**PENERAPAN ALGORITMA BRANCH AND BOUND PADA RUTE JARAK GEDUNG KAMPUS J1 – J5 UNIVERSITAS GUNADARMA**

**LATAR BELAKANG**

Jarak sering kali dipermasalahkan oleh mahasiswa terutama ketika jam kuliah saling berdekatan dan harus berpindah ke gedung kampus lainnya. Terlebih lagi mahasiswa baru yang harus pindah ke gedung lainnya tetapi hanya tau 1 jalur yang belum tentu jalur itu adalah jalur tercepat yang ada.

Untuk menghindari keterlambatan para mahasiswa baru, di makalah ini akan dibahas tentang jalur tercepat antar gedung kampus J1 sampai J5 dengan melewati gedung-gedung tersebut tanpa memperhatikan kemacetan, jalannya bagus atau tidak, penutupan jalan, dll

Untuk menyelesaikan jalur tercepatnya menggunakan “Algoritma Branc & Bound”, karena algoritma B&B ini cocok untuk menentukan Traversal Salesman Problem karena B&B ini akan mencari solusi dari berbagai permasalahan optimasi.

Dengan dibuatnya makalah ini diharapkan mahasiswa tidak kesulitan lagi dalam mencari jalur tercepat antar kampus J Universitas Gunadarma.

**BATASAN MASALAH**

Pada pembahasan pencarian Travelling Salesman Problem ini, penulis membatasi masalah sebagai berikut:

1. Hanya menghitung jarak antar rute tanpa menghitung kemacetan, lampu merah, dan hambatan perjalanan lainnya.
2. Semua jalur yang digunakan diidentifikasi sebagai jalur 2 arah.
3. Pemilihan formula fungsi menggunakan metode herusitic, sehingga tidak ada bukti matematis.
4. Hasil yang diperoleh dari algoritma ini bergantung pada keakuratan pemilihan fungsi batas tersebut. Tidak ada cara baku untuk menentukan fungsi batas.

**TUJUAN**

Tujuan dari pembuatan makalah ini bahwa penerapan *algoritma branch and bound* dapat digunakan untuk menyelesaikan permasalah TSP atau Travelling Salesman Problem dalam mencari rute tercepat antar gedung kampus J1 sampai J5 Universitas Gunadarma.

**ISI**

Universitas Gunadarma memiliki sejumlah gedung kampus yang tersebar di beberapa kota besar yaitu Depok, Jakarta, dan Bekasi. Untuk gedung kampus Universitas Gunadarma yang berlokasi di bekasi memiliki kode gedung kampus yaitu J yang tersebar dibeberapa wilayah sekitaran kota bekasi. Pada kasus kali ini penulis ingin mencari rute tercepat (optimalisasi) dari jarak antar gedung kampus J1 sampai J5. Berikut tabel jarak antar gedung kampus J1 sampai J5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kampus** | **J1** | **J2** | **J3** | **J4** | **J5** |
| **J1** |  | **650 m** | **6850 m** | **4800 m** | **10000 m** |
| **J2** | **650 m** |  | **6200 m** | **4200 m** | **9400 m** |
| **J3** | **6850 m** | **6200 m** |  | **6200 m** | **11000 m** |
| **J4** | **4800 m** | **4200 m** | **6200 m** |  | **14850 m** |
| **J5** | **10000 m** | **9400 m** | **11000 m** | **19350 m** |  |

**Tabel Jarak antar gedung kampus J1-J5 Universitas Gunadarma**

Langkah-langkah penyelesaian masalah Travelling Salesman Problem (TSP) pada kasus ini menggunakan *algoritma branch and bound* adalah sebagai berikut.

* **Langkah 1.**

Tetapkan penyelesaian awal masalah. Penyelesaian yang ditetapkan merupakan rute perjalanan lengkap. Tentukan batas tertinggi pada nilai minimum fungsi objektif dengan mencari berbagai kemungkinan rute perjalanan. Batas atas ini dinotasikan dengan *fu*.

Cara penyelesaian langkah 1.

