

METAHEURISTICS UNTUK MENYELESAIKAN PERMASALAHAN VEHICLE ROUTING PROBLEM: PARTIAL COMPARISON OPTIMIZATION

¹Enty Nur Hayati, ²Antono Adhi, ³Lie Liana, ⁴Evitriani, ⁵Retno Maulana Sari

^{1,2,3}Program Studi Teknik Industri, Universitas Stikubank (UNISBANK) Semarang

enty@edu.unisbank.ac.id

antonoadhi@edu.unisbank.ac.id

lieliana@edu.unisbank.ac.id

evitrianievi401@gmail.com

retnomaulana45@gmail.com

Abstrak

Vehicle Routing Problem (VRP) adalah pencarian rute terpendek dari sebuah depot dengan mempertimbangkan kapasitas dari alat transportasi. VRP adalah varian dari *Traveling Salesman Problem* (TSP).

Dalam penelitian ini, metode *metaheuristics Partial Comparison Optimization* (PCO) dikembangkan untuk menyelesaikan masalah *combinatorial optimization*. Dalam penelitian ini, PCO diuji untuk dapat menyelesaikan permasalahan yang ada pada VRP. Beberapa data contoh dari TSPLIB digunakan untuk mengevaluasi kinerja PCO.

Hasil numerik yang diperoleh menunjukkan bahwa metode optimasi yang diusulkan dapat mencapai hasil yang dekat dengan solusi terbaik yang diketahui.

Kata kunci: *Combinatorial Optimization, Metaheuristics, Partial Comparison Optimization, Vehicle Routing Problem*

Abstract

Vehicle Routing Problem (VRP) is a problem of searching shortest route from origin point and back to the same point by considering transportation capacity. VRP is variant of *Traveling Salesman Problem* (TSP).

In this research a metaheuristics method called *Partial Comparison Optimization* (PCO) was developed to search close optimal solution in order to solve *combinatorial optimization* such as TSP. Some data instance from TSPLIB was used to evaluate the performance of PCO.

The numerical result showed that this proposed optimisation algorithm could achieve near optimum solution.

Key words : *Combinatorial Optimization, Metaheuristics, Partial Comparison Optimization, Vehicle Routing Problem*

I. PENDAHULUAN

Logistik adalah salah satu bagian yang penting dalam perusahaan. Salah satu tugas dari logistik adalah mengirim produk dari sebuah perusahaan yang disebut depot ke pelanggan yang memesan produk tersebut. Meskipun logistik bukanlah bagian yang memberikan nilai tambah secara langsung pada sebuah produk, tetapi ketidakmampuan mengelola logistik dapat menyebabkan biaya tinggi yang mengganggu daya saing perusahaan. Oleh karena itu pengelolaan pengiriman barang dari depot perusahaan ke pelanggan harus diatur dan dilakukan secara efektif dan efisien. Efektif dalam arti semua permintaan pelanggan dipenuhi sepenuhnya sehingga produk sampai ke tangan pelanggan dan efisien berarti biaya-biaya yang timbul dari proses logistik tersebut dapat ditekan sekecil mungkin. Salah satu aktivitas yang mampu untuk melakukan hal ini adalah mengatur jalur transportasi pengiriman dari depot ke pelanggan dan dari pelanggan satu ke pelanggan lainnya dengan biaya yang sekecil-kecilnya. Aktivitas ini dikenal dengan sebutan *Vehicle Routing Problem* (VRP).

VRP adalah salah satu kasus *combinatorial optimization* (Fajarwati and Anggraeni, 2012). Dia adalah masalah klasik dari bidang logistik dan transportasi yang berkaitan dengan perencanaan rute untuk kendaraan yang dimulai dari depot pusat dan pergi ke satu set pelanggan (Zhang, *et.al.*, 2017). Braysy dan Gendreau (2001) menyatakan bahwa VRP adalah permasalahan pencarian rute dengan biaya terkecil dari satu depot ke sejumlah titik yang tersebar dalam area geografis yang berupa kota, toko, gudang, sekolah, pelanggan, dan sebagainya. Rute didesain setiap titik hanya akan dilewati 1 kali oleh 1 kendaraan yang dimulai dan diakhiri pada suatu depot dan total permintaan dari sebuah rute tidak melebihi dari kapasitas kendaraan.

