

BAB I

PENDAHULUAN

1.1 Latar Belakang

Robot otonom (*autonomous robot*) adalah sebuah perangkat mekanis yang mampu bergerak bebas di lingkungan yang terdapat rintangan, dapat menjalankan berbagai jenis fungsi, mampu menyimpan program yang telah dibuat dan memperoleh informasi lingkungan melalui sensor yang dimiliki. Pada proses pengendalian *autonomous robot* dibutuhkan perencanaan jalur agar robot mampu mencapai tujuan yang dituju dengan jalur yang dilewati bebas tabrakan. Perencanaan jalur pada robot dapat bermanfaat pada berbagai bidang diantaranya bidang kimia protein *protein folding* [1] *virtual prototype* pada manufaktur [2] dan sistem transportasi [3].

Perencanaan jalur dibagi menjadi dua kelompok yaitu perencanaan jalur berbasis grafik dan berbasis *sampling*. Salah satu contoh perencanaan jalur berbasis grafik yang sering digunakan yaitu algoritma Dijkstra [4] dan A* [5]. Algoritma pencarian A* memiliki kelebihan dalam mencari jalur terpendek dengan perhitungan terkecil pada jalur pencarian [4]. Perencanaan jalur berbasis *sampling* biasa disebut dengan *Sampling Based Planning* (SBP). Algoritma SBP memiliki keunggulan dalam perencanaan jalur yang lebih singkat pada jalur yang sulit. Algoritma SBP yang paling sering digunakan adalah algoritma *Probabilistic Roadmap Planners* (PRM) [6] dan algoritma *Rapidly Exploring Random Tree* (RRT) [7].

Tahun 1998 algoritma RRT diperkenalkan oleh Steven M. LaValle dan James J. Kuffner Jr yang merupakan salah satu algoritma populer dan sering digunakan hingga saat ini [7]. Tahun 2011 RRT dikembangkan menjadi versi lebih optimal oleh Karaman dan Frazolli yang disebut dengan *Rapidly Exploring Random Tree Star* (RRT*) [8]. RRT* merupakan algoritma RRT dalam versi lebih optimal.

Pada algoritma SBP terdapat strategi perencanaan pengambilan sampel untuk mempersempit penjelajahan ruang pencarian jalur. Terdapat beberapa sampel strategi yang populer diantaranya *boundary sampling* [9], *goal biasing sampling* [10] dan *gaussian sampling* [11]. *Gaussian sampling* atau bisa disebut juga distribusi normal merupakan jenis *Gaussian sampling* merupakan fungsi probabilitas yang menunjukkan penyebaran suatu variable acak yang diindekskan berdasarkan nilai waktu dan ruang konfigurasi. Pada metode *gaussian sampling* jumlah *sampling* yang digunakan setiap iterasi hanya sebanyak 1 buah *sampling*. Sehingga membutuhkan banyak jumlah iterasi yang digunakan untuk mencapai titik goal dan waktu komputasi yang dibutuhkan semakin lama. Maka pada penelitian ini, penulis merancang sebuah metode peningkatan dari metode *gaussian sampling* yaitu *metode improved gaussian sampling*. Metode *improved gaussian sampling* memiliki kelebihan dalam pengambilan jumlah *sampling* yang digunakan yaitu sebanyak 10 *sampling*. Dengan adanya peningkatan metode ini maka diharapkan proses pencarian jalur akan lebih optimal dan waktu komputasi akan lebih cepat pada algoritma RRT*.

Valérie telah mempublikasikan metode strategi *gaussian sampling* pada algoritma PRM tetapi dibuat dengan menggunakan program MATLAB [11]. Lars juga telah mempublikasikan perencanaan jalur pada algoritma *probabilistic path*

planning dengan menggunakan distribusi normal pada bahasa C+ [12]. Tetapi belum adanya perencanaan jalur RRT* yang menggunakan metode *improved gaussian sampling* pada bahasa pemrograman LabVIEW. Maka pada tugas akhir ini peneliti akan mengimplementasikan perencanaan jalur RRT* dengan metode pengambilan *sampling* secara *improved gaussian sampling* pada bahasa pemrograman LabVIEW.

1.2 Identifikasi Masalah

Berdasarkan penjelasan pada latar belakang yang telah dikemukakan, maka dapat diidentifikasi permasalahan dalam tugas akhir ini yaitu sebagai berikut.

