

Optimasi Rute Distribusi Hasil Food Estate Berbasis Indeks Ketahanan Pangan (IKP) di Indonesia Timur dengan Algoritma Branch and Bound

Izhar Alif Akbar - 18223129

Program Studi Sistem dan Teknologi Informasi
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: iizharalif@gmail.com , 18223129@std.stei.itb.ac.id

Abstract—Program Food Estate di Indonesia Timur, khususnya di Merauke, Papua Selatan, dirancang sebagai pusat produksi pangan nasional dalam skala besar. Namun, efektivitasnya bergantung pada sistem distribusi yang efisien dan tepat sasaran. Penelitian ini mengusulkan pendekatan optimasi rute distribusi dengan mengintegrasikan Indeks Ketahanan Pangan (IKP), populasi, dan Indeks Kemahalan Konstruksi (IKK) ke dalam model biaya distribusi, kemudian diselesaikan menggunakan algoritma Branch and Bound berbasis *reduced cost matrix*. Permasalahan dimodelkan sebagai variasi *Asymmetric Travelling Salesman Problem* (ATSP) dengan delapan node yang terdiri atas satu depot Food Estate Merauke dan tujuh provinsi tujuan di Indonesia Timur. Algoritma Branch and Bound dengan strategi *best-first search* menghasilkan rute optimal dengan total cost 1128.229 dan total jarak 6564.98 km. Dari 5040 kemungkinan rute yang ada, hanya 170 node pencarian yang dibuat dan 132 di antaranya dipangkas, sehingga proses pencarian berlangsung dalam 19.49 ms. Hasil ini identik dengan Brute Force, yang membuktikan optimalitas solusi. Metode Greedy Nearest Neighbor menghasilkan cost 1269.023, menunjukkan bahwa pendekatan greedy tidak menjamin solusi optimal. Implementasi membuktikan bahwa integrasi IKP dalam fungsi biaya menghasilkan prioritisasi yang berbeda dibanding optimasi berbasis jarak murni, dengan wilayah berkerawanan pangan tinggi memperoleh kunjungan lebih awal dalam urutan distribusi.

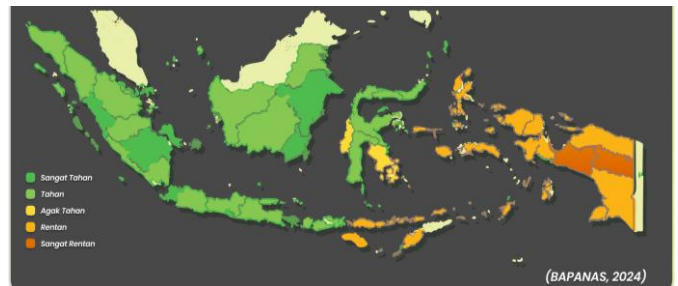
Keywords—Branch and Bound, Reduced Cost Matrix, Indeks Ketahanan Pangan, Food Estate, Optimasi Rute Distribusi, ATSP

I. PENDAHULUAN

A. Latar Belakang

Ketahanan pangan merupakan salah satu aspek strategis dalam pembangunan nasional karena berkaitan langsung dengan kesejahteraan masyarakat dan stabilitas ekonomi. Pemerintah Indonesia mengembangkan program *Food Estate* sebagai upaya memperkuat ketahanan pangan nasional melalui pembangunan kawasan lumbung pangan skala besar, terutama di wilayah Indonesia Timur. Program ini diharapkan dapat meningkatkan kapasitas produksi pangan sekaligus pemerataan distribusi ke daerah-daerah rentan terhadap kerawanan pangan. Berdasarkan data Badan Pangan Nasional (Bapanas) tahun 2024, sejumlah provinsi di Indonesia Timur mencatatkan Indeks Ketahanan Pangan (IKP) yang secara konsisten berada

di bawah rata-rata nasional, antara lain Nusa Tenggara Timur (49.53), Papua (53.12), dan Papua Barat (49.59) [1].



Gambar. 1. Peta Persebaran Indeks Ketahanan Pangan (IKP)

Food Estate Merauke, yang berlokasi di Provinsi Papua Selatan, diproyeksikan sebagai produsen komoditas pangan utama yang dapat memenuhi kebutuhan pangan regional maupun nasional. Keberhasilan program tersebut, bagaimanapun, tidak hanya ditentukan oleh kapasitas produksi, melainkan juga oleh efektivitas sistem distribusi yang mampu menyalurkan hasil pangan ke wilayah-wilayah yang membutuhkan secara tepat sasaran.

Indonesia Timur memiliki karakteristik geografis yang kompleks, seperti rentang jarak antarpulau yang besar, keterbatasan infrastruktur darat, ketergantungan terhadap moda transportasi laut, serta disparitas kondisi wilayah yang signifikan. Akibatnya, distribusi pangan kerap menghadapi beberapa kendala seperti biaya logistik yang tinggi dan ketidaktepatan prioritas pengiriman.

B. Rumusan Masalah

Sistem perencanaan distribusi yang ada saat ini umumnya hanya mempertimbangkan efisiensi jarak atau biaya transportasi semata, tanpa memperhatikan tingkat urgensi kebutuhan pangan pada wilayah tujuan. Permasalahan utama yang diidentifikasi adalah sebagai berikut.

