

Penerapan Heuristik Lin-Kernighan-Helsgaun untuk Pencarian Rute Minimum Antar Kabupaten/Kota di Indonesia

Yusuf Faishal Listyardi - 13524014

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10, Bandung

Email: yusuffaishalals@gmail.com, 13524014@std.stei.itb.ac.id

Abstract—Travelling Salesman Problem (TSP) merupakan salah satu persoalan optimasi kombinatorial klasik yang bertujuan mencari siklus Hamilton berbobot minimum, yaitu rute terpendek yang mengunjungi setiap simpul tepat satu kali dan kembali ke simpul awal. Pada data geografis, simpul dapat dimodelkan sebagai kota atau kabupaten, sedangkan bobot sisi menyatakan jarak antarwilayah. Makalah ini membahas penerapan algoritma Lin-Kernighan-Helsgaun (LKH) untuk mencari rute TSP pada data kabupaten/kota di Indonesia berdasarkan koordinat lintang dan bujur. Jarak antar simpul dihitung menggunakan rumus Haversine agar sesuai dengan karakteristik koordinat permukaan bumi. Program yang dikembangkan membaca data kota, membentuk matriks jarak, menjalankan LKH, lalu menampilkan hasil rute akhir dalam bentuk visualisasi simpul dan sisi. Fokus makalah adalah mengamati kualitas rute terbaik yang ditemukan, waktu eksekusi, serta pengaruh jumlah simpul terhadap proses optimasi. Hasil yang diharapkan adalah sistem eksperimen dan visualisasi yang mampu menunjukkan penerapan heuristik LKH pada permasalahan rute nyata secara informatif.

Index Terms—Travelling Salesman Problem, Lin-Kernighan-Helsgaun, heuristik, Haversine, visualisasi graf, rute kota.

I. PENDAHULUAN

Pencarian rute minimum merupakan persoalan yang sering muncul dalam berbagai bidang, seperti distribusi barang, perencanaan perjalanan, logistik, dan analisis jaringan transportasi. Salah satu formulasi paling terkenal dari persoalan tersebut adalah Travelling Salesman Problem (TSP). Pada TSP, seorang penjual diasumsikan harus mengunjungi sejumlah kota tepat satu kali dan kembali ke kota awal dengan total jarak sekecil mungkin. Meskipun definisinya sederhana, TSP termasuk persoalan yang sulit diselesaikan secara eksak ketika jumlah simpul bertambah besar.

Secara graf, TSP dapat dipandang sebagai pencarian siklus Hamilton dengan bobot minimum pada graf lengkap berbobot. Setiap kota direpresentasikan sebagai simpul, sedangkan jarak antarkota direpresentasikan sebagai bobot sisi. Jika terdapat n kota, jumlah kemungkinan rute yang perlu diperiksa secara brute force bertumbuh secara faktorial. Pertumbuhan ini membuat pendekatan eksak menjadi tidak praktis untuk banyak kasus nyata, terutama ketika jumlah kota mencapai puluhan, ratusan, atau lebih.

Salah satu pendekatan eksak yang lebih baik daripada brute force adalah dynamic programming dengan bitmask, misalnya algoritma Held-Karp. Pendekatan ini dapat mengurangi kompleksitas dari faktorial menjadi sekitar $O(n^2 2^n)$, tetapi tetap membutuhkan waktu dan memori yang bertumbuh eksponensial terhadap jumlah kota [1]. Representasi bitmask juga membuat kebutuhan penyimpanan status meningkat sangat cepat karena setiap subset kota harus dipertimbangkan. Akibatnya, pendekatan ini masih sulit digunakan untuk dataset kabupaten/kota nyata yang dapat berisi puluhan hingga ratusan simpul.

Karena keterbatasan tersebut, pendekatan heuristik banyak digunakan untuk memperoleh solusi berkualitas baik dalam waktu yang lebih singkat. Salah satu heuristik yang dikenal sangat kuat untuk TSP adalah Lin-Kernighan-Helsgaun (LKH), yaitu pengembangan dari heuristik Lin-Kernighan. LKH melakukan pencarian lokal berbasis pertukaran sisi secara adaptif dan menggunakan strategi pemilihan kandidat sisi agar proses pencarian tetap efisien [2], [3]. Dalam banyak eksperimen TSP, LKH dikenal mampu menghasilkan rute yang sangat baik, bahkan sering mendekati solusi optimal pada berbagai instans benchmark.

Makalah ini berfokus pada penerapan LKH untuk menyelesaikan TSP pada data kabupaten/kota Indonesia. Dataset yang digunakan berisi nama provinsi, nama kabupaten/kota, serta koordinat lintang dan bujur dalam sistem koordinat WGS84. Dengan data tersebut, setiap wilayah dapat dipetakan sebagai titik geografis. Bobot antartitik dihitung menggunakan jarak great-circle melalui rumus Haversine, sehingga jarak yang diperoleh lebih sesuai dibandingkan jarak Euclidean biasa pada bidang datar.

