

Optimasi Rute Distribusi Menggunakan Metode Nearest Neighbor dan Algoritma Tabu Search di PT. XYZ

Distribution Route Optimization Using the Nearest Neighbor Method and the Tabu Search Algorithm at PT. XYZ

Syakri Lasau
Program Studi Teknik Industri
Universitas Negeri Gorontalo
Gorontalo, Indonesia
syakri07@gmail.com

Idham Halid Lahay
Program Studi Teknik Industri
Universitas Negeri Gorontalo
Gorontalo, Indonesia
idham-lahay@ung.ac.id

Rahmat Deddy Riyanto Dako
Program Studi Teknik Komputer
Universitas Negeri Gorontalo
Gorontalo, Indonesia
rahmatdeddy@ung.ac.id

Jamal Darusalam Giu
Program Studi Teknik Industri
Universitas Negeri Gorontalo
Gorontalo, Indonesia
jamaldarusalam@ung.ac.id

Amirudin Yunus Dako
Program Studi Teknik Komputer
Universitas Negeri Gorontalo
Gorontalo, Indonesia
amirudin.dako@ung.ac.id

Syahrir Abdussamad
Program Studi Teknik Komputer
Universitas Negeri Gorontalo
Gorontalo, Indonesia
syahirabdussamad@ung.ac.id

Diterima : Mei 2025
Disetujui : Desember 2025
Dipublikasi : Januari 2026

Abstrak— PT. Sinar Kristal Nusantara saat ini hanya mengandalkan pengalaman pengemudi dalam menentukan rute distribusi tanpa studi khusus, yang menyebabkan keterlambatan pengiriman produk es kristal karena arus distribusi yang tidak teratur. Penelitian ini bertujuan untuk menentukan rute distribusi yang optimal dengan jarak terpendek agar pengiriman lebih efisien dan tepat waktu. Metode yang digunakan adalah algoritma *Nearest Neighbor* dan *Tabu Search*. Proses dimulai dengan mengklasifikasi rute awal menggunakan Saving Matrix, kemudian menentukan rute optimal melalui *Nearest Neighbor* dan *Tabu Search*. Hasil penelitian menunjukkan bahwa metode ini menghasilkan total jarak tempuh 31,05 km dengan waktu tempuh 37,26 menit, lebih efisien dari rute saat ini yang mencapai 63,90 km dengan waktu tempuh 75 menit. *Tabu Search* terbukti menjadi metode optimasi yang lebih unggul karena menghasilkan rute yang lebih pendek dibandingkan dengan menggunakan *Nearest Neighbor* saja. Kesimpulannya, kombinasi *Nearest Neighbor* dan *Tabu Search* dapat meningkatkan efisiensi distribusi es kristal di PT. Sinar Kristal Nusantara.

Kata Kunci— : Optimization; *Nearest Neighbor*; *Tabu Search*; Vehicle routing problem

Abstract— PT. Sinar Kristal Nusantara currently only relies on driver experience in determining distribution routes without special studies, which causes delays in the delivery of crystal ice products due to irregular distribution flow. This study aims to determine the optimal distribution route with the shortest distance to make deliveries more efficient and timely. The methods used are *Nearest Neighbor* and *Tabu Search* algorithm. The process starts by classifying the initial route using Saving Matrix, then determining the optimal route through *Nearest Neighbor* and

Tabu Search. The results showed that this method resulted in a total mileage of 31.05 km with a travel time of 37.26 minutes, more efficient than the current route which reached 63.90 km with a time of 75 minutes. *Tabu Search* proved to be a superior optimization method because it produced a shorter route than using *Nearest Neighbor* alone. In conclusion, the combination of *Nearest Neighbor* and *Tabu Search* can improve the efficiency of crystal ice distribution at PT. Sinar Kristal Nusantara.

Keywords— Optimization; *Nearest Neighbor*; *Tabu Search*; Vehicle routing problem

I. PENDAHULUAN

PT. Xyz, produsen es kristal (*Ice Tube*) di Kota Gorontalo, menghadapi kendala dalam proses distribusi yang masih bergantung pada pengalaman pengemudi tanpa sistem penentuan rute yang terstruktur. Ketidakteraturan dalam alur distribusi ini berdampak pada waktu pengiriman yang tidak konsisten serta peningkatan biaya operasional. Seiring meningkatnya permintaan, dibutuhkan sistem penjadwalan rute yang lebih efisien dan dioptimasi untuk memastikan pengiriman tepat waktu dan hemat biaya.

Permasalahan penentuan rute dalam distribusi dikenal sebagai **Vehicle Routing Problem (VRP)**, yaitu masalah optimasi dalam menentukan rute terbaik bagi kendaraan untuk mengunjungi sejumlah pelanggan dengan jarak dan waktu tempuh minimum. Salah satu metode yang umum digunakan untuk menyelesaikan VRP adalah **Nearest Neighbor (NN)**, yang menyusun rute berdasarkan urutan kedekatan lokasi pelanggan secara bertahap. Namun, pendekatan ini cenderung menghasilkan solusi lokal yang

belum tentu optimal secara global karena bersifat greedy dan tidak mempertimbangkan kombinasi alternatif rute secara menyeluruh.