- 1) **Distribusi Berbasis Jarak Semata.** Pendekatan konvensional mengabaikan tingkat kerawanan pangan wilayah tujuan. Sebagai ilustrasi, apabila terdapat dua provinsi dengan jarak berbeda dari depot, sistem konvensional akan mendahulukan provinsi yang lebih dekat secara geografis meskipun provinsi yang lebih jauh

memiliki IKP lebih rendah dan kebutuhan pangan yang jauh lebih mendesak.

- 2) **Kompleksitas Kombinatorial.** Jika terdapat satu depot Food Estate dan n wilayah tujuan, jumlah kemungkinan urutan distribusi adalah $n!$. Untuk skenario dengan tujuh wilayah tujuan, terdapat $7! = 5040$ kemungkinan rute. Evaluasi seluruh kemungkinan menggunakan metode brute force menjadi tidak skalabel seiring bertambahnya jumlah wilayah tujuan.
- 3) **Heterogenitas Biaya Distribusi.** Dua wilayah yang memiliki jarak fisik serupa belum tentu memiliki biaya operasional yang sama, karena kondisi infrastruktur, kemahalan konstruksi, dan skala populasi berbeda antarsatu wilayah dengan wilayah lain.

C. Solusi yang Diusulkan

Penelitian ini mengusulkan penggunaan algoritma Branch and Bound berbasis *reduced cost matrix* untuk menyelesaikan permasalahan optimasi rute distribusi Food Estate. Masalah dimodelkan sebagai *Asymmetric Travelling Salesman Problem* (ATSP), yaitu pencarian urutan kunjungan dengan total biaya minimum dari depot ke seluruh node tujuan dan kembali ke depot.

Kontribusi utama penelitian ini terletak pada integrasi IKP, populasi, dan IKK ke dalam fungsi biaya distribusi, sehingga rute yang dihasilkan tidak hanya efisien secara logistik, tetapi juga memprioritaskan wilayah yang lebih rentan pangan.

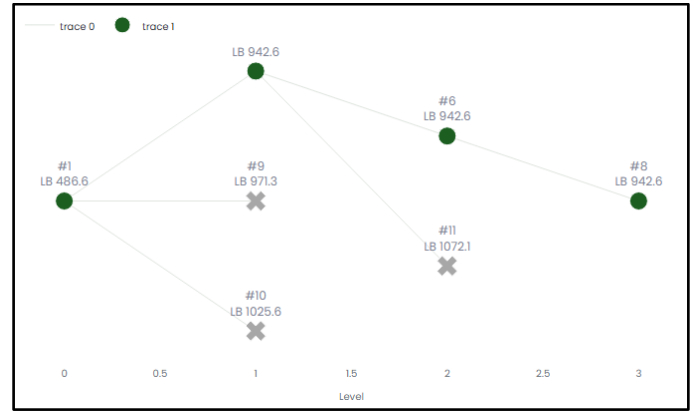
II. LANDASAN TEORI

A. Algoritma Branch and Bound

Branch and Bound adalah suatu paradigma pencarian sistematis yang digunakan untuk menyelesaikan permasalahan optimasi kombinatorial [2]. Algoritma ini beroperasi pada pohon ruang status (*state space tree*) dengan dua mekanisme utama.

- 1) **Branching.** Setiap node pada pohon pencarian diperluas (*expand*) menjadi beberapa *child node* yang merepresentasikan pilihan berikutnya yang mungkin. Pada konteks TSP, setiap percabangan merepresentasikan pemilihan kota berikutnya dalam urutan kunjungan.
- 2) **Bounding.** Setiap *child node* dihitung nilai batas bawahnya (*lower bound*). Apabila *lower bound* suatu node lebih besar atau sama dengan solusi terbaik yang telah ditemukan (*best solution so far*), node tersebut tidak perlu dieksplorasi lebih lanjut karena tidak mungkin menghasilkan solusi yang lebih baik. Proses ini disebut **pruning** [3].

Jaminan optimalitas **Branch and Bound** berasal dari penggunaan *lower bound* yang bersifat *admissible*, yaitu tidak pernah melebihi biaya aktual solusi. Dengan demikian, ketika suatu node dipangkas karena nilai *lower bound*-nya lebih besar daripada solusi terbaik yang telah ditemukan, cabang tersebut dipastikan tidak dapat menghasilkan solusi yang lebih baik. Oleh karena itu, proses pemangkasan dapat dilakukan dengan aman tanpa menghilangkan kemungkinan ditemukannya solusi optimal [2].



Gambar. 2. Ilustrasi Pohon Branch and Bound

Strategi eksplorasi yang digunakan pada penelitian ini adalah **best-first search**, yaitu node dengan *lower bound* terkecil dieksplorasi terlebih dahulu. Strategi ini memungkinkan algoritma menemukan solusi optimal lebih cepat dibanding *depth-first* atau *breadth-first* karena batas atas (*upper bound*) yang baik diperoleh lebih awal, sehingga pruning lebih agresif dapat dilakukan [3].

