

Implementasi Sistem Navigasi *Behavior Based* dan Kontroler PID pada Manuver Robot *Maze*

Fahmizal, Rusdhianto Effendi AK, Eka Iskandar

Jurusan Teknik Elektro, Fakultas Teknologi Industri – Institut Teknologi Sepuluh Nopember

Surabaya

Abstrak - Tugas akhir ini merancang dan mengimplementasikan algoritma kendali *behavior based* (berbasis perilaku) pada robot *maze* beroda tipe *differential steering* dan kontroler PID (Proporsional, Integral, Diferensial) pada manuver robot ini. Tugas robot ini adalah menyusuri dan mencari target yang diletakan secara acak (*random*) pada labirin, setelah menemui target robot kembali pulang ke *home* melalui jalur terdekat dari target tersebut. Kendali berbasis perilaku menggunakan modul-modul perilaku yang bekerja secara bersamaan membentuk perilaku robot yang ingin dicapai. Masing-masing perilaku sifatnya independen, memiliki hubungan langsung dengan sensor dan aktuator. Masing-masing perilaku juga dapat mengirim pesan pada perilaku lain atau menghambat *output* perilaku lainnya. Selain menerapkan kendali berbasis perilaku tugas akhir ini juga merancang dan mengimplementasikan kontroler PID pada manuver robot *maze* terhadap jarak dinding labirin. Kontroler PID bertujuan untuk memuluskan pergerakan robot saat menelusur ruangan/lorong labirin. Dengan bantuan kontroler PID robot *maze* mampu bermanuver dengan aman, halus, responsif dan cepat. Hasil parameter kontroler PID yang dicapai dari penelitian tugas akhir ini diperoleh dari hasil *tuning* eksperimen dengan hasil $K_p=17$, $K_i=1$ dan $K_d=50$ untuk hasil yang sangat memuaskan.

Kata Kunci : *behavior based*, kontroler PID, robot *maze*.

I. PENDAHULUAN

Perkembangan dunia robotika memiliki unsur yang sedikit berbeda dengan ilmu-ilmu dasar atau terapan lainnya. Ilmu dasar biasanya berkembang dari suatu asas atau hipotesa yang kemudian diteliti secara metodis, sedangkan ilmu robotika lebih sering berkembang melalui pendekatan praktis. Kemudian melalui pendekatan atau asumsi dari hasil pengamatan perilaku makhluk hidup atau peralatan bergerak lainnya dikembangkanlah penelitian secara teoritis. Teknologi robot berkembang semakin pesat sejak ditemukannya piranti semikonduktor. Piranti semikonduktor ini memungkinkan penciptaan robot dalam ukuran yang lebih kecil, kecepatan kerja yang lebih tinggi dan akurasi yang lebih handal. Robot *maze* ialah robot yang berjalan mencari target pada suatu labirin. Saat ini, perkembangan robot *maze* sedang gencar-gencarnya dijadikan objek teliliti dan kontrol, banyak event internasional yang telah diselenggarakan. Hal inilah yang

mendasari tugas akhir ini bertemakan robot *maze* dan kendalinya.

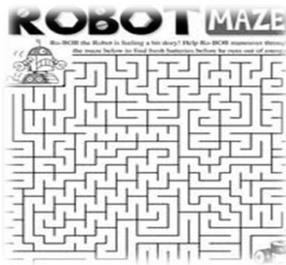
Dalam banyak aplikasi robot, sering kali dibutuhkan reaksi yang cepat dari robot. *Behavior Based Robotic* (BBR) merupakan suatu konsep sistem kendali (*knowledge based*) berdasarkan tingkah laku/perilaku robot terhadap adaptasi lingkungannya. Karena sistem navigasi robot *maze* secara otonom, maka robot harus memiliki beberapa kemampuan (*behavior*) untuk melaksanakan tugas (*task*) yang diberikan secara terhadap lingkungan. Arsitektur *behavior based robotic* merupakan suatu sistem kendali yang tidak berbasiskan model, karena memiliki struktur *behavior* yang bekerja bersama secara paralel. Pada pendekatan ini, sistem diuraikan menjadi beberapa modul yang masing-masingnya bertanggung jawab untuk melakukan satu perilaku (*behavior*). Tiap *behavior* mengandung jalur lengkap mulai dari sensing sampai aksi. Semua modul yang mewakili satu *behavior* bekerja bersama-sama [2].

Dalam *behavior based robotic*, sistem kendali robot dipisahkan berdasarkan tugas yang ingin dicapai yang disebut dengan *behavior*. Semakin banyak tugas sistemnya semakin kompleks, sehingga dapat menimbulkan konflik antar *behavior*. Oleh karena itu, dikembangkan metode koordinasi antar *behavior*. Terdapat dua pendekatan mekanisme koordinasi, yaitu *competitive/arbitrator* dan *cooperative/command fusion* [3]. Pada metode *competitive*, hanya satu *behavior* yang diijinkan memberikan sinyal kendali. Sedangkan koordinasi *cooperatif* menggabungkan semua keluaran *behavior* yang ada.

