

## Algoritma *Backtracking* Pada Perjalanan Pulang Robot Pemadam Api

Arya Sukma Krisnayoga<sup>1</sup>, Yoga Divayana<sup>2</sup>, Pratolo Rahardjo<sup>3</sup>  
Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana  
Email : [Aryasukma24@yahoo.com](mailto:Aryasukma24@yahoo.com)<sup>1</sup>, [yogadivayana@gmail.com](mailto:yogadivayana@gmail.com)<sup>2</sup>, [pratolo@unud.ac.id](mailto:pratolo@unud.ac.id)<sup>3</sup>

### Abstrak

Kontes Robot Pemadam Api Indonesia yang biasa disingkat KRPAI merupakan kompetisi robot yang mempunyai misi utama memadamkan api. Pada pelaksanaan kompetisi terdapat beberapa *point* bonus yang wajib diambil. Salah satu bonus yang wajib diambil adalah robot berhasil kembali pulang setelah memadamkan api. Robot KRPAI Universitas Udayana menggunakan algoritma *backtracking* sebagai algoritma perjalanan pulang. Tujuan penggunaan algoritma *backtracking* adalah untuk mempersingkat jarak perjalanan pulang. Robot ini menggunakan sensor *ultrasonic