Sebagai solusi peningkatan, **Tabu Search (TS)** merupakan algoritma metaheuristik yang mampu menghindari jebakan *local optimum* dengan mengeksplorasi ruang solusi lebih luas menggunakan *tabu list* untuk melacak dan mencegah solusi berulang. TS bekerja dengan memperbaiki solusi awal dari NN secara iteratif hingga mencapai solusi yang lebih optimal.

Beberapa studi sebelumnya telah mengkaji penerapan metode optimasi dalam konteks distribusi dan peramalan spasial. Misalnya, Mogale et al. (2025) memodelkan dan mengoptimalkan permasalahan multi-depot VRP untuk distribusi barang di jaringan logistik ritel menggunakan pendekatan canggih berbasis komputasi industri. Sementara itu, Kuronen et al. (2025) menerapkan metode *k-nearest neighbor* dalam prediksi spasial dengan pendekatan *conformal prediction* untuk menangani ketidakpastian dalam pemetaan atribut hutan. Namun, studi yang secara khusus menggabungkan metode Nearest Neighbor dan Tabu Search dalam konteks distribusi es kristal, yang memiliki tantangan logistik tersendiri seperti kebutuhan pendinginan dan keterbatasan waktu distribusi, masih sangat jarang ditemukan.

Penelitian ini bertujuan untuk mengoptimalkan rute distribusi es kristal pada PT. Sinar Kristal Nusantara dengan menggabungkan metode Nearest Neighbor sebagai solusi awal dan Tabu Search sebagai metode peningkatan solusi. Melalui pendekatan ini, diharapkan dapat diperoleh rute distribusi yang lebih efisien dalam hal jarak dan waktu tempuh, sekaligus menekan biaya operasional.

Kontribusi utama dari penelitian ini terletak pada penerapan kombinasi metode NN dan TS dalam industri es kristal yang belum banyak dieksplorasi. Aspek kebaruan (*novelty*) dari penelitian ini terletak pada pendekatan hybrid algoritmik dalam konteks distribusi dengan karakteristik unik, yang dapat menjadi rujukan bagi industri serupa dalam meningkatkan efisiensi operasional berbasis teknologi optimasi.

II. METODE

Penelitian ini menggunakan pendekatan kuantitatif untuk mengoptimalkan jalur distribusi es kristal di PT. Xyz. Data yang dikumpulkan mencakup alamat pelanggan, koordinat geografis masing-masing pelanggan, serta jarak antar titik distribusi. Data koordinat diperoleh menggunakan perangkat lunak *Google Earth Pro*.

TABEL 1. KOORDINAT OUTLET PELANGGAN

Kode	Nama Outlet	Latitude	Longitude
1	Gudang	0,5791267	123,093361
2	Rm adinda	0,55377642	123,049005
3	Kedai adiya	0,54765375	123,052214
4	Rm ainun	0,55381611	123,050163
5	Rm Ampera	0,53656538	123,059184
6	Anggelato	0,53100126	123,058723
7	Rm anugerah	0,55872295	123,053227
8	Ayam geprek akbar	0,53566512	123,053531
9	Star chicken	0,54648875	123,061031
....
72	23 pool larde & cafe	0,557813	123,053459

A. Nearest Neighbor

Nearest Neighbor adalah metode yang pertama digunakan untuk mendapatkan solusi vehicle routing problem. Metode ini sangat mudah dan cepat untuk diimplementasikan. Dengan memilih satu titik sebagai titik awal, lalu bergerak ke titik selanjutnya yang terdekat [1]. *Nearest Neighbor* termasuk algoritma pembelajaran yang diawasi. Dalam prosesnya, algoritma ini mengklasifikasikan instansi baru berdasarkan mayoritas kategori dari tetangga terdekatnya. Dengan pendekatan ini, kelas yang paling sering muncul di antara tetangga menjadi kelas yang diatributkan pada hasil klasifikasi [2].

Nearest Neighbor adalah metode klasifikasi yang memanfaatkan data latih untuk mengategorikan objek berdasarkan kedekatannya objek lain [3]. Algoritma NN dimulai dengan memperkirakan jarak antara sampel baru yang tidak terklasifikasi dan sampel berlabel (pelatihan) untuk menemukan tetangga terdekatnya, kemudian mengklasifikasikan sampel baru tersebut ke dalam kelas berdasarkan mayoritas tetangga terdekatnya [4]. *Nearest Neighbor* termasuk algoritma pembelajaran yang diawasi. Dalam prosesnya, algoritma ini mengklasifikasikan instansi baru berdasarkan mayoritas kategori dari tetangga terdekatnya. Dengan pendekatan ini, kelas yang paling sering muncul di antara tetangga menjadi kelas yang diatributkan pada hasil klasifikasi [5]

Penentuan urutan rute pendistribusian menggunakan metode *Nearest Neighbor* memiliki prinsip bahwa pemilihan lokasi pengiriman berdasarkan jarak terdekat dari kunjungan terakhir yang dilakukan oleh truk [6].

