Epsilon: Jurnal Matematika Murni dan Terapan
http://ppjp.ulm.ac.id/journals/index.php/epsilon
p-ISSN: 1978-4422 | e-ISSN 2656-7660
Vol.19 No.1 (Juni 2025) Hal. 11-26

PENERAPAN MODIFIKASI ALGORITMA WARNSDORFF DALAM IDENTIFIKASI LINTASAN DAN SIRKUIT HAMILTON PADA GRAF SEDERHANA

Fadilatul Amalia, Nilamsari Kusumastuti, Yudhi

Program Studi Matematika Fakultas MIPA Universitas Tanjungpura
Jl. Prof. Dr. H. Hadari Nawawi, 78124, Kalimantan Barat
email: *fadilatulamalia28@gmail.com*

ABSTRACT

Graph Hamilton is one of the concepts in graph theory related to a path that visits each vertex exactly once before returning to the starting vertex in the case of a Hamiltonian circuit. This research focuses on determining whether a graph has a Hamiltonian path or circuit using the Warnsdorff algorithm. The Warnsdorff algorithm was originally developed to solve the Knight's Tour problem on a chessboard by selecting the vertex with the smallest degree to reduce the likelihood of getting stuck without any further moves. However, this algorithm has a weakness as it does not always succeed in finding a Hamiltonian path in various types of graphs. Therefore, this study modifies the Warnsdorff algorithm by implementing a strategy of selecting the starting vertex based on the smallest degree and applying a re-search mechanism if a Hamiltonian path is not found. If the path is still not found, the starting vertex is replaced with the next smallest-degree vertex until a path is found or all possibilities have been tested. This modification is expected to increase the algorithm's chances of finding a Hamiltonian path compared to the unmodified Warnsdorff algorithm. The research methodology includes applying the modified Warnsdorff algorithm to various simple graphs and analyzing its performance in finding Hamiltonian paths. The results show that the modified Warnsdorff algorithm can identify Hamiltonian paths in certain graph structures. This modification not only improves the success rate of finding Hamiltonian paths but also reduces the number of steps in some cases, with potential applications in route optimization and network design. The conclusion of this study confirms that although modifying the Warnsdorff algorithm improves its success in finding Hamiltonian paths, further development is still necessary to make it more reliable in handling graphs with more complex structures.

Keywords: Algorithm Modification, Heuristic, Path Identification.

ABSTRAK

Graf Hamilton adalah salah satu konsep dalam teori graf yang berkaitan dengan lintasan yang mengunjungi setiap simpul tepat satu kali sebelum kembali ke simpul awal dalam kasus sirkuit Hamilton. Penelitian ini berfokus untuk menentukan apakah suatu graf memiliki lintasan atau sirkuit Hamilton dengan menggunakan algoritma Warnsdorff. Algoritma Warnsdorff awalnya dikembangkan untuk menyelesaikan permainan Knight's Tour pada papan catur dengan memilih simpul yang memiliki derajat terkecil untuk mengurangi kemungkinan terjebak dalam kondisi tanpa langkah lanjutan. Namun, algoritma ini memiliki kelemahan karena tidak selalu berhasil menemukan lintasan Hamilton pada berbagai jenis graf. Oleh karena itu, penelitian ini memodifikasi algoritma Warnsdorff dengan strategi pemilihan simpul awal berdasarkan derajat terkecil dan menerapkan mekanisme pencarian ulang jika lintasan Hamilton tidak ditemukan. Jika tetap tidak ditemukan, maka simpul awal akan diganti dengan simpul berderajat terkecil berikutnya hingga lintasan ditemukan atau seluruh kemungkinan telah diuji. Modifikasi ini diharapkan dapat meningkatkan peluang algoritma dalam menemukan lintasan Hamilton dibandingkan dengan

algoritma Warnsdorff yang belum dimodifikasi. Metodologi penelitian mencakup penerapan algoritma Warnsdorff yang telah dimodifikasi pada berbagai graf sederhana serta analisis kinerjanya dalam menemukan lintasan Hamilton. Hasil penelitian menunjukkan bahwa modifikasi algoritma Warnsdorff dapat mengidentifikasi lintasan Hamilton dalam beberapa struktur graf. Modifikasi ini tidak hanya meningkatkan keberhasilan pencarian lintasan Hamilton tetapi juga mengurangi jumlah langkah dalam beberapa kasus, dengan potensi aplikasi pada optimasi rute dan jaringan. Kesimpulan penelitian ini menegaskan bahwa meskipun modifikasi algoritma Warnsdorff meningkatkan keberhasilannya dalam menemukan lintasan Hamilton, pengembangan lebih lanjut masih diperlukan agar lebih andal dalam menangani graf dengan struktur yang lebih kompleks.

Kata kunci: Heuristik, Identifikasi Lintasan, Modifikasi Algoritma.

Received: 9 Desember 2024, Accepted: 10 Januari 2025, Published: 30 April 2025

PENDAHULUAN

Teori graf merupakan salah satu cabang ilmu matematika yang digunakan untuk memodelkan objek-objek diskrit dan hubungan di antara mereka salah satu pembelajaran dalam teori graf adalah Graf Hamilton (Mahmudah & Irawati, 2018). Pada tahun 1895, matematikawan Irlandia Hamilton pertama kali mengusulkan masalah "perjalanan global". Dengan menggunakan 20 simpul dari Dodecahedron untuk mewakili 20 kota besar di dunia, para pelancong diminta untuk menyusuri sisi Dodecahedron, mulai dari salah satu kota dan melewati semua kota lainnya, setiap kota hanya satu kali dilewati, dan akhirnya kembali ke titik awal (Pegg, 2009), persoalan ini dinamakan mencari sirkuit Hamilton. Graf yang mengandung sirkuit Hamilton disebut graf Hamilton, sementara graf yang hanya memiliki lintasan Hamilton dikenal sebagai graf semi-Hamilton (Zahro, 2018).