* X12 = X23 = X34 = X45 = X51 = 1 dengan masing – masing nilai ditambah

= 650 + 6200 + 6200 + 19350 + 10000 = 42.400 . sehingga nilai 42.400

* **Langkah 2.**

Buat cabang awal dengan mengatur X1 = 1 untukn masing-masing kampus *j = 2, 3,….., n*. Untuk *i = j*, nilai *Cij = M* untuk menyatakan rute yang tidak mungkin. Hitung batas terendah yang dinotasikan *fL* pada nilai minimum fungsi objektif di setiap titik. Dari data awal, hapus baris pertama dan kolom ke *j* serta ganti *Cij = M.* tentukan penyelesaian masalah dan tambahkan harga *f* ke *Cij* untuk memperoleh *fL* sehingga *fL = Cij + f*. Jika *fL* ≤ *fU* aktifkan simpul dan jika sebaliknya hapus simpul.

Cara penyelesaian langkah 2.

* Buat cabang X12 = 1 (simpul 2), X13 = 1 (simpul 3), X14 = 1 (simpul 4), X15 = 1 (simpul 5).

Pada **Simpul 2** hapus baris pertama dan kolom kedua dari data awal serta ganti *C21 = M* , sehingga diperoleh :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kampus** | **J1** | **J3** | **J4** | **J5** |
| **J2** | **M** | **6200 m** | **4200 m** | **9400 m** |
| **J3** | **6850 m** | **M** | **6200 m** | **11000 m** |
| **J4** | **4800 m** | **6200 m** | **M** | **14850 m** |
| **J5** | **10000 m** | **11000 m** | **19350 m** | **M** |

Penyelesaiannya adalah X23 = X35 = X54 = X41 = 1 dengan nilai 6200 + 11000 + 14850 + 4800 = 36.850, sehingga nilai *f* = 36.850. Jadi *fL* = C12 + *f* = 650 + 36850 = 37.500. 37.500. kemudian simpan penyelesaian baru dan simpul 2 sebagai simpul aktif sementara.

Pada **Simpul 3** hapus baris pertama dan kolom ketiga dari data awal serta ganti C31 = M , sehingga diperoleh :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kampus** | **J1** | **J2** | **J4** | **J5** |
| **J2** | **650 m** | **M** | **4200 m** | **9400 m** |
| **J3** | **M** | **6200 m** | **6200 m** | **11000 m** |
| **J4** | **4800 m** | **4200 m** | **M** | **14850 m** |
| **J5** | **10000 m** | **9400 m** | **19350 m** | **M** |

Penyelesaiannya adalah X35 = X54 = X42 = X21 = 1 dengan nilai 11000 + 14850 + 4200 + 650 = 30.700, sehingga nilai *f* = 30.700. Jadi *fL* = C13 + *f* = 6850 + 30700 = 37.550. Karena nilai *fL* > maka hapus simpul 3.

Pada **Simpul 4** hapus baris pertama dan kolom keempat dari data awal serta ganti C41 = M, sehingga diperoleh :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kampus** | **J1** | **J2** | **J3** | **J5** |
| **J2** | **650 m** | **M** | **6200 m** | **9400 m** |
| **J3** | **6850 m** | **6200 m** | **M** | **11000 m** |
| **J4** | **4800 m** | **4200 m** | **6200 m** | **14850 m** |
| **J5** | **10000 m** | **9400 m** | **11000 m** | **M** |

Penyelesaiannya adalah X43 = X35 = X52 = X21 = 1 dengan nilai 6200 + 11000 + 9400 + 650 = 27.250, sehingga nilai *f* = 27.250. Jadi *fL* = C14 + *f* = 4800 + 27250 = 32.050. Karena nilai *fL* < maka simpan penyelesaian ini sebagai penyelesaian baru lalu hapus simpul 2 dan atur simpul 4 sebagai simpul aktif sementara.

