

# Analisis Komparatif Strategi Algoritma Pencarian Dalam Penyelesaian Masalah Kecerdasan Buatan

<sup>1</sup>Budiansyah, <sup>2</sup>Iyat Ratna Komala, <sup>3</sup>Leni Nurhayati

<sup>1,2,3</sup>Universitas Sebelas April

<sup>1,2,3</sup>Jl. Angkrek Situ No.19, Sumedang, Jawa Barat 45323

email : [1240160121013@student.unsap.ac.id](mailto:1240160121013@student.unsap.ac.id), [iratna@unsap.ac.id](mailto:iratna@unsap.ac.id), [leni@unsap.ac.id](mailto:leni@unsap.ac.id)

## ABSTRACT

Search algorithms are a fundamental component in computer science, specifically in the domain of Artificial Intelligence (AI) for solving state space search problems. This study aims to conduct a comparative analysis between Uninformed Search strategies (BFS, DFS) and Informed Search strategies (A\*, Hill Climbing, Simulated Annealing). The research method used is a Systematic Literature Review (SLR) by synthesizing data from primary and secondary sources. The results indicate a significant trade-off; Uninformed Search such as BFS guarantees optimality but has high space complexity  $O(b^d)$  while DFS is memory efficient but not complete. Conversely, Informed Search significantly increases efficiency, requiring only about 4.45% of computation compared to blind search. The A\* algorithm is identified as the most effective strategy for pathfinding by balancing actual cost and heuristic estimation, whereas Simulated Annealing overcomes the local optima problem found in Hill Climbing. The selection of the right algorithm depends on the specific constraints of the problem faced.

**Keywords** - Algorithm, Artificial Intelligence, Heuristic Search, State Space Search

## 1. Introduction

Pencarian (*Searching*) merupakan salah satu proses paling fundamental dan esensial dalam ilmu komputer. Signifikansi dari proses ini terletak pada perannya sebagai operasi prasyarat (*pre-requisite*) bagi sebagian besar aktivitas pengolahan data lainnya[1-3]. Sebagai contoh, dalam proses penambahan data baru, pencarian harus dilakukan terlebih dahulu untuk memastikan tidak ada duplikasi; proses penambahan hanya dilanjutkan jika pencarian mengonfirmasi ketiadaan data tersebut[4]. Evolusi konsep pencarian terbagi menjadi dua paradigma: *Data Retrieval* dan *State Space Search*. *Data Retrieval* berfokus pada optimasi akses data dalam dataset terstruktur[5].

Sedangkan *State Space Search* adalah inti dari Kecerdasan Buatan (AI) di mana tujuannya menemukan urutan tindakan dari keadaan awal ke tujuan, yang dimodelkan dalam sebuah Graf[4]. Pada dataset besar, algoritma pencarian buta (*blind search*) dengan kompleksitas  $O(b^d)$  menjadi tidak praktis, sehingga memaksa pengembangan strategi heuristik untuk memangkas ruang pencarian secara efektif[6-9]. Dalam konteks *Big Data*, algoritma yang tidak efisien bahkan tidak dapat digunakan sama sekali[10].

Berbagai algoritma memiliki relevansi spesifik. Algoritma A\* (*A-Star*) diakui sebagai standar industri untuk *pathfinding* karena kemampuannya memberikan solusi optimal (rute terpendek) secara efisien [11-14]. Sementara itu, *Depth-First Search* (DFS) fundamental dalam analisis topologi jaringan seperti deteksi siklus [15] dan penemuan *Strongly Connected Components* [16]. Pada penyelesaian *puzzle*, BFS lebih unggul dalam pencarian terstruktur untuk menemukan semua kemungkinan solusi jika jawaban lebih dari satu [17-19].

Tantangan utama penelitian ini adalah mengatasi kesenjangan performa dan *trade-off*. Algoritma *Uninformed* seringkali tidak efisien, sementara studi empiris menunjukkan *Informed Search* jauh lebih unggul, hanya membutuhkan sekitar 4,45% komputasi dari algoritma buta [20]. Selain itu, terdapat masalah pada algoritma heuristik "rakus" seperti *Hill Climbing* yang berisiko terjebak dalam *local optima* sehingga gagal menemukan solusi global [21-24]. Pemilihan algoritma yang tepat sangat krusial untuk domain spesifik seperti *pathfinding* [25].