Dalam sistem logistik, transportasi memainkan peran penting untuk membawa bahan mentah dari pemasok ke produsen dan mengirimkan barang jadi dari produsen ke pelanggan (Yu, *et.al.*, 2017). Di AS, transportasi itu sendiri mencakup 28% dari total konsumsi energi, dan dengan demikian biaya transportasi secara signifikan memengaruhi harga akhir barang (Farahani,*et.al.*, 2011). Karena jangkauan aplikasinya yang luas baik dalam entitas komersial maupun publik, VRP dianggap sebagai salah satu masalah yang paling penting dalam riset operasional.

VRP pertama kali diperkenalkan oleh Dantzig dan Ramser (1956).Mereka meneliti tentang pengiriman bahan bakar dari depot ke stasiun bahan bakar.Setelah penelitian yang mereka lakukan, banyak peneliti yang menyusul melakukan penelitian tentang VRP.Perbaikan-perbaikan pada masalah VRP dilakukan untuk dapat menyelesaikan permasalahan ini lebih baik lagi (Zhang,*et.al.*, 2017). Kondisi di lapangan yang ada juga memunculkan varian baru yang membutuhkan teknik-teknik penanganan yang berbeda pula. Varian-varian dari VRP yang muncul dan diperkenalkan sebagian besar masih terkait dengan penambahan kendala baru dari kasus VRP yang semula (Zhang,*et.al.*, 2017). Kendala yang sering ditambahkan sebagai varian dalam penyelesaian masalah VRP adalah palet dan *window time* (Zhang,*et.al.*, 2017). Dalam batasan palet, barang dengan ukuran berbeda harus diangkut dalam kotak ukuran standar dan kapasitas terbatas (Zhang,*et.al.*, 2017). Dalam batasan *window time*, barang harus dikirim selama jangka waktu tertentu (Zhang,*et.al.*, 2017).

VRP adalah bagian dari Traveling Salesman Problem (TSP) (Sandhya dan Kumar, 2013) namun dengan mempertimbangkan kapasitas alat transportasi. Jika TSP akan memenuhi semua jalur dari titik awal hingga kembali ke titik awal lagi, maka VRP akan memenuhi permintaan pelanggan dengan jalur tertentu dan akan kembali ke titik awal atau depot lagi untuk mengambil barang jika kapasitas alat transportasi sudah tidak dapat memenuhi permintaan pelanggan lagi.

VRP dan variannya digolongkan ke dalam *NP-Hard Problem* (Lau, *et.al.*,2003; Ghani, *et.al.*, 2016; Csiszar, 2005; Yu, *et.al.*,2017) karena secara teori ataupun praktik pada dunia nyata memiliki permasalahan yang sangat banyak dan kompleks sehingga sulit untukdipecahkan dengan metode eksak. Kasus *NP-Hard* dapat diselesaikan dengan menggunakan pendekatan solusi optimal dengan metode heuristik yang memberikan perkiraan solusi dengan waktu penyelesaian yang masih dapat diterima.Oleh karen itu, penyelesaian VRP dapat dilakukan oleh metode *heuristics* atau *metaheuristics* (Yakici, 2017).

Dalam ilmu komputer dan optimisasi matematis, metaheuristik adalah prosedur atau heuristik tingkat tinggi yang dirancang untuk menemukan,

menghasilkan, atau memilih heuristik (algoritma pencarian parsial) yang dapat memberikan solusi yang cukup baik untuk masalah optimasi, terutama dengan informasi yang tidak lengkap atau tidak sempurna. atau kapasitas komputasi terbatas (Balamurugan, *et.al.*, 2015; Bianchi, *et.al.*, 2009). Metaheuristik mengambil sampel serangkaian solusi yang terlalu besar untuk dijadikan sampel sepenuhnya. Metaheuristik mungkin membuat beberapa asumsi tentang masalah optimisasi yang diselesaikan, sehingga mereka dapat digunakan untuk berbagai masalah (Blum *and* Roli, 2003).

