Penggunaan Strategi Algoritma Branch and Bound Dalam Menyelesaikan Quadratic Assignment Problem

Muhamad Ilyas Mustafa / 13516123
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
emailpribadiilyas@gmail.com

***Abstrak*— *Quadratic Assignment Problem* adalah sebuah permasalah kombinatorial yang dikemukakan oleh Koopmans dan Beckmann pada tahun 1957. Permasalahan ini bertujuan untuk meminimalisir cost yang dikeluarkan ketika membangun berbagai fasilitas pada tempat yang telah ditentukan. Oleh karena itu, makalah ini bertujuan untuk menunjukkan salah satu cara alternatif dalam menyelesain permasalahan ini yaitu dengan menggunakan strategi algoritma *Branch and Bound* baik melalui teori maupun praktik.**

***Kata Kunci*—Algoritma, Brach and Bound, simpul, cost.**

I. PENDAHULUAN

Terdapat berbagai macam permasalahan di muka bumi ini yang kita hadapi tiap hari. Dan tentunya dalam menyelesaikan masalah tersebut, terdapat berbagai macam cara untuk menyelesaikan problematika tersebut. Telah banyak strategi pemecahan masalah yang dikemukakan di dunia ini oleh para ahli. Salah satu strategi pemecahan masalah tersebut adalah algoritma *Branch and Bound*.

Branch and Bound (B&B) merupakan suatu strategi algoritma memecah masalah - masalah tersebut menjadi tahpa-per tahap. Strategi algoritma ini terdiri dari suatu ruang status, yang melambangkan kondisi dari suatu permasalah, beserta dengan cost yang dihasilkan masing – masing langkah pada algoritma tersebut. Algoritma ini baik untuk menyelesaikan permasalahan diskrit maupun optimasi kombinatorial.

Sebagai contoh, ketika Andi akan pergi ke sekolah, Andi dihadapkan oleh beberapa pilihan rute keberangkatan yang memiliki jarak dan waktu tempuh yang berbeda. Rute A berjarak dari rumah Andi sejauh 10 km dengan waktu tempuh menggunakan kendaraan sebesar 30 menit, namun bila Andi memutuskan untuk berjalan kaki, waktu tempuh adalah 1 jam. Sedangkan rute B, hanya berjarak 5 km dari rumah Andi, namun karna pada saat itu sedang macet waktu tempuh pada rute tersebut sebesar 1 jam demgam kendaraan bermotor atau 30 menit dengan berjalan kaki. Kedua rute B merupakan sama – sama pilihan yang valid dan merupakan jawaban terbaik tergantung *cost* yang ingin dicari oleh Andi. Bila Andi pada saat itu tidak memiliki uang, rute B merupakan rute yang akan Andi ambil. Sedangkan apabila pada saat itu Andi sedang sakit kaki dan tidak perduli dengan ongkos yang dikeluarkan untuk sampai ke sekolah, Andi akan menyewa ojek dengan melalui rute A.

Masih banyak berbagai permasalahan yang dapat diselesaikan dengan beragam strategi penyelesaian permasalahan yang beragam pula. Salah satu permasalahan itu adalah *Quadratic Assignment Problem* atau dapat disingkat sebagai QAP.

Quadratic Assignment Problem adalah suatu permasalahan optimasi kombinasi yang bertujuan unyuk meminimalisir biaya yang dikeluarkan dalam membangun n buah fasilitas pada n buah lokasi. Setiap lokasi hanya dapat diisi oleh satu fasilitas. Pemecahan masalah ini dapat membantu menemukan solusi yang baik dalam menyusun suatu fasilitas sering kita lihat di kehidupan sehari - hari. Contohnya adalah penyususan ruangan rumah sakit. Penyusunan ruangan rumah sakit tidak boleh sembarangan. Dengan menggunakan model QAP, penyusunan ruangan rumah sakit dapat meiliki cost yang lebih rendah, namun memiliki produktivitas yang tinggi. Oleh karena itu pada makalah ini akan dibahas salah satu cara pemecahan *Quadratic Assignment Problem* dengan menggunakan pendekatan algoritma *Branch and Bound*.

