

**PENENTUAN RUTE OPTIMAL PADA DISTRIBUSI BARANG
MENGUNAKAN ALGORITMA *GREEDY*
(Studi Kasus: UD XYZ)**

**Ardy Januanto^{1,*}, Abdullah Faqih Septiyanto², Andrey Kartika Widhy
Hapantenda³**

^{1,3}Teknik Informatika, Fakultas Teknik, Universitas 17 Agustus 1945

²Teknik Informatika, Fakultas Teknologi Elektro dan Infomatika Cerdas, Institut
Teknologi Sepuluh Nopember

Email: ardyjanuanto@untag-sby.ac.id

ABSTRACT

The distribution of goods is an important thing that must consider in the continuity of a business. The better the distribution of goods will have an impact on increasing customer satisfaction. UD XYZ is a distributor of motorcycle masks in Surabaya. The distribution route, which is carried out every day, does not have a fixed route selection reference and is constantly changing without a clear basis. The distribution process often experiences delays in the delivery of goods. For this reason, optimization is needed in determining the optimal distribution route to save time, costs, and other operations. A greedy algorithm is one of the algorithms used to determine the optimal solution. The greedy algorithm is an algorithm that follows a problem-solving heuristic making a locally optimal choice at each stage in the hope of finding a global optimal. With the implementation of the greedy algorithm, this research produces an optimal distance and can provide benefits to UD XYZ.

Keywords: Distribution of goods, optimization, distribution routes, greedy algorithm

ABSTRAK

Distribusi barang merupakan hal penting yang harus diperhatikan pada keberlangsungan sebuah bisnis. Semakin baik distribusi barang maka akan berdampak terhadap meningkatnya kepuasan pelanggan. UD XYZ merupakan salah satu distributor masker motor di Surabaya. Rute distribusi yang setiap hari dilakukan tidak memiliki acuan pemilihan rute yang tetap dan selalu berubah – ubah tanpa ada dasar yang jelas, sehingga proses distribusi sering mengalami kendala keterlambatan pengiriman barang. Untuk itu diperlukan optimasi dalam menentukan rute distribusi yang optimal agar lebih menghemat waktu, biaya dan operasional lainnya. Algoritma *greedy* merupakan salah satu algoritma yang digunakan untuk menentukan solusi optimal. Algoritma *greedy* adalah algoritma yang mengikuti heuristik pemecahan masalah membuat pilihan optimal lokal pada setiap tahap dengan harapan menemukan optimal global. Pada penelitian ini, pemilihan lokal digambarkan dengan *node – node* yang diterapkan sebagai lokasi distribusi barang kemudian memilih jalur terbaik antar lokasi distribusi. Dengan implementasi algoritma *greedy*, penelitian ini menghasilkan jarak yang optimal dan mampu memberikan keuntungan pada UD XYZ.

Kata Kunci: Distribusi barang, optimasi, rute distribusi, algoritma *greedy*

1. Pendahuluan

Distribusi barang dalam konsep manajemen secara umum dikenal dengan sebutan logistik, yaitu proses

penyediaan barang berdasarkan dari tersedianya barang pada penyimpanan sampai dengan proses pengiriman. Proses distribusi barang dimulai dari

distributor kemudian dialokasikan menuju retail, grosir maupun pengecer [1]. UD XYZ adalah sebuah usaha dagang dan juga merupakan salah satu distributor di Surabaya yang bergerak di bidang distributor masker motor. Dalam penerapannya, UD XYZ sering mengalami aduan dari pelanggan karena keterlambatan distribusi. Alasan keterlambatan tersebut disebabkan karena pengambilan rute distribusi hanya melalui asumsi *driver* saja. Untuk meminimalisir aduan dari pelanggan, diperlukan sebuah metode penentuan rute agar dapat mengoptimalkan distribusi barang tersebut.

Dari beberapa metode algoritma yang ada, penelitian ini menggunakan algoritma *greedy* sebagai penentuan rute optimal, karena algoritma *greedy* lebih optimal jika titik tujuan kurang dari 10 titik [2]. Titik tujuan yang akan dijadikan rute distribusi barang adalah 8 titik. Titik – titik tersebut tersebar di berbagai daerah di Surabaya. Algoritma *greedy* mampu menunjukkan beberapa pilihan rute yang ditempuh, dan yang nantinya akan dipilih adalah hasil yang paling kecil yang berarti bahwa jarak yang ditempuh lebih pendek.

