

PENERAPAN PARTICLE SWARM OPTIMATION (PSO) UNTUK MENENTUKAN RUTE DISTRIBUSI PADA KANTOR POS MPC XYZ

Yoga Adi Pratama Putra¹, Muhammad Firmansyah², Dwi Prayitno³, Irsyadin Ainul Hakim⁴, Samsul Amar⁵, Trisita Novianti⁶
220421100115@student.trunojoyo.ac.id¹

Program Studi Teknik Industri, Fakultas Teknik, Universitas Trunojoyo Madura

ABSTRAK

Kantor Pos MPC XYZ merupakan pusat penyortiran dan pengiriman surat serta paket ke beberapa delivery center di wilayah sekitarnya. Dalam sistem pengiriman, terdapat tiga jenis transportasi, yaitu transportasi primer, sekunder, dan tersier. Transportasi primer digunakan untuk mengirimkan barang dari pusat sortir nasional ke MPC, transportasi sekunder dari MPC ke kantor pos besar di daerah, sedangkan transportasi tersier digunakan untuk mengirimkan surat dan paket dari MPC ke delivery center sebelum diteruskan ke penerima. Saat ini pengiriman dari Center Kantor Pos MPC XYZ ke 12 delivery center dilakukan menggunakan 3 unit kendaraan. Kondisi tersebut masih bisa ditingkatkan agar penggunaan biaya, waktu, dan jarak tempuh menjadi lebih baik lagi. Oleh karena itu, penelitian ini dilakukan untuk mencari rute pengiriman tersier yang lebih baik lagi dari Center Kantor Pos MPC XYZ ke 12 delivery center dengan mempertimbangkan kapasitas kendaraan dan jumlah permintaan di setiap delivery center, sehingga total jarak, waktu, dan biaya pengiriman dapat ditekan seminimal mungkin. Metode yang digunakan dalam penelitian ini adalah Particle Swarm Optimization (PSO), yaitu metode optimasi yang meniru perilaku sekelompok burung dalam mencari posisi terbaik. Dengan metode ini, diharapkan dapat ditemukan rute pengiriman yang lebih baik lagi dan sesuai dengan kondisi sebenarnya di lapangan, sehingga sistem distribusi di Center Kantor Pos MPC XYZ dapat berjalan lebih baik dari sebelumnya.

Kata Kunci: Rute Distribusi, Jarak Terpendek, Kapasitas Dan Kendaraan.

PENDAHULUAN

Distribusi adalah proses penyaluran barang atau jasa dari produsen ke konsumen melalui tahapan tertentu agar produk dapat sampai ke tangan pelanggan tepat waktu, dalam jumlah yang sesuai, dan dengan biaya yang efisien (Afandy, 2023). Tantangan utama dalam proses distribusi adalah penentuan rute distribusi untuk mencapai hasil jarak yang terdekat atau yang sering dikenal dengan nama Traveling Salesman Problem (TSP).

Center Kantor Pos MPC XYZ memiliki permasalahan dalam menentukan rute distribusi dengan jarak yang optimal. Saat ini, proses distribusi surat dan paket dari Center Kantor Pos MPC XYZ ke 12 delivery center masih dilakukan berdasarkan intuisi operator lapangan, di mana urutan pengantaran ditentukan dari jumlah kiriman terbanyak terlebih dahulu. Hal ini menyebabkan proses pengiriman berlangsung secara acak tanpa mempertimbangkan jarak terpendek sebagai faktor utama dalam perencanaan rute. Akibatnya, sangat mungkin terjadi pemborosan dalam hal biaya bahan bakar, waktu tempuh, serta efisiensi operasional kendaraan. Oleh karena itu, diperlukan upaya untuk menentukan rute distribusi yang efisien melalui pendekatan optimasi. Salah satu metode yang dapat digunakan adalah algoritma Particle Swarm Optimization (PSO), yang termasuk dalam pendekatan metaheuristik dan mampu menemukan solusi rute terpendek secara cepat dan efektif.

