

Algoritma Perencanaan Jalur Asimptotik Optimal Menggunakan *Improved Probabilistic Road Map*

Asymptotically-Optimal Path Planning Using the Improved Probabilistic Road Map Algorithm

Muhammad Aria Rajasa Pohan

Teknik Elektro, Universitas Komputer Indonesia
Jalan Dipati Ukur No 112 – 116 Bandung, Indonesia
Email : muhammad.aria@email.unikom.ac.id

Abstrak - Suatu algoritma perencanaan jalur bersifat asimptotik optimal jika dapat menjamin bahwa akan menghasilkan solusi optimal jika diberikan waktu yang memadai. Algoritma perencanaan jalur yang dapat memberikan solusi optimal sangat penting dalam banyak aplikasi robotik. Penelitian ini bertujuan untuk mengusulkan algoritma perencanaan jalur baru yang bersifat asimptotik optimal. Metode yang digunakan adalah dengan meningkatkan algoritma *probabilistic road map* (PRM) melalui tiga strategi. Strategi pertama adalah menggunakan teknik pembangkitan sampel berbasis informasi. Strategi kedua adalah area pencarian dimulai dari subset ellipsoid yang berukuran kecil dahulu. Strategi ketiga adalah melakukan perbaikan jalur menggunakan proses *wrapping*. Algoritma yang diusulkan ini dinamakan algoritma *wrapping-based informed PRM* (WIPRM). Selanjutnya, performansi algoritma WIPRM dibandingkan dengan algoritma PRM, *informed rapidly-exploring random tree** (RRT*), dan *informed PRM*. Hasil pengujian menunjukkan bahwa algoritma WIPRM dapat membangun jalur yang optimal untuk semua scenario pengujian. Kriteria jalur optimal yang digunakan pada penelitian ini adalah jalur dengan jarak terpendek antara node awal dengan node tujuan. Waktu komputasi yang dibutuhkan algoritma WIPRM untuk membangun jalur optimal lebih baik daripada algoritma *informed RRT** dan *informed PRM*. Dengan demikian, algoritma WIPRM berpotensi untuk diimplementasikan dalam berbagai sistem robotik yang membutuhkan algoritma perencanaan jalur yang optimal seperti mobil otonom, *unmanned aerial vehicles* (UAV), maupun *autonomous underwater vehicles* (AUV).

Kata kunci : Asimptotik optimal, informed sampling, perencanaan jalur, probabilistic road map, proses *wrapping*

Abstract - A path planning algorithm is asymptotically optimal if it can guarantee that it will produce an optimal solution if given sufficient time. Path-planning algorithms that can provide optimal solutions are essential in many robotic applications. The objective of this study is to propose a new asymptotic optimal path planning algorithm. The method is to improve the probabilistic road map (PRM) algorithm through three strategies. The first strategy is to use an information-based sampling technique. The second strategy is that the search area starts from a small ellipsoid subset first. The third strategy is to improve the path using the wrapping process. We call this proposed algorithm the wrapping-based informed PRM (WIPRM) algorithm. Furthermore, the performance of the WIPRM algorithm was compared with the PRM algorithm, informed rapidly-exploring random tree* (RRT*), and informed PRM. The test results show that the WIPRM algorithm can build optimal paths for all test scenarios. The optimal criterion used in this study is the shortest distance between the initial node and the destination node. The computational time needed by the WIPRM algorithm to build the optimal path is better than the informed RRT* and informed PRM algorithms. These results indicate that the WIPRM algorithm could be used in various robotic systems requiring optimal path planning algorithms, such as autonomous cars, unmanned aerial vehicles (UAV), and autonomous undersea vehicles (AUV).

Keywords : Optimal asymptotic, informed sampling, path planning, probabilistic road map, wrapping process.

I. PENDAHULUAN

Algoritma perencanaan jalur disebut bersifat asimtotik optimal jika dapat menjamin bahwa algoritma tersebut akan menghasilkan solusi optimal jika diberikan jumlah iterasi/waktu yang mencukupi [1, 2]. Kriteria solusi optimal dapat didasarkan pada satu atau beberapa kondisi, seperti jarak terpendek, kenyamanan, risiko terendah, ataupun kebutuhan bahan bakar yang lebih sedikit [3]. Algoritma perencanaan jalur yang dapat memberikan solusi optimal sangat penting dalam banyak aplikasi robotik [4]. Proses otomasi pada industri [5], navigasi robot [6], mobil otonom tanpa pengemudi [7], proses bedah robotik [8] adalah beberapa contoh aplikasi robotik yang memerlukan algoritma perencanaan jalur yang optimal.

Beberapa peneliti telah mengusulkan algoritma perencanaan jalur yang bersifat asimtotik optimal. Tetapi, setiap algoritma tersebut memiliki kinerja yang berbeda. Salah satu parameter yang umum digunakan untuk mengukur kinerja algoritma perencanaan jalur adalah waktu komputasi untuk menghasilkan jalur yang optimal [9-11]. Karaman dan Frazzoli mengusulkan algoritma RRT* yang memberikan solusi optimal asimtotik [12]. Namun, Qureshi dkk. [13], Wang dkk. [14], dan Yuncheng dan Jie [15] melaporkan bahwa kecepatan komputasi RRT* untuk mencapai nilai optimal masih harus ditingkatkan. Salah satu hal yang membebani kecepatan komputasi dari algoritma RRT* adalah algoritma ini harus melakukan pengambilan sampel di seluruh ruang pencarian.

Untuk meningkatkan kinerja dari algoritma RRT*, Gammel dkk. [16] mengusulkan algoritma *informed RRT** yang membatasi area pengambilan sampel berdasarkan informasi dari jalur yang telah diketahui saat ini (yang masih belum optimal). Dengan menggunakan strategi pembatasan area pengambilan sampel ini, Wang [17] melaporkan bahwa algoritma *informed RRT** dapat mencapai solusi optimal 1,2-10,3 kali lebih cepat daripada algoritma RRT*. Penelitian-penelitian untuk meningkatkan kecepatan komputasi dari algoritma *informed RRT** juga masih terus dilakukan. Jiang dkk. [18] mengusulkan *adjoined obstacle informed RRT**, Dai dkk [19] mengusulkan kombinasi *informed RRT** dengan *dynamic window approach* [DWA], dan Ryu dan Park [20] mengusulkan penggunaan *gridmap skeletonization* pada *informed RRT**. Adapun Kim dan Song [21] mengusulkan peningkatan kecepatan komputasi dari algoritma *informed RRT** dengan

menggunakan proses *wrapping*. Kim dan Song melaporkan bahwa dengan menggunakan proses *wrapping*, performansi kecepatan komputasi dari algoritma *informed RRT** dapat ditingkatkan 45%-89%.

Peningkatan performansi algoritma RRT melalui strategi sampling juga telah dilakukan oleh beberapa peneliti. Fauzi [22] mengembangkan metode hybrid sampling agar RRT dapat menemukan solusi jalur lebih cepat. Pakaya [23] mengembangkan metode *improved gaussian sampling* yaitu penerapan distribusi gaussian pada sampling RRT. Ferida dan Rahajoeningroem [24] mengembangkan metode sampling menuju lokasi goal pada algoritma RRT. Sopa dan Hartono [25] mengembangkan strategi sampling RRT dengan menggabungkan metode *sampling goal biasing*, gaussian, dan boundary.