Dalam penelitian tugas akhir ini akan dikembangkan suatu metode koordinasi *competitive coordinator* untuk mengkoordinasikan stuktur kendali *behavior based* pada robot *maze*. Untuk memuluskan pergerakan robot saat menelusur ruangan/lorong labirin diperlukan kontroler yang tepat agar pergerakan robot saat dioperasikan dapat sesuai dengan harapan. Akan tetapi dari pengendalian gerak robot *maze* tersebut pasti memiliki kendala terbesar yakni kestabilan robot dalam mencermati jarak yang dibaca, maka dengan bantuan kontroler PID (*Proportional, Integral, Diferensial*) dapat menjadi solusinya. Dengan kontroler PID diharapkan robot mampu bermanuver dengan baik, halus, responsif dan cepat dengan kecermatan tinggi sehingga pergerakan robot lebih stabil sesuai dengan medan dinamis yang dihadapi.

II. SEKILAS TENTANG ROBOT MAZE

Maze adalah suatu jaringan jalan yang rumit. Pada bidang robotika ada dua jenis *maze* yang umum digunakan, yaitu *wall maze* dan *line maze*. *Wall maze* pada umumnya dikenal dengan istilah labirin, yakni suatu jaringan jalan yang terbentuk atas lorong-lorong dengan dinding tanpa atap. Permasalahan yang timbul pada *maze* adalah cara untuk mendapatkan jalur terpendek, sehingga dibutuhkan metode untuk menyelesaikannya. *Maze mapping* merupakan algoritma yang digunakan untuk memecahkan *maze*, yakni mencari dan menggambarkan peta dari *maze* [5]. *Maze mapping* pada umumnya di berbagai sumber dijelaskan dengan istilah *path mapping* yang konsep dasar dalam pencariannya mengikuti aturan *left/right hand rule*. *Path Mapping* adalah mode *maze mapping* yang digunakan pada robot *wall follower*, biasanya pada event internasional dikenal dengan nama *micro-mouse* [4]. Algoritma ini merupakan *the basic algorithm*. Di dalamnya ber-opsi-kan untuk berjalan mengikuti dinding kiri atau dinding kanan pada proses memetakan *maze*. Selanjutnya, bila peta yang sudah dibuat tersebut dijalankan, maka robot bisa kembali ke posisi *start* melalui jalur terpendeknya atau mampu juga mengulangi kembali melewati jalur terpendek dari *start* menuju target (*finish*).

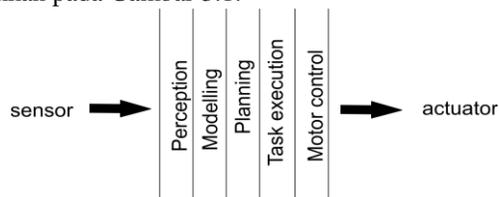


Gambar 2.1 Ilustrasi robot *maze*[4]

III. TEORI PENUNJANG

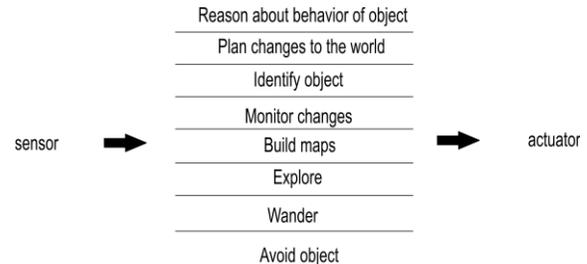
A. Behavior Based Robotic

Pendekatan yang biasa digunakan untuk membangun sistem kendali robot adalah dengan menguraikan setiap masalah kedalam rangkaian unit fungsional sebagaimana ditunjukkan pada Gambar 3.1.



Gambar 3.1 Teknik penguraian tradisional untuk sistem kendali *mobile robot* ke dalam unit-unit fungsional [1]

Berbeda dengan pendekatan di atas, *behavior based robotic* mendesain sistem kendali robot menggunakan pendekatan *task achieving behaviors* (perilaku dalam menunaikan tugas) sebagaimana ditunjukkan oleh Gambar 3.2. Tiap tugas disebut dengan *behavior*.



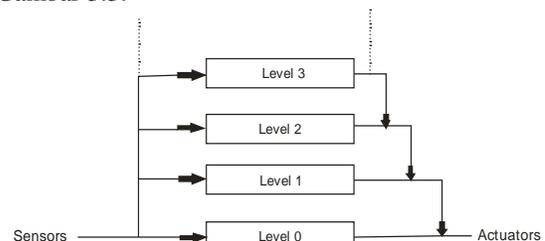
Gambar 3.2 Dekomposisi sistem kendali *mobile robot* dengan *task achieving behaviors* [1]

Metode dekomposisi ini memiliki arsitektur *mobile robot* yang sangat berbeda dengan dekomposisi yang berdasarkan unit fungsional (Gambar 3.2). Berbeda secara *hardware*, dan sejumlah kelebihan lain seperti *robustness*, *buildability* dan *testability*.

Arsitektur Subsumption

Arsitektur *subsumption* adalah struktur BBR yang diusulkan oleh Rodney Brooks [1]. Dalam membangun robotnya, Rodney Brooks menguraikan permasalahan sistem kendali robot sesuai dengan manifestasi luar yang diinginkan oleh sistem kendali robot, tidak berdasarkan pada operasi internal dari sistem kendali robot sebagaimana yang telah dilakukan oleh beberapa peneliti sebelumnya. Oleh karena itu, Brooks mendefinisikan sejumlah level kompetensi pada *mobile robot* mandiri. Level kompetensi adalah spesifikasi informal dari sekelompok perilaku yang diinginkan robot bekerja pada semua lingkungan yang akan dihadapi. Level kompeten yang lebih tinggi menunjukkan kelompok perilaku yang lebih khusus/spesifik.

