

JUDUL SKRIPSI/THESIS/DISERTASI

THESIS

**TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL DAN
ELEKTRONIKA**

**Ditujukan untuk memenuhi persyaratan
memperoleh gelar Magister Teknik**



**ANGGORO DWI NUR ROHMAN
186060300111007**

**UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
JANUARI 2019**



UNIVERSITAS BRAWIJAYA

JUDUL SKRIPSI/THESIS/DISERTASI

THESIS

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Magister Teknik**

ANGGORO DWI NUR ROHMAN

186060300111007

FAKULTAS TEKNIK

PROGRAM STUDI KONTROL DAN ELEKTRONIKA

MALANG

JANUARI 2019

HALAMAN PERSETUJUAN

Judul : Judul Skripsi/Thesis/Disertasi
Nama : angoro dwi nur rohman
NPM : 186060300111007

Laporan Thesis ini telah diperiksa dan disetujui.

XX Januari 2010

Prof. XXXX
Pembimbing Thesis

HALAMAN PERNYATAAN ORISINALITAS

**Thesis ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : anggoro dwi nur rohman
NPM : 186060300111007
Tanda Tangan :

Tanggal : XX Januari 2010

HALAMAN PENGESAHAN

Thesis ini diajukan oleh :
Nama : angoro dwi nur rohman
NPM : 186060300111007
Program Studi : Kontrol dan Elektronika
Judul Thesis : Judul Skripsi/Thesis/Disertasi

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Teknik pada Program Studi Kontrol dan Elektronika, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Prof. XXXX ()

Penguji : Prof. XXX ()

Penguji : Prof. XXXX ()

Penguji : Prof. XXXXXX ()

@todo

Jangan lupa mengisi nama para penguji.

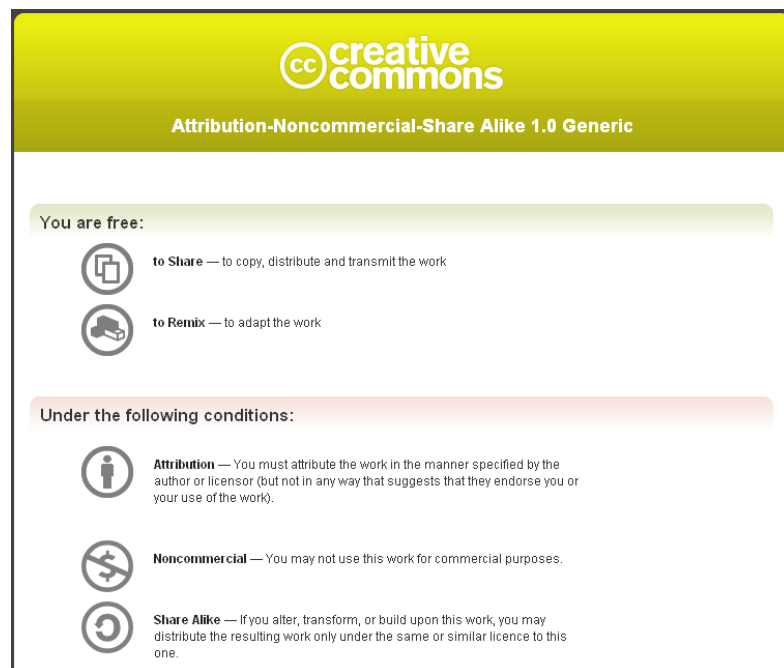
Ditetapkan di : Depok
Tanggal : XX Januari 2010

KATA PENGANTAR

Template ini disediakan untuk orang-orang yang berencana menggunakan \LaTeX untuk membuat dokumen tugas akhirnya. Mengapa \LaTeX ? Ada banyak hal mengapa menggunakan \LaTeX , diantaranya:

1. \LaTeX membuat kita jadi lebih fokus terhadap isi dokumen, bukan tampilan atau halaman.
2. \LaTeX memudahkan dalam penulisan persamaan matematis.
3. Adanya otomatis dalam penomoran caption, bab, subbab, subsubbab, referensi, dan rumus.
4. Adanya otomatisasi dalam pembuatan daftar isi, daftar gambar, dan daftar tabel.
5. Adanya kemudahan dalam memberikan referensi dalam tulisan dengan menggunakan label. Cara ini dapat meminimalkan kesalahan pemberian referensi.

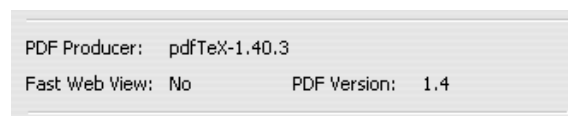
Template ini bebas digunakan dan didistribusikan sesuai dengan aturan *Creative Common License 1.0 Generic*, yang secara sederhana berisi:



Gambar 1: *Creative Common License 1.0 Generic*

Gambar 1 diambil dari http://creativecommons.org/licenses/by-nc-sa/1.0/deed.en_CA. Jika ingin mengetahui lebih lengkap mengenai *Creative Common License 1.0 Generic*, silahkan buka <http://creativecommons.org/licenses/by-nc-sa/1.0/legalcode>. Seluruh dokumen yang dibuat dengan menggunakan template ini sepenuhnya menjadi hak milik pembuat dokumen dan bebas didistribusikan sesuai dengan keperluan masing-masing. Lisensi hanya berlaku jika ada orang yang membuat template baru dengan menggunakan template ini sebagai dasarnya.

Dokumen ini dibuat dengan \LaTeX juga. Untuk meyakinkan Anda, coba lihat properti dari dokumen ini dan Anda akan menemukan bagian seperti Gambar 2. Dokumen ini dimaksudkan untuk memberikan gambaran kepada Anda seperti apa mudahnya menggunakan \LaTeX dan juga memperlihatkan betapa bagus dokumen yang dihasilkan. Seluruh url yang Anda temukan dapat Anda klik. Seluruh referensi yang ada juga dapat diklik. Untuk mengerti template yang disediakan, Anda tetap harus membuka kode \LaTeX dan bermain-main dengannya. Penjelasan dalam PDF ini masih bersifat gambaran dan tidak begitu mendetail, dapat dianggap sebagai pengantar singkat. Jika Anda merasa kesulitan dengan template ini, mungkin ada baiknya Anda belajar sedikit dasar-dasar \LaTeX .



Gambar 2: Dokumen Dibuat dengan PDF \LaTeX

Semoga template ini dapat membantu orang-orang yang ingin mencoba menggunakan \LaTeX . Semoga template ini juga tidak berhenti disini dengan ada kontribusi dari para penggunanya. Kami juga ingin berterima kasih kepada Andreas Febrian, Lia Sadita, Fahrurrozi Rahman, Andre Tampubolon, dan Erik Dominikus atas kontribusinya dalam template ini.

Depok, 30 Desember 2009

anggoro dwi nur rohman

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : angoro dwi nur rohman
NPM : 186060300111007
Program Studi : Kontrol dan Elektronika
Fakultas : Teknik
Jenis Karya : Thesis

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty Free Right)** atas karya ilmiah saya yang berjudul:

Judul Skripsi/Thesis/Disertasi

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : XX Januari 2010
Yang menyatakan

(angoro dwi nur rohman)

ABSTRAK

Nama : anggoro dwi nur rohman
Program Studi : Kontrol dan Elektronika
Judul : Judul Skripsi/Thesis/Disertasi

@todo

Tuliskan abstrak laporan disini.

Kata Kunci:

@todo

Tuliskan kata kunci yang berhubungan dengan laporan disini

ABSTRACT

Name : angoro dwi nur rohman
Program : Kontrol dan Elektronika
Title : Unknown Title for Final Report/Thesis/Disertation

@todo

Write your abstract here.

Keywords:

@todo

Write up keywords about your report here.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PERNYATAAN ORISINALITAS	iii
LEMBAR PENGESAHAN	iv
KATA PENGANTAR	v
LEMBAR PERSETUJUAN PUBLIKASI ILMIAH	vii
ABSTRAK	viii
Daftar Isi	x
Daftar Gambar	xii
Daftar Tabel	xiii
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi dan Perumusan Masalah	3
1.3 Tujuan dan Manfaat	4
2 KAJIAN PUSTAKA	5
2.1 Pemodelan Robot	5
2.2 Formasi Multi Robot	8
2.2.1 Pendahuluan Formasi Multi Robot	8
2.2.1.1 Teori Graf	8
2.2.1.2 Teori Kekakuan Graf	9
2.2.2 Kendali Formasi Multi-Robot	9
2.3 Solusi Persamaan Differensial Secara Numerik	11
2.3.1 Stabilitas Metode Euler	12