Dibandingkan dengan algoritma optimasi dan metode berulang, metaheuristik tidak menjamin bahwa solusi optimal secara global dapat ditemukan pada beberapa kelas masalah (Blum *and* Roli, 2003). Banyak metaheuristik menerapkan beberapa bentuk optimasi stokastik, sehingga solusi yang ditemukan tergantung pada set variabel acak yang dihasilkan (Bianchi, *et.al.*, 2009). Dalam optimasi kombinatorial, dengan mencari lebih dari seperangkat solusi yang layak, metaheuristik sering kali dapat menemukan solusi yang baik dengan upaya komputasi yang lebih sedikit daripada algoritma optimasi, metode berulang, atau heuristik sederhana (Blum dan Roli, 2003). Karena itu, mereka adalah pendekatan yang berguna untuk masalah optimasi (Bianchi, *et.al.*, 2009). Beberapa buku dan makalah survei telah diterbitkan tentang masalah ini (Bianchi, *et.al.*, 2009; Blum *and* Roli, 2003; Goldberg, 1989; Glover *and* Kochenberger, 2003; Talbi, 2009).

Kebanyakan literatur tentang metaheuristik bersifat eksperimental, menggambarkan hasil empiris berdasarkan percobaan komputer dengan algoritma. Tetapi beberapa hasil teoritis formal juga tersedia, seringkali pada konvergensi dan kemungkinan menemukan optimum global (Blum *and* Roli, 2003). Banyak metode metaheuristik telah diterbitkan dengan klaim kebaruan dan kemanjuran praktis. Sementara bidang ini juga menampilkan penelitian berkualitas tinggi, banyak publikasi memiliki kualitas yang buruk; kelemahan termasuk ketidakjelasan, kurangnya elaborasi konseptual, eksperimen yang buruk, dan ketidaktahuan literatur sebelumnya (Sörensen, 2013).

Penelitian tentang VRP masih banyak dilakukan. Banyak di antaranya yang diselesaikan dengan metode *metaheuristics* seperti pada Tabel 1.1. Metode heuristik yang paling umum digunakan untuk menyelesaikan permasalahan NP-hard termasuk pada algoritma evolusioner yang diinspirasi dari proses yang terjadi di alam seperti *genetic algorithm*, *particle swarm optimization*, *differential evolution*, *and colony optimization*, dan sebagainya. Masing-masing algoritma diujikan dan dibandingkan untuk mencari algoritma mana yang paling baik dan sesuai untuk diterapkan dalam rangka mengatasi suatu permasalahan yang bersifat NP-hard seperti VRP. Dalam penelitian yang akan dilakukan, akan dikembangkan metode baru yang akan digunakan untuk mengatasi permasalahan dalam VRP dan memberikan solusi yang optimum. Metode ini bernama *Partial Comparison Optimization* (PCO).

Tabel 1.1 Studi VRP yang diselesaikan dengan *metaheuristics*.

| Peneliti | Algoritma |
|---|--|
| Yu, <i>et. al.</i> (2017) | <i>simulated annealing</i> |
| Costa, <i>et. al.</i> (2018), Mohammed, <i>et. al.</i> (2017) | <i>genetic algorithm</i> |
| Baniamerian, <i>et. al.</i> (2019) | <i>genetic algorithm, variable neighborhood search</i> |
| Peng, <i>et al.</i> (2017) | <i>particle swarm optimization</i> |

| | |
|--|--|
| Bula, <i>et al.</i> (2017), Huber, <i>et. al.</i> (2017), Polat(2017) Euchi (2017) | <i>variable neighborhood search</i> <i>ejections chains neighborhood</i> <i>combinatorial neighborhood</i> <i>topology glowworm swarm</i> |
| Marinaki, <i>et. al.</i> (2016) | <i>optimization, variable</i> <i>neighborhood search, path</i> <i>relinking</i> |
| Ng, <i>et. al.</i> (2017) | <i>bee colony</i> |
| Silvestrin, <i>et. al.</i> (2017) | <i>tabu search</i> |
| Yakici (2017) | <i>ant colony</i> |
| Yu, <i>et. al.</i> (2017) | <i>symbiotic organisms search</i> |

PERMASALAHAN

Menurut Laporte (1992), VRP dapat diformulasikan sebagai berikut:
Jika x_{ij} ($i \neq j$) adalah variabel biner sama dengan 1 jika dan hanya jika terdapat *arc* (i, j) dari A'

$$\text{Minimize } \sum_{i \neq j} c_{ij} x_{ij} \dots\dots\dots(1)$$

s.t.

