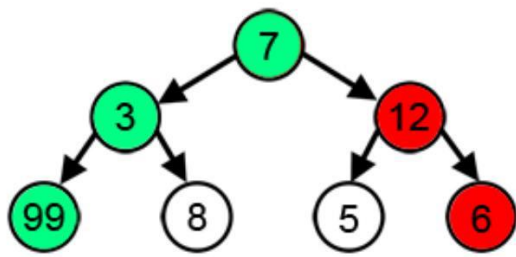
```
function greedy(input C: himp_kandidat)
  → himp_kandidat
  { Mengembalikan solusi dari persoalan optimasi
  dengan algoritma greedy.
  Masukan: himpunan kandidat C
  Keluaran: himp. Solusi yang bertipe
  himp_kandidat }
Deklarasi
  x : kandidat
  S : himp_kandidat
Algoritma
  S ← { } {inisialisasi S dengan kosong}
  while (not SOLUSI(S)) and (C!={}) do
    x ← ELEKSI(C) {pilih kandidat dari C}
    C ← C - {x} {elemen C berkurang satu}
    if LAYAK(S U {x}) then
      S ← S U {x}
    endif
  endwhile
  {SOLUSI(S) atau C kosong}
  if SOLUSI(S) then
    return S
  else
    {tidak ditemukan solusi global}
  endif
```

Pertama-tama, algoritma greedy akan mengiterasi semua elemen himpunan kandidat yang ada dan menyeleksi kandidat yang akan menghasilkan solusi optimal lokal. Kemudian dilakukan pengecekan dahulu terhadap kandidat tersebut dengan fungsi kelayakan, apakah sudah sesuai dengan ketentuan persoalan. Jika sudah cocok, maka kandidat dimasukkan ke dalam himpunan solusi.

Hal ini terus dilakukan periterasi sampai elemen di himpunan kandidat habis, solusi telah ditemukan atau telah mencapai kondisi akhir yang diinginkan.

Dalam permainan Magic Wingdom ini, pemain dituntut untuk menghindari telur saling bertabrakan. Ada dua cara optimal yang dapat dilakukan, yaitu dengan cara menghindari telur yang sudah akan bertabrakan, atau mencocokkan telur dengan jumlah terbanyak terlebih dahulu sehingga akan menciptakan layar yang lebih leluasa.

Actual Largest Path Greedy Algorithm



Gambar 3 Ilustrasi algoritma greedy

Dalam gambar ilustrasi diatas, dicari nilai dengan poin terbanyak. Pada kasus diatas, diterapkan algoritma *greedy* dimana di setiap pilihan dilakukan pencarian solusi optimal. Dalam percabangan pertama, dipilih 12 sebagai nilai paling besar, lalu pada percabangan kedua dipilih 6. Apakah solusi ini solusi terbaik? Ternyata tidak, karena dalam alternative lain terdapat poin yang esbsar yaitu 99.

Ilustrasi ini menggambarkan bahwa algoritma *greedy* tidak selalu menghasilkan solusi optimal global. Karena yang dilakukan algoritma *greedy* hanya pencarian solusi optimal dalam suatu pilihan saja tanpa mempertimbangkan pilihan alternative sebelumnya.

III. IMPLEMENTASI

Seperti yang disebutkan sebelumnya, pemenuhan objektif dari permainan Magic wingdom ini dapat diwujudkan dengan menerapkan penyelesaian dengan algoritma *Greedy*. Pada bagian ini akan dibahas bagaimana mencari cara tercepat dalam pencarian poin dan konsep survival yang paling optimal.

Metode yang diterapkan akan menggunakan konsep *greedy* yaitu :

1. *Greedy by Location*
2. *Greedy by Point*

A. *Greedy by Location*

Strateginya adalah bagaimana memusnahkan telur yang memiliki okasi berdekatan dengan telur berbeda warna terlebih dahulu. Meskipun poin yang didapat tidak terlalu banyak namun dengan cara ini dapat dipastikan pemain akan dapat bertahan hingga waktu habis.