II. DASAR TEORI

A. Branch and Bound

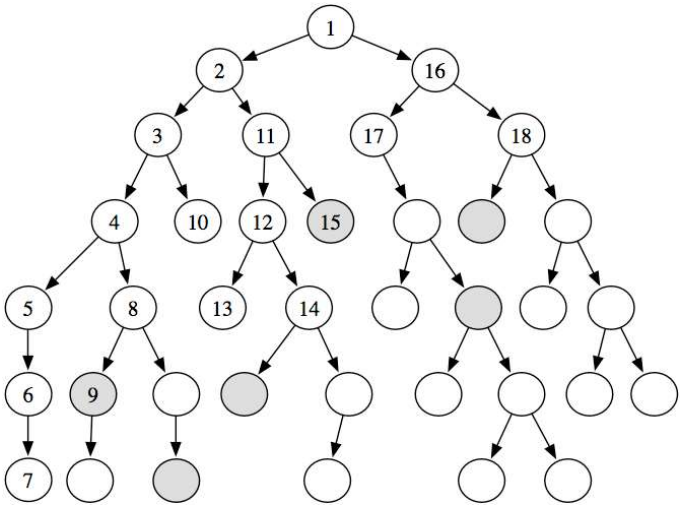
Branch and Bound (B&B) merupakan sebuah metoda sistematis untuk menyelesaikan permasalahn optimasi. B&B menggunakan suatu ruang solusi, yang dalam bentuk ruang status, untuk melakukan pencarian secara sistematis.

Strategi algoritma *Branch and Bound* memiliki kemiripan dengan strategi algoritma runut balik yaitu pada penggunaan ruang status. Namun pada strategi algoritma runut balik, proses pencarian pencarian pada ruang status menggunakan skema DFS (*Depth First Search*). Sedangkan pada strategi algoritma *Branch and Bound* menggunakan skema BFS (*Breadth-First Search*).

Dalam suatu ruang status, setiap simpul akan dikunjungi untuk mendapatkan simpul solusi. Oleh karena itu, untuk

mempercepat proses pencarian pada strategi algoritma ini, setiap simpul akan diberikan suatu cost c(i), dimana cost ini akan menentukan simpul mana yang akan di-*expand* sehingga dapat menghindari simpul – simpul yang tidak relevan dengan simpul tujuan (*goal node*). Hal ini berbeda dengan skema BFS murni yang meng-*expand* simpul berdasarkan urutan kebangkitan simpul.

Pada algoritma BFS, terdapat sebuah antrian / *queue* untuk menyimpan semua simpul yang baru dibangkitkan. Simpul yang baru dibangkitkan akan disimpan di akhir antrian / *queue*. Simpul selanjutnya yang akan di-*expand* adalah simpul yang berada pada kepala antrian. Hal ini sama seperti skema FIFO (First In First Out), dimana simpul hidup (*live node*, adalah simpul yang sedang mengekspansi) akan dimasukan ke dalam antrian, sedangkan simpul selanjutnya yang akan menjadi simpul-E (simpul yang akan diekspansi) adalah simpul yang berada pada kepala antrian.



Gambar 1. Contoh Graph Berarah Branch and Bound Sumber : http://artint.info/figures/ch03/sgraph_bb.png

Begitu juga dengan strategi algoritma B&B. Namun pada algoritma ini, tidak menganut skema FIFO, melainkan berdasarkan cost yang ada. Sebagai contoh, perhatikan gambar 1 :

1. Pada BFS murni, urutan pembangkitan 7 simpul pertama adalah 1 -> 2 -> 16 ->3 ->11 -> 17 -> 18.
2. Sedangkan pada B&B kita perlu memperhatikan cost yang berlaku. Apabila simpul 16 memiliki cost yang lebih "diinginkan" daripada simpul 3 maka simpul 16 yang akan diekspansi terlebih dahulu. Begitu seterusnya hingga mendapatkan simpul yang diinginkan (*goal node*). Diinginkan disini berarti memiliki cost yang lebih sesuai dengan aturan masalah yang berlaku. Contohnya, dalam masalah pencarian kombinasi dengan nilai minimum, maka simpul yang lebih sesuai adalah simpul yang memiliki cost paling kecil dibandingkan dengan yang lain.

