

Penentuan Rute Terpendek pada Pengiriman Kendaraan Motor menggunakan Algoritma Dijkstra di PT X

¹Intan Pramestiana, ²Asep Anwar

^{1,2}Program Studi Teknik Industri, Universitas Widyatama, Kota Bandung

¹intan.pramestiana@widyatama.ac.id, ²asep.anwar@widyatama.ac.id

Abstrak

PT X merupakan sebuah perusahaan di bidang jasa logistik, dan transaksi keuangan, dimana salah satu bisnis utamanya adalah melakukan pengiriman barang. Proses pengiriman barang sangat berkaitan dengan waktu pengiriman sebagai salah satu faktor utama kepuasan pelanggan. Saat ini PT X menghadapi permasalahan yaitu lamanya waktu yang dibutuhkan dalam proses pengiriman terutama kendaraan motor. Perusahaan berupaya menghemat waktu pengiriman dengan cara mencari rute terpendek yang harus dilewati. Penentuan jalur terpendek adalah salah satu hal yang diperlukan sehubungan dengan waktu tempuh yang dipergunakan untuk penghematan pada faktor lainnya. Penelitian ini bertujuan mencari jalur terpendek yang dapat ditempuh untuk mengirimkan kendaraan oleh kurir di PT X. Algoritma Dijkstra merupakan sebuah algoritma untuk menentukan rute terpendek dengan bobot terkecil dari suatu titik menuju titik lainnya. Algoritma ini melakukan kalkulasi terhadap semua kemungkinan bobot terkecil dari setiap titik. Penelitian ini menggunakan algoritma tersebut sebagai alat untuk menentukan jalur distribusi kendaraan bermotor PT X yang berlokasi di Kota Bandung ke gudang logistik yang berlokasi di Jakarta Timur. Berdasarkan hasil perhitungan didapatkan hasil jalur terpendek mulai dari titik awal di Kota Bandung, melalui Padalarang, Cipatat, Purwakarta, Karawang, Bekasi, dan terakhir cabang yang dituju yang berada di daerah Jakarta Timur dengan total jarak tempuh sebesar 179,3 km. Dari penelitian dianalisis beberapa faktor yang berpotensi mempengaruhi jarak tempuh perjalanan yaitu volume kendaraan, waktu tempuh, persimpangan ke persimpangan lainnya, dan kondisi jalan yang rusak.

Kata Kunci: Algoritma, Dijkstra, Lintasan Terpendek, Logistik.

I. Pendahuluan

PT X pada awalnya merupakan sebuah *Strategic Business Unit* (SBU) logistik dari induk perusahaan yang dibentuk pada tahun 2007. Pada akhir tahun 2011 induk perusahaan melakukan *spin-off* dengan mengubah SBU logistik menjadi perusahaan dengan kepemilikan saham berada di tangan induk perusahaan utama. Maret 2012, PT X resmi didirikan sebagai anak perusahaan, PT X diharapkan dapat beroperasi secara independen dan profesional untuk dapat memaksimalkan peluang bisnis logistik di Indonesia sekaligus memanfaatkan jaringan fisik yang sudah terbangun. Jasa yang ditawarkan oleh PT X terdiri dari *national transport*, *contract logistics*, *freight forwarding*, dan *E-Commerce*. Jasa-jasa yang tersedia pada PT X ini berasal dari kebutuhan konsumen yang sangat tinggi akan permintaan jenis jasa yang berbeda-beda, maka dari itu PT X menawarkan jasa-jasa tersebut untuk memenuhi kebutuhan konsumen. PT X memiliki beberapa *branch office*, salah satunya yang berada di Kota Bandung.

Dalam dunia logistik, efisiensi dan kecepatan pengiriman adalah kunci utama keberhasilan. PT X, sebagai salah satu perusahaan logistik, berupaya untuk terus meningkatkan layanannya. Salah satu tantangan utama yang dihadapi adalah penentuan rute terpendek untuk pengiriman kendaraan motor dari Kota Bandung ke Jakarta Timur. Rute pengiriman yang dipilih tidak hanya berdampak pada waktu pengiriman, tetapi juga pada biaya operasional dan pemanfaatan sumber daya. Pilihan rute yang optimal dapat mengurangi biaya bahan bakar, waktu tempuh, serta kerusakan pada kendaraan. Selain itu, rute yang lebih efisien dapat meningkatkan kepuasan pelanggan dengan menyediakan layanan yang lebih cepat dan handal.

