

Optimasi Pengantaran Barang dengan Vehicle Routing Problem Time Window menggunakan Algoritma Bee colony

Optimizing Goods Delivery with Vehicle Routing Problem Time Window using the Bee Colony Algorithm

Mas Ayoe Elhias Nst¹, Herman Mawengkang², Sutarman³, Syahril Efendi⁴

^{1,4} Faculty of Computer Science and Information Technology, University of Sumatera Utara, Medan, Indonesia

^{2,3} Department of Mathematics and Natural Sciences, University of Sumatera Utara. Medan, Indonesia

ABSTRAK

Pada bagian ini, akan dijelaskan secara singkat tentang masalah VRPTW. Masalah ini melibatkan pengantaran barang dari depot ke sejumlah pelanggan dengan mempertimbangkan batasan waktu yang telah ditentukan. Kendaraan diharuskan untuk mengatur jadwal pengantaran yang efisien agar waktu perjalanan minimal dan semua pelanggan terlayani tepat waktu. Dalam penelitian ini, digunakan algoritma Bee Colony untuk mengatasi masalah VRPTW. Algoritma ini terinspirasi dari perilaku lebah dalam mencari makanan. Algoritma ini terdiri dari sejumlah lebah pekerja yang mencari solusi terbaik melalui iterasi dan proses pertukaran informasi. Melalui mekanisme ini, algoritma Bee Colony mampu menemukan solusi optimal atau mendekati solusi optimal untuk masalah VRPTW. Pada bagian ini, akan dijelaskan tentang implementasi algoritma Bee Colony dalam penyelesaian VRPTW. Dengan melakukan uji coba dan eksperimen menggunakan dataset yang relevan dengan masalah VRPTW. Hasil eksperimen dijelaskan secara singkat. Dalam hal ini penulis mencantumkan temuan utama dari penelitian ini. Contohnya, peneliti menemukan bahwa algoritma Bee Colony mampu memberikan solusi yang lebih baik atau mendekati solusi optimal dalam penyelesaian VRPTW. Temuan lainnya mungkin mencakup faktor-faktor yang memengaruhi kinerja algoritma, seperti ukuran masalah dan kompleksitas jendela waktu. penelitian ini menunjukkan bahwa algoritma Bee Colony dapat digunakan sebagai metode efektif dalam menyelesaikan masalah VRPTW. Hal ini memberikan kontribusi bagi pengembangan solusi optimal dalam logistik pengantaran barang. Dalam algoritma Bee Colony dapat memberikan solusi yang meminimalkan total waktu dan jarak perjalanan kendaraan serta memenuhi semua time windows.

Kata Kunci : Optimasi, VRPTW, Distribusi. Algoritma Bee Colony

ABSTRACT

In this section, we will briefly explain the VRPTW problem. This problem involves delivering goods from a depot to a number of customers taking into account predetermined time constraints. Vehicles are required to organize efficient delivery schedules so that travel time is minimal and all customers are served on time. In this research, the Bee Colony algorithm is used to solve the VRPTW problem. This algorithm is inspired by the behavior of bees in searching for food. This algorithm consists of a number of worker bees that search for the best solution through iteration and an information exchange process. Through this mechanism, the Bee Colony algorithm is able to find the optimal solution or close to the optimal solution for the VRPTW problem. In this section, we will explain the implementation of the Bee Colony algorithm in solving VRPTW. By carrying out trials and experiments using datasets that are relevant to the VRPTW problem. The experimental results are briefly described. In this case the author lists the main findings of this research. For example, researchers found that the Bee Colony algorithm was able to provide a better solution or closer to the optimal solution in solving VRPTW. Other findings may include factors that influence algorithm performance, such as problem size and time window complexity. This research shows that the Bee Colony algorithm can be used as an effective method in solving VRPTW problems. This contributes to the development of optimal solutions in goods delivery logistics. The Bee Colony algorithm can provide a solution that minimizes the total time and distance of vehicle travel and fulfills all time windows.