Secara umum, fungsi heuristik yang digunakan dalam menghitung taksiran nilai / cost suatu simpul dinyatakan sebagai berikut :

c(i) = f(i) + g(i)

dimana :

c(i) adalah ongkos / cost simpul i

f(i) adalah ongkos / cost untuk mencapai simpul i dari simpul awal / akar

g(i) adalah ongkos / cost untuk mencapai simpul tujuan dari simpul i

Nilai c(i) akan digunakan untuk menentukan urutan pencarian pada pohon status yang dibentuk oleh algoritma *Branch and Bound*. Berikut merupakan algoritma *Branch and Bound* secara garis besar :

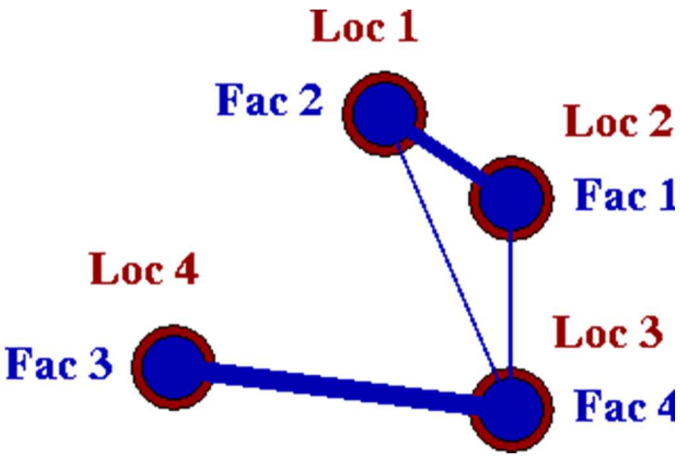
1. Masukkan simpul akar ke antrian Q (antrian telah dikunjungi). Periksa apakah simpul tersebut merupakan simpul solusi. Jika simpul tersebut simpul solusi pencarian berhenti.
2. Jika antrian Q kosong, maka tidak ada solusi yang tepat dan pencarian berhenti.
3. Jika antrian Q tidak kosong, urutkan antrian sesuai dengan nilai c(i). Apabila yang dicari adalah nilai maksimum maka pengurutan dilakukan dari nilai cost terbesar hinnga nilai cost terkecil. Apabila yang dicari adalah nilai minimum maka pengurutan dilakukan dari nilai c(i) terkecil hingga nilai c(i) terbesar.
4. Periksa simpul yang berada pada kepala / *head* yang ada pada antrian Q. Jika merupakan simpul solusi maka pencarian berhenti. Bila tidak, bangkitkan simpul anak dari simpul tersebut. Apabila tidak terdapat simpul anak, kembali ke langkah 2.
5. Hitung masing – masing cost dari simpul anak tersebut, masukkan simpul anak tersebut ke antrian.
6. Kembali ke langkah 2.

Branch and Bound merupakan strategi algoritma yang bagus untuk mencari suatu solusi pada permasalahan diskrit maupun permasalahan optimasi kombinasi. Pada kasus terburuk, semua simpul yang berada pada ruang status harus diperhitungkan untuk medapatkan simpul solusi. Namun pada kasus terbaiknya, kita hanya perlu memeriksa satu bagian dari pohon ruang status untuk mendapatkan simpul solusi.

B. Quadratic Assignment Problem

Permasalahan QAP pertama kali diperkenalkan oleh Koopmans dan Beckman pada tahun 1957. QAP merupakan salah satu dari permasalahan kombinatorial tersulit. Tujuan dari *Quadratic Assignment Problem* (QAP) adalah untuk meminimasi cost yang diperlukan untuk membangun suatu n buah fasilitas pada n buah lokasi dimana satu lokasi hanya bisa ditempati oleh suatu fasilitas.