Dalam menghadapi tantangan ini, PT X berencana untuk menerapkan algoritma Dijkstra, sebuah metode algoritmik yang terkenal dalam ilmu komputer untuk menemukan rute terpendek antara dua titik dalam graf. Algoritma ini bekerja dengan cara mengunjungi *node-node* dalam graf secara sistematis dan mengupdate jarak terpendek yang ditemukan dari titik awal ke titik tujuan.

Implementasi Algoritma Dijkstra dalam konteks ini akan melibatkan pemetaan jaringan jalan antara Bandung dan Jakarta Timur, termasuk penentuan bobot atau jarak untuk setiap segmen jalan. Dengan menggunakan algoritma ini, PT X dapat menentukan rute terbaik yang meminimalkan jarak tempuh dan waktu pengiriman, serta mengoptimalkan penggunaan sumber daya logistik mereka. Dalam jangka panjang, penerapan algoritma Dijkstra tidak hanya akan memberikan manfaat operasional bagi PT X, tetapi juga memberikan kontribusi terhadap pengurangan emisi karbon dan dampak lingkungan dari operasi logistik. Dengan mengurangi waktu tempuh dan bahan bakar yang digunakan, perusahaan dapat berkontribusi dalam mendukung keberlanjutan lingkungan.

Secara keseluruhan, penggunaan algoritma Dijkstra dalam menentukan rute pengiriman terpendek dari Bandung ke Jakarta Timur adalah langkah strategis yang dapat membantu PT X dalam meningkatkan efisiensi, mengurangi biaya, dan memberikan layanan yang lebih baik kepada pelanggan mereka.

II. Studi Literatur

Distribusi

Distribusi merupakan sebuah proses penyampaian barang atau jasa dari produsen menuju ke konsumen pada saat lokasi barang atau jasa tersebut diperlukan. Proses distribusi yang dilakukan pada dasarnya menciptakan sebuah utilitas dari waktu dan juga tempat. Dalam melakukan distribusi, perlu pertimbangan beberapa faktor yang bertujuan untuk meminimumkan jarak dan juga waktu yang digunakan. Salah satu cara untuk meminimumkan jarak dan juga waktu adalah dengan mencari rute terpendek (*shortest path*).

Shortest Path

Shortest path (jalur terpendek) adalah konsep dasar dalam teori graf yang mewakili jarak minimum antara dua objek. Berbagai algoritma, termasuk Algoritma Dijkstra, Algoritma Prim, dan Algoritma Floyd Washall, telah dikembangkan untuk secara efisien menentukan jalur dalam grafik (Amruta, dkk., 2024). Masalah jalur terpendek melibatkan penentuan jalur dengan biaya terendah antara dua simpul dalam grafik (Weiss, dkk., 2023). Penelitian ini memperkenalkan jalur terpendek yang dapat digunakan, yang memperhitungkan ketidakpastian. Lintasan terpendek dapat dicari dengan menggunakan graf. Graf yang digunakan adalah graf yang berbobot, yaitu graf yang setiap sisinya diberikan suatu nilai atau bobot. Bobot pada sisi graf dapat menyatakan, waktu, biaya dan sebagainya.

Jalur terpendek pada permasalahan lintasan yaitu sebuah lintasan graph yang diminimalisasi nilai bobotnya. Lintasan terpendek terdiri dari beberapa jenis, diantaranya sebagai berikut:

- shortest path* antara dua buah *node*.
- shortest path* antara semua pasangan *node*.
- shortest path* dari *node* tertentu ke semua *node* yang lain.
- shortest path* antara dua buah *node* yang melalui beberapa *node* tertentu (Ahadi, dkk., 2022).

Teori Graf

Teori graf adalah kerangka matematika yang digunakan dalam ilmu komputer untuk menganalisis dan mengoptimalkan sistem yang kompleks, termasuk perawatan kesehatan. Ini memfasilitasi pengambilan keputusan yang lebih baik, meningkatkan konektivitas dan keamanan, dan mendukung aplikasi seperti pemantauan pasien jarak jauh dan pengobatan yang dipersonalisasi melalui database grafik dan analitik (Sowrirajan & Manimekalai, 2024). Teori graf memberikan wawasan dasar untuk mengembangkan algoritma, seperti menangani masalah *Node-Disjoint Paths*, yang sangat penting untuk perutean yang efisien dalam jaringan (Nimavat, 2023).