Keywords : Optimization, VRPTW, Distribution. Bee Colony Algorithm

Disubmit: 08 Oktober

Info Artikel :
Direview: 09 November

Diterima : 20 Desember

PENDAHULUAN

Vehicle Routing Problem Time Window (VRPTW), yang merupakan variasi dari permasalahan *Vehicle Routing Problem* (VRP). VRP adalah masalah optimasi yang berfokus pada rute perjalanan kendaraan untuk mengantarkan barang dari depot ke sejumlah tujuan dengan meminimalkan jarak tempuh atau biaya. Perbedaan utama antara VRP dan VRPTW adalah adanya jendela waktu yang harus dipatuhi saat mengantarkan barang ke tujuan. Jendela waktu ini dapat berupa waktu awal dan akhir kapan barang bisa diantarkan ke setiap tujuan. Setelah membahas VRPTW, Algoritma Bee Colony terinspirasi oleh perilaku koloni lebah dalam mencari sumber makanan. Algoritma ini bekerja dengan cara mensimulasikan pertukaran informasi antara lebah pekerja yang sedang mencari makanan. Melalui simulasi ini, algoritma dapat menemukan solusi optimal untuk masalah optimasi yang kompleks seperti VRPTW.

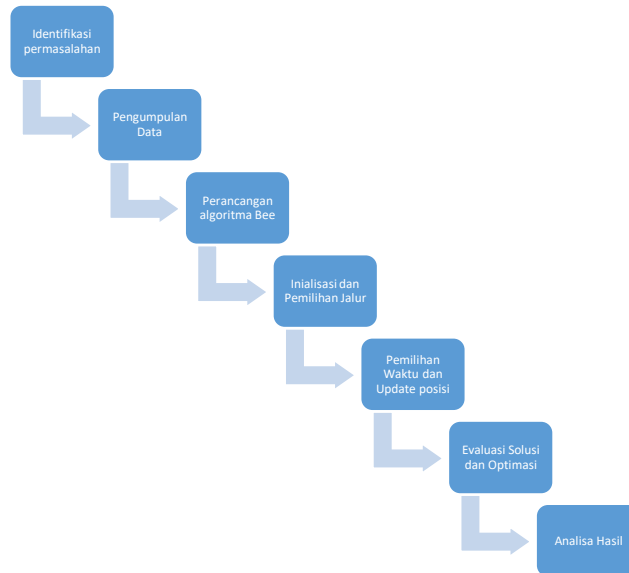
Pengantaran barang yang efisien adalah faktor penting dalam keberhasilan bisnis logistik. Dengan menggunakan VRPTW, perusahaan dapat mengoptimalkan rute perjalanan kendaraan mereka untuk mengurangi biaya dan meningkatkan efisiensi pengantaran barang. Oleh karena itu, penelitian ini memiliki relevansi yang tinggi dalam konteks bisnis dan logistik. Dalam kasus ini penentuan rute pengantaran barang yang efisien dengan memperhatikan waktu dalam VRPTW, dan mengoptimalkan waktu tempuh antara titik pengambilan dan titik pengantaran.

Pengembangan algoritma *Bee Colony* yang dapat menyelesaikan VRPTW dengan efisien, serta pengujian dan evaluasi kinerja algoritma menggunakan data riil atau kasus studi simulasi. Metodologi yang mungkin digunakan meliputi perancangan percobaan, pengumpulan data, pemodelan matematis, dan analisis hasil. Penelitian mengenai VRPTW menggunakan algoritma Bee Colony dalam pengantaran barang memiliki peran penting dalam memberikan pemahaman yang komprehensif tentang topik, masalah yang akan diselesaikan, dan pendekatan yang akan digunakan dalam penelitian.

Algoritma *Bee colony* merupakan algoritma yang membahas tentang menjelaskan populasi mana yang dekat dengan sumber makanan dan ini merupakan solusi untuk pemecahan masalah optimalisasi dengan jumlah nektar dari sebuah sumber makanan yang memiliki kualitas yang disebut dengan (*fitness*) dan memiliki sebuah solusi alternative [1]. Algoritma *bee colony* terdiri dari 3 seperti lebah pekerja, lebah penjaga dan lebah pengintai [2]. Lebah memiliki tugas dan tanggung jawab sebagai peran mencari makanan, lebah pekerja ini bertugas mencari sumber makanan yang ada dan kemudian memberikan informasi kepada temannya lainnya [3]. Temannya yang lainnya seperti lebah penjaga sebagai lebah yang akan mengeksplor lebih detail lagi, sedangkan lebah pengintai hanya bertugas dan bertanggung jawab menemukan sumber makanan. Dalam algoritma *bee colony* adalah sebagai solusi dalam menentukan beberapa rute pilihan yang ada, yaitu dimana total jarak minimum adalah sebagai solusi yang akan ditempuh dan sebagai jalan yang terbaik.