Dengan menggunakan algoritma *greedy*, diharapkan mampu mengoptimalkan rute pada proses distribusi barang di UD.XYZ untuk memperbaiki layanan ke pelanggan. selain itu, hasil optimal dari rute distribusi barang dapat meningkatkan keuntungan UD.XYZ karena dapat melakukan penekanan biaya dan waktu tempuh.

2. Tinjauan Pustaka

Terdapat beberapa penelitian terdahulu dengan kasus yang sama dengan melakukan penerapan seperti algoritma genetika untuk pencarian

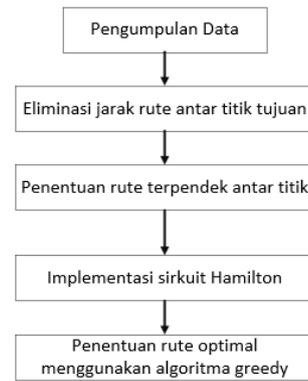
rute terpendek pada dinas kebersihan [3] dan juga penentuan rute distribusi pos [4]. Penelitian lain yang dilakukan oleh Hasnah [5] yaitu menggunakan algoritma *heuristic* untuk optimalisasi distribusi barang dan juga penelitian lain sama – sama optimasi tetapi dengan tujuan yang berbeda yaitu metode optimasi pada layanan purna jual [6]. Optimasi lain juga terdapat pada penelitian Karina [1] dengan menggunakan algoritma *branch and bound*.

Pada penelitian ini, penulis menggunakan algoritma *greedy* sebagai metode untuk menentukan solusi optimum rute. Optimum rute pada penelitian ini adalah rute terpendek. Algoritma *greedy* adalah algoritma yang mengikuti heuristik pemecahan masalah membuat pilihan optimal lokal pada setiap tahap dengan harapan menemukan optimal global [7]–[11]. Dalam banyak masalah, strategi *greedy* secara umum tidak menghasilkan solusi optimal, tetapi heuristik *greedy* dapat menghasilkan solusi optimal lokal yang mendekati solusi optimal global dalam waktu yang wajar [12]. Seperti pada penelitian yang dilakukan Donny [13] menggunakan algoritma *greedy* untuk penentuan rute di museum Yogyakarta dan juga penentuan distribusi PMI yang dilakukan oleh Mahendra [14].

Jika algoritma *greedy* dapat memecahkan masalah, maka secara umum itu menjadi metode terbaik untuk menyelesaikan masalah itu karena algoritma *greedy* secara umum lebih efisien daripada teknik lain seperti pemrograman dinamis. Tapi algoritma *greedy* tidak selalu bisa diterapkan. Misalnya, masalah *Fractional Knapsack* dapat diselesaikan menggunakan *greedy*, tetapi 0-1 *knapsack* tidak dapat diselesaikan menggunakan *greedy* [12]

3. Metode

Terdapat beberapa langkah yang digunakan pada penelitian ini meliputi pengumpulan data, eliminasi jarak rute antar titik tujuan, penentuan rute terpendek antar titik, implementasi sirkuit *hamilton*, dan penentuan rute optimal menggunakan algoritma *greedy*, seperti disajikan pada Gambar 1.



Gambar 1. Alur metodologi penelitian

Tabel 1. Matriks Jarak Antar Titik Distribusi

| | UD.XYZ | R1 | R2 | R3 | R4 | R5 | R6 | R7 |
|--------|--------|------|------|------|-------|------|------|-------|
| UD.XYZ | 0 | 600m | 920m | 880m | 1200m | 650m | 550m | 469m |
| R1 | 600m | 0 | 670m | 890m | 860m | 910m | 750m | 880m |
| R2 | 1300m | 780m | 0 | 650m | 788m | 865m | 897m | 886m |
| R3 | 860m | 890m | 932m | 0 | 976m | 677m | 658m | 1032m |
| R4 | 1200m | 869m | 800m | 975m | 0 | 994m | 760m | 750m |
| R5 | 650m | 900m | 845m | 677m | 882m | 0 | 876m | 455m |
| R6 | 650m | 612m | 671m | 991m | 701m | 876m | 0 | 987m |
| R7 | 500m | 833m | 921m | 592m | 699m | 796m | 907m | 0 |

3.1. Pengumpulan Data

Pengumpulan data dilakukan dengan melakukan wawancara terhadap pihak UD.XYZ. Data yang dikumpulkan adalah data titik distribusi. Untuk penentuan jarak, peneliti menggunakan google maps sebagai acuan jarak tempuh antar titik distribusi dengan satuan meter (m). Dari hasil tersebut diperoleh data matriks jarak seperti pada Tabel 1.