3	KERANGKA KONSEP PENELITIAN	13
3.1	Definisi Permasalahan Kendali Formasi	13
3.2	Permasalahan dan Solusi	14
4	METODE PENELITIAN	15
4.1	Strategi Kendali Multi Robot	15
4.1.1	Kendali Robot	15
4.1.1.1	State Feedback	15
4.1.1.2	Desain Kendali	17
4.1.2	Kendali Formasi Multi Robot	17
4.1.2.1	Strategi Penentuan Koordinat Tetangga	17
4.2	Kestabilan Perangkat Percobaan	19
4.2.1	Kestabilan Model	20
4.2.2	Rencana Hardware-in-Loop	23
4.2.3	Rencana Uji Lapangan	23
5	PERINTAH DALAM UITHESIS.STY	24
5.1	Mengubah Tampilan Teks	24
5.2	Memberikan Catatan	24
5.3	Menambah Isi Daftar Isi	25
5.4	Memasukan PDF	25
5.5	Membuat Perintah Baru	29
6	??	30
7	KESIMPULAN DAN SARAN	31
7.1	Kesimpulan	31
7.2	Saran	31
	DAFTAR REFERENSI	32
	LAMPIRAN	1
	Lampiran 1	2

DAFTAR GAMBAR

1	<i>Creative Common License 1.0 Generic</i>	v
2	Dokumen Dibuat dengan PDFLatex	vi
2.1	(a) Geometri Robot (Correia, dkk (2012)) (b) Grafik Gaya Robot . .	5
3.1	Kerangka Penelitian	13
4.1	Strategi Penentuan Koordinat	18
4.2	(a)Grafik Hardware-in-the-loop (Ledin (1999)). (b) HIL Kendali Multi-Robot.	19
4.3	(a) $w_1 = -6; w_2 = 3; w_3 = 3$. (b) $w_1 = 0; w_2 = 6; w_3 = -6$ (c) $w_1 =$ $6; w_2 = 6; w_3 = 6$	22

DAFTAR TABEL

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Multi-robot adalah sekelompok mobile robot yang bekerja sama untuk mencapai tujuan tertentu. Tujuan tersebut dapat menjadi sebuah topik dalam penelitian seperti yang dipaparkan dalam literatur oleh Parker (2003), yaitu mendemonstrasikan atau menerapkan tingkahlaku biologis; komunikasi antara robot secara langsung atau tidak langsung; pengembangan arsitektur kendali yang memungkinkan untuk diterapkan pada robot yang lebih banyak; memecahkan masalah dalam eksplorasi, pemetaan, dan lokalisasi; memecahkan masalah dalam transportasi obyek pada multi-robot; permasalahan dalam koordinasi pergerakan, seperti kendali formasi; dan topik yang lebih terkemuka seperti *machine learning* terhadap robot. Pada penelitian ini akan ditunjukkan ke permasalahan kendali formasi. Kendali formasi ini adalah salah satu permasalahan dalam kerjasama antar robot. Kendali formasi memiliki tujuan untuk mengendalikan sekelompok agen dalam mencapai formasi tertentu dan dapat mempertahankan formasi tersebut ketika bermanuver menuju arah yang diinginkan. Sehingga kemampuan ini tepat diterapkan dalam bidang militer, seperti patroli yang dilakukan oleh sejumlah kendaraan tanpa awak untuk tugas penyelamatan dan pencarian didaerah berbahaya. Dalam literatur yang dipaparkan oleh Guanhua, dkk (2013), permasalahan kendali formasi ditunjukkan pada pengembangan arsitektur. Pengembangan dilakukan karena untuk memecahkan permasalahan dalam hal mendistribusikan tugas pada setiap robot yang terbatas dan juga berdasarkan keterbatasannya pada robot itu sendiri. Selain itu juga dilakukan pengembangan dalam algoritma strategi, contoh strategi tersebut adalah *leader-follower*, struktur virtual, berdasarkan tingkahlaku, menggunakan teori graph, dan memanfaatkan medan potensial buatan.

Dalam literatur oleh Oh, dkk (2015), kendali formasi dikategorikan menjadi 3 bagian, yaitu berdasarkan posisi, perpindahan, dan jarak. Ketiga bagian tersebut tertuju pada jawaban dari pertanyaan, "variable apa yang digunakan sebagai sensor" dan "variable apa yang aktif dikendalikan oleh sistem multi-agent untuk mencapai formasi yang diinginkan". Untuk menetapkan variable sebagai sensor dapat dilakukan berdasarkan ketentuan kemampuan individu agent. Berikut adalah penjelasan singkat dari ketiga bagian tersebut: Pada formasi berdasarkan posisi, dimana

agent diharuskan memiliki kemampuan untuk mengetahui koordinatnya sendiri berdasarkan koordinat global. Sehingga, koordinat tujuan didistribusikan kepada setiap agent dan agent bekerja untuk mencapai koordinat tersebut. Karena itu, kebutuhan individu untuk berinteraksi dengan individu lain sangat kecil. Metode formasi ini pada praktiknya, interaksi antar individu dilakukan untuk menangani masalah disturbance, saturasi akselerasi, dan lain-lain. Karena metode ini membutuhkan kemampuan untuk mengetahui koordinat global, dibutuhkan biaya yang lebih dibanding metode lain dalam perangkat sensor yang *advance*, seperti sensor GPS; Pada formasi kendali berdasarkan perpindahan, secara individu agent tidak mengetahui koordinatnya berdasarkan koordinat global. Akan tetapi, individu agent memiliki koordinatnya sendiri terhadap individu agent tetangganya dan harus dilakukan penyearahan terhadap koordinat setiap robot dengan koordinat global. Koordinat relatif itulah yang menjadi variable yang dikendalikan oleh agent. Oleh karena itu agent diharuskan memiliki kemampuan untuk mengetahui perpindahan dari individu lain berdasarkan koordinat agent itu sendiri, dan semua agent harus menyearahkan koordinatnya berdasarkan koordinat global, serta dibutuhkan interaksi antara individu lain untuk mencapai formasi yang diinginkan. Permasalahan pada metode ini ditujukan pada kendali formasi pada agent yang bersifat heterogen, pemeliharaan dalam komunikasi, dan kemampuan dalam menghindari rintangan; Pada formasi berdasarkan jarak, dimana setiap individu agent memiliki koordinatnya masing-masing dan tidak perlu disearahkan dengan koordinat global. Variable yang dikendalikan pada metode ini adalah variabel jarak antar agent yang terhubung, sehingga dibutuhkan kemampuan untuk agent saling berkomunikasi antar agent lain. Permasalahan pada metode ini ditujukan pada analisa stabilitas secara general; tapi hasil penelitian untuk formasi segitiga telah dipaparkan kestabilannya. Permasalahan pada praktik juga masih perlu untuk dilakukan investigasi pada penerapan model yang lebih nyata. Pemeliharaan komunikasi juga menyumbang dalam permasalahan secara praktik, dan kemampuan untuk menghindari rintangan juga dibutuhkan.

Dari ketiga metode tersebut, formasi berdasarkan jarak merupakan metode yang dimungkinkan untuk diterapkan sensor lebih sedikit dari metode lainnya. Teknologi komunikasi sekarang pun juga sudah bisa dikatakan bisa untuk diterapkan pada metode tersebut secara praktiknya. Pemaparan dengan menggunakan model yang lebih real sangat dibutuhkan sebagai kontribusi dalam bidang kendali multi-robot. Dengan harapan penerapan real model tersebut dapat bermanfaat terhadap masyarakat luas.

1.2 Identifikasi dan Perumusan Masalah

Tiga kategori metode formasi yaitu berdasarkan posisi, perpindahan, dan jarak hampir diperlukan analisa terhadap model yang nyata. Pada penelitian oleh Rozenheck, dkk (2015), yang memaparkan permasalahan kendali formasi berdasarkan jarak menggunakan kendali *Proportional-Integral*(PI). Peneliti memberikan kecepatan referensi secara konstan terhadap salah satu dari agent. Lalu agent lainnya memberikan respon untuk tetap menjaga formasi yang diinginkan. Tidak dijelaskan alasan oleh peneliti kenapa salah satu agent diberi kecepatan referensi, akan tetapi metode tersebut hampir sama dengan strategi *leader-follower*. *Leader-follower* mengharuskan agent tetangga untuk beradaptasi terhadap perubahan tetangga lainnya secara spesifik. Akan tetapi ada perbedaan antara *leader-follower* dengan metode berdasarkan jarak, yaitu terhadap metode pertukaran informasinya. Pada *Leader-follower*, agent yang berperan sebagai *leader* tidak memberikan informasi terhadap *follower*-nya. Tugas *follower* adalah untuk beradaptasi terhadap pergerakan *leader*, sedangkan *leader* bertugas untuk bermanuver sesuai jalur yang diinginkan. Sedangkan pada metode berdasarkan jarak, terdapat dua jenis, *direct* dan *undirect*. Strategi *leader-follower* lebih sama dengan jenis *direct*. Kedua jenis ini berhubungan dengan konfigurasi jaringan. Jenis *direct* adalah jaringan satu arah, dimana alur informasi diberikan secara satu arah dari agent ke tetangga atau sebaliknya. Sedangkan jenis *undirect* adalah jaringan dua arah, dimana setiap agent dengan tetangganya saling bertukar informasi. Metode tersebut menghasilkan formasi pada multi agent tetap terjaga ketika salah satu agent diberikan kecepatan secara konstan dan memberikan respon yang baik ketika pengaturan konstanta PI dengan tepat. Tetapi model yang digunakan masih menggunakan model orde satu, dengan kata lain metode tersebut dimungkinkan untuk diterapkan model yang lebih kompleks. Penelitian oleh Correia, dkk (2012), memaparkan formula model orde dua *holonomic mobile robot* secara detail dan kompleks. Model tersebut dapat digunakan untuk diterapkan metode formasi berdasarkan jarak sebagai langkah awal analisa terhadap model yang nyata. Karena kendali formasi yang digunakan adalah kendali-PI, maka untuk kendali robot keseluruhan akan dikembangkan menggunakan metode *self-tune control*.