Travelling Salesman Problem (TSP) merupakan suatu problem optimasi yang digunakan untuk menentukan rute distribusi paling efisien dengan jarak dan biaya tempuh yang minimal. Dalam penelitian yang dilakukan di PT. Inbisco Niagatama Semesta (Mayora Group) di Makassar, metode TSP diterapkan untuk mencari rute pengiriman produk yang optimal guna meningkatkan efisiensi operasional dan menekan biaya distribusi perusahaan. Distribusi produk menjadi salah satu elemen penting dalam rantai pasok, sehingga

penentuan rute yang tidak optimal dapat menyebabkan pemborosan biaya transportasi, keterlambatan waktu pengiriman, serta penurunan efisiensi sistem logistik secara keseluruhan (Kharisma, 2024). Oleh karena itu, penerapan metode TSP dalam penentuan rute distribusi dinilai mampu membantu perusahaan mencapai sistem pengiriman yang lebih efektif dan hemat biaya.

Menurut Febiani (2024). Algoritma Particle Swarm Optimization (PSO) pertama kali diperkenalkan oleh Dr. Eberhart dan Dr. Kennedy pada tahun 1995. Algoritma ini termasuk dalam kategori swarm intelligence, yaitu metode optimasi yang terinspirasi dari perilaku sosial sekelompok makhluk hidup seperti burung dan ikan dalam mencari makanan dan bertahan hidup. Dalam algoritma PSO, setiap individu disebut partikel, dan sekumpulan partikel tersebut disebut swarm (kerumunan). Setiap partikel mewakili sebuah solusi kandidat dari permasalahan optimasi yang sedang dicari. Partikel-partikel tersebut akan bergerak dalam ruang pencarian dengan menyesuaikan posisi dan kecepatannya berdasarkan pengalaman terbaiknya sendiri (pbest) dan pengalaman terbaik kelompoknya (global best). Sejak diperkenalkan, PSO berkembang pesat karena memiliki struktur yang sederhana, mudah dipahami, dan mampu memberikan hasil optimasi yang andal di berbagai bidang, termasuk penjadwalan, pengelolaan sumber daya, dan perencanaan system.

Penerapan metode Particle Swarm Optimization (PSO) dinilai sesuai untuk membantu menyelesaikan permasalahan distribusi di Kantor Pos MPC XYZ karena kemampuannya dalam mencari solusi yang mendekati terbaik dengan waktu perhitungan yang relatif cepat. PSO bekerja dengan menyesuaikan posisi solusi berdasarkan pengalaman individu dan kelompok, sehingga dapat membantu menemukan kombinasi rute yang lebih efisien dengan jarak tempuh yang lebih singkat. Selain itu, algoritma ini tergolong mudah diterapkan dan memiliki tingkat kestabilan yang cukup baik, sehingga cocok digunakan pada kondisi operasional Kantor Pos yang memerlukan ketepatan waktu serta pengelolaan biaya pengiriman yang lebih terkendali. Beberapa penelitian sebelumnya juga menunjukkan bahwa PSO mampu memberikan hasil yang lebih baik dibandingkan beberapa metode lainnya. Menurut (Ramadhani, 2021) PSO dapat menghemat bahan bakar hingga 13% dibandingkan rute awal perusahaan. (Emambocus, 2021) menyebutkan bahwa PSO menghasilkan rute yang lebih singkat dengan waktu perhitungan yang lebih cepat. Hasil penelitian (Guntara, 2023) menunjukkan bahwa PSO memberikan peningkatan efisiensi sekitar 4,8% dibandingkan metode lain, sementara (Kurniawan, 2021) mencatat penurunan jarak tempuh hingga 13%. Berdasarkan hasil-hasil tersebut, PSO dapat menjadi salah satu alternatif metode yang cukup efektif untuk membantu Kantor Pos MPC XYZ dalam meningkatkan sistem distribusi.

METODOLOGI

Penelitian ini dilakukan dengan menggunakan data yang diperoleh dari jurnal terkait rute distribusi dari center Kantor Pos MPC XYZ hingga ke 12 delivery center, sehingga diperoleh data matriks jarak lokasi dari center Kantor Pos MPC XYZ hingga ke 12 delivery center lokasi distribusi. Penelitian ini bertujuan untuk mencari rute terpendek distribusi dari center Kantor Pos MPC XYZ ke 12 delivery center yang harus didistribusikan dengan menggunakan algoritma Particle Swarm Optimization (PSO). Particle Swarm Optimization (PSO) merupakan metode komputasi yang digunakan untuk melakukan proses penilaian atau optimasi dengan menggunakan algoritma pencarian berbasis populasi. Dalam metode ini, pencarian dimulai dari sekumpulan partikel yang dibentuk secara acak. Berbeda dengan metode optimasi lainnya, PSO memberikan kecepatan pada setiap partikel yang akan terus berubah berdasarkan hasil pencarian sebelumnya. Dengan mekanisme tersebut, partikel-partikel akan secara bertahap bergerak menuju posisi yang dianggap paling optimal atau