Graf Hamilton adalah jenis graf yang menarik perhatian karena banyak digunakan dalam berbagai aplikasi. Salah satu masalah paling terkenal adalah Traveling Salesman Problem tujuannya untuk menemukan urutan rute atau jalur terpendek (Wibawa, 2022), penggunaan lainnya yaitu untuk menyelesaikan permainan Knight's Tour pada papan catur. Knight's Tour adalah rangkaian perjalanan kuda catur pada papan catur sehingga seluruh kotak terlewati oleh kuda catur tepat satu kali (Joni & Erwin, 2017). Ada beberapa cara yang dapat digunakan untuk menyelesaikan permainan Knight's Tour, diantaranya menggunakan metode De Moivre, algoritma Warnsdorff, algoritma Backtracking, dan algoritma Divide and Conquer (Fauziyah dkk, 2020) Algoritma Warnsdorff lebih baik dari pada algoritma lainnya untuk penyelesaian Knight's Tour karena Algoritma Warnsdorff menggunakan pendekatan heuristik yang meminimalkan backtracking (Serdano dkk., 2019). Keunggulan utama Algoritma Warnsdorff terletak pada pendekatan heuristiknya yang secara efisien mengurangi eksplorasi lintasan yang tidak pasti, sehingga lebih cepat dibandingkan metode lain seperti backtracking yang sering kali mengalami banyak eksplorasi. Selain itu, pendekatan ini dapat diterapkan tidak hanya dalam permainan Knight's Tour, tetapi juga dalam kecerdasan buatan navigasi robot, optimasi rute dan jaringan, yang membutuhkan

solusi cepat tanpa harus mengevaluasi setiap kemungkinan lintasan (Ihsan dkk., 2024).

Algoritma Warnsdorff memiliki keterkaitan dengan graf Hamilton karena Knight's Tour dapat direpresentasikan sebagai masalah lintasan Hamilton (Olsen & Babicheva, 2024). Meskipun Algoritma Warnsdorff adalah heuristik dan tidak menjamin solusi dalam semua kasus, efektivitasnya dalam memecahkan Knight's Tour dapat menginspirasi teknik serupa dalam mencari lintasan Hamilton pada sebarang graf (Pohl, 1967). Mengingat bahwa keberadaan lintasan atau sirkuit Hamilton sulit diidentifikasi karena beberapa teorema hanya memberi syarat cukup yang tidak selalu memastikan adanya sirkuit Hamilton pada setiap graf (Wijaya, 2016). Untuk itu, dikembangkan program berbasis algoritma Warnsdorff yang telah dimodifikasi untuk meningkatkan kemungkinan menemukan lintasan Hamilton. Modifikasi ini dilakukan karena algoritma Warnsdorff asli cenderung kurang efektif pada graf yang lebih kompleks, sehingga diperlukan pendekatan yang lebih adaptif untuk mempercepat proses pencarian lintasan atau sirkuit Hamilton.

TINJAUAN PUSTAKA

Penelitian ini membahas tentang algoritma Warnsdorff untuk menemukan lintasan Hamilton dengan mengacu pada beberapa jurnal dan artikel yang saling berkaitan. Lintasan sederhana dalam suatu perjalanan yang melewati setiap simpul dalam graf tepat satu kali disebut sebagai lintasan Hamilton. Dalam lintasan ini, simpul awal dan simpul akhir berbeda. Namun, jika lintasan tersebut membentuk siklus dengan simpul awal dan akhir yang sama, maka disebut sirkuit Hamilton (Rozi & Multahadah, 2021). Jika setiap siklus dalam graf mencakup semua simpul, maka graf tersebut disebut sebagai graf Hamilton. Graf yang memiliki siklus disebut sebagai sirkuit Hamilton. Sementara itu, graf semi Hamilton adalah graf yang memiliki lintasan dengan melewati setiap simpul dalam graf, tetapi bukan lintasan tertutup (Yuniar & Panjaitan, 2024).

Dasar awal penelitian ini adalah jurnal dari Pohl (1967) yang berjudul *A Method for Finding Hamilton Paths and Knight's Tour*. Jurnal ini memperkenalkan metode dasar dalam mencari lintasan Hamilton dan menyelesaikan masalah *Knight's Tour*, yang kemudian menjadi fondasi penting bagi pengembangan algoritma di masa mendatang. Berdasarkan konsep awal tersebut, Pranav (2019) dalam artikelnya *A Comparison of Warnsdorff's Rule and Backtracking for Knight's Tour* mengembangkan studi dengan membandingkan algoritma Warnsdorff dan metode backtracking. Pranav menganalisis kelebihan dan kekurangan algoritma Warnsdorff dalam menyelesaikan masalah *Knight's Tour*, memberikan wawasan tentang efektivitasnya dibandingkan dengan metode lain. Artikel ini memperkuat konsep dasar dari Pohl dengan menguji efisiensi algoritma. Penelitian dilanjutkan oleh Fauziyah dkk., (2020) dalam artikelnya yang berjudul *Aturan Warnsdorff dan Algoritma Backtracking pada Permainan The Knight's*

Tour. Fauziyah memperjelas implementasi algoritma Warnsdorff dengan menjelaskan langkah-langkah penerapannya secara rinci. Terakhir, artikel terbaru dari Edvard Olsen (2024) yang berjudul *On the Convergence of the Warnsdorff's Algorithm on Rectangular Boards* membahas tentang faktor-faktor yang menentukan keberhasilan algoritma dalam menemukan lintasan Hamilton, menambah atau menyempurnakan hasil penelitian sebelumnya.

Selain itu, penelitian terbaru yang lebih luas tentang implementasi algoritma heuristik dalam penyelesaian graf Hamilton juga telah dilakukan. Misalnya, studi oleh Jin *et al*., (2022) dalam artikelnya *Heuristic Approaches for Hamiltonian Path and Cycle Problems in Large-Scale Graphs* mengkaji efektivitas berbagai metode heuristik, termasuk algoritma genetika dan optimasi berbasis semut (ant colony optimization), dalam menemukan lintasan Hamilton secara efisien pada graf berskala besar. Penelitian ini memberikan perspektif baru tentang bagaimana teknik heuristik dapat meningkatkan performa algoritma klasik seperti Warnsdorff.