Kontribusi utama makalah ini adalah penerapan LKH pada dataset kota nyata Indonesia dan penyajian hasil akhirnya dalam bentuk visualisasi graf. Program yang digunakan tidak hanya menjalankan algoritma, tetapi juga membantu analisis dengan menampilkan simpul kota, sisi rute, total jarak, dan waktu eksekusi. Visualisasi ini diharapkan memudahkan pembaca memahami bentuk rute yang dihasilkan oleh algoritma.

Ruang lingkup makalah dibatasi pada TSP simetris, yaitu jarak dari kota A ke kota B dianggap sama dengan jarak

dari kota B ke kota A . Makalah ini tidak mempertimbangkan kondisi jalan sebenarnya, kemacetan, moda transportasi, batas administratif, atau kendala waktu. Dengan demikian, rute yang dihasilkan merepresentasikan solusi berdasarkan jarak geografis antarkoordinat, bukan rute transportasi aktual di jalan raya.

II. LANDASAN TEORI

A. Algoritma Greedy dan Heuristik

Algoritma greedy merupakan salah satu strategi algoritma yang banyak digunakan untuk menyelesaikan persoalan optimasi. Persoalan optimasi dapat berupa maksimasi, misalnya memaksimalkan keuntungan, atau minimasi, misalnya meminimumkan biaya, waktu, atau jarak. Prinsip utama greedy adalah membangun solusi secara bertahap dengan memilih keputusan terbaik yang tersedia pada saat itu [4].

Pada setiap langkah, algoritma greedy memilih optimum lokal dengan harapan bahwa rangkaian keputusan lokal tersebut akan mengarah pada solusi optimum global. Secara umum, algoritma greedy memiliki beberapa komponen penting, yaitu himpunan kandidat, himpunan solusi, fungsi solusi, fungsi seleksi, fungsi kelayakan, dan fungsi objektif. Pada persoalan rute seperti TSP, pendekatan greedy sederhana dapat berupa pemilihan kota terdekat yang belum dikunjungi. Strategi tersebut cepat, tetapi rentan menghasilkan rute suboptimal karena tidak mempertimbangkan akibat pemilihan kota terhadap struktur rute secara keseluruhan.

Heuristik adalah pendekatan penyelesaian masalah yang dirancang untuk menemukan solusi baik dalam waktu yang relatif efisien, meskipun tidak selalu disertai jaminan optimalitas global. Dalam konteks TSP, heuristik menjadi penting karena pencarian solusi eksak membutuhkan waktu yang sangat besar ketika jumlah kota meningkat. LKH dapat ditempatkan sebagai heuristik pencarian lokal yang menggunakan keputusan-keputusan lokal untuk memperbaiki tur secara bertahap. Berbeda dari greedy konstruktif murni yang membangun rute dari awal dengan satu aturan seleksi sederhana, LKH memperbaiki rute yang sudah ada melalui pertukaran sisi adaptif sehingga mampu mengeksplorasi ruang solusi dengan lebih kuat.

B. Travelling Salesman Problem

Travelling Salesman Problem adalah persoalan untuk mencari tur terpendek dari sejumlah kota dengan jarak antarkota yang sudah diketahui. Seorang pedagang berangkat dari sebuah kota asal, mengunjungi setiap kota tepat satu kali, lalu kembali lagi ke kota asal. Dalam teori graf, persoalan ini setara dengan mencari sirkuit Hamilton berbobot minimum [5].

Lintasan Hamilton adalah lintasan yang melalui setiap simpul tepat satu kali, sedangkan sirkuit Hamilton adalah sirkuit yang melalui setiap simpul tepat satu kali kecuali simpul asal yang juga menjadi simpul akhir. Pada TSP, kota direpresentasikan sebagai simpul dan jarak antarkota direpresentasikan sebagai bobot sisi. Karena rute harus kembali ke kota awal, solusi TSP berbentuk sirkuit Hamilton.

Jika V adalah himpunan kota dan π adalah urutan kunjungan kota, maka tujuan TSP adalah meminimalkan

$$D(\pi) = d(\pi_1, \pi_2) + d(\pi_2, \pi_3) + \dots + d(\pi_n, \pi_1). \quad (1)$$

Suku terakhir $d(\pi_n, \pi_1)$ menunjukkan bahwa rute harus kembali ke kota awal.

