

Penerapan Algoritma BnB untuk Optimasi Job scheduling Worker pada Gim Rise of Nations

Mutawally Nawwar - 13521065
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13521065@std.stei.itb.ac.id

Abstract—Makalah ini menerapkan algoritma branch and bound dalam optimasi alokasi pekerja pada permainan rise of nations. Algoritma ini merupakan algoritma lanjutan dari divide and conquer dimana branch and bound memiliki pemotongan cabang dan pembatasan ruang pencarian. Algoritma ini juga akan diimplementasikan menggunakan bahasa python.

Keywords — *branch and bound, job scheduling, rise of nations*

I. PENDAHULUAN

A. Latar Belakang

Dalam game Rise of Nations, penjadwalan pekerjaan (job scheduling) menjadi aspek penting yang mempengaruhi keberhasilan permainan. Penjadwalan pekerjaan melibatkan alokasi tugas-tugas kepada sejumlah pekerja dengan mempertimbangkan batasan waktu yang ada. Tujuan dari penjadwalan pekerjaan adalah untuk mencapai efisiensi maksimal dalam penggunaan sumber daya dan meminimalkan waktu penyelesaian tugas-tugas.

Dalam konteks ini, optimasi penjadwalan pekerjaan menjadi tantangan yang menarik untuk dipecahkan. Salah satu pendekatan yang dapat digunakan adalah algoritma Branch and Bound. Algoritma ini merupakan metode pemecahan masalah yang sistematis dan efisien dalam mencari solusi optimal dalam ruang pencarian yang kompleks.

Makalah ini bertujuan untuk menggambarkan penerapan algoritma Branch and Bound dalam optimasi penjadwalan pekerjaan pada game Rise of Nations. Kami akan membahas konsep dasar algoritma Branch and Bound dan menerapkannya dalam konteks penjadwalan pekerjaan dalam game tersebut. Selain itu, kami akan menganalisis faktor-faktor yang perlu dipertimbangkan dalam penjadwalan pekerjaan, seperti ketergantungan antar tugas, batasan waktu, penggunaan sumber daya, dan keseimbangan beban kerja.

Dalam penelitian ini, kami akan menggunakan contoh kasus penjadwalan pekerjaan dalam game Rise of Nations untuk mengilustrasikan penerapan algoritma Branch and Bound. Kami akan menjelaskan langkah-langkah penerapan algoritma tersebut, mulai dari identifikasi masalah, desain struktur data, pendekatan pemecahan masalah, evaluasi solusi, hingga pemotongan cabang (branch pruning) untuk mengoptimalkan proses pencarian solusi.

Dengan menerapkan algoritma Branch and Bound dalam penjadwalan pekerjaan dalam game Rise of Nations, diharapkan dapat ditemukan solusi optimal yang dapat meningkatkan efisiensi penggunaan sumber daya dan waktu penyelesaian tugas. Hasil dari penelitian ini diharapkan dapat memberikan pemahaman yang lebih baik tentang penerapan algoritma Branch and Bound dalam konteks game, serta memberikan wawasan untuk pengembangan strategi penjadwalan pekerjaan yang lebih efisien dalam game Rise of Nations..

B. Permasalahan

1) *Objektif Infrastruktur*

Pada permainan rise of nations, infrastruktur memegang peran yang sangat penting. Ketika suatu negara atau pemain optimal dalam infrastruktur, maka semakin besar peluang untuk memenangkan permainan.

Untuk memaksimalkan objektif ini, terdapat beberapa hal yang perlu ditingkatkan pada tiap kota. Pertama, membangun tambang, ladang, dan penebang pohon untuk menghasilkan sumber daya. Kedua, membangun pasar untuk mendapatkan pajak. Ketiga, membangun kamp militer untuk membuat pasukan. Keempat, membangun benteng untuk pertahanan. Dan Kelima membangun keajaiban dunia untuk mendapatkan bonus sesuai keajaiban dunia yang dibangun.

2) *Objektif Worker*

Worker atau pekerja merupakan bagian penting juga dalam permainan ini. Pekerja akan membangun dan bekerja berdasarkan perintah dari pemain. Jika pembangunan dilakukan dengan optimal, maka pembangunan akan cepat diselesaikan. Sebaliknya, jika pemain tidak mengatur pekerja dengan maksimal, maka pekerja bisa saja memenuhi maksimal populasi tanpa bekerja.