## 2. Research Method

Penelitian ini menggunakan pendekatan kualitatif dengan metodologi Studi Literatur Sistematis (*Systematic Literature Review*) untuk mengidentifikasi *trade-off* fundamental dan konteks aplikasi yang relevan dari setiap strategi algoritma [26]. Sumber data terdiri dari sumber primer berupa materi perkuliahan sebagai dasar pembandingan [27-30], dan sumber sekunder berupa jurnal serta dokumentasi teknis yang menyediakan analisis mendalam tentang *heuristic search* dan studi kasus. Adapun analisis data dilakukan melalui tiga tahap:

1. **Analisis Deskriptif:** Mengekstrak mekanisme kerja, *pseudo-code*, serta kelebihan dan kelemahan algoritma.
2. **Analisis Komparatif:** Membandingkan *head-to-head* berdasarkan parameter standar: *Completeness*, *Optimality*, *Time Complexity*, dan *Space Complexity*.
3. **Analisis Kontekstual:** Menelaah hasil studi kasus implementasi untuk memahami implikasi praktis di domain spesifik.

## 3. Result and Analysis

Bagian ini membahas hasil perbandingan strategi algoritma berdasarkan literatur yang telah dianalisis. Dimulai dari Analisis Komparatif Algoritma *Uninformed Search* yaitu dengan bekerja tanpa informasi tambahan (buta). Selanjutnya *Breadth-First Search (BFS)*, yaitu Algoritma yang menjamin *completeness* dan *optimality* (jika biaya seragam). Namun, kelemahan utamanya adalah *space complexity* yang boros  $O(b^d)$  karena harus menyimpan semua node di perbatasan. Terakhir menggunakan *Depth-First Search (DFS)*. Algoritma ini digunakan karena memiliki keunggulan utamanya adalah *space complexity* yang sangat hemat  $O(d)$ . Namun, kelemahannya adalah tidak *complete* (bisa terjebak dalam *loop*) dan tidak menjamin *optimality*. Berikut perbandingan kinerja kedua algoritma ini dapat dilihat pada Tabel 1.

Tabel 1. Perbandingan Kinerja Algoritma *Uninformed Search*

Kriteria	Breadth-First Search (BFS)	Depth-First Search (DFS)
Prinsip Kerja	Mengeksplorasi <i>level-by-level</i>	Mengeksplorasi <i>path-by-path</i> (mendalam)
Struktur Data	Queue (FIFO)	Stack (LIFO)
Completeness	Ya (jika <i>branching factor</i> terbatas)	Tidak (bisa terjebak <i>infinite loop</i> )
Optimality	Ya (jika biaya <i>edge</i> seragam)	Tidak
Time Complexity	$O(b^d)$	$O(b^d)$
Space Complexity	$O(b^d)$ (Buruk/Boros)	$O(b \times d)$ (Baik/Hemat)

Algoritma pencarian yang menggunakan fungsi heuristik atau disebut sebagai *informed search* memiliki efisiensi yang sangat tinggi, di mana studi menunjukkan metode ini hanya memerlukan sekitar 4,45% komputasi dan 5,33% memori dibandingkan dengan *uninformed search*. Terdapat beberapa varian algoritma dalam kategori ini dengan karakteristik yang berbeda, mulai dari *Generate and Test* yang kurang efisien, hingga *Hill Climbing* yang hemat memori namun rentan terhadap masalah *local optima*. Untuk mengatasi kelemahan tersebut, algoritma *Simulated Annealing (SA)* memungkinkan