Pada graf lengkap dengan n simpul dan $n \geq 3$, jumlah sirkuit Hamilton berbeda adalah $(n-1)!/2$ [5]. Jumlah ini tumbuh sangat cepat. Sebagai contoh, untuk $n = 20$, terdapat $19!/2$ kemungkinan sirkuit Hamilton. Pertumbuhan faktorial tersebut menjelaskan mengapa pemeriksaan seluruh kemungkinan rute tidak praktis untuk dataset yang besar.

TSP dikenal sebagai persoalan NP-hard [6]. Hal ini berarti belum diketahui algoritma yang dapat menyelesaikan seluruh instans TSP secara optimal dalam waktu polinomial. Untuk jumlah kota yang kecil, solusi eksak masih dapat dicari dengan brute force, dynamic programming, atau branch and bound. Namun, untuk jumlah kota yang lebih besar, metode heuristik sering lebih praktis karena mampu menghasilkan solusi yang baik dalam waktu yang lebih masuk akal.

C. Jarak Haversine

Dataset yang digunakan berisi koordinat lintang dan bujur. Karena bumi berbentuk mendekati bola, jarak antarkoordinat lebih sesuai dihitung menggunakan jarak great-circle daripada jarak Euclidean pada bidang datar. Salah satu rumus yang umum digunakan adalah Haversine.

Misalkan dua titik memiliki koordinat (ϕ_1, λ_1) dan (ϕ_2, λ_2) , dengan ϕ menyatakan lintang dan λ menyatakan bujur dalam radian. Jarak Haversine dihitung sebagai

$$\Delta\phi = \phi_2 - \phi_1, \quad (2)$$

$$\Delta\lambda = \lambda_2 - \lambda_1, \quad (3)$$

$$a = \sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\Delta\lambda}{2}\right), \quad (4)$$

$$c = 2 \operatorname{atan2}(\sqrt{a}, \sqrt{1-a}), \quad (5)$$

$$d = Rc. \quad (6)$$

Dalam makalah ini, nilai R dapat diambil sebesar 6371 km. Hasil jarak kemudian dapat dibulatkan ke satuan kilometer atau dikalikan dengan faktor tertentu agar sesuai dengan format masukan solver.

D. Heuristik Lin-Kernighan

Heuristik Lin-Kernighan merupakan metode pencarian lokal untuk TSP yang dikembangkan dari konsep k -opt [2]. Pada pendekatan k -opt, sejumlah k sisi pada tur dihapus lalu diganti dengan sisi lain sehingga terbentuk tur baru. Jika tur baru memiliki total jarak lebih kecil, perubahan tersebut diterima.

Perbedaan utama Lin-Kernighan dibandingkan 2-opt atau 3-opt biasa adalah nilai k tidak ditentukan secara tetap sejak awal. Algoritma dapat menyesuaikan jumlah sisi yang ditukar selama proses pencarian. Karena itu, Lin-Kernighan sering disebut sebagai variable-depth search. Strategi ini membuat

pencarian lebih fleksibel karena algoritma dapat mengeksplorasi perubahan rute yang lebih kompleks daripada pertukaran dua sisi saja.

Secara umum, proses Lin-Kernighan dapat dijelaskan sebagai berikut: mulai dari sebuah tur awal, memilih sisi-sisi yang akan dihapus dari tur, memilih sisi-sisi baru yang akan ditambahkan, menghitung keuntungan dari pertukaran tersebut, menyimpan tur baru jika pertukaran memperbaiki tur, lalu mengulangi proses sampai tidak ditemukan perbaikan yang memenuhi kriteria.

E. Lin-Kernighan-Helsgaun

Lin-Kernighan-Helsgaun adalah implementasi dan pengembangan lebih lanjut dari heuristik Lin-Kernighan [3]. LKH memperbaiki efektivitas pencarian dengan menggunakan beberapa strategi, antara lain pembentukan candidate set, penggunaan ukuran alpha-nearness, dan mekanisme pencarian lokal yang dioptimalkan. Candidate set membatasi sisi-sisi yang dipertimbangkan oleh algoritma sehingga pencarian tidak perlu memeriksa seluruh kemungkinan sisi pada graf lengkap.

Gagasan penting dari LKH adalah bahwa tidak semua sisi memiliki peluang yang sama untuk muncul pada tur berkualitas baik. Dengan memilih kandidat sisi yang lebih menjanjikan, algoritma dapat mengurangi ruang pencarian tanpa mengorbankan kualitas solusi secara signifikan.

Dalam makalah ini, LKH digunakan sebagai algoritma utama untuk mencari rute terbaik yang ditemukan pada dataset kabupaten/kota. Karena LKH adalah heuristik, istilah “rute terbaik” pada makalah ini merujuk pada rute terbaik yang dihasilkan oleh proses pencarian LKH, bukan klaim bahwa rute tersebut pasti optimal secara matematis.