paling sesuai setelah melalui beberapa tahap pencarian. Berikut penjelasan istilah-istilah umum yang sering digunakan dalam Particle Swarm Optimization (PSO) dengan bahasa yang lebih sederhana:

1. Swarm adalah kumpulan partikel atau populasi dalam algoritma PSO. Setiap partikel mewakili kemungkinan solusi terhadap suatu masalah.
2. Particle merupakan individu dalam swarm. Setiap partikel menggambarkan satu solusi potensial, dan posisinya menunjukkan keadaan solusi tersebut pada saat tertentu.
3. Pbest (Personal Best) adalah posisi terbaik yang pernah dicapai oleh suatu partikel selama proses pencarian solusi berlangsung.
4. Gbest (Global Best) merupakan posisi terbaik dari seluruh partikel yang ada dalam swarm, atau posisi terbaik di antara semua nilai Pbest.
5. Velocity (Kecepatan, v) adalah nilai yang menentukan arah dan seberapa jauh sebuah partikel akan bergerak untuk mencari solusi yang lebih baik. Nilai ini terus berubah agar partikel dapat memperbaiki posisinya.
6. Inertia Weight (w) adalah parameter yang mengatur seberapa besar pengaruh kecepatan sebelumnya terhadap pergerakan partikel berikutnya. Dengan kata lain, ini mengontrol keseimbangan antara eksplorasi dan eksploitasi dalam pencarian solusi.
7. Learning Rates ($c1$ dan $c2$) adalah konstanta yang berfungsi untuk menilai sejauh mana partikel dipengaruhi oleh pengalamannya sendiri ($c1$) dan oleh pengalaman partikel lain dalam swarm ($c2$). Kedua nilai ini menentukan bobot atau pengaruh dari masing-masing faktor tersebut (Suprianto, 2024).

Berikut merupakan tabel jarak antar delivery center sebagaimana dapat dilihat pada tabel 1.

Tabel 1. Jarak antar delivery center

Distance	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12
D0	0	10	13	72	64	78	73	20	107	222	253	2012	215
D1	10	0	14	147	103	72	151	33	314	108	35	246	228
D2	132	134	0	63	33	67	141	212	114	41	176	139	3412
D3	72	151	63	0	3	822	32	1127	11	256	1102	164	357
D4	10	1112	6	41	0	6	101	32	113	252	20	163	281
D5	11	86	612	83	6	0	143	25	146	122	218	173	35
D6	73	153	144	712	83	131	0	177	247	221	262	238	18
D7	20	342	24	172	2181	11212	165	0	11712	3125	255	361	323
D8	208	21	85	112	118	144	187	207	0	331	113	2012	367
D9	222	111	281	222	222	18	226	357	33	0	40	384	166
D10	132	273	147	137	103	207	252	263	124	312	0	212	432
D11	264	216	127	1104	17	175	253	342	1122	331	221	0	468
D12	2112	224	3012	258	368	2712	18	323	363	163	44	412	0

Berikut merupakan tabel jumlah muatan barang dari MPC XYZ ke delivery center sebagaimana dapat dilihat pada tabel 2.

Tabel 2 Jumlah Berat Barang Tiap Delivery center

No	Nama delivery center	Berat kiriman (dalam kg)
1	D1	423
2	D2	311
3	D3	342
4	D4	223
5	D5	121
6	D6	476
7	D7	342
8	D8	331
9	D9	123
10	D10	185
11	D11	391
12	D12	232
Total Berat Kiriman		3500

Berdasarkan data pada Tabel 2, diketahui bahwa MPC XYZ rata-rata hanya mengirimkan sekitar 400 kg barang ke setiap delivery center. Pengiriman dengan muatan terbesar dilakukan ke delivery center D6 dengan berat 476 kg, sedangkan yang paling ringan adalah ke D5 dengan berat hanya 121 kg. Padahal, setiap kendaraan yang disediakan MPC XYZ memiliki kapasitas angkut hingga 1500 kg, namun penggunaannya masih jauh dari kapasitas maksimal karena rata-rata muatan tidak mencapai 400 kg per rute.