$$\sum_{j=1}^{n'} x_{ij} = 1, \quad (i = 1, \dots, n') \dots\dots\dots(2)$$

$$\sum_{i=1}^{n'} x_{ij} = 1, \quad (j = 1, \dots, n') \dots\dots\dots(3)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - v(S), \quad (S \subset V \setminus \{1\}; |S| > 2) \dots\dots\dots(4)$$

$$x_{ij} \in \{0,1\}, \quad (i, j = 1, \dots, n'; i \neq j) \dots\dots\dots(5)$$

(1), (2), dan (3) mendefinisikan penugasan. Jarak pada diagonal tidak diperbolehkan. Batasan (4) adalah hambatan eliminasi *subtour*: $v(S)$ adalah batas bawah pada jumlah kendaraan yang diperlukan untuk mengunjungi semua simpul S dalam solusi optimal. Kendala-kendala ini diperoleh dengan mengamati bahwa untuk setiap $S \subset V \setminus \{1\}; |S| > 2, S = V \setminus V$, kita harus memiliki:

$$\sum_{i \in S} \sum_{j \in S} x_{ij} > v(S), \quad (i \neq j) \dots\dots\dots(6)$$

dan bahwa identitas berikut ini berlaku:

$$|S| = \sum_{i,j \in S} x_{ij} + \sum_{i \in S} \sum_{j \in S} x_{ij}, \quad (i \neq j) \dots\dots(7)$$

Nilai $v(S)$ tergantung pada jenis VRP yang sedang dipertimbangkan. Dalam *Capacitated VRP* (CVRP), nilai yang diambil:

$$v(S) = \left\lceil \frac{\sum_{i \in S} d_i}{D} \right\rceil \dots\dots\dots(8)$$

Yaitu merupakan nilai jumlah muatan yang dibawah oleh kendaraan berdasarkan kapasitas kendaraan D .

II. TELAAH PUSTAKA

Algoritma *metaheuristics* baru yang dikembangkan untuk menyelesaikan permasalahan *combinatorial optimization* adalah *partial comparison optimization* (PCO). Algoritma ini dinamakan demikian karena langkah dasar pencarian titik optimum adalah dilakukan secara parsial hingga lengkap dari suatu deret permutasi. PCO adalah metode metaheuristik baru. PCO digunakan untuk menyelesaikan permasalahan optimasi dengan mencari solusi terbaik dari alternatif-alternatif yang dipertimbangkan untuk memecahkan permasalahan. PCO akan memecahkan permasalahan *combinatorial optimization* yang menggabungkan urutan elemen. Solusi terbaik yang dihasilkan oleh PCO adalah solusi optimal seperti biaya minimum, keuntungan maksimum, atau waktu penyelesaian tercepat. Kendala permasalahan optimisasi yang menjadi permasalahan utama penyelesaian *metaheuristics* cukup dapat dihindari dengan baik oleh PCO.

PCO cukup efektif untuk menemukan nilai optimum. PCO tidak akan terjebak dalam optimum lokal dan lebih memungkinkan untuk memperoleh nilai global optimal. Hal ini disebabkan karena proses pengarahannya dalam proses PCO ada dalam area nilai kombinasi saja. Dengan proses pengacakan, pergeseran radikal dari kombinasi akan dimungkinkan. Proses ini akan memberikan kemungkinan hasil baru yang mungkin lebih optimal. Namun, proses pengacakan ini masih dikendalikan oleh pedoman yang hanya memilih hasil yang lebih optimal di area kombinasi. Meskipun untuk kasus yang sangat kompleks waktu yang dibutuhkan PCO untuk menyelesaikan satu iterasi dapat lebih lama dari algoritma lainnya, namun waktu yang dibutuhkan untuk mencapai nilai optimal relatif tidak terlalu lama karena PCO relatif lebih cepat mencapai global optimal. PCO memiliki kondisi fakultatif mendapatkan nilai optimal dalam iterasi yang lebih cepat. Namun, opsi ini memiliki konsekuensi waktu pemrosesan yang lebih lama.

III. METODE PENELITIAN

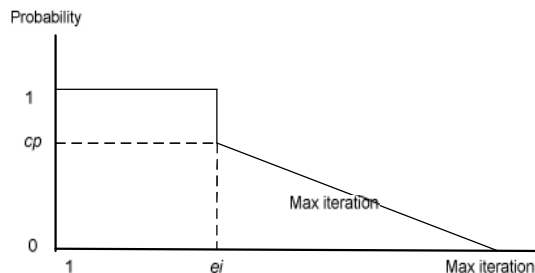
PCO memiliki pedoman dasar yang menjadi prinsip pencarian nilai optimal yaitu:

1. *Random Choosing* (RC)
RC adalah pemilihan elemen dalam urutan secara acak. Elemen dalam hal ini adalah kota atau tempat yang menjadi tujuan pengiriman pada TSP. Pemilihan acak ini memiliki tujuan untuk memperluas area penemuan titik optimum.
2. *Partial Comparison* (PC)
PC adalah pemilihan posisi suatu elemen dalam suatu urutan. Posisi yang memiliki nilai *fitness* terbaik akan dipilih. Proses ini dilakukan seperti pada algoritma NEH (Nawaz, *et al.*, 1983).
3. *Changing Neighborhood* (CN)
CN adalah usaha untuk menukar posisi elemen tetangga dari posisi suatu elemen yang akan ditempatkan. Jika penukaran akan memberikan nilai *fitness* yang lebih baik, maka pada posisi baru elemen tersebut akan dilakukan penukaran elemen tetangga.
4. *Looping Process* (LP)

Langkah LP akan mengulangi pencarian nilai optimum secara iteratif. Perolehan nilai *fitness* yang lebih optimum untuk setiap iterasi dimungkinkan tercapai karena PCO memiliki langkah RC. Dari langkah LP, akan diperoleh global optimum dengan membandingkan nilai global optimum dengan lokal optimum.

5. *Stopping Iteration* (SI)

PCO biasanya memperoleh solusi optimum di iterasi-iterasi awal. Pada mulanya, PC dan CN diproses dengan nilai probabilitas 1 hingga iterasi efektif *ei*. *ei* ditentukan sebagai nilai iterasi yang biasanya sudah mencapai nilai optimum. Untuk mengurangi waktu proses namun juga masih diinginkannya kemungkinan diperolehnya nilai yang lebih baik, PC dan CN akan dilakukan pada posisi tertentu pada elemen baru jika nilai acak yang dibangkitkan di bawah probabilitas perbandingan *cp* setelah iterasi ke *ei*. Nilai *cp* akan ditentukan antara 0 dan 1. Nilai *cp* akan dikurangi setelah *ei* hingga iterasi maksimum seperti yang diperlihatkan pada Gambar 3.1.



Gambar 3.1. Iterasi efektif *ei* dan probabilitas perbandingan *cp* menentukan diprosesnya PC dan CN.

Algoritma PCO adalah sebagai berikut:

- Step 1. Tentukan $iteration = 1$, $GlobalFitness = \max integer$, elemen-elemen yang akan diproses $J = \{1, 2, 3, \dots, j\}$, iterasi maksimum $MaxItr$, iterasi efektif *ei*, dan probabilitas perbandingan maksimum *cp*.
- Step 2. Hapus *K*. Ambil satu elemen dari *J* dan masukkan *K*
- Step 3. Ambil satu elemen dari *J*
- Step 4. Jika iterasi $> ei$ maka $prob = ce - (ce \times ((iteration - ei) / (MaxItr - ei)))$ lainnya $prob = 1$
- Step 5. Proses perhitungan posisi elemen jika bilangan random $rand < prob$. Letakkan dan coba posisi elemen baru tersebut pada sebelum, antara, dan setelah dari deret elemen-elemen yang ada di *K*. Hitung nilai *fitness* dari urutan yang baru dari sebuah posisi yang ditempati oleh elemen baru tersebut. Jika urutan dengan elemen baru menempati sebuah posisi mempunyai nilai *fitness* yang lebih baik, set urutan ini sebagai urutan baru sebagai urutan parsial terbaik untuk elemen tersebut.
Untuk setiap posisi yang ditempati elemen baru, proses langkah berikut jika $rand < prob$. Jika posisi elemen baru berada di antara urutan elemen-elemen di *K*, tukar posisi dari dua elemen yang berada di antara elemen baru tersebut dan hitung *fitness* dari urutan baru dengan elemen baru. Jika urutan baru memiliki nilai *fitness* yang lebih baik, tentukan urutan baru sebagai urutan parsial terbaik untuk posisi elemen baru tersebut.
Lakukan Step 5 hingga seluruh posisi pada *K* ditempati oleh elemen baru.

Step 6. Lakukan Step 3 hingga semua elemen dalam J ditempatkan pada K .

K adalah urutan lokal terbaik untuk iterasi tersebut.

Step 7 Jika $fitness(K) < GlobalFitness$ maka K menjadi urutan global terbaik baru.