Algoritma lain yang bersifat optimal asimtotik adalah algoritma *informed probabilistic road map* (PRM) yang diusulkan oleh penulis [24]. Aria melaporkan bahwa dengan mengkombinasikan *informed searching* dengan algoritma PRM, maka performansi algoritma yang diusulkan dapat meningkat hingga 25%. Penelitian-penelitian untuk meningkatkan kinerja dari algoritma PRM juga masih terus dilakukan. Chen dkk. [27] mengusulkan strategi pengambilan sampel PRM baru untuk menghasilkan konfigurasi yang lebih sesuai pada aplikasi praktis. Ravankar dkk. [28] mengusulkan penggunaan *layered hybrid PRM* dengan *Artificial Potential Field* (APF). Sedangkan Chen dkk. [29] mengusulkan kombinasi PRM dengan *potential field*.

Tujuan dari penelitian ini adalah untuk mengusulkan algoritma perencanaan jalur asimtotik optimal yang baru berbasis peningkatan algoritma PRM (*improved PRM*). Terdapat tiga strategi yang digunakan untuk meningkatkan algoritma PRM. Strategi pertama adalah menggunakan teknik *sampling* berbasis informasi. Strategi kedua adalah memulai area pencarian yang dimulai dari subset ellipsoid yang berukuran kecil dahulu. Strategi ketiga adalah memperbaiki jalur yang telah diperoleh dengan menggunakan proses *wrapping*. Algoritma yang diusulkan ini dikenal sebagai PRM informasi berbasis pembungkus (WIPRM). Selanjutnya, performansi algoritma WIPRM dibandingkan dengan algoritma PRM, *informed RRT**, dan *informed PRM*. Hasil pengujian menunjukkan bahwa algoritma WIPRM dapat membangun jalur yang optimal untuk semua

scenario pengujian. Waktu komputasi yang dibutuhkan algoritma WIPRM untuk membangun jalur optimal lebih baik daripada algoritma *informed RRT** dan *informed PRM*. Dengan demikian, algoritma WIPRM berpotensi untuk diimplementasikan dalam berbagai sistem robotik yang membutuhkan algoritma perencanaan jalur yang optimal seperti mobil otonom, *unmanned aerial vehicles* (UAV), maupun *autonomous underwater vehicles* (AUV).

Makalah ini disusun dengan struktur sebagai berikut. Bagian II menyajikan desain algoritma yang diusulkan: algoritma WIPRM. Pembahasan mengenai strategi-strategi yang digunakan untuk meningkatkan kinerja PRM dibahas pada bagian ini. Bagian III membahas hasil dan pembahasan. Pada bagian pertama-tama didiskusikan mengenai hasil pengukuran efek dari setiap strategi yang diusulkan untuk meningkatkan kinerja dari PRM. Kemudian performansi dari algoritma WIPRM yang diusulkan dibandingkan terhadap performansi dari algoritma PRM, *informed RRT**, dan *informed PRM*. Selanjutnya pada bagian IV menyajikan beberapa kesimpulan.

II. ALGORITMA YANG DIUSULKAN: WIPRM

Algoritma yang diusulkan merupakan algoritma PRM yang ditingkatkan melalui tiga strategi. Strategi pertama adalah menggunakan teknik pembangkitan sampel berbasis informasi. Strategi kedua adalah area pencarian dimulai dari subset ellipsoid yang berukuran kecil dahulu. Strategi ketiga adalah melakukan perbaikan jalur menggunakan proses *wrapping*.

A. Strategi Pertama: Pembangkitan sampel berbasis informasi

Strategi pembangkitan sampel berbasis informasi ini meniru algoritma *informed RRT** yang diusulkan oleh Gammel dkk. [16]. Jika pada suatu iterasi berhasil ditemukan solusi jalur yang menghubungkan node awal dengan node tujuan, maka akan dibentuk suatu area untuk membatasi pembangkitan sampel. Area tersebut berbentuk ellipsoid yang eksentrisitasnya bergantung pada panjang solusi jalur terpendek yang telah ditemukan pada iterasi tersebut. Dengan adanya area ellipsoid tersebut, maka proses pembangkitan sampel pada iterasi berikutnya hanya akan dilakukan di dalam area ellipsoid tersebut saja. Hal ini akan meningkatkan konsentrasi pencarian hanya pada area yang berpotensi meningkatkan

kualitas jalur solusi saja. Gammel dkk. telah menunjukkan bahwa jika area ellipsoid ini telah ditemukan, maka pembangkitan sampel diluar area ellipsoid ini tidak akan membantu meningkatkan kualitas jalur yang dihasilkan.

Jika pada iterasi berikutnya ditemukan solusi jalur yang lebih pendek, maka luas area ellipsoid ini akan semakin mengecil dan konsentrasi pencarian jalur akan semakin terpusat. Gammel dkk [26] melaporkan bahwa dengan menggunakan strategi ini, algoritma *informed RRT** dapat mencapai solusi optimal 3,4 kali lebih cepat daripada algoritma *RRT**.

Ilustrasi dari proses pembangkitan sampel berbasis informasi pada algoritma PRM ditunjukkan pada **Gambar 1**. Saat iterasi pertama, pembangkitan sampel dilakukan secara acak diseluruh area (**Gambar 1a**). Lalu algoritma dijkstra dijalankan untuk mencari jalur yang dapat menghubungkan node awal dengan node akhir melalui node-node sampel yang telah dibangkitkan. Contoh jalur yang berhasil dibuat oleh algoritma dijkstra ditunjukkan pada garis berwarna merah pada **Gambar 1a**.

Setelah ditemukan suatu solusi jalur, maka dibentuk suatu area untuk membatasi area pembangkitan sampel, yaitu area ellipsoid berwarna abu-abu pada **Gambar 1b**. Maka pada iterasi selanjutnya, prosedur pembangkitan sampel hanya akan diterapkan pada area ellipsoid ini saja, seperti yang ditunjukkan pada **Gambar 1c**. Jika pada iterasi berikutnya ditemukan solusi jalur yang lebih pendek, maka luas area ellipsoid ini akan semakin mengecil dan pencarian jalur akan lebih terkonsentrasi seperti ditunjukkan pada **Gambar 1d**. Pada ilustrasi **Gambar 1**, dapat terlihat bahwa solusi optimal harus melalui suatu jalur sempit. Menggunakan strategi pertama ini, solusi yang mendekati jalur optimal ini baru dapat dicapai pada iterasi ke-10, seperti yang terlihat pada **Gambar 1d**. Oleh karena itu, strategi peningkatan algoritma PRM kedua diperlukan. Yaitu area pencarian dimulai dari subset ellipsoid berukuran kecil terlebih dahulu.

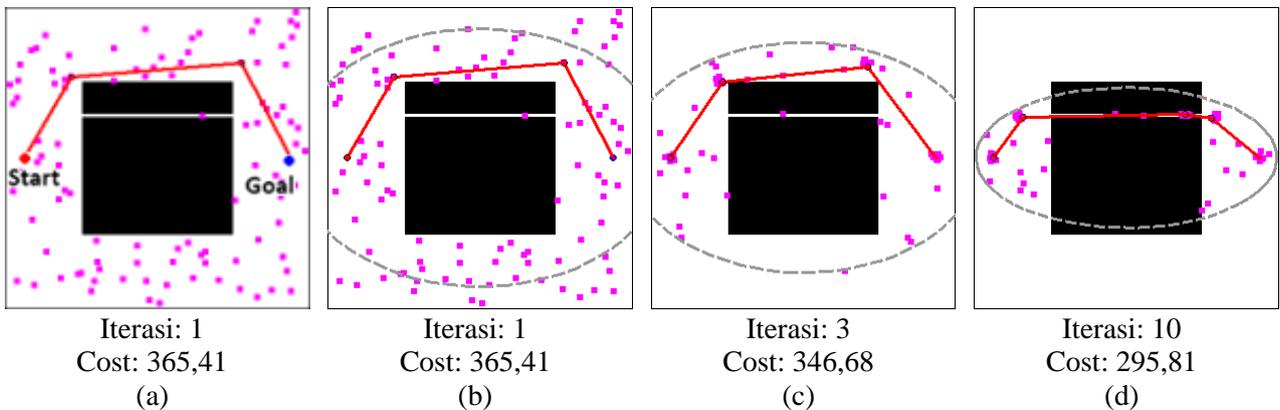
B. Strategi Kedua: Area Pencarian dimulai dari Subset Ellipsoid Berukuran Kecil Dahulu