eksplorasi ruang pencarian yang lebih global, sementara Best-First Search (BeFS) memprioritaskan eksplorasi pada node yang dianggap terbaik. Penggunaan fungsi heuristik dalam metode informed search secara signifikan meningkatkan efisiensi pencarian, dengan studi menunjukkan hanya dibutuhkan 4,45% komputasi dan 5,33% memori dibandingkan dengan uninformed search. Terdapat beberapa strategi utama dalam kategori ini, di antaranya adalah Generate and Test yang cenderung tidak efisien karena hanya praktis untuk ruang pencarian kecil. Strategi lainnya meliputi Hill Climbing yang merupakan pendekatan local search hemat memori namun mudah terjebak dalam local optima, serta Simulated Annealing (SA) yang dirancang untuk mengatasi masalah tersebut melalui gerakan probabilistik guna mengeksplorasi ruang pencarian secara lebih global. Sementara itu, Best-First Search (BeFS) menggunakan priority queue untuk mengeksplorasi node terbaik namun tidak menjamin optimalitas. Terakhir, algoritma A\* (A-Star) hadir sebagai sintesis sempurna yang menyeimbangkan biaya aktual  $g(n)$  dan estimasi heuristik  $h(n)$ , sehingga mampu menjamin optimalitas asalkan fungsi heuristik yang digunakan bersifat admissible. Sebagai puncaknya, algoritma A\* (A-Star) hadir sebagai sintesis yang menyeimbangkan antara biaya aktual dan estimasi heuristik, sehingga mampu menjamin optimalitas selama fungsi heuristik yang digunakan bersifat admissible. Perbandingan strategi heuristik disajikan dalam Tabel 2.

**Tabel 2. Perbandingan Strategi Algoritma Heuristic Search**

Kriteria	Hill Climbing	Simulated Annealing	Greedy Best-First	A* (A-Star)
Mekanisme	<i>Local search</i>	<i>Probabilistic local search</i>	<i>Global search</i>	<i>Global search</i>
Fungsi Evaluasi	$f(n) = h(n)$	Fungsi Energi + Prob. $P(T)$	$f(n) = h(n)$	$f(n) = g(n) + h(n)$
Optimalitas	Tidak	Probabilistik	Tidak	Ya (jika $h(n)$ admissible)
Local Optima	Buruk (Terjebak)	Sangat Baik	Bisa terjebak	Menjamin optimal

Penerapan fungsi heuristik dalam metode pencarian terbukti sangat efektif karena hanya memerlukan sumber daya komputasi dan memori yang sangat kecil dibandingkan metode konvensional. Di antara berbagai strategi yang ada, Hill Climbing menjadi pilihan paling hemat memori karena mekanisme local search-nya hanya fokus pada tetangga terbaik ( $f(n) = h(n)$ ), meski berisiko tinggi terjebak dalam local optima. Untuk menutupi kelemahan tersebut, Simulated Annealing (SA) menggunakan pendekatan probabilistik yang memungkinkan algoritma keluar dari jebakan lokal guna mengeksplorasi ruang pencarian secara global. Sementara itu, Greedy Best-First Search memprioritaskan kecepatan dengan mengeksplorasi node yang dianggap paling menjanjikan, namun tetap tidak menjamin hasil yang paling optimal. Sebagai solusi paling komprehensif, Algoritma A\* (A-Star)\* menggabungkan biaya jalur aktual  $g(n)$  dengan estimasi heuristik  $h(n)$  untuk menciptakan keseimbangan yang menjamin optimalitas selama fungsi heuristiknya bersifat admissible atau tidak pernah melebihi-lebihkan biaya aslinya.

#### 4. Conclusion

Berdasarkan analisis komparatif yang telah dilakukan, dapat disimpulkan bahwa pemilihan strategi pencarian harus disesuaikan dengan karakteristik masalah. Algoritma *Uninformed Search* seperti BFS menjamin solusi optimal namun tidak layak untuk ruang pencarian besar karena boros memori. DFS menawarkan efisiensi memori namun berisiko tidak menemukan solusi. Solusi untuk masalah kompleksitas tersebut adalah penggunaan *Informed Search*. Algoritma A\* terbukti menjadi solusi paling *robust* untuk pencarian jalur karena menjamin optimalitas dengan efisiensi waktu yang lebih baik dibandingkan pencarian buta. Sementara itu, untuk masalah optimasi di mana memori sangat terbatas,

*Simulated Annealing* memberikan alternatif yang lebih baik daripada *Hill Climbing* karena kemampuannya menghindari *local optima*. Saran untuk pengembangan selanjutnya adalah penerapan algoritma hibrida yang menggabungkan kecepatan *Greedy* dan ketepatan  $A^*$  dalam skenario dinamis *real-time*.

