

Penerapan Algoritma *Greedy* Dalam Pemilihan Formasi *Game PES 2017*

(Studi Kasus: *Greedy by Player's Rating and Condition*)

Ridho Daffasyah / 13519038

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10 Bandung

E-mail (gmail): ridhodaffasyah01@gmail.com

Abstract—Pro Evolution Soccer atau biasa disebut PES merupakan salah satu permainan bertema sepakbola yang diterbitkan oleh perusahaan *game* di Jepang yaitu KONAMI. Permainan ini diluncurkan pada PS2 oleh Konami pada tahun 2001 sebagai Pro Evolution Soccer dan masih berkembang hingga saat ini. Permainan ini mempunyai fitur yang unik yaitu kondisi dan *rating* dari setiap pemain. Dengan adanya fitur tersebut, maka kita yang berperan sebagai pelatih atau pengatur strategi harus menyusun dan mengatur pemain sesuai dengan kondisi dan *rating* tersebut. Pada makalah ini, akan dibahas bagaimana sebuah konsep strategi algoritma yaitu *Greedy* bisa membantu kita untuk mengatur formasi pada PES 2017 .

Keywords—*sepakbola; kondisi; rating; PES; formasi; Greedy;*

I. PENDAHULUAN

Pro Evolution Soccer merupakan seri permainan ber-genre sepak bola yang dikembangkan dan diterbitkan oleh Konami. PES ini pertama kali dirilis pada tahun 2001 dan diluncurkan pada konsol permainan PS2. PES 2017 merupakan angsuran ke-16 di seri *Pro Evolution Soccer* yang dirilis pada 15 September 2016. Permainan PES ini bukan hanya sekadar permainan yang dimainkan biasa saja, akan tetapi sudah dijadikan ajang perlombaan di seluruh dunia.



Gambar 1.1 PES 2017

<https://www.konami.com/games/eu/en/topics/13924/>

PES 2017 ini merupakan permainan sepakbola yang bisa dimainkan secara *offline* dan *online*. PES 2017 mempunyai beberapa mode permainan seperti *myClub*, *Master League*, *Become a Legend*, *Exhibiton Match*, *Versus Mode*, *Cup*, *Training mode* dan *Online Match*. Selain mode permainan,

diberikan juga mode *extra* yaitu *edit player*, *edit team*, *edit manager*, *edit stadium*, *edit transfer player from one team to another team* dan *team statistic*.



Gambar 1.2 Mode Permainan PES 2017

Pada kali ini akan dibahas mengenai mode permainan *Exhibiton Match*. *Exhibiton Match* merupakan mode permainan dimana kita bisa bermain melawan CPU dan 2nd *Player* ataupun bisa juga kita sebagai pelatih dan tim yang kita latih akan dimainkan oleh CPU yang akan melawan CPU. Di mode permainan ini selain kita bisa memilih tim sendiri, disediakan juga fitur *random team* yang dimana akan menentukan tim yang akan kita pakai sesuai hasil random tersebut. Selain fitur *random team*, disediakan juga fitur *coach mode* yang membuat kita menjadi pelatih dan tim yang kita atur dimainkan oleh CPU.



Gambar 1.3 Mode *Exhibiton Match* PES 2017

Sebelum dimulai pertandingan, kita harus memilih terlebih dahulu tim mana yang akan kita gunakan atau kita latih. Setelah memilih tim untuk kita, selanjutnya memilih tim mana yang akan kita lawan. Setelah itu, kita diberikan pilihan untuk mengatur strategi (formasi) apa yang akan kita gunakan untuk melawan tim musuh. Dalam pengaturan formasi tersebut, kita diberikan fitur kondisi dan rating dari setiap pemain pada tim kita dan tim musuh. Sehingga kita bisa melihat pemain-pemain mana saja yang sedang tidak baik performanya ataupun *ratingnya* memang kecil yang artinya masih kurang berpengalaman baik di tim kita maupun tim musuh. Oleh karena itu, yang menjadi bahan pertimbangan dalam memilih formasi didasarkan pada kondisi dan *rating* pemain itu sendiri.

