

Optimasi Rute Drone Multi-Objektif Menggunakan Algoritma A* dengan Heuristik Adaptif

Boye Mangaratua Ginting - 13523127

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10 Bandung

E-mail: boyeginting@gmail.com , 13523127@std.stei.itb.ac.id

Abstract— Navigasi otonom adalah komponen krusial untuk operasi *Unmanned Aerial Vehicle* (UAV) atau drone yang efisien dan aman. Algoritma pathfinding konvensional seperti A* sering digunakan, namun memiliki keterbatasan karena sifatnya yang *single-objective* dan umumnya hanya mengoptimalkan jarak. Hal ini tidak memadai untuk skenario dunia nyata yang menuntut pertimbangan simultan terhadap berbagai faktor seperti konsumsi energi dan keamanan sinyal. Makalah ini mengusulkan sebuah metode A* yang dimodifikasi dengan pendekatan multi-objektif dan heuristik adaptif. Metode ini menggunakan fungsi biaya tertimbang untuk menyeimbangkan jarak, energi, dan sinyal, serta fungsi heuristik yang perilakunya dapat berubah berdasarkan sisa energi drone. Hasil simulasi pada skenario yang dirancang khusus menunjukkan bahwa metode yang diusulkan mampu menghasilkan rute yang lebih aman dan andal. Meskipun rute yang ditempuh lebih panjang, total biaya tertimbang secara signifikan lebih rendah dan drone tiba di tujuan dengan sisa energi yang jauh lebih aman dibandingkan A* standar. Kontribusi penelitian ini adalah sebuah kerangka kerja *pathfinding* yang sadar-konteks, menghasilkan navigasi drone yang lebih cerdas dan tangguh.

Keywords— *Algoritma A*, Pathfinding, Drone, UAV, Optimisasi Multi-Objektif, Heuristik Adaptif*

I. PENDAHULUAN

Teknologi *Unmanned Aerial Vehicle* (UAV) atau drone telah mengalami perkembangan pesat, menjadikannya instrumen vital di berbagai sektor industri seperti logistik, survei pertanian, hingga operasi pencarian dan penyelamatan. Efektivitas operasi ini sangat bergantung pada kemampuan drone untuk melakukan navigasi otonom dalam lingkungan yang kompleks dan dinamis. Fondasi dari navigasi otonom adalah algoritma pencarian rute (*pathfinding*) yang efisien. Di antara berbagai pendekatan yang ada, Algoritma A* menjadi pilihan utama karena kemampuannya menyeimbangkan antara keoptimalan rute dengan efisiensi komputasi.

Meskipun demikian, keunggulan Algoritma A* konvensional terbatas pada masalah *single-objective*, yang umumnya hanya bertujuan menemukan jarak terpendek. Pendekatan ini tidak lagi memadai untuk operasi drone di dunia nyata yang menuntut pertimbangan simultan terhadap berbagai faktor kritis. Rute terpendek bisa jadi mengorbankan konsumsi energi akibat kondisi angin, mengabaikan keamanan sinyal komunikasi di area tertentu, atau bahkan melintasi zona berbahaya. Kesenjangan antara keterbatasan algoritma *single-*

objective dengan tuntutan operasi drone yang multifaset inilah yang menjadi motivasi utama penelitian ini.

Untuk menjawab tantangan tersebut, makalah ini mengusulkan sebuah metode A yang dimodifikasi untuk bekerja dalam kerangka multi-objektif. Pendekatan yang diajukan memiliki dua pilar utama: pertama, penggunaan fungsi biaya yang mempertimbangkan secara simultan tiga metrik krusial: jarak, estimasi energi, dan kualitas sinyal. Kedua, dan yang menjadi inovasi inti, adalah pengembangan fungsi heuristik yang bersifat adaptif. Heuristik ini mampu menyesuaikan strateginya secara dinamis berdasarkan kondisi drone, seperti sisa daya baterai. Hal ini memungkinkan algoritma untuk secara cerdas memprioritaskan aspek yang paling kritis pada waktu yang tepat, menghasilkan rute yang tidak hanya pendek, tetapi juga efisien dan aman. *Kontribusi utama dari makalah ini adalah sebuah kerangka kerja A multi-objektif dengan heuristik adaptif untuk optimisasi rute drone yang lebih realistis dan andal.*

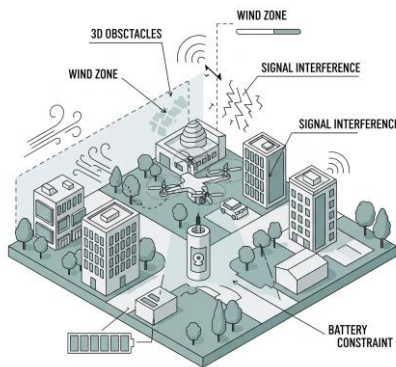
Makalah ini disusun dengan sistematika sebagai berikut. Bab 2 mengulas landasan teori mengenai Algoritma A* dan tinjauan penelitian terkait optimisasi rute multi-objektif. Bab 3 memerinci metodologi yang diusulkan, mencakup perancangan lingkungan simulasi, formulasi fungsi biaya, serta desain heuristik adaptif. Bab 4 menyajikan hasil dan analisis dari eksperimen yang dilakukan untuk memvalidasi performa metode yang diusulkan. Terakhir, Bab 5 merangkum kesimpulan penelitian dan memberikan arahan untuk pengembangan di masa depan.