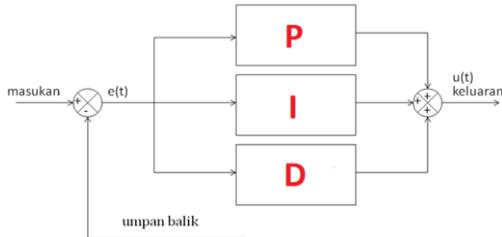
Tiap level kompetensi memasukkan sub kelompok dari level kompetensi sebelumnya. Karena level kompetensi mendefinisikan kelompok perilaku yang valid, dapat dianggap bahwa level yang lebih tinggi memberikan tambahan batasan pada kelompok perilaku tersebut. Rodney Brooks memulai dengan membangun sistem kendali robot yang melaksanakan level kompetensi nol. Perbaikan kesalahan dilakukan dengan teliti. Brooks tidak pernah mengubah sistem ini dan menyebutnya sistem kendali level ke nol. Selanjutnya, dibangun lapisan kendali yang lain yang disebut sistem kendali level kesatu. Level ini dapat menguji data dari level nol dan juga memberikan data ke dalam *internal interface* level nol, serta menekan data normal yang mengalir. Lapisan ini, dengan tambahan dari lapisan nol melaksanakan level kompetensi pertama. Lapisan ke nol melanjutkan untuk bekerja tanpa mengetahui lapisan di atasnya yang terkadang mengganggu aliran data. Proses yang sama diulangi untuk mendapatkan level kompetensi yang lebih tinggi sebagaimana ditunjukkan oleh Gambar 3.3.



Gambar 3.3 Arsitektur *subsumption*[1]

B. Kontroler PID

Setiap kekurangan dan kelebihan dari masing-masing kontroler Proporsional, Integral dan Diferensial dapat saling menutupi dengan menggabungkan ketiganya menjadi kontroler proporsional plus integral plus diferensial (kontroler PID). Elemen-elemen kontroler P, I dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem, menghilangkan offset dan menghasilkan perubahan awal besar. Gambar 3.4 menunjukkan diagram blok kontroler PID.



Gambar 3.4 Diagram blok kontroler PID

Kontroler PID digital merupakan bentuk lain dari kontroler PID yang diprogram dan dijalankan menggunakan komputer atau mikrokontroler. Untuk dapat mengimplementasikan PID digital di komputer atau mikrokontroler, maka kontroler PID analog harus diubah terlebih dahulu ke bentuk digital. Penurunan kontroler PID digital dapat dilihat pada Persamaan 3.1 sampai dengan Persamaan 3.4.

Bentuk persamaan matematis dari kontroler PID adalah sebagai berikut.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (3.1)$$

Dengan $K_i = \frac{1}{\tau_i}$ dan $K_d = \tau_d$, bentuk integral dan

diferensial dapat ditulis dalam bentuk diskrit seperti pada Persamaan 3.2 dan Persamaan 3.3.

$$\int_0^t e(t) dt \approx T \sum_0^k e(k) \quad (3.2)$$

$$\frac{de(t)}{dt} \approx \frac{e_k - e_{k-1}}{T} \quad (3.2)$$

Sehingga diperoleh dalam bentuk kontroler PID diskrit ialah sebagai berikut:

$$u_{(k)} = K_p e_k + K_i T \sum_0^k e_k + \frac{1}{T} K_d (e_k - e_{k-1}) \quad (3.4)$$

$$u = K_p \times error + K_i \times (error + last_error) \times Ts + \frac{K_d}{Ts} \times (error - last_error)$$

Dimana:

- K_p ialah konstanta proporsional
- K_i ialah konstanta integral
- K_d ialah konstanta diferensial
- $error$ ialah nilai kesalahan
- $last_error$ ialah nilai kesalahan sebelumnya
- T_s ialah *sampling time* (waktu cuplik)

IV. PERANCANGAN SISTEM

A. Spesifikasi Sistem

Definisi sistem pada tugas akhir ini adalah robot *maze* beserta algoritma dan kontroler pergerakannya. Yang dikerjakan dalam tugas akhir ini adalah perancangan serta implementasi algoritma *behavior based* dan kontroler PID pada manuver robot tersebut. Spesifikasi yang diharapkan dari sistem yang akan dirancang adalah sebagai berikut.

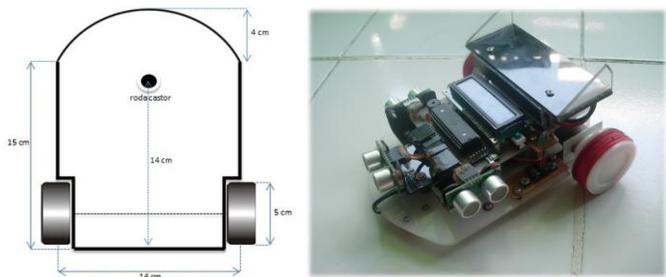
- Robot beroda menggunakan metoda *differential steering* mampu melakukan pergerakan manuver yang halus dan aman pada labirin.
- Robot dilengkapi sensor yang menunjang algoritma telusur kiri atau kanan pada labirin.
- Robot mampu menelusur sisi kiri atau kanan dari dinding labirin.
- Robot memiliki kemampuan untuk menyelesaikan *maze*.