Cost dari penempatan suatu fasilitas merupakan hasil perkalian dari jarak antara fasilitas dengan *flow* (alur produk) antar fasilitas.



Gambar 2. Contoh Graph Lokasi dan Fasilitas Sumber : https://neos-guide.org/sites/default/files/qap.png

Sebagai contoh terdapat 4 buah lokasi sebagaimana terlihat pada gambar 2. Salah satu kombinasi penempatan yang mungkin adalah fasilitas 2 di lokasi 1, fasilitas 1 di lokasi 2, fasilitas 3 di lokasi 4, dan fasilitas 4 di lokasi 3.

Untuk menghitung cost dari penempatan fasilitas tersebut diperlukan data jarak dan flow antar fasilitas yang dapat dilihat pada tabel berikut.

Facility i	Facility j	Flow (i, j)
1	2	3
1	4	2
2	4	1
3	4	4

Tabel 1. Tabel Flow Antar Fasilitas Sumber : https://neos-guide.org/content/quadratic-assignment-problem#problem

Location i	Location j	Distance (i, j)
1	3	53
2	1	22
2	3	40
3	4	55

Tabel 2. Tabel Jarak antar Lokasi Sumber : https://neos-guide.org/content/quadratic-assignment-problem#problem

Berdasarkan data diatas, cost dari penempatan fasilitas dapat dihitung sebagai f(1,2) * d(2,1) + f(1,4) * d(2,3) + f(2,4) * d(1,3) + f(3,4) * d(3,4) = 419. Perlu diperhatikan bahwa kombinasi ini bukan merupakan kombinasi yang paling minimum.

Beberapa contoh dari aplikasi QAP adalah :

- Penentuan tata letak rumah sakit
- Penempatan pintu bandara, sehingga meminimalisir pergerakan barang dan penumpang.
- Penempatan komponen elektrik pada circuit
- Dan lain sebagainya.

III. PEMBAHASAN

A. Teori

Seperti yang telah dibahas sebelumnya, dalam strategi algoritma *Branch and Bound*, diperlukan suatu fungsi matematis untuk memberikan cost pada satu simpul. Berdasarkan Koopmans – Beckmann, formula matematis untuk QAP adalah sebagai berikut.

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} f(i,j) * d(l(i), l(j))$$

Keterangan :

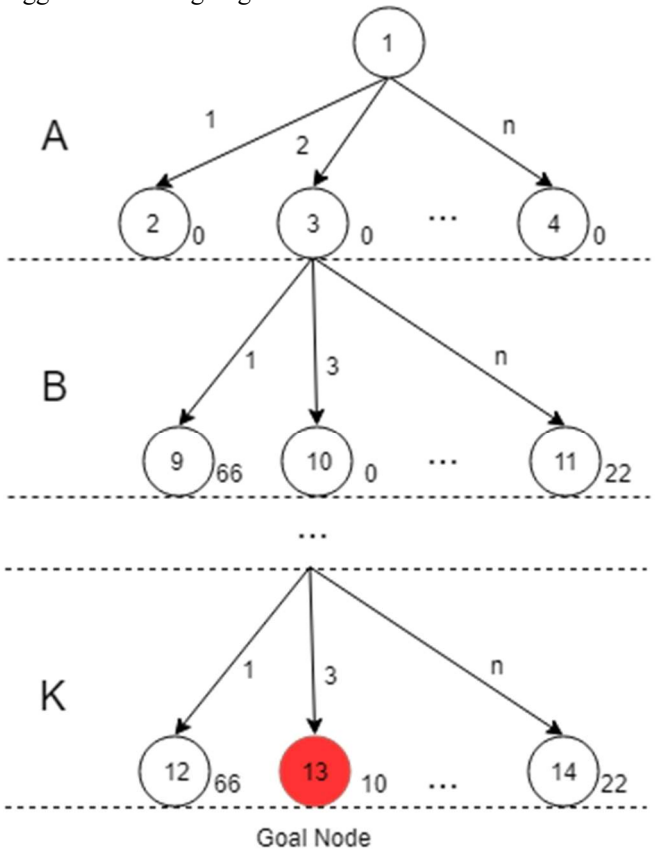
- N adalah jumlah lokasi yang tersedia = fasilitas yang akan dibangun
- F = *f(i,j)* adalah sebuah fungsi yang menghasilkan flow antar fasilitas i dan fasilitas j.
- D = *d(l(i),l(j))* adalah sebuah fungsi yang menghasilkan jarak dari lokasi fasilitas i (l(i)) dan lokasi fasilitas j (l(j)).