Berikut merupakan gambar lokasi DC yang digambarkan oleh google maps untuk dicari rute distribusi dengan jarak terpendek yang dapat dilihat pada gambar 1.



Gambar 1 Delivery center

Tahapan yang dilakukan untuk menentukan rute dengan jarak terpendek menggunakan algoritma Particle Swarm Optimization (PSO) melalui software Google Colab dimulai dengan proses penginputan kode awal. Langkah awal ini dilakukan dengan menuliskan dan menjalankan skrip dasar PSO di Google Colab sebagai tahap persiapan sebelum masuk ke proses perhitungan dan optimasi jarak. Gambaran proses awal penginputan kode pada Google Colab dapat dilihat pada gambar 2.

```
import numpy as np
import random
import matplotlib.pyplot as plt
```

Gambar 2 Input Kode Awal Google Colab

Langkah awal dalam proses pengolahan dan analisis data dilakukan dengan mengimpor beberapa pustaka utama seperti `pandas` yang digunakan untuk pengelolaan serta analisis data (Guntara, 2023). Selanjutnya, `NumPy` dimanfaatkan untuk memasukkan dan mengelola data numerik, sedangkan `Matplotlib` berperan dalam menampilkan hasil analisis melalui visualisasi grafik (Syaifudin, 2024). Setelah itu, dilakukan proses penginputan data berupa jarak antar lokasi retail menggunakan matriks jarak yang diimplementasikan pada Google Colab. Ilustrasi mengenai penginputan matriks jarak dari distribusi center ke retail dapat dilihat pada gambar 3.

[illegible]

Gambar 3 Input data matriks jarak dari distribusi center ke retail

Proses penginputan data ini dilakukan dengan tujuan untuk memperoleh hasil akhir yang ingin dicapai. Setelah data jarak selesai dimasukkan, tahap berikutnya adalah menuliskan kode program untuk menjalankan algoritma Particle Swarm Optimization (PSO), termasuk menentukan parameter-parameter yang dibutuhkan agar algoritma tersebut

dapat berjalan dengan baik. Ilustrasi proses penjalanan kode dan parameter PSO ditunjukkan pada Gambar 4.

```
# Konversi ke matriks numpy
cities = ['DC', 'D1', 'D2', 'D3', 'D4', 'D5', 'D6', 'D7', 'D8', 'D9', 'D10', 'D11', 'D12']
distance_matrix = np.array(distance_data)

# Jalankan PSO dengan DC sebagai titik awal dan akhir
pso = PSO_TSP_DC(distance_matrix, cities, num_particles=60, max_iter=10000)
best_route, best_distance, convergence = pso.optimize()

# Visualisasi hasil
pso.visualize_results(best_route, best_distance, convergence)
```

Gambar 4 kode algoritma PSO

Kode tersebut menggambarkan tahapan penting dalam penerapan algoritma Particle Swarm Optimization (PSO) untuk menyelesaikan permasalahan optimasi rute distribusi dengan pusat distribusi (DC) sebagai titik awal dan akhir perjalanan. Pada tahap awal, data jarak antar setiap lokasi retail dikonversi ke dalam bentuk matriks menggunakan library NumPy agar dapat diolah secara matematis dan komputasional. Kemudian, fungsi PSO_TSP_DC dijalankan dengan jumlah partikel sebanyak 60 dan iterasi maksimum 10.000 untuk menemukan kombinasi rute paling efisien dengan jarak total terpendek. Hasil optimasi berupa rute terbaik (best_route) dan jarak minimum (best_distance) selanjutnya divisualisasikan menggunakan Matplotlib, menampilkan kurva konvergensi serta rute distribusi terbaik. Dengan cara ini, proses analisis menjadi lebih mudah dipahami karena hasilnya ditampilkan secara visual. Tahap berikutnya adalah memasukkan kode program yang berfungsi untuk menampilkan hasil akhir dari algoritma PSO, yaitu total jarak tempuh dan urutan rute terbaik yang diperoleh dari proses optimasi. Pada bagian ini, sistem akan menghitung serta menampilkan rute distribusi paling efisien yang dimulai dan berakhir di pusat distribusi. Tampilan hasil kode setelah dijalankan untuk memperoleh output algoritma PSO dapat dilihat pada gambar 5.

```
# Tampilkan hasil detail
print("\n" + "="*70)
print("HASIL OPTIMASI RUTE DENGAN DC SEBAGAI AWAL/AKHIR")
print("="*70)

print("Rute terbaik:")
route_names = [cities[i] for i in best_route]
print(" -> ".join(route_names))

print(f"\nTotal jarak: {best_distance:.2f}")

# Hitung jarak antar segmen
print("\nDetail jarak per segmen:")
total_calc_distance = 0
for i in range(len(best_route) - 1):
    from_city = best_route[i]
    to_city = best_route[i + 1]
    segment_distance = distance_matrix[from_city][to_city]
    total_calc_distance += segment_distance
print(f"(cities[from_city]:3) -> (cities[to_city]:3): {segment_distance:5.2f} km")
```