Pada **Simpul 5** hapus baris pertama dan kolom kelima dari data awal serta ganti C51 = M, sehingga diperoleh :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kampus** | **J1** | **J2** | **J3** | **J4** |
| **J2** | **650 m** | **M** | **6200 m** | **4200 m** |
| **J3** | **6850 m** | **6200 m** | **M** | **6200 m** |
| **J4** | **4800 m** | **4200 m** | **6200 m** | **M** |
| **J5** | **10000 m** | **9400 m** | **11000 m** | **19350 m** |

Penyelesaiannya adalah X53 = X34 = X42 = X21 = 1 dengan nilai 11000 + 6200 + 4200 + 650 = 22.050, sehingga nilai *f* = 22.050. Jadi *fL* = C15 + *f* = 10000 + 22050 = 32.050. Karena nilai *fL* < maka simpan penyelesaian ini sebagai penyelesaian baru lalu hapus simpul 4 dan atur simpul 5 sebagai simpul aktif sementara.

* **Langkah 3**

Jika tidak ada simpul aktif pada langkah ini maka penyelesaian terbaik saat ini adalah optimal. Jika tidak, pilih simpul dengan nilai *fL* terkecil dan buat cabang baru dengan mengatur Xjk = 1 untuk setiap gedung yang belum dikunjungi sebelumnya.

Cara penyelesaian langkah 3.

* Terdapat satu simpul aktif di simpul 5 dengan *fL*= 32.050. buat cabang X32 = 1 (simpul 6), X34 = 1 (simpul 7), X35 = 1 (simpul 8). Buat batasan pada simpul 6, 7, 7, dengan memodifikasi data pada simpul 5.

Pada **simpul 6** hapus baris kedua dan kolom pertama dari data di simpul 5 serta ganti C52. Sehingga diperoleh:

|  |  |  |  |
| --- | --- | --- | --- |
| **Kampus** | **J2** | **J3** | **J4** |
| **J3** | **6200 m** | **M** | **6200 m** |
| **J4** | **4200 m** | **6200 m** | **M** |
| **J5** | **M** | **11000 m** | **19350 m** |

Penyelesaiannya adalah

X53 = x34 = x42 = 1 dengan nilai 11000 + 6200 + 4200 = 21.400, dengan *f* = 21.400. jadi *fL* = C13 + C32 + *f* = 6850 + 6200 + 21.400 = 34.450 Atur 34.450, kemudian simpan penyelesaian ini dengan penyelesaian baru dan simpul 6 sebagai simpul aktif sementara.

Pada **simpul 7** hapus baris kelima dan kolom keempat dari data di simpul 5 serta ganti C41=M, sehingga diperoleh:

|  |  |  |  |
| --- | --- | --- | --- |
| **Kampus** | **J1** | **J2** | **J3** |
| **J2** | **650 m** | **M** | **6200 m** |
| **J3** | **6850 m** | **6200 m** | **M** |
| **J4** | **M** | **4200 m** | **6200 m** |

Penyelesaiannya adalah X43 = x32 = x21 = 1 dengan nilai 6200 + 6200 + 650 = 13.050, dengan *f* = 13.050. jadi *fL* = C13 + C34 + *f* = 6850 + 6200 + 13.050 = 26.100 karena nilai *fL* < maka simpan penyelesaian ini sebagai penyelesaian baru lalu hapus simpul 6 dan atur simpul 7 sebagai simpul aktif sementara.

Pada **simpul 8** hapus baris ketiga dan kolom kelima dari data di simpul 5 serta ganti baris keempat kolom pertama serta ganti C54 = M, sehingga diperoleh:

|  |  |  |  |
| --- | --- | --- | --- |
| **Kampus** | **J2** | **J3** | **J4** |
| **J2** | **M** | **6200 m** | **4200 m** |
| **J3** | **6200 m** | **M** | **6200 m** |
| **J5** | **9400 m** | **11000 m** | **M** |

Penyelesaiannya adalah X53 = x32 = x24 = 1 dengan nilai 11000 + 6200 + 4200 = 21.400, dengan *f* = 21.400. jadi *fL* = C13 + C34 + *f* = 6850 + 6200 + 21.400 = 34.450 karena nilai *fL* < maka simpan penyelesaian ini sebagai penyelesaian baru lalu hapus simpul 7 dan atur simpul 8 sebagai simpul aktif.