F. Format TSPLIB

Banyak solver TSP, termasuk LKH, menggunakan format berkas yang mengikuti gaya TSPLIB [7]. TSPLIB adalah kumpulan instans benchmark dan format representasi masalah TSP yang umum digunakan dalam makalah optimasi kombinatorial. Dalam format ini, sebuah instans TSP dapat dinyatakan melalui daftar koordinat simpul atau matriks bobot sisi.

Pada makalah ini, program dapat menghasilkan berkas masukan berformat TSPLIB dari dataset kota. Setiap kota diberi indeks numerik, sedangkan informasi nama kota dan provinsi disimpan pada struktur data program agar hasil rute dapat diterjemahkan kembali dari indeks menjadi nama wilayah.

III. METODOLOGI

A. Gambaran Umum Sistem

makalah ini menggunakan program eksperimen dan visualisasi untuk menjalankan LKH pada data kabupaten/kota Indonesia. Program berfungsi sebagai penghubung antara dataset geografis, solver LKH, dan antarmuka visualisasi. Secara umum, sistem terdiri atas lima tahap utama: pemuatan data, pemilihan simpul, perhitungan jarak, eksekusi LKH, dan visualisasi hasil rute.

Alur umum sistem adalah sebagai berikut.

- 1) Program membaca dataset kabupaten-kota.csv.

- 2) Pengguna memilih subset kota berdasarkan provinsi, pulau, jumlah kota, atau pemilihan manual.
- 3) Program menghitung jarak Haversine untuk setiap pasangan kota.
- 4) Program membentuk berkas masukan TSP dan berkas parameter LKH.
- 5) LKH dijalankan untuk mencari rute terbaik.
- 6) Program membaca hasil tur dari LKH.
- 7) Program menghitung ulang total jarak rute.
- 8) GUI menampilkan simpul kota, sisi rute, urutan kunjungan, total jarak, dan waktu eksekusi.

Fokus visualisasi berada pada hasil akhir rute, bukan animasi setiap langkah pencarian. Keputusan ini diambil karena proses internal LKH cukup kompleks dan tidak selalu mudah direpresentasikan sebagai perubahan sederhana dari satu rute ke rute berikutnya. Dengan menampilkan hasil akhir, GUI tetap dapat memberikan informasi utama yang dibutuhkan untuk analisis makalah.

B. Dataset

Dataset yang digunakan adalah kabupaten-kota.csv, yang berisi data wilayah kabupaten/kota di Indonesia. Berdasarkan pembacaan awal, dataset terdiri atas 514 baris data. Kolom yang digunakan dalam makalah ini ditunjukkan pada Tabel I.

TABLE I
KOLOM DATASET YANG DIGUNAKAN

Kolom	Keterangan
KODE_PROV	Kode provinsi
KODE_KK	Kode kabupaten/kota
PROVINSI	Nama provinsi
KAB_KOTA	Nama kabupaten/kota
LATITUDE	Koordinat lintang
LONGITUDE	Koordinat bujur

Data koordinat menggunakan sistem WGS84. Oleh karena itu, perhitungan jarak dilakukan dengan pendekatan great-circle distance. Untuk menjaga konsistensi eksperimen, baris yang tidak memiliki nilai lintang atau bujur valid perlu dihapus atau diperbaiki sebelum digunakan.

Selain pemilihan berdasarkan jumlah simpul, eksperimen juga menggunakan skenario berdasarkan cakupan pulau. Dataset asli memiliki kolom `PROVINSI`, tetapi tidak memiliki kolom `PULAU`. Oleh karena itu, program menambahkan pemetaan sederhana dari provinsi ke pulau.

Beberapa skenario dataset yang direncanakan ditunjukkan pada Tabel II.

Skenario tersebut digunakan untuk mengamati pengaruh jumlah simpul dan cakupan geografis terhadap waktu eksekusi serta panjang rute yang dihasilkan. Pemilihan berdasarkan pulau juga memungkinkan analisis visual yang lebih jelas karena rute pada satu pulau cenderung lebih mudah diamati dibandingkan rute nasional yang tersebar luas.

C. Prapemrosesan Data

Prapemrosesan dilakukan untuk memastikan data dapat digunakan sebagai masukan TSP. Tahap prapemrosesan meliputi