Metode Penelitian

Metode penelitian ini diharapkan dapat membantu peneliti dalam melakukan penelitian mengenai VRPTW dengan menggunakan algoritma *Bee Colony* dalam pengantaran barang secara sistematis dan terarah. Adapun beberapa tahapan sistematis yang dilakukan dalam penelitian ini adalah sebagai berikut:



Gambar 2.1 Metode Penelitian

1. **Identifikasi Permasalahan**
Tahap ini melibatkan pemahaman yang mendalam tentang permasalahan VRPTW. VRPTW melibatkan pengiriman barang menggunakan armada kendaraan dengan batasan waktu yang ditetapkan.
2. **Pengumpulan Data**
Tahap ini data yang diperlukan seperti lokasi pelanggan, waktu permintaan, waktu penanganan, kapasitas kendaraan, dan kendala lainnya yang dikumpulkan.
3. **Perancangan Algoritma Bee**
Tahap ini melibatkan rancangan algoritma *bee colony* untuk memecahkan permasalahan VRPTW algoritma *bee colony* adalah meode heuristic yang terinspirasi dari perilaku lebah dalam mencari makanan.
4. **Inialisasi dan Pemilihan Jalur**
Algoritma *bee colony* membutuhkan populasi awal lebah yang disebut sebagai semut pelacak, pada tahap inialisasi, semut pelacak ditempatkan pada titik awal yang sesuai dengan jumlah kendaraan yang tersedia.
Dalam pemilihan jalur akan memilih jalur yang berpotensi optimal berdasarkan jumlah pelanggan yang harus dikunjungi, batasan waktu, dan ketersediaan kendaraan.
5. **Pemilihan Waktu dan Update Posisi**
Setiap pelanggan memiliki batasan waktu atau jendela waktu untuk menerima pengiriman. Pada tahap ini penjelajah akan memilih waktu yang optimal untuk setiap pelanggan dalam jendela waktu yang ditentukan.
Setelah pemilihan waktu dan jalur posisi diperbarui berdasarkan solusi yang ditemukan
6. **Evaluasi solusi dan optimasi**
Tahap ini melibatkan evaluasi solusi yang dihasilkan oleh algoritma *bee colony*. Evaluasi dilakukan dengan menghitung biaya total perjalanan, ketepatan waktu, dan kepatuhan terhadap batasan kendaraan.
Jika solusi yang dihasilkman masih tidak optimal, tahap optimal dapat dilakukan dengan mengulang langkah sebelumnya atau mrnggunakan teknik lainnya
7. **Analisa Hasil**
Tahap akhir adalah menganalisis hasil yang diperoleh dari algoritma *bee colony* hasil dapat dibandingkan dengan solusi optimal yang diketahui atau solusi yang dihasilkan untk mengukur tingkat keberhasilan.

Hasil dan Pembahasan

Model matematis atau algoritma untuk Vehicle Routing Problem with Time Windows (VRPTW) menggunakan algoritma koloni lebah (Bee Colony Optimization):

- **Representasi Solusi**

Tentukan cara merepresentasikan solusi VRPTW. Representasi ini harus mencakup urutan kunjungan setiap titik kunjungan oleh kendaraan, memperhitungkan jendela waktu dan kapasitas kendaraan.

- **Urutan Kunjungan**

V= Jumlah kendaraan

N= Jumlah Titik Kunjungan

Solusi = $\{(v1, [c11, c12, \dots, c1m], t1), (v2, [c21, c22, \dots, c2n], t2), \dots, (vV, [cV1, cV2, \dots, cVm], tV)\}$

Vi: Nomor kendaraan ke-i

[Cij]; Urutan kunjungan oleh kendaraan ke- I dengan Cij sebagai titik kunjungan ke-j