Rute distribusi awal dibentuk menggunakan metode *Nearest Neighbor*. Proses ini dimulai dari depot sebagai titik awal, kemudian secara iteratif memilih pelanggan berikutnya yang memiliki jarak terdekat dari lokasi saat ini. Proses ini berlanjut hingga seluruh pelanggan telah dikunjungi. Metode ini digunakan untuk menghasilkan solusi awal yang cukup baik sebagai dasar untuk tahap optimalisasi selanjutnya. Proses menggunakan metode *Nearest Neighbor* adalah sebagai berikut.

1. Memilih titik pusat sebagai titik awal pengiriman.
2. Menentukan titik dengan jarak terkecil dari gudang atau titik awal, yang selanjutnya adalah melakukan penggabungan antar kedua titik tersebut.
3. Titik konsumen yang terakhir dikunjungi menjadi titik awal, dan selanjutnya mencari konsumen selanjutnya dengan jarak terdekat dari titik awal.
4. Melakukan proses pengulangan sampai dengan kapasitas kendaraan sudah tidak mencukupi untuk melakukan pengiriman.
5. Tarik titik tersebut pada satu garis, titik ini yang dinamakan dengan satu rute perjalanan, dengan kapasitas kendaraan sebagai kendala dalam bentuk pembentukan satu rute perjalanan pengiriman barang.
6. Melakukan proses yang sama pada langkah satu sampai dengan langkah lima.

B. Tabu Search

TS adalah metaheuristik efisien yang digunakan untuk memecahkan masalah optimasi, yang awalnya dikembangkan untuk masalah kombinatorial, tetapi diperluas ke variabel keputusan berkelanjutan. TS dibedakan dengan penggunaan memori adaptif untuk mengeksplorasi ruang

pencairan secara efisien, menghindari solusi sub-optimal lokal berkat mekanisme memori dinamis [7]. Algoritma *Tabu Search* merupakan salah satu metode metaheuristik yang dapat menuntun prosedur pencarian lokal heuristik untuk menjelajahi daerah solusi di luar titik optimal local. Algoritma *Tabu Search* dapat digunakan untuk mencari solusi optimal VRP yaitu rute yang memiliki total jarak tempuh minimum dengan mempertimbangkan kapasitas kendaraan. Langkah Algoritma *Tabu Search* dimulai dengan penentuan initial solution menggunakan *Nearest Neighbor*, evaluasi move menggunakan metode 2-Opt, *Relocated*, dan *Exchange*, *update Tabu List*, kemudian apabila kriteria pemberhentian terpenuhi maka proses Algoritma *Tabu Search* berhenti jika tidak, maka kembali pada evaluasi move. Pembuatan suatu program (rancang bangun) dapat mempercepat proses pencarian solusi optimal pada VRP [8]. Program (rancang bangun) dibuat menggunakan MATLAB yang dimulai dengan membuat *source code* utama menggunakan m.file kemudian desain tampilan dirancang menggunakan fig-file sehingga diperoleh program dalam bentuk GUI (*Graphical User Interface*).

Setelah rute awal terbentuk, optimalisasi dilakukan menggunakan algoritma *Tabu Search*. Algoritma ini mengembangkan solusi dengan mencari kemungkinan pertukaran urutan kunjungan antar pelanggan untuk meminimalkan total jarak tempuh. Untuk menghindari solusi berulang dan meningkatkan eksplorasi ruang solusi, diterapkan mekanisme *tabu list* yang mencatat langkah-langkah yang dilarang dalam iterasi tertentu. Proses iteratif ini berlangsung hingga memenuhi kriteria penghentian berdasarkan jumlah iterasi maksimum atau tidak ditemukannya perbaikan solusi dalam sejumlah iterasi berturut-turut.