Algoritma Dijkstra

Algoritma Dijkstra adalah teknik kunci dalam ilmu komputer untuk menemukan jalur terpendek dalam grafik berbobot. Ini melibatkan inisialisasi, relaksasi, dan pemilihan jalur biaya minimum, dengan pengoptimalan seperti antrian prioritas untuk meningkatkan efisiensi dalam berbagai aplikasi (Avvari, 2024). Dalam hal ini algoritma dapat digunakan sebagai metode untuk mengetahui langkah-langkah secara urut untuk mencapai tujuan. Setiap algoritma memiliki perbedaan dalam mencapai tujuan. Algoritma Dijkstra ditemukan oleh Edsger Wybe Dijkstra pada tahun 1959, algoritma ini dibuat dengan tujuan memecahkan masalah pencarian jalur terpendek dari suatu graf pada setiap simpul yang bernilai tidak

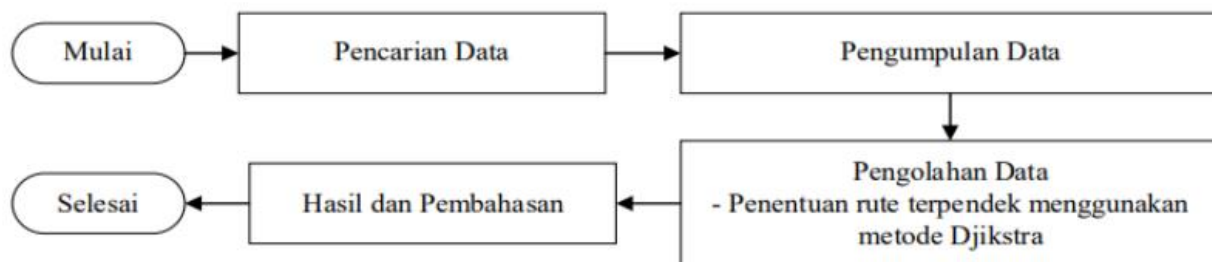
negatif dan termasuk dalam algoritma *greedy* (algoritma yang sering digunakan untuk memecahkan masalah yang berhubungan dengan suatu optimasi). Algoritma Dijkstra dapat meningkatkan efisiensi dalam pemilihan rute, serta memprediksi dan mengoptimalkan biaya lalu lintas serta akurasi waktu pengiriman (Utomo et al., 2023).

Dalam pencarian jalur terpendeknya, algoritma Dijkstra bekerja dengan mencari bobot yang paling minimal dari suatu graf berbobot, jarak terpendek akan diperoleh dari dua atau lebih titik dari suatu graf dan nilai total yang didapat adalah yang bernilai paling kecil. Langkah-langkah dalam menentukan lintasan terpendek pada algoritma Dijkstra yaitu:

1. Beri nilai bobot (jarak) untuk setiap titik ke titik lainnya, lalu set nilai 0 pada *node* awal dan nilai tak hingga terhadap *node* lain (yang belum terisi).
2. Set semua *node* “Belum terjamah” dan set *node* awal sebagai “Node keberangkatan”.
3. Dari *node* keberangkatan, pertimbangkan *node* tetangga yang belum terjamah dan hitung jaraknya dari titik keberangkatan.
4. Setelah selesai menghitung setiap jarak terhadap *node* tetangga, tandai *node* yang telah terjamah sebagai “Node terjamah”. *Node* terjamah tidak akan pernah di cek kembali, jarak yang disimpan adalah jarak terakhir dan yang paling minimal bobotnya.
5. Set “Node belum terjamah” dengan jarak terkecil (dari *node* keberangkatan) sebagai “Node Keberangkatan” selanjutnya dan lanjutkan dengan kembali ke step 3 (Parapat, dkk., 2017).

III. Metodologi Penelitian

Langkah-langkah penelitian yang digunakan dalam pemecahan permasalahan penelitian ditunjukkan pada Gambar 1.



Gambar 1 Alur flowchart penelitian

Secara lengkap, langkah pengolahan data algoritma Dijkstra adalah sebagai berikut:

Tahap 1. Memberikan nilai atau bobot jarak pada setiap *node* ke *node* lainnya.

Tahap 2. Isi semua *node* yang belum diberi nilai atau bobot jarak kemudian tentukan *node* awal sebagai *node* keberangkatan.

Tahap 3. Pertimbangan *node* yang berdekatan yang belum diberi nilai dan hitung jarak dari titik awal.

Tahap 4. Apabila *node* terdekat yang belum terlewati belum diberikan tanda maka *node* tersebut harus ditandai.

Tahap 5. Tentukan *node* yang belum terlewati dengan jarak terkecil dari *node* awal.