II. DASAR TEORI

A. *Pathfinding* untuk *Unmanned Aerial Vehicle* (UAV)

Pathfinding atau pencarian rute merupakan proses komputasi untuk menentukan jalur optimal dari titik awal menuju titik tujuan dalam suatu lingkungan yang memiliki berbagai rintangan dan batasan. Dalam konteks navigasi drone otonom, *pathfinding* menjadi masalah fundamental yang menentukan keberhasilan misi dan efisiensi operasional sistem. Berbeda dengan kendaraan darat yang bergerak pada permukaan 2D, drone beroperasi dalam ruang 3D yang memberikan fleksibilitas pergerakan lebih besar namun juga kompleksitas komputasi yang signifikan lebih tinggi.

Pathfinding untuk UAV menghadapi tantangan unik yang tidak ditemukan pada aplikasi konvensional lainnya. Pertama, lingkungan operasi drone bersifat tiga dimensi dengan rintangan yang dapat berupa bangunan, pohon, atau zona larangan terbang yang memiliki ketinggian berbeda-beda. Kedua, drone seringkali beroperasi dalam lingkungan dinamis di mana rintangan dapat berubah posisi atau muncul secara tidak terduga, seperti pesawat lain atau kondisi cuaca buruk. Ketiga, keterbatasan sumber daya khususnya kapasitas baterai menjadi faktor kritis yang membatasi jarak tempuh dan waktu operasi drone. Keempat, faktor lingkungan seperti kecepatan dan arah angin sangat mempengaruhi konsumsi energi dan stabilitas penerbangan. Kelima, kualitas sinyal komunikasi antara drone dengan ground control station dapat bervariasi tergantung lokasi geografis dan adanya interferensi elektromagnetik.



Gambar 2.1 Ilustrasi Tantangan Pathfinding untuk Drone

(Sumber : Dihasilkan oleh AI generatif berdasarkan deskripsi penulis).

B. Algoritma A*

Algoritma A* (dibaca "A-star") adalah salah satu algoritma pencarian jalur terbaik yang paling populer dan efisien dalam bidang kecerdasan buatan dan robotika. Algoritma ini dikembangkan oleh Peter Hart, Nils Nilsson, dan Bertram Raphael pada tahun 1968 sebagai perbaikan dari algoritma Dijkstra dengan menambahkan komponen heuristik untuk mengarahkan pencarian menuju tujuan. A* bekerja dengan cara menjelajahi graf atau grid secara sistematis menggunakan strategi pencarian terbaik-pertama (best-first search) yang dipandu oleh fungsi evaluasi.

1) Mekanisme Dasar

A* mengoperasikan dua struktur data utama dalam proses pencariannya: Open list dan Closed list. Open list berisi kumpulan node yang telah ditemukan tetapi belum dievaluasi, diurutkan berdasarkan nilai fungsi evaluasi $f(n)$ dari yang terkecil hingga terbesar. Closed list berisi kumpulan node yang telah dievaluasi dan tidak akan dipertimbangkan lagi dalam iterasi selanjutnya. Pada setiap iterasi, algoritma memilih node dengan nilai $f(n)$ terkecil dari Open list untuk dievaluasi, kemudian mengeksplorasi semua node tetangga (*neighbor*) yang dapat dicapai dari node tersebut.

2) Formula Inti

Formula fundamental algoritma A* adalah $f(n) = g(n) + h(n)$, dimana n adalah node yang sedang dievaluasi dalam graf. $g(n)$ adalah biaya aktual (actual cost) yang telah dikeluarkan untuk mencapai node n dari node awal (start node). Nilai ini bersifat akurat dan sudah diketahui karena merupakan akumulasi biaya perjalanan yang telah dilalui. $h(n)$ adalah estimasi biaya heuristik (heuristic cost) untuk mencapai node tujuan (goal node) dari node n . Nilai ini bersifat perkiraan dan dihitung menggunakan fungsi heuristik tertentu. Dan $f(n)$ adalah total estimasi biaya untuk mencapai tujuan melalui node n , yang merupakan penjumlahan biaya aktual dan estimasi biaya yang tersisa. Nilai $f(n)$ digunakan sebagai prioritas untuk menentukan node mana yang akan dievaluasi selanjutnya, dengan prioritas tertinggi diberikan kepada node yang memiliki nilai $f(n)$ terkecil.

3) Fungsi Heuristik

Fungsi heuristik $h(n)$ memberikan estimasi biaya dari node current menuju node tujuan. Fungsi ini berfungsi untuk mengarahkan pencarian agar lebih fokus menuju tujuan daripada mengeksplorasi area yang kurang relevan. Kualitas fungsi heuristik sangat mempengaruhi efisiensi algoritma A*: heuristik yang baik akan mempercepat pencarian dengan mengurangi jumlah node yang perlu dievaluasi, sedangkan heuristik yang buruk dapat memperlambat atau bahkan menyestatkan pencarian.