Untuk para pemain pemula, banyak diantaranya masih bingung dalam melakukan pengaturan strategi (formasi) tersebut. Karena kondisi dan *rating* cukup mempengaruhi pola permainan dari para pemain bola tersebut. Maka dari itu diperlukan pemahaman terhadap formasi tersebut.

Algoritma *Greedy* adalah salah satu jenis strategi algoritma yang seringkali digunakan dalam memecahkan persoalan optimasi. Algoritma ini sering digunakan untuk kasus yang membutuhkan waktu penyelesaian cepat dan tidak harus mangkus namun dapat memberikan hasil yang tepat. Maka dari itu, dalam makalah ini penulis menggunakan algoritma *greedy* untuk membantu menemukan formasi yang optimum dalam mode permainan *Exhibiton Match* pada *Pro Evolution Soccer* (PES) 2017.

II. DASAR TEORI

A. Algoritma Greedy

Algoritma *Greedy* merupakan metode yang paling populer dan sederhana untuk memecahkan persoalan optimasi. Dalam pengertian lain yaitu algoritma yang memecahkan persoalan secara langkah per langkah (step by step) sedemikian sehingga, pada setiap langkah :

1. Mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu tanpa memperhatikan konsekuensi ke depan.
2. dan “berharap” bahwa dengan memilih optimum lokal pada setiap langkah akan berakhir dengan optimum global.

Elemen-elemen algoritma *greedy* :

1. Himpunan kandidat,
C : berisi kandidat yang akan dipilih pada setiap Langkah (misal: simpul/sisi di dalam graf, job, task, koin, benda, karakter, dsb)
2. Himpunan solusi,
S : berisi kandidat yang sudah dipilih.
3. Fungsi solusi
Menentukan apakah himpunan kandidat yang dipilih sudah memberikan solusi.
4. Fungsi seleksi (selection function)

Memilih kandidat berdasarkan strategi *greedy* tertentu. Strategi *greedy* ini bersifat heuristik.

5. Fungsi kelayakan (feasible)

Memeriksa apakah kandidat yang dipilih dapat dimasukkan ke dalam himpunan solusi (layak atau tidak).

6. Fungsi obyektif

Memaksimumkan atau meminimumkan.

Algoritma *greedy* tidak beroperasi secara menyeluruh terhadap semua kemungkinan solusi yang ada. Terdapat beberapa fungsi SELEKSI yang berbeda, sehingga kita harus memilih fungsi yang tepat jika kita ingin algoritma menghasilkan solusi optimal.

Jadi, pada sebagian persoalan, algoritma *greedy* tidak selalu berhasil memberikan solusi yang optimal, namun sub-optimal. Namun bila algoritma *greedy* dapat menghasilkan solusi optimal, maka keoptimalannya itu harus dapat dibuktikan secara matematis. Membuktikan optimalitas algoritma *greedy* secara matematis adalah tantangan tersendiri. Lebih mudah memperlihatkan algoritma *greedy* tidak selalu optimal dengan menunjukkan *counterexample*.

Kelebihan dari Algoritma *greedy* dibandingkan dengan algoritma *exhaustive search* ialah lebih cepat, karena semua alternatif solusi tidak harus diproses melainkan hanya cukup menimbang solusi optimum lokal saja. Selain itu, algoritma *greedy* merupakan algoritma yang sederhana sehingga mudah dipahami.

B. Susunan Formasi dalam Permainan Sepak Bola

Sepakbola adalah olahraga yang dimainkan oleh 11 orang per tim atau biasanya jika dibuat dalam formasi hanya 10 pemain dengan posisi tertentu dan 1 pemain dengan posisi penjaga gawang. Formasi sepakbola dirancang untuk melakukan pengeksekusian strategi. Formasi ini diatur dalam 4 lini, yaitu depan, tengah depan, tengah belakang, dan belakang.