TABLE II
SKENARIO PEMILIHAN DATASET

Skenario	Cara Pemilihan Data	Jumlah Simpul
Satu provinsi	Kabupaten/kota pada satu provinsi, misalnya Jawa Timur	10–40
Pulau Jawa	Seluruh kabupaten/kota pada provinsi-provinsi di Pulau Jawa	data valid
Pulau Sumatra	Seluruh kabupaten/kota pada provinsi-provinsi di Pulau Sumatra	data valid
Pulau Kalimantan	Seluruh kabupaten/kota pada provinsi-provinsi di Pulau Kalimantan	data valid
Pulau Sulawesi	Seluruh kabupaten/kota pada provinsi-provinsi di Pulau Sulawesi	data valid
Papua	Seluruh kabupaten/kota pada provinsi-provinsi di wilayah Papua	data valid
Sampel nasional	Sebagian kabupaten/kota dari berbagai provinsi Indonesia	100–300
Penuh	Seluruh data kabupaten/kota valid	hingga 514

pembacaan berkas CSV, pengambilan kolom nama provinsi, nama kabupaten/kota, lintang, dan bujur, konversi nilai lintang dan bujur menjadi tipe numerik, penghapusan data dengan koordinat kosong atau tidak valid, penambahan atribut pulau berdasarkan nama provinsi, pemilihan subset data, pemberian indeks unik untuk setiap simpul, dan penyimpanan pemetaan indeks ke nama wilayah.

Indeks numerik diperlukan karena solver TSP umumnya menggunakan nomor simpul, bukan nama kota. Misalnya, simpul 1 dapat merujuk pada Kota Surabaya, simpul 2 pada Kabupaten Sidoarjo, dan seterusnya. Setelah LKH menghasilkan urutan simpul, program menerjemahkan kembali indeks tersebut menjadi nama kabupaten/kota agar hasilnya mudah dibaca.

D. Pembentukan Matriks Jarak

Setelah subset kota dipilih, program menghitung jarak antara setiap pasangan kota. Untuk n kota, akan terbentuk matriks jarak berukuran $n \times n$. Nilai diagonal matriks bernilai nol karena jarak dari sebuah kota ke dirinya sendiri adalah nol. Pseudocode pembentukan matriks jarak ditunjukkan pada Algoritma 1.

Karena TSP yang digunakan bersifat simetris, nilai $D[i][j]$ sama dengan $D[j][i]$. Implementasi dapat memanfaatkan sifat ini untuk menghindari perhitungan ganda, tetapi secara konseptual matriks tetap dianggap lengkap.

E. Representasi Masukan untuk LKH

Setelah matriks jarak dibentuk, program menghasilkan berkas instans TSP. Terdapat dua pendekatan representasi yang dapat digunakan, yaitu koordinat geografis dan matriks bobot eksplisit. Pada makalah ini, representasi matriks bobot eksplisit lebih disarankan karena jarak dihitung menggunakan rumus Haversine. Dengan demikian, bobot yang digunakan oleh LKH sama dengan bobot yang digunakan program saat menghitung ulang total jarak.

Fig. 1. Algoritma 1 Pembentukan Matriks Jarak Haversine

```

Require: Daftar kota C berisi nama, latitude,
        longitude
Ensure: Matriks jarak D

1: n ← jumlah kota dalam C
2: buat matriks D berukuran n x n
3: for i ← 1 to n do
4:   for j ← 1 to n do
5:     if i = j then
6:       D[i][j] ← 0
7:     else
8:       D[i][j] ← Haversine(C[i], C[j])
9:     end if
10:  end for
11: end for
12: return D

```

Contoh struktur berkas TSP yang dihasilkan adalah sebagai berikut.

```

NAME: indonesia_city_tsp
TYPE: TSP
DIMENSION: n
EDGE_WEIGHT_TYPE: EXPLICIT
EDGE_WEIGHT_FORMAT: FULL_MATRIX
EDGE_WEIGHT_SECTION
...
EOF

```

Selain berkas `.tsp`, program juga membuat berkas parameter `.par` untuk LKH. Berkas ini berisi lokasi berkas masalah, lokasi keluaran tur, jumlah run, seed, dan parameter lain yang diperlukan.

```

PROBLEM_FILE = instance.tsp
OUTPUT_TOUR_FILE = result.tour
RUNS = 10
SEED = 1
TRACE_LEVEL = 1

```

Nilai RUNS dapat dinaikkan untuk memberi kesempatan LKH mencoba beberapa pencarian dan memilih hasil terbaik. Namun, semakin besar nilai RUNS, semakin besar pula waktu eksekusi.

F. Eksekusi Algoritma LKH

Eksekusi LKH dilakukan dengan memanggil program solver dari backend. Backend mengirimkan berkas parameter sebagai masukan, kemudian menunggu sampai LKH selesai menghasilkan berkas tur. Waktu eksekusi diukur sejak proses LKH dimulai sampai proses selesai. Pseudocode tahap eksekusi ditunjukkan pada Algoritma 2.

Perhitungan total jarak dilakukan kembali oleh program menggunakan matriks jarak yang sama. Langkah ini penting agar nilai jarak yang dilaporkan pada GUI dan makalah konsisten dengan data yang digunakan.