B. Reduced Cost Matrix

Reduced cost matrix adalah teknik yang digunakan untuk menghitung *lower bound* yang ketat pada permasalahan TSP [4]. Reduksi dilakukan melalui dua tahap berikut.

1. **Reduksi Baris.** Setiap baris i pada matriks cost dikurangi dengan nilai minimum baris tersebut, $r_i = \min_j C_{ij}$. Hal ini valid karena setiap kota pasti harus ditinggalkan melalui salah satu edge, sehingga biaya minimum baris tersebut pasti termasuk dalam solusi manapun.
2. **Reduksi Kolom.** Setelah reduksi baris, setiap kolom j dikurangi dengan nilai minimum kolom tersebut, $c_j = \min_i C'_{ij}$. Hal ini valid karena setiap kota pasti harus dikunjungi melalui salah satu edge.

Lower bound awal diperoleh sebagai jumlah seluruh nilai reduksi:

$$LB_0 = \sum_i r_i + \sum_j r_j$$

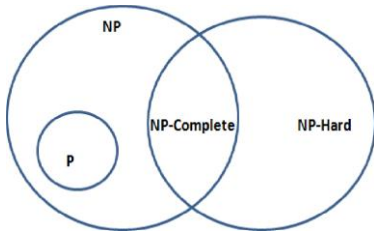
Pada setiap langkah branching, ketika edge (u, v) dipilih, *lower bound* anak diperbarui dengan menambahkan C'_{ij} (biaya edge setelah reduksi sebelumnya) dan mengurangi kembali baris u serta kolom v dari matriks yang tersisa. Hal ini menghasilkan *lower bound* yang semakin ketat seiring bertambah dalamnya pohon pencarian [4].

Perhatikan bahwa matriks cost yang direduksi memiliki setidaknya satu nol pada setiap baris dan kolom yang belum tereliminasi. Nol-nol inilah yang merepresentasikan edge-edge *candidate* yang paling murah pada setiap tahap eksplorasi.

C. Travelling Salesman Problem dan Variannya

Travelling Salesman Problem (TSP) adalah permasalahan optimasi kombinatorial yang bertujuan mencari permutasi

kunjungan n kota dengan total biaya minimum, di mana setiap kota dikunjungi tepat sekali dan perjalanan berakhir kembali di kota asal [5]. TSP termasuk ke dalam kelas NP-Hard, sehingga tidak ada algoritma determinisme yang diketahui dapat menyelesaikannya dalam waktu polinom untuk semua kasus [5].



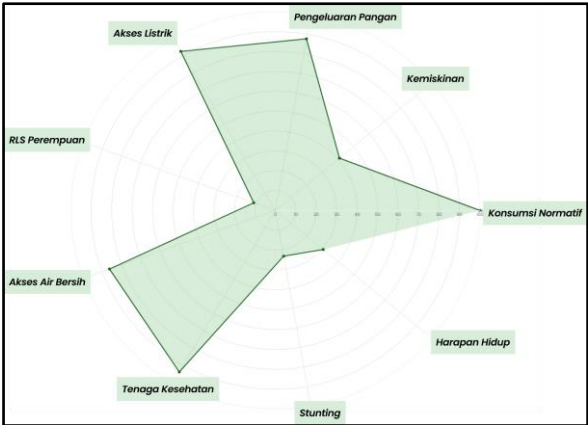
Gambar. 3. Hubungan P, NP, NP-Complete, NP-Hard

Penelitian ini menggunakan variasi **Asymmetric TSP** (ATSP), di mana biaya dari node i ke node j tidak harus sama dengan biaya dari j ke i . Asimetri ini muncul secara alami pada model yang diusulkan karena *PriorityFactor* dan *InfrastructurePenalty* merupakan atribut node tujuan j , sehingga $Cost(i, j) \neq Cost(j, i)$ pada umumnya.

D. Indeks Ketahanan Pangan

IKP merupakan instrumen pengukuran ketahanan pangan yang dikembangkan oleh Bapanas bersama World Food Programme (WFP). IKP dihitung berdasarkan sembilan indikator yang mencakup aspek ketersediaan pangan, aksesibilitas, dan pemanfaatan pangan [1]. Nilai IKP berada pada rentang 0 hingga 100, di mana nilai yang lebih tinggi menunjukkan kondisi ketahanan pangan yang lebih baik.

Pada penelitian ini, IKP digunakan sebagai dasar perhitungan *PriorityFactor*, yaitu faktor yang mencerminkan urgensi distribusi ke suatu wilayah. Semakin rendah IKP suatu wilayah, semakin tinggi prioritasnya untuk menerima distribusi pangan.



Gambar. 4. Grafik Radar Pembentuk Indeks Ketahanan Pangan (IKP)

E. Indeks Kemahalan Konstruksi sebagai Proksi Infrastruktur

Indeks Kemahalan Konstruksi (IKK) yang diterbitkan oleh Badan Pusat Statistik (BPS) mengukur relatif mahalnnya biaya konstruksi suatu daerah dibandingkan rata-rata nasional [6]. Pada penelitian ini, IKK digunakan sebagai proksi kondisi infrastruktur dan hambatan logistik distribusi. Wilayah dengan

IKK tinggi diasumsikan memiliki biaya operasional distribusi yang lebih besar.