Lini depan memiliki tugas untuk menyerang dan mencetak gol sebanyak-banyaknya. Lini tengah depan dan belakang memiliki tugas untuk menghubungkan pemain depan dan pemain belakang dan juga untuk mengalirkan tempo serangan dari lini belakang. Lini belakang memiliki tugas untuk menjaga dan mengatur pertahanan tim sehingga tidak kemasukan gol

Terdapat banyak sekali formasi yang dipakai oleh pelatih-pelatih terkenal di dunia. Susunan formasi biasanya digambarkan sebagai berikut :

 - <TB> - <TD> - <D>

Keterangan :

B : Jumlah pemain belakang

TB : Jumlah pemain tengah belakang

TD : Jumlah pemain tengah depan

D : Jumlah pemain depan



Gambar 2.1 Salah satu contoh formasi 4-3-2-1

<https://denmono.wordpress.com/2017/05/28/formasi-4-2-3-1-dalam-dunia-sepak-bola/>

C. Pro Evolution Soccer 2017






PES 2017 menyediakan fitur kondisi dan *rating* dari masing-masing pemain di setiap tim. Fitur ini bisa digunakan ketika sedang mengatur formasi di sebelum pertandingan dimulai.



Gambar 2.2 Fitur kondisi dan *rating* pemain

Rating menggambarkan atribut dari seorang pemain seperti kekuatan, kelincahan, kecepatan, keseimbangan, dan lain-lain. Selain memiliki *rating*, pemain juga memiliki kondisi dimana kondisi ini menentukan apakah pemain tersebut bugar atau tidak.

Pada PES 2017 ini, fitur kondisi bisa direpresentasikan sebagai berikut :

-  : Pemain dalam kondisi sangat bugar
-  : Pemain dalam kondisi bugar
-  : Pemain dalam kondisi seimbang atau normal
-  : Pemain dalam kondisi tidak bugar
-  : Pemain dalam kondisi sangat tidak bugar

Dari data kondisi diatas, pemain harus menentukan formasi yang sesuai dengan kondisi dan *rating* dari pemain

yang ia miliki. Jika menggunakan formasi yang tidak optimal, maka permainan tim pun tidak optimal.

III. PERANCANGAN ALGORITMA

A. Algoritma Greedy

Algoritma *greedy* yang digunakan dalam persoalan ini adalah *Greedy by rating and condition of player*. Keputusan pemilihan formasi didasarkan pada *rating* dan kondisi pemain, yaitu diambil *rating* yang tertinggi yang menempati posisi itu yang dimana pemain itu memiliki kondisi minimal kurang bugar.

Langkah-langkah algoritma *greedy* yang digunakan sebagai berikut :

1. Memisahkan pemain-pemain berdasarkan posisi yang dimiliki dan memasukkannya ke list berdasarkan posisi tersebut (4 list posisi formasi) dan lakukan sorting berdasarkan *rating* dan kondisi pemain dan juga eliminasi pemain yang sedang dalam kondisi sangat tidak bugar.
2. Lalu lakukan pengecekan formasi apa saja yang bisa digunakan oleh komposisi pemain pada tim tersebut.
3. Lakukan pengecekan di setiap rekomendasi formasi yang sudah didapatkan pada langkah 2 dengan pemain-pemain yang bisa menempati posisi itu dengan *rating* tertinggi dan mempunyai kondisi minimal **kurang bugar**.
4. Jika ada pemain yang memenuhi algoritma *greedy* maka masukkan ke dalam list pada posisi itu, lalu catat jumlah *rating* nya.
5. Jika tidak maka kembali ke langkah 3 tetapi dengan formasi yang lain.
6. Jika semua formasi yang ada di rekomendasi formasi (pada langkah 2) sudah di cek semua, maka pilih formasi yang memiliki jumlah *rating* paling besar dan itulah yang akan dijadikan solusi optimal.

```

list_pemain_tengah_senang([MF, LMF, RMF])
list_tengah_senang = []
for i in range(len(list_pemain)):
    if list_pemain[i][1] == "LMF" or list_pemain[i][1] == "RMF" or list_pemain[i][1] == "MF":
        list_tengah_senang.append(list_pemain[i])

#sort pemain dari yg punya rating terbesar
list_tengah_senang = sorted(list_tengah_senang, key=lambda x: x[2], reverse = True)

```

Gambar 3.1 Potongan kode *python* untuk langkah no 3

B. Pengujian dan Analisis

Pengujian dilakukan dengan beberapa aturan sebagai berikut :

1. Mode permainan yang dimainkan adalah *Exhibition Match*.
2. Pertandingan harus dimainkan oleh CPU atau CPU vs CPU, sehingga kita hanya sebagai pelatih atau pengatur strategi dan formasi saja. Hal ini dilakukan untuk menghindari *human skill's* yang dapat mengganggu keefektifan algoritma.