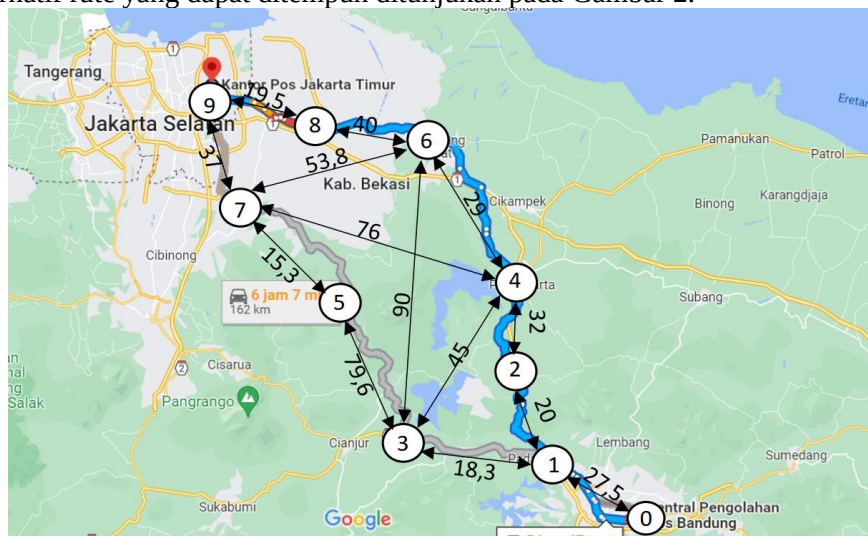
IV. Hasil dan Pembahasan

Data yang digunakan dalam penelitian ini adalah jarak antara PT X Kota Bandung ke gudang yang berlokasi di Jakarta Timur yang didapat dari *google maps*, data yang didapat ditunjukkan pada Tabel 1.

Tabel 1 Jarak antar <i>node</i>		
No	Lokasi	Jarak Antar Node (km)
1	PT X – Padalarang	27,5
2	Padalarang – Cikalong	20
3	Cikalong – Cipatat	32

No	Lokasi	Jarak Antar Node (km)
4	Padalarang – Cipatat	18,3
5	Purwakarta – Karawang	29
6	Cipatat – Jonggol	79,6
7	Karawang – Bekasi	40
8	Karawang – Cileungsi	53,8
9	Purwakarta – Cileungsi	76
10	Cileungsi – Gudang Jakarta Timur	37
11	Bekasi – Gudang Jakarta Timur	19,5

Visualisasi alternatif rute yang dapat ditempuh ditunjukkan pada Gambar 2.

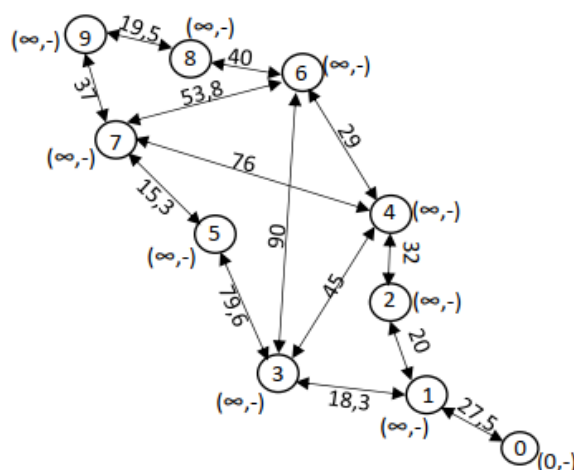


Gambar 2 Alternatif Rute Pengiriman

Penentuan *node* lokasi

Setelah menentukan *node* dan lokasi yang bisa dilewati maka langkah selanjutnya adalah menentukan rute terpendek dengan menggunakan Algoritma Dijkstra.