III. METODOLOGI

A. Data dan Sumber

Data yang digunakan dalam penelitian ini bersumber dari lembaga resmi pemerintah Indonesia.

TABEL I. DATA DAN SUMBER

Data	Sumber	Keterangan
IKP per provinsi	Bapanas/WFP [1]	9 indikator (0-100)
Populasi per provinsi	BPS [6]	Sensus/proyeksi terbaru
Indeks Kemahalan Konstruksi (IKK)	BPS [6]	Proksi biaya logistik
Koordinat provinsi	Koordinat representatif	Latitude, longitude
Matriks jarak	Dihitung via Haversine	Berdasarkan koordinat

Berdasarkan data di atas, skenario eksperimen default menggunakan delapan node: satu depot (Food Estate Merauke) dan tujuh provinsi tujuan di Indonesia Timur, yaitu Nusa Tenggara Timur, Sulawesi Selatan, Sulawesi Tengah, Maluku, Maluku Utara, Papua Barat, dan Papua.

TABEL II. ATRIBUT NODE MODEL DISTRIBUSI

Node	IKP	Popoulasi	IKK	Lat	Lon
Food Estate Merauke	53.12	542100	134.96	-8.496	140.402
NTT	49.53	5656000	92.42	-10.177	123.607
Sulawesi Selatan	78.422	9463400	95.91	-5.148	119.433
Sulawesi Tengah	68.637	3121800	91.82	-0.9	119.878
Maluku	64.33	1945600	106.52	-3.3695	128.181
Maluku Utara	75.122	1355600	101.425	0.79	127.384
Papua Barat	49.595	578700	124.71	-0.862	134.062
Papua	53.12	1060600	134.96	-2.534	140.718

B. Formulasi Model

Didefinisikan beberapa formulasi model sebagai berikut.

- 1) **Representasi Graf.** Permasalahan distribusi direpresentasikan sebagai graf berarah $G = (V, E)$, di mana V adalah himpunan node yang terdiri atas depot Food Estate dan seluruh wilayah tujuan, sedangkan E adalah himpunan edge yang menghubungkan setiap pasangan

node. Setiap edge $(i, j) \in E$ memiliki bobot berupa nilai *cost* distribusi.

- 2) **Jarak Antarsimpul.** Jarak antara dua node dihitung menggunakan rumus Haversine berdasarkan koordinat geografi masing-masing provinsi karena tidak memerlukan data jalan nyata, sehingga cocok digunakan pada tahap simulasi akademik awal.
- 3) **Formulasi Fungsi Biaya.** Fungsi biaya distribusi dari node i ke node j didefinisikan sebagai berikut:

$$\text{PriorityFactor}(j) = \frac{100 - \text{IKP}[j]}{10} \quad (1)$$

$$\text{InfrastructurePenalty}(j) = \text{normalisasi IKK}[j] \text{ ke skala } [1.0, 2.0] \quad (2)$$

$$\text{PopulationFactor}(j) = \log(\text{Populasi}[j]) \quad (3)$$

Total biaya (*cost*) kemudian dihitung menggunakan persamaan berikut:

$$\text{Cost}(i, j) = \left(\frac{\text{Distance}(i, j) \cdot (1 + \alpha_{\text{norm}})}{\text{PopulationFactor}(j)} \right) \times \text{InfrastructurePenalty}(j)^{\gamma_{\text{norm}}} \cdot \text{PriorityFactor}(j)^{\beta_{\text{norm}}} \quad (4)$$

Keterangan Variabel dan Parameter:

- i, j : Indeks lokasi asal (i) dan lokasi tujuan (j).
- $\text{IKP}[j]$: Indeks Ketahanan Pangan lokasi j .
- $\text{IKK}[j]$: Indeks Kemahalan Konstruksi lokasi j .
- $\alpha_{\text{norm}}, \beta_{\text{norm}}, \gamma_{\text{norm}}$: Parameter penyesuaian yang telah dinormalisasi.

PriorityFactor meningkat ketika IKP wilayah tujuan rendah, sehingga wilayah yang lebih rentan pangan memperoleh nilai *cost* yang lebih besar dalam perhitungan rute. Kondisi ini mendorong algoritma untuk memprioritaskan kunjungan ke wilayah tersebut lebih awal, karena penundaan distribusi ke daerah rawan pangan akan meningkatkan total *cost* rute secara keseluruhan. Sementara itu, **PopulationFactor** berbasis logaritma populasi memberikan efek skala yang lebih proporsional, di mana wilayah dengan jumlah penduduk lebih besar dianggap memiliki kapasitas penyerapan distribusi yang lebih baik sehingga *cost* relatif menjadi lebih rendah dibandingkan wilayah dengan populasi kecil.

Tahapan implementasi algoritma adalah sebagai berikut.

Langkah 1: Inisialisasi. Matriks *cost* $n \times n$ dibentuk berdasarkan formula pada Bagian III-B. Diagonal matriks diisi dengan nilai ∞ agar tidak terdapat edge dari suatu node ke dirinya sendiri.