B. Perancangan dan Implementasi Perangkat Keras

Perancangan perangkat keras pada tugas akhir ini terdiri dari dua bagian yaitu rancang bangun mekanik robot dan desain sistem elektronik.

1. Desain Mekanik

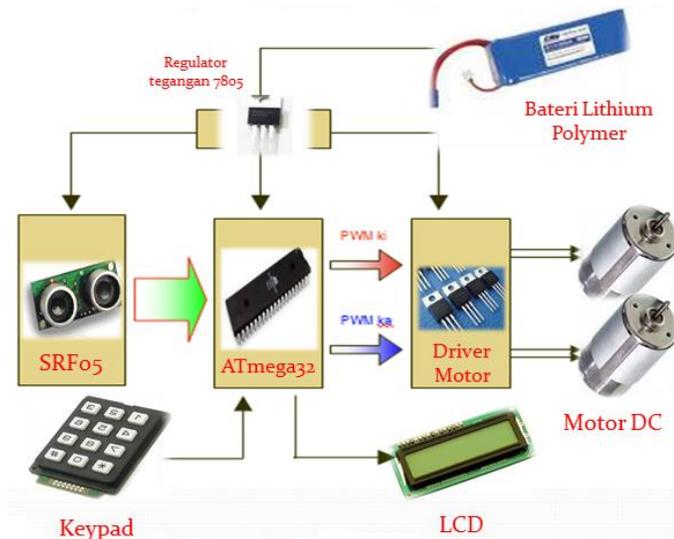
Sistem mekanik yang baik, mendukung pergerakan robot menjadi lebih baik, oleh karena itu perancangan mekanik dalam hal ini bodi dan rangka robot haruslah proporsional dengan panjang dan lebar serta tinggi dari robot. Material yang digunakan pada pembuatan rangka bodi robot *maze* ini menggunakan bahan *acrylic* dengan ketebalan 3 mm, bentuk dan dimensi robot dirancang sesuai secara proporsional dengan harapan robot dapat bermanuver dengan baik salah satunya adalah gerakan pivot pada satu titik.



Gambar 4.1 Rancangan desain mekanik robot *maze* dan hasil implementasi rancangan mekanik robot *maze*

2. Desain Sistem Elektronik

Dengan mengacu kepada diagram blok sistem elektronik, terdiri dari bagian masukan, bagian kendali, bagian keluaran dan bagian catu daya (baterai). Pada bagian masukan berupa tiga buah sensor ultrasonik (SRF05), satu buah sensor pendeteksi target *maze* dan tombol *keypad* yang berfungsi untuk memilih dan mengatur settingan pada robot. Pada bagian kendali menggunakan mikrokontroler ATmega32. Pada bagian keluaran berupa penampil LCD 2*16 dan *driver* motor sebagai penggerak aktuator robot.

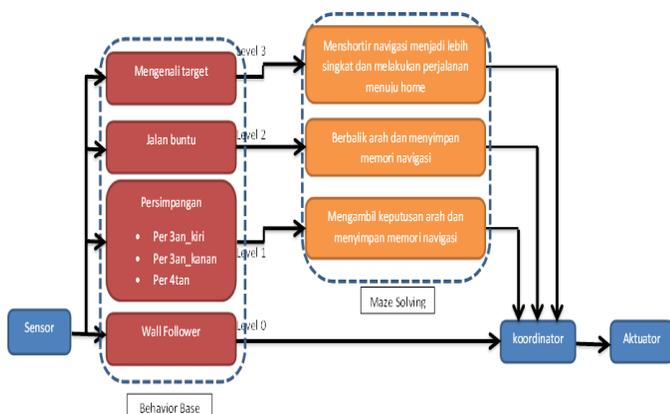


Gambar 4.2 Diagram blok sistem elektronik

C. Desain Sistem Navigasi Berbasis Perilaku

Aksi-aksi yang diperlukan untuk mengatur navigasi robot *maze* di dalam labirin adalah:

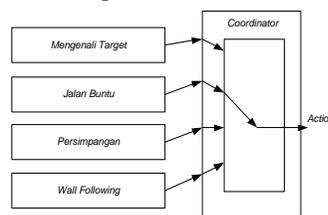
- navigasi aman (tanpa merusak dinding, tanpa menyentuh dinding labirin).
- mengikuti dinding (*wall following*).
- mengenali persimpangan.
- mengambil keputusan arah yang dituju ketika berada di persimpangan.
- mengambil arah persimpangan sesuai dengan mode telusur kiri atau kanan.
- mengenali jalan buntu.
- mengenali target.



Gambar 4.5 Rancangan *behavior based* pada robot *maze*

Untuk menghubungkan keempat *behavior* (layer-0 hingga layer-3), ditambahkan satu bagian koordinator *arbitrase*. Fungsinya adalah menentukan siapa yang akan memegang kendali aktuator serta menentukan gerakan robot. *Arbitrase* bertugas mengatur kapan perilaku persimpangan mengambil alih kendali dari perilaku *wall follower* yang ada di level sebelumnya. Koordinator *arbitrase* ini menggunakan *competitive coordinator*, pengambilan keputusan pada masing-masing *behavior* dapat diatur melalui level kompetensinya,

serta antar sesama *behavior* tidak dapat saling mempengaruhi. Hasil rancangan koordinator kompetitif terhadap beberapa *behavior* dapat dilihat pada Gambar 4.4.