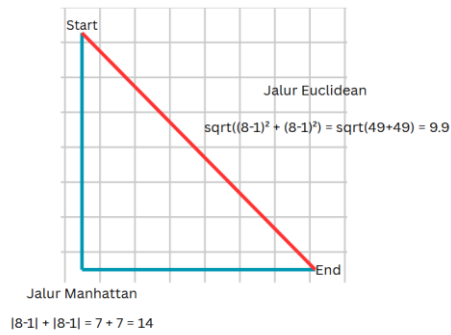
4) Contoh Heuristic Standar

Dalam implementasi konvensional, terdapat dua fungsi heuristik yang paling umum digunakan:

Jarak Manhattan cocok digunakan untuk pergerakan pada grid di mana pergerakan hanya diizinkan dalam arah horizontal dan vertikal (4-directional movement). Formula jarak Manhattan adalah: $h(n) = |x_1 - x_2| + |y_1 - y_2|$

Jarak Euclidean lebih sesuai untuk pergerakan bebas di mana objek dapat bergerak ke segala arah. Formula jarak Euclidean adalah: $h(n) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

Kedua heuristik standar ini hanya mempertimbangkan faktor jarak geometris dan mengabaikan faktor-faktor lain yang mungkin relevan seperti kondisi terrain, konsumsi energi, atau aspek keamanan.



Gambar 2.2 Ilustrasi Perbandingan Jarak Manhattan vs Euclidean

5) Sifat Optimalitas

Algoritma A* menjamin menghasilkan solusi optimal (jalur dengan biaya minimum) dengan syarat fungsi heuristik yang digunakan bersifat *admissible*. Suatu heuristik dikatakan *admissible* jika tidak pernah melebihi (overestimate) biaya sebenarnya untuk mencapai tujuan. Dengan kata lain, untuk setiap node n , nilai $h(n)$ harus selalu kurang dari atau sama dengan biaya aktual terpendek dari n ke tujuan. Jika syarat *admissible* terpenuhi, A* akan selalu menemukan jalur optimal jika solusi ada, atau melaporkan bahwa tidak ada solusi jika memang tidak ada jalur yang menghubungkan start dan goal.

C. Pendekatan Multi-Objektif dalam Pathfinding

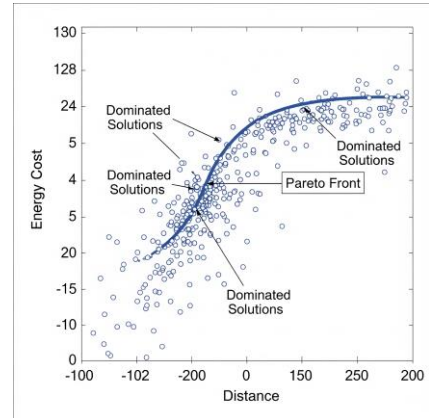
Multi-Objective Shortest Path Problem (MOSPP) adalah pengembangan dari masalah pencarian jalur terpendek klasik di mana optimasi dilakukan terhadap lebih dari satu kriteria secara bersamaan. Dalam konteks MOSPP, setiap edge atau arc dalam graf memiliki multiple cost attributes yang mewakili berbagai aspek yang ingin dioptimalkan, seperti jarak tempuh, waktu perjalanan, konsumsi bahan bakar, tingkat keamanan, atau kualitas layanan. Tujuan MOSPP adalah menemukan jalur yang memberikan trade-off terbaik di antara semua objektif yang dipertimbangkan.

Secara formal, MOSPP dapat didefinisikan sebagai masalah pencarian jalur P dari node sumber s ke node tujuan t dalam graf $G = (V, E)$ di mana setiap edge $e \in E$ memiliki vektor biaya $c(e) = [c_1(e), c_2(e), \dots, c_k(e)]$ dengan k adalah jumlah objektif. Biaya total jalur P dihitung sebagai penjumlahan vektor biaya semua edge yang dilalui: $C(P) = \sum c(e)$ untuk semua $e \in P$.

MOSPP memiliki kompleksitas yang secara signifikan lebih tinggi dibandingkan dengan single-objective shortest path problem karena beberapa alasan fundamental. Pertama, dalam masalah multi-objektif, seringkali tidak ada satu solusi tunggal yang optimal untuk semua kriteria secara bersamaan. Konsep ini dikenal sebagai *Pareto optimality*, di mana suatu solusi dikatakan Pareto optimal jika tidak ada solusi lain yang lebih baik untuk semua objektif secara bersamaan. Akibatnya,

MOSPP biasanya menghasilkan multiple solutions yang membentuk Pareto front.

Kedua, ruang pencarian dalam MOSPP menjadi jauh lebih besar karena algoritma harus mempertimbangkan kombinasi berbagai nilai objektif. Ketiga, tidak ada cara langsung untuk membandingkan dua jalur yang memiliki trade-off berbeda antar objektif. Misalnya, jalur A mungkin lebih pendek tetapi lebih boros energi, sementara jalur B lebih panjang tetapi lebih hemat energi.



Gambar 2.3 Diagram Konsep Pareto Optimality dan Pareto Front

(Sumber : Dihasilkan oleh AI generatif berdasarkan deskripsi penulis)

Untuk mengatasi kompleksitas MOSPP, telah dikembangkan berbagai pendekatan. Salah satu metode yang paling umum dan praktis adalah **weighted-sum method** atau metode pembobotan. Pendekatan ini mengubah masalah multi-objektif menjadi single-objective dengan cara memberikan bobot (weight) pada setiap kriteria dan menjumlahkannya menjadi satu nilai biaya agregat.

Secara matematis, weighted-sum method dapat diformulasikan sebagai:

$$f(n) = w_1 \times c_1(n) + w_2 \times c_2(n) + \dots + w_k \times c_k(n)$$

Di mana w_i adalah bobot untuk objektif ke- i , dan $\sum w_i = 1$. Pemilihan nilai bobot mencerminkan preferensi relatif terhadap setiap objektif: objektif dengan bobot lebih besar akan memiliki pengaruh lebih dominan dalam keputusan pathfinding.