Langkah 2: Reduksi Awal. Matriks *cost* direduksi baris dan kolom. Jumlah seluruh nilai reduksi menjadi *lower bound* awal LB_0 untuk *root node*.

Langkah 3: Inisialisasi Root Node. *Root node* dibuat dengan path yang hanya berisi depot asal, *lower bound* LB_0 , dan matriks yang telah direduksi. *Root node* dimasukkan ke dalam *priority queue*.

Langkah 4: Iterasi Best-First Search. Selama *priority queue* tidak kosong:

1. Node dengan *lower bound* terkecil dikeluarkan dari *priority queue*.
2. Apabila semua node telah dikunjungi, path lengkap dievaluasi. Jika total *cost* lebih kecil dari solusi terbaik sementara, solusi terbaik diperbarui.
3. Apabila belum semua node dikunjungi, dilakukan *branching* ke setiap node yang belum dikunjungi sebagai kandidat kunjungan berikutnya.

Langkah 5: Perhitungan Lower Bound Child Node. Untuk setiap edge (u, v) yang dipilih saat *branching*:

1. Baris u dan kolom v dihapus dari matriks yang tersisa.
2. Nilai C'_{uv} (biaya edge pada matriks tereduksi) ditambahkan ke *lower bound* induk.
3. Matriks yang tersisa direduksi kembali; jumlah reduksi tambahan dijumlahkan ke *lower bound*.
4. Untuk mencegah *sub-tour*, entri (v, u_0) di mana u_0 adalah depot diset ke ∞

Langkah 6: Pruning. Apabila *lower bound* suatu *child node* lebih besar atau sama dengan solusi terbaik sementara, node tersebut dipangkas dan tidak dimasukkan ke dalam *priority queue*.

Langkat 7: Terminasi. Algoritma berhenti ketika *priority queue* kosong. Solusi terbaik yang tercatat adalah rute distribusi optimal.

Pseudocode algoritma disajikan sebagai berikut.

Algorithm 1 Branch and Bound Solver

Require: *cost_matrix*, *origin*

Ensure: *best_route*, *best_cost*

```

1: reduced, lb0 ← REDUCE(cost_matrix)
2: root ← NODE(path = [origin], matrix = reduced, lb = lb0)
3: pq ← PRIORITYQUEUE()
4: pq.PUSH(root)
5: best_cost ← ∞
6: best_route ← None

7: while pq is not empty do
8:   node ← pq.POP_MIN_LB()
9:   if len(node.path) == n then
10:    total ← node.lb + node.matrix[node.path[-1]][origin]
11:    if total < best_cost then
12:      best_cost ← total
13:      best_route ← node.path + [origin]
14:    end if
15:  else
16:    for v ∈ UNVISITED(node.path) do
17:      child_matrix ← COPY(node.matrix)
18:      child_matrix[node.path[-1]][*] ← ∞
19:      child_matrix[*][v] ← ∞
20:      child_matrix[v][origin] ← ∞
21:      child_reduced, dr ← REDUCE(child_matrix)
22:      child_lb ← node.lb + node.matrix[node.path[-1]][v] + dr
23:      if child_lb < best_cost then
24:        pq.PUSH(NODE(path = node.path + [v], matrix =
          child_reduced, lb = child_lb))
25:      end if
26:    end for
27:  end if
28: end while

```

29: return *best_route*, *best_cost*

C. Metode Pembanding

Dua metode pembanding digunakan untuk mengevaluasi kinerja Branch and Bound.

1. **Greedy Nearest Neighbor.** Pada setiap langkah, dipilih node belum dikunjungi dengan cost terendah dari posisi saat ini. Metode ini memiliki kompleksitas $O(n^2)$ dan berjalan sangat cepat, tetapi tidak menjamin solusi optimal karena keputusan lokal yang optimal tidak selalu menghasilkan solusi global yang optimal [3].
2. **Brute Force.** Seluruh kemungkinan permutasi rute dievaluasi. Metode ini menjamin solusi optimal dengan kompleksitas $O(n!)$, tetapi hanya layak untuk jumlah node kecil. Pada penelitian ini, Brute Force dijalankan untuk memverifikasi optimalitas solusi Branch and Bound.

IV. HASIL DAN PEMBAHASAN

A. Representasi Data dan Matriks Biaya

Atribut node yang digunakan pada skenario default ditampilkan pada Tabel III. Provinsi dengan IKP terendah adalah Nusa Tenggara Timur (49.53) dan Papua Barat (49.59), yang mengindikasikan tingkat kerawanan pangan tertinggi di antara node tujuan yang dipilih.