Gambar 5 Input untuk Hasil algoritma PSO

Penginputan kode tersebut berfungsi untuk menampilkan hasil akhir dari proses optimasi algoritma Particle Swarm Optimization (PSO) yang telah dijalankan sebelumnya. Pada bagian ini, program membuat rute terbaik yang diperoleh dari hasil perhitungan, lengkap dengan urutan rute distribusi setiap lokasi yang dimulai dan berakhir di pusat distribusi. Selain itu, kode ini juga menghitung jarak antar setiap rute menggunakan matriks jarak yang telah dimasukkan sebelumnya, kemudian menjumlahkannya untuk mendapatkan total jarak keseluruhan perjalanan. Hasil akhirnya menampilkan informasi berupa daftar rute, jarak per segmen, serta total jarak tempuh yang menunjukkan tingkat efisiensi dari rute distribusi yang dihasilkan oleh algoritma PSO dengan menggunakan Google Colab. Berikut merupakan hasil algoritma PSO dari Google Colab dapat dilihat pada gambar 6.

```

Rute terbaik:
DC → D3 → D4 → D5 → D2 → D11 → D10 → D8 → D7 → D6 → D12 → D9 → D1 → DC

Total jarak: 162.72

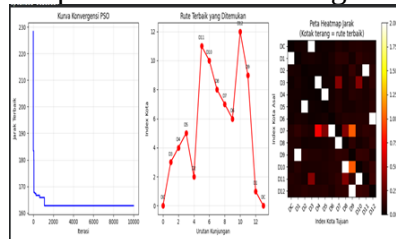
Detail jarak per segmen:
DC → D3 : 7.20 km
D3 → D4 : 3.00 km
D4 → D5 : 6.00 km
D5 → D2 : 6.12 km
D2 → D11: 13.30 km
D11 → D10: 22.10 km
D10 → D8 : 12.40 km
D8 → D7 : 20.70 km
D7 → D6 : 16.50 km
D6 → D12: 18.00 km
D12 → D9 : 16.30 km
D9 → D1 : 11.10 km
D1 → DC : 10.00 km

Total jarak (dihitung ulang): 162.72 km

```

Gambar 6 Hasil algoritma PSO

Hasil perhitungan dari algoritma Particle Swarm Optimization (PSO) menunjukkan bahwa rute distribusi paling efisien dimulai dari pusat distribusi (DC) menuju D3, kemudian berlanjut ke D4, D5, D2, D11, D10, D8, D7, D6, D12, D9, D1, dan kembali lagi ke DC. Total jarak tempuh yang dihasilkan dari rute terbaik ini adalah 162,72 km, dengan rincian jarak antar segmen yang bervariasi mulai dari sekitar 3 km hingga 22 km. Perhitungan ulang menunjukkan hasil total jarak yang sama, yaitu 162,72 km, yang menandakan bahwa proses optimasi berjalan dengan akurat. Hasil ini menggambarkan bahwa algoritma PSO berhasil menemukan jalur paling efisien untuk mengoptimalkan jarak tempuh dalam distribusi ke-12 lokasi retail dengan tetap mempertahankan DC sebagai titik awal dan akhir perjalanan.