Gambar 4.4 Rancangan koordinator *arbitrase competitive coordinator*

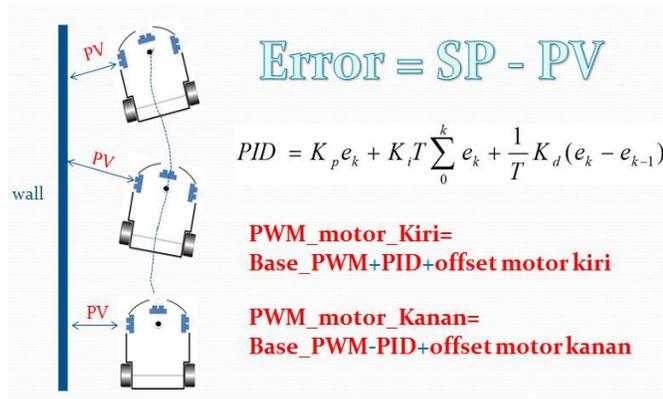
Navigasi aman mengacu kepada kemampuan robot untuk dapat bergerak di dalam labirin secara aman. Aman di sini diartikan sebagai tidak terbentur dinding, serta tidak menyentuh dinding. Ini adalah syarat dasar bagi perilaku-perilaku lainnya karena berkaitan langsung dengan keselamatan robot itu sendiri. Apapun perilaku lain yang dimiliki, keselamatan senantiasa menjadi syarat awal yang mutlak dimiliki oleh robot. Kemudian, agar robot senantiasa dapat memperbaiki posisinya dalam labirin, kemampuan mengikuti dinding amat diperlukan. Di dalam labirin yang hanya ada dinding, panduan terbaik bagi gerakan robot adalah dinding itu sendiri karena dinding sifatnya tidak berubah, posisinya tetap, serta banyak tersedia.

Setelah robot dapat bergerak dengan aman dan memperbaiki posisinya, yang selanjutnya diperlukan oleh robot agar dapat menemukan alur rekaman menuju target adalah kemampuan mengenali persimpangan. Persimpangan adalah saat dimana sebuah jalur terbagi menjadi dua atau tiga. Ketidakmampuan mengenali persimpangan akan membuat usaha robot mencapai target menjadi amat sulit. Jadi tidak hanya mengenali persimpangan saja, robot juga harus mampu mengambil keputusan arah yang akan diambil serta menyimpan memori navigasi kedalam kode-kode unik. Jalur terpendek diperoleh dari kode-kode unik yang telah dikonversi. Algoritma pemecahan *maze* ini diberi nama dengan algoritma *maze mapping*, algoritma ini menggunakan metode telusur kiri atau kanan saja.

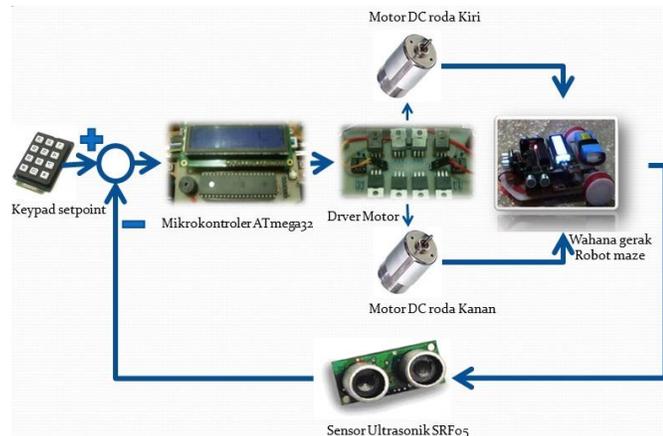
D. Pemrograman Kontroler PID pada Robot *Maze*

Pembuatan program kontroler PID ini dilakukan berdasarkan persamaan kontroler PID digital. Pertama-tama akan dibuat dua buah variabel berupa *error*, *last_error*. Gunanya adalah untuk menyimpan data *error* dan *last_error* yang akan digunakan pada perhitungan aksi kontroler PID. Setiap satu kali *looping* program, *error* akan diperbaharui dengan data yang diambil dari sensor, dan sebelumnya akan disimpan di *last_error*. Keluaran dari perhitungan program kontroler PID ini adalah nilai PWM. Nilai PWM ini dapat bernilai positif ataupun negatif. Positif dan negatif pada nilai PWM ini menandakan arah putaran motor. Keluaran kontroler berupa nilai PWM ini akan memanggil fungsi pengendali *driver motor*, pada fungsi ini apabila PWM bernilai positif, maka motor akan berputar maju, sebaliknya bila PWM bernilai negatif, maka motor akan berputar mundur. Nilai PWM keluaran dari kontroler PID ini akan ditambahkan dan dikurangkan terlebih dahulu dengan *base PWM* untuk masing-

masing motor dan dijumlahkan dengan *offset* PWM pada masing-masing motor tersebut. Ilustrasi aplikasi kontroler PID dapat dijelaskan pada diagram blok berikut ini.