G. Visualisasi Hasil Rute

Visualisasi bertujuan menampilkan bentuk rute akhir yang ditemukan oleh LKH. Setiap kota digambar sebagai simpul berdasarkan posisi relatif dari koordinat lintang dan bujur. Setiap pasangan kota yang berurutan dalam tur digambar sebagai

Fig. 2. Algoritma 2 Eksekusi LKH dan Pembacaan Hasil

```

Require: Dataset kota C, parameter eksperimen P
Ensure: Rute terbaik T, total jarak L, waktu
       eksekusi t

1: C_selected <- pilih subset kota dari C
2: D <- bentuk matriks jarak Haversine(C_selected)
3: tulis berkas instance.tsp dari D
4: tulis berkas parameter instance.par dari P
5: start_time <- waktu saat ini
6: jalankan LKH dengan instance.par
7: end_time <- waktu saat ini
8: T <- baca urutan simpul dari result.tour
9: L <- hitungTotalJarak(T, D)
10: t <- end_time - start_time
11: return T, L, t

```

sisi. Sisi terakhir menghubungkan kota terakhir kembali ke kota awal.

Informasi yang ditampilkan pada GUI meliputi titik kabupaten/kota sebagai simpul, garis rute sebagai sisi hasil LKH, nama kota atau tooltip saat simpul dipilih, urutan kunjungan kota, total jarak rute dalam kilometer, jumlah simpul dan jumlah sisi, waktu eksekusi, serta parameter eksperimen seperti jumlah run dan seed. Untuk menjaga keterbacaan, visualisasi pada dataset besar dapat menyediakan fitur zoom, pan, atau filter label.

H. Desain Eksperimen

Eksperimen dilakukan untuk mengamati performa LKH pada beberapa ukuran dataset. Karena makalah ini berfokus pada LKH, pembahasan tidak diarahkan pada perbandingan banyak algoritma, melainkan pada perubahan hasil LKH ketika jumlah simpul dan cakupan wilayah berubah.

Variabel yang diamati adalah jumlah simpul, cakupan wilayah dataset, total jarak rute terbaik yang ditemukan, waktu eksekusi, bentuk visual rute, dan pengaruh parameter RUNS terhadap hasil dan waktu. Konfigurasi skenario eksperimen yang digunakan ditunjukkan pada Tabel III.

TABLE III
KONFIGURASI SKENARIO EKSPERIMEN

Dataset	Simpul	RUNS	Seed
Jawa Timur 10	10	5	7
Jawa Timur	38	5	7
Pulau Jawa	119	5	7
Pulau Sumatra	154	5	7
Pulau Kalimantan	56	5	7
Pulau Sulawesi	81	5	7
Papua	42	5	7
Indonesia 100	100	5	7
Indonesia penuh	514	5	7

Eksperimen dapat dijalankan beberapa kali dengan seed berbeda untuk melihat kestabilan hasil. Jika hasil berbeda, makalah dapat melaporkan rute terbaik, rata-rata jarak, dan rata-rata waktu eksekusi. Jika keterbatasan waktu hanya memungkinkan satu konfigurasi seed, makalah perlu menyebutkan konfigurasi tersebut secara eksplisit.

I. Batasan Metodologi

Metodologi ini memiliki beberapa batasan. Pertama, jarak Haversine hanya menghitung jarak langsung di permukaan bumi, bukan jarak jalan sebenarnya. Kedua, TSP yang digunakan bersifat simetris dan tidak mempertimbangkan arah jalan, kondisi lalu lintas, waktu tempuh, biaya transportasi, atau keberadaan laut antarwilayah. Ketiga, LKH adalah algoritma heuristik, sehingga hasil yang diperoleh tidak selalu dapat diklaim sebagai solusi optimal global.

IV. EKSPERIMEN DAN HASIL

A. Lingkungan Eksperimen

Seluruh skenario pada eksperimen ini dijalankan dengan parameter RUNS = 5 dan Seed = 7. Parameter RUNS menyatakan jumlah percobaan pencarian yang dilakukan LKH dalam satu eksekusi, sedangkan Seed digunakan untuk menjaga agar pola randomisasi dapat direproduksi. Waktu eksekusi diukur dari sisi program sejak proses LKH dimulai hingga proses selesai dan berkas tur dapat dibaca kembali.

B. Hasil Eksperimen

Tabel IV menunjukkan hasil eksperimen pada beberapa cakupan data. Dataset kecil menggunakan 10 kabupaten/kota di Jawa Timur. Dataset menengah menggunakan seluruh kabupaten/kota pada satu pulau, sedangkan dataset besar menggunakan sampel nasional dan seluruh dataset valid.

Pada skenario Jawa Timur 10, LKH menghasilkan rute sepanjang 789.293 km. Ketika seluruh 38 kabupaten/kota Jawa Timur digunakan, total jarak meningkat menjadi 1343.346 km. Kenaikan ini wajar karena jumlah simpul bertambah dan rute harus mengunjungi lebih banyak wilayah. Namun, waktu eksekusi tetap rendah, yaitu 40.68 ms pada konfigurasi eksperimen ini.