Step 8. Tambahkan 1 untuk $iteration$. Lakukan Step 2 dan ulang perhitungan hingga $iteration = MaxItr$. Urutan global terbaik adalah solusi yang diberikan PCO yang akan memberikan nilai optimum.

IV. DATA, HASIL DAN PEMBAHASAN

Untuk membuktikan bahwa PCO adalah algoritma yang baik untuk menyelesaikan VRP, sejumlah contoh data VRP simetrik dari VRPLIB digunakan untuk mengevaluasi kinerja PCO. 6 contoh data kota berskala normal diambil dari contoh VRPLIB seperti yang ditunjukkan pada Tabel 2.

Data-data dari VRPLIB hanya data koordinat untuk setiap kota. Data jarak antar kota tidak disediakan. Oleh karena itu solusi jarak optimal yang diberikan oleh VRPLIB dihitung kembali berdasarkan urutan rute optimum yang ditunjukkan oleh VRPLIB.

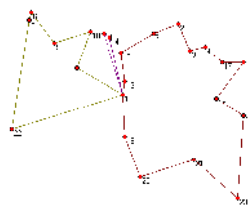
Tabel 3.1 Hasil perhitungan VRP oleh PCO.

| VRP LIB | Best-known Solution | Min | Max | SD (%) |
|---------|---------------------|--------|---------|--------|
| eil22 | 375.28 | 375.28 | 1162.43 | 0.00 |
| eil23 | 568.56 | 568.55 | 1917.67 | 0.00 |
| eil30 | 497.53 | 508.13 | 2521.56 | 0.52 |
| eil33 | 826.41 | 845.81 | 2346.39 | 1.28 |
| eil51 | 524.61 | 572.31 | 2338.82 | 2.63 |
| eilA76 | 823.89 | 989.25 | 3559 | 6.05 |

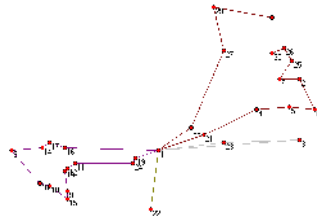
Rute dari VRPLIB ditunjukkan seperti pada Gambar 4.1 hingga 4.6.



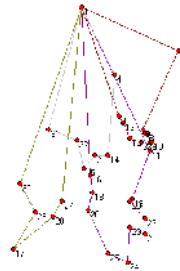
Gambar 4.1. Rute eil22.



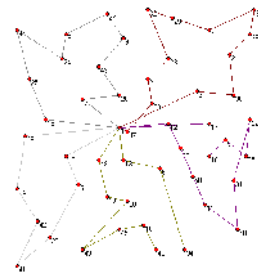
Gambar 4.2. Rute eil23.



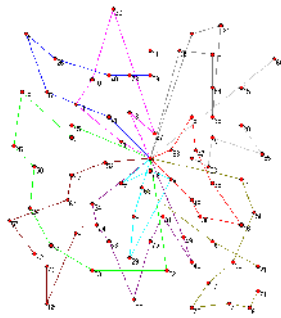
Gambar 4.3. Rute eil30.



Gambar 4.4. Rute eil33.



Gambar 4.5. Rute eil51.



Gambar 4.6. Rute eilA76.

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil pengembangan aplikasi dan implementasi algoritma PCO untuk menyelesaikan VRP, dapat disimpulkan sebagai berikut bahwa algoritma PCO mempunyai kinerja yang baik untuk menyelesaikan proses permasalahan optimisasi pada VRP. Kinerja PCO yang baik tersebut dihasilkan dari nilai *fitness* yang diperoleh dan kecepatan waktu proses untuk menghasilkan nilai optimum.

Dari penelitian yang telah dikembangkan untuk mengimplementasikan algoritma PCO pada permasalahan VRP, diperoleh agar algoritma PCO memungkinkan dipergunakan untuk menyelesaikan permasalahan optimisasi

diskrit lain. Algoritma PCO dapat dinilai tingkat efektifnya untuk menyelesaikan permasalahan optimisasi jika dapat dibandingkan dengan algoritma *meta-heuristic* lainnya.