3.2. Eliminasi Jarak Rute Antar Titik Tujuan

Eliminasi jarak rute antar titik tujuan dilakukan karena mengacu pada algoritma *greedy*, dimana sifat dari algoritma *greedy* yaitu mencari nilai seminimal mungkin untuk mendapatkan nilai optimal [15]. Pada Tabel 1 dapat dilihat terdapat perbedaan jarak antara rute berangkat dan rute kembali, hal itu dikarenakan terdapat jalan satu arah yang mengakibatkan jalur rute berangkat

tidak sama dengan jalur rute kembali. Maka dari itu proses penyederhanaan rute dilakukan dengan mengeliminasi rute yang jauh dan mengambil rute yang dekat.

3.3. Penentuan Rute Terpendek Antar Titik

Setelah dilakukan eliminasi jarak yang jauh antar rute, proses selanjutnya yaitu melakukan penentuan rute terpendek dari tiap titik. Penentuan rute tersebut berdasarkan hasil eliminasi pada proses sebelumnya.

3.4. Implementasi sirkuit *Hamilton*.

Pada proses ini penentuan rute tiap titik di implementasikan dengan gambar sirkuit *Hamilton*. Model sirkuit *Hamilton* adalah sebuah aliran rute antar titik yang melewati tiap titik tepat satu kali kecuali untuk titik awal karena nantinya titik awal tersebut nantinya menjadi titik akhir [16]. Sirkuit *Hamilton* digunakan untuk

menggambarkan beberapa alternatif alur rute dari awal di UD XYZ sampai Kembali lagi ke titik awal yaitu di UD XYZ.

3.6. Penentuan Rute Optimal Menggunakan Algoritma Greedy

Penentuan rute optimal menggunakan algoritma *greedy* yaitu dengan membandingkan beberapa alternatif rute dengan jarak yang sudah diketahui sesuai implementasi sirkuit *Hamilton*. Hasil alternatif rute tersebut dapat disebut sebagai solusi optimum lokal. Setelah penentuan solusi optimum lokal, selanjutnya yaitu menentukan jarak terpendek dari semua kemungkinan alternatif solusi yang disebut solusi optimum global. Implementasi algoritma *greedy* untuk pemilihan rute optimal dapat dijelaskan pada notasi *pseudocode* pada Gambar 2.

```

Deklarasi
  int J[n] //alternatif solusi
  lokal
  int hasil //jarak rute terpendek
hasil = 0
For i=1 to n do
  if (hasil < J[i]) then
    hasil = J[i]
  endif
end
    
```

Gambar 2. *Pseudocode* optimasi menggunakan algoritma *greedy*

Dari Gambar 2 dapat dilihat bagaimana algoritma *greedy* mengambil kemungkinan jarak terpendek dari beberapa alternatif solusi yang ada di lambangkan dalam notasi $J[n]$, dimana n adalah jumlah alternatif solusi. Kemudian setiap alternatif solusi tersebut akan dibandingkan dengan alternatif lain yang akan menghasilkan solusi optimum global yaitu jarak rute terpendek.

4. Hasil Dan Pembahasan

Bab Hasil dan Pembahasan berisi tentang hasil dari penelitian/pengujian yang telah dilakukan.

4.1. Penyederhanaan rute

Hasil dari penyederhanaan rute dapat dilihat pada Tabel 2.

Pada Tabel 2 dapat dilihat terdapat beberapa rute yang di eliminasi karena memiliki jarak tempuh rute yang panjang. Selain itu poses eliminasi tidak berlaku pada panjang rute yang sama seperti rute dari UD XYZ menuju titik R1 yang memiliki jarak yang sama dengan rute dari R1 menuju UD XYZ maka semua rute tersebut di terima.