Keunggulan weighted-sum method terletak pada kesederhanaannya yang memungkinkan penggunaan algoritma single-objective seperti A* tanpa modifikasi fundamental pada struktur algoritma. Metode ini juga memberikan kontrol yang intuitif kepada pengguna untuk menyesuaikan prioritas objektif sesuai dengan kebutuhan aplikasi spesifik. Namun, kelemahannya adalah ketidakmampuan untuk menemukan solusi Pareto optimal yang terletak pada bagian non-convex

dari Pareto front, dan sensitivitas hasil terhadap pemilihan nilai bobot.

III. METODE

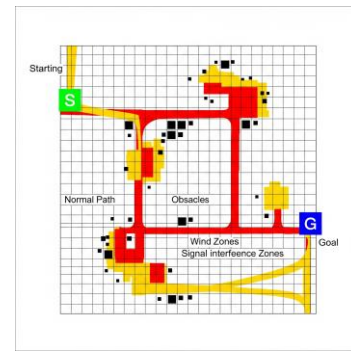
A. Perancangan Lingkungan Simulasi

Untuk memvalidasi metode yang diusulkan, sebuah lingkungan simulasi berbasis grid 2D dengan ukuran 20x20 sel dikembangkan. Penggunaan grid 2D merupakan simplifikasi yang umum dilakukan untuk fokus pada analisis logika inti algoritma pathfinding tanpa kompleksitas tambahan dari representasi 3D. Meskipun drone beroperasi dalam ruang tiga dimensi, representasi 2D tetap mampu menangkap aspek-aspek fundamental dari masalah multi-objektif pathfinding yang menjadi fokus penelitian ini.

Setiap sel dalam grid merepresentasikan suatu area geografis dan memiliki salah satu dari empat tipe keadaan sebagai berikut:

- Tipe 0 (Jalur Normal): Sel yang dapat dilalui dengan biaya standar. Sel ini merepresentasikan area terbuka yang aman untuk navigasi drone tanpa penalti khusus pada aspek jarak, energi, maupun sinyal.
- Tipe 1 (Rintangan/Obstacle): Sel yang tidak dapat dilalui sama sekali. Tipe ini merepresentasikan halangan fisik seperti gedung tinggi, menara, gunung, atau zona larangan terbang yang mutlak harus dihindari oleh drone.
- Tipe 2 (Zona Angin Kencang): Sel yang dapat dilalui namun memberikan penalti signifikan pada konsumsi energi drone. Area ini mensimulasikan kondisi lingkungan yang memaksa drone untuk bekerja lebih keras, seperti daerah dengan turbulensi udara tinggi, angin kencang, atau updraft/downdraft yang kuat.
- Tipe 3 (Zona Gangguan Sinyal): Sel yang dapat dilalui namun memberikan penalti pada aspek keamanan sinyal komunikasi. Zona ini merepresentasikan area dengan interferensi elektromagnetik tinggi, dead zone komunikasi, atau daerah dengan banyak obstacle yang menghalangi transmisi sinyal.

Dalam setiap skenario eksperimen, satu titik Awal (Start) dan satu titik Tujuan (Goal) ditentukan secara strategis. Rintangan serta zona-zona khusus ditempatkan secara sistematis di antara titik Awal dan Tujuan untuk menguji kemampuan algoritma dalam menavigasi dan membuat keputusan trade-off yang optimal. Penempatan elemen-elemen ini dirancang untuk menciptakan situasi di mana jalur terpendek secara geometris tidak selalu menjadi jalur terbaik secara keseluruhan, sehingga memaksa algoritma untuk melakukan optimasi multi-objektif.



Gambar 2.4 Contoh Peta Grid Simulasi

(Sumber : Dihasilkan oleh AI generatif berdasarkan deskripsi penulis)

B. Formulasi Fungsi Biaya Multi-Objektif

Berbeda dengan A* standar yang hanya menghitung jarak geometris, metode yang diusulkan menggunakan fungsi biaya multi-objektif dengan pendekatan weighted-sum method. Biaya pergerakan dari sel A ke sel B dihitung dengan formula komprehensif yang mempertimbangkan tiga aspek kritis dalam operasi drone:

$$\text{cost}(A,B) = w_{\text{jarak}} \times C_{\text{Jarak}} + w_{\text{energi}} \times C_{\text{Energi}} + w_{\text{Sinyal}} \times C_{\text{Sinyal}}$$

Komponen biaya jarak (C_{Jarak}) merepresentasikan cost geometris untuk berpindah sel. Implementasinya menggunakan sistem $C_{\text{Jarak}} = 1$ untuk berpindah horizontal atau vertikal (*4-directional movement*), dan $C_{\text{Jarak}} = 1,414$ ($\sqrt{2}$) untuk pergerakan diagonal (*8-directional movement*). Nilai ini konsisten dengan representasi jarak Euclidean pada grid diskrit dan memastikan bahwa jalur diagonal tidak mendapat keuntungan artifisial dibandingkan jalur yang menggunakan kombinasi gerakan horizontal-vertikal.