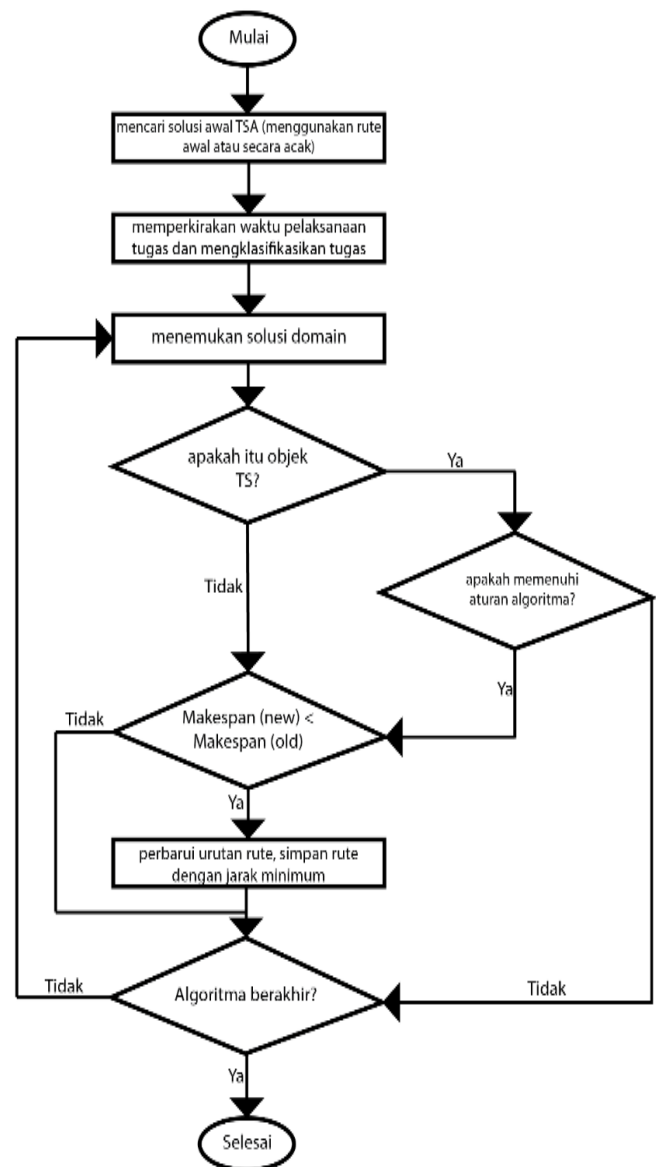
Tabu Search adalah sebuah metode optimasi yang berbasis pada *local search*. Proses pencarian bergerak dari satu solusi ke solusi berikutnya, dengan cara memilih solusi terbaik *Neighborhood* sekarang (*current*) yang tidak tergolong solusi terlarang (*Tabu*). Ide dasar dari algoritma *Tabu Search* adalah mencegah proses pencarian dari *local search* agar tidak melakukan pencarian ulang pada ruang solusi yang sudah pernah ditelusuri, dengan memanfaatkan suatu struktur memori yang mencatat sebagian jejak proses pencarian yang telah dilakukan [9]. Pada tahap ini solusi rute sementara yang didapatkan dari *Nearest Neighbor* akan dilakukan optimalisasi kembali menggunakan algoritma *Tabu Search*. Dengan melakukan perhitungan kemungkinan solusi alternatif yang dapat ditemukan dengan menggunakan persamaan kombinasi. Selanjutnya membuat iterasi dengan melakukan penukaran antara dua titik node dari hasil *Nearest Neighbor* [10].

Hasil *nearest neighbour* akan dimasukkan kedalam daftar *Tabu list*. Setelah seluruh solusi alternatif ditemukan tahap selanjutnya yaitu mengevaluasi solusi alternatif yang dihasilkan dari percobaan pada jalur-jalur selama iterasi tersebut, dan kemudian menetapkan solusi tersebut sebagai solusi sementara yang optimal. Kemudian lakukan pemilihan solusi terbaik dari semua opsi alternatif dilakukan, dan jika nilai solusi yang baru lebih rendah dibandingkan dengan nilai solusi awal, maka solusi yang baru tersebut akan diambil sebagai solusi optimum yang terbaru. Apabila solusi optimum baru telah ditemukan tahap selanjutnya yaitu

memperbarui daftar *Tabu list* setelah mendapatkan solusi optimum yang baru. Solusi ini akan dimasukkan ke dalam *Tabu list* dan digunakan sebagai solusi awal pada iterasi berikutnya [11]. Hal tersebut dilakukan seterusnya hingga iterasi ke 100. Langkah ini berakhir jika semua kriteria telah terpenuhi, langkah ini disebut terminasi [12].

TS telah secara konsisten dibandingkan dengan goritma metaheuristik lainnya untuk mengevaluasi efektivitasnya [13]. Karya ini menyajikan algoritma pencarian tabu (TS) khusus untuk memecahkan jenis masalah tertentu, seperti masalah VRP, di mana upaya komputasi yang diperlukan untuk mengevaluasi daftar tabu (dengan semua solusi yang dikunjungi sebelumnya tersimpan) tidak signifikan dibandingkan dengan evaluasi lingkungan sekitar [14]. *Tabu Search* digunakan sebagai metode optimasi yang dapat memperbaiki solusi awal dengan menghindari eksplorasi solusi yang sama melalui pencatatan riwayat pencarian [15].

Pada proses ini digunakan Matlab untuk mengolah data. Gambar 1, merupakan *flowchart* algoritma *Tabu Search*.