Pada skenario berdasarkan pulau, Pulau Jawa menghasilkan rute 3568.847 km untuk 119 simpul, sedangkan Pulau Sumatra menghasilkan rute 9496.918 km untuk 154 simpul. Perbedaan jarak tidak hanya dipengaruhi jumlah simpul, tetapi juga sebaran geografis. Sumatra memiliki bentang wilayah yang lebih panjang dari barat laut ke tenggara, sehingga rute yang menghubungkan seluruh kabupaten/kota valid menjadi jauh lebih panjang.

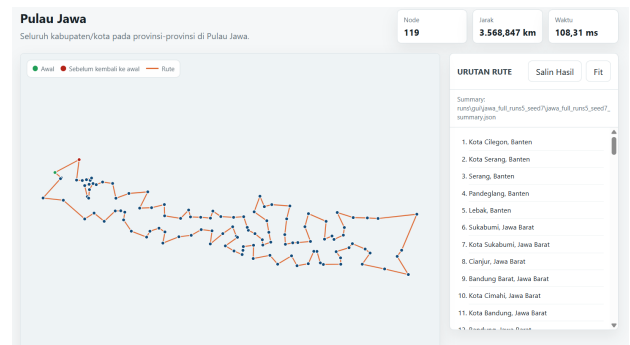


Fig. 3. Visualisasi rute akhir TSP pada skenario Pulau Jawa. Sumber: dokumen penulis.

TABLE IV
HASIL EKSPERIMEN LKH PADA DATASET KABUPATEN/KOTA INDONESIA

Dataset	Jumlah Simpul	RUNS	Total Jarak (km)	Waktu Eksekusi (ms)
Jawa Timur 10	10	5	789.293	14.13
Jawa Timur	38	5	1343.346	40.68
Pulau Jawa	119	5	3568.847	96.31
Pulau Sumatra	154	5	9496.918	280.32
Pulau Kalimantan	56	5	4469.330	60.83
Pulau Sulawesi	81	5	5205.377	63.58
Papua	42	5	4242.884	32.87
Indonesia 100	100	5	15597.068	56.55
Indonesia penuh	514	5	32353.247	7420.58

Pulau Kalimantan, Sulawesi, dan Papua menunjukkan karakteristik yang berbeda. Walaupun Papua hanya memiliki 42 simpul pada dataset valid, total jaraknya mencapai 4242.884 km. Hal ini menunjukkan bahwa jumlah simpul bukan satu-satunya faktor yang menentukan panjang rute; penyebaran koordinat juga sangat berpengaruh. Sulawesi memiliki 81 simpul dengan total jarak 5205.377 km, sedangkan Kalimantan memiliki 56 simpul dengan total jarak 4469.330 km.

Skenario nasional memperlihatkan perbedaan skala yang lebih jelas. Sampel nasional 100 simpul menghasilkan rute 15597.068 km, sedangkan seluruh dataset 514 simpul menghasilkan rute 32353.247 km. Waktu eksekusi dataset penuh mencapai 7420.58 ms, lebih besar dibandingkan skenario lainnya. Hal ini menunjukkan bahwa ketika jumlah simpul meningkat secara signifikan, ukuran matriks jarak dan ruang pencarian lokal yang perlu dipertimbangkan juga bertambah besar.

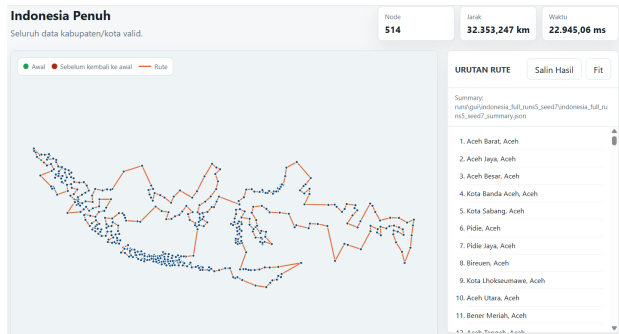


Fig. 4. Visualisasi rute akhir TSP pada skenario Nasional. Sumber: dokumen penulis.

V. PEMBAHASAN

Hasil eksperimen menunjukkan bahwa LKH dapat digunakan untuk menyelesaikan TSP pada data kabupaten/kota Indonesia dengan jumlah simpul yang bervariasi. Untuk dataset kecil dan menengah, waktu eksekusi berada pada orde puluhan hingga ratusan milidetik pada konfigurasi eksperimen ini. Untuk dataset penuh 514 simpul, waktu eksekusi meningkat menjadi beberapa detik, tetapi masih praktis untuk eksperimen interaktif.