* **Langkah 4**

Buat batasan *f* pada setiap simpul dengan menghapus baris *j* dan kolom *k* dari data pada simpul aktif diatasnya. Tambahkan nilai *f* ke Cjk dengan seluruh nilai sebelumnya.

Cara penyelesaian langkah 4.

* Terdapat satu simpul aktif yaitu simpul 8 dengan *fL* = 34.450. Buat cabang X52 = 1 (Simpul 9), X53 = 1 (Simpul 10).

Pada **simpul 9** hapus baris kelima dan kolom ketiga dari data di simpul 8 serta ganti C23 = M, Sehingga diperoleh :

|  |  |  |
| --- | --- | --- |
| **Kampus** | **J3** | **J4** |
| **J2** | **M** | **4200 m** |
| **J3** | **M** | **6200 m** |

Penyelesaiannya adalah X24 = X41 = 1 dengan nilai 4200 + 6200 = 10.400, dengan *f* = 10.400.

Jadi

*f1*= C13 + C35 + C52 + *f*

= 6850 + 11000 + 9400 + 10400

= 37.650

Karena *fL* > , Maka hapus simpul 9.

Pada **simpul 10** hapus baris kelima dan kolom ke empat dari data di simpul 8, Sehingga diperoleh:

|  |  |  |
| --- | --- | --- |
| **Kampus** | **J2** | **J3** |
| **J2** | **M** | **6200 m** |
| **J3** | **6200 m** | **M** |

Penyelesaiannya adalah X32 = X23 = 1

Dengan nilai 6200 + 6200 = 12.400, dengan *f*  = 12.400.

Jadi :

*f1*= C13 + C35 + C52 + *f*

= 6850 + 11000 + 9400 + 12400

= 39.650

Karena *fL* > , Maka hapus simpul 10.

Tidak ada simpul yang aktif. Terdapat alternative perjalanan yang optimal yaitu 1-2-4-3-5-1 dengan nilai 32.050

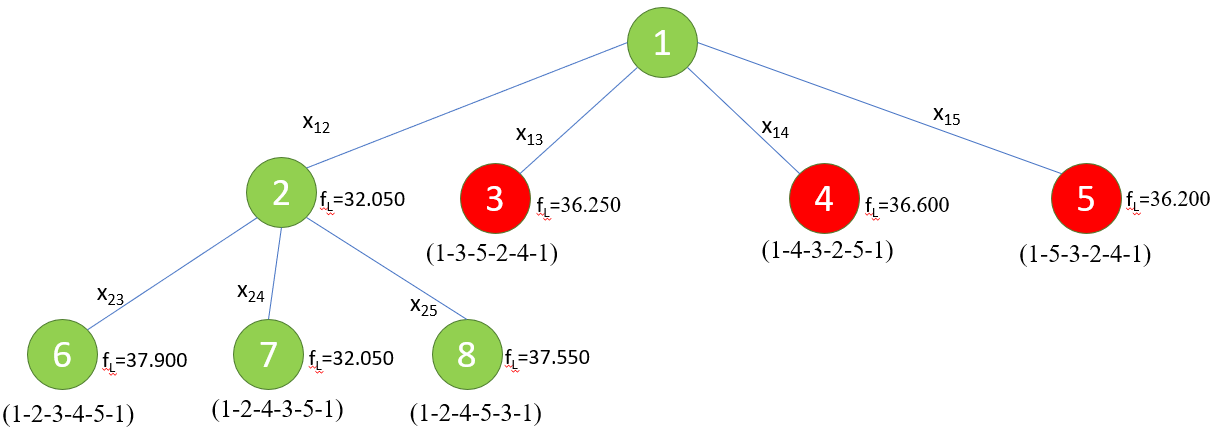
**Perbandingan Hasil Algoritma Branch and Bound dengan Exhaustive Search**