1. Metode *improved gaussian sampling* memiliki kelemahan berupa pencarian jalur pada algoritma RRT* yang masih belum banyak diimplementasikan pada bahasa pemrograman LabVIEW.
2. Belum adanya perbandingan performansi algoritma RRT* antara metode *gaussian sampling*, *improved gaussian sampling* dan *random sampling* pada bahasa pemrograman LabVIEW.
3. Belum adanya perbandingan performansi sistem algoritma RRT* metode *improved gaussian sampling* terhadap jumlah *sampling*.

1.3 Rumusan Masalah

Adapun rumusan masalah yang diperlukan agar dapat membantu dalam penyelesaian masalah dalam tugas akhir ini yaitu sebagai berikut.

1. Bagaimana merancang metode *improved gaussian sampling* yang dapat mengatasi kekurangan dari metode *gaussian sampling* dalam hal

mengimplementasikan pada algoritma RRT* pada Bahasa pemrograman LabVIEW?

2. Bagaimana perbandingan performansi algoritma RRT* menggunakan metode *improved gaussian sampling*, *gaussian sampling* dan *random sampling*?
3. Bagaimana performansi sistem algoritma RRT* metode *improved gaussian sampling* jika menggunakan jumlah *sampling* yang berbeda-beda?

1.4 Tujuan

Berdasarkan penjelasan pada latar belakang yang telah dikemukakan, maka dapat diidentifikasi tujuan dalam tugas akhir ini yaitu sebagai berikut

1. Merancang metode *improved gaussian sampling* yang dapat menghasilkan *path cost* dan waktu komputasi lebih baik dari metode *gaussian sampling* pada algoritma RRT* pada Bahasa pemrograman LabVIEW.
2. Membandingkan performansi algoritma RRT* yang menggunakan metode *improved gaussian sampling* terhadap algoritma RRT* yang menggunakan metode *gaussian sampling* dan *random sampling*.
3. Membandingkan pengukuran performansi sistem algoritma RRT* metode *improved gaussian sampling* terhadap jumlah *sampling*.

1.5 Batasan Masalah

Beberapa batasan masalah yang didefinisikan oleh penulis dalam pengerjaan tugas akhir adalah sebagai berikut.

1. Pengambilan sampel pada algoritma RRT* yang menggunakan metode *improved gaussian sampling* menggunakan nilai antara 1-10.
2. Pengujian menggunakan lingkungan jenis *obstacle clutter*, *narrow passagges* dan *trap*.
3. Perbandingan performansi algoritma RRT* *improved gaussian sampling* hanya dengan algoritma RRT* dengan metode *gaussian sampling* dan *random sampling*.

1.6 Metode Penelitian

Metode penelitian yang akan dilaksanakan oleh peneliti untuk tugas akhir ini adalah sebagai berikut.

1. Tinjauan Pustaka

Metode mencari dan mempelajari penelitian pah planning yang telah ada sebelumnya.

2. Studi literatur

Mempelajari teknik yang akan penulis gunakan, seperti pembacaan algoritma, perancangan program pada Bahasa LabVIEW dan sistem kerja algoritma.

3. Perancangan Software

Proses perancangan algoritma pada Bahasa pemograman LabVIEW berdasarkan teknik yang telah dipelajari.

1.7 Sistematika Penulisan

Sistematika penulisan pada laporan penelitian ini disusun secara sistematis, berikut merupakan susunan dari sistematika penulisan laporan penelitian ini.

BAB I PENDAHULUAN

Merupakan uraian tentang pendahuluan dari laporan penelitian ini, yang berisikan latar belakang masalah, identifikasi masalah, rumusan masalah, tujuan, batasan masalah, metode penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Merupakan uraian mengenai penelitian-penelitian sebelumnya yang terkait dengan penelitian yang dilakukan dan berisi teori-teori yang melandasi acuan pada laporan penelitian ini.

BAB III PERANCANGAN ALAT

Menjelaskan tentang proses perancangan program dan pembuatan program, yang didalamnya terdapat cara kerja program, pseudo code program, dan hasil perancangan program.

BAB IV PENGUJIAN DAN ANALISIS

Merupakan uraian mengenai proses pengambilan data, hasil pengujian program dan analisis data hasil pengujian.

BAB V PENUTUP

Berisi kesimpulan dari laporan penelitian yang telah dilakukan dan berisikan saran bagi peneliti agar program dapat berkembang lebih baik.