Dalam penelitian ini akan digunakan batasan-batasan permasalahan sebagai berikut :

1. Variable sensor yang digunakan adalah jarak antar individu robot.
2. Komunikasi antar robot diasumsikan ideal, dalam artian percobaan tidak dilakukan diluar jarak jangkauan prangkat komunikasi.

Berikut adalah beberapa point permasalahan yang ditunjukkan pada penelitian ini, yaitu:

1. Bagaimanakah strategi untuk kendali formasi apabila variable yang dikendalikan adalah jarak antar robot?.
2. Bagaimanakah pergerakan kendali formasi berdasarkan jarak apabila model yang digunakan adalah holonomic mobile robot ?.

1.3 Tujuan dan Manfaat

Tujuan dari penelitian ini adalah

1. Mengetahui strategi untuk kendali formasi apabila variable yang dikendalikan adalah jarak antar robot.
2. Mengetahui pergerakan kendali formasi berdasarkan jarak apabila model yang digunakan adalah holonomic mobile robot.

Manfaat dari penelitian ini adalah

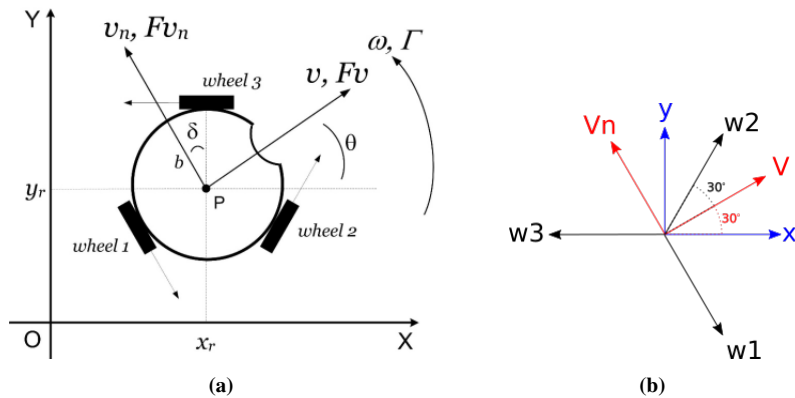
1. Memberikan referensi untuk permasalahan kendali multi-robot, khususnya pada permasalahan kendali formasi, terhadap model yang lebih nyata.
2. Membuka peluang penelitian dibidang kendali mengenai kendali formasi pada kendali multi-robot dilingkungan Fakultas Teknik Elektro, Universitas Brawijaya.

BAB 2

KAJIAN PUSTAKA

2.1 Pemodelan Robot

Robot menggunakan 3 aktuator penggerak dengan roda *omniwheel*, sehingga robot dapat bergerak kesegala arah. Pemasangan roda *omniwheel* memiliki sudut 120° terhadap roda lainnya. Sehingga setiap roda memiliki gaya dengan arah 90° dari sudut pemasangannya. Agar robot bergerak kesegala arah, ketiga aktuator harus dikendalikan untuk menghasilkan resultan gaya dengan arah yang diinginkan.



Gambar 2.1: (a) Geometri Robot (Correia, dkk (2012)) (b) Grafik Gaya Robot

Kinematika robot dapat dirumuskan menjadi

$$\dot{\mathbf{x}}_p = R^T(\theta) \cdot \dot{\mathbf{x}}_r, \quad (2.1)$$

dimana $R(\theta)$ adalah matrik rotasi ortogonal

$$R(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Koordinat robot dideskripsikan menggunakan vector $\mathbf{x}_p = [x_p \ y_p \ \theta]^T$, dimana x_p dan y_p adalah titik pusat, P , pada frame robot dan θ_p adalah selisih sudut antara *angular* frame global dengan robot. Vector $\dot{\mathbf{x}}_r = [\dot{x}_r \ \dot{y}_r \ \dot{\theta}_r]^T$ mendeskripsikan variable kecepatan terhadap titik pusat, P , dimana w sebagai kecepatan angular robot terhadap frame global. Karena robot memiliki aktuator, maka kecepatan roda memi-

liki hubungan terhadap kecepatan robot, dengan kata lain kecepatan pada titik pusat adalah sebuah fungsi dengan kecepatan roda sebagai parameternya. Untuk mendapatkan persamaan tersebut, maka dapat dianalisis dengan hukum *Power*. Apabila didefinisi hubungan antara gaya resultan robot dengan gaya yang dihasilkan roda

$$\begin{aligned} F_{\dot{x}_r} &= \cos 90^\circ \cdot F_{w1}(t) + \cos 30^\circ \cdot F_{w2}(t) + (-\cos 30^\circ) \cdot F_{w3}(t) \\ F_{\dot{y}_r} &= (-1) \cdot F_{w1}(t) + \cos 60^\circ \cdot F_{w2}(t) + \cos 60^\circ \cdot F_{w3}(t) \\ \Gamma &= d \cdot F_{w1}(t) + d \cdot F_{w2}(t) + d \cdot F_{w3}(t) \end{aligned}$$

dimana d adalah jarak dari titik P ke lokasi roda, maka akan didapat matriks geometri antara $F_R = \begin{bmatrix} F_{\dot{x}_r} & F_{\dot{y}_r} & \Gamma \end{bmatrix}^T$ dengan $F_w = \begin{bmatrix} F_{w1} & F_{w2} & F_{w3} \end{bmatrix}^T$

$$\begin{aligned} F_R &= \begin{bmatrix} 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ -1 & \frac{1}{2} & \frac{1}{2} \\ l & l & l \end{bmatrix} \cdot F_w \\ F_R &= A \cdot F_w. \end{aligned} \quad (2.2)$$

Dalam kasus robot, *power* yang dihasilkan oleh setiap roda sama dengan *power* dari robot itu sendiri (Hacene, dkk (2019)). Dengan menggunakan persamaan (2.2) akan menghasilkan persamaan kinematika robot menggunakan 3 roda *omniwheel*

$$\begin{aligned} P_w &= P_R \\ F_w^T \cdot \dot{x}_w &= F_R^T \cdot \dot{x}_r \\ F_w^T \cdot \dot{x}_w &= (A \cdot F_w)^T \cdot \dot{x}_r \\ \dot{x}_w &= A^T \cdot \dot{x}_r \\ \dot{x}_r &= (A^T)^{-1} \cdot \dot{x}_w. \end{aligned} \quad (2.3)$$

dengan mensubstitusi persamaan (2.3) pada (2.1)

$$\dot{x}_p = R^T(\theta) \cdot (A^T)^{-1} \cdot \dot{x}_w \quad (2.4)$$

Pergerakan robot juga dideskripsikan secara dinamika menggunakan hukum pergerakan dari *Newton*.

$$F_{\dot{x}_r}(t) - B_{\dot{x}_r} \cdot \dot{x}_r(t) - C_{\dot{x}_r} \cdot \text{sgn}(\dot{x}_r(t)) = M \cdot \ddot{x}_r(t) \quad (2.5)$$

$$F_{\dot{y}_r}(t) - B_{\dot{y}_r} \cdot \dot{y}_r(t) - C_{\dot{y}_r} \cdot \text{sgn}(\dot{y}_r(t)) = M \cdot \ddot{y}_r(t) \quad (2.6)$$

$$\Gamma(t) - B_{\dot{\theta}} \cdot \dot{\theta}(t) - C_{\dot{\theta}} \cdot \text{sgn}(\dot{\theta}(t)) = I \cdot \ddot{\theta}(t) \quad (2.7)$$

Dimana B_i adalah *viscous frictions* yang mempresentasikan perbandingan terbalik dari gaya yang bersifat linier terhadap gaya dorong dan kecepatan robot. $C_i \cdot \text{sgn}(\dot{i})$ adalah *coulumb frictions* yang mempresentasikan perbandingan terbalik terhadap perubahan kecepatan, dimana tanda bilangan berubah kebalikan dari kecepataannya.

$$\text{sgn}(\alpha) = \begin{cases} 1, & \alpha > 0 \\ 0, & \alpha = 0 \\ -1, & \alpha < 0. \end{cases}$$

seperti pada persamaan (2.2), resultan gaya robot berhubungan dengan gaya roda. Maka gaya roda dapat dideskripsikan dengan menghubungkan antara gaya yang dihasilkan oleh motor

$$F_{wi} = \frac{\tau_i(t)}{r_i} \quad (2.8)$$

dimana $\tau_i(t)$ adalah torsi dari motor

$$\tau_i(t) = l_i \cdot K_{ti} \cdot i_{ai}(t). \quad (2.9)$$

Untuk mendapatkan persamaan $i_{ai}(t)$, dapat digunakan deskripsi persamaan dinamika motor

$$u_i(t) = L_{ai} \cdot \frac{di_{ai}(t)}{dt} + R_{ai} \cdot i_{ai}(t) + K_{vi} \cdot \omega_{mi} \quad (2.10)$$

dimana L_{ai} dan R_{ai} adalah Induktansi dan resistansi armature motornya. K_{vi} adalah konstanta torsi motor dimana dalam satuan SI yang sama dengan K_v . Dalam praktiknya apabila motor dalam kecepatan *steady state* maka $\frac{di_{ai}}{dt}$ bernilai kecil, dan dalam persamaan (2.10) nilai induktansi dapat diabaikan.