- CPU musuh boleh dilevel mana saja, tetapi untuk kali ini akan dilakukan dilevel *Super Star* (level tersulit di PES 2017).
- Waktu pertandingan setiap babak adalah 5 menit. Ini hanya untuk testing saja.
- Kedua tim diperbolehkan melakukan pergantian pemain jika memang dibutuhkan.

Metode pengujian :

Pengujian algoritma ini adalah dengan mengambil data pemain dari PES 2017 secara langsung. Data tersebut disimpan di file txt yang nantinya akan dibaca oleh program. Di File tersebut terdapat *nama pemain, posisi pemain, rating*, dan *kondisi* (direpresentasikan sebagai angka 1-5, dimana 1 adalah kondisi sangat tidak bugar dan 5 kondisi sangat bugar).

Analisis :

- Melawan klub sepakbola yang sama

Analisis kali ini dilakukan dengan melawan klub yang sama agar bisa lebih melihat keefektifan algoritma yang dibuat karena pemain dan komposisi *ratingnya* pun sama.



Gambar 3.1.1 Keadaan tim sebelum diubah formasi

```

Aguero,CF,90,5
De Bruyne,AMF,87,4
Nasri,LMF,82,5
Jesus Navas,RMF,80,3
Fernandinho,DMF,77,3
Yaya Toure,DMF,79,3
Kolarov,LB,80,3
E. Mangala,CB,80,4
Otamendi,CB,85,3
Zabaleta,RB,83,3
Joe Hart,GK,84,3
Caballero,GK,75,3
Demichelis,CB,76,2
Kompany,CB,85,3
Sagna,RB,80,3
Clichy,LB,80,3
Fernando,DMF,79,3
F. Delph,CMF,77,5
David Silva,AMF,87,3
Manuel Garcia,AMF,72,3
Sterling,LWF,82,4
Bony,CF,80,5
Iheanacho,CF,76,2

```

Gambar 3.1.2 Translasi data pemain ke txt

Input Squad Success!

Formasi yang bisa dipakai :

```

[4, 2, 3, 1]
[4, 1, 4, 1]
[4, 3, 2, 1]
[4, 2, 2, 2]
[4, 3, 1, 2]
[4, 2, 1, 3]
[4, 1, 2, 3]
[3, 2, 4, 1]
[3, 2, 3, 2]
[3, 3, 2, 2]
[3, 2, 2, 3]

```

Rekomendasi formasi :

3-2-3-2

Susunan pemain untuk formasi : [3, 2, 3, 2]

Penyerang (CF/RWF/LWF) :

- Aguero
- Sterling

Pemain Tengah Serang (AMF/LMF/RMF) :

- De Bruyne
- David Silva
- Nasri

Pemain Tengah Bertahan (CMF/DMF) :

- Yaya Toure
- Fernando

Pemain Belakang (CB/LB/RB) :

- Otamendi
- Kompany
- Zabaleta

Kiper (GK) :

- Joe Hart

Gambar 3.1.3 Hasil penggunaan program



Gambar 3.1.4 Formasi setelah mengikuti hasil program



Gambar 3.1.5 Hasil Pertandingan

Hasil pertandingan berhasil memberikan bukti bahwa perancangan algoritma *greedy* yang digunakan saat melawan klub yang sama dapat menghasilkan kemenangan sehingga algoritma ini dapat kita anggap efektif dalam menentukan formasi.

- b. Melawan klub sepakbola yang berbeda namun dengan *rating* yang hampir sama

Analisis kali ini dilakukan dengan melawan klub sepakbola yang beda namun dengan *rating* yang hampir sama. Hal ini dilakukan untuk menguji lagi seberapa efektif algoritma yang dibuat ketika melawan pemain dengan *rating* yang hampir sama tetapi beda klub. Untuk kasus uji kali ini akan melawan tim London FC dimana mempunyai *rating* yang hampir sama dengan Man Blue.