Hasil iterasi pada exhaustive search sebagai berikut

|  |  |  |  |
| --- | --- | --- | --- |
| Alternatif Solusi | Nilai | Alternatif Solusi | Nilai |
| 1-2-3-4-5-1 | 650 + 6200 + 6200 + 14850 + 10000 = 37.900 | 1-4-3-5-2-1 | 4800 + 6200 + 11000 + 9400 + 650 = 32.050 |
| 1-2-3-5-4-1 | 620 + 6200 + 11000 + 19350 + 4800 = 42.000 | 1-4-3-2-5-1 | 4800 + 6200 + 6200 + 9400 + 10000 = 36.600 |
| 1-2-4-5-3-1 | 650 + 4200 + 14850 + 11000 + 6850 = 37.550 | 1-4-5-2-3-1 | 4800 + 19350 + 9400 + 6200 + 6850 = 46.600 |
| 1-2-4-3-5-1 | 650 + 4200 + 6200 + 11000 + 10000 = 32.050 | 1-4-5-3-2-1 | 4800 + 19350 + 11000 + 6200 + 650 = 42.000 |
| 1-2-5-3-4-1 | 650 + 9400 + 11000 + 6200 + 4800 = 32.050 | 1-4-2-3-5-1 | 4800 + 4200 + 6200 + 11000 + 10000 = 36.200 |
| 1-2-5-4-3-1 | 650 + 9400 + 19350 + 6200 + 6850 = 42.450 | 1-4-2-5-3-1 | 4800 + 4200 + 9400 + 11000 + 6850 = 36.250 |
| 1-3-4-5-2-1 | 6850 + 6200 + 14850 + 9400 + 650 = 37.950 | 1-5-4-3-2-1 | 10000 + 19350 + 6200 + 6200 + 650 = 42.400 |
| 1-3-4-2-5-1 | 6850 + 6200 + 4200 + 9400 + 10000 = 36.650 | 1-5-4-2-3-1 | 10000 + 19350 + 4200 + 6200 + 6850 = 46.600 |
| 1-3-5-2-4-1 | 6850 + 11000 + 9400 + 4200 + 4800 = 36.250 | 1-5-3-2-4-1 | 10000 + 11000 + 6200 + 4200 + 4800 = 36.200 |
| 1-3-5-4-2-1 | 6850 + 11000 + 14850 + 4200 + 650 = 37.550 | 1-5-3-4-2-1 | 10000 + 11000 + 6200 + 4200 + 650 = 32.050 |
| 1-3-2-4-5-1 | 6850 + 6200 + 4200 + 19350 + 10000 = 46.600 | 1-5-2-4-3-1 | 10000 + 9400 + 4200 + 6200 + 6850 = 36.650 |
| 1-3-2-5-4-1 | 6850 + 6200 + 9400 + 14850 + 4800 = 42.100 | 1-5-2-3-4-1 | 10000 + 9400 + 6200 + 6200 + 4800 = 36.600 |

**Tabel *Exhaustive search***

Dari tabel di atas terlihat bahwa penyelesaian optimal yang diperoleh dengan *exhaustive search* sama dengan algoritma *branch and bound* yaitu 1-2-4-3-5-1 dengan nilai optimal 32.050. karena *exhaustive search* akan memberikan hasil 100% benar, maka untuk kasus ini hasil yang diperoleh melalui algoritma *branch and bound* juga pasti benar.



**KESIMPULAN**

Banyak metode untuk menyelesaikan permasalahan optimisasi kombinatorial, salah satu diantaranya adalah algoritma branch and bound. Algoritma ini menggunakan skema Breadth First Search (BFS) yang lebih pintar, yaitu menggunakan fungsi pembatas bound untuk menentukan simpul expand yaitu simpul yang akan diperluas berikutnya. Kelemahan dari algoritma ini adalah sama seperti exhaustive search yaitu n!, namun hal ini sangat jarang terjadi karena algoritma ini menggunakan fungsi pembatas dalam pencarian solusi.

Keakuratan penyelesaian optimal dari algoritma branch and bound, sangat tergantung pada fungsi pembatas yang dipilih. Formula dipilih berdasarkan insting dan pengalaman sehingga terkadang tidak memberikan hasil yang optimal. Namun dari aspek waktu, algoritma ini bisa menjadi pilihan dalam menyelesaikan masalah optimisasi kombinatorial.