Gambar 1. Flowchart *Tabu Search*

III. HASIL DAN PEMBAHASAN

Penelitian ini dilakukan guna menentukan rute yang optimal pada proses distribusi es kristal pada PT. Sinar Kristal Nusantara. Pada proses pengolahan datanya digunakan aplikasi Matlab untuk menjalankan *Nearest Neighbor* dan algoritma *Tabu Search* yang bertujuan untuk menemukan rute paling optimal pada rute distribusi.

A. Nearest Neighbor

Data matrix jarak antar lokasi pelanggan diperlukan untuk melakukan pengurutan rute menggunakan *Nearest Neighbor*. matrix jarak antar lokasi pelanggan dapat dilihat pada tabel 2.

TABEL 2. MATRIX JARAK ANTAR LOKASI PELANGGAN

Dari/ke (NODE)	(GUDANG) 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	72
(GUDANG) 1	0,00	3,27	3,67	3,05	4,75	5,43	1,26	4,64	4,38	2,57	2,23	2,81	3,44	2,75	4,76	3,09
2	3,27	0,00	0,56	0,22	2,68	2,54	2,43	1,56	1,78	1,10	1,09	1,70	0,23	0,63	2,09	0,40
3	3,67	0,56	0,00	0,70	2,20	1,98	2,70	1,02	1,26	1,24	1,44	1,64	0,32	1,18	1,55	0,59
4	3,05	0,22	0,70	0,00	2,70	2,68	2,20	1,72	1,86	0,91	0,86	1,56	0,41	0,49	2,19	0,30
5	4,75	2,68	2,20	2,70	0,00	1,49	3,49	1,81	1,00	2,39	2,91	1,94	2,48	3,17	0,96	2,41
6	5,43	2,54	1,98	2,68	1,49	0,00	4,28	1,01	1,09	2,87	3,27	2,77	2,30	3,16	0,73	2,49
7	1,26	2,43	2,70	2,20	3,49	4,28	0,00	3,58	3,20	1,48	1,34	1,57	2,55	2,09	3,58	2,14
8	4,64	1,56	1,02	1,72	1,81	1,01	3,58	0,00	0,86	2,11	2,42	2,24	1,33	2,19	0,88	1,58
9	4,38	1,78	1,26	1,86	1,00	1,09	3,20	0,86	0,00	1,83	2,28	2,28	1,69	1,57	2,35	0,38
10	2,57	1,10	1,24	0,91	2,39	2,87	1,48	2,11	1,83	0,00	0,53	0,72	1,14	1,11	2,22	0,72
11	2,23	1,09	1,44	0,86	2,91	3,27	1,34	2,42	2,28	0,53	0,00	1,20	1,23	0,79	2,66	0,86
12	2,81	1,70	1,64	1,56	1,94	2,77	1,57	2,24	1,69	0,72	1,20	0,00	1,67	1,82	2,06	1,30
13	3,44	0,23	0,32	0,41	2,48	2,30	2,55	1,33	1,57	1,14	1,23	1,67	0,00	0,86	1,87	0,42
14	2,75	0,63	1,18	0,49	3,17	3,16	2,09	2,19	2,35	1,11	0,79	1,82	0,86	0,00	2,68	0,76
15	4,76	2,09	1,55	2,19	0,96	0,73	3,58	0,88	0,38	2,22	2,66	2,06	1,87	2,68	0,00	1,95
16	2,57	1,84	1,85	1,68	2,18	3,05	1,31	2,51	1,97	0,78	1,17	0,28	1,84	1,88	2,33	1,44
17	0,92	3,71	4,02	3,48	4,66	5,57	1,32	4,90	4,49	2,80	2,62	2,81	3,85	3,29	4,86	3,44
...
72	3,09	0,40	0,59	0,30	2,41	2,49	2,14	1,58	1,60	0,72	0,86	1,30	0,42	0,76	1,95	0,00

Setelah mendapatkan nilai matrix jarak antar lokasi pelanggan langkah pertama pada metode *Nearest Neighbor* adalah identifikasi jarak dari gudang ke masing masing lokasi outlet. Jarak pendistribusian dapat dilihat pada tabel matrix jarak antar lokasi pelanggan. Jarak terdekat dari titik awal (Gudang) ke 72 node (outlet) ada pada node 17 yaitu 0,92. Kemudian setelah mendapatkan jarak terdekat dengan gudang, langkah selanjutnya yaitu mencari jarak terdekat dari node 17 ke masing-masing node untuk menentukan urutan selanjutnya sampai semua node telah dikunjungi.

Iterasi 1

Gudang (*Node 1*) diketahui merupakan titik awal dan titik akhir dari sebuah rute, sehingga pada iterasi pertama (1) outlet dengan jarak terdekat dari titik awal merupakan tujuan selanjutnya. Jarak terdekat dari *node 1* yaitu *node 17* dengan jarak **0,92 km**. nilai jarak dapat dilihat pada tabel 2.