Penjabaran dinamika robot bisa diubah dalam bentuk *state-space*

$$\dot{x}(t) = A_r \cdot x(t) + B_r \cdot u(t) + K \cdot \text{sgn}(x(t)) \quad (2.11)$$

$$y(t) = C \cdot x(t) \quad (2.12)$$

dimana vektor *state* adalah $x(t) = [x_p \ y_p \ \theta \ \dot{x}_r \ \dot{y}_r \ \dot{\theta}_r]^T$, dan vektor output $y(t) = [x_p \ y_p \ \theta]^T$. Dimana $l = l_{1...3}, r = r_{1...3}, R_a = R_{a1...3}$ and $K_t = K_{t1...3}$, maka

didapat matriks yang dapat mendeskripsikan sistem robot

$$\begin{aligned}
 A_r &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\frac{3.l^2.K_t^2}{2.M.R_a.r^2} - \frac{B_{\dot{x}_r}}{M} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{3.l^2.K_t^2}{2.M.R_a.r^2} - \frac{B_{\dot{y}_r}}{M} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{3.l^2.K_t^2}{2.I.R_a.r^2} - \frac{B_{\dot{\theta}_r}}{I} \end{bmatrix}, \\
 B_r &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{l.K_t}{R_a.r} \cdot \frac{\cos(30^\circ)}{M} & -\frac{l.K_t}{R_a.r} \cdot \frac{\cos(30^\circ)}{M} \\ \frac{l.K_t}{R_a.r} \cdot \frac{-1}{M} & \frac{l.K_t}{R_a.r} \cdot \frac{\cos(60^\circ)}{M} & \frac{l.K_t}{R_a.r} \cdot \frac{\cos(60^\circ)}{M} \\ \frac{l.K_t}{R_a.r} \cdot \frac{b}{I} & \frac{l.K_t}{R_a.r} \cdot \frac{b}{I} & \frac{l.K_t}{R_a.r} \cdot \frac{b}{I} \end{bmatrix}, K = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{C_{\dot{x}_r}}{M} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{C_{\dot{y}_r}}{M} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{C_{\dot{\theta}_r}}{M} \end{bmatrix}, \\
 C &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

2.2 Formasi Multi Robot

Pembahasan kendali formasi mutli-robot dikutip dari paper oleh Rozenheck, dkk (2015). Dimana peneliti membahas mengenai kendali formasi robot berdasarkan jaraknya lalu dikendalikan dengan kendali PI. Dari subbab ini akan dirangkum dari paper tersebut, yaitu mulai dari pendahuluan sampai kendalinya.

2.2.1 Pendahuluan Formasi Multi Robot

2.2.1.1 Teori Graf

Suatu graf \mathcal{G} , dinotasikan sebagai $\mathcal{G} = (\mathcal{V}, \varepsilon)$, merupakan pasangan \mathcal{V} dan ε , di mana \mathcal{V} merupakan himpunan tak kosong berisikan simpul pada graf tersebut dan ε merupakan himpunan sisi pada graf tersebut. Secara formal, himpunan ε dapat dinyatakan sebagai suatu koleksi subhimpunan berkardinalitas dua dari himpunan \mathcal{V} , atau dalam notasi matematika $\varepsilon \subseteq \mathcal{V} \times \mathcal{V}$. Sebuah \mathcal{G} dikatakan tak berarah (*undirected graph*), dimana himpunan sisi terdiri dari pasangan node (i, j) , maka

sisi tersebut tidak memiliki urutan arah antara node i dengan j . Dinotasikan $n \triangleq |\mathcal{V}|$ sebagai jumlah dari node dan $m \triangleq |\mathcal{E}|$ sebagai jumlah dari sisinya. Apabila $(i, j) \in \mathcal{E}$ maka dapat disebut node i dan j berdekatan (*adjacent*). Himpunan dari node yang terhubung dari setiap simpul i dinotasikan dengan $\mathcal{N}_i \triangleq \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$, dan juga $i \sim j$. Matriks insidensi (*incidence*), $E \in \mathbb{R}^{n \times m}$, adalah matriks $\{0, \pm 1\}$ dimana baris matriks mengindikasikan simpulnya dan kolomnya sebagai sisinya. Matriks *laplacian* didefinisikan dengan $L(\mathcal{V}) = EE^T$

2.2.1.2 Teori Kekakuan Graf

Koordinat multi dimensi adalah konfigurasi matrik vector yang terdiri dari beberapa koordinat node, $x = [x_1^T \ \dots \ x_n^T]^T \in \mathbb{R}^{2n}$, dimana $x_i \in \mathbb{R}^2$ dan $x_i \neq x_j$ untuk semua $i \neq j$. Didefinisikan sebuah kerangka (*framework*), dinotasikan dengan $\mathcal{G}(x)$, adalah graf tak berarah \mathcal{G} dengan konfigurasi x , dimana simpul i pada graf dipetakan kedalam koordinat x_i . Misalkan $(i, j) \in \mathcal{E}$ sama dengan sisi ke k dari graf langsung dan mendefinisikan vektor sisi dari kerangka, atau dapat disebut sebagai vektor posisi relatif, dengan $e_k \triangleq x_j - x_i$. Untuk semua vektor sisi dapat dinotasikan dengan $e = [e_1^T \ \dots \ e_m^T]^T \in \mathbb{R}^{2m}$.

Apabila kerangka $\mathcal{V}(x)$ dengan vektor sisi $\{e_k\}_{k=1}^m$, maka didefinisikan fungsi sisi (*edge function*), $F : \mathbb{R}^{2n} \times \mathcal{V} \rightarrow \mathbb{R}^m$ dengan

$$F(x, \mathcal{E}) \triangleq [\|e_1\|^2 \ \dots \ \|e_m\|^2]^T \quad (2.13)$$

Matrik kekakuan $R(x)$ yang berhubungan erat dengan kerangka $\mathcal{G}(x)$ dapat didefinisikan dengan *Jacobian* dari fungsi sisi (Rozenheck, dkk (2015)),

$$\begin{aligned} R(x) &\triangleq \frac{\partial F(x, \mathcal{G})}{\partial x} \in \mathbb{R}^{m \times 2n} \\ &\triangleq \text{diag}(e_i^T)(E^T \otimes I_2) \end{aligned} \quad (2.14)$$

2.2.2 Kendali Formasi Multi-Robot

Pembahasan kendali dari formasi multi robot menggunakan gradient control. Apabila $n(n \geq 2)$ dimodelkan sebagai titik yang memiliki masa jenis bergerak diatas dimensi 2 (*Euclidean Space*), maka pergerakan dimodelkan dengan

$$\dot{x}_i(t) = u_i(t), \quad i = 1, \dots, n. \quad (2.15)$$

dimana $x_i(t) \in \mathbb{R}^2$ adalah posisi dari robot- i dan $u_i(t) \in \mathbb{R}^2$ adalah input dari kendali. Dinotasikan $d \in \mathbb{R}^m$ adalah vector jarak dimana isi dari matrik tersebut adalah d_k^2 yang mempresentasikan jarak yang diinginkan antara setiap robot i dan j untuk sisi $(i, j) \in \epsilon$. Lalu didefinisi persamaan potensial yang memiliki hubungan antara jarak robot yang diinginkan dengan jarak yang sekarang

$$\Phi(e) = \frac{1}{2} \sum_{k=1}^m (||e_k||^2 - d_k^2)^2. \quad (2.16)$$

Pengamatan dilakukan agar $\Phi(e) = 0$ jika dan hanya jika $||e_k||^2 = d_k^2, \forall k = 1, \dots, m$. Kendali dari setiap robot menggunakan gradien negatif dari fungsi potensial

$$u_i(t) = -\left(\frac{\partial \Phi(e)}{\partial x_i}\right) = -\sum_{j \sim i} \left(||e_k||^2 - d_k^2 \right) \cdot e_k. \quad (2.17)$$

Dengan itu, dapat disubstitusi kedalam persamaan dinamika pada persamaan (2.15)

$$\dot{x}(t) = -R(x)^T R(x)x(t) + R(x)^T d \quad (2.18)$$

Penambahan refrensi kecepatan pada salah satu robot dapat menjadikan formasi bermanuver. Skema kendali secara general dapat didefinisi dengan

$$\dot{x}(t) = u(t) + B \cdot v_{ref} \quad (2.19)$$

$$u(t) = -R(x)^T C \left(R(x)x(t) - d \right) \quad (2.20)$$

dimana $B \in \mathbb{R}^{2n \times 2}$ digunakan untuk indikasi robot ke i sebagai leader atau penerima kecepatan refrensinya, $v_{ref} \in \mathbb{R}^2$ sebagai kecepatan refrensi, dan C adalah konstanta pengendali yang akan digantikan dengan algoritma kendali. Dengan menerapkan kendali Proportional-Integral, konstanta C pada persamaan (2.20) dapat diubah dengan

$$u(t) = u_{k_p}(t) + u_{k_i}(t) \quad (2.21)$$

$$u_{k_p}(t) = -R(x)^T k_p \left(R(x)x(t) - d \right) \quad (2.22)$$

$$u_{k_i}(t) = -R(x)^T k_i \int_0^T \left(R(x)x(\tau) - d \right) d\tau. \quad (2.23)$$