Gambar 3.2.1 Formasi sebelum memakai program

Aguero,CF,90,3
 De Bruyne,AMF,87,3
 Nasri,LMF,82,3
 Jesus Navas,RMF,80,3
 Fernandinho,DMF,77,5
 Yaya Toure,DMF,79,5
 Kolarov,LB,80,4
 E. Mangala,CB,80,3
 Otamendi,CB,85,4
 Zabaleta,RB,83,3
 Joe Hart,GK,84,2
 Caballero,GK,75,3
 Demichelis,CB,76,4
 Kompany,CB,85,3
 Sagna,RB,80,3
 Clichy,LB,80,3
 Fernando,DMF,79,3
 F. Delph,CMF,77,1
 David Silva,AMF,87,3
 Manuel Garcia,AMF,72,5
 Sterling,LWF,82,3
 Bony,CF,80,3
 Iheanacho,CF,76,4

Gambar 3.2.2 Translasi data pemain ke txt

Input Squad Success!

Formasi yang bisa dipakai :

- [4, 2, 3, 1]
- [4, 3, 2, 1]
- [4, 2, 2, 2]
- [4, 3, 1, 2]
- [4, 2, 1, 3]
- [4, 1, 2, 3]
- [3, 2, 3, 2]
- [3, 3, 2, 2]
- [3, 2, 2, 3]

Rekomendasi formasi :

3-2-3-2

Susunan pemain untuk formasi : [3, 2, 3, 2]

Penyerang (CF/RWF/LWF) :

- 1. Aguero
- 2. Sterling

Pemain Tengah Serang (AMF/LMF/RMF) :

- 1. De Bruyne
- 2. David Silva
- 3. Nasri

Pemain Tengah Bertahan (CMF/DMF) :

- 1. Yaya Toure
- 2. Fernando

Pemain Belakang (CB/LB/RB) :

- 1. Otamendi
- 2. Kompany
- 3. Zabaleta

Kiper (GK) :

- 1. Caballero

Gambar 3.2.3 Penggunaan program



Gambar 3.2.4 Formasi setelah mengikuti Program



Gambar 3.2.5 Hasil Pertandingan

Walaupun membutuhkan waktu tambahan (*extra time* + *penalty*) tetapi tetap memberikan bukti bahwa perancangan algoritma *greedy* yang digunakan saat melawan klub yang berbeda dan dengan *rating* yang lebih kecil dapat menghasilkan kemenangan sehingga algoritma ini dapat kita anggap efektif dalam menentukan formasi.

- c. Melawan klub sepakbola yang berbeda namun dengan *rating* yang lebih besar

Analisis kali ini akan diuji dengan melawan klub sepakbola yang berbeda dan memiliki *rating* lebih besar dari tim kita yaitu Man Blue dimana tim tersebut adalah tim Real Madrid.



Gambar 3.3.1 Formasi sebelum menggunakan Program

Aguero,CF,90,5
 De Bruyne,AMF,87,4
 Nasri,LMF,82,3
 Jesus Navas,RMF,80,3
 Fernandinho,DMF,77,3
 Yaya Toure,DMF,79,3
 Kolarov,LB,80,5
 E. Mangala,CB,80,3
 Otamendi,CB,85,3
 Zabaleta,RB,83,3
 Joe Hart,GK,84,3
 Caballero,GK,75,2
 Demichelis,CB,76,4
 Kompany,CB,85,3
 Sagna,RB,80,3
 Clichy,LB,80,4
 Fernando,DMF,79,2
 F. Delph,CMF,77,3
 David Silva,AMF,87,3
 Manuel Garcia,AMF,72,2
 Sterling,LWF,82,4
 Bony,CF,80,2
 Iheanacho,CF,76,2

Gambar 3.3.2 Hasil translasi data pemain ke file txt

Input Squad Success!