Strategi untuk memulai pembangkitan sampel berbasis informasi pada area subset ellipsoid berukuran kecil dahulu ini meniru algoritma BIT* yang diusulkan oleh Gammell dkk., pada [30]. Strategi kedua ini berbeda dengan algoritma dasar *informed RRT**. Pada algoritma dasar *informed RRT**, pada iterasi pertama belum ada area

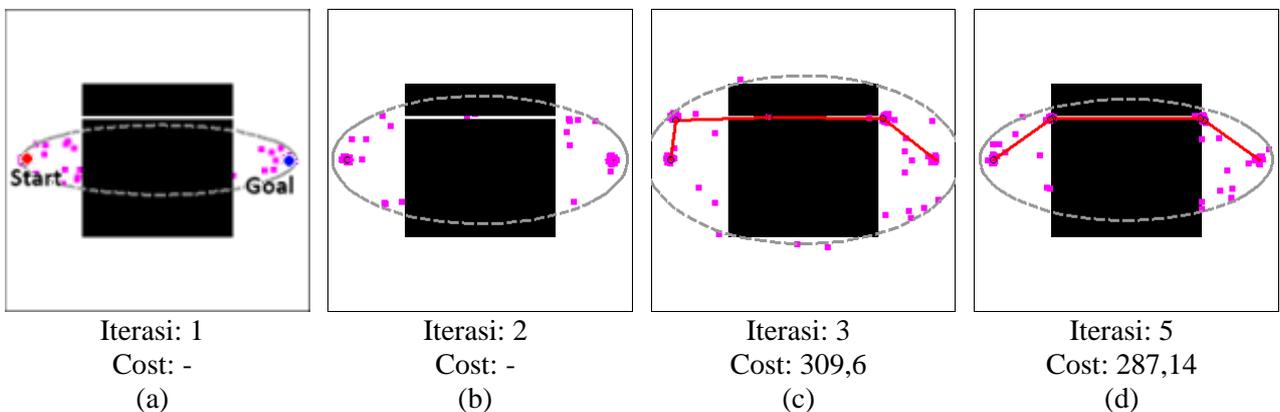
ellipsoid yang membatasi area pembangkitan sampel (seperti yang ditunjukkan pada **Gambar 1a**). Sedangkan untuk algoritma BIT*, pada iterasi pertama telah dibuat area ellipsoid berukuran kecil yang akan membatasi area pembangkitan sampel. Jika solusi jalur di area kecil ini tidak dapat ditemukan, maka area ellipsoid akan diperbesar secara iteratif hingga ditemukan solusi jalur yang menghubungkan node awal ke node tujuan. Setelah solusi jalur tersebut ditemukan, maka pada iterasi selanjutnya area ellipsoid yang membatasi area pembangkitan sampel tidak akan diperbesar lagi. Melainkan akan diperkecil jika diperoleh panjang jalur solusi yang lebih pendek. Gammell dkk. melaporkan bahwa dengan menggunakan strategi ini, maka algoritma BIT* dapat mencapai solusi optimal 6,8 kali lebih cepat daripada algoritma RRT*.

Ilustrasi dari strategi kedua ini ditunjukkan pada **Gambar 2**. Saat iterasi pertama, telah dibuat suatu area ellipsoid yang membatasi area pembangkitan sampel, seperti yang ditunjukkan pada **Gambar 2a**. Eksentrisitas area ellipsoid yang membatasi area pembangkitan ini ditentukan oleh suatu garis yang menghubungkan node awal dengan node akhir. Karena pada iterasi pertama belum diketahui

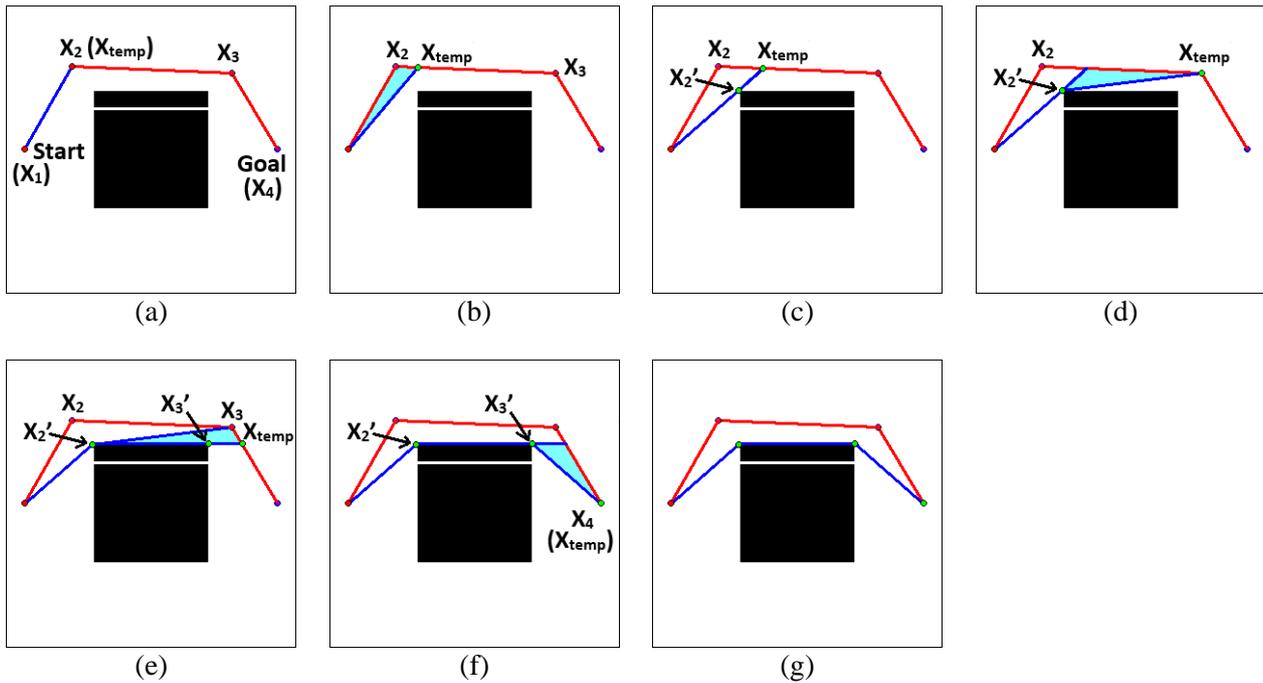
panjang jalur yang dapat menghubungkan node awal dengan node akhir, maka garis yang menentukan eksentrisitas ellipsoid tersebut ditentukan berdasarkan asumsi. Digunakan asumsi suatu garis lurus yang menghubungkan node awal dengan node akhir, kemudian pada garis tersebut ditambahkan suatu toleransi panjang tertentu. Dengan strategi ini, jika pada iterasi pertama belum ditemukan jalur yang dapat menghubungkan node awal dengan node akhir, maka luas eksentrisitas ellipsoid yang membatasi area pembangkitan ini diperbesar, seperti yang ditunjukkan pada **Gambar 2b**. Proses meningkatkan luas eksentrisitas area ellipsoid ini dilakukan bertahap pada setiap iterasi hingga diperoleh jalur yang dapat menghubungkan node awal dengan node akhir, seperti yang ditunjukkan pada **Gambar 2c**. Setelah solusi jalur tersebut ditemukan, maka pada iterasi selanjutnya area ellipsoid yang membatasi area pembangkitan sampel tidak akan diperbesar lagi. Melainkan akan diperkecil jika diperoleh panjang jalur solusi yang lebih pendek, seperti yang ditunjukkan pada **Gambar 2d**. Perbandingan performansi antara algoritma PRM yang menggunakan strategi kedua ini terhadap algoritma PRM dasar akan ditunjukkan pada Bab III.