TABEL III. ATRIBUT NODE SKENARIO DEFAULT

No	Node	IKP	Jarak dari Depot (km)	IKK
1	Food Estate Merauke	53.12	0	-
2	Nusa Tenggara Timur	49.53	1851.97	Tinggi
3	Sulawesi Selatan	78.42	724.04	Sedang
4	Sulawesi Tengah	68.64	474.87	Sedang
5	Maluku Utara	75.12	855.52	Tinggi
6	Maluku	64.33	506.55	Tinggi
7	Papua Barat	49.59	725.34	Sangat Tinggi
8	Papua	53.12	762.78	Tinggi

Matriks cost yang dihasilkan dari formula pada Bagian III-B memperlihatkan sifat asimetris yang cukup jelas. Sebagai contoh, cost dari Food Estate Merauke menuju Nusa Tenggara Timur lebih besar dibandingkan cost pada arah sebaliknya. Perbedaan ini terjadi karena nilai **PriorityFactor** bergantung pada karakteristik wilayah tujuan, sehingga Nusa Tenggara Timur yang memiliki nilai IKP lebih rendah (IKP = 49,53) memperoleh bobot prioritas yang lebih tinggi. Akibatnya, distribusi menuju wilayah tersebut menghasilkan cost yang lebih besar, yang secara tidak langsung mendorong algoritma untuk memprioritaskan kunjungan ke wilayah dengan tingkat kerawanan pangan yang lebih tinggi.

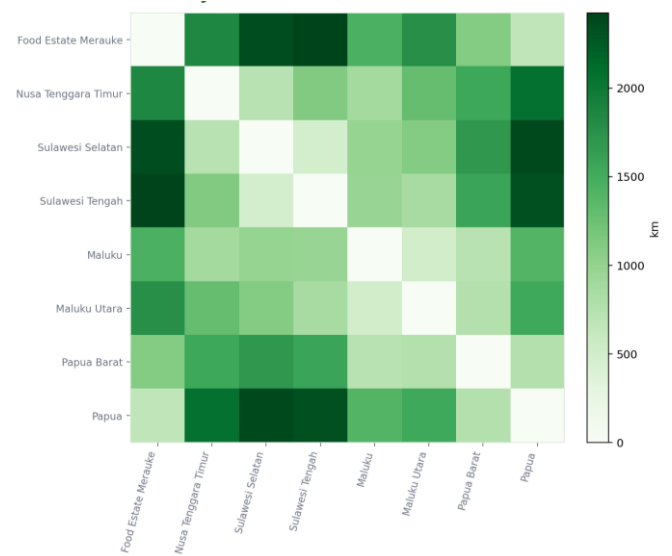
Iterasi 1 - Root matrix direduksi

Path: Food Estate Merauke

Lower bound: 835.817

	0	1	2	3	4	5	6	7
0	INF	117.705	145.189	213.874	103.794	124.046	80.008	0.0
1	314.099	INF	0.0	73.59	58.682	102.484	226.881	340.01
2	439.19	0.0	INF	0.0	94.229	93.474	275.584	421.774
3	468.123	75.311	0.0	INF	104.164	68.692	262.235	421.839
4	240.143	19.873	43.796	66.317	INF	0.0	61.254	213.693
5	300.393	77.882	48.726	39.825	0.0	INF	60.343	231.006
6	126.161	92.659	91.144	115.641	7.696	0.0	INF	44.762
7	0.0	146.973	142.406	192.268	87.81	80.851	0.0	INF

Gambar. 5.Reduced Matrix Awal



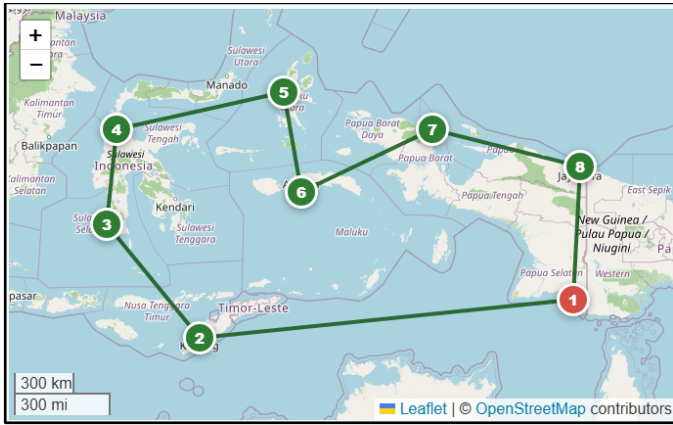
Gambar. 6.Matriks jarak Haversine

B. Rute Distribusi Optimal

Algoritma Branch and Bound menghasilkan rute distribusi optimal sebagai berikut:

Food Estate Merauke → Nusa Tenggara Timur → Sulawesi Selatan → Sulawesi Tengah → Maluku Utara → Maluku → Papua Barat → Papua → Food Estate Merauke

Urutan ini menempatkan Nusa Tenggara Timur (IKP = 49.53) sebagai tujuan pertama meskipun jaraknya paling jauh dari depot (1851.97 km). Hal ini disebabkan oleh nilai **PriorityFactor** Nusa Tenggara Timur yang paling tinggi di antara seluruh node tujuan, sehingga menunda kunjungan ke provinsi ini akan meningkatkan total cost rute secara signifikan. Fenomena ini membuktikan bahwa integrasi IKP dalam fungsi biaya menghasilkan urutan prioritas distribusi yang berbeda secara substantif dibanding optimasi berbasis jarak semata.