DAFTAR PUSTAKA

- Balamurugan R., et.al., (2015)**, "*Stellar-Mass Black Hole Optimization for Biclustering Microarray Gene Expression Data*,". Applied Artificial Intelligence an International Journal.taylor& francis. 29 (4): 353–381.
- Baniamerian, et.al., (2019)**, "*Modified variable neighborhood search and genetic algorithm for profitable heterogeneous vehicle routing problem with cross-docking*", *Applied Soft Computing*, 75, p. 441-460.
- Bianchi, L.,et.al.,(2009)**."A survey on metaheuristics for stochastic combinatorial optimization". *Natural Computing: an international journal*. 8 (2): 239–287.
- Blum, C.,and Roli, A., (2003)**, "*Metaheuristics in combinatorial optimization: Overview and conceptual comparison*" (PDF). 35 (3). *ACM Computing Surveys*: 268–308.
- Braysy, O., and Gendreau, M., (2001)**, "*Genetic Algorithms for the Vehicle Routing Problem with Time Windows*", *Arpakannus*, 1, p.33-38.
- Bula, et.al.,(2017)**, "*Variable neighborhood search to solve the vehicle routing problem for hazardous materials transportation*", *Journal of Hazardous Materials*, 324, p. 472–480.
- Csiszár, S., (2005)**, "*Route Elimination Heuristic for Vehicle Routing Problem with Time Windows*", *Acta Polytechnica Hungarica*, 2, pp. 77–89.
- Dantzig, G.B., and Ramser, J.H., (1959)**, "*The truck dispatching problem*", *Manage. Sci.* 6 (1), p. 80–91.
- Dorigo, M., (1992)**,*Optimization, Learning and Natural Algorithms*, PhD "thesis", Politecnico di Milano, Italie.
- Euchi, J., (2017)**, "*The vehicle routing problem with private fleet and multiple common carriers: Solution with hybrid metaheuristic algorithm*", *Vehicular Communications*, 9, p. 97–108.
- Farahani, et.al., (2011)**,"*Logistics Operations and Management:Concepts and Models*", Elsevier.
- Ghani, et.al.,(2016)**, "*An Alternative Algorithm for Vehicle Routing Problem with Time Windows for Daily Deliveries*", *Advances in Pure Mathematics*, 6, 342-350.
- Glover, F., and Kochenberger, G.A. (2003)**,*Handbook of metaheuristics*.57. Springer, "*International Series in Operations Research and Management Science*", ISBN 978-1-4020-7263-5.
- Goldberg, D.E. (1989)**,"*Genetic Algorithms in Search, Optimization and Machine Learning*", Kluwer Academic Publishers. ISBN 0-201-15767-5.
- Huber, et.al., (2017)**, "*Order matters – A Variable Neighborhood Search for the Swap-Body Vehicle Routing Problem*", *European Journal of Operational Research*, 263, p. 419–445.
- Laporte, G., (1992)**, "*The Vehicle Routing Problem: An overview of exact and approximate algorithms*", *European Journal of Operational Research*, 59, p. 345-358.

-
- Lau, et.al., (2003)**, “*Vehicle routing problem with time windows and a limited number of vehicles*”, European Journal of Operational Research, 148, pp. 559–569.
- Marinaki, et.al., (2016)**, “*A Glowworm Swarm Optimization algorithm for the Vehicle Routing Problem with Stochastic Demands*”, Expert Systems With Applications, 46, p. 145–163.
- Moscato, P., (1989)**, “*On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms*” (PDF). Caltech Concurrent Computation Program (report 826).
- Ng, et.al.,(2017)**, “*A multiple colonies artificial bee colony algorithm for a capacitated vehicle*”
- Peng, Z., et.al., (2017)**, “*Particle Swarm Optimization for Capacitated Location-Routing Problem*”, IFAC-PapersOnLine, 50, p. 14668-1467.
- Polat, O., (2017)**, “*A parallel variable neighborhood search for the vehicle routing problem with divisible deliveries and pickups*”, Computers and Operations Research, 85, 71–86.
- Sandhya, et.al.,(2013)**, “*Issues in Solving Vehicle Routing Problem with Time Window and its Variants using Meta heuristics - A Survey*”, International Journal of Engineering and Technology, 3, 668–672.
- Silvestrin, et.al., (2017)**, “*An iterated tabu search for the multi-compartment vehicle routing problem*”, Computers and Operations Research, 81, p. 192–202.
- Sörensen, K., (2013)**, “*Metaheuristics—the metaphor exposed*” (PDF).International Transactions in Operational Research. doi:10.1111/itor.12001.
- Talbi, E-G. (2009)**, “*Metaheuristics: from design to implementation*”, Wiley. ISBN 0-470-27858-7.
- Toth, V., (2002)**, “*Vehicle Routing Problem*”, Philadelphia, SIAM Monographs on Discrete Mathematics and Application.
- Yakici, E., (2017)**, “*A heuristic approach for solving a rich min-max vehicle routing problem with mixed fleet and mixed demand*”, Computers & Industrial Engineering, 109, p. 288–294.
- Yu, et.al., (2017)**, “*Symbiotic organisms search and two solution representations for solving the capacitated vehicle routing problem*”, Applied Soft Computing, 52, p. 657–672.
- Zhang, et.al.,(2017)**, “*A hybrid algorithm for a vehicle routing problem with realistic constraints*”, Information Sciences, 394–395, 167–182.