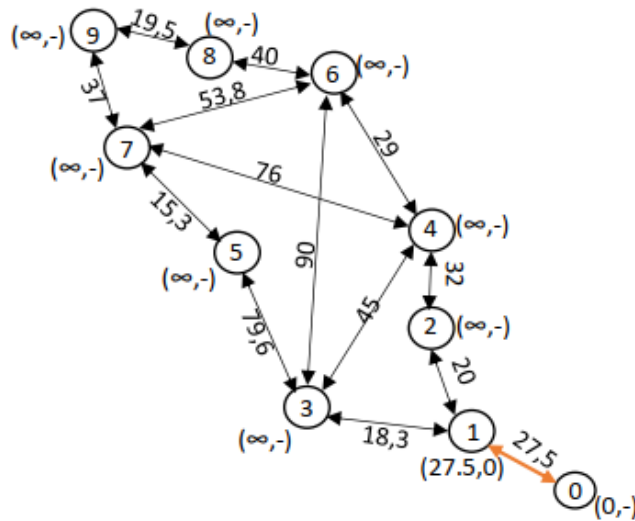
Iterasi 0



Gambar 3 Algoritma Dijkstra Iterasi 0

Pada iterasi 0 dilakukan pemberian nilai pada setiap *node* berdasarkan hasil pengumpulan data yang sudah didapatkan. Selain itu, menentukan titik awal sebagai titik keberangkatan yaitu pada *node* 0 dan titik akhir atau tujuan yaitu pada *node* 9.

Iterasi 1



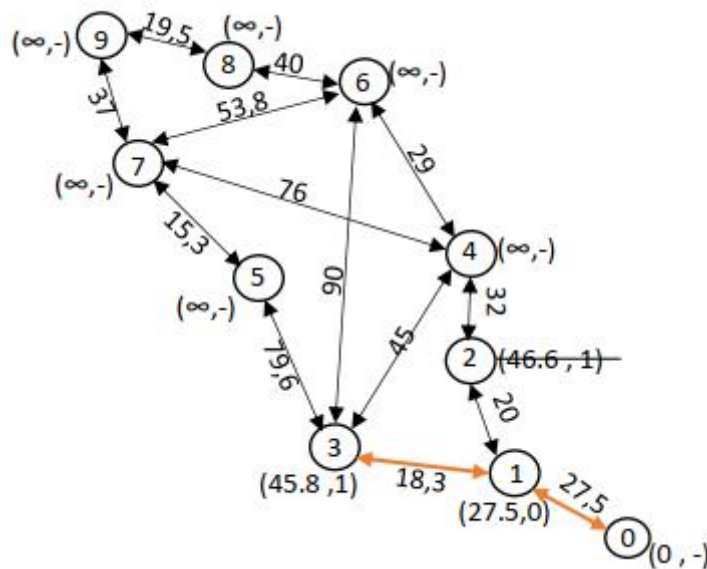
Gambar 4 Algoritma Dijkstra Iterasi 1

Langkah 1: Memilih *node* 0 sebagai simpul asal kemudian diberi label $[0, -]$.

Langkah 2: Dari *node* 0 terdapat satu *node* yang terhubung yaitu *node* 1. *Node* yang dipilih yaitu *node* yang memiliki jarak terpendek dengan cara menjumlahkan jarak dari *node* permanen terbaru. *Node* 1: $[27.5, 0]$

Langkah 3: Jarak terpendek dari *node* 0 yaitu *node* 1 dengan label $27.5, 0]$.

Iterasi 2



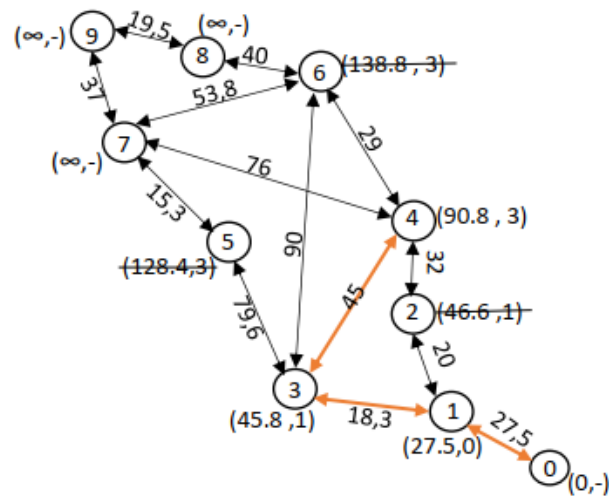
Gambar 5 Algoritma Dijkstra Iterasi 2

Langkah 1: *Node* 1 merupakan *node* dengan jarak terpendek, sehingga *node* 3 dipilih sebagai permanen label terbaru.

Langkah 2: *Node* yang terhubung dengan *node* 1 terdiri dari dua *node*, yaitu *node* 2 dan *node* 3. *Node* yang dipilih yaitu *node* yang memiliki jarak terpendek dengan cara menjumlahkan jarak dari *node* permanen terbaru *Node* 2: $27.5 + 19.1 = 46.6 \rightarrow [46.6, 1]$ *Node* 3: $27.5 + 18.3 = 45.8 \rightarrow [45.8, 1]$ Maka, dipilih *node* 3 dengan label $[45.8, 1]$.