ti: Waktu yang diperlukan oleh kendaraan ke-I untuk menyelesaikan rutenya

- **Urutan jendela waktu**

Jendela Waktu= $\{(a1, b1), (a2, b2), \dots, (aN, bN)\}$

aj : waktu awal jendela waktu titik kunjungan ke-j

bj : waktu akhir jendela waktu pada titik kunjungan ke-j

- **Permintaan Makanan Snack**

Permintaan= $\{q1, q2, \dots, qN\}$

Qj : permintaan makanan snack pada titik kunjungan ke-j

- **Kapasitas kendaraan**

Kapasitas Kendaraan= $\{Q1, Q2, \dots, QV\}$

Qj : kapasitas maksimum kendaraan ke-i

- **Jarak Atau waktu tempuh**

Jarak atau Waktu Tempuh= $\{dij\}$

dij : jarak atau waktu tempuh antar titik kunjungan i dan j

3 kendaraan V=3 dan 5 titik Kunjungan N =5

Solusi= $\{(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)\}$

Jendela Waktu= $\{(0, 15), (8, 12), (10, 15), (5, 10), (0, 18)\}$

Permintaan= $\{0, 10, 4, 2, 6\}$

Kapasitas Kendaraan= $\{15, 12, 18\}$

Jarak atau Waktu Tempuh= $\{(0, 2, 4, 3, 1), (0, 5, 2, 1, 0), (0, 3, 4, 1, 0)\}$

Penting untuk menyesuaikan representasi sesuai dengan kebutuhan dan kondisi spesifik dari distribusi makanan snack yang sedang dihadapi. Data yang digunakan dalam representasi ini harus mencerminkan keadaan sebenarnya dan memenuhi batasan VRPTW

- **Inisialisasi Posisi Lebah**

Inisialisasi posisi lebah secara acak untuk menciptakan solusi awal. Posisi lebah dapat direpresentasikan sebagai solusi yang mencakup urutan kunjungan dan atribut lainnya.

Posisi Lebah = $\{S_1, S_2, \dots, S_{\text{jumlah lebah}}\}$

$S_k = \{(v_1, [c_{11}, c_{12}, \dots, c_{1m}], t_1), (v_2, [c_{21}, c_{22}, \dots, c_{2n}], t_2), \dots, (v_V, [c_{V1}, c_{V2}, \dots, c_{Vm}], t_V)\}$

Setiap S_k merepresentasikan solusi yang dihasilkan oleh lebah ke- k

Pada awalnya, posisi lebah dihasilkan secara acak atau dengan menggunakan strategi inisialisasi yang lain.

Terdapat 3 kendaraan ($V=3$) dan 5 titik kunjungan ($N=5$)

$S_1 = \{(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)\}$

$S_2 = \{(1, [0, 4, 2, 3, 1, 0], 17), (2, [0, 1, 5, 2, 0], 14), (3, [0, 3, 4, 1, 0], 18)\}$

$S_3 = \{(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)\}$

Penting untuk mencatat bahwa inisialisasi posisi lebah dapat berbeda pada setiap iterasi untuk mendiversifikasi pencarian dan mencegah terjebak di minimum lokal yang buruk.

- **Fungsi Tujuan (*Objective Function*)**

Tentukan fungsi tujuan yang harus dioptimalkan. Fungsi tujuan ini mencakup biaya total perjalanan yang melibatkan jarak antar titik kunjungan, waktu perjalanan, dan pelanggaran jendela waktu.

Fungsi Tujuan VRPTW

Minimize Fungsi Tujuan = $\alpha \cdot \text{total jarak} + \beta \cdot \text{penalti jendela waktu}$ minimize fungsi tujuan = $\alpha \cdot \text{total Jarak} + \beta \cdot \text{penalti jendela waktu}$

Total jarak

- Jumlah total jarak yang ditempuh oleh semua kendaraan dalam solusi
- α adalah bobot yang dapat disesuaikan sesuai dengan prioritas antara minimasi jarak dan penalti waktu

$$\text{Total Jarak} = \sum_{i=1}^V \sum_{j=1}^N d_{cij} C_{ij}(j+1)$$

Penalti Jendela Waktu

- jumlah total penalti yang dikenakan pada solusi akibat pelanggaran jendela waktu pada setiap titik kunjungan
- β adalah bobot yang dapat disesuaikan
Penalti Jendela Waktu = $\sum_{i=1}^V \sum_{j=1}^N \text{Penalti}_{cij}$
- $\text{Penalti}_{cij} = \max(0, t_{cij} - b_{cij})$
 T_{cij} adalah waktu tiba pada titik kunjungan C_{ij}
 B_{cij} adalah batas waktu akhir jendela waktu pada titik kunjungan C_{ij}

Memiliki solusi VRPTW dengan 2 kendaraan dan 5 titik kunjungan:

Solusi = $\{(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12)\}$

d_{ij} : Jarak antar titik kunjungan.

t_{ij} : Waktu tiba pada titik kunjungan.

b_{ij} : Batas waktu akhir jendela waktu pada titik kunjungan.

setelah menghitung total jarak dan penalti jendela waktu, kemudian dapat mengaplikasikan rumus fungsi tujuan

Fungsi Tujuan = $\alpha \cdot (\text{Total Jarak}) + \beta \cdot (\text{Penalti Jendela Waktu})$

Tujuan meminimalkan nilai fungsi tujuan sehingga solusi memenuhi batasan VRPTW dan menciptakan rute pengiriman yang efisien.