Komponen biaya energi (C_{Energi}) merepresentasikan konsumsi daya yang dibutuhkan untuk navigasi berdasarkan kondisi lingkungan. $C_{\text{Energi}} = 1$ untuk pergerakan di sel normal (Tipe 0) dan sel gangguan sinyal (Tipe 3). $C_{\text{Energi}} = 10$ untuk pergerakan menuju sel zona angin kencang (Tipe 2). Rasio penalti 10:1 dipilih berdasarkan pertimbangan bahwa navigasi dalam kondisi turbulensi dapat meningkatkan konsumsi energi secara signifikan, namun tidak hingga membuat jalur tersebut sama sekali tidak layak.

Komponen penalti sinyal (C_{Sinyal}) akan mengevaluasi risiko terhadap kualitas komunikasi drone dengan *ground control station*. $C_{\text{Sinyal}} = 0$ untuk pergerakan di sel normal (Tipe 0) dan sel zona angin kencang (Tipe 2). $C_{\text{Sinyal}} = 20$ untuk pergerakan menuju sel zona gangguan sinyal (Tipe 3). Nilai penalti yang lebih tinggi untuk aspek sinyal mencerminkan kriticalitas komunikasi dalam operasi drone, di mana kehilangan sinyal dapat berakibat fatal pada keseluruhan misi.

Terakhir untuk sistem pembobotan, yakni w_{Jarak} , w_{Energi} , dan w_{Sinyal} . Variabel w_{jarak} , w_{energi} , dan w_{sinyal} adalah bobot normalisasi yang merepresentasikan prioritas relatif dari setiap objektif. Nilai bobot ini berkisar antara 0 dan 1, dengan batasan $w_{\text{jarak}} + w_{\text{energi}} + w_{\text{sinyal}} = 1$.

= 1. Konfigurasi bobot memungkinkan adaptasi algoritma terhadap berbagai skenario operasional:

- Konfigurasi Seimbang (0.34, 0.33, 0.33): Memberikan prioritas yang relatif sama untuk semua aspek
- Konfigurasi Hemat Energi (0.2, 0.6, 0.2): Memprioritaskan efisiensi energi untuk misi jarak jauh
- Konfigurasi Komunikasi Kritis (0.2, 0.2, 0.6): Mengutamakan keandalan sinyal untuk misi yang memerlukan kontrol real-time

C. Desain Heuristik Adaptif

Inovasi utama dari metode ini terletak pada fungsi heuristik $h(n)$ yang bersifat adaptif dan mampu mengubah strategi estimasinya berdasarkan kondisi internal drone, khususnya level sisa energi. Pendekatan ini meniru perilaku pilot manusia yang akan mengubah strategi penerbangan ketika bahan bakar menipis, dari fokus pada efisiensi rute menjadi fokus pada keselamatan dan konservasi energi.

Sistem heuristik adaptif diimplementasikan dengan dua kondisi operasional utama berdasarkan threshold sisa energi drone:

- Kondisi Energi Aman (Sisa Energi > 50%)

Ketika level energi drone masih berada di atas 50% dari kapasitas maksimal, algoritma mengasumsikan bahwa drone masih memiliki fleksibilitas dalam pemilihan rute. Pada kondisi ini, prioritas pencarian adalah efisiensi jarak untuk memaksimalkan kinerja misi. Heuristik yang digunakan adalah Jarak Manhattan standar:

$$h(n) = |x_n - x_{goal}| + |y_n - y_{goal}|$$

Pemilihan Manhattan distance dibandingkan Euclidean distance didasarkan pada karakteristik pergerakan pada grid dan sifat admissible yang terjamin pada lingkungan dengan pergerakan 8-directional.

- Kondisi Energi Kritis (Sisa Energi ≤ 50%)

Ketika energi drone turun hingga 50% atau kurang, prioritas pencarian bergeser menjadi mode "self-preservation" untuk meminimalkan risiko kehabisan daya sebelum mencapai tujuan. Heuristik dimodifikasi dengan menambahkan komponen penalti yang memperkirakan risiko energi pada jalur yang akan dilalui:

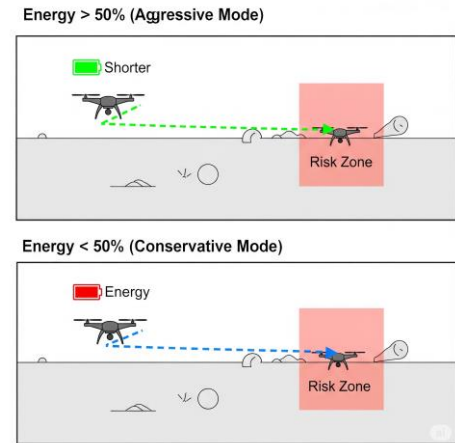
$$h(n) = \text{JarakManhattan}(n, \text{goal}) + w_{h_energi} \times \text{EstimasiRisikoEnergi}(n)$$

Fungsi EstimasiRisikoEnergi(n) adalah komponen heuristik tambahan yang menghitung potensi bahaya energi dari posisi saat ini (n) menuju tujuan. Implementasi fungsi ini menggunakan pendekatan *line-of-sight* yang menghitung jumlah sel zona angin kencang (Tipe 2) yang berada pada garis lurus imajiner antara posisi current dan posisi goal:

$$\text{EstimasiRisikoEnergi}(n) = \sum(\text{sel_angin_kencang pada garis } n \rightarrow \text{goal})$$

Variabel w_{h_energi} adalah bobot yang menentukan seberapa besar pengaruh "antisipasi risiko" terhadap keputusan

heuristik. Nilai w_{h_energi} yang lebih tinggi akan membuat algoritma lebih "takut" terhadap zona berbahaya dan cenderung mengambil jalur yang lebih aman meskipun mungkin lebih panjang.