STUDI PENGARUH PENGENDALI ATC TERHADAP KOMPONEN TEKNOLOGI DALAM SISTEM ATC

¹Tranggono, ²Rusindiyanto

^{1,2}Prodi Teknik Industri dan Universitas Pembangunan Nasional “Veteran” Jawa Timur
tranggono.ti@upnjatim.ac.id
rusindiyanto.ti@upnjatim.ac.id

Abstrak

Pengendali *Air Traffic Control* (ATC) merupakan salah satu profesi dengan tingkat keahlian yang khusus dan kompleks. Hal ini dikarenakan seorang pengendali ATC memiliki aktivitas yang melibatkan pola pikir, komunikasi, pengambilan keputusan, interaksi dan hubungan antara manusia-komputer yang rentan terhadap error. Dan lagi memiliki resiko yang sangat tinggi karena berhubungan dengan keselamatan banyak orang. Tugas dari seorang pengendali ATC adalah mempertahankan jarak aman antar pesawat yang memiliki variasi dengan kepadatan lalu lintas dan wilayah udara.

Penelitian ini bertujuan untuk mengetahui hubungan antara pengendali ATC (*humanware*) dengan komponen teknologi dalam sistem ATC antara lain: Organisasi (*orgaware*), perangkat lunak (*infoware*) dan peralatan/mesin (*technoware*) dari peralatan ATC. Untuk mengidentifikasi dan mengetahui hubungan antar variabel, dipergunakan metode *Partial Least Square* (PLS).

Berdasarkan nilai *t*-statistic, diperoleh faktor yang signifikan berpengaruh menunjukkan adanya hubungan antara pengendali ATC (*humanware*) dengan komponen sistem yang lain (*infoware*, *orgaware*, dan *technoware*).

Kata Kunci: *Air Traffic Control, Techno-metric, Partial Least Square (PLS).*

Abstract

Air Traffic Controller is one profession with special and complex level of expertise. This is because an ATC controller has activities that involve mindset, communication, decision making, interaction and relationships between human-computers that are vulnerable to errors. And it has a very high risk because it deals with the safety of many people. The task of an ATC controller is to maintain a safe distance between planes that have variations with the density of traffic and airspace.

This study aims to determine the relationship between ATC (humanware) controllers and technological components in ATC systems, among others: Organization (orgaware), software (infoware) and equipment / machinery (technoware) of ATC equipment. To identify and know the relationship between variables, with Partial Least Square (PLS).

Based on the t-statistical value, obtained significant influential factors indicating the existence of a relationship between the ATC controller (humanware) and other system components (infoware, orgaware, and technoware).

Keywords: *Air Traffic Control, Techno-metric, Partial Least Square (PLS).*

I. PENDAHULUAN

Penerapan teknologi dalam sistem ATC merupakan faktor penting dimana kecepatan dan ketepatan sangat diutamakan dalam pekerjaan sebagai pengendali ATC. Teknologi memiliki empat komponen yang terpadu secara dinamis. Antara lain: 1. *Humanware* (perangkat manusia); 2. *Technoware* (perangkat keras/peralatan); 3. *Infoware* (perangkat informasi); dan 4. *Orgaware* (perangkat organisasi). Sebagai bentuk keterpaduan keempat komponen tersebut, dimana untuk pengembangan dan pengendalian komponen *technoware* dilakukan oleh komponen *humanware*, melalui *informasi* dari *infoware*, dan secara keseluruhannya diatur oleh *orgaware* (Pailin, 2013). *Human error* sebagai salah satu faktor yang berkontribusi dalam kecelakaan pesawat terbang yang disebabkan ATC. Hal ini dikarenakan aktivitas para pengendali ATC (*humanware*)