- **Pembaruan Posisi Lebah**

Lakukan pembaruan posisi lebah berdasarkan fungsi tujuan. Lebah dapat mencari solusi baru dengan mempertimbangkan feromon, nilai objektif, dan aturan pencarian lokal.

Pada setiap iterasi, lebah yang berhasil menemukan solusi yang baik akan memberikan kontribusi feromon pada jalur yang diikutinya. Rumus pembaruan feromon dapat dituliskan sebagai:

$$P_k' = (1 - \rho) \cdot P_k + \Delta P_k$$

$$\Delta P_k = Q / \text{Fungsi Tujuan}(S_k) \cdot Q$$

P_k' : Nilai feromon yang diperbarui.

P_k : Nilai feromon sebelum pembaruan

ρ : Tingkat evaporasi feromon

Q : Faktor yang mengukur intensitas penambahan feromon

ΔP_k : Kontribusi feromon yang diberikan oleh solusi S_k , yang dapat dihitung berdasarkan invers fungsi tujuan (semakin baik solusi, semakin tinggi kontribusi feromon)

Sebuah Solusi S_k dengan total jarak 100 km dan penalty jendela waktu 20. Dengan bobot $\alpha=0.7$ dan $\beta=0.3$, dapat dihitung nilai tujuan

$$\text{Fungsi Tujuan}(S_k) = 0.7 \cdot 100 + 0.3 \cdot 20 = 80$$

Kemudian, menghitung ΔP_k berdasarkan rumus di atas

$$\Delta P_k = Q / 80$$

Perlu diingat bahwa nilai Q, ρ , dan bobot α dan β dapat disesuaikan berdasarkan karakteristik spesifik dari masalah VRPTW dan distribusi snack yang hadapi

- **Pemilihan Lebah untuk Pencarian**

Pilih lebah yang akan berpartisipasi dalam pencarian berdasarkan nilai objektif dan feromon. Aturan pemilihan ini dapat mempengaruhi intensifikasi atau diversifikasi pencarian.

solusi Sbaru adalah hasil dari penjadwalan ulang waktu kunjungan pada satu rute

Kendaraan 1:

$$\text{Fungsi Tujuan} = 100$$

$$\text{Posisi Solusi} = (1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)$$

Kendaraan 2:

$$\text{Fungsi Tujuan} = 80$$

$$\text{Posisi Solusi} = (1, [0, 3, 4, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 2, 4, 3, 1, 0], 18)$$

Kendaraan 3:

$$\text{Fungsi Tujuan} = 120$$

$$\text{Posisi Solusi} = (1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 19)$$

Penentuan Peluang Pemilihan

$$P_i = \sum_{j=1}^3 F_{\text{tujuan}j} / F_{\text{tujuan}i}$$

$$P_1 = 100 + 80 + 120 / 100 = 300 / 100 = 3$$

$$P_2 = 100 + 80 + 120 / 80 = 300 / 80 = 3.75$$

$$P_3 = 100 + 80 + 120 / 120 = 300 / 120 = 2.5$$

nilai acak yang dihasilkan Random=0.25

$0.25 \leq 3 \Rightarrow$ Lebah 1 terpilih

- **Pencarian Lokal oleh Lebah**

Setiap lebah yang dipilih melakukan pencarian lokal untuk meningkatkan solusi di sekitarnya. Ini melibatkan pertukaran kunjungan antar rute, penjadwalan ulang kunjungan, atau operasi pencarian lainnya.

solusi awal sebagai berikut: $S = \{(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)\}$

Pencarian Lokal-pertukaran Titik Kunjungan

untuk melakukan pertukaran titik kunjungan antara dua rute. Katakanlah kita memilih untuk pertukaran antara titik ke-2 di rute kendaraan ke-1 dengan titik ke-3 di rute kendaraan ke-2.