Langkah 3: Jarak terpendek dari *node* 1 yaitu *node* 3 dengan label [45.8, 1].

Iterasi 3



Gambar 6 Algoritma Dijkstra Iterasi 3

Langkah 1: Node 3 merupakan *node* dengan jarak terpendek, sehingga *node* 3 dipilih sebagai permanen label terbaru.

Langkah 2: Node yang terhubung dengan *node* 3 terdiri dari tiga *node*, yaitu *node* 4, *node* 5, dan *node* 6. *Node* yang dipilih yaitu *node* yang memiliki jarak terpendek dengan cara menjumlahkan jarak dari *node* permanen terbaru.

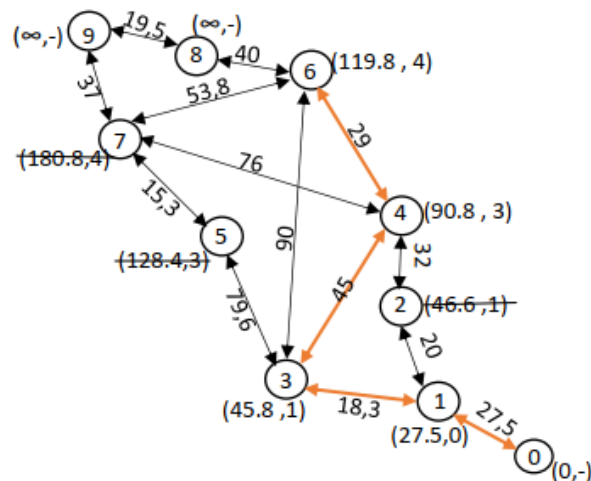
Node 4: $45.8 + 45 = 90.8 \rightarrow [90.8, 3]$

Node 5: $45.8 + 79.6 = 125.4 \rightarrow [125.4, 3]$

Node 6: $45.8 + 90 = 135.8 \rightarrow [135.8, 3]$

Maka, dipilih *node* 4 dengan label [90.8, 3].

Iterasi 4



Gambar 7 Algoritma Dijkstra Iterasi 4

Langkah 1: Node 4 merupakan *node* dengan jarak terpendek, sehingga *node* 4 dipilih sebagai permanen label terbaru.

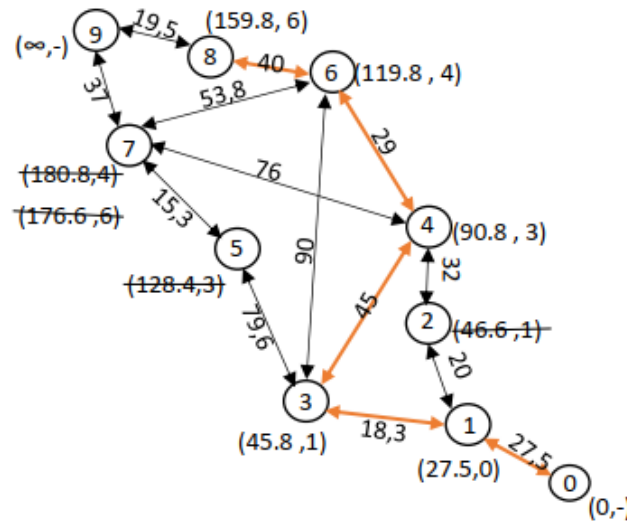
Langkah 2: Node yang terhubung dengan *node* 4 terdiri dari dua *node* yaitu *node* 6 dan *node* 7. *Node* yang dipilih yaitu *node* yang memiliki jarak terpendek dengan cara menjumlahkan jarak dari *node* permanen terbaru.

Node 6: $90.8 + 29 = 119.8 \rightarrow [119.8, 4]$

Node 7: $104.8 + 76 = 180.8 \rightarrow [180.8, 4]$

Maka, dipilih *node* 6 dengan label $[119.8, 4]$.

Iterasi 5



Gambar 8 Algoritma Dijkstra Iterasi 5

Langkah 1: Node 6 merupakan *node* dengan jarak terpendek, sehingga *node* 6 dipilih sebagai permanen label terbaru.

Langkah 2: Node yang terhubung dengan *node* 6 terdiri dari *node* 7 dan *node* 8.