Gambar 7 Visualisasi hasil dari proses optimasi algoritma PSO

Gambar tersebut menampilkan tiga visualisasi dari proses optimasi menggunakan algoritma Particle Swarm Optimization (PSO). Grafik pertama menunjukkan kurva konvergensi, di mana nilai jarak terbaik terus menurun hingga stabil di sekitar 160 km, menandakan bahwa algoritma berhasil menemukan solusi optimal setelah beberapa iterasi. Grafik kedua memperlihatkan rute terbaik yang ditemukan, yaitu urutan kunjungan dari pusat distribusi (DC) ke setiap titik retail hingga kembali lagi ke DC, yang menggambarkan jalur paling efisien berdasarkan hasil perhitungan PSO. Sementara itu, grafik ketiga berupa peta heatmap jarak, di mana warna terang menunjukkan pasangan kota yang termasuk dalam rute terbaik.



Gambar 8 Rute distribusi algoritma PSO

Rute distribusi yang dihasilkan oleh algoritma PSO mencakup 12 titik retail yang harus dikunjungi sebelum akhirnya kembali ke gudang sebagai titik awal pengiriman. Penentuan rute ini menggunakan Google Maps sebagai acuan untuk memastikan jalur yang ditempuh merupakan rute terpendek dan paling efisien.

Berdasarkan penelitian Lusul (2024), menggunakan data yang sama dan memperoleh hasil rute distribusi menggunakan metode nearest neighbour menghasilkan jarak 239,72 km, sementara di penelitian ini menggunakan algoritma PSO dan memperoleh total jarak 162,72

km, sehingga terdapat selisih jarak 77 km. Hasil uji menggunakan algoritma PSO, menunjukkan jarak dan urutan rute yang optimal, yang berarti mampu mencapai solusi global optimal. Hal ini disebabkan karena algoritma tersebut menggunakan pendekatan metaheuristik yang mencari solusi global terbaik. PSO bekerja dengan konsep partikel yang bergerak untuk menemukan solusi optimal (Saputra, 2021).

Hasil penelitian ini sejalan dengan penelitian sebelumnya yang menunjukkan penggunaan algoritma PSO mampu memberikan hasil yang efisien dalam penyelesaian masalah Travelling Salesman Problem (TSP). Misalnya, penelitian oleh Ramadhani dan Garside (2021), menemukan bahwa penerapan PSO dapat menghemat bahan bakar hingga 13% dibandingkan dengan rute distribusi awal perusahaan. Hasil serupa juga disampaikan oleh Emambocus (2021), yang menjelaskan bahwa algoritma PSO mampu menghasilkan rute terpendek dengan waktu konvergensi yang lebih cepat dibandingkan algoritma lain, sehingga memberikan solusi yang lebih optimal. Penelitian Guntara (2023), bahkan membandingkan tiga metode penyelesaian TSP dan menemukan bahwa PSO memiliki kinerja paling baik, dengan tingkat efisiensi sekitar 4,8% lebih unggul dari metode lainnya. Sementara itu, penelitian oleh Kurniawan dan Farida (2021), menunjukkan bahwa PSO mampu menurunkan total jarak tempuh dari 115.950 meter menjadi 110.550 meter, yang berarti terjadi peningkatan efisiensi sebesar 13%. Hal ini membuktikan bahwa pendekatan metaheuristik yang digunakan dalam PSO mampu menghasilkan solusi yang optimal dan efisien. Berdasarkan temuan-temuan tersebut, dapat disimpulkan bahwa algoritma Particle Swarm Optimization (PSO) sangat efektif dalam menentukan rute distribusi yang lebih pendek, mengurangi biaya operasional, meningkatkan ketepatan waktu pengiriman, serta memberikan efisiensi jarak tempuh, sehingga layak dijadikan alternatif unggulan dalam penyelesaian masalah TSP (Silalahi, 2020).

KESIMPULAN

Penelitian ini menunjukkan bahwa penerapan algoritma Particle Swarm Optimization (PSO) sangat efektif dalam menentukan rute distribusi yang paling efisien dari Kantor Pos MPC XYZ menuju 12 delivery center. Dengan pendekatan metaheuristik, algoritma PSO berhasil menemukan rute optimal dengan total jarak tempuh 162,72 km, jauh lebih pendek dibandingkan dengan penelitian sebelumnya yang menggunakan metode konvensional nearest neighbour yang mencapai 239,72 km. Rute terbaik yang dihasilkan dimulai dari pusat distribusi, kemudian menuju D3 → D4 → D5 → D2 → D11 → D10 → D8 → D7 → D6 → D12 → D9 → D1, dan akhirnya kembali lagi ke pusat distribusi. Hasil ini menunjukkan adanya peningkatan efisiensi jarak sekitar 32%.