Gambar 4.6 Ilustrasi kontroler PID pada robot maze



Gambar 4.7 Aplikasi kontroler PID pada robot maze

Deviasi/simpangan antar variabel terukur (PV) dengan nilai acuan (SP) disebut error (galat). *Setpoint* (SP) adalah suatu parameter nilai acuan atau nilai yang diinginkan. *Present Value* (PV) adalah nilai pembacaan sensor saat itu atau variabel terukur yang di umpan balik oleh sensor (sinyal *feedback*).

Tuning Eksperimen

Tuning eksperimen adalah proses yang dilakukan untuk mendapatkan hasil kontroler yang optimal dengan cara suatu percobaan. Inti dari *tuning* eksperimen adalah menentukan nilai dari tiga buah parameter yang terdapat pada kontroler PID yaitu konstanta proporsional (K_p), konstanta integral (K_i) dan konstanta diferensial (K_d). Ada beberapa metode yang dapat dilakukan dalam penalaan parameter kontroler PID dengan eksperimen seperti metode Ziegler-Nichols, metode Cohen-Coon dan metode empirik. Pada perancangan kontroler PID pada robot maze ini, penulis menggunakan metode empirik yang didasarkan pada efek masing-masing kontroler terhadap sistem, seperti yang tertera pada Tabel 4.1. Langkah atau acuan penentuan parameter K_p , K_i dan K_d diadopsi dari (Williams, 2006). Langkah metode tersebut ialah sebagai berikut:

- Langkah awal gunakan kontrol proporsional terlebih dahulu, abaikan konstanta integratif dan derivatifnya dengan memberikan nilai nol pada integratif dan derivatif.
- Tambahkan terus konstanta proporsional maksimum hingga keadaan stabil namun robot masih berosilasi.
- Untuk meredam osilasi, tambahkan konstanta diferensial dengan membagi dua nilai proporsional, amati keadaan sistem robot hingga stabil dan lebih responsif.
- Jika sistem robot telah stabil, kontrol integral dapat menjadi opsional, dalam artian jika ingin mencoba-coba tambahkan kontrol integral tersebut, namun pemberian nilai integral yang tidak tepat dapat membuat sistem robot menjadi tidak stabil.
- Nilai *sampling time* (waktu cuplik) juga mempengaruhi perhitungan PID, tentunya saat penggunaan kontrol integral dan diferensial.
- Periksa kembali performa sistem hingga mendapatkan hasil yang memuaskan.

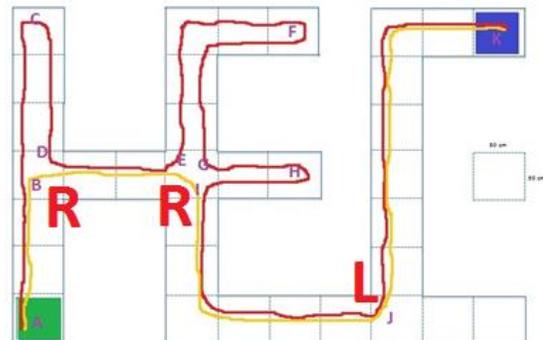
Tabel 4.1 Langkah atau acuan penentuan parameter K_p , K_i dan K_d diadopsi dari (Williams, 2006)

Respon Loop Tertutup	Waktu Naik	Overshoot	Waktu Turun	Kesalahan Keadaan Tunak
K_p	Menurun	Meningkat	Perubahan Kecil	Menurun
K_i	Menurun	Meningkat	Meningkat	Hilang
K_d	Perubahan Kecil	Menurun	Menurun	Perubahan Kecil

V. PENGUJIAN DAN ANALISIS SISTEM

A. Analisis Behavior Based pada Robot Maze

Analisis ini dilakukan dengan meletakkan sebuah target (berwarna biru) pada lorong labirin, lalu robot berjalan dari *home* (berwarna hijau), menggunakan mode telusur kiri atau kanan menuju target tersebut. Selama proses berjalannya robot tersebut maka sensor akan menstimuli berbagai perilaku (*behavior*). Selama robot mencari target maka aksitektur kendali *behavior based* memegang peranan penting. Saat menjumpai persimpangan mengakibatkan *behavior* persimpangan aktif maka koordinasi perilaku (*koordinasi kompetitif*) mengutamakan *behavior* ini dalam aksi pergerakan aktuator. Untuk lebih jelas mengenai hal pengujian aksitektur *behavior based* pada robot maze, perhatikan ilustrasi pengujian robot maze menggunakan metode telusur kiri dan kode-kode unik yang terbentuk didalamnya.

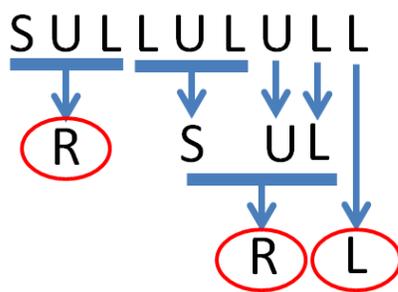


Gambar 5.1 Ilustrasi percobaan telusur kiri dan proses pemecahan maze-nya

Tabel 5.1 Pengujian kondisi dan perilaku percobaan telusur kiri dan proses pemecahan *maze*-nya