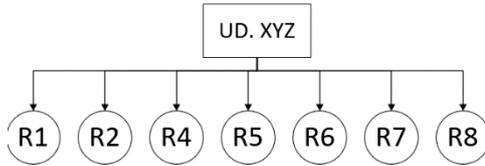
4.2. Penentuan Rute Optimal dari Tiap Titik

Dari hasil proses eliminasi rute pada Tabel 2, pada proses ini

Tabel 2. Matriks jarak setelah proses penyederhanaan rute

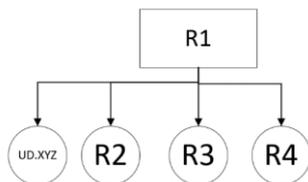
| | UD.XYZ | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 |
|--------|--------|------|------|------|-------|------|------|------|------|
| UD.XYZ | 0 | 600m | 920m | - | 1200m | 650m | 550m | 469m | 498m |
| R1 | 600m | 0 | 670m | 890m | 860m | - | - | - | - |
| R2 | - | - | 0 | 650m | 788m | - | - | 886m | - |
| R3 | 860m | 890m | - | 0 | - | 677m | 658m | - | - |
| R4 | 1200m | - | - | 975m | 0 | - | - | - | 605m |
| R5 | 650m | 900m | 845m | 677m | 882m | 0 | 876m | 455m | - |
| R6 | - | 612m | 671m | - | 701m | 876m | 0 | - | 450m |
| R7 | - | 833m | - | 592m | 699m | - | 907m | 0 | - |
| R8 | 498m | 810m | 663m | 655m | - | 589m | - | 450m | 0 |

penentuan rute optimal digambarkan dalam sebuah graf sesuai hasil dari Tabel 2. Untuk lebih jelasnya bisa dilihat pada Gambar 3.



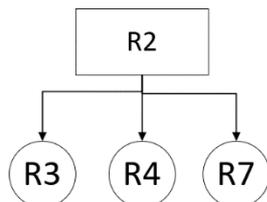
Gambar 3. Rute optimal dari UD XYZ

Dari Gambar 3 dapat dilihat bahwa terdapat tujuh alternatif titik tujuan yang dipilih berdasarkan jarak terpendek yaitu dari titik awal menuju R1 dengan jarak 600m, menuju R2 dengan jarak 920m, menuju R4 dengan jarak 1200m, menuju R5 dengan jarak 650m, menuju R6 dengan jarak 550m. Menuju R7 dengan jarak 469m, dan juga menuju R8 dengan jarak 498m.



Gambar 4. Rute optimal dari titik R1

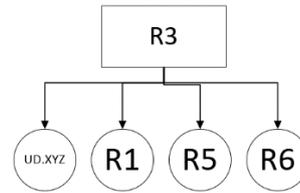
Pada Gambar 4 terdapat empat rute optimal dari titik R1 yaitu menuju titik UD XYZ dengan jarak 600m, menuju titik R2 dengan jarak 670m, menuju titik R3 dengan jarak 890m, dan menuju titik R4 dengan jarak 860m



Gambar 5. Rute optimal dari titik R2

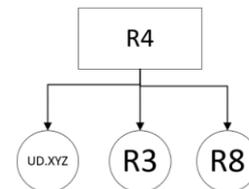
Pada Gambar 5 terdapat tiga titik terpendek dimulai dari titik R2 menuju

R3 dengan jarak 650m, menuju titik R4 dengan jarak 788m, dan menuju titik R7 dengan jarak 886m.



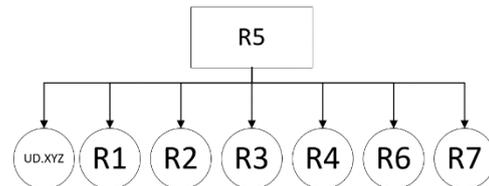
Gambar 6. Rute optimal dari titik R3

Pada Gambar 6 terdapat empat rute optimal yaitu dari titik R3 menuju UD.XYZ dengan jarak 860m, menuju titik R1 dengan jarak 890m, menuju titik R5 dengan jarak 677m, dan juga menuju titik R6 dengan jarak 658m.



Gambar 7. Rute optimal dari titik R4

Pada Gambar 7 terdapat tiga rute optimal yaitu dari titik R4 yaitu menuju titik pada UD XYZ dengan jarak 1200m, menuju titik R3 dengan jarak 975m, dan menuju titik R8 dengan jarak 605m.



Gambar 8. Rute optimal dari titik R5

Pada Gambar 8 terdapat tujuh rute jarak optimal yang dimulai dari titik R5 menuju UD XYZ dengan jarak 650m, menuju titik R1 dengan jarak 900m, menuju titik R2 dengan jarak 845m, menuju titik R3 dengan jarak 677m, menuju titik R4 dengan jarak 882m, menuju titik R6 dengan jarak

menandakan bahwa rute tersebut diterima karena kembali ke titik awal.