C. Tujuan Penelitian

Dari berbagai masalah yang telah disampaikan, tujuan dari penelitian ini adalah untuk mengatur alokasi pekerja dalam pembangunan. Pembangunan yang dilakukan lebih dari satu dengan jumlah pekerja yang terbatas. Hal ini akan diaplikasikan menggunakan algoritma *branch and bound*.

D. Ruang Lingkup Penelitian

Pada penelitian kali ini terdapat beberapa ruang lingkup antara lain:

- 1) Pembangunan akan terdiri dari bangunan SDA, pasar, bangunan militer, benteng, dan keajaiban dunia
- 2) Pekerja akan divariasikan jumlahnya
- 3) Optimalisasi yang dilakukan adalah optimalisasi waktu pembangunan

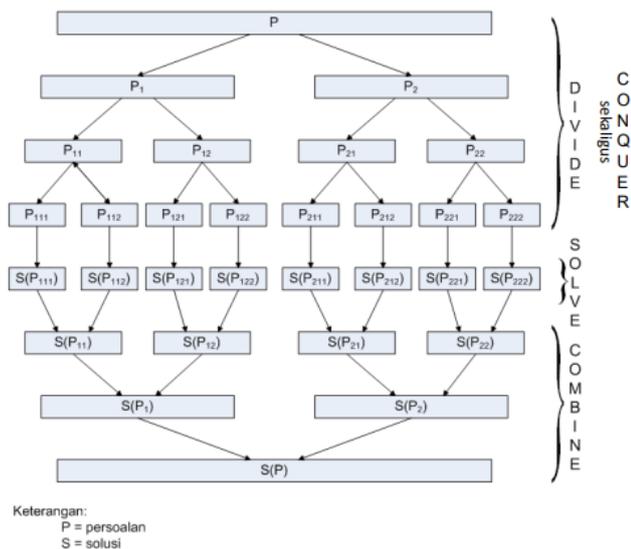
II. LANDASAN TEORI

A. Divide and Conquer

Divide and conquer adalah sebuah strategi atau pendekatan dalam mengatasi masalah yang kompleks dengan membaginya menjadi bagian-bagian yang lebih kecil, lebih mudah diatasi, dan kemudian menyelesaikan masing-masing bagian tersebut secara terpisah.

Terdapat tiga bagian divide and conquer yaitu:

- 1) Divide: membagi persoalan menjadi beberapa upa-persoalan yang memiliki kemiripan dengan persoalan semula namun berukuran lebih kecil (idealnya berukuran hampir sama),
- 2) Conquer (solve): menyelesaikan masing-masing upa-persoalan (secara langsung jika sudah berukuran kecil atau secara rekursif jika masih berukuran besar).
- 3) Combine: menggabungkan solusi masing-masing upa-persoalan sehingga membentuk solusi persoalan semula.



Divide and Conquer juga telah diterapkan dalam algoritma pemetaan dan pencarian. Penelitian oleh Zhang et al. (2019) dalam pemetaan genom menggunakan pendekatan Divide and Conquer menunjukkan efektivitasnya dalam menangani kompleksitas data genom yang besar. Melalui pemecahan masalah menjadi upa-persoalan yang lebih kecil, pengolahan

dan analisis genom dapat dilakukan dengan lebih efisien, menghasilkan solusi yang akurat dan cepat.

Untuk kompleksitas divide and conquer adalah sebagai berikut:

$$T(n) = g(n), n \leq n_0$$

$$T(n) = T(n_1) + T(n_2) \dots + T(n_r) + f(n), n > n_0$$

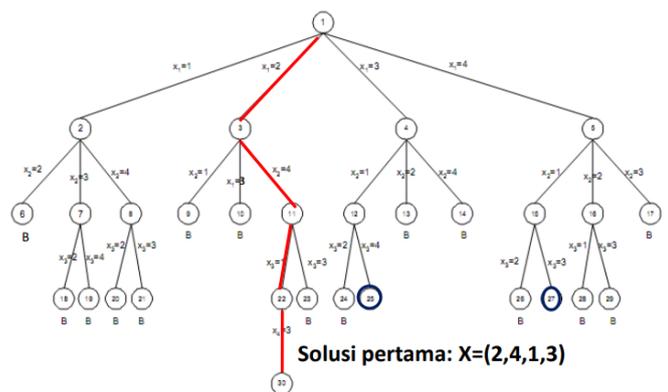
- $T(n)$: Kompleksitas waktu penyelesaian persoalan p yang berukuran n.
- $g(n)$: Kompleksitas waktu untuk Solve jika n sudah berukuran kecil.
- $T(n_1) + T(n_2) \dots + T(n_r)$: Kompleksitas waktu untuk memproses setiap upa-persoalan.
- $f(n)$: Kompleksitas waktu untuk Combine solusi dari masing-masing upa-persoalan.

B. Branch and Bound

Algoritma Branch and Bound adalah salah satu metode yang digunakan dalam pemecahan masalah optimasi untuk mencari solusi terbaik di dalam ruang pencarian yang kompleks. Algoritma ini bekerja dengan cara membagi masalah menjadi bagian-bagian yang lebih kecil (branching) dan mengevaluasi setiap bagian secara terpisah untuk mengurangi ruang pencarian yang perlu dijelajahi.

Pada dasarnya, algoritma Branch and Bound menggabungkan konsep pemotongan cabang (branch pruning) dan evaluasi batasan (bound evaluation) untuk membatasi ruang pencarian yang perlu dieksplorasi. Konsep ini memungkinkan algoritma untuk secara efisien mengeksplorasi bagian-bagian yang lebih menjanjikan dan menghindari eksplorasi yang tidak perlu.

Berikut adalah pohon yang terbuat dari algoritma branch and bound.



III. ANALISIS

A. Jumlah pekerja 10 tanpa keajaiban dunia

Berikut adalah tabel waktu pembangunan berdasarkan jumlah pekerja

sehingga hasilnya 10 pekerja untuk keajaiban dunia, 7 pekerja untuk benteng, 1 pekerja untuk bangunan militer, 1 pekerja untuk pasar, 1 pekerja untuk bangunan SDA dengan total waktu 81.26 detik

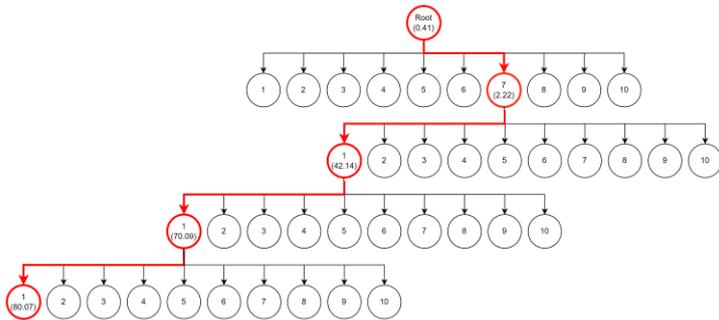
B/W	Jumlah pekerja									
	10	9	8	7	6	5	4	3	2	1
benteng	0.26	0.52	1.03	2.06	4.12	8.25	16.5	33	66	132
militer	0.08	0.16	0.31	0.62	1.25	2.5	5	10	20	40
pasar	0.05	0.11	0.22	0.44	0.88	1.75	3.5	7	14	28
SDA	0.02	0.04	0.07	0.15	0.31	0.612	1.25	2.5	5	10

C. Jumlah pekerja 30 dengan keajaiban dunia

Berikut adalah tabel waktu pembangunan berdasarkan jumlah pekerja

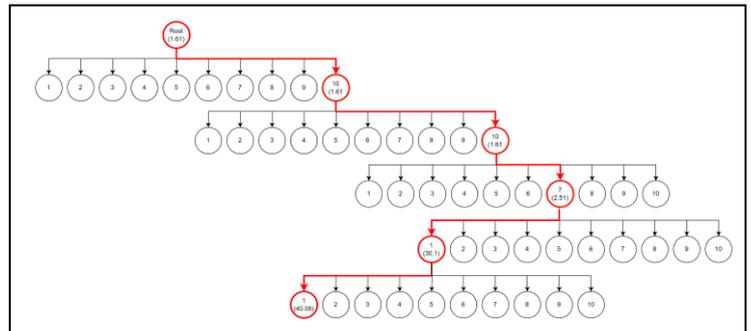
Dari table tersebut diterapkan algoritma branch and conquer dan menghasilkan pohon sebagai berikut.