Gambar 1. Ilustrasi dari proses pembangkitan sampel berbasis informasi pada algoritma PRM.



Gambar 2. Ilustrasi dari proses untuk memulai pembangkitan sampel berbasis informasi pada area subset ellipsoid berukuran kecil dahulu pada algoritma PRM.



Gambar 3. Ilustrasi dari proses *wrapping* untuk mengoptimalkan jalur yang dihasilkan. Garis merah menunjukkan jalur awal. Garis biru menunjukkan jalur perbaikan/hasil proses *wrapping*

C. Strategi Ketiga: Perbaikan Jalur menggunakan Proses *Wrapping*

Strategi memperbaiki jalur menggunakan proses *wrapping* ini meniru algoritma *wrapping-based informed RRT** yang diusulkan oleh Kim dan Song [21]. Proses *wrapping* ini bertujuan mencari jalur baru yang lebih pendek dengan cara membuat node-node baru yang berdekatan dengan hambatan. Ilustrasi dari strategi ketiga ini ditunjukkan pada **Gambar 3**.

Pada contoh kasus yang ditunjukkan pada **Gambar 3**, terdapat jalur awal berwarna merah yang terdiri dari empat node. Proses *wrapping* dimulai dengan membuat node sementara/temporary (X_{temp}) yang berlokasi pada node X_{i+1} atau node X_2 . Node X_{temp} dengan node X_1 dihubungkan dengan garis berwarna biru, seperti yang ditunjukkan pada **Gambar 3a**. Lalu posisi node X_{temp} akan dimajukan sesuai jalur yang menghubungkan node X_{i+1} dengan node X_{i+2} , seperti pada **Gambar 3b**. Area berwarna biru muda menunjukkan area yang telah dilalui oleh garis biru yang menghubungkan X_1 dengan X_{temp} . Posisi node X_{temp} akan terus dimajukan sehingga terdeteksi adanya hambatan yang menghalangi garis biru yang menghubungkan X_1 dengan X_{temp} , seperti yang ditunjukkan pada **Gambar 3c**. Maka posisi garis biru yang bertemu hambatan tersebut akan dijadikan node baru bagi X_2 (dinamakan X_2'). Pada iterasi selanjutnya, posisi X_{temp} kembali dimajukan, tetapi karena telah ditemukan node

baru X_2' , maka garis biru menghubungkan X_{temp} dengan X_2' , seperti yang ditunjukkan pada **Gambar 3d**. Posisi X_{temp} terus dimajukan hingga mencapai node X_{i+2} atau node X_3 . Karena node X_3 telah dicapai, maka posisi X_{temp} dilanjutkan untuk maju sepanjang jalur yang menghubungkan node X_{i+2} dengan X_{i+3} (atau node X_3 dengan X_4). Proses ini ditunjukkan pada **Gambar 3e**. Jika garis biru yang menghubungkan node X_2' dengan X_{temp} bertemu hambatan, maka posisi garis biru yang bertemu hambatan tersebut akan dijadikan node baru bagi X_3 (dinamakan X_3'). Iterasi ini akan terus berulang hingga node X_{temp} mencapai node X_{tujuan} , seperti yang ditunjukkan pada **Gambar 3f**. Maka ilustrasi perbandingan jalur awal dengan jalur hasil proses *wrapping* dapat dilihat pada **Gambar 3g**. Garis merah menunjukkan jalur awal. Sedangkan garis biru menunjukkan jalur perbaikan/hasil proses *wrapping*. Node berwarna hijau menunjukkan node baru yang dihasilkan dari proses *wrapping*.

D. Algoritma lengkap WIPRM

Algoritma lengkap yang diusulkan ditunjukkan pada **Gambar 4 - Gambar 6**. Algoritma PRM terdiri dari pembangkitan sampel (baris 4-12 pada algoritma 1), pembuatan *roadmap* (baris 13-21 pada algoritma 1) dan membuat jalur (algoritma yang diusulkan menggunakan algoritma dijkstra) yang dapat menghubungkan node awal dengan

node akhir melalui node-node sampel yang telah dibangkitkan (baris 22 pada algoritma 1).

Strategi kedua algoritma WIPRM, yaitu memulai pembangkitan sampel berbasis informasi pada area subset ellipsoid berukuran kecil terlebih dahulu, diimplementasikan pada baris 1 algoritma 1. Dengan mengeset nilai c_{max} pada nilai minimal, akan membuat area subset ellipsoid berukuran kecil. Jika solusi jalur di area kecil ini tidak dapat ditemukan, maka area ellipsoid akan diperbesar secara iteratif hingga ditemukan solusi jalur yang menghubungkan node awal ke node tujuan. Proses ekspansi area ellipsoid selama jalur belum ditemukan ini ditunjukkan pada baris 23 dan 26 algoritma 1.

Strategi pertama algoritma WIPRM, yaitu melakukan pembangkitan sampel berdasarkan informasi, diimplementasikan pada baris 8 algoritma 1 dan algoritma 2. Pada algoritma 2, pembangkitan sampel x_{rand} hanya akan dilakukan pada area ellipsoid yang mengelilingi x_{start} dan x_{goal} dengan eksentrisitas bergantung pada

panjang c_{max} . Setiap kali algoritma menemukan jalur yang lebih pendek, maka nilai c_{max} akan diperbaharui (baris 24 algoritma 1), sehingga konsentrasi pencarian jalur akan semakin meningkat.

Adapun strategi ketiga algoritma WIPRM, yaitu memperbaiki jalur menggunakan proses *wrapping* diimplementasikan pada baris 29 algoritma 1 dan algoritma 3. Baris 4 – 6 pada algoritma 3 menunjukkan proses memajukan node X_{temp} sepanjang jalur yang menghubungkan node X_{i+1} dengan node X_{i+2} . Q_1 adalah panjang jalur yang menghubungkan X_{i+1} dengan X_{i+2} . Adapun baris 7 – 13 pada algoritma 3 menunjukkan proses untuk memeriksa apakah ada hambatan yang menghalangi garis yang menghubungkan X_i dengan X_{temp} . Jika terdapat posisi pada garis penghubung X_i dengan X_{temp} yang terhalangi oleh suatu hambatan, maka posisi tersebut akan dijadikan node baru X_{new} . Jika tidak ada, maka node X_{temp} akan terus dimajukan hingga mencapai X_{i+2} .