Lalu pada bagian integrator(k_i), menghasilkan *state* baru

$$\dot{\xi}(t) = k_i \left(R(x)x(t) - d \right). \quad (2.24)$$

Dengan itu dapat digabungkan menjadi persamaan *state-space* menggunakan persamaan (2.18)

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\xi}(t) \end{bmatrix} = \begin{bmatrix} -k_p R(x)^T R(x) & -R(x)^T \\ k_i R(x) & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \xi(t) \end{bmatrix} + \begin{bmatrix} k_p R(x)^T \\ -k_i I \end{bmatrix} d + \begin{bmatrix} B \\ 0 \end{bmatrix} v_{ref} \quad (2.25)$$

2.3 Solusi Persamaan Differensial Secara Numerik

Persamaan (2.11) dan (2.12) adalah persamaan differensial kontinu orde satu. Dalam memecahkan persamaan differensial dapat dilakukan dalam bentuk kontinyu atau numerik. Dalam kasus kendali, persamaan differensial dikalkulasi menggunakan komputer, sehingga persamaan tersebut dapat dicari solusi pendekatannya menggunakan cara numerik. Persamaan orde satu dapat direpresentasikan dengan persamaan

$$\dot{x}(t) = f(x, t), t_0 \leq t \leq t_f \quad (2.26)$$

$$y(t_0) = x(t_0). \quad (2.27)$$

Dimana $x(t) \in \mathbb{R}^n$, adalah vector yang setiap iterasi waktu berubah, $f(x, t) \in \mathbb{R}^n$ adalah fungsi sistem, t_0 dan t_f adalah waktu inisial dan waktu akhir. Pada persamaan (2.26) dan (2.27) adalah persamaan dengan permasalahan nilai inisial Fabien (2009).

Apabila $t(0) = t(t_i)$ maka $t(1) = t(0) + h$, dimana h adalah perubahan kecil yang memiliki hubungan terhadap waktu. Didalam metode algoritma yang akan dibahas, h juga dapat disebut sebagai *step size*, dan juga $t[k] = t[k-1] + h$ adalah bentuk diskretnya untuk $k = 0, 1, 2, 3, \dots$

Apabila $y(t[k])$ adalah nilai inisial ketika waktu $t[k]$, maka menggunakan deret *taylor* akan didapat pendekatan solusi untuk $y(t[k+1])$. Menggunakan orde pertama deret *taylor* saja maka didapat persamaan diskret solusi pendekatan $y(t[k]) \approx y[k]$

$$y[k+1] = y[k] + f(y[k])h. \quad (2.28)$$

Pendekatan lain dari persamaan (2.28) dengan mendefinisikan turunan $y(t[k])$ sebagai

$$\dot{y}(t[k]) = \frac{y[k+1] - y[k]}{h}. \quad (2.29)$$

Persamaan (2.28) dan (2.29) dinamakan dengan persamaan *explicite Euler method* dan *forward Euler formula*. Apabila persamaan (2.29) disubstitusikan pada (2.26)

dan (2.27) maka didapat persamaan (2.28). Untuk diterapkan dalam komputer, dapat mengikuti algoritme 2.1.

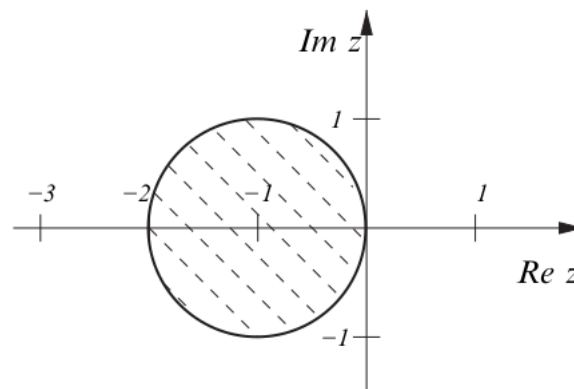
Algoritme 2.1: *Explicite Euler Method*

Masukan: Integer $N > 0$, $h = (t_f - t_i)/N$, $t[0] = t_i$, $y[0] = y[t_i] = y_i$.

Keluaran: $y[k]$, $k = 1, 2, \dots, N$.

- 1 Untuk $k = 0, 1, \dots, N - 1$ Lakukan
 - 2 $y[k + 1] = y[k] + h \cdot f(y[k])$
 - 3 $t[k + 1] = t[k] + h$
-

2.3.1 Stabilitas Metode Euler



Gambar 2.2: Area stabilitas metode explicit euler. (Fabien (2009))

Properti dari stabilitas metode Euler dapat diperoleh dengan mendefinisikan persamaan differensial secara general $\dot{x} = \alpha x$, dimana α adalah bilangan complex dari parameter sistem. Dengan menggunakan pendekatan sebelumnya maka persamaan masalah dapat didefinisikan

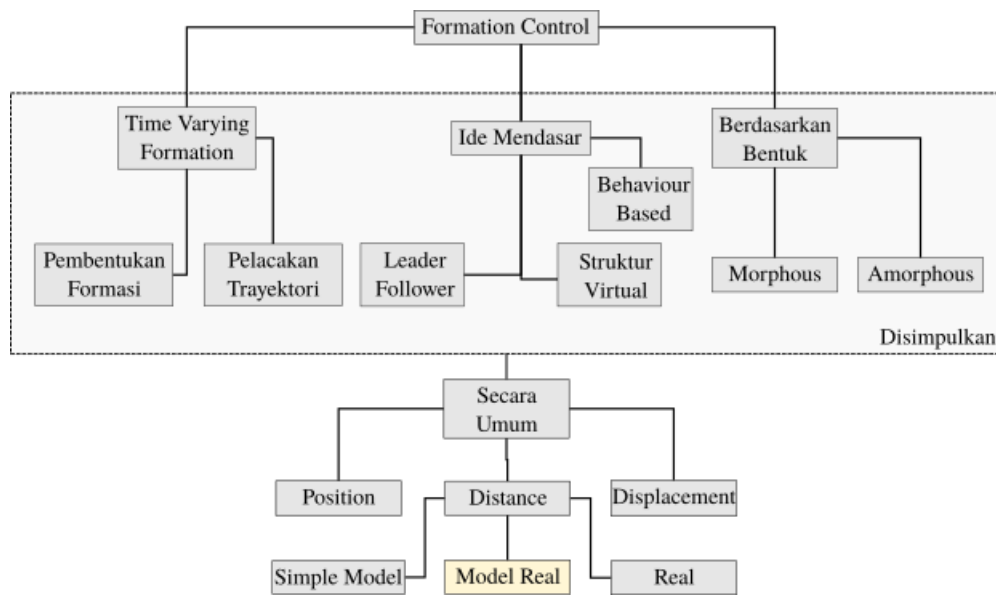
$$y[k + 1] = (1 + h\lambda)y[k] = (1 + z)y[k] = R(z)y[k]. \quad (2.30)$$

Dari persamaan (2.30), sistem akan stabil apabila $|R(z)| \leq 1$. Jika digambarkan dalam grafik complex stabilitas maka dapat dilihat pada gambar 2.2

BAB 3

KERANGKA KONSEP PENELITIAN

Krangka konsep penelitian akan dibahas mengenai potensi permasalahan yang timbul dalam topik kendali formasi. Krangka penelitian ini berdasarkan literatur oleh Oh, dkk (2015), dimana didalam literatur tersebut, peneliti menguraikan berbagai metode yang digunakan dalam bidang kendali multi-robot, khususnya dalam kendali formasi.



Gambar 3.1: Kerangka Penelitian

3.1 Definisi Permasalahan Kendali Formasi

Kendali formasi adalah kendali multi-agent untuk mencapai suatu formasi yang diinginkan. Banyak metode yang telah digunakan berdasarkan berbagai macam kategori. Dapat diperhatikan dalam gambar 3.1, dari berbagai metode tersebut dapat disimpulkan dalam 3 kategori secara general. Yaitu berbasis posisi, pergerakan, dan jarak. Pembagian kategori tersebut berdasarkan kemampuan sensor yang digunakan dan penggunaan komunikasi dalam metodenya. Dari ketiga kategori tersebut, kendali formati berbasis jarak sangat dibutuhkan pembahasan mengenai penerapan metode tersebut pada agent yang nyata. Pada penelitian oleh Rozenheck, dkk (2015), kendali formasi berdasarkan jarak menggunakan kendali PI untuk mengendalikan multi-robot dan menghasilkan pergerakan yang baik. Maka dari itu sebagai

langkah awal, kerangka kendali-PI dapat diterapkan menggunakan agent nyata.