B/W	Jumlah pekerja									
	10	9	8	7	6	5	4	3	2	1
wonder	1.2	2.4	4.7	9.4	18.8	37.5	75	150	300	600
benteng	0.26	0.52	1.03	2.06	4.12	8.25	16.5	33	66	132
militer	0.08	0.16	0.31	0.62	1.25	2.5	5	10	20	40
pasar	0.05	0.11	0.22	0.44	0.88	1.75	3.5	7	14	28
SDA	0.02	0.04	0.07	0.15	0.31	0.612	1.25	2.5	5	10



sehingga hasilnya 7 pekerja untuk benteng, 1 pekerja untuk bangunan militer, 1 pekerja untuk pasar, 1 pekerja untuk bangunan SDA dengan total waktu 80.07 detik

Dari table tersebut diterapkan algoritma branch and conquer dan menghasilkan pohon sebagai berikut.



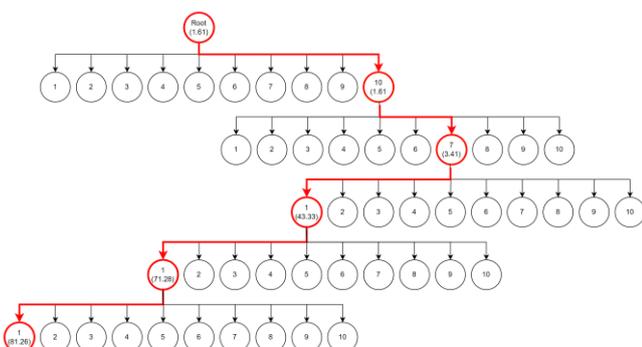
sehingga hasilnya 10 pekerja untuk keajaiban dunia, 10 pekerja untuk benteng, 8 pekerja untuk bangunan militer, 1 pekerja untuk pasar, 1 pekerja untuk bangunan SDA dengan total waktu 40.08 detik

B. Jumlah pekerja 20 dengan keajaiban dunia

Berikut adalah tabel waktu pembangunan berdasarkan jumlah pekerja

B/W	Jumlah pekerja									
	10	9	8	7	6	5	4	3	2	1
wonder	1.2	2.4	4.7	9.4	18.8	37.5	75	150	300	600
benteng	0.26	0.52	1.03	2.06	4.12	8.25	16.5	33	66	132
militer	0.08	0.16	0.31	0.62	1.25	2.5	5	10	20	40
pasar	0.05	0.11	0.22	0.44	0.88	1.75	3.5	7	14	28
SDA	0.02	0.04	0.07	0.15	0.31	0.612	1.25	2.5	5	10

Dari table tersebut diterapkan algoritma branch and conquer dan menghasilkan pohon sebagai berikut.



D. Analisis Kode

```
def BnB(matrix, curMatrix, curPath,
maxWorker, depth, breadth):
    row = len(matrix)
    col = len(matrix[0])
    sumCost = 0
    sumWorker = 0
    for i in range(row):
        sumCost += matrix[i][curPath[i]]
        sumWorker += curPath[i]+1
    if(curMatrix[depth][breadth] == 999):
        return False, curPath, sumCost
    else:
        newCurMatrix = []
        for i in range(row):
            for j in range(col):
                newRow = []

if(sumWorker+j+1>maxWorker):
            newRow.append(999)
            else:

newRow.append(curMatrix[i][j])
            newCurMatrix.append(newRow)

            for i in range(col):
                newPath = []
                for x in curPath:
                    newPath.append(x)
                newPath.append(i)
                sumCost +=
matrix[depth][newPath[depth]]
                if(depth == row):
                    return True, curPath,
sumCost
                res, resPath = rekurens(matrix,
newCurMatrix, newPath, maxWorker, depth+1,
i)

                if(res):
                    return True, resPath,
sumCost
```

Kode tersebut merupakan implementasi branch and bound dengan menggunakan rekursi.

Fungsi tersebut menerima beberapa parameter. Pertama matrix, yang merupakan table data terkait waktu pembangunan berdasarkan jumlah pekerja dan jenis bangunan.