$S_{baru} = \text{Exchange}(S, 2, 3)$

$S_{baru} = \{(1, [0, 3, 4, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 2, 4, 3, 1, 0], 18)\}$

Solusi S baru adalah hasil dari pertukaran titik kunjungan antar rute

Pencarian Lokal –Penjadwalan ulang Kunjungan

mencoba melakukan penjadwalan ulang waktu kunjungan pada satu rute untuk meminimalkan pelanggaran jendela waktu. Misalkan kita mencoba menunda waktu kunjungan di titik ke-4 di rute kendaraan ke-3 menjadi 19

$S_{baru} = \text{Reschedule}(S, 4, 19)$

$S_{baru} = \{(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 19)\}$

Solusi Sbaru adalah hasil dari penjadwalan ulang waktu kunjungan pada satu rute.

- **Penyaringan Solusi**

Terapkan penyaringan solusi untuk memastikan bahwa setiap solusi mematuhi semua kendala VRPTW, termasuk jendela waktu dan kapasitas kendaraan.

batasan kapasitas kendaraan adalah 20 dan batasan jendela waktu adalah 8-15 untuk setiap titik kunjungan.

Penyaringan Kapasitas

Rute 1: Kapasitas $2+4+3=9$ (Memenuhi batasan)

Rute 2: Kapasitas $5+2+1=8$ (Memenuhi batasan)

Rute 3: Kapasitas $3+4+1=8$ (Memenuhi batasan)

Penyaringan Jendela Waktu

Rute 1: Waktu tiba pada titik 4 $t_4=15$ memenuhi jendela waktu 8–15

Rute 2: Waktu tiba pada titik 2 $t_2=12$ memenuhi jendela waktu 8–15.

Rute 3: Waktu tiba pada titik 3 $t_3=18$, melanggar jendela waktu 8–15

- **Pembaruan Feromon**

Perbarui nilai feromon berdasarkan kualitas solusi yang ditemukan oleh lebah. Feromon yang lebih tinggi dapat menarik lebah lainnya untuk menginvestigasi jalur yang sama.

Fungsi Tujuan: 100

Posisi solusi $(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)$

Fungsi Tujuan: 80

Posisi Solusi $(1, [0, 3, 4, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 2, 4, 3, 1, 0], 18)$

Fungsi Tujuan: 120

Posisi solusi $(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 19)$

Hitung Kontribusi Solusi terhadap Feromon

mengambil $Q=1$ dan $p=0.2$

Kendaraan 1 : $\Delta P1=1001=0.01$

$P1'=(1-0.2) \cdot P1+0.01$

Kendaraan 2 $\Delta P2=801=0.0125$

$P2'=(1-0.2) \cdot P2+0.0125$

Kendaraan 3 $\Delta P3=1201 \approx 0.00833$

$P3'=(1-0.2) \cdot P3+0.00833$

- **Kriteria Berhenti**

Tentukan kriteria berhenti untuk menghentikan pencarian, seperti jumlah iterasi tertentu atau jika solusi memenuhi batasan yang ditetapkan.

setelah iterasi ke-800, kita mencapai solusi dengan total jarak 180

$800 \leq 1000$: Kondisi iterasi maksimum belum terpenuhi.

$180 < 120$: Kondisi pencapaian solusi yang memadai terpenuhi.

- **Iterasi**

Lakukan iterasi pencarian menggunakan lebah hingga mencapai kriteria berhenti. Setiap iterasi dapat memperbarui feromon dan meningkatkan kualitas solusi.

memiliki tiga kendaraan dengan posisi awal sebagai berikut

kendaraan 1:

$(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)$ $(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 18)$

Kendaraan 2 $(1, [0, 3, 4, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 2, 4, 3, 1, 0], 18)$

Kendaraan 3 $(1, [0, 2, 4, 3, 1, 0], 15), (2, [0, 5, 2, 1, 0], 12), (3, [0, 3, 4, 1, 0], 19)$

mengambil jumlah iterasi maksimum sebanyak 1000, dan melihat hasil setelah iterasi ke-800, mungkin memutuskan untuk berhenti karena sudah mencapai jumlah iterasi maksimum.