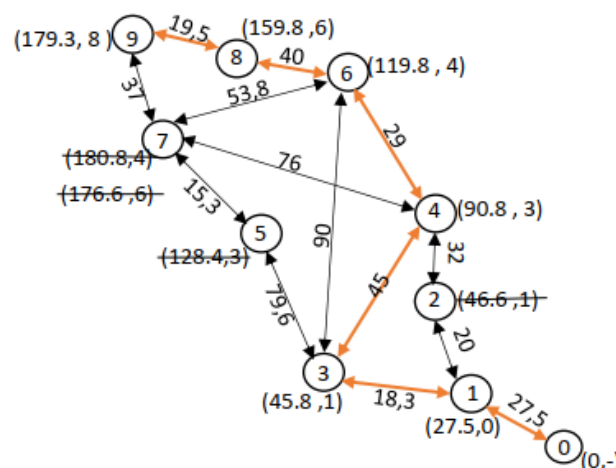
Node yang dipilih yaitu *node* yang memiliki jarak terpendek dengan cara menjumlahkan jarak dari *node* permanen terbaru.

Node 7: $119.8 + 53.8 = 173.6 \rightarrow [173.6, 6]$

Node 8: $119.8 + 40 = 159.8 \rightarrow [159.8, 6]$

Maka, dipilih *node* 8 dengan label $[159.8, 6]$

Iterasi 6



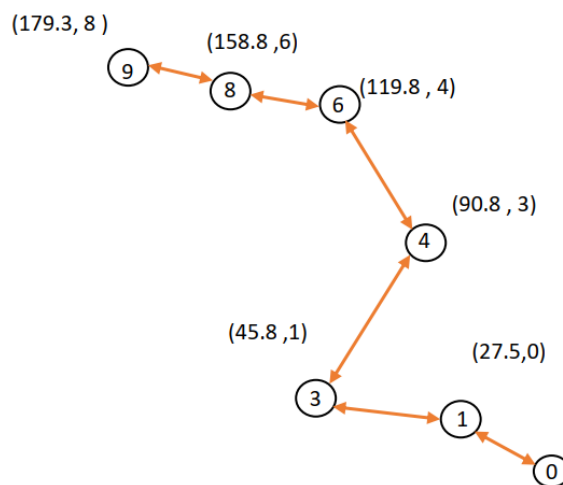
Gambar 9 Algoritma Dijkstra Iterasi 6

Langkah 1: Node yang terhubung dengan *node* 8 terdiri dari satu *node*, yaitu *node* 9. *Node* yang dipilih yaitu *node* yang memiliki jarak terpendek dengan cara menjumlahkan jarak dari *node* permanen terbaru.

Node 9: $159.8 + 19.5 = 179.3 \rightarrow [179.3, 8]$

Maka, dipilih *node* 9 dengan label $[179.3, 8]$.

Hasil Iterasi



Gambar 10 Hasil Iterasi Akhir Algoritma Dijkstra

Untuk sampai ke simpul tujuan yaitu *node* 9 dari simpul asal *node* 0 jarak yang harus ditempuh yaitu dengan melewati 0-1, 1-3, 3-4, 4-6, 6-8, 8-9. Atau dengan cara menjumlahkan yaitu $27,5 + 18,3 + 45 + 29 + 40 + 19,5 = 179,3$ km. Jalan yang harus dilalui yaitu dari PT X Bandung → Padalarang → Cipatat → Purwakarta → Karawang → Bekasi → gudang Jakarta Timur.

Faktor yang Berpengaruh

Dalam proses distribusi dari PT X di Kota Bandung ke gudang Jakarta Timur tentunya selalu ada kendala mengenai waktu dan jarak tempuh yang mengakibatkan pengiriman terlambat sampai ke tempat tujuan sehingga membuat konsumen merasa kurang puas. Beberapa faktor-faktor yang mempengaruhi jarak tempuh diantaranya.

1. Volume Kendaraan

Kepadatan volume kendaraan yang meningkat akan mengakibatkan jarak tempuh terhambat karena terjadinya kemacetan yang dimana kondisi arus lalu lintas yang lewat pada ruas jalan yang ditinjau melebihi kapasitas rencana jalan tersebut yang mengakibatkan kecepatan bebas ruas jalan tersebut mendekati atau melebihi 0 km/jam sehingga menyebabkan terjadinya antrian. Jika dilihat rute jalan di *google maps* jalan yang sering mengalami peningkatan volume kendaraan yaitu Padalarang, Karawang, dan Jakarta.