## References

- [1] W. Kustiawan, U. Sebelas, and A. Sumedang, "Pengamatan tentang optimalisasi algoritma pencarian dalam pemrosesan data besar".
- [2] D. Markuci and C. Prianto, "Analisis Perbandingan Penggunaan Algoritma Sequential Analisis Perbandingan Penggunaan Algoritma Sequential Search Dan Binary Search Pada Aplikasi Surat Perjalanan Dinas," vol. 6, no. 1, pp. 110–119, 2022.
- [3] Daswanta kumar routhu, "State Space Search in AI," [geeksforgeeks.org](https://www.geeksforgeeks.org). Accessed: Nov. 10, 2025. [Online]. Available: <https://www.geeksforgeeks.org/artificial-intelligence/state-space-search-in-ai/>
- [4] W. Hidayat, F. Susanti, D. R. Wijaya, and W. Hidayat, "A Comparative Study of Informed and Uninformed Search Algorithm to Solve Eight-Puzzle Problem," 2021, doi: 10.3844/jcssp.2021.1147.1156.
- [5] Rahmayani, R., Firmansyah, E., & Hikmah, H. U. (2025). Inovasi Layanan Antar Jemput Paket Surat PTPos Indonesia Berdasarkan Penjualan dan Minat Beli JOVISHE: Journal of Visionary Sharia Economy,4(1), 33-47.
- [6] Firmansyah, E., Rahman, A. B. A., & Subiyakto, A. A. (2023). Pengukuran Kesiapan Kota Cerdas Berdasarkan SNI ISO 37122: 2019. *Infoman's: Jurnal Ilmu-ilmu Informatika dan Manajemen*,17(2).
- [7] Zulfikar, W.B., Irfan, M., Ghufroon, M., Jumadi, J., & Firmansyah, E. (2020). Marketplace affiliates potential analysis using cosine similarity and vision-based page segmentation. *Bulletin of Electrical Engineering and Informatics*,9(6), 2492-2498.
- [8] Ramadhan, N. D., Fhatturohmah, S., Ramadhani, S., & Firmansyah, E. (2023). Analysis of Digital Wallet Usage on Consumptive Lifestyle. *Journal of Islamic Economics and Business*,3(2), 118-136.
- [9] Sutara, B., & Firmansyah, E. (2021). Design and Build Student Attendance System Using Fingerprint. *J-Tin's-Jurnal Teknik Informatika*,5(1).
- [10] Wahana, A., Firmansyah, E., Al Rosyid, H. I., Fuadi, R. S., & Maylawati, D. S. A. (2021). Fuzzy Tahani Method in the Recommendation System for Selecting Mountain Tourism Destinations in West Java.
- [11] Tamrin, M. A., Rizki, B., Nodas, A., Rahman, A., & Firmansyah, E. (2020). Perbandingan Penggunaan Metode Topsis dan Metode AHP dalam Penilaian Kinerja pada Karyawan (PT XYZ). *Infoman's: Jurnal Ilmu-ilmu Informatika dan Manajemen*,14(1).
- [12] Firmansyah, E., Herdiana, D., & Yuniarto, D. (2020, October). Examining readiness of e-Learning implementation using information system readiness impact model. In *2020 8th International Conference on Cyber and IT Service Management (CITSM)* (pp. 1-5). IEEE.
- [13] Firmansyah, E., Rosmawati, R., Fuadi, R. S., Fauzy, D., & Ramdhani, M. A. (2019, December). Design of expert system to determine the proper diet using harmony search method. In *Journal of Physics: Conference Series* (Vol. 1402, No. 7, p. 077006). IOP Publishing.
- [14] Firmansyah, E., & Helmiawan, M. A. (2025). Pengukuran Kesiapan Transformasi Digital Desa Kaduwulung Menuju Desa Cerdas Berbasis SNI ISO 37122: 2019 Melalui Pemetaan Data Desa. *Infoman's: Jurnal Ilmu-ilmu Informatika dan Manajemen*,19(1).
- [15] Firmansyah, E., Herdiana, D., Yuniarto, D., & Junaedi, D. I. (2021, September). The K-Nearest Neighbor Algorithm for the Classification of Internet Users in Rural Campus. In *2021 9th International Conference on Cyber and IT Service Management (CITSM)* (pp. 1-6). IEEE.
- [16] Zaliluddin, D., Bastian, A., Yuliani, M. S. S., Firmansyah, E., & Sumaryana, Y. (2024). Engaging Teens in History through a Mobile Game Utilizing the Fisher-Yates Shuffle Algorithm and Honeycomb UX Design. *International Journal of Interactive Mobile Technologies*, 18(22).