4.4. Penentuan Rute Optimal Menggunakan Algoritma Greedy

Setelah melakukan percobaan kemungkinan rute yang dilalui, maka langkah selanjutnya adalah penentuan rute terpendek. Dari 30 alternatif rute, pada penelitian ini peneliti menampilkan 18 kemungkinan rute jarak terbaik. Alternatif tersebut dapat dilihat pada Tabel 3.

Tabel 3. Rute Alternatif

| No. | Rute alternatif | Jarak |
|-----|--|--------------|
| 1 | UD.XYZ-R5-R1-R3-R6-R2-R7-R4-R8- UD.XYZ | 6457m |
| 2 | UD.XYZ-R5-R1-R3-R6-R8-R2-R7-R4- UD.XYZ | 6996m |
| 3 | UD.XYZ-R5-R1-R4-R8-R7-R6-R2-R3- UD.XYZ | 6553m |
| 4 | UD.XYZ-R2-R3-R6-R4-R8-R7-R1- UD.XYZ | 5992m |
| 5 | UD.XYZ-R5-R4-R3-R6-R6-R8-R7-R1- UD.XYZ | 6531m |
| 6 | UD.XYZ-R5-R2-R4-R8-R7-R3-R6-R1- UD.XYZ | 5800m |
| 7 | UD.XYZ-R5-R2-R4-R8-R7-R6-R1-R3-UD.XYZ | 6607m |
| 8 | UD.XYZ-R5-R2-R7-R1-R3-R6-R4-R8-UD.XYZ | 6566m |
| 9 | UD.XYZ-R5-R2-R7-R3-R6-R1-R4-R8-UD.XYZ | 6206m |
| 10 | UD.XYZ-R5-R2-R7-R3-R6-R4-R8-R1-UD.XYZ | 6347m |
| 11 | UD.XYZ-R5-R2-R7-R3-R6-R8-R1-R4-UD.XYZ | 6301m |
| 12 | UD.XYZ-R5-R2-R7-R4-R3-R6-R8-R1-UD.XYZ | 6573m |
| 13 | UD.XYZ-R5-R2-R7-R4-R8-R3-R6-R1-UD.XYZ | 5560m |
| 14 | UD.XYZ-R5-R2-R7-R6-R1-R4-R8-R3-UD.XYZ | 6880m |
| 15 | UD.XYZ-R5-R2-R7-R6-R4-R8-R1-R3-UD.XYZ | 7154m |
| 16 | UD.XYZ-R5-R2-R7-R6-R4-R8-R3-R1-UD.XYZ | 6739m |
| 17 | UD.XYZ-R5-R2-R7-R6-R8-R1-R4-R3-UD.XYZ | 7243m |
| 18 | UD.XYZ-R5-R2-R7-R6-R8-R3-R1-R4-UD.XYZ | 7343m |

Dari Tabel 3 dapat diketahui bahwa rute yang paling optimal yaitu rute dengan urutan **UD.XYZ - R5 - R2 - R7 - R4 - R8 - R3 - R6 - R1 - UD.XYZ** dengan panjang total jarak perjalanan yaitu **5560** dalam satuan meter.

5. Penutup

Algoritma *greedy* memungkinkan kita memilih hasil minimum dari beberapa alternatif solusi lokal. Dalam hal ini hasil minimum yang dimaksud adalah jarak rute terpendek yaitu pada rute alternatif nomor 13 dengan panjang rute **5560 meter**. Kelebihan dari algoritma *greedy* adalah kecepatan dalam menentukan rute optimal karena langsung mencoba semua peluang yang ada. Adapun kelemahan dari algoritma *greedy* adalah membutuhkan waktu yang sangat lama apabila titik – titik sebagai alternatif rute lebih banyak, karena pengecekan jarak antar titik dilakukan dengan mengecek satu per satu jarak tempuh antar titik. Dengan demikian algoritma *greedy* lebih cocok untuk jumlah alternatif titik yang sedikit sedangkan untuk titik yang lebih besar tidak cocok.

Dalam penelitian ini variabel yang digunakan hanya berupa jarak tempuh antar titik. Untuk alternatif lain seperti waktu, efisiensi bahan bakar dan kemacetan tidak dihitung. Kedepannya penelitian ini akan menambahkan beberapa variabel pendukung agar dapat digunakan dalam kondisi yang bervariasi.