2. Waktu Tempuh

Berdasarkan dari *Google Maps* jika melalui jalan alternatif Jalan Raya Pantura dengan jarak 184 km maka waktu yang ditempuh adalah 5 jam 34 menit, jika melalui jalan alternatif Jalan Nasional III dengan jarak tempuh 182 km maka perkiraan waktu yang akan ditempuh adalah 5 jam 22 menit, dan jika melalui Padalarang → Cipatat → Purwakarta → Karawang → Bekasi → gudang Jakarta Timur akan dengan jarak tempuh 179,3 km waktu yang akan ditempuh 4 jam 54 menit estimasi tersebut tidak ditambahkan dengan lama berhenti di lampu merah.

3. Persimpangan ke Persimpangan lainnya

Jumlah simpangan berpengaruh terhadap lama perjalanan karena setiap simpangan dimungkinkan adanya lampu merah yang tentunya dapat menyita waktu.

4. Kondisi Jalan yang Rusak

Kondisi jalan yang rusak juga berpengaruh pada cepat sampainya ke tujuan, banyaknya jalan yang rusak atau berlubang mempengaruhi lambatnya perjalanan.

V Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan rute terpendek distribusi dari PT X yang terletak di Kota Bandung untuk mengirimkan kendaraan motor ke gudang yang berada di Jakarta Timur. Jalur yang harus dilalui dari PT X untuk sampai ke tujuan yaitu titik 0-1-3-4-6-8-9 atau melalui

Padalarang → Cipatat → Purwakarta → Karawang → Bekasi → gudang Jakarta Timur. Terdapat beberapa faktor yang dapat mempengaruhi jarak tempuh perjalanan diantaranya persimpangan ke persimpangan lainnya karena setiap simpangan dimungkinkan adanya lampu merah yang tentunya dapat menyita waktu, volume kendaraan, waktu tempuh, dan kondisi jalan yang rusak karena banyaknya jalan yang rusak atau berlubang mempengaruhi lambatya perjalanan.

Daftar Pustaka

- [1] Ahadi, I., Habibah, M. N., Deria, P. P. D., & Fauzi, M. (2022). Penerapan Algoritma Dijkstra untuk Mencari Rute Terpendek pada Pengiriman Produk Wafer di PT. XYZ. *JURMATIS (Jurnal Manajemen Teknologi Dan Teknik Industri)*, 4(1), 1. <https://doi.org/10.30737/jurmatiss.v4i1.1838>
- [2] Amruta, A., Sneha Pawar, & Bharati Bhamare. (2024). Implementation of Shortest Path Algorithms. *International Journal of Scientific Research in Science, Engineering and Technology*, 11(2), 392–397. <https://doi.org/10.32628/IJSRSET2411261>
- [3] Avvari, R. L. (2024). A Dijkstra's Algorithm: Theory and Application to Asia Map. *Communications on Applied Nonlinear Analysis*, 32(2), 187–194. <https://doi.org/10.52783/cana.v32.1735>
- [4] Nimavat, R. (2023). *Graph Theory and its Uses in Graph Algorithms and Beyond* (Version 1). arXiv. <https://doi.org/10.48550/ARXIV.2308.15473>
- [5] Parapat, M. N., Kusbianto, D., & Rahmad, C. (2017). RANCANG BANGUN APLIKASI PENCARIAN RUTE TERPENDEK JASA KIRIMAN BARANG BERBASIS MOBILE DENGAN METODE ALGORITMA DIJKSTRA. *Jurnal Informatika Polinema*, 3(3), 15–19. <https://doi.org/10.33795/jip.v3i3.28>
- [6] Sowrirajan, R., & Manimekalai, S. (2024). Graph-Theoretic Approaches to Optimizing Connectivity and Security in Ubiquitous Healthcare Systems: In A. Suresh Kumar, G. Ganesan, R. Sekaran, B. Krishnan, & N. V. Kousik (Eds.), *Advances in Medical Technologies and Clinical Practice* (pp. 327–351). IGI Global. <https://doi.org/10.4018/979-8-3693-2268-0.ch015>
- [7] Utomo, D. D., Aurelia, M., Tanasia, S. M., Nurhasanah, & Handoyo, A. T. (2023). Implementation of Dijkstra Algorithm in Vehicle Routing to Improve Traffic Issues in Urban Areas. *2023 3rd International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS)*, 73–78. <https://doi.org/10.1109/ICON-SONICS59898.2023.10435225>
- [8] Weiss, E., Felner, A., & Kaminka, G. A. (2023). *Tightest Admissible Shortest Path* (Version 2). arXiv. <https://doi.org/10.48550/ARXIV.2308.08453>