Pada setiap langkah, sistem akan mencatat jumlah setiap telur dalam layar, serta warna dan lokasi dari telur tersebut. Semua telur ini mempunyai prioritas yang sama.

Sistem akan mencocokkan telur jika telur tersebut telah menemukan warna pasangannya didalam layar, dan telah menempuh selisih lokasi yang telah ditentukan ke telur dengan warna yang berbeda.

Elemen-elemen greedy :

1. Himpunan kandidat

Seluruh rangkaian telur tunggal yang berada didalam layar. Setiap rangkaian telur memiliki prioritas sesuai lokasi masing-masing

2. Himpunan solusi

Kumpulan dari kandidat yang memiliki nilai paling besar.

3. Fungsi seleksi

Memilih salah satu dari himpunan kandidat yang berwarna sama dan meng-assign nilainya berdasar prioritas.

4. Fungsi kelayakan

Memeriksa apakah kandidat bisa dibuat line untuk ditabrakkan atau tidak. Artinya kandidat tidak berada di jalur yang sulit atau tidak bisa dicapai..

5. Fungsi obyektif

Memilih kandidat yang memiliki nilai terbesar. Jika terdapat lebih dari satu kandidat, maka kandidat dipilih secara acak.



Gambar 4 Pencocokkan telur dengan prioritas lokasi yang sudah rawan

B. *Greedy by Point*

Strateginya adalah bagaimana memusnahkan telur dengan memperhatikan jumlah telur yang sama. Pemusnahan telur yang sama ini akan memberikan poin yang besar dan dapat memudahkan permainan dengan memusnahkan banyak telur sekaligus. Dengan banyaknya pemusnahan telur, maka level meter akan terisi dengan penuh sehingga level akan berakhir lebih cepat. Misalkan saja dalam satu layar terdapat banyak telur, dimana hanya terdapat beberapa warna, namun terdapat satu warna dominan. Jika kita memilih memusnahkan / mencocokkan warna ini terlebih dahulu maka layar akan lebih leluasa untuk dimainkan dan kedatangan telur berikutnya dapat diantisipasi dengan lebih baik.

Pada setiap langkah, sistem akan mencatat jumlah

telur dan warna dari masing-masing telur. Setiap telur baru muncul di layar, maka jumlah warna dari telur tersebut akan bertambah satu. Terdapat pula elemen bonus yang memiliki prioritas lebih tinggi, sehingga apabila terdapat elemen ini maka elemen inilah yang dimusnahkan terlebih dahulu karena memiliki efek yang menguntungkan bagi pemain.

Elemen bonus memiliki prioritas tertinggi, disusul dengan telur dengan jumlah warna yang sama terbanyak. Prioritas pemusnahan/pencocokkan telur selanjutnya dihitung berdasarkan telur dengan jumlah telur di layar yang memiliki warna sama terbanyak. Sehingga telur dengan pasangan paling sedikit akan memiliki prioritas terendah.

Sistem akan memusnahkan telur dengan prioritas tertinggi lebih dahulu.

Elemen-elemen greedy :

1. Himpunan kandidat

Seluruh rangkaian telur tunggal yang berada didalam layar. Setiap rangkaian telur memiliki prioritas sesuai jumlah pasangan dengan warna yang sama didalam layar. Serta elemen bonus yang memiliki prioritas tertinggi.

2. Himpunan solusi

Kumpulan dari kandidat yang memiliki nilai paling besar.