Sehingga

$$c(k) = \sum_{i=1}^{n} \sum_{j=1}^{n} f(i,j) * d(l(i), l(j))$$

Dimana n adalah jumlah fasilitas yang sudah ditempatkan sampai dengan simpul k.

Untuk menyelesaikan permasalahan QAP, diperlukan suatu ruang status dalam bentuk pohon yang sesuai sehingga penyelesaian tersebut dapat diselesesaikan dengan menggunakan strategi algoritma Branch and Bound.



Gambar 3. Pohon Status Dari QAP

Gambar 3 merupakan gambar pohon status dari pemecahan permasalahan *Quadratic Assignment Problem* dengan pendekatan *Branch and Bound*. Angka *n* pada gambar diatas menunjukkan jumlah fasilitas yang akan ditempatkan pada *K* buah lokasi. Setiap kedalaman menentukan fasilitas mana (fasillitas 1 hingga n) yang ditempatkan pada lokasi tertentu.

1. Kedalaman 1 menentukan fasilitas yang akan ditempatkan pada lokasi pertama (A).
2. Kedalaman 2 menentukan fasilitas yang akan

ditempatkan pada lokasi kedua (B).

3. Begitu seterusnya hingga kedalaman K, dimana *goal node* berada.

Perlu diperhatikan lagi, pada kedalaman 0 dan 1, cost yang terhitung adalah 0 karena pada fungsi nilai taksiran dari QAP membutuhkan setidaknya 2 fasilitas telah ditempatkan pada 2 lokasi untuk dapat dihitung.

Untuk membuktikan kevalidan model pohon status tersebut akan dilakukan suatu pengujian. Berikut adalah data yang dipakai untuk melakukan pengujian. Data tersebut berupa matriks *flow*/alur dan matriks jarak yang didefinisikan sebagai berikut :

Matrix Flow

	1	2	3	4
1	0	3	0	2
2	3	0	0	1
3	0	0	0	4
4	2	1	4	0

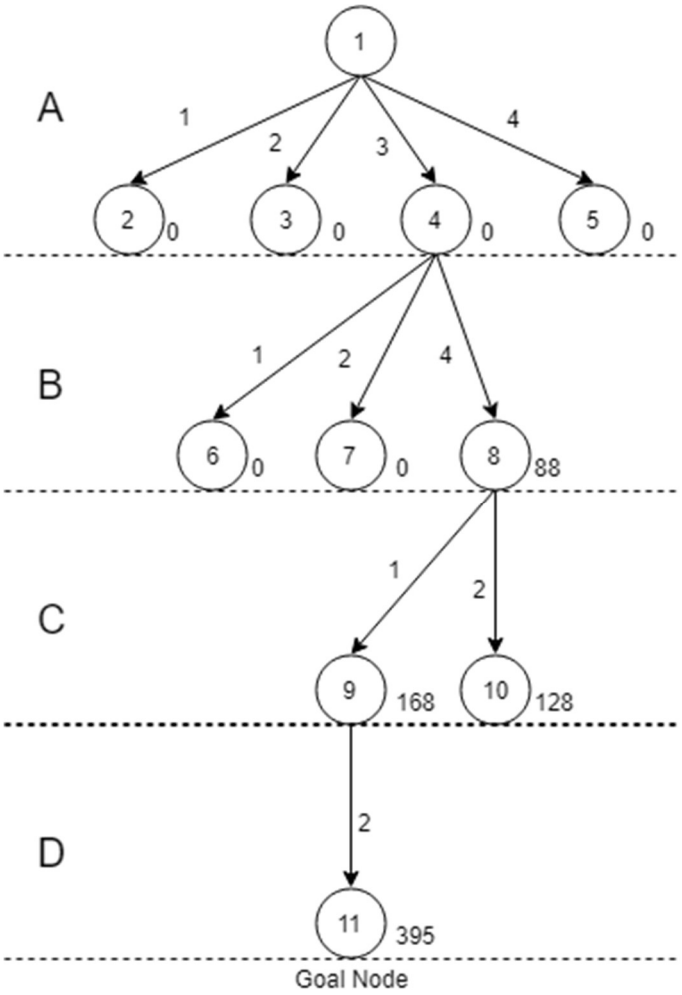
Tabel 3. Matrix Flow Antar Fasilitas Sumber : https://neos-guide.org/content/qap4