Formasi yang bisa dipakai :

- [4, 2, 3, 1]
- [4, 1, 4, 1]
- [4, 3, 2, 1]
- [4, 2, 2, 2]
- [4, 3, 1, 2]
- [4, 2, 1, 3]
- [4, 1, 2, 3]
- [3, 2, 4, 1]
- [3, 2, 3, 2]
- [3, 3, 2, 2]
- [3, 2, 2, 3]

Rekomendasi formasi :

3-2-3-2

Susunan pemain untuk formasi : [3, 2, 3, 2]

Penyerang (CF/RWF/LWF) :

- 1. Aguero

- 2. Sterling

Pemain Tengah Serang (AMF/LMF/RMF) :

- 1. De Bruyne

- 2. David Silva

- 3. Nasri

Pemain Tengah Bertahan (CMF/DMF) :

- 1. Yaya Toure

- 2. Fernando

Pemain Belakang (CB/LB/RB) :

- 1. Otamendi

- 2. Kompany

- 3. Zabaleta

Kiper (GK) :

- 1. Joe Hart

Gambar 3.3.3 Penggunaan program



Gambar 3.3.4 Formasi setelah menggunakan Program



Gambar 3.3.5 Hasil Pertandingan

Hasil pertandingan dapat memberikan bukti bahwa perancangan algoritma *greedy* yang digunakan saat melawan klub yang berbeda dan dengan *rating* yang lebih tinggi dapat menghasilkan kemenangan sehingga algoritma ini dapat kita anggap efektif dalam menentukan formasi

IV. KESIMPULAN DAN SARAN

Dari hasil pengujian yang dilakukan, dapat memberikan kesimpulan bahwa algoritma *greedy* pada kasus ini sudah bisa dibilang optimal. Namun, masih ada beberapa bukti yang membuat algoritma *greedy* ini belum optimal, dari percobaan-percobaan diluar makalah ini formasi yang dihasilkan bisa saja memungkinkan untuk kalah tetapi tidak mengalami kekalahan telak, hanya sekitar 1-0 saja.

Saran untuk pembaca adalah temukan Kembali algoritma *greedy* yang memang memungkinkan menghasilkan rate menang jauh lebih tinggi, akan tetapi perlu diingat bahwa algoritma *greedy* **tidak selalu menemukan solusi yang optimal.**

V. UCAPAN TERIMAKASIH

Puji syukur penulis panjatkan kepada tuhan YME, karena berkat rahmat dan karunia-nya makalah ini dapat diselesaikan tepat waktu. Rasa terimakasih juga penulis ucapkan kepada Bapak Ir. Rila Mandala, M.Eng.,Ph.D. atas bimbingannya selaku dosen mata kuliah strategi algoritma IF2210 kelas K-01 pada tahun ajaran 2020/2021. Penulis juga mengucapkan terimakasih kepada orang tua penulis dan rekan-rekan sekalian atas semua bentuk doa, bantuan, masukan, sehingga makalah ini dapat selesai.

VIDEO LINK AT YOUTUBE AND PROGRAM LINK AT GITHUB

Berikut link untuk video dan program terkait makalah ini :

1. Github

<https://github.com/ridhodaffasyah/BestFormation-PES2017>

2. Youtube

<https://youtu.be/vqnIsD-WkSs>

REFERENCES

- [1] <https://www.konami.com/games/eu/en/topics/13924/> , diakses pada 11 Mei 2021 pukul 03.14
- [2] Munir, Rinaldi. Algoritma Greedy. [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag1.pdf). 2021.
- [3] Fikri Hizbullah, Muhammad. *Optimalisasi Pemilihan Formasi dengan Algoritma Greedy pada Permainan PES*. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2018-2019/Makalah/Makalah-Stima-2019-102.pdf>. 2019.
- [4] Alif Goenawan, Muhammad. *Review PES 2017 Bermuculan, Hasilnya?*. <https://inet.detik.com/games-news/d-3301340/review-pes-2017-bermuculan-hasilnya>. 2016. Diakses pada 11 Mei 2021 pukul 03.14
- [5] <https://www.konami.com/wepes/2018/manual/ps4/en/plan.html>. Diakses pada 11 Mei 2021 pukul 03.14.

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.

Bekasi, 11 Mei 2021



Ttd
Ridho Daffasyah / 13519038