3.2 Permasalahan dan Solusi

Pada kerangka kendali-PI pada persamaan (2.25), state yang digunakan membutuhkan koordinat relatif dari tetangganya. Akan tetapi pada batasan penelitian ini, sensor yang digunakan hanya memberikan jarak terhadap tetangganya. Secara pendekatan, digunakan koordinat polar dan diubah ke koordinat kartesian. Akan tetapi koordinat polar membutuhkan sudut antara agent dan tetangganya. Oleh karena itu dibutuhkan algoritma khusus untuk menutup permasalahan tersebut. Untuk mengembangkan algoritma tersebut, dapat menggunakan hukum *cosinus* segitiga untuk menentukan sudutnya. Algoritma *cosinus* tersebut hanya berlaku apabila tetangga tidak melakukan pergerakan dan akan dijalankan algoritma tersebut ketika inisialisasi. Ketika tetangga melakukan pergerakan, tetangga mengirimkan informasi percepatan koordinatnya pada agent. Kegunaannya adalah sebagai referensi perubahan koordinat terhadap tetangga. Sehingga harapannya adalah kerangka kendali-PI dapat digunakan menggunakan sensor yang hanya mendeteksi jarak saja.

BAB 4

METODE PENELITIAN

4.1 Strategi Kendali Multi Robot

Analisa akan dilakukan dalam beberapa bagian agar mudah dipahami dan diterapkan. Analisa tersebut adalah mengenai kendali dari model dinamika robot dan kendali formasi, dan mengenai metode percobaan akan dibahas secara matematis, simulasi, dan HIL.

4.1.1 Kendali Robot

Pada kendali robot akan dibahas mengenai analisis kendali robot menggunakan *state-space feedback*. Kendali robot ini adalah kendali tahap akhir dari kendali keseluruhan. Dapat diperhatikan pada persamaan (2.25), sebagai kendali tahap awal, bahwa *state* yang digunakan adalah koordinat. Maka koordinat tersebut akan menjadi *set point* bagi robot. Variable yang dikendalikan pada kendali robot adalah koordinat robot dari kondisi inisial. Koordinat disini adalah koordinat *state* pada persamaan (2.12). Pada sub bab ini akan didefinisi mengenai kriteria pencapaian *set point* dan membahas parameter kendali *state-space feedback* agar mencapai kriteria yang diinginkan.

4.1.1.1 State Feedback

@todo
Gambar grafik state space feedback

Pada persamaan (2.12) diketahui bahwa *state* memiliki dimensi 6×1 . Dimensi tersebut tidak menunjukkan sistem memiliki orde 6. Apabila diperhatikan orde dari sistem adalah orde 2. Dengan membaginya kedalam 3 persamaan *state-space* akan lebih mudah dalam analisis parameter kendalinya. Berikut adalah persamaan

state-space

$$\begin{bmatrix} \dot{x}_p \\ \ddot{x}_r \end{bmatrix} = \begin{bmatrix} 0 & A_{14} \\ 0 & A_{44} \end{bmatrix} \begin{bmatrix} x_p \\ \dot{x}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ B_{11} & B_{12} & B_{13} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + K_{44} \text{sgn}(\dot{x}_r) \quad (4.1)$$

$$\begin{bmatrix} \dot{y}_p \\ \ddot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & A_{25} \\ 0 & A_{55} \end{bmatrix} \begin{bmatrix} y_p \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ B_{21} & B_{22} & B_{23} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + K_{55} \text{sgn}(\dot{y}_r) \quad (4.2)$$

$$\begin{bmatrix} \dot{\theta}_p \\ \ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & A_{35} \\ 0 & A_{66} \end{bmatrix} \begin{bmatrix} y_p \\ \dot{y}_r \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ B_{31} & B_{32} & B_{33} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + K_{66} \text{sgn}(\dot{\theta}_r) \quad (4.3)$$

State feedback membutuhkan kembalian nilai *state* dari sistem dan mengkalikanya dengan besaran tertentu agar nilai karakteristik sistem tetap dalam keadaan stabil atau sesuai ketentuan. Secara umum, *state* tidak dapat diperoleh langsung dari sistem. Kemampuan untuk memperoleh *state* dari sistem langsung disebut dengan kemampuan *Observability*. Apabila sebuah sistem tidak *Observable*, maka dalam kendalinya dibutuhkan *Observer*. Dimana tugasnya adalah mengestimasi *state* pada sistem dengan membandingkan keluaran dan masukan. Syarat untuk dapat diterapkan *state feedback*, sistem harus *observable* dan *controlable*. Berikut adalah rumus untuk menguji apakah sistem bersifat *controlable* atau tidak (Dorf, dkk (2010)).

$$P_c = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}$$

$$\text{rank}(P_c) = n \quad (4.4)$$

Apabila hasil dari $\text{rank}(P_c) \neq n$ maka sistem tidak *controlable*. Sedangkan untuk menguji *observable* dapat menggunakan rumus berikut.

$$P_o = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

$$\text{rank}(P_o) = n \quad (4.5)$$

Apabila sistem *observable* determinan dari matriks *Observability* P_o tidak nol.

Menggunakan parameter robot oleh Correia, dkk (2012) yang diterapkan pada persamaan (2.12)-(2.11), hasil pengujian *controlable rank*(P_c) = 6, maka dapat disimpulkan sistem robot *controlable*. Hasil pengujian *observable rank*(P_o) = 6, maka sistem robot juga *observable*. Karena sistem robot *observable*, maka dalam desain kendali tidak diperlukan *observer*.

4.1.1.2 Desain Kendali

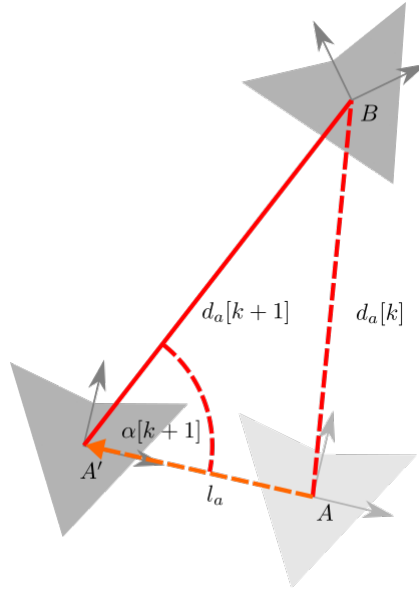
@todo
Kriteria
@todo
Parameter K
@todo
Parameter N

4.1.2 Kendali Formasi Multi Robot

Pada sub bab 2.2.2 dijabarkan bagaimana kendali formasi menggunakan kendali-PI dan menghasilkan persamaan (2.25). Persamaan tersebut adalah persamaan *state-space* kendali formasi. Apabila diperhatikan *state* yang digunakan adalah koordinat relatif dari robot. Akan tetapi dalam batasannya, robot hanya bisa mengetahui nilai jarak dari robot lain. Dengan kata lain, yang dibutuhkan dalam metode kendali formasi adalah jarak dalam bentuk koordinat, $x \in \mathbb{R}^2$. Sedangkan dalam kenyataannya yang diketahui adalah jarak, $r \in \mathbb{R}$. Apabila hanya variable jarak tersebut sebagai acuan kendali, maka robot tidak mengerti kearah mana harusnya robot itu bergerak untuk meminimalisasi error jaraknya.

4.1.2.1 Strategi Penentuan Koordinat Tetangga

Penentuan koordinat tetangga dapat ditemukan dengan mengubah koordinat polar menjadi koordinat kartesian. Koordinat polar membutuhkan panjang, d_a , dan sudut, α . Variable d_a dapat diperoleh dari sensor, akan tetapi sudut α tidak bisa dideteksi secara langsung oleh sensor. Dengan menggunakan *cosinus* pada segitiga dimungkinkan untuk mendapatkan sudut tersebut.