Matrix Distance

	A	B	C	D
A	0	22	53	53
B	22	0	40	62
C	53	40	0	55
D	53	62	55	0

Tabel 4. Matrix Jarak Antar Lokasi Sumber : https://neos-guide.org/content/qap4

Dari data tersebut dapat kita modelkan kedalam model pohon status yang telah ditentukan.



Gambar 4. Pohon Status Dari QAP

Gambar 4 merupakan pohon status yang dihasilkan berdasarkan data diatas. Pada pohon status diatas tidak ditampilkan pohon status dengan lengkap karena banyaknya simpul yang perlu dibangkitkan. Namun dari gambar diatas sendiri dapat membuktikan bahwa permasalahan *Quadratic Assignment Problem* dapat diselesaikan menggunakan pendekatan algoritma *Branch and Bound*.

Pada gambar 4, kita dapat melihat bahwa simpul akar membangkitkan 4 simpul anak yang masing-masing mewakili status fasilitas apa yang akan dibangun dilokasi A. Begitu juga dengan simpul 4 akan membangkitkan 3 simpul yang mewakili fasilitas apa yang akan dibangun pada lokasi B. Begitu seteresnya hingga mencapai simpul tujuan. Dari gambar diatas dapat ditarik kesimpulan, kombinasi fasilitas yang dibangun untuk mendapatkan cost yang minimum pada lokasi A, B, C dan D secara beurutan adalah fasilitas 3, 4 ,1 dan 2.

Perlu diperhatikan bahwa model ini belum tentu model yang efisien untuk menyelesaikan permasalahan QAP dengan pendekatan algoritma *Branch and Bound*. Karena dalam pembuatan pohon berdasarkan data diatas, setidaknya harus mengekspansi seluruh simpul dengan kedalaman 0 hingga K-2, dimana K merupakan jumlah kedalaman dari pohon. Sehingga

B. Perbandingan

1. Brute Force

Brute Force merupakan sebuah strategi algoritma yang memecahkan masalah dengan sangat sederhana (*simple*), langsung (*straightforward*) dan dengan cara yang jelas (*obvious way*) atau dengan kata lain adalah strategi algoritma naif.

Cara penyelesain *Quadratic Assignment Problem* dengan menggunakan strategi algoritma Brute Force adalah dengan cara membandingkan seluruh kombinasi fasilitas yang ada. Kombinasi dengan cost terkecil merupakan kombinasi solusi. Namun karena kita memerlukan membandingkan seluruh kombinasi yang mungkin untuk mendapatkan kombinasi yang diinginkan maka jelas Branch and Bound merupakan pendekatan yang lebih baik dibandingkan dengan pendekeatan Brute Force. Kompleksitas waktu Brute Force dalam menyelesaikan persolaan *Quadratic Assignment Problem* sama dengan kompleksitas waktu terburu

2. Greedy

Greedy merupakan pendekatan yang mengambil keputusan terbaik setiap langkah namun tanpa melihat konsekuensi yang akan terjadi jika mengambil langkah tersebut. Pendekatan ini tidak selalu mendapatkan hasil yang diinginkan (optimum global).