- [17] Syaripudin, U., Zaenal, R., Duri, M. F. A., Firmansyah, E., & Rahman, A. (2019, December). Comparison between Naïve Bayes and certainty factor to predict big five personality. In *Journal of Physics: Conference Series* (Vol. 1402, No. 7, p. 077030). IOP Publishing.
- [18] K. I. Lestari, S. Putra, E. Rilvani, U. P. Bangsa, and K. Bekasi, "Permasalahan Efisiensi Algoritma Apriori Dan Evaluasi Fp-Growth Berdasarkan Studi Literatur Evaluasi Fp-Growth Berdasarkan Studi Literatur," vol. 3, no. 7, 2025.
- [19] Divyan shuraj, "A\* algorithm and its Heuristic Search Strategy in Artificial Intelligence," *geeksforgeeks.org*. [Online]. Available: <https://www.geeksforgeeks.org/artificial-intelligence/a-algorithm-and-its-heuristic-search-strategy-in-artificial-intelligence/>
- [20] T. N. Lina *et al.*, "Comparison Analysis of Breadth First Search and Depth Limited Search Algorithms in Sudoku Game," vol. 2, no. 2, pp. 74–83, 2021, doi: 10.25008/bcsee.v2i2.1146.
- [21] T. M. Hutagalung, "Perbandingan Algoritma Floyd Warshall Dan Algoritma A\* Dalam Pencarian Rute Terpendek Menuju Lokasi Rumah Sakit Di Palembang," 2016.
- [22] P. Studi, T. Informatika, F. Teknologi, U. Kristen, D. Wacana, and A. Christian, "Studi literatur perbandingan algoritma dijkstra dan bellman-ford dalam pencarian jarak terdekat," 2013.
- [23] P. Pangestu and R. Novita, "Systematic Literature Review : Perbandingan Algoritma Klasifikasi," pp. 431–440, 2023.
- [24] Devendewc0m, "What is Simulated Annealing," *geeksforgeeks.org*. [Online]. Available: <https://www.geeksforgeeks.org/artificial-intelligence/what-is-simulated-annealing/>
- [25] M. A. B. Ade Candra, "Application of A-Star Algorithm on Pathfinding Game," 2021, doi: 10.1088/1742-6596/1898/1/012047.
- [26] R. Gayathri, "Comparative Analysis of Various Uninformed Searching Algorithms in AI," vol. 8, no. 6, pp. 57–65, 2019.
- [27] Payalcs, "Generate and Test Search," *geeksforgeeks.org*. [Online]. Available: <https://www.geeksforgeeks.org/machine-learning/generate-and-test-search/>
- [28] Ujjwal rawat, "Hill Climbing in Artificial Intelligence," *geeksforgeeks.org*. [Online]. Available: <https://www.geeksforgeeks.org/artificial-intelligence/introduction-hill-climbing-artificial-intelligence/>
- [29] S. Singh, "Best First Search (Informed Search)," *geeksforgeeks.org*. [Online]. Available: <https://www.geeksforgeeks.org/dsa/best-first-search-informed-search/>
- [30] Mygreatlearning, "Best First Search Algorithm in AI," *Mygreatlearning*. [Online]. Available: <https://www.mygreatlearning.com/blog/best-first-search-bfs/>