Algoritma 1. $X_{sol} = (map, x_{start}, x_{goal})$

```

1.  $c_{max} \leftarrow \|x_{goal} - x_{start}\|_2$ 
2.  $X_{sol} \leftarrow \emptyset$ 
3. while kondisi berhenti tidak terpenuhi do
4.    $V \leftarrow \emptyset$ 
5.    $E \leftarrow \emptyset$ 
6.   while  $|V| < n$  do
7.     repeat
8.        $x_{rand} \leftarrow \text{SampleWithNormalDistribution}(x_{start}, x_{goal}, c_{max})$ 
9.        $q \leftarrow x_{rand}$ 
10.      until  $q$  is collision-free
11.       $V \leftarrow V \cup \{q\}$ 
12.    end while
13.    for all  $q \in V$  do
14.       $N_q \leftarrow$  the neighbors of  $q$  chosen from  $V$  according to  $dist$ 
15.      for all  $q' \in N_q$  do
16.        if  $(q, q')$  is collision-free then
17.           $E \leftarrow E \cup \{(q, q')\}$ 
18.        end if
19.      end for
20.    end for
21.     $T = (V, E)$ 
22.     $X_{sol} \leftarrow \text{Dijkstra}(q_{init}, q_{goal}, T)$ 
23.    if  $X_{sol} \neq \emptyset$  then
24.       $c_{max} \leftarrow \min(x_{sol} \in X_{sol})\{Cost(x_{sol})\}$ 
25.    else
26.       $c_{max} \leftarrow c_{max} \times \text{konstanta ekspansi}$ 
27.    end if
28.  end while
29.  $X_{sol} \leftarrow \text{Wrapping}(X_{sol})$ 

```

Gambar 4. Algoritma WIPRM

Algorithm 2. $\text{SampleWithNormalDistribution}(x_{start}, x_{goal}, c_{max})$

1. **if** $c_{max} < \infty$ **then**
2. $c_{min} \leftarrow \|x_{goal} - x_{start}\|_2$
3. $x_{centre} \leftarrow (x_{goal} + x_{start})/2$
4. $C \leftarrow \text{RotationToWorldFrame}(x_{start}, x_{goal})$
5. $r_1 \leftarrow c_{max}/2$
6. $\{r_i\}_{i=2,\dots,n} \leftarrow \left(\sqrt{c_{max}^2 - c_{min}^2} \right) / 2$
7. $L \leftarrow \text{diag}\{r_1, r_2, \dots, r_n\}$
8. $x_{ball} \leftarrow \text{SampleUnitBallWithNormalDistribution}$
9. $x_{rand} \leftarrow (CLx_{ball} + x_{centre}) \cap X$
10. **else**
11. $x_{rand} \leftarrow \text{RandomSampling}(map)$
12. **end if**
13. **return** x_{rand}

Gambar 5. Strategi pembangkitan sampel pada algoritma WIPRM

Algorithm 3. $X_{sol} \leftarrow \text{Wrapping}(X_{sol})$

1. $i \leftarrow 1$
2. $P \leftarrow |X_{sol}|$
3. **while** $i < P - 1$
4. $Q_1 \leftarrow \|x_{i+1} - x_{i+2}\|_2$
5. **for** $k = 0$ **to** Q_1
6. $x_{temp} \leftarrow x_{i+1} + (x_{i+2} - x_{i+1}) \cdot \frac{k}{Q_1}$
7. $Q_2 \leftarrow \|x_i - x_{temp}\|_2$
8. **for** $m = 0$ **to** Q_2
9. $x_{new} \leftarrow x_i + (x_{temp} - x_i) \cdot \frac{m}{Q_2}$
10. **if** $x_i \rightarrow x_{new}$ **is not** collision-free
11. $x_{i+1} \leftarrow x_{new}$
12. **break** (**for** $k = 0$ **to** Q_1)
13. **end if**
14. **end for**
15. **end for**
16. $i \leftarrow i + 1$
17. **end while**
18. **return** X_{sol}

Gambar 6. Proses wrapping pada algoritma WIPRM

III. HASIL DAN PEMBAHASAN

Untuk memvalidasi kinerja algoritma perencanaan jalur yang diusulkan, kami melakukan beberapa pengujian yang berbeda. Pengujian pertama adalah untuk memverifikasi efektifitas dari strategi pertama, yaitu pembangkitan sampel berdasarkan informasi. Pengujian kedua adalah untuk memverifikasi efektifitas dari strategi kedua, yaitu area pencarian dimulai dari subset ellipsoid berukuran kecil dahulu. Pengujian ketiga adalah untuk memverifikasi efektifitas dari strategi ketiga, yaitu perbaikan jalur menggunakan proses

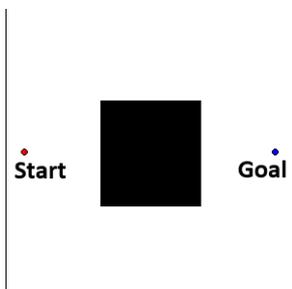
wrapping. Sedangkan pengujian keempat adalah untuk membandingkan kinerja antara algoritma WIPRM dengan algoritma PRM, *informed* RRT*, dan *informed* PRM. Ukuran kinerja yang diukur adalah waktu komputasi setiap algoritma untuk mencapai nilai optimal. Semua pengujian dijalankan secara independen sebanyak 30 kali dengan parameter yang sama. Perbandingan dilakukan berdasarkan kinerja rata-rata masing-masing algoritma pada 30 pengujian tersebut. Semua pengujian dilakukan menggunakan PC dengan CPU Core i5 3.20 GHz dan RAM 4 GB

yang berjalan pada Windows 10 64-bit. Algoritma WIPRM dan algoritma-algoritma pembandingan diimplementasikan pada LabVIEW 7.1 menggunakan *LabVIEW Libraries* untuk Perencanaan Jalur Robotik [31]. Adapun pembangkitan variabel sampel secara acak dilakukan menggunakan distribusi normal. Distribusi normal digunakan untuk mengurangi waktu pemrosesan. Titik-titik sampling hanya dihasilkan dalam standar deviasi tertentu dari garis lurus yang menghubungkan node awal dengan node tujuan. Dengan mencontoh yang diusulkan oleh Mohammed dkk. [32], maka parameter standar deviasi ditetapkan berdasarkan nilai jarak antara node awal dengan node akhir. Adapun nilai tengah dari distribusi normal ditetapkan berdasarkan nilai sampling pada garis lurus yang menghubungkan node awal dengan node tujuan.

A. Skenario Percobaan

Untuk menguji kecepatan konvergensi dan kinerja optimalitas algoritma yang diusulkan, algoritma WIPRM dibandingkan dengan algoritma-algoritma lainnya. Pengujian dilakukan dengan menggunakan empat kasus skenario yang umum digunakan pada pengukuran performansi algoritma perencanaan jalur. Empat skenario yang digunakan adalah lingkungan dengan hambatan tunggal, lingkungan dengan celah sempit, lingkungan dengan hambatan berbentuk T, dan lingkungan dengan multi-hambatan.

Skenario pengujian dengan hambatan tunggal ditunjukkan pada **Gambar 7**. Skenario ini digunakan untuk menguji apakah suatu algoritma dapat menghasilkan jalur konvergen yang optimal atau tidak. Jiang dkk. [18] dan Mashayekhi dkk. [33] menggunakan skenario pengujian seperti ini untuk menguji algoritma perencanaan jalur yang mereka usulkan.



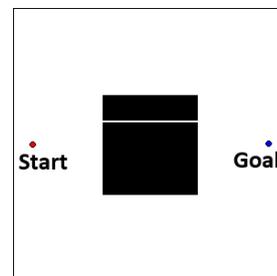
Gambar 7. Skenario pengujian I: lingkungan dengan hambatan tunggal

Skenario pengujian pada lingkungan dengan celah sempit ditunjukkan pada **Gambar 8**. Skenario ini digunakan untuk menguji keefektifan algoritma perencanaan jalur ketika node tujuan

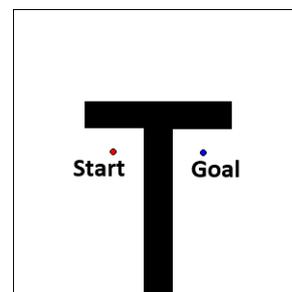
tersembunyi di balik lorong sempit. Gammel dkk. [16] dan Mashayekhi dkk. [33] menggunakan skenario pengujian seperti ini.

Skenario pengujian pada lingkungan dengan hambatan berbentuk huruf T ditunjukkan pada **Gambar 9**. Skenario ini digunakan untuk menguji keefektifan algoritma untuk bekerja pada lingkungan dimana jalur yang dihasilkan harus memutar. Jiang dkk. [18] dan Islam dkk. [34] menggunakan skenario pengujian seperti ini.