Perbandingan antar skenario menunjukkan bahwa total jarak tidak meningkat hanya berdasarkan jumlah simpul. Cakupan geografis dan distribusi koordinat memiliki pengaruh besar. Sebagai contoh, Papua memiliki jumlah simpul lebih sedikit dibandingkan Kalimantan dan Sulawesi, tetapi total jaraknya tetap besar karena titik-titiknya tersebar pada wilayah yang luas. Sebaliknya, Jawa Timur memiliki 38 simpul tetapi total jaraknya jauh lebih kecil karena wilayahnya lebih terlokalisasi.

Skenario Indonesia penuh memberikan gambaran batas atas dari eksperimen ini. Dengan 514 simpul, pendekatan eksak seperti brute force atau dynamic programming bitmask tidak realistis digunakan. LKH memungkinkan pencarian rute berkualitas baik pada ukuran data tersebut dalam waktu yang masih dapat diterima. Meskipun demikian, hasil LKH tetap harus dipahami sebagai rute terbaik yang ditemukan oleh heuristik, bukan bukti solusi optimal global.

Keterbatasan utama eksperimen ini adalah penggunaan jarak Haversine. Jarak tersebut merepresentasikan jarak lurus permukaan bumi, bukan jarak jalan raya, waktu tempuh, atau biaya transportasi nyata. Akibatnya, rute yang dihasilkan lebih tepat dipandang sebagai model geografis TSP, bukan rekomendasi rute perjalanan aktual. Untuk pengembangan lebih lanjut, bobot sisi dapat diganti dengan jarak jalan dari layanan peta atau data jaringan transportasi.

VI. KESIMPULAN

Makalah ini membahas penerapan algoritma Lin-Kernighan-Helsgaun untuk menyelesaikan Travelling Salesman Problem pada data kabupaten/kota Indonesia. Data wilayah direpresentasikan sebagai simpul, sedangkan bobot antarsimpul dihitung menggunakan jarak Haversine dari koordinat lintang dan bujur. Program yang dikembangkan mampu membentuk masukan TSPLIB, menjalankan LKH, membaca hasil tur, menghitung total jarak, serta memvisualisasikan rute akhir melalui GUI.

Berdasarkan hasil eksperimen, LKH mampu menghasilkan rute untuk berbagai skenario, mulai dari 10 simpul hingga 514 simpul. Pada dataset penuh, rute terbaik yang ditemukan memiliki total jarak 32353.247 km dengan waktu eksekusi 7420.58 ms pada konfigurasi RUNS = 5 dan Seed = 7. Hasil ini menunjukkan bahwa LKH praktis digunakan untuk eksperimen TSP berbasis data kota nyata yang ukurannya tidak sesuai untuk pendekatan eksak seperti brute force atau dynamic programming bitmask.

Eksperimen juga menunjukkan bahwa jumlah simpul bukan satu-satunya faktor yang memengaruhi total jarak. Sebaran geografis wilayah memiliki pengaruh besar terhadap panjang rute. Oleh karena itu, skenario berdasarkan pulau memberikan gambaran yang lebih informatif dibandingkan hanya membandingkan jumlah simpul.

Pengembangan lanjutan dapat dilakukan dengan menambah variasi seed dan nilai RUNS, membandingkan hasil LKH dengan heuristik lain, serta menggunakan bobot berbasis jarak jalan agar model lebih mendekati kondisi transportasi nyata.

DATASET LINK (DRIVE)

<http://bit.ly/4oAD0r6>

REFERENCES

- [1] M. Held and R. M. Karp, "A dynamic programming approach to sequencing problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 10, no. 1, pp. 196–210, 1962.
- [2] S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Operations Research*, vol. 21, no. 2, pp. 498–516, 1973.
- [3] K. Helsgaun, "An effective implementation of the Lin-Kernighan traveling salesman heuristic," *European Journal of Operational Research*, vol. 126, no. 1, pp. 106–130, 2000.
- [4] R. Munir, "Algoritma Greedy (Bagian 1)," Bahan Kuliah IF2211 Strategi Algoritma, Program Studi Teknik Informatika, Institut Teknologi Bandung, 2026.
- [5] R. Munir, "Graf (Bagian 3)," Bahan Kuliah IF1220 Matematika Diskrit, Program Studi Teknik Informatika, Institut Teknologi Bandung, 2026.
- [6] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys, *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Chichester, United Kingdom: Wiley, 1985.
- [7] G. Reinelt, "TSPLIB: A traveling salesman problem library," *ORSA Journal on Computing*, vol. 3, no. 4, pp. 376–384, 1991.

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

A handwritten signature in black ink on a grid background. The signature is stylized, starting with a large 'Y' and ending with 'is'.

Yusuf Faishal Listyardi (13524014)