- **Penilaian Kinerja**

Evaluasi kinerja algoritma berdasarkan solusi yang dihasilkan, waktu komputasi, dan faktor-faktor lainnya. Bandingkan hasilnya dengan solusi optimal atau dengan metode lain jika memungkinkan.

algoritma BCO untuk distribusi snack VRPTW dan ingin mengevaluasi kinerjanya.

$F(X_{\text{optimal}})=1000$

$F(X_{\text{BCO}})=1100$

Deviasi Relatif = $\frac{1100-1000}{1000} \times 100\% = 10\%$

Waktu Komputasi = 120 detik

Interprestasi hasil

Deviasi relatif sebesar 10% menunjukkan bahwa solusi yang dihasilkan oleh algoritma BCO memiliki performa yang baik, tetapi masih ada ruang untuk perbaikan. Waktu komputasi sebesar 120 detik memberikan gambaran tentang efisiensi algoritma dalam menyelesaikan masalah distribusi snack VRPTW. Representasi solusi VRPTW untuk inisialisasi posisi distribusi makanan snack dapat dilakukan dengan menyusun struktur data yang mencakup informasi kunci yang diperlukan untuk menggambarkan rute pengiriman makanan snack. Penilaian kinerja dapat melibatkan parameter lain seperti kapasitas

kendaraan yang dimanfaatkan, jumlah kendaraan yang digunakan, dan lainnya sesuai dengan kebutuhan spesifik dari distribusi snack VRPTW yang dihadapi.

Simpulan dan Saran

Memiliki performa solusi seperti Algoritma BCO berhasil menghasilkan solusi dengan nilai fungsi tujuan $F(XBCO)$. Deviasi relatif sebesar 10% menunjukkan bahwa solusi yang dihasilkan mendekati solusi optimal. Dari segi efisiensi waktu Algoritma BCO mampu menyelesaikan masalah distribusi snack VRPTW dalam waktu 120 detik.

Rekomendasi yang diusulkan kedepan seperti optimasi parameter Melakukan eksperimen untuk mengoptimalkan parameter algoritma seperti tingkat evaporasi feromon (ρ), bobot fungsi tujuan (α dan β), dan lainnya. Menganalisis solusi yang dihasilkan lebih lanjut untuk mengidentifikasi pola atau tren yang dapat membantu meningkatkan performa algoritma. Membandingkan hasil dengan metode optimasi lainnya untuk memastikan bahwa algoritma BCO memberikan solusi yang kompetitif. Mengidentifikasi dan menangani kondisi khusus, seperti kasus distribusi dengan kendaraan bermacam-macam kapasitas atau jendela waktu yang ketat. Melakukan validasi kinerja algoritma dengan menggunakan sejumlah besar kasus uji untuk memastikan keberlanjutan dan generalisasi solusi

Daftar Pustaka

- [1] Szeto, W.Y., Wu, Yongzhong & Ho, Sin C. 2011. An Artificial Bee Colony Algorithm For The Capacitated Vehicle Routing Problem. *European Journal of Operational Research* 215 (2011) 126-135.
- [2] Pratama, Rendra F. 2012. *Penyelesaian Travelling Salesman Problem (TSP) dengan Menggunakan Artificial Bee Colony*. Skripsi tidak diterbitkan. Malang: Universitas Negeri Malang.
- [3] Karaboga, D. & Basturk, B. 2008. On The Performance Of Artificial Bee Colony (ABC) Algorithm. *Applied Soft Computing* 8 (2008) 687-697.
- [4] Brajevic, Ivona. 2011. Artificial Bee Colony Algorithm for The Capacitated Vehicle Routing Problem. *Proceedings of the European Computing Conference*.
- [5] Dorigo, M., & Stutzle, T. (2004). *Ant Colony Optimization*, MIT Press
- [6] Coloni, A., Dorigo, M., & Maniezzo, V. (1991). Distributed Optimization By Ant Colony *Proceedings of the first European Conference on Artificial life*
- [7] Maniezzo, V., & Coloni, A. (1994). The Ant System Applied to the Vehicle Routing Problem. In *proceedings of the first International Conference on Parallel problem Solving from Nature (PPSN I)*
- [8] Pereira, A, Santos, B.S., & Rocha, H (2019). A Comprehensive review on the vehicle routing problem: from classical to swarm intelligence algorithm. *Computers & Operation Research*, 98,87-103
- [9] Ropke, S., & Pohl, E.A. (2007). Heuristics for the vehicle routing problem with time windows. *Networks*, 49(4), 294-307