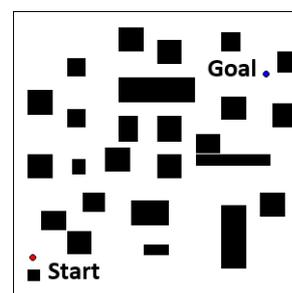
Skenario pengujian pada lingkungan dengan multi-hambatan ditunjukkan pada **Gambar 10**. Skenario ini digunakan untuk menguji kecepatan konvergensi dari algoritma perencanaan jalur. Gammel dkk. [16] dan Jiang dkk. [18] menggunakan skenario pengujian seperti ini.



Gambar 8. Skenario pengujian II: lingkungan dengan celah sempit



Gambar 9. Skenario pengujian III: lingkungan dengan hambatan berbentuk T



Gambar 10. Skenario pengujian IV: lingkungan dengan multi-hambatan

B. Verifikasi Efektifitas Strategi Pertama: Pembangkitan sampel berbasis informasi

Pengujian pertama adalah untuk memverifikasi efektifitas dari strategi pertama, yaitu pembangkitan sampel berdasarkan informasi.

Pengujian dilakukan dengan membandingkan algoritma dasar PRM dengan algoritma PRM yang ditambahkan strategi pertama, yaitu pembangkitan sampel berdasarkan informasi. Pengujian dilakukan pada empat skenario yang telah disebutkan pada Bab III.A. Performansi yang diukur adalah waktu komputasi masing-masing algoritma untuk mencapai jalur yang optimal. Hasil pengujian dapat dilihat pada **Tabel I**. Adapun analisa persentase rata-rata perbandingan waktu konvergensi untuk mencapai jalur optimal dari kedua algoritma dapat dilihat pada **Tabel II**.

Table I. Perbandingan algoritma PRM dengan strategi pertama terhadap algoritma PRM dasar (dalam detik)

Skenario	Waktu konvergensi untuk mencapai jalur optimal	PRM dengan Strategi Pertama	PRM Dasar
Skenario I: Hambatan Tunggal	Terbaik	0,25	25,79
	Rata-rata	4,37	26,19
	Terlama	9,83	27,12
Skenario II: Celah Sempit	Terbaik	0,83	5,58
	Rata-rata	3,23	17,21
	Terlama	7,42	26,84
Skenario III: Hambatan Berbentuk T	Terbaik	1,52	20,93
	Rata-rata	4,17	28,01
	Terlama	12,14	61,93
Skenario IV: Multi-hambatan	Terbaik	0,97	9,22
	Rata-rata	3,13	12,28
	Terlama	7,89	27,02

Table II. Perbandingan waktu konvergensi rata-rata dari algoritma PRM dengan strategi pertama terhadap algoritma PRM dasar

Skenario	Perbandingan waktu konvergensi (berapa kali lebih cepat)
Skenario I	5,99
Skenario II	5,33
Skenario III	6,72
Skenario IV	3,92
Rata-rata	5,49

Berdasarkan data pada **Tabel II** dapat terlihat bahwa waktu rata-rata algoritma PRM dengan strategi pertama adalah 5,49 kali lebih cepat dibandingkan dengan algoritma dasar PRM. Hasil ini konsisten dengan hasil pengukuran kinerja algoritma *informed RRT** (yang memiliki strategi peningkatan algoritma yang sama) yang dilaporkan oleh Gammel dkk. pada [30]. Gammel dkk. melaporkan bahwa dengan membatasi area pengambilan sampel pada area subset ellipsoid yang eksentrisitasnya sesuai panjang solusi jalur pada iterasi tersebut, algoritma *informed RRT** menjadi lebih cepat 3,4 kali dibandingkan algoritma *RRT** dalam mencapai jalur optimal. Hal ini memverifikasi efektivitas strategi pertama, yaitu pembangkitan sampel berdasarkan informasi, untuk meningkatkan kinerja dari algoritma PRM.

C. Verifikasi Efektifitas Strategi Kedua: Area Pencarian dimulai dari Subset Ellipsoid Berukuran Kecil Dahulu

Pengujian kedua adalah untuk memverifikasi efektifitas dari strategi kedua. Pada pengujian kedua ini, strategi pertama tidak diikutsertakan. Sehingga peningkatan algoritma PRM pada pengujian kedua ini hanya berasal dari strategi kedua saja. Pengujian dilakukan dengan membandingkan algoritma dasar PRM dengan algoritma PRM yang ditambahkan strategi kedua. Pengujian dilakukan pada empat skenario yang telah disebutkan pada Bab III.A. Performansi yang diukur adalah waktu komputasi masing-masing algoritma untuk mencapai jalur yang optimal. Hasil pengujian dapat dilihat pada **Tabel III**. Adapun analisa persentase rata-rata perbandingan waktu konvergensi untuk mencapai jalur optimal dari kedua algoritma dapat dilihat pada **Tabel IV**.

Table III. Perbandingan algoritma PRM dengan strategi kedua terhadap algoritma PRM dasar

Skenario	Waktu konvergensi untuk mencapai jalur optimal	PRM dengan Strategi kedua	PRM Dasar
Skenario I: Hambatan Tunggal	Terbaik	0,16	25,79
	Rata-rata	2,71	26,19
	Terlama	6,09	27,12
Skenario II: Celah Sempit	Terbaik	0,54	5,58
	Rata-rata	2,1	17,21
	Terlama	4,82	26,84
Skenario III: Hambatan Berbentuk T	Terbaik	1,78	20,93
	Rata-rata	4,89	28,01
	Terlama	14,23	61,93
Skenario IV: Multi-hambatan	Terbaik	0,61	9,22
	Rata-rata	1,94	12,28
	Terlama	4,89	27,02

Table IV. Perbandingan waktu konvergensi rata-rata dari algoritma PRM dengan strategi kedua terhadap algoritma PRM dasar

Skenario	Perbandingan waktu konvergensi (berapa kali lebih cepat)
Skenario I	9,68
Skenario II	8,61
Skenario III	5,72
Skenario IV	6,34
Rata-rata	7,59

Berdasarkan data pada **Tabel III** dapat terlihat bahwa waktu rata-rata algoritma PRM jika ditambahkan strategi kedua adalah 7,59 kali lebih cepat dibandingkan dengan algoritma dasar PRM. Hasil ini konsisten dengan yang dilaporkan oleh Gammel dkk. [30] mengenai hasil pengukuran kinerja algoritma *BIT** (yang memiliki strategi serupa untuk meningkatkan algoritma *RRT**). Gammel dkk. melaporkan bahwa dengan melakukan pengambilan sampel pada area subset

elipsoid yang berukuran kecil terlebih dahulu, maka algoritma BIT* dapat mencapai solusi optimal 6,8 kali lebih cepat daripada algoritma RRT*. Hal ini memverifikasi efektivitas strategi kedua, yaitu area pencarian dimulasi dari subset elipsoid berukuran kecil terlebih dahulu, untuk meningkatkan kinerja dari algoritma PRM.

D. Verifikasi Efektifitas Strategi Ketiga: Perbaikan Jalur menggunakan Proses *Wrapping*

Pengujian ketiga adalah untuk memverifikasi efektifitas dari strategi ketiga. Pada pengujian ketiga ini, strategi pertama dan strategi kedua tidak diikutsertakan. Sehingga peningkatan algoritma PRM pada pengujian ketiga ini hanya berasal dari strategi ketiga saja. Pengujian dilakukan dengan membandingkan algoritma dasar PRM dengan algoritma PRM yang ditambahkan strategi ketiga. Pengujian dilakukan pada empat skenario yang telah disebutkan pada Bab III.A. Performansi yang diukur adalah waktu komputasi masing-masing algoritma untuk mencapai jalur yang optimal. Hasil pengujian dapat dilihat pada **Tabel V**. Adapun analisa persentase rata-rata perbandingan waktu konvergensi untuk mencapai jalur optimal dari kedua algoritma dapat dilihat pada **Tabel VI**.