Gambar. 7. Peta Rute Optimal

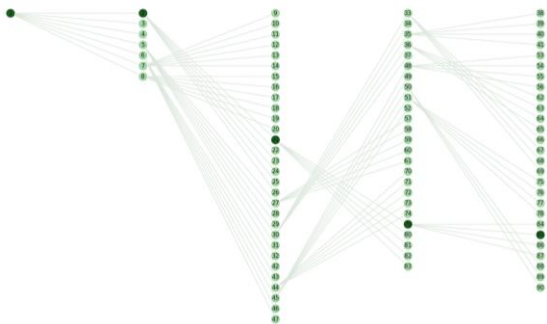
C. Kinerja Algoritma

Tabel IV menyajikan ringkasan kinerja algoritma Branch and Bound pada skenario *default*.

TABEL IV. RINGKASAN KINERJA BRANCH AND BOUND

Metrik	Node
Total cost optimal	1128.229
Total jarak	6564.98 km
Node dibuat	170
Node dieksplorasi	38
Node dipangkas	132
Total kemungkinan rute (7!)	5040
Efisiensi pencarian	3.37% dari seluruh permutasi
Runtime	19.49 ms

Dari 5040 kemungkinan rute yang ada secara teori, algoritma hanya perlu membuat 170 node pencarian (3.37%) dan memangkas 132 node di antaranya. Hal ini menunjukkan efisiensi pruning yang signifikan: lebih dari 96% ruang pencarian berhasil dieliminasi tanpa kehilangan jaminan optimalitas.



Gambar. 8. Potongan Pohon Pencarian Branch and Bound

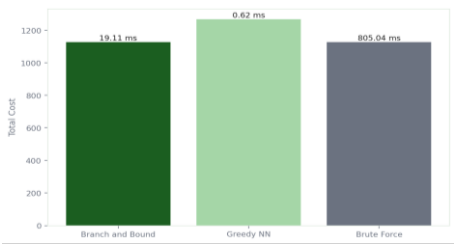
D. Perbandingan Metode

Tabel V membandingkan ketiga metode yang diuji.

TABEL V. PERBANDINGAN METODE

Metode	Total Cost	Runtime	Optimalitas
Branch and Bound	1128.229	19.11 ms	Terjamin
Brute Force	1128.229	~ lebih lama	Terjamin
Greedy Nearest Neighbor	1269.023	< 1 ms	Tidak terjamin

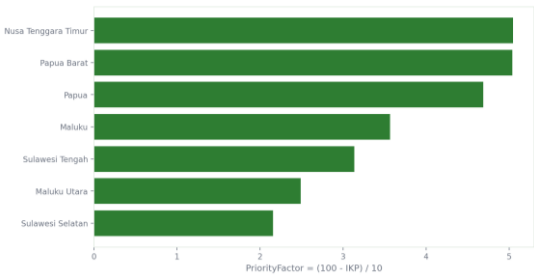
Branch and Bound menghasilkan total cost yang identik dengan Brute Force (1128.229), yang membuktikan bahwa solusi yang diperoleh adalah optimal secara global. Greedy Nearest Neighbor menghasilkan cost 1269.023, yaitu 12.5% lebih buruk dibanding solusi optimal. Hal ini menunjukkan bahwa pendekatan *greedy* dapat terjebak pada optimum lokal karena keputusan pemilihan node berikutnya hanya mempertimbangkan kondisi saat ini tanpa memproyeksikan dampak terhadap sisa rute.



Gambar. 9. Perbandingan Metode

E. Pengaruh Integrasi IKP terhadap Prioritas Distribusi

Untuk mengilustrasikan pengaruh integrasi IKP, dilakukan analisis perbandingan antara rute yang dihasilkan model (dengan IKP) dan rute yang akan dihasilkan apabila distribusi semata-mata didasarkan pada jarak Haversine terpendek. Rute berbasis jarak cenderung mendahulukan Sulawesi Tengah (474.87 km) dibanding Nusa Tenggara Timur (1851.97 km). Namun, dengan integrasi IKP, Nusa Tenggara Timur (IKP = 49.53) didahulukan karena kerawanan pangannya yang lebih tinggi.



Gambar. 10. Prioritas Node berdasarkan IKP

Perbedaan ini memiliki implikasi praktis yang penting: sistem distribusi yang hanya mengoptimalkan jarak berisiko menunda pengiriman ke wilayah yang justru paling membutuhkan intervensi pangan segera.

V. KESIMPULAN

Penelitian ini telah mengembangkan pendekatan optimasi distribusi pangan berbasis algoritma Branch and Bound dengan *reduced cost matrix* yang mengintegrasikan Indeks Ketahanan Pangan, populasi, dan Indeks Kemahalan Konstruksi ke dalam model biaya distribusi. Beberapa kesimpulan utama yang diperoleh adalah sebagai berikut.

Pertama, algoritma Branch and Bound dengan strategi *best-first search* mampu menghasilkan solusi optimal yang identik dengan Brute Force, dengan efisiensi pencarian yang jauh lebih tinggi. Dari 5040 kemungkinan rute, hanya 3.37% ruang pencarian yang perlu dieksplorasi berkat mekanisme pruning berbasis *reduced cost matrix*.