3. Fungsi seleksi

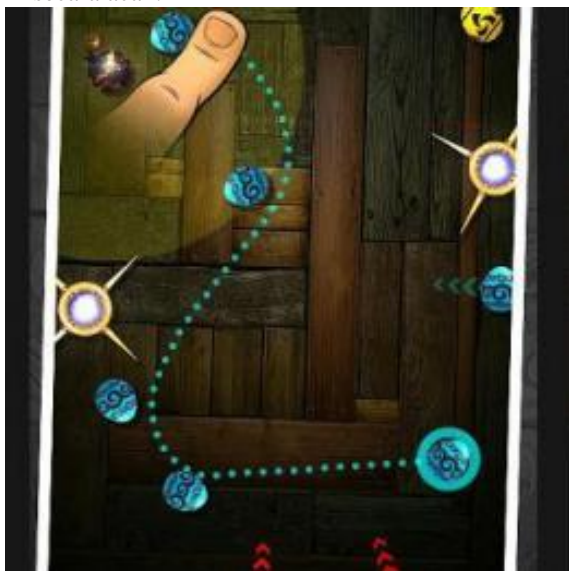
Memilih salah satu dari himpunan kandidat yang berwarna sama dan meng-assign nilainya berdasar prioritas.

4. Fungsi kelayakan

Memeriksa apakah kandidat bisa dibuat line untuk ditabrakkan atau tidak. Artinya kandidat tidak berada di jalur yang sulit atau tidak bisa dicapai..

5. Fungsi obyektif

Memilih kandidat yang memiliki nilai terbesar. Jika terdapat lebih dari satu kandidat, maka kandidat dipilih secara acak.



Gambar 5 Pencocokkan telur berdasarkan jumlah telur dengan warna yang sama terbanyak

IV. ANALISIS

Pada awal permainan, dimana telur yang ditimbulkan / dimunculkan oleh layar masih sedikit :



Gambar 6 Awal permainan

1. Dengan algoritma *greedy by location*, maka sistem akan memusnahkan telur dengan lokasi yang berdekatan terlebih dahulu. Pada awal permainan hal ini tidak terlalu sulit karena kuantitas telur yang belum terlalu banyak sehingga mudah mencari jalan untuk menabrakkan telur dengan warna yang sama.
2. Dengan algoritma *greedy by point*, maka sistem akan berlaku hampir sama dengan algoritma diatas. Karena kuantitas telur masih sedikit maka efektifitas dari algoritma ini tidak terlalu terasa.

Pada klimaks permainan , dimana kuantitas telur sudah banyak dan kemunculan telur sangat cepat :



Gambar 7 Klimaks permainan

1. Dengan algoritma *greedy by location*, maka sistem akan memusnahkan sesuai prioritas lokasi, namun

dengan kuantitas yang sudah banyak, maka sistem akan menemukan kesulitan dalam prioritasasi pemusnahan telur karena telur akan saling berdekatan satu sama lain.

2. Dengan algoritma *greedy by point*, maka sistem akan memusnahkan telur dengan jumlah terbanyak dahulu. Sehingga akan terjadi pemusnahan *consecutive* yang akan berakibat positif. Dengan adanya pemusnahan beruntun ini maka layar akan lebih luas sehingga memudahkan permainan. Apabila elemen bonus sudah bertebaran, maka elemen bonus ini akan dimusnahkan/diambil terlebih dahulu karena akan memudahkan pemain.

V. KESIMPULAN

Berdasarkan penerapan algoritma *greedy* pada permainan ini, ada dua cara untuk mengoptimalkan penyelesaiannya, yaitu :

1. *Greedy by location*.
2. *Greedy by point*.

Namun dari kedua cara ini , cara kedua dapat dikatakan lebih efektif dalam menyelesaikan permainan karena fleksibilitas kondii yang dapat ditempuh oleh cara ini. Cara pertama hanya efektif dalam awal sampai tengah permainan.

Dengan penerapan algoritma *greedy* ini pada game Magic Wingdom, terbukti dapat cukup efektif untuk menyelesaikan permainan ini.

REFERENCES

- [1] Munir, Rinaldi. 2009. *Strategi Algoritma*. Bandung : Penerbit ITB. Waktu akses 20 Desember 2012
- [2] http://en.wikipedia.org/wiki/Greedy_algorithm . Waktu akses 20 Desember 2012

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, 21 Desember 2012

ttd



Nama dan NIM