Berdasarkan data pada **Tabel VI** dapat terlihat bahwa waktu rata-rata algoritma PRM jika ditambahkan strategi ketiga adalah 11,59 kali lebih cepat dibandingkan dengan algoritma dasar PRM. Hal ini memverifikasi efektivitas strategi ketiga, yaitu area perbaikan jalur menggunakan proses *wrapping*, untuk meningkatkan kinerja dari algoritma PRM.

E. Perbandingan Kinerja Antara Algoritma WIPRM dan Algoritma Lainnya

Pengujian keempat adalah untuk membandingkan performansi algoritma WIPRM (yang menerapkan ketiga strategi yang diusulkan) terhadap algoritma *informed* RRT* dan algoritma *informed* PRM. Pengujian dilakukan pada empat skenario yang telah disebutkan pada Bab III.A. Performansi yang diukur adalah waktu komputasi masing-masing algoritma untuk mencapai jalur yang optimal. Hasil pengujian dapat dilihat pada **Tabel VII**. Adapun analisa persentase rata-rata perbandingan waktu konvergensi untuk mencapai jalur optimal dari kedua algoritma dapat dilihat pada **Tabel VIII**.

Table V. Perbandingan algoritma PRM dengan strategi ketiga terhadap algoritma PRM dasar

Skenario	Waktu konvergensi untuk mencapai jalur optimal	PRM dengan Strategi ketiga	PRM Dasar
Skenario I: Hambatan Tunggal	Terbaik	0,05	25,79
	Rata-rata	2,07	26,19
	Terlama	7,96	27,12
Skenario II: Celah Sempit	Terbaik	0,17	5,58
	Rata-rata	1,53	17,21
	Terlama	6,01	26,84
Skenario III: Hambatan Berbentuk T	Terbaik	0,3	20,93
	Rata-rata	1,98	28,01
	Terlama	9,83	61,93
Skenario IV: Multi-hambatan	Terbaik	0,19	9,22
	Rata-rata	1,48	12,28
	Terlama	6,39	27,02

Table VI. Perbandingan waktu konvergensi rata-rata dari algoritma PRM dengan strategi ketiga terhadap algoritma PRM dasar

Skenario	Perbandingan waktu konvergensi (berapa kali lebih cepat)
Skenario I	12,65
Skenario II	11,25
Skenario III	14,18
Skenario IV	8,28
Rata-rata	11,59

Table VII. Perbandingan algoritma WIPRM terhadap algoritma *informed* RRT dan *informed* PRM

Skenario	Waktu konvergensi untuk mencapai jalur optimal	WIPRM	<i>Informed</i> RRT*	<i>Informed</i> PRM
Skenario I: Hambatan Tunggal	Terbaik	0,05	1,11	0,25
	Rata-rata	1,81	6,47	4,37
	Terlama	4,06	14,31	9,83
Skenario II: Celah Sempit	Terbaik	0,17	3,74	0,83
	Rata-rata	1,4	23,26	3,23
	Terlama	3,21	63,25	7,42
Skenario III: Hambatan Berbentuk T	Terbaik	0,3	7,79	1,52
	Rata-rata	1,98	13,46	4,17
	Terlama	8,09	38,69	12,14
Skenario IV: Multi-hambatan	Terbaik	0,19	7,04	0,97
	Rata-rata	1,29	27,33	3,13
	Terlama	3,26	57,57	7,89

Table VIII. Perbandingan waktu konvergensi rata-rata dari algoritma WIPRM terhadap algoritma *informed* RRT dan *informed* PRM dasar

Skenario	Perbandingan waktu konvergensi (berapa kali lebih cepat)	
	<i>Informed</i> RRT*	<i>Informed</i> PRM
Skenario I	3,58	2,42
Skenario II	16,61	2,31
Skenario III	6,8	2,11
Skenario IV	21,13	2,42
Rata-rata	12,03	2,31

Berdasarkan data pada **Tabel VIII** dapat terlihat bahwa waktu rata-rata algoritma WIPRM adalah 12,03 kali lebih cepat dibandingkan dengan algoritma *informed* RRT* dan 2,31 kali lebih cepat daripada algoritma *informed* PRM. Maka dapat disimpulkan bahwa waktu komputasi yang dibutuhkan algoritma WIPRM untuk membangun jalur optimal lebih baik daripada algoritma *informed* RRT* dan *informed* PRM. Hasil pengujian menunjukkan bahwa algoritma WIPRM dapat membangun jalur yang optimal untuk semua scenario pengujian.

IV. KESIMPULAN

Algoritma WIPRM berhasil merancang jalur yang hampir optimal untuk setiap benchmark. Hasil pengujian menunjukkan bahwa algoritma WIPRM dapat membangun jalur yang optimal untuk semua scenario pengujian. Waktu komputasi yang dibutuhkan algoritma WIPRM untuk membangun jalur optimal lebih baik daripada algoritma *informed* RRT* dan *informed* PRM.

Algoritma WIPRM dapat memiliki kecepatan konvergensi yang baik karena menerapkan tiga strategi. Strategi pertama adalah menggunakan teknik pembangkitan sampel berbasis informasi. Strategi kedua adalah area pencarian dimulai dari subset ellipsoid yang berukuran kecil dahulu. Strategi ketiga adalah melakukan perbaikan jalur menggunakan proses *wrapping*. Efektivitas setiap strategi telah diverifikasi melalui hasil pengujian pada empat skenario lingkungan.

Dengan demikian, algoritma WIPRM berpotensi untuk diimplementasikan dalam berbagai sistem robotik yang membutuhkan algoritma perencanaan jalur yang optimal seperti mobil otonom, *unmanned aerial vehicles* (UAV), maupun *autonomous underwater vehicles* (AUV).