Selain itu, hasil penelitian juga memperlihatkan bahwa algoritma PSO memiliki kemampuan konvergensi yang cepat dan stabil dalam mencari solusi terbaik. Proses optimasi yang dilakukan melalui Google Colab menghasilkan visualisasi rute yang jelas, sehingga memudahkan analisis dan penerapan di lapangan. PSO terbukti unggul karena dapat menyesuaikan posisi solusi secara dinamis berdasarkan pengalaman individu dan kelompok, menghasilkan solusi global yang optimal.

Secara keseluruhan, penerapan PSO dalam sistem distribusi Kantor Pos MPC XYZ memberikan peningkatan efisiensi operasional, pengurangan konsumsi bahan bakar, serta peningkatan ketepatan waktu pengiriman. Dengan demikian, metode ini layak dijadikan alternatif strategis dalam penyelesaian masalah Travelling Salesman Problem (TSP) dan dapat diterapkan pula pada sistem logistik atau distribusi di sektor lainnya.

DAFTAR PUSTAKA

- Afandy, F. R. (2023). Optimasi Pendistribusian Barang dengan Metode Clarke and Wright dan Metode Nearest Neighbour. *Jurnal LOCUS: Penelitian & Pengabdian*, 833–845.
- Emambocus, B. A. (2021). Algoritma pengoptimalan kumpulan partikel berbasis urutan pertukaran yang disempurnakan untuk menyelesaikan TSP. *Ieee Access*, 164820-164836.
- Febiani, A. (2024). Implementasi Algoritma Particle Swarm Optimization (PSO) Penjadwalan Belajar Mengajar. *IKRAITH-INFORMATIKA*, 152–161.
- Guntara, R. G. (2023). Visualisasi Data Laporan Penjualan Toko Online Melalui Pendekatan Data Science Menggunakan Google Colab. *ULIL ALBAB. Jurnal Ilmiah Multidisiplin*, 2091-2100.
- Kharisma, M. L. (2024). Product Distribution Route Optimization Using The Traveling Salesman Problem (TSP) Method At PT. Inbisco Niagatama Semesta (Mayora Group) In Makassar. *Journal of Industrial System Engineering and Management (JISEM)*, 29–37.
- Kurniawan, M. &. (2021). Algoritma Rute Terpendek Particle Swarm Optimization dan Brute Force untuk Optimasi Traveling Salesman Problem. *Jurnal Teknik Informatika*, 191-200.
- Lusel, P. M. (2024). Analisis Perbandingan Rute Distribusi Menggunakan Metode Saving Matrix Dan Nearest Neighbor Di Center Kantor Pos Mpc Bandung. *Journal Of Social Science Research*, 1-15.
- Ramadhani, B. N. (2021). Particle Swarm Optimization algorithm to solve vehicle routing problem with fuel consumption minimization. *Jurnal Optimasi Sistem Industri*, 1-10.
- Saputra, I. (2021). A Comparative Analysis of C4. 5 Classification Algorithm, Naïve Bayes and Support Vector Machine Based on Particle Swarm Optimization (PSO) for Heart Disease Prediction. *Indonesian Scientific Journal*, 84-95.
- Silalahi, B. P. (2020). Algoritme Sweep dan Particle Swarm Optimization dalam Optimisasi Rute Kendaraan dengan Kapasitas. *Jurnal Matematika Integratif*, 29-40.
- Suprianto, K. P. (2024). PENERAPAN PARTICLE SWARM OPTIMIZATION DALAM PENYUSUNAN JADWAL MATA KULIAH DI PRODI MATEMATIKA. *Buletin Ilmiah Math. Stat dan Terapannya (Bimaster)*, 609-618.
- Syaifudin, H. &. (2024). Pengembangan Model Perutean Kendaraan Berbasis Green Logistic dalam Pendistribusian Makanan Ringan pada UD Sumber Rejeki. *Jurnal Kajian Riset Multisiplin*, 1-9.