Gambar 4.1: Strategi Penentuan Koordinat

Dapat diperhatikan pada gambar 4.1 untuk gambaran strateginya. Robot $B \in \mathcal{N}_A$, adalah tetangga dari robot A. Pertama-tama, sebelum robot A bergerak, disimpan terlebih dahulu nilai d_a , atau dinotasikan dengan $d_a[k]$ sebagai jarak sebelum bergerak. Lalu robot A berjalan secara random kesegala arah dengan jarak l_a . Di-simpan kembali nilai jara d_a , atau dinotasikan dengan $d_a[k+1]$. Setelah itu dapat ditentukan sudut $\alpha[k+1]$

$$\alpha[k+1] = \cos^{-1} \left[\frac{l_a^2 + d[k+1]^2 - d_a[k]^2}{2d_a[k+1]l_a} \right]. \quad (4.6)$$

Sebelum $\alpha[k+1]$ digunakan, jarak $d_a[k+1]$ dan $d_a[k]$ berpengaruh dalam penentuan koordinat. Sehingga diperlukan sedikit algoritma

$$\alpha_i = \begin{cases} \alpha[k+1] & , d_a[k+1] > d_a[k] \\ 180 - \alpha[k+1] & , d_a[k+1] < d_a[k] \end{cases}. \quad (4.7)$$

Strategi pada gambar 4.1 hanya berlaku apabila target ukur berhenti. Apabila dinotasikan koordinat x_B^A adalah koordinat relatif robot B terhadap A, maka \dot{x}_B^A adalah notasi kecepatan koordinat dari robot B. Dengan menggunakan persamaan (2.1) untuk menyelesaikan koordinat dalam keadaan robot B bergerak, yaitu mengirimkan informasi kecepatan koordinatnya ke robot A. Lalu robot A dapat

mengkalkulasi koordinat relatif dengan persamaan berikut

$$\alpha[k+1] = \alpha[k] + \tan^{-1} \left[\frac{\dot{x}_B^A}{\dot{y}_B^A} \right] \quad (4.8)$$

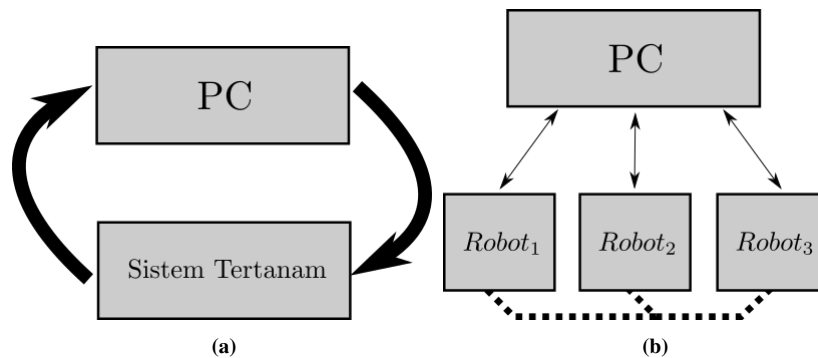
dimana kondisi inisial adalah $\alpha[k] = \alpha_i$ diperoleh dari hasil strategi pada persamaan (4.7). Dengan memanfaatkan kedua strategi tersebut dapat digunakan untuk mengkalkulasi koordinat robot B relatif terhadap robot A

$$x_B^A = \begin{bmatrix} x_B = d_a[k] \cdot \cos \alpha[k] \\ y_B = d_a[k] \cdot \sin \alpha[k] \end{bmatrix} \quad (4.9)$$

Dalam strategi ini akan terjadi ketidak akuratan dalam pengukuran apabila target ukur berada pada sudut 90° . Akan tetapi, Cao, dkk (2007) sudah menjelaskan mengenai kriteria posisi agent ketika dalam kondisi inisial. Yaitu semua agent tidak berada pada kondisi sejajar secara koordinat global pada kondisi inisial.

4.2 Kestabilan Perangkat Percobaan

Sub bab ini akan dibahas mengenai perangkat penunjang sebagai pembantu dalam menyelesaikan penelitian. Sebagai langkah awal pengembangan, metode yang digunakan adalah *Hardware-In Loop*.



Gambar 4.2: (a) Grafik Hardware-in-the-loop (Ledin (1999)). (b) HIL Kendali Multi-Robot.

Hardware-in-the-loop (HIL) adalah metode untuk pengembangan perangkat kendali dengan memanfaatkan model sebagai objek kendalinya. Seperti pada gambar 4.2a, bahwa HIL terdiri dari dua perangkat, yaitu perangkat untuk menjalankan objek kendali atau dapat disebut sebagai model/plant dan perangkat sistem kontrolnya, dalam kasus ini sistem kontrol menggunakan sistem tertanam (*embedded system*). Metode HIL, banyak digunakan oleh peneliti dalam proses pengembangan dengan pertimbangan efisiensi terhadap berbagai hal. Seperti yang digunakan oleh Ir-

wanto (2018), mengembangkan kendali UAV menggunakan HIL; dan Quesada, dkk (2019), mengembangkan prangkat pankreas buatan yang digunakan untuk mengendalikan kadar gula pada pengidap diabetes.

Pada penelitian ini akan digunakan *microcontroller*(MCU) STM32F466 sebagai prangkat kendalinya. MCU tersebut ber-arsitektur ARM Cortex-M4 dengan clock 180MHz, menampung ukuran program sampai 256K didalam memori Flash, serta fitur komunikasi standart MCU dengan lengkap. *Platform Library* yang digunakan dalam pembuatan aplikasi didalamnya adalah *Mbed*, yang menyediakan berbagai banyak fungsi yang lengkap dan mudah untuk berinteraksi dengan fitur-fitur MCU. *Mbed* juga menyediakan fungsi untuk mengaplikasikan RTOS (Real-time Operating System) dengan mudah dan terdokumentasi secara jelas didalam lamannya. Pada prangkat PC akan dikembangkan program berbasis *Python* yang akan menjalankan simulasi model dan berkomunikasi dengan MCU secara *real-time*. Program *Python* akan menjalankan model pada persamaan (2.11)-(2.12) dengan metode yang dijabarkan pada sub bab 2.3. Dapat diperhatikan pada gambar 4.2b, pada HIL untuk kendali multi robot akan menggunakan tiga kendali untuk mempresentasikan tiga robot. Setiap prangkat pengendali akan saling terhubung satu sama lain dan semua prangkat pengendali terhubung dengan prangkat PC. Komunikasi antar prangkat pengendali akan digunakan untuk pertukaran informasi. Sedangkan komunikasi dengan PC akan mempresentasikan aktuator dan sensor untuk setiap prangkat kendali. PC akan merekam setiap keluaran dari model dan masukan dari setiap prangkat kendali sebagai tampilan pergerakan robotnya.

4.2.1 Kestabilan Model

Pada persamaan (2.30) apabila model dikalkulasi akan bergantung dengan besarnya *step size*, h . Oleh karena itu, setelah persamaan (2.11)-(2.12) dilakukan parameterisasi harus dilakukan penentuan *step size* agar model tersebut stabil dalam mensimulasikan modelnya. Penentuan *step size* harus berdasarkan kriteria kestabilan pada gamabar 2.2.

Apabila didefinisi ulang *state* pada persamaan (2.11)-(2.12) dengan $x(t) = \begin{bmatrix} \dot{x}_r & \dot{y}_r & \dot{\theta}_r \end{bmatrix}^T$, maka akan lebih mudah untuk menghitung kestabilan dari matriks $A \in \mathbb{R}^{3 \times 3}$. Dengan menggunakan parameter dari penelitian oleh Correia, dkk

(2012), maka akan diperoleh matriks A, B, K , dan C .

$$A = \begin{bmatrix} -6.69666 & 0.00000 & 0.00000 \\ 0.00000 & -6.71000 & 0.00000 \\ 0.00000 & 0.00000 & -4.04200 \end{bmatrix}; \quad B = \begin{bmatrix} 0.00000 & 0.57735 & -0.57735 \\ -0.66667 & 0.33333 & 0.33333 \\ 4.00000 & 4.00000 & 4.00000 \end{bmatrix};$$

$$K = \begin{bmatrix} -1.46667 & 0.00000 & 0.00000 \\ 0.00000 & -1.00000 & 0.00000 \\ 0.00000 & 0.00000 & -0.06600 \end{bmatrix}; \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Dengan menggunakan pendekatan pada persamaan (2.29) untuk persamaan (2.11) maka diperoleh bentuk diskretnya

$$x[k+1] = (I + A.h).x[k] + B.h.u[k] + K.h.sgn(x[k]). \quad (4.10)$$

$$(4.11)$$

Pengali $sgn(\cdot)$ bersifat penambah dari sistem, maka dalam penentuan kestabilan ini akan dianggap penambah dari matriks sistem.

$$x[k+1] = (I + (A + K).h).x[k] + B.h.u[k]. \quad (4.12)$$

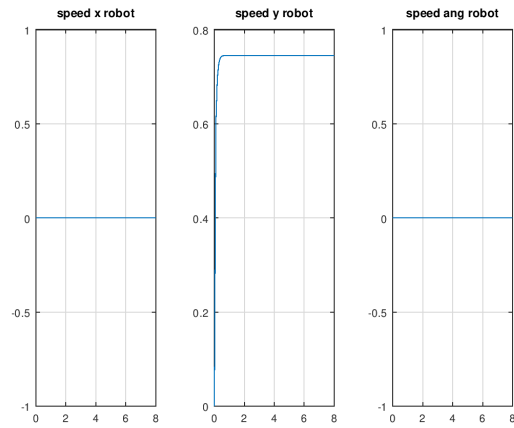
$$(4.13)$$

Kriteria kestabilan akan bergantung dari hasil penentuan h pada $I + (A + K)h = \Lambda$. Untuk semua nilai λ pada matriks Λ harus memenuhi kriteria $\lambda \leq 1$. Dimungkinkan akan mengalami kebingungan ketika menentukan besar h , akan tetapi nantinya persamaan ini akan diterapkan dan diselesaikan oleh komputer. Alangkah baiknya apabila diidentifikasi terlebih dahulu konsumsi waktu yang dibutuhkan untuk menyelesaikan satu iterasi dari persamaan tersebut. Setelah dilakukan identifikasi, waktu yang dibutuhkan untuk satu kali iterasi berkisar 0.001 ms (Pembulatan). Sehingga penentuan *step size* sebesar 0.1 ms sangat dimungkinkan, dengan pertimbangan sisa dari waktu yang digunakan kalkulasi dapat digunakan untuk waktu *idle* dan menjalankan program yang lain. Berikut adalah matriks Λ setelah dikalkulasi menggunakan $h = 0.1$