Kedua, integrasi IKP dalam fungsi biaya menghasilkan urutan distribusi yang berbeda secara substantif dibanding optimasi berbasis jarak semata. Wilayah berkerawanan pangan tinggi, seperti Nusa Tenggara Timur (IKP = 49.53), memperoleh prioritas kunjungan yang lebih awal meskipun jaraknya lebih jauh dari depot.

Ketiga, metode Greedy Nearest Neighbor menghasilkan cost 12.5% lebih tinggi dibanding solusi optimal, yang menunjukkan bahwa pendekatan greedy tidak menjamin optimalitas pada permasalahan distribusi multifaktor ini.

A. Pengembangan Lanjutan

Beberapa arah pengembangan diidentifikasi berdasarkan keterbatasan implementasi saat ini.

Pertama, granularitas node dapat ditingkatkan dari tingkat provinsi ke tingkat kabupaten/kota agar rute distribusi lebih representatif terhadap kebutuhan riil di lapangan. Peningkatan ini sekaligus memungkinkan integrasi data IKP yang lebih granular yang telah tersedia pada level kabupaten dalam publikasi FSVA Bapanas.

Kedua, matriks jarak dapat diganti dari pendekatan Haversine ke data jarak jalan aktual menggunakan layanan seperti OSRM atau OpenRouteService. Jarak jalan aktual akan menghasilkan model biaya yang lebih realistis, terutama untuk wilayah yang memiliki ketergantungan tinggi terhadap moda transportasi darat.

Ketiga, model dapat dikembangkan menjadi *Capacitated Vehicle Routing Problem* (CVRP) dengan mempertimbangkan kapasitas kendaraan distribusi dan volume permintaan pangan per wilayah. Ekstensi ini akan menjadikan model lebih aplikatif untuk perencanaan logistik operasional.

Keempat, skenario multi-depot dapat dieksplorasi dengan menambahkan Food Estate lain sebagai depot tambahan, sehingga koordinasi distribusi dari beberapa pusat produksi secara simultan dapat dioptimalkan.

Kelima, pendekatan Branch and Bound dapat dikombinasikan dengan metode metaheuristik seperti

Simulated Annealing atau algoritma genetika untuk menangani skalabilitas pada skenario dengan jumlah node yang jauh lebih besar, di mana Branch and Bound exact mulai kehilangan keunggulan komputasinya.

VI. APENDIKS

Video Penjelasan: <https://youtu.be/PqniQL9D4AI>

Sumber Kode: <https://github.com/Izhr/foodestate-bnb-router>

Implementasi: <https://foodestate-bnb-router.streamlit.app/>

ACKNOWLEDGMENT

Penulis menyampaikan rasa terima kasih yang setulus-tulusnya kepada Prof. Dr. Ir. Rinaldi Munir, M.T. atas bimbingan, arahan, serta dedikasi beliau dalam mengampu mata kuliah IF2211 Strategi Algoritma. Ucapan terima kasih juga penulis sampaikan kepada keluarga dan sahabat yang senantiasa memberikan dukungan, motivasi, serta semangat selama proses penyusunan makalah ini. Selain itu, penulis memanjatkan puji dan syukur ke hadirat Tuhan Yang Maha Esa atas limpahan rahmat, karunia, dan penyertaan-Nya sehingga penulisan makalah ini dapat diselesaikan dengan baik.

REFERENCES

- [1] Badan Pangan Nasional (Bapanas) dan World Food Programme (WFP), "Peta Ketahanan dan Kerentanan Pangan Indonesia (FSVA) 2023," Bapanas, Jakarta, 2024. [Online]. Tersedia: <https://www.badanpangan.go.id>. Diakses: Juni 2026.
- [2] A. V. Aho, J. E. Hopcroft, dan J. D. Ullman, *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley, 1974.
- [3] R. Munir, "Algoritma Branch and Bound," Bahan Kuliah IF2211 Strategi Algoritma, Program Studi Teknik Informatika, STEI ITB, 2026. [Online]. Tersedia: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2025-2026/stima25-26.htm>. Diakses: Juni 2026.
- [4] J. D. C. Little, K. G. Murty, D. W. Sweeney, dan C. Karel, "An Algorithm for the Travelling Salesman Problem," *Operations Research*, vol. 11, no. 6, hlm. 972–989, 1963. doi: 10.1287/opre.11.6.972.
- [5] G. Gutin dan A. P. Punnen (Ed.), *The Travelling Salesman Problem and Its Variations*. Dordrecht: Kluwer Academic Publishers, 2002.
- [6] Badan Pusat Statistik (BPS), "Indeks Kemahalan Konstruksi 2024," BPS, Jakarta, 2024. [Online]. Tersedia: <https://www.bps.go.id>. Diakses: Juni 2026.
- [7] Badan Pusat Statistik (BPS), "Proyeksi Penduduk Indonesia 2020–2030," BPS, Jakarta, 2023. [Online]. Tersedia: <https://www.bps.go.id>. Diakses: Juni 2026.

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, 19 Juni 2026

Izhar Alif Akbar
18223129