Untuk iterasi 2 dan seterusnya dilakukan dengan cara yang sama yaitu mencari jarak terdekat dari titik sebelumnya (*node 17*) ke seluruh node yang belum dikunjungi hingga semua nodes telah dikunjungi.

Setelah seluruh node telah dikunjungi/ telah masuk ke dalam rute maka di dapatkan urutan rute berdasarkan metode *Nearest Neighbor* yaitu: 1 – 17 – 71 – 7 – 19 – 41 – 16 – 12

– 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 31 – 42 – 9 – 56 – 15 – 47 – 25 – 62 – 22 – 45 – 53 – 6 – 33 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 48 – 14 – 37 – 46 – 38 – 23 – 68 – 52 – 5 – 55 – 60 – 51 – 1, dengan jarak total **39,31 Km**.

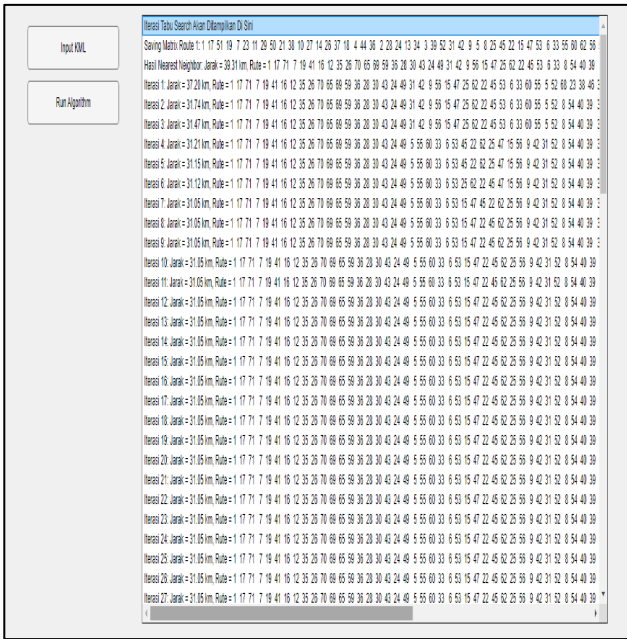
B. Tabu Search

Setelah menjalankan program komputasi algoritma *Tabu Search* pada Matlab maka akan menampilkan data penukaran node pada setiap iterasi *Tabu Search* hingga menemukan solusi paling optimal dari tiap iterasi yang telah dibuat. Hasil iterasi algoritma *Tabu Search* dapat dilihat pada tabel 3.

TABEL 3. HASIL ITERASI ALGORITMA *TABU SEARCH*

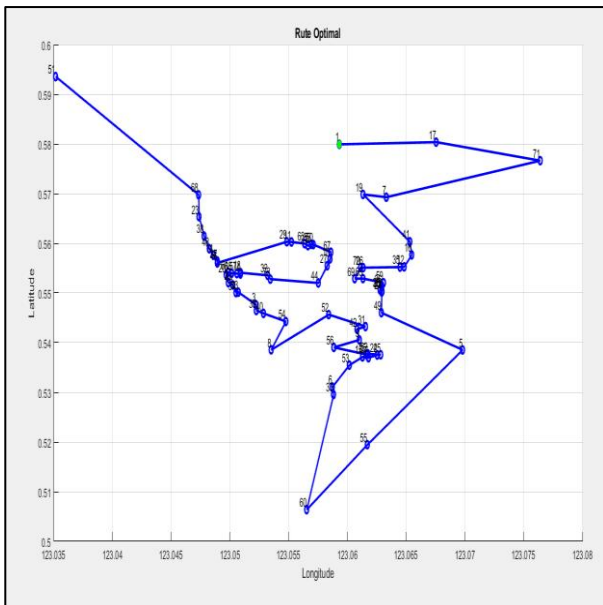
Iterasi (37,20 Km)	1	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 31 – 42 – 9 – 56 – 15 – 47 – 25 – 62 – 22 – 45 – 53 – 6 – 33 – 60 – 55 – 5 – 52 – 68 – 23 – 38 – 46 – 37 – 14 – 48 – 29 – 11 – 63 – 21 – 58 – 50 – 67 – 10 – 27 – 44 – 72 – 32 – 18 – 4 – 57 – 66 – 61 – 20 – 2 – 64 – 34 – 13 – 3 – 39 – 40 – 54 – 8 – 51
Iterasi (31,74 Km)	2	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 31 – 42 – 9 – 56 – 15 – 47 – 25 – 62 – 22 – 45 – 53 – 6 – 33 – 60 – 55 – 5 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 48 – 14 – 37 – 46 – 38 – 23 – 68 – 51
Iterasi (31,47 Km)	3	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 31 – 42 – 9 – 56 – 15 – 47 – 25 – 62 – 22 – 45 – 53 – 6 – 33 – 60 – 55 – 5 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51
Iterasi (31,21 Km)	4	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 45 – 22 – 62 – 25 – 47 – 15 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51
Iterasi (31,15 Km)	5	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 45 – 22 – 62 – 25 – 47 – 15 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51
Iterasi (31,12 Km)	6	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 25 – 62 – 22 – 45 – 47 – 15 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51
Iterasi (31,05 Km)	7	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 15 – 47 – 45 – 22 – 62 – 25 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51
Iterasi (31,05 Km)	8	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 15 – 47 – 45 – 22 – 62 – 25 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51
Iterasi (31,05 Km)	100	1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 15 – 47 – 45 – 22 – 62 – 25 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51