Bisa dilihat pada gambar 4. Bila kita memutuskan untuk mencari nilai minimum dengan cara pendekatan greedy, maka kita akan selalu mengambil simpul dengan cost paling sedikit. Pada gambar 4, simpul 8 merupakan simpul yang menuju simpul tujuan, namun algoritma greedy akan memilih simpul 6 atau 7 karena nilai tersebut merupakan nilai paling minimum dibandingkan simpul 8. Contoh lainnya adalah pada simpul 8. Pada algoritma greedy, ia akan lebih memilih simpul 10 dibandingkan dengan simpul 9 yang merupakan simpul yang menuju simpul tujuan (goal node). Oleh karena itu pendekatan *Branch and Bound* lebih diutamakan untuk menyelesaikan permasalahan QAP dibandingkan dengan pendengan greedy. Namun apabila strategi algoritma greedy menghasilkan kombinasi yang tepat, waktu komputasi yang dibutuhkan akan sangat cepat.

V. KESIMPULAN'

Quadratic Assignment Problem merupakan salah satu permasalahan kombinatorial tersulit. Sehingga pada permasalahan ini, efisiensi penyelesain permasalahan masalah QAP dengan *Branch and Bound* tidak berbeda jauh dengan algoritma *Brute Force*. Walaupun hasil yang dihasilkan bukan merupakan solusi yang paling mangkus, strategi algoritma *Branch and Bound* merupakan salah satu solusi alternatif dalam menyelesaikan permasalahn *Quadratic Assignment Problem* dibandingkan dengan beberapa strategi algoritma yang lainnya seperti greedy ataupun strategi algoritma.

VI. SARAN

Untuk memberikan perbandingan yang lebih akurat, dapat dilakukan implementasi algoritma diatas pada suatu bahasa tertentu, sehingga terlihat hasil dan waktu komputasi yang dibutuhkan. Dapat juga dilakukan implementasi lainnya dengan berbagai macam strategi algoritma lainnya seperti algoritma Greedy, runut-balik, ataupun yang lainnya. Kemudian dilakukan perbandingan dengan paramater yang sama sehingga dapat menghasilkan strategi yang tepat untuk menangani permasalahan Quadratic Assignment Problem.

VII. UCAPAN TERIMA KASIH

Puji syukur saya panjatkan ke hadirat Allah SWT yang mana telah memberikan saya kekuatan, ketabahan dan inspirasi dalam menyelesaikan makalah ini. Terima kasih saya ucapkan kepada Bapak Rinaldi Munir, Bu Nur Ulfa Maulidevi, dan Bu Masayu Leylia Khodra selaku dosen pengajar mata kuliah IF2211 Strategi Algoritma yang telah membimbing saya selama satu semester ini. Tak lupa saya juga berterima kasih kepada rekan - rekan, saudara – saudara dan orang tua yang telah memberikan semangat dan bantuan selama masa pengerjaan makalah ini hingga makalah ini selesai.

REFERENCES

[1] Munir, Rinaldi, *Diktat Kuliah IF 2211 : Strategi Algoritma*, Bandung, 2018.

[2] https://neos-guide.org/content/quadratic-assignment-problem#references Diakses pada tanggal 12 Mei 2018, pukul 20.00.

[3] https://neos-guide.org/content/qap4 Diakses pada tanggal 12 Mei 2018 pukul 20.05

[4] Sharifah Shuthairah Syed-Abdullah et al, 2018, *Solving Quadratic Assignment Probllem with Fixed Assignment (QAPFA) using Branch and Bound*, IOP Publishing, http://iopscience.iop.org/article/10.1088/1757-899X/300/1/012002/pdf, Diakses tanggal 13 Mei 2018 pukul 16.00.

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, 13 Mei 2018

Muhamad Ilyas Mustafa
13516123