6. Daftar Pustaka

- [1] K. Auliasari, M. Kertaningtyas, and D. W. L. Basuki, "Optimalisasi Rute Distribusi Produk Menggunakan Metode Traveling Salesman Problem," *Jurnal Sains, Teknologi dan*

- Industri*, vol. 16, no. 1, 2018.
- [2] D. T. Wiyanti, "Algoritma Optimasi Untuk Penyelesaian Travelling Salesman Problem," *Jurnal Transformatika*, vol. 11, no. 1, 2013.
- [3] M. D. A. Cipta Hasibuan and L. -, "Pencarian Rute Terbaik Pada Travelling Salesman Problem (TSP) Menggunakan Algoritma Genetika pada Dinas Kebersihan dan Pertamanan Kota Pekanbaru," *SATIN - Sains dan Teknologi Informasi*, vol. 1, no. 1, 2016.
- [4] S. Rohman, L. Zakaria, A. Asmiati, and A. Nuryaman, "Optimisasi Travelling Salesman Problem dengan Algoritma Genetika pada Kasus Pendistribusian Barang PT. Pos Indonesia di Kota Bandar Lampung," *Jurnal Matematika Integratif*, vol. 16, no. 1, p. 61, 2020.
- [5] Mutia Hasanah, Nazaruddin Matondang, and Aulia Ishak, "Penentuan Rute Distribusi Barang yang Optimal dengan Menggunakan Algoritma Heuristik pada PT. XYZ," *Jurnal Teknik Industri USU*, vol. 3, no. 3, 2013.
- [6] A. Januanto and R. Sarno, "Repair and replacement strategy for optimizing cost and time of warranty process using integer programming," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 16, no. 6, 2018.
- [7] H. A. Abdulkarim and I. F. Alshammari, "Comparison of Algorithms for Solving Traveling Salesman Problem," *International Journal of Engineering and Advanced Technology*, vol. 4, no. 6, pp. 76–79, 2015.
- [8] M. F. A. Sahputra, R. N. Devita, S. A. Siregar, and K. C. Kirana, "Implementation of Traveling Salesman Problem (TSP) based on Dijkstra ' s Algorithm in Logistics System," *International Journal of Electrical and Electronics Engineering*, vol. 14, no. 1, pp. 39–44, 2016.
- [9] B. C. Geiger and G. Böcherer, "Greedy algorithms for optimal distribution approximation," *Entropy*, vol. 18, no. 7, 2016.
- [10] M. Z. Usman and T. Oktiarso, "Implementasi Algoritma Greedy untuk Menyelesaikan Travelling Salesman Problem di Distributor PT . Z," *Journal of Integrated System*, vol. 1, no. 2, pp. 216–229, 2018.
- [11] H. Fahmi, M. Zarlis, E. B. Nababan, and P. Sihombing, "Implementation of the Greedy Algorithm to determine the nearest route Search in distributing food production," *IOP Conference Series: Materials Science and Engineering*, vol. 769, no. 1, pp. 0–7, 2020.
- [12] S. Ejim and R. N. D, "Implementation of Greedy Algorithm in Travel Salesman Problem," no. September, pp. 1–8, 2012.
- [13] D. L. Ichsan, L. D. N. Reni, and P. K. Abiyasa, "Menentukan jalur terpendek dalam mengunjungi museum di yogyakarta dengan traveling salesman problem (tsp)," pp. 32–37, 1800.
- [14] Y. D. Mahendra, N. Nuryanto, and A. Burhanuddin, "Sistem Penentuan Jarak Terdekat dalam Pengiriman Darah di PMI Kota Semarang dengan Metode Algoritma Greedy," *Jurnal Komtika*, vol. 2, no. 2, 2019.
- [15] Y. Liu, E. K. P. Chong, A. Pezeshki, and Z. Zhang,

“Submodular optimization problems and greedy strategies: A survey,” *Discrete Event Dynamic Systems: Theory and Applications*, vol. 30, no. 3, 2020.

- [16] A. Kosasih, Y. D. Aprianto, Sukardi, M. Sya'Roni, and R. Setiadi, “Application of the Hamilton circuit algorithm for optimal route search in city transport routes,” *International Journal of Mechanical Engineering and Technology*, vol. 9, no. 8, 2018.