Untuk memudahkan proses optimalisasi pada Matlab, dibuatkan GUI khusus dengan tampilan sederhana dan mudah digunakan untuk menampilkan visualisasi perbandingan rute hasil optimalisasi menggunakan metode *Nearest Neighbor* dan algoritma *Tabu Search*.



Gambar 2. Output Iterasi Pada Matlab

Gambar 2, merupakan tampilan output program algoritma *Tabu Search* pada gui Matlab yang berisi pengurutan rute dengan melakukan penukaran node hingga iterasi maksimum dicapai.



Gambar 3. Visualisasi Rute Optimal *Tabu Search*

Berdasarkan hasil optimalisasi algoritma *Tabu Search* Gambar3, didapatkan rute yang lebih optimal dibanding rute pada metode sebelumnya dengan max iterasi 100 : 1 – 17 – 71 – 7 – 19 – 41 – 16 – 12 – 35 – 26 – 70 – 65 – 69 – 59 – 36 – 28 – 30 – 43 – 24 – 49 – 5 – 55 – 60 – 33 – 6 – 53 – 15 – 47 – 45 – 22 – 62 25 – 56 – 9 – 42 – 31 – 52 – 8 – 54 – 40 – 39 – 3 – 13 – 34 – 64 – 2 – 20 – 61 – 66 – 57 – 4 – 18 – 32 – 72 – 44 – 27 – 10 – 67 – 50 – 58 – 21 – 63 – 11 – 29 – 46 – 14 – 37 – 48 – 38 – 23 – 68 – 51 – 1, dengan total jarak **31,05 Km.**

Berdasarkan hasil optimasi diatas diperoleh persentase penghematan total jarak tempuh sebagai berikut :

Persentase jarak tempuh :

$$= \frac{\text{jarak rute NN} - \text{jarak rute TS}}{\text{jarak rute NN}} \times 100\%$$

$$= \frac{39,31 - 31,05}{39,31} \times 100\%$$

$$= 21,01\%$$

IV. KESIMPULAN

Hasil optimasi rute yang dilakukan menggunakan algoritma *Tabu Search* menghasilkan total jarak tempuh yaitu 31,05 Km. lebih optimal dibanding *Nearest Neighbor* yaitu 39,31 km dengan persentase penghematan yaitu 21,01%. Algoritma *Tabu Search* dapat digunakan sebagai metode optimasi lanjutan karena dapat menghasilkan rute dengan total jarak yang lebih optimal dibandingkan hanya menggunakan satu metode optimasi seperti *Nearest Neighbor*. Hasil penelitian ini memberikan implikasi praktis bagi manajemen logistik PT. Xyz dalam mengelola distribusi produk secara lebih efisien. Dengan penerapan algoritma *Nearest Neighbor* dan *Tabu Search*, perusahaan dapat menentukan rute distribusi yang lebih optimal, sehingga dapat menurunkan biaya operasional seperti bahan bakar dan jam kerja pengemudi. Selain itu, efisiensi pengiriman yang lebih tinggi juga berpotensi meningkatkan kepuasan pelanggan serta daya saing perusahaan dalam industri distribusi produk es kristal. Penelitian ini dapat dikembangkan lebih lanjut dengan menerapkan algoritma lain seperti Ant Colony Optimization dan Genetic Algorithm yang mampu menangani masalah rute berskala besar. Selain itu, beberapa algoritma metaheuristik yang jarang dibahas namun menjanjikan, seperti Particle Swarm Optimization, Artificial Bee Colony, Grey Wolf Optimizer, dan Firefly Algorithm, juga dapat dieksplorasi untuk meningkatkan kualitas solusi. Pengujian pada skenario dinamis seperti perubahan permintaan, kemacetan lalu lintas, atau gangguan distribusi secara real-time juga penting dilakukan agar solusi yang dihasilkan lebih adaptif dan robust terhadap kondisi nyata dalam sistem logistik.