Posisi	Perilaku (<i>behavior</i>) dan strategi penyelesaian maze	Memori
A	Robot berangkat dari <i>home</i> menuju target dengan menggunakan metode telusur kiri.	-
B	Pada posisi ini, sensor ultrasonik depan dan kanan mendeteksi adanya perilaku “persimpangan tiga kekanan”, maka yang dilakukan robot adalah berjalan lurus dan menyimpan memori navigasinya dengan kode “S”.	S
C	Pada posisi ini, sensor ultrasonik depan, kiri dan kanan mendeteksi adanya perilaku “jalan buntu”, maka yang dilakukan robot adalah berbalik arah dan menyimpan memori navigasinya dengan kode “U”.	U
D	Pada posisi ini, sensor ultrasonik depan dan kiri mendeteksi adanya perilaku “persimpangan tiga kekiri”, maka yang dilakukan robot adalah belok kiri dan menyimpan memori navigasinya dengan kode “L”.	L
E	Pada posisi ini, sensor ultrasonik depan, kiri dan kanan mendeteksi adanya perilaku “persimpangan per-empat-an”, maka yang dilakukan robot adalah belok kiri dan menyimpan memori navigasinya dengan kode “L”.	L
F	Sama dengan kondisi pada posisi C.	U
G	Sama dengan kondisi pada posisi E.	L
H	Sama dengan kondisi pada posisi C dan F.	U
I	Sama dengan kondisi pada posisi E dan G.	L
J	Sama dengan kondisi pada posisi D.	L
K	Robot berhenti karena target telah ditemukan.	-

Dari hasil penjelajahan robot *maze* menggunakan telusur kiri pada *track maze* maka terbentuklah rekaman memori navigasi dengan kode berikut dan di sederhanakan menjadi “RRL”. Itu artinya bahwa disetiap persimpangan dimulai dari keberangkatan robot diperoleh arah navigasi ke kanan (R) untuk simpang pertama, ke kanan (R) untuk simpang yang kedua dan ke kiri (L) untuk simpang yang ketiga.



Gambar 5.2 Skema penerjemahan kode-kode unik yang terbentuk

B. Analisis Kontroler PID pada Robot Maze

Berikut ini hasil *tuning* nilai-nilai parameter kontroler PID robot *maze* dengan medan uji coba pada *track maze*. Acuan pengujian ini adalah robot diberi *setpoint* 6 cm dari dinding labirin lalu diamati respon masing-masing parameter kontroler PID yang menyebabkan sehingga robot tersebut mampu

menjaga dan mempertahankan posisinya pada jarak 6 cm dari dinding labirin.

Tabel 5.2 Pengujian kontroler proporsional

Kp	Ki	Kd	Kondisi Robot
5	0	0	Respon robot terlihat sangat lambat dan robot tidak stabil
10	0	0	Respon robot terlihat meningkat, berosilasi rendah dan masih tidak stabil
17	0	0	Respon robot terlihat cepat, stabil namun masih mengalami osilasi
25	0	0	Respon robot terlihat cepat, osilasi meningkat dan tidak stabil
30	0	0	Tidak stabil dan osilasi sangat tinggi

Dari hasil pengujian kontroler proporsional ini dapat disimpulkan bahwa robot mengalami osilasi tinggi saat penggunaan nilai proporsional yang besar. Kontroler proporsional belum mampu membuat pergerakan robot *maze* menjadi stabil.

Tabel 5.3 Pengujian kontroler proporsional dan diferensial

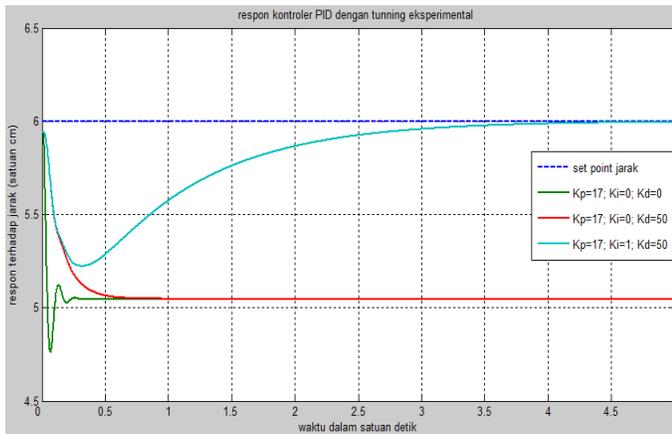
Kp	Ki	Kd	Kondisi Robot
17	0	10	Respon robot terlihat lambat, osilasi rendah dan robot tidak stabil
17	0	30	Respon robot terlihat meningkat, osilasi berkurang dan stabil
17	0	50	Respon robot terlihat cepat, osilasi jauh berkurang dan stabil
17	0	70	Respon robot terlihat melambat dan stabil
17	0	90	Respon robot terlihat semakin melambat dan stabil

Dari hasil pengujian kontroler proporsional ditambah dengan kontroler diferensial dapat disimpulkan bahwa kontroler diferensial mampu menghilangkan dan meredam osilasi yang disebabkan oleh kontroler proporsional dan membuat pergerakan robot *maze* menjadi stabil.