DAFTAR PUSTAKA

- [1] J. D. Marble dan K. E. Bekris, "Asymptotically Near-Optimal Planning with Probabilistic Road Map Spanners," in *IEEE Transactions on Robotics*, vol. 29, no. 2, hlm. 432-444, April 2013, doi: 10.1109/TRO.2012.2234312.
- [2] Y. Guo, X. Liu, X. Liu, Y. Yang, dan W. Zhang, "FC-RRT*: An Improved Path Planning Algorithm for UAV in 3D Complex Environment," *ISPRS International Journal of Geo-Information*, vol. 11, no. 2, hlm. 112, 2022.
- [3] I. Noreen, A. Khan, dan Z. Habib, Z. "Optimal path planning using RRT* based approaches: a survey and future directions," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 11, 2016.
- [4] M. Korkmaz and A. Durdu, "Comparison of optimal path planning algorithms," 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, hlm. 255-258, 2018.
- [5] C. Friedrich, A. Csizsar, A. Lechler dan A. Verl, "Efficient Task and Path Planning for Maintenance Automation Using a Robot System," in *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, hlm. 1205-1215, July 2018, doi: 10.1109/TASE.2017.2759814.
- [6] B. M. ElHalawany, H. M. Abdel-Kader, A. TagEldeen, A. E. Elsayed dan Z. B. Nossair, "Modified A* algorithm for safer mobile robot navigation," *2013 5th International Conference on Modelling, Identification and Control (ICMIC)*, 2013, hlm. 74-78.
- [7] Y. Rasekhipour, A. Khajepour, S. -K. Chen dan B. Litkouhi, "A Potential Field-Based Model Predictive Path-Planning Controller for Autonomous Road Vehicles," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, hlm. 1255-1267, May 2017, doi: 10.1109/TITS.2016.2604240.
- [8] P. Sudhakara, V. Ganapathy, dan K. Sundaran, "Trajectory Planning Using Enhanced Probabilistic Road maps For Pliable Needle Robotic Surgery," In *2018 International Conference on Recent Trends in Electrical, Control and Communication (RTECC)*, 2018, hlm. 61-64.
- [9] M. Elbanhawi dan M. Simic, "Sampling-Based Robot Motion Planning: A Review," in *IEEE Access*, vol. 2, hlm. 56-77, 2014, doi: 10.1109/ACCESS.2014.2302442.
- [10] Y. Yang, J. Pan, and W. Wan, "Survey of optimal motion planning," *IET Cyber-Systems and Robotics*, vol. 1, no. 1, hlm. 13-19, 2019.
- [11] R. Mashayekhi, M. Y. I. Idris, M. H. Anisi dan I. Ahmady, "Hybrid RRT: A Semi-Dual-Tree RRT-Based Motion Planner," in *IEEE Access*, vol. 8, hlm. 18658-18668, 2020, doi: 10.1109/ACCESS.2020.2968471.
- [12] S. Karaman, and E. Frazzoli, "Sampling-based Algorithms for Optimal Motion Planning," *Int. J. Robot. Res.*, vol. 30, no. 7, hlm. 846-894, Jun. 2011
- [13] A. H. Qureshi, S. Mumtaz, K.F. Iqbal, Y. Ayaz, M. S. Muhammad, O. Hasan, W. Y. Kim, dan M. Ra, "Triangular geometry based optimal motion planning using RRT*-motion planner," *2014 IEEE 13th Int. Workshop Adv. Motion Control (AMC)*, Yokohama, 2014, hlm. 380-385, doi: 10.1109/AMC.2014.6823312.
- [14] W. Wang, H. Gao, Q. Yi, K. Zheng dan T. Gu, "An Improved RRT* Path Planning Algorithm for Service Robot," *2020 IEEE 4th Inf. Technol., Netw., Electron. Automat. Control Conf. (ITNEC)*, Chongqing, China, 2020, hlm. 1824-1828, doi: 10.1109/ITNEC48623.2020.9085226.
- [15] Li Yuncheng dan Shao Jie, "A revised Gaussian distribution sampling scheme based on RRT* algorithms in robot motion planning," *2017 3rd Int. Conf. Control, Automat. Robot. (ICCAR)*, Nagoya, 2017, hlm. 22-26, doi: 10.1109/ICCAR.2017.7942654.
- [16] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic," *2014 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Chicago, IL, 2014, hlm. 2997-3004, doi: 10.1109/IROS.2014.6942976.
- [17] W. Wang, H. Gao, Q. Yi, K. Zheng and T. Gu, "An Improved RRT* Path Planning Algorithm for Service Robot," *2020 IEEE 4th Inf. Technol., Netw., Electron. Automat. Control Conf. (ITNEC)*, Chongqing, China, 2020, hlm. 1824-1828, doi: 10.1109/ITNEC48623.2020.9085226.
- [18] H. Jiang, Q. Chen, Y. Zheng dan Z. Xu, "Informed RRT with Adjoining Obstacle Process for Robot Path Planning," *2020 IEEE 20th International Conference on Communication Technology (ICCT)*, 2020, hlm. 1471-1477, doi: 10.1109/ICCT50939.2020.9295809.
- [19] J. Dai, D. Li, J. Zhao, dan Y. Li, "Autonomous Navigation of Robots Based on the Improved Informed-RRT Algorithm and DWA," *Journal of Robotics*, vol. 2022, hlm. 1-9, 2022.
- [20] H. Ryu, dan Y. Park, "Improved informed RRT* using gridmap skeletonization for mobile robot path planning," *International Journal of Precision Engineering and Manufacturing*, vol. 20, no. 11, hlm. 2033-2039, 2019
- [21] M. -C. Kim and J. -B. Song, "Informed RRT* towards optimality by reducing size of hyperellipsoid," *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2015, hlm. 244-248, doi: 10.1109/AIM.2015.7222539.

- [22] M. Aria, "Optimal Path Planning using Informed Probabilistic Road Map Algorithm," *Journal of Engineering Research*, ASSEEE Special Issue, hlm. 1-15, 2021
- [23] M. Fauzi, M.A.R Pohan, "Informed-RRT* Using Hybrid Sampling to Finding Fast Final Path Solution", *Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan*, vol. 9, no. 2, hlm. 94-105, 2021
- [24] H.O. Pakaya, M.A.R. Pohan, M. A. R., "Metode Improved Gaussian Sampling pada Algoritma Rapidly Exploring Random Tree", *Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan*, vol. 9, no. 2, hlm. 106-118, 2021
- [25] D. Ferida, T. Rahajoeningroem, "Analisa Performansi Metode Sampling Goal Biasing dan Metode Sampling Gaussian pada Algoritma Rapidly-exploring Random Tree", *Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan*, vol. 9, no. 2, hlm. 119-128, 2021
- [26] A. Sopa, R. Hartono, "Algoritma Rapidly Exploring Random Tree Star Dengan Integrasi Metode Sampling Goal Biasing, Gaussian, Dan Boundary", *Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan*, vol. 9, no. 2, hlm. 129-138, 2021.
- [27] G. Chen, N. Luo, D. Liu, Z. Zhao, dan C. Liang, "Path planning for manipulators based on an improved probabilistic roadmap method," *Robotics and Computer-Integrated Manufacturing*, vol. 72, hlm. 102196, 2021, doi: 10.1016/j.rcim.2021.102196.
- [28] J. Chen, Y. Zhou, J. Gong dan Y. Deng, "An Improved Probabilistic Roadmap Algorithm with Potential Field Function for Path Planning of Quadrotor," *2019 Chinese Control Conference (CCC)*, 2019, hlm.. 3248-3253, doi: 10.23919/ChiCC.2019.8865585.
- [29] A. A. Ravankar, A. Ravankar, T. Emaru dan Y. Kobayashi, "HPPRM: Hybrid Potential Based Probabilistic Roadmap Algorithm for Improved Dynamic Path Planning of Mobile Robots," in *IEEE Access*, vol. 8, hlm. 221743-221766, 2020, doi: 10.1109/ACCESS.2020.3043333.
- [30] J. D. Gammell, S. S. Srinivasa dan T. D. Barfoot, "Batch Informed Trees (BIT*): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, hlm. 3067-3074, doi: 10.1109/ICRA.2015.7139620.
- [31] M. A. R. Pohan, "LabVIEW Libraries untuk Algoritma Perencanaan Jalur Robotik," *Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan*, vol. 10, no. 1, hlm. 47-62, 2022
- [32] H. Mohammed, M. Jaradat, dan L. Romdhane, L. "RRT N: An improved rapidly-exploring random tree approach for reduced processing times," *2018 11th International Symposium on Mechatronics and its Applications (ISMA) IEEE*, hlm. 1-6. 2018
- [33] R. Mashayekhi, M. Y. I. Idris, M. H. Anisi, I. Ahmedy dan I. Ali, "Informed RRT*-Connect: An Asymptotically Optimal Single-Query Path Planning Method," in *IEEE Access*, vol. 8, hlm. 19842-19852, 2020, doi: 10.1109/ACCESS.2020.2969316.
- [34] F. Islam, J. Nasir, U. Malik, Y. Ayaz dan O. Hasan, "RRT*-Smart: Rapid convergence implementation of RRT* towards optimal solution," *2012 IEEE Int. Conf. Mechatronics Automat.*, 2012, hlm. 1651-1656, doi: 10.1109/ICMA.2012.6284384.