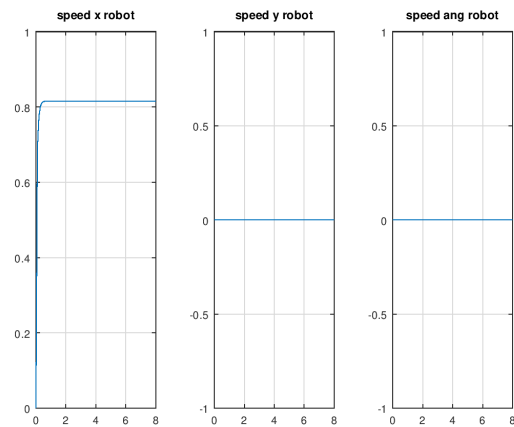
$$\Lambda = \begin{bmatrix} 0.18367 & 0.00000 & 0.00000 \\ 0.00000 & 0.22900 & 0.00000 \\ 0.00000 & 0.00000 & 0.58920 \end{bmatrix}.$$

Terbukti bahwa semua nilai item didalam matriks kurang dari sama dengan satu. Sehingga menggunakan algoritma *Expilic Euler* sudah cukup untuk menjalankan

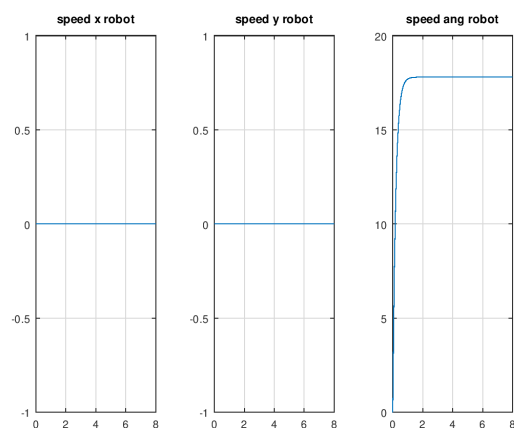
model robot *omni 3-wheel* sebagai model *holonomic* yang akan digunakan untuk percobaan kendali multi robot. Hasil plot dari simulasi model dapat dilihat pada gambar 4.3.



(a)



(b)



(c)

Gambar 4.3: (a) $w_1 = -6; w_2 = 3; w_3 = 3$. (b) $w_1 = 0; w_2 = 6; w_3 = -6$ (c) $w_1 = 6; w_2 = 6; w_3 = 6$

4.2.2 Rencana Hardware-in-Loop

@todo

kutip hasil HIL yang sudah ada lalu gabungkan model dan kendali jadi satu secara sederhana

4.2.3 Rencana Uji Lapangan

@todo

Membahas mengenai cara pengambilan data penerapan pada robot aslinya

BAB 5

PERINTAH DALAM UITHESIS.STY

@todo

Tambahkan kata-kata pengantar bab 5 disini.

5.1 Mengubah Tampilan Teks

Beberapa perintah yang dapat digunakan untuk mengubah tampilan adalah:

- `\f`
Merupakan alias untuk perintah `\textit`, contoh *contoh hasil tulisan*.
- `\bi`
Contoh hasil tulisan.
- `\bo`
Contoh hasil tulisan.
- `\m`
Contohhasiltulisan.
- `\mc`

Contohhasiltulisan

- `\code`
Contoh hasil tulisan.

5.2 Memberikan Catatan

Ada dua perintah untuk memberikan catatan penulisan dalam dokumen yang Anda kerjakan, yaitu:

- `\todo`

Contoh:

@todo

Contoh bentuk todo.

- `\todoCite`

Contoh:

@todo

Referensi

5.3 Menambah Isi Daftar Isi

Terkadang ada kebutuhan untuk memasukan kata-kata tertentu kedalam Daftar Isi. Perintah `\addChapter` dapat digunakan untuk judul bab dalam Daftar isi. Contohnya dapat dilihat pada berkas `thesis.tex`.

5.4 Memasukan PDF

Untuk memasukan PDF dapat menggunakan perintah `\inpdf` yang menerima satu buah argumen. Argumen ini berisi nama berkas yang akan digabungkan dalam laporan. PDF yang dimasukan dengan cara ini akan memiliki header dan footer seperti pada halaman lainnya.

Untitled

Ini adalah berkas pdf yang dimasukkan dalam dokumen laporan.

Cara lain untuk memasukan PDF adalah dengan menggunakan perintah `\putpdf` dengan satu argumen yang berisi nama berkas pdf. Berbeda dengan perintah sebelumnya, PDF yang dimasukan dengan cara ini tidak akan memiliki footer atau header seperti pada halaman lainnya.

Untitled

Ini adalah berkas pdf yang dimasukkan dalam dokumen laporan.

5.5 Membuat Perintah Baru

Ada dua perintah yang dapat digunakan untuk membuat perintah baru, yaitu:

- `\Var`
Digunakan untuk membuat perintah baru, namun setiap kata yang diberikan akan diproses dahulu menjadi huruf kapital. Contoh jika perintahnya adalah `\Var{adalah}` maka ketika perintah `\Var` dipanggil, yang akan muncul adalah ADALAH.
- `\var`
Digunakan untuk membuat perintah atau baru.

BAB 6

??

@todo

tambahkan kata-kata pengantar bab 6 disini

BAB 7

KESIMPULAN DAN SARAN

@todo

Tambahkan kesimpulan dan saran terkait dengan pekerjaan yang dilakukan.

7.1 Kesimpulan

7.2 Saran

DAFTAR REFERENSI

- Cao, M., dkk (2007). “Controlling a triangular formation of mobile autonomous agents”. In: *2007 46th IEEE Conference on Decision and Control*, pp. 3603–3608. DOI: 10.1109/CDC.2007.4434757.
- Correia, Mariane Dourado, André Gustavo, dan Scolari Conceição (2012). “Modeling of a Three Wheeled Omnidirectional Robot Including Friction Models”. In: *IFAC Proceedings Volumes 45.22*. 10th IFAC Symposium on Robot Control, pp. 7 –12. ISSN: 1474-6670. DOI: <https://doi.org/10.3182/20120905-3-HR-2030.00002>. URL: <http://www.sciencedirect.com/science/article/pii/S1474667016335807>.
- Dorf, Richard, dan Robert Bishop (July 2010). *Modern Control Systems, 12th Edition*. ISBN: ISBN-10: 0136024580; ISBN-13: 978-0136024583.
- Fabien, Brian (2009). “Numerical Solution of ODEs and DAEs”. In: *Analytical System Dynamics: Modeling and Simulation*. Boston, MA: Springer US, pp. 1–59. ISBN: 978-0-387-85605-6. DOI: 10.1007/978-0-387-85605-6.5. URL: <https://doi.org/10.1007/978-0-387-85605-6.5>.
- Guanghua, Wang, dkk (Jan. 2013). “Study on Formation Control of Multi-Robot Systems”. In: pp. 1335–1339. ISBN: 978-1-4673-4893-5. DOI: 10.1109/ISDEA.2012.316.
- Hacene, Nacer, dan Boubekur Mendil (Apr. 2019). “Fuzzy Behavior-based Control of Three Wheeled Omnidirectional Mobile Robot”. In: *International Journal of Automation and Computing* 16.2, pp. 163–185. ISSN: 1751-8520. DOI: 10.1007/s11633-018-1135-x. URL: <https://doi.org/10.1007/s11633-018-1135-x>.
- Irwanto, Herma Yudhi (2018). *Development of Mobile Ground Control System and GPS Base Auto Tracking Antenna*.
- Ledin, Jim A. (1999). “Hardware-in-the-Loop Simulation”. In: *Embedded Systems Programming*.
- Oh, Kwang-Kyo, Myoung-Chul Park, dan Hyo-Sung Ahn (2015). “A survey of multi-agent formation control”. In: *Automatica* 53, pp. 424 –440. ISSN: 0005-1098. DOI: <https://doi.org/10.1016/j.automatica.2014.10.022>. URL: <http://www.sciencedirect.com/science/article/pii/S0005109814004038>.
- Parker, Lynne (Mar. 2003). “Current research in multirobot systems”. In: *Artificial Life and Robotics* 7, pp. 1–5. DOI: 10.1007/BF02480877.

- Quesada, Luisa Fernanda, dkk (2019). “Open-source low-cost Hardware-in-the-loop simulation platform for testing control strategies for artificial pancreas research”. In: *IFAC-PapersOnLine* 52.1. 12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems DYCOPS 2019, pp. 275 – 280. ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2019.06.074>. URL: <http://www.sciencedirect.com/science/article/pii/S2405896319301594>.
- Rozenheck, O., S. Zhao, dan D. Zelazo (2015). “A proportional-integral controller for distance-based formation tracking”. In: *2015 European Control Conference (ECC)*, pp. 1693–1698. DOI: 10.1109/ECC.2015.7330781.

LAMPIRAN

LAMPIRAN 1