Tabel 5.4 Pengujian kontroler proporsional, diferensial dan integral

Kp	Ki	Kd	Kondisi Robot
17	0.1	50	Respon robot lambat, osilasi rendah dan robot tidak stabil
17	0.5	50	Respon robot terlihat meningkat, osilasi berkurang dan stabil
17	1	50	Respon robot terlihat cepat dan stabil tanpa osilasi
17	5	50	Respon robot terlihat cepat, namun berosilasi dan tidak stabil
17	10	50	Berosilasi dan tidak stabil

Dari hasil pengujian kontroler proporsional, diferensial ditambah kontroler integral dapat disimpulkan bahwa respon robot akan semakin baik bila penggunaan integral tepat, namun saat penggunaan integral yang berlebih justru menyebabkan robot *maze* mengalami osilasi dan menjadi sangat tidak stabil. Berikut hasil respon dari parameter *tuning* eksperimen dengan menggunakan simulink-matlab pada Gambar 5.3. Nilai parameter kontroler PID dengan kondisi respon robot terbaik adalah dengan $K_p=17$, $K_i=1$ dan $K_d=50$.



Gambar 5.3 Respon hasil *tuning* ekperimental kontroler PID

Dari hasil respon kontroler proporsional ini dapat disimpulkan bahwa respon kontroler ini adalah cepat dan beresilasi sesaat namun belum mampu menuju keadaan *steady*. Dari hasil tuning ini nilai kontroler proporsional yang di implementasikan kedalam robot *maze* adalah saat $K_p = 17$. Dengan ditambah dengan kontroler diferensial, mampu menghilangkan dan meredam osilasi yang disebabkan oleh kontroler proporsional namun pemberian nilai diferensial yang besar akan memperlambat respon sistem. Dari hasil tuning ini nilai kontroler diferensial yang di implementasikan kedalam robot *maze* adalah saat $K_d = 50$. Dengan ditambah dengan kontroler integral, maka respon mampu menuju keadaan *steady* namun pemberian nilai integral yang tidak tepat justru membuat respon menjadi beresilasi dan tidak stabil. Dari hasil tuning ini nilai kontroler integral yang di implementasikan kedalam robot *maze* adalah saat $K_i = 1$.

VI. PENUTUP

Dari hasil penelitian yang telah dilakukan pada pengerjaan tugas akhir ini, maka dapat diperoleh beberapa simpulan diantaranya:

1. Perancangan algoritma kendali berbasis perilaku (*behavior based*) untuk membuat robot *maze* dapat melakukan navigasi di dalam labirin telah berhasil dilakukan.
2. Arsitektur kendali berbasis perilaku sangat mementingkan pengujian yang terus menerus pada tiap tingkatan perilaku-nya. Bukanlah ide yang baik untuk menambahkan perilaku baru ketika perilaku yang ada di bawahnya belum dapat bekerja dengan baik.
3. Penerapan kontroler PID pada robot *maze* telah mampu membuat pergerakan robot *maze* menjadi sangat stabil dan mampu membuat robot *maze* bermanuver dengan aman, halus, responsif dan cepat, parameter kontroler PID tersebut diperoleh dari hasil *tuning* eksperimen dengan $K_p=17$, $K_i=1$ dan $K_d=50$.

Untuk kelanjutan riset yang akan datang, diharapkan adanya pengembangan metode kontroler dengan menggunakan kontroler cerdas seperti logika fuzzy, neuro-fuzzy, algoritma genetik atau kontroler cerdas lainnya. Selain itu juga perlu dikembangkan metode atau cara dalam *maze solving* seperti metode *flood fill* atau yang lainnya.

DAFTAR PUSTAKA

- [1] Brooks, R. (1986). "A robust layered control system for a mobile robot", IEEE Journal of Robotics and Automation Vol. 2, No. 1, hal.14–23.
- [2] Kweon, I, Kuno, Y, Watanabe, M, Onoguchi, K. (1992). "Behavior Based Mobile Robot Using Active Sensor Fusion", IEEE Journal of Robotics and Automation, Vol. 2, hal.1675 – 1682.
- [3] Pirjanian, P. (1999). "Behavior coordination mechanisms—State-of-the-art", Technical Report IRIS (Institute of Robotics and Intelligent Systems), University of Southern California, hal 99- 375.
- [4] Giessel David. "Building a Mouse", UAF MicroMouse Home Page: 2008
- [5] Mishra, Swati. Maze Solving Algorithm for Micro Mouse. IEEE International Conference on Signal Image Technology and Internet Based Systems: 2008
- [6] William, C. (2006). "Feedback and Temperature Control" diakses pada 21 Desember 2010, Tuning a PID Temperature Controller: <http://newton.ex.ac.uk/teaching/CDHW/feedback/setup-PID.html>

RIWAYAT PENULIS



Fahmizal dilahirkan di Bengkulu, 23 Juli 1988. Merupakan putra pertama dari pasangan Dr. Afrizal Mayub, M.Kom dan Dasmiroza. Lulus dari SDN 03 Bengkulu pada tahun 2000, kemudian melanjutkan studinya ke SLTPN 1 Depok Yogyakarta dan lulus pada tahun 2003. Kemudian melanjutkan ke SMAN 9 Yogyakarta dan lulus pada tahun 2006. Setelah menamatkan SMA, penulis melanjutkan studinya di Universitas Gadjah Mada tepatnya pada jurusan Diploma Teknik Elektro UGM dan lulus pada tahun 2009. Selanjutnya penulis meneruskan studi sarjana di Teknik Elektro ITS, kemudian fokus pada bidang studi Teknik Sistem Pengaturan. Pada bulan Juli 2011 penulis mengikuti seminar dan ujian Tugas Akhir sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik Elektro dari Institut Teknologi Sepuluh Nopember Surabaya.