Gambar 2.5 Ilustrasi Perubahan Perilaku Heuristik Adaptif
(Sumber : Dihasilkan oleh AI generatif berdasarkan deskripsi penulis)

D. Pseudocode Algoritma

Implementasi lengkap algoritma A* Multi-Objektif dengan heuristik adaptif dapat direpresentasikan dalam pseudocode berikut:

```
Function AStarMultiObjective(start, goal, map, weights):
    // Inisialisasi struktur data
    OPEN_list = {start}
    CLOSED_list = {}

    // Inisialisasi score untuk node awal
    g_score[start] = 0
    h_score[start] = calculate_h_adaptive(start, goal, drone_energy)
    f_score[start] = g_score[start] + h_score[start]

    while OPEN_list is not empty:
        // Pilih node dengan f_score terendah
        current = node in OPEN_list with the lowest f_score

        // Cek apakah tujuan telah tercapai
        if current == goal:
            return reconstruct_path(current)

        // Pindahkan current dari OPEN ke CLOSED
        remove current from OPEN_list
        add current to CLOSED_list

        // Evaluasi semua neighbor
        for each neighbor of current:
            if neighbor in CLOSED_list:
                continue // Skip node yang sudah dievaluasi

            if neighbor is obstacle (Type 1):
                continue // Skip obstacle

            // Hitung biaya menggunakan fungsi multi-objektif
```

```

        move_cost = calculate_cost_multi_objective(current, neighbor, weights)
        tentative_g_score = g_score[current] + move_cost

        // Update drone energy (simulasi konsumsi)
        update_drone_energy(move_cost)

        if neighbor not in OPEN_list:
            add neighbor to OPEN_list
        else if tentative_g_score >= g_score[neighbor]:
            continue // Jalur ini tidak lebih baik

        // Jalur ini adalah yang terbaik sejauh ini
        came_from[neighbor] = current
        g_score[neighbor] = tentative_g_score
        h_score[neighbor] = calculate_h_adaptive(neighbor, goal, drone_energy)
        f_score[neighbor] = g_score[neighbor] + h_score[neighbor]

        return failure // Tidak ada jalur yang ditemukan

Function calculate_cost_multi_objective(nodeA, nodeB, weights):
    C_jarak = calculate_distance_cost(nodeA, nodeB)
    C_energi = calculate_energy_cost(nodeA, nodeB)
    C_sinyal = calculate_signal_cost(nodeA, nodeB)

    return weights.jarak * C_jarak + weights.energi * C_energi + weights.sinyal * C_sinyal

Function calculate_h_adaptive(node, goal, energy_level):
    base_heuristic = manhattan_distance(node, goal)

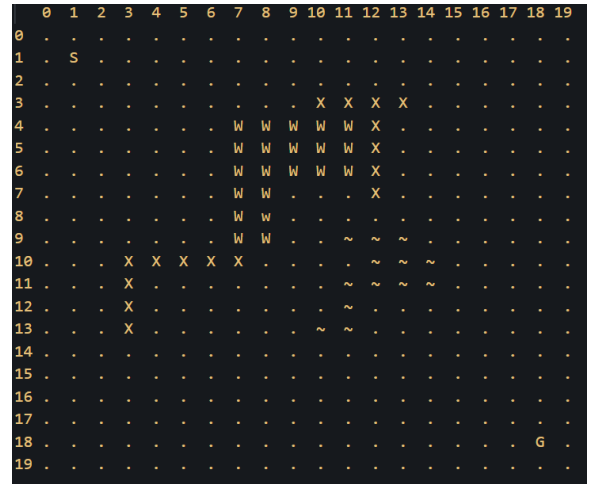
    if energy_level > 0.5: // Energi aman
        return base_heuristic
    else: // Energi kritis
        risk_penalty = estimate_energy_risk(node, goal)
        return base_heuristic + risk_penalty

Function estimate_energy_risk(node, goal):
    risk_count = 0
    line_cells = bresenham_line(node, goal) // Algoritma line drawing

    for each cell in line_cells:
        if cell.type == 2: // Zona Angin Kencang
            risk_count += 1

    return risk_count

```



Gambar 4.1 Konfigurasi Peta Grid 20x20 Pembahasan

Peta tersebut memiliki titik Awal (S) di (1, 1) dan titik Tujuan (G) di (18, 18). Karakteristik utama dari skenario ini adalah:

- Rintangan (X): Terdapat dua blok rintangan utama yang menciptakan koridor-koridor sempit dan memaksa algoritma untuk melakukan manuver yang lebih kompleks.
- Zona Angin Kencang (W): Sebuah zona berbentuk 'L' yang dominan di sisi kiri-tengah peta, memberikan penalti energi yang signifikan.
- Zona Gangguan Sinyal (~): Sebuah zona yang tersebar di sisi kanan-tengah, memberikan penalti keamanan sinyal.

Desain ini secara strategis menempatkan zona-zona berisiko tinggi tepat di jalur diagonal terpendek antara S dan G, sehingga menguji secara langsung kemampuan algoritma untuk melakukan *trade-off* antara jarak dan keamanan. Dua algoritma berikut dibandingkan:

1. Algoritma A* Standar: Dikonfigurasi untuk tujuan tunggal meminimalkan jarak, dengan bobot ($w_{\text{jarak}} = 1, w_{\text{energi}} = 0, w_{\text{sinyal}} = 0$).
2. A* Multi-Objektif Adaptif (Metode Usulan): Dikonfigurasi dengan bobot seimbang ($w_{\text{jarak}} = 0.2, w_{\text{energi}} = 0.4, w_{\text{sinyal}} = 0.4$) dan menggunakan heuristik adaptif.