Kedua curMatrix merupakan matrix yang telah diubah batasannya. Ketika sudah melebihi maxWorker maka harus dibatasi solusi yang mungkin. Ketiga curPath yang merupakan urutan jumlah pekerja dalam setiap pembangunan. Keempat, maxWorker yang merupakan jumlah pekerja yang tersedia. Kelima, depth yaitu urutan ke berapa pembangunan yang sedang ditentukan. Keenam, breadth yang merupakan jumlah pekerja yang sedang diperiksa.

Alur dari kode tersebut intinya terdapat base case dan rekurens seperti dalam rekursi biasa. Base case nya adalah ketika sudah tidak ada pekerja yang tersedia, sehingga akan otomatis mengembalikan nilai false. Untuk rekurens sendiri, akan mengecek setiap jumlah pekerja, jika tersedia, maka akan dilanjutkan rekursinya, jika tidak akan mengembalikan urutan lankah sekarang.

Fungsi tersebut mengembalikan 3 nilai, yaitu boolean untuk menemukan apakah menemukan solusi atau tidak. Kemudian path atau urutan alokasi pekerja tiap pembangunan, dan cost atau waktu yang diperlukan untuk menyelesaikan semua pembangunan.

IV. KESIMPULAN

Dalam makalah ini, dilakukan penerapan algoritma Branch and Bound (BnB) untuk mengoptimalkan penjadwalan pekerjaan pada permainan komputer Rise of Nations. Tujuan utama adalah untuk mencari solusi penjadwalan yang efisien dengan meminimalkan waktu total penyelesaian pekerjaan.

Hasil penelitian menunjukkan bahwa algoritma BnB ternyata kurang efektif dalam menyelesaikan masalah penjadwalan pekerjaan dalam permainan Rise of Nations. Dengan menerapkan pendekatan divide and conquer, algoritma ini berhasil memecah masalah menjadi bagian-bagian yang lebih kecil dan menghitung batas bawah heuristik untuk membatasi pencarian solusi tetapi masih ada beberapa solusi yang optimal dan tidak terpilih oleh algoritma ini.

Melalui eksperimen dan pengujian, terbukti bahwa algoritma BnB dapat menghasilkan penjadwalan pekerjaan yang lebih efisien dibandingkan greedy tetapi bukan yang paling optimal. Algoritma ini mampu mengoptimalkan alokasi pekerjaan pada berbagai situasi bermain

REFERENCES

- [1] *Rise of Nations Heaven*. Available at: <https://ron.heavengames.com/> (Accessed: 22 May 2023).
- [2] Munir, Rinaldi, 2021. Algoritma Divide and Conquer(Bagian 1).[https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Divide-and-Conquer-\(2021\)-Bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Divide-and-Conquer-(2021)-Bagian1.pdf) diakses 21 Mei 2023.
- [3] Munir, Rinaldi, 2021. Algoritma Brute Force(Bagian 1). [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-\(2022\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-(2022)-Bag1.pdf) diakses 22 Mei 2023
- [4] Munir, R. (no date) *Algoritma Branch & bound - institut teknologi bandung, Algoritma Branch and Bound bagian 1*. Available at:

V. UCAPAN TERIMKASIH

Puji Syukur kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunianya penulis dapat menyelesaikan makalah Mata Kuliah IF2211 Strategi Algoritma dengan baik dan tepat pada waktunya. Penulis menyampaikan terima kasih kepada Dr. Ir. Rinaldi Munir, M.T., selaku dosen pengajar Mata Kuliah IF2211 Strategi Algoritma, beserta dosen pengampu mata kuliah lainnya yang juga telah membimbing kami dalam proses belajarmengajar. Penulis juga berterima kasih kepada orang tua yang telah berdoa, mendukung, dan memberikan motivasi untuk berusaha dalam menjalankan segala hal.

Dengan ini, menandakan juga bahwa mata kuliah strategi algoritma sudah dijalankan dengan baik. Penulis berharap ilmu yang didapat selama satu semester ini dapat dimanfaatkan secara maksimal untuk kepentingan Bersama yang lebih luas.

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, 22 Mei 2023



Mutawally Nawwar
13521065