REFERENSI

- [1] R. T. Payungi, "Optimalisasi Rute Distribusi dengan Menggunakan Metode Nearest Neighbour Dan Algoritma Tabu Search Pada PT RTP," vol. 3, no. 2, pp. 80–85, 2022.
- [2] R. A. Nolly, A. Fitria, and K. Saputra S, "Penerapan Algoritma K-Nearest Neighbors untuk Klasifikasi Fragmen Metagenom Berdasarkan Ekstraksi Fitur K-Mers," *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 17, no. 1, p. 52, 2023, doi: 10.30872/jim.v17i1.5779.
- [3] M. Kuronen, J. Rätty, P. Packalen, and M. Myllymäki, "Uncertainty quantification for forest attribute maps with conformal prediction and k-nearest neighbor method," *Remote Sens. Environ.*, vol. 325, no. April, p. 114758, 2025, doi: 10.1016/j.rse.2025.114758.
- [4] M. M. Kumbure and P. Luukka, "Generalizing fuzzy k-nearest neighbor classifier using an OWA operator with a RIM quantifier," *Expert Syst. Appl.*, vol. 282, no. April, 2025, doi: 10.1016/j.eswa.2025.127795.

- [5] A. Amrulloh and E. I. Sela, "Course scheduling optimization using genetic algorithm and tabu search," *J. Teknol. dan Sist. Komput.*, vol. 9, no. 3, pp. 157–166, 2021, doi: 10.14710/jtsiskom.2021.14137.
- [6] P. Lestari, A. Hasibuan, and B. Harahap, "Analisis Penentuan Rute Distribusi menggunakan Metode Nearest Neighbor di PT Medan Juta Rasa Tanjung Morawa," *Fact. J. Ind. Manaj. dan Rekayasa Sist. Ind.*, vol. 1, no. 1, pp. 26–32, 2022, doi: 10.56211/factory.v1i1.110.
- [7] S. El Bakali, H. Ouadi, G. Fouad, S. Gheouany, M. Nada, and J. Ismael, "Tabu Search Algorithm for Optimal Electric Water Heating Energy Management," *IFAC-PapersOnLine*, vol. 58, no. 13, pp. 835–840, 2024, doi: 10.1016/j.ifacol.2024.07.586.
- [8] R. Rumaida, F. Rakhmawati, D. Juliandri, P. S. Matematika, U. Islam, and N. Sumatera, "Penerapan Algoritma Tabu Search pada Capacitated Vehicle Routing Problem Pengangkutan Sampah di Kota Padang Sidempuan," vol. 2, no. 5, 2024.
- [9] R. P. Ritonga, M. Zakaria, and D. Syukriah, "Distribution Route Assignment Using Tabu Search Algorithm at PT. Yakult Indonesia Persada Lhokseumawe Branch (Penugasan Rute Distribusi Menggunakan Algoritma Tabu Search pada PT. Yakult Indonesia Persada Cabang Lhokseumawe)," *Ind. Eng. J.*, vol. 10, no. 1, 2021.
- [10] A. I. Pratiwi *et al.*, "Penentuan Rute Terbaik Pendistribusian Produk Wafer dengan Metode Algoritma Genetika (Studi Kasus di Perusahaan Jasa Pergudangan Produk Wafer Karawang)," *JISI J. Integr. Sist. Ind.*, vol. 10, no. 2, p. 69, 2023, doi: 10.24853/jisi.10.2.69-75.
- [11] Riswan, A. Sahari, and D. Lusiyanti, "Penentuan Rute Terpendek Pendistribusian Tabung Gas Lpg 3 Kg Pt. Fega Gas Palu Pratama Menggunakan Algoritma Tabu Search," *J. Ilm. Mat. Dan Terap.*, vol. 16, no. 2, pp. 221–229, 2020, doi: 10.22487/2540766x.2019.v16.i2.15004.
- [12] R. E. Toyosito, L. Citra Ramadhanti, and A. Y. Bustommy, "Penjadwalan Flow Shop dengan Metode Algoritma Heuristik Pour, Algoritma Campbell Dudek And Smith, Algoritma Tabu Search di Industri Porcelain Tableware," *J. JITES*, vol. 1, no. 1, 2021.
- [13] P. Jartnillaphand, "Computers and Operations Research A tabu search algorithm for Unspecified Parallel Machine scheduling with shift consideration," *Comput. Oper. Res.*, vol. 183, no. June, p. 107151, 2025, doi: 10.1016/j.cor.2025.107151.
- [14] R. Y. Yamamoto, T. Pinto, R. Romero, and L. H. Macedo, "Specialized tabu search algorithm applied to the reconfiguration of radial distribution systems," *Int. J. Electr. Power Energy Syst.*, vol. 162, no. August, 2024, doi: 10.1016/j.ijepes.2024.110258.
- [15] D. G. Mogale, A. Ghadge, and S. K. Jena, "Modelling and optimising a multi-depot vehicle routing problem for freight distribution in a retail logistics network," *Comput. Ind. Eng.*, vol. 207, no. March, p. 111315, 2025, doi: 10.1016/j.cie.2025.111315.