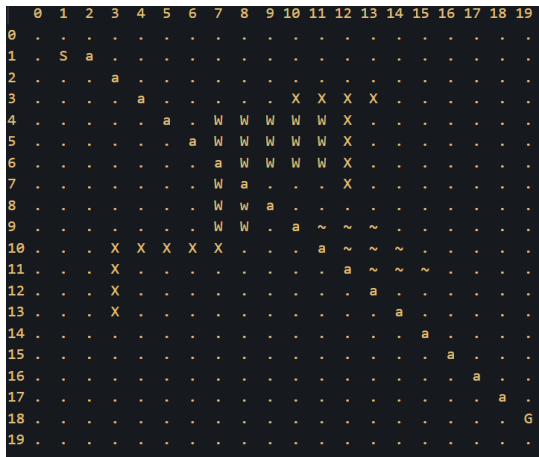
IV. HASIL DAN ANALISIS

A. Skenario Pengujian

Skenario pengujian menggunakan lingkungan simulasi grid 20x20 yang konfigurasinya disajikan pada Gambar 4.1. Peta ini dirancang dengan kompleksitas yang lebih tinggi untuk memberikan ujian yang lebih rigor terhadap kemampuan adaptasi algoritma.

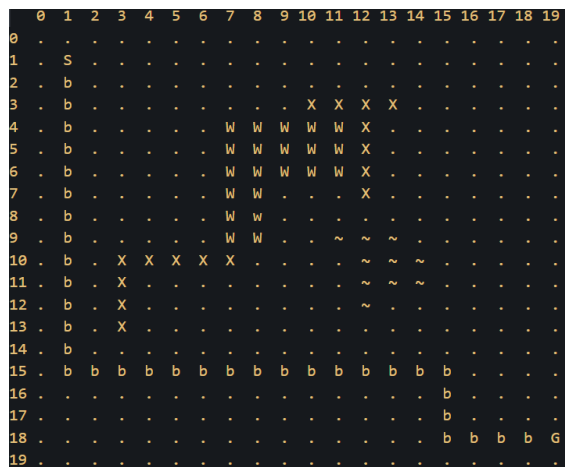
B. Hasil Visual Perbandingan Rute

Perbandingan visual rute yang dihasilkan oleh kedua algoritma disajikan pada Gbr. 4.2 dan Gbr. 4.3.



Gambar 4.2 Rute Algoritma A* Standar

Pada Gambar 4.2, rute A* Standar menunjukkan perilaku yang dapat diprediksi. Algoritma ini memprioritaskan jarak di atas segalanya, menghasilkan jalur yang berusaha sedekat mungkin dengan garis lurus diagonal. Akibatnya, rute ini secara langsung memotong Zona Angin Kencang (W) dan kemudian melewati beberapa sel Zona Gangguan Sinyal (~) untuk mencapai tujuan. Perilaku ini menunjukkan "kebutaan" algoritma terhadap risiko yang ada di peta.



Gambar 4.2 Rute Algoritma A* Standar

Pada Gambar 4.3, rute yang dihasilkan oleh Metode Usulan menunjukkan kecerdasan situasional. Menyadari biaya tinggi yang akan dikenakan jika melewati zona W dan ~, algoritma ini membuat keputusan strategis untuk mengambil rute memutar ke bawah. Jalur ini secara aktif menghindari klaster utama dari kedua zona berbahaya, melewati koridor aman di antara rintangan. Meskipun rute ini jelas lebih panjang secara geometris, ia merepresentasikan pilihan yang jauh lebih aman dan efisien secara keseluruhan.

C. Analisis Kerja Kuantitatif.

Metrik	A* Standar	A* Multi Objektif (Usulan)
Total Langkah	30	42

Total Biaya Energi	84	42
Total Penalti Sinyal	100	0
Total Biaya	79.6	25.2
Sisa Energi Drone (akhir)	15% (sangat kritis)	60% (aman)

Tabel 4.1 Hasil Perbandingan Algoritma A* Standar dan A* Multi Objektif

Seperti yang terlihat pada Tabel 1, metode A* Standar menghasilkan rute yang lebih pendek secara geometris dengan 30 langkah. Namun, efisiensi jarak ini dicapai dengan 'harga' yang sangat mahal, yaitu akumulasi biaya energi sebesar 84 dan penalti sinyal sebesar 100."

"Di sisi lain, Metode Usulan, dengan mengambil rute yang jauh lebih panjang (42 langkah), berhasil menghilangkan penalti sinyal sepenuhnya (0) dan menjaga biaya energi tetap minimal (42). Hasilnya, Total Biaya dari metode usulan (25.2) lebih baik dibandingkan A* Standar (79.6). Hal ini membuktikan keberhasilan algoritma dalam melakukan trade-off yang cerdas antara jarak demi keselamatan dan keandalan operasional, yang tercermin pada sisa energi drone yang jauh lebih aman (60% vs 15%).