METODE PENELITIAN

Metode penelitian yang digunakan adalah studi pustaka, yang melibatkan pencarian, pembacaan, dan pemahaman literatur yang berkaitan dengan graf Hamilton dan algoritma Warnsdorff. Literatur tersebut diperoleh dari berbagai sumber, termasuk buku, jurnal, artikel, dan penelusuran *online*. Langkah-langkah yang akan diambil untuk membahas penelitian ini adalah sebagai berikut:

1. Mengumpulkan data graf yang akan digunakan untuk pengujian, Graf tersebut adalah gabungan dari beberapa jenis graf, yaitu: graf sederhana (*simple graph*), tanpa adanya sisi ganda maupun *loop*, graf tak-berarah (*undirected graph*), sehingga setiap sisi tidak memiliki arah tertentu, graf terhubung (*connected graph*), di mana terdapat setidaknya satu lintasan antara setiap pasangan simpul, dan graf dengan derajat setiap simpulnya minimal 2, untuk memenuhi syarat dasar keberadaan lintasan Hamilton. Pemilihan graf dilakukan dengan mempertimbangkan variasi jumlah simpul dan sisi untuk mengamati kinerja algoritma pada kompleksitas graf yang berbeda.
2. Mengimplementasikan algoritma Warnsdorff dan mengidentifikasi lintasan Hamilton pada data graf. Memodifikasi algoritma Warnsdorff agar memiliki peluang lebih besar untuk menemukan lintasan Hamilton dibandingkan dengan algoritma Warnsdorff versi asli.
3. Menganalisis hasil pengujian untuk mengidentifikasi kekurangan dan kelemahan dari algoritma Warnsdorff dalam mendeteksi lintasan Hamilton.
4. Implementasikan program yang dikembangkan pada graf, di mana program tersebut berbasis Algoritma Warnsdorff yang telah dimodifikasi untuk mendeteksi lintasan Hamilton.

Pengujian kompleksitas algoritma, dilakukan dengan pendekatan kompleksitas waktu, mengukur berapa lama waktu yang dibutuhkan untuk menyelesaikan proses pencarian lintasan Hamilton pada graf. Hasil yang diperoleh

dari implementasi akhir akan dibandingkan dengan implementasi algoritma Warnsdorff versi asli untuk menunjukkan peningkatan efisiensi.

HASIL DAN PEMBAHASAN

Dimisalkan graf dengan himpunan simpul-simpul yang dilambangkan dengan V dan himpunan sisi-sisi yang dilambangkan dengan E. Untuk menemukan lintasan Hamilton pada graf tersebut, langkah-langkah algoritma Warnsdorff dapat digunakan sebagai berikut:

Diketahui

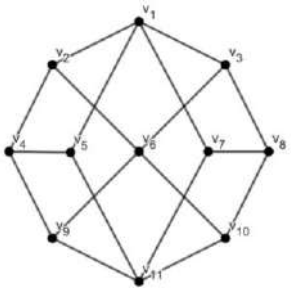
i. V adalah himpunan semua simpul dalam graf
ii. v_i adalah simpul awal atau simpul saat ini
iii. T adalah himpunan simpul pada lintasan Hamilton
iv. $d(v_j)$ adalah derajat simpul v_j
v. $N(v_i)$ adalah himpunan simpul di V yang ber-*adjacent* dengan simpul v_i
vi. $|T \cap N(v_j)|$ adalah banyaknya elemen T yang ber-*adjacent* dengan simpul v_j
vii. $d'(v_j)$ adalah derajat simpul v_j setelah dikurangi dengan banyaknya elemen T yang ber-*adjacent* dengan simpul v_j
viii. $\min\{d'(v_j)\}_{v_j \in N(v_i)}$ adalah nilai minimum dari $d'(v_j)$ untuk setiap v_j dengan v_j merupakan elemen $N(v_i)$

Langkah-langkah algoritma Warnsdorff:

a) Pilih simpul awal $v_i \in V$.
   $T \coloneqq \emptyset$
   $V \coloneqq \{v_i | v_i$ adalah simpul dari graf$\}$
b) Tandai v_i sebagai simpul yang telah dikunjungi.
   $T \coloneqq T \cup \{v_i\}$
   $V \coloneqq V - \{v_i\}$
c) Selama $|N(v_i)| \neq 0$, ulangi proses berikut!
   i. $\forall v_j \in N(v_i)$:
      Hitung $d'(v_j) \coloneqq d(v_j) - |T \cap N(v_j)|$
   ii. Pilih simpul dengan $d'(v_j) = \min\{ d'(v_j)\}_{v_j \in N(v_i)}$
   iii. Tandai v_j sebagai simpul yang telah dikunjungi:
      $T \coloneqq T \cup \{v_j\}$
      $V \coloneqq V - \{v_j\}$
   iv. Perbarui simpul saat ini:
      Perbarui $N(v_i) = N(v_i) - \{v_j\}$
      $v_i \coloneqq v_j$
d) Periksa lintasan Hamilton.
   Jika $|V| = 0$, maka ada lintasan Hamilton.

Jika $|V| \neq 0$, maka tidak ada lintasan Hamilton.

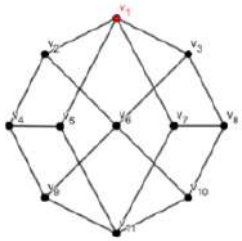
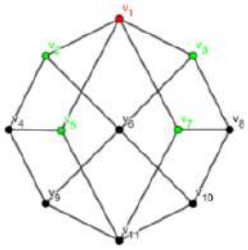
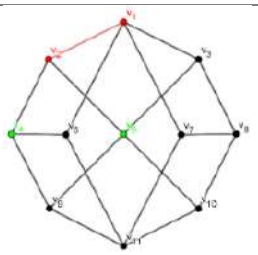
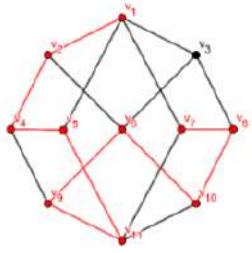
Pengimplementasian algoritma Warnsdorff pada Gambar 1 graf G_2.



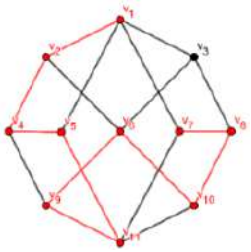
Gambar 1. Graf G_2

Diketahui simpul dari graf G_2 yaitu $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$.

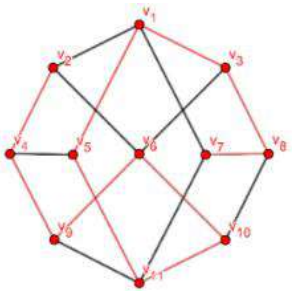
Tabel 1. Implementasi Algoritma Warnsdorff pada graf G_2

No	Langkah	Gambar
a	Pilih simpul awal $v_i \in V$. $T \coloneqq \emptyset$ $V \coloneqq \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$ Dipilih simpul secara acak yaitu v_1	
b	Tandai v_i sebagai simpul yang telah dikunjungi. $T \coloneqq T \cup \{v_1\} = \{v_1\}$ $V \coloneqq V - \{v_1\} = \{v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$	
c	Selama $\|N(v_i)\| \neq 0$, ulangi proses! Untuk v_1 i. $N(v_1) \coloneqq \{v_2, v_3, v_5, v_7\}$ ii. $\forall v_j \in N(v_1)$: $d'(v_2) = d'(v_3) = d'(v_5) = d'(v_7) = 2$ iii. Pilih simpul dengan $d'(v_j) = \min\{d'(v_j)\}_{v_j \in N(v_1)}$ yaitu v_2 (karena sama maka pilih simpul secara acak) iv. Tandai v_2 sebagai simpul yang telah dikunjungi: $T \coloneqq T \cup \{v_2\} = \{v_1, v_2\}$ $V \coloneqq V - \{v_2\} = \{v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$ v. Perbarui simpul saat ini: $N(v_1) = N(v_1) - \{v_2\} = \{v_3, v_5, v_7\}$ $v_1 \coloneqq v_2$ ⋮	 

	. Untuk v_7 i. $N(v_7) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d
d	Periksa lintasan Hamilton. $T := \{v_1, v_2, v_4, v_5, v_{11}, v_9, v_6, v_{10}, v_8, v_7\}$ $V := \{v_3\}$ Karena $\|V\| \neq 0$, maka tidak ada lintasan Hamilton

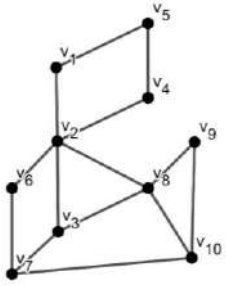


Dari Tabel 1, implementasi algoritma Warnsdorff dengan simpul awal v_1 diperoleh lintasan pada graf G_2 yaitu $T = \{v_1, v_2, v_4, v_5, v_{11}, v_9, v_6, v_{10}, v_8, v_7\}$ ada satu simpul yang belum dikunjungi yaitu v_3. Akan dipilih simpul awal yang berbeda, algoritma Warnsdorff dengan simpul awal v_2 diperoleh lintasan pada graf G_2 yaitu $T = \{v_1, v_2, v_4, v_5, v_{11}, v_9, v_6, v_{10}, v_8, v_7\}$.



Gambar 2. Lintasan diawali simpul v_2

Pengimplementasian algoritma Warnsdorff dengan simpul awal v_1, tidak diperoleh lintasan Hamilton pada graf karena ada satu simpul yang belum dikunjungi dalam penyelesaian tersebut. Hal ini menunjukkan bahwa pemilihan simpul awal sangat mempengaruhi hasil akhir. Oleh karena itu, untuk mengatasi masalah ini, algoritma Warnsdorff akan dimodifikasi dengan fokus pada pemilihan simpul awal, dengan memilih simpul yang memiliki derajat paling kecil. Diketahui Gambar 3 Graf G_4 sebagai berikut.



Gambar 3. Graf G_4

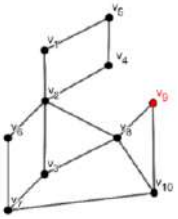
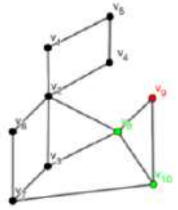
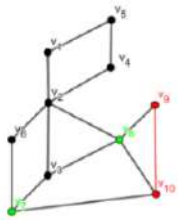
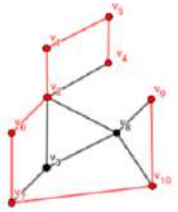
Dari Gambar 3 Graf $\boldsymbol{G_4}$ diketahui simpul dan derajat disajikan pada Tabel 2.

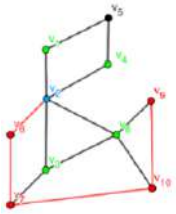
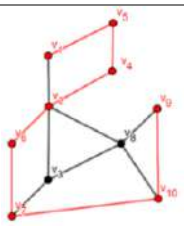
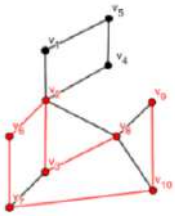
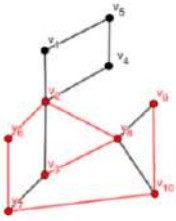
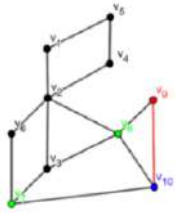
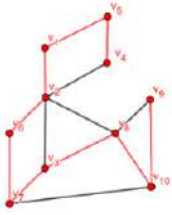
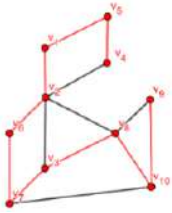
Tabel 2. Derajat setiap simpul G_4

Simpul (v_i)	v_1	v_2	v_3	v_4	v_5	v_6	v_8	v_9	v_{10}
Derajat $(d(v_i))$	2	5	3	2	2	2	4	2	3

Implementasikan algoritma Warnsdorff yang sudah dimodifikasi pada Gambar 3 Graf G_4 dalam Tabel 3.

Tabel 3. Implementasi Algoritma Warnsdorff pada G_4 (1)

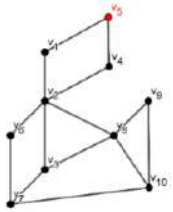
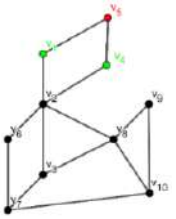
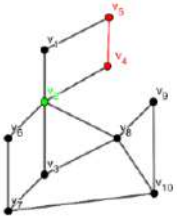
No	Langkah	Gambar
a	Pilih simpul awal $v_i \in V$. $T \coloneqq \emptyset$ $V \coloneqq \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$ Pilih simpul dengan $d(v_i) = \min\{d(v_i)\}_{v_i \in V}$ yaitu v_9	
b	Tandai v_i sebagai simpul yang telah dikunjungi. $T \coloneqq T \cup \{v_9\} = \{v_9\}$ $V \coloneqq V - \{v_9\} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_{10}\}$	
c	Selama $\lvert N(v_i)\rvert \neq 0$, ulangi proses! Untuk v_9 i. $N(v_9) \coloneqq \{v_8, v_{10}\}$ ii. $\forall v_j \in N(v_5)$: $d'(v_8) = 3$, $d'(v_{10}) = 2$ iii. Pilih simpul dengan $d'(v_j) = \min\{d'(v_j)\}_{v_j \in N(v_9)}$ yaitu v_{10} iv. Tandai v_{10} sebagai simpul yang telah dikunjungi: $T \coloneqq T \cup \{v_{10}\} = \{v_9, v_{10}\}$ $V \coloneqq V - \{v_{10}\} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$ v. Perbarui simpul saat ini: $N(v_9) = N(v_9) - \{v_{10}\} = \{v_8\}$ $v_9 \coloneqq v_{10}$ . . . Untuk v_4 i. $N(v_4) \coloneqq \emptyset$ Karena $\lvert N(v_i)\rvert = 0$, maka lanjut ke langkah d	 

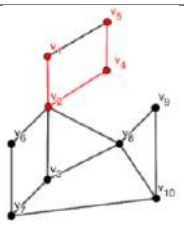
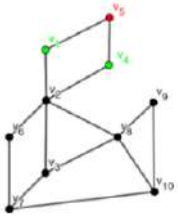
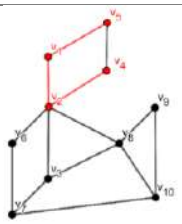
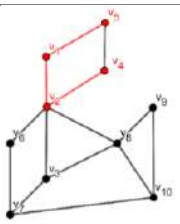
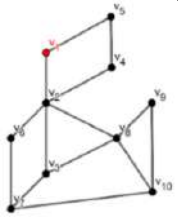
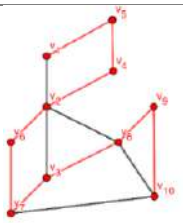
d	Periksa lintasan Hamilton. $T := \{v_9, v_{10}, v_7, v_6, v_2, v_1, v_5, v_4\}$ $V := \{v_3, v_8\}$ Karena $\|V\| \neq 0$, maka lakukan pencarian ulang, dengan memilih simpul lain pada saat pemilihan simpul yang memiliki jumlah derajat terkecil yang sama. Reset kembali himpunan T dan himpunan V saat berada di v_2 lalu ulangi langkah c	
	. . . Untuk v_1 i. $N(v_1) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d Untuk v_8 i. $N(v_8) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d Untuk v_3 i. $N(v_3) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d	  
	Periksa lintasan Hamilton. Karena $\|V\| \neq 0$, maka lakukan pencarian ulang, dengan memilih simpul lain pada saat pemilihan simpul yang memiliki jumlah derajat terkecil yang sama. Reset kembali himpunan T dan himpunan V saat berada di v_{10} lalu ulangi langkah c	
	. . . Untuk v_4 i. $N(v_4) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d	
	Periksa lintasan Hamilton. $T := \{v_9, v_{10}, v_8, v_3, v_7, v_6, v_2, v_1, v_5, v_4\}$ $V := \emptyset$ Karena $\|V\| = 0$ maka ada lintasan Hamilton.	

Dengan memilih simpul awal v_9 diperoleh lintasan Hamilton $T = \{v_9, v_{10}, v_8, v_3, v_7, v_6, v_2, v_1, v_5, v_4\}$, dengan simpul awal v_9 saat $T = \{v_9, v_{10}, v_7, v_6, v_2, v_1, v_5, v_4\}$ tidak terdapat lintasan hamilton karena memilih langkah selanjutnya ke simpul v_7 pada saat berada di simpul v_{10} sehingga masih terdapat simpul yang belum dikunjungi yaitu simpul v_3 dan v_8, namun saat $T = \{v_9, v_{10}, v_8, v_3, v_7, v_6, v_2, v_1, v_5, v_4\}$ dengan simpul awal v_9 terdapat lintasan Hamilton karena memilih langkah selanjutnya ke simpul v_8 pada saat berada di simpul v_{10} sehingga semua simpul dikunjungi.

Dengan permasalahan yang diperoleh, algoritma Warnsdorff akan dimodifikasi dengan menambahkan poin "Lakukan Pencarian Ulang (*Re-search*) dengan memilih simpul lain pada saat pemilihan simpul yang memiliki jumlah derajat terkecil yang sama". Akan dicoba memilih simpul awal dengan derajat terkecil yang lain. Berikut diberikan Tabel 4 sebagai implementasi Algoritma Warnsdorff pada graf G_4 (2)

Tabel 3. Implementasi Algoritma Warnsdorff pada graf G_4 (2)

No	Langkah	Gambar
a	Pilih simpul awal $v_i \in V$. $T \coloneqq \emptyset$ $V \coloneqq \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$ Pilih simpul dengan $d(v_i) = \min\{d(v_i)\}_{v_i \in V}$ yaitu v_5	
b	Tandai v_i sebagai simpul yang telah dikunjungi. $T \coloneqq T \cup \{v_5\} = \{v_5\}$ $V \coloneqq V - \{v_5\} = \{v_1, v_2, v_3, v_4, v_6, v_7, v_8, v_9, v_{10}\}$	
c	Selama $\lvert N(v_i)\rvert \neq 0$, ulangi proses! Untuk v_5 i. $N(v_1) \coloneqq \{v_1, v_4\}$ ii. $\forall v_j \in N(v_5)$: $d'(v_1) = 1, d'(v_4) = 1$ iii. Pilih simpul dengan $d'(v_j) = \min\{d'(v_j)\}_{v_j \in N(v_5)}$ yaitu v_4 (karena sama maka pilih simpul secara acak) iv. Tandai v_4 sebagai simpul yang telah dikunjungi: $T \coloneqq T \cup \{v_4\} = \{v_5, v_4\}$ $V \coloneqq V - \{v_2\} = \{v_1, v_2, v_3, v_6, v_7, v_8, v_9, v_{10}\}$ v. Perbarui simpul saat ini: $N(v_5) = N(v_5) - \{v_4\} = \{v_1\}$ $v_5 \coloneqq v_4$	

	. . . Untuk v_1 i. $N(v_1) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d	
d	Periksa lintasan Hamilton. $T := \{v_5, v_4, v_2, v_1\}$ $V := \{v_3, v_6, v_7, v_8, v_9, v_{10}\}$ Karena $\|V\| \neq 0$, maka lakukan pencarian ulang, dengan memilih simpul lain pada saat pemilihan simpul yang memiliki jumlah derajat terkecil yang sama. Reset kembali himpunan T dan himpunan V saat berada di v_5 lalu ulangi langkah c	
	. . . Untuk v_4 i. $N(v_4) := \emptyset$ Karena $\|N(v_i)\| = 0$, maka lanjut ke langkah d	
	Periksa lintasan Hamilton. $T := \{v_5, v_1, v_2, v_4\}$ $V := \{v_3, v_6, v_7, v_8, v_9, v_{10}\}$ Karena $\|V\| \neq 0$, dan sudah melakukan pencarian ulang, dengan memilih simpul lain pada saat pemilihan simpul yang memiliki jumlah derajat terkecil yang sama, maka pilih simpul awal yang lain. Reset kembali himpunan T dan himpunan V, lalu ulangi langkah a	
	Pilih simpul awal $v_i \in V$. $T := \emptyset$ $V := \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$ Pilih simpul dengan $d(v_i) = \min\{d(v_i)\}_{v_i \in V}$ yaitu v_1	
	Dengan mengulangi langkah a saat langkah d Periksa lintasan Hamilton. $T := \{v_1, v_5, v_4, v_2, v_6, v_7, v_3, v_8, v_9, v_{10}\}$ $V := \emptyset$ Karena $\|V\| = 0$ maka ada lintasan Hamilton.	

Simpul yang diawali dengan simpul v_5 tidak memiliki lintasan Hamilton. Dengan permasalahan yang diperoleh, algoritma Warnsdorff akan dimodifikasi dengan menambahkan poin "Jika masih tidak ditemukan lintasan setelah pencarian

ulang (*re-search*), maka simpul awal akan diganti dengan simpul yang memiliki derajat terkecil berikutnya".

Analisis Hasil

Keterbatasan, tantangan, dan syarat dalam penerapan algoritma Warnsdorff pada sebuah graf adalah sebagai berikut:

a) Algoritma Warnsdorff bekerja sangat baik pada graf yang memiliki struktur simpul mirip dengan papan catur, yaitu ketika setiap simpul memiliki jumlah tetangga yang hampir sama atau tidak terlalu bervariasi.
b) Graf yang hanya memiliki Lintasan Hamilton, tidak semua berhasil menemukan lintasannya, posisi awal menentukan keberhasilan. Pemilihan simpul dengan derajat terkecil, algoritma cenderung memiliki lebih banyak kemungkinan langkah untuk dieksplorasi di tahap awal, yang dapat meningkatkan peluang menemukan lintasan Hamilton.
c) Dengan selalu memilih simpul berikutnya yang memiliki derajat terkecil, algoritma ini dapat meminimalkan potensi terjebak pada simpul yang sulit untuk dilalui. Jika terdapat jumlah derajat terkecil yang sama maka pilih secara acak, hal ini ada kemungkinan menemukan atau tidak menemukan lintasan, untuk meningkatkan keberhasilan dengan memperkenalkan mekanisme pencarian ulang (*re-search*) memiliki peluang lebih besar untuk menemukan lintasan Hamilton dibandingkan dengan algoritma Warnsdorff asli.
d) Saat diimplementasikan algoritma Warnsdorff pada Gambar 2 graf G_4 memperlihatkan bahwa jika diawali simpul dengan derajat terkecil tidak selalu mendapatkan solusi, oleh karena itu pada algoritma Warnsdorff pentingnya mekanisme pencarian ulang (*re-search*) dan mengganti simpul awal dengan simpul terkecil berikutnya.
e) Keterbatasannya sebagai heuristik menunjukkan bahwa algoritma ini tidak selalu dapat menjamin solusi. Namun, dalam banyak kasus, seperti kasus contoh graf sebelumnya, algoritma Warnsdorff yang dimodifikasi mampu menemukan lebih banyak lintasan Hamilton dibandingkan versi aslinya yang belum dimodifikasi. Peningkatan ini dicapai melalui perbaikan yang mengatasi beberapa kelemahan, sehingga menghasilkan kinerja yang lebih baik.

Dengan analisis hasil yang telah dipaparkan maka algoritma Warnsdorff dapat dimodifikasi menjadi:

a) Pilih simpul awal $v_i \in V$ yang memiliki derajat terkecil.
   $T := \emptyset$
   $V := \{v_i | v_i$ adalah simpul dari graf$\}$
   Pilih simpul dengan $d(v_i) = \min \{d(v_i)\}_{v_i \in V}$
b) Tandai v_i sebagai simpul yang telah dikunjungi.
   $T := T \cup \{v_i\}$
   $V := V - \{v_i\}$
c) Selama $|N(v_i)| \neq 0$, ulangi proses berikut!

i. $\forall v_j \in N(v_i)$:
   $d'(v_j) \coloneqq d(v_j) - |T \cap N(v_j)|$
ii. Pilih simpul dengan $d'(v_j) = \min\{d'(v_j)\}_{v_j \in N(v_i)}$
iii. Tandai v_j sebagai simpul yang telah dikunjungi:
   $T \coloneqq T \cup \{v_j\}$
   $V \coloneqq V - \{v_j\}$
iv. Perbarui simpul saat ini:
   Perbarui $N(v_i) = N(v_i) - \{v_j\}$
   $v_i \coloneqq v_j$

d) Periksa lintasan Hamilton.
   Jika $|V| = 0$ maka ada lintasan Hamilton.
   Jika $|V| \neq 0$
   - Lakukan Pencarian Ulang (*Re-search*)
     - Kembali ke langkah c poin ii, pilih simpul lain pada saat pemilihan simpul yang memilih jumlah derajat terkecil yang sama
     - Jika lintasan Hamilton masih tidak ditemukan.
       - Coba simpul awal lain dengan derajat terkecil berikutnya.
       - Ulangi proses.
   - Jika tidak ada lintasan Hamilton setelah pencarian ulang maka tidak ada lintasan Hamilton.

Berikut diberikan Gambar 4, pseudocode dari algoritma Warnsdorff yang telah dimodifikasi dalam Python:

```
1. Inisialisasi
        T = set()      V = {semua_simpul}
2. Pilih simpul awal dengan derajat terkecil
        v_i = pilih_simpul_berderajat_terkecil(V)
3. Tandai simpul awal sebagai telah dikunjungi
        T.add(v_i)      V.remove(v_i )
4. Ulangi sampai semua simpul telah dikunjungi
        while len(V) ! = 0:
            N(v_i) = tetangga(v_i)
            for v_j in N(v_i):
                d_prim_v_j = derajat(v_j) - len(T.intersection(tetangga(v_j)))
            v_j = pilih_simpul_berderajat_terkecil(N(v_i))
            T.add(v_j)
            V.remove(v_j)
            v_i = v_j
5. Periksa lintasan Hamilton
if len(T) == len(semua_simpul):
    print("Lintasan Hamilton ada")
else:
    while True:
        v_j_backtrack = pilih_simpul_berderajat_terkecil_berikutnya(N(v_i))
        if v_j_backtrack is None:
            break
        T.remove(v_j)
        V.add(v_i)
        v_i = v_j_backtrack
        T.add(v_i)
        V.remove(v_i)
# Lanjutkan proses seperti sebelumnya
6. Selesai
    Jika v = 0, maka ada lintasan Hamilton
    Jika v ≠ 0, maka tidak ada lintasan Hamilton
```

Gambar 4. Pseudocode Algoritma Warnsdorff Modifikasi

Berikut adalah output dari implementasi program Python pada graf berdasarkan algoritma Warnsdorff modifikasi:
Hasil graf G_2 menggunakan program python yaitu
Lintasan Hamilton ada: ['v10', 'v8', 'v3', 'v6', 'v2', 'v4', 'v9', 'v11', 'v5', 'v1', 'v7']
Hasil graf G_4 menggunakan program python yaitu
Lintasan Hamilton ada: ['v1', 'v5', 'v4', 'v2', 'v6', 'v7', 'v3', 'v8', 'v9', 'v10']

Tabel 5. Perbandingan Kompleksitas pada graf G_4

Simpul Awal	Algoritma Warnsdorff	Algoritma Warnsdorff Modifikasi
v_1	Waktu: 0.002177 detik Berhasil	Waktu: 0.002177 detik Berhasil
v_2	Waktu: 0.000449 detik Tidak Berhasil	Waktu: 0.003041 detik Berhasil
v_3	Waktu: 0.00471 detik Tidak Berhasil	Waktu: 0.004309 detik Berhasil
v_4	Waktu: 0.002213 detik Berhasil	Waktu: 0.002213 detik Berhasil
v_5	Waktu: 0.000402 detik Tidak Berhasil	Waktu: 0.003292 detik Berhasil
v_6	Waktu: 0.00472 detik Tidak Berhasil	Waktu: 0.002192 detik Berhasil
v_7	Waktu: 0.000490 detik Tidak Berhasil	Waktu: 0.003403 detik Berhasil
v_8	Waktu: 0.000520 detik Tidak Berhasil	Waktu: 0.004181 detik Berhasil
v_9	Waktu: 0.000548 detik Tidak Berhasil	Waktu: 0.003070 detik Berhasil
v_{10}	Waktu: 0.002947 detik Berhasil	Waktu: 0.002947 detik Berhasil

Dari Tabel 5, menunjukkan bahwa waktu eksekusi algoritma yang dimodifikasi cenderung lebih lama daripada yang belum dimodifikasi, karena mungkin algoritma dimodifikasi menggunakan lebih banyak strategi misalnya pencarian ulang. Karena algoritma yang dimodifikasi lebih sering berhasil menemukan lintasan Hamilton dibandingkan dengan yang belum dimodifikasi, maka modifikasi tersebut dapat dianggap sebagai peningkatan meskipun memerlukan lebih banyak waktu

KESIMPULAN

Algoritma Warnsdorff bisa diterapkan dalam menemukan lintasan Hamilton pada graf dengan baik. Keberhasilannya sangat bergantung pada simpul awal yang dipilih dan pemilihan simpul berikutnya ketika terdapat derajat yang sama. Pemilihan simpul awal dan pemilihan simpul berikutnya ketika terdapat derajat yang sama yang tepat pada algoritma Warnsdorff mampu menemukan lintasan Hamilton di berbagai graf yang diuji. Namun, ada beberapa kasus di mana pilihan simpul awal dan pemilihan simpul berikutnya ketika terdapat derajat sama yang

tidak tepat menyebabkan algoritma tidak berhasil menemukan lintasan Hamilton. Untuk mengatasi hal ini, diperlukan beberapa modifikasi pada algoritma, seperti pencarian ulang (*re-search*) dan memilih simpul awal dengan derajat terkecil berikutnya jika pilihan pertama gagal. Modifikasi ini terbukti membantu dalam menemukan lintasan Hamilton, terutama pada graf yang lebih rumit. Jadi, algoritma Warnsdorff merupakan metode yang efektif untuk menemukan lintasan Hamilton pada graf, namun pemilihan simpul awal dan pemilihan simpul berikutnya ketika terdapat derajat sama yang tepat dan penyesuaian algoritma sangat penting untuk memastikan keberhasilannya dalam menemukan lintasan Hamilton pada graf. Implikasi praktis dari penelitian ini adalah bahwa algoritma Warnsdorff, dapat diterapkan dalam berbagai bidang, seperti perencanaan rute atau jaringan optimal, pengendalian robot, yang memerlukan pencarian lintasan Hamilton secara efisien. Untuk pengembangan lebih lanjut, disarankan untuk mengkombinasikan algoritma Warnsdorff dengan teknik pencarian lainnya, seperti algoritma backtracking, algoritma genetika, atau algoritma berbasis kecerdasan buatan. Pendekatan semacam ini diharapkan dapat meningkatkan tingkat keberhasilan dan fleksibilitas algoritma dalam menghadapi graf dengan ukuran besar dan struktur kompleks.

REFERENSI

Fauziyah, W. N., Permanasari, Y., Matematika, R.P. (2020). Aturan Warnsdorff dan Algoritma Backtracking pada Permainan The Knight's Tour. *SPeSIA*. Vol. 6(1), 9–14.

Ihsan, A., Arif Adlie, T., & Harliansyah, S. (2024). Optimalisasi Pencarian Jalur Terpendek Mobile Robot dengan Menggunakan Metode Ant Colony Optimization (ACO). *Jurnal Ilmiah Elektroteknika*. Vol. 23(1), 39–54.

Jin, D., Li, Q., & Lu, M. (2022). Heuristic Approaches for Hamiltonian Path and Cycle Problems in Large-Scale Graphs. *Wireles Networks*. Vol. 28(1), 979–989.

Joni, & Erwin. (2017). Analisis Teori Graf Pada Persoalan Knight's Tour Dengan J2ME. *TIMES*. Vol. 6(2), 52–57.

Mahmudah, M., & Irawati, T. N. (2018). Aplikasi Pewarnaan Graf Terhadap Pembuatan Jadwal Ujian Semester di Jurusan Pendidikan Matematika Universitas Islam Jember. *Kadikma*. Vol. 9(2), 12–21.

Olsen, E., & Babicheva, T. (2024). On the convergence of the Warnsdorff's algorithm on the rectangular boards. *Procedia Computer Science* . Vol. 242 (1), 722–728. DOI:10.1016/j.procs.2024.08.153

Pegg, E. (2009). The Icosian Game, Revisited. *The Mathematica Journal*. Vol. 11(3), 310–314.

Pohl, I. (1967). A Method for Finding Hamilton Paths and Knight's Tour. *Communications of the ACM*. Vol. 10(7), 446–449.

Pranav, M., Nithin, S., & Guruprasad, N. (2019). A Comparison of Warnsdorff's Rule and Backtracking for Knight's Tour on Square Boards. *Lecture Notes in Electrical Engineering*. Vol. 545(1), 171–185.

Rozi, S., & Multahadah, C. (2021). Rute Terpendek untuk Pengangkutan Sampah dengan Pendekatan Lintasan Hamilton. *E-Jurnal Matematika*. Vol. 10(2), 115.
https://doi.org/10.24843/mtk.2021.v10.i02.p330

Serdano, A., Zarlis, M., & Hartama, D. (2019). Implementasi Algoritma Backtracking pada Knight's Tour Problem. *SeNTIK*. Vol. 3(1), 179–184.

Wibawa, C. (2022). Optimasi Rute Wisata di Yogyakarta Menggunakan Metode Travelling Salesman Person dan Algoritma Brute Force. *Jurnal Teknik Dan Science*. Vol. 1(3), 59–65.

Wijaya, E. (2016). Penerapan Sirkuit Hamilton untuk Menentukan Rute Terpendek Perjalanan Salesman PT Health Wealth Internasional (HWI). *Jurnal TIMES*. Vol. 5(1), 17–19.

Yuniar, N., & Panjaitan, D. J. (2024). Implementasi *Nearest Neighbor Methods* dan *Closed Insertion Methods* pada Penentuan Rute Bus Trans Metro Deli Menuju Pusat Kota Medan. *Jurnal Lebesgue : Jurnal Ilmiah Pendidikan Matematika, Matematika Dan Statistika*. Vol. 5(3), 1958–1978.
https://doi.org/10.46306/lb.v5i3.789

Zahro, F. (2018). Indeks Harary Graf Hamilton, Semi-Hamilton dan Hamilton-Kuat. *MATHunesa*. Vol. 6(2), 16–20.