D. Pembahasan

Hasil eksperimen pada skenario yang lebih kompleks ini semakin mempertegas keunggulan pendekatan multi-objektif dan adaptif. A* standar, dengan logikanya yang kaku, gagal menginterpretasikan peta sebagai sebuah lingkungan dengan risiko yang bervariasi. Ia hanya melihat peta sebagai jarak dan rintangan, sehingga menghasilkan solusi yang efisien di atas kertas namun rapuh dalam praktik.

Keberhasilan Metode Usulan berasal dari kemampuannya untuk melakukan penalaran yang lebih kompleks. Fungsi biaya multi-objektif memungkinkannya untuk "merasakan" tingkat bahaya dari setiap sel. Sementara itu, heuristik adaptif memberikan kemampuan untuk "berpikir ke depan", mengubah strategi ketika sumber daya (energi) mulai menipis. Kombinasi inilah yang menghasilkan navigasi yang tangguh.

Temuan ini sangat relevan untuk pengembangan sistem otonom di dunia nyata. Kemampuan sebuah agen (seperti drone) untuk menimbang berbagai faktor dan beradaptasi dengan kondisinya sendiri adalah kunci untuk menciptakan sistem yang tidak hanya pintar, tetapi juga bijaksana, aman, dan andal.

V. KESIMPULAN DAN SARAN

A. Ringkasan dan Kontribusi

Penelitian ini berangkat dari permasalahan fundamental dalam navigasi drone otonom, di mana algoritma pathfinding konvensional seperti A* standar tidak memadai karena sifatnya yang *single-objective* dan hanya berfokus pada jarak terpendek. Keterbatasan ini kontras dengan kebutuhan operasi drone di dunia nyata yang menuntut pertimbangan simultan terhadap berbagai faktor kritis seperti konsumsi energi dan

keamanan sinyal. Untuk mengatasi kesenjangan ini, makalah ini telah merancang dan mempresentasikan sebuah metode A* yang dimodifikasi, yang mengintegrasikan fungsi biaya multi-objektif dan mekanisme heuristik yang bersifat adaptif. Metode yang diusulkan mampu menimbang secara dinamis antara jarak, energi, dan sinyal, serta menyesuaikan perilakunya berdasarkan sisa energi drone.

Melalui eksperimen simulasi pada skenario yang dirancang khusus, penelitian ini berhasil menunjukkan bahwa metode yang diusulkan secara signifikan lebih unggul daripada A* standar. Meskipun menghasilkan rute dengan total jarak yang lebih panjang, metode usulan mampu menekan biaya energi dan risiko sinyal secara drastis, yang menghasilkan total biaya tertimbang yang jauh lebih rendah. Dampak praktis yang paling signifikan adalah peningkatan keamanan operasional, di mana drone tiba di tujuan dengan sisa energi yang aman. Dengan demikian, kontribusi utama dari penelitian ini adalah sebuah kerangka kerja *pathfinding* yang lebih cerdas, andal, dan sadar-konteks (*context-aware*), yang lebih sesuai untuk aplikasi drone di lingkungan yang kompleks.

B. Keterbatasan dan Saran Pengembangan Selanjutnya

Penulis menyadari adanya beberapa keterbatasan dalam penelitian ini yang dapat menjadi titik awal untuk pengembangan di masa depan. Pertama, simulasi dilakukan dalam lingkungan grid 2D yang merupakan simplifikasi dari ruang 3D yang sesungguhnya. Kedua, faktor lingkungan seperti zona angin dan gangguan sinyal bersifat statis dan tidak berubah selama simulasi.

Berdasarkan keterbatasan tersebut, beberapa arah pengembangan selanjutnya yang dapat dilakukan adalah:

1. Implementasi dalam Lingkungan 3D: Mengembangkan metode ini untuk bekerja dalam ruang tiga dimensi agar lebih merepresentasikan navigasi drone yang sebenarnya.
2. Penanganan Rintangan Dinamis: Mengintegrasikan mekanisme untuk mendeteksi dan bereaksi terhadap rintangan yang bergerak, seperti drone lain atau perubahan cuaca secara tiba-tiba.
3. Pengujian pada Perangkat Keras: Memvalidasi algoritma ini pada drone fisik untuk mengukur performa dan efisiensi energi secara *real-time*.

VI. UCAPAN TERIMA KASIH

Puji dan syukur saya ucapkan kepada Tuhan Yang Maha Esa, karena atas berkat dan karunia-Nya saya dapat menyelesaikan penulisan makalah ini. Saya juga berterima kasih kepada keluarga saya serta kepada dosen pengampu kelas 03 mata kuliah Strategi Algoritma tahun 2025, yaitu Bapak Monterico Adrian, atas bimbingannya di semester ini.

REFERENCES

- [1] Z. G. Zhang, S. Y. Wang, dan G. G. Wang, "Adaptive heuristic search algorithm for discrete variables based multi-objective optimization," *Structural and Multidisciplinary Optimization*, vol. 48, hlm. 821-836, April 2013.
- [2] Munir, Rinaldi. "Penentuan Rute (Route Planning) Bagian 1." Institut Teknologi Bandung, 2020-2021. [Penentuan Rute \(Route/Path Planning\) Bagian 1](#). Diakses pada 22 Juni 2025.
- [3] Munir, Rinaldi. "Penentuan Rute (Route Planning) Bagian 2." Institut Teknologi Bandung, 2020-2021. [Penentuan Rute \(Route/Path Planning\) Bagian 2](#). Diakses pada 22 Juni 2025.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 23 Juni 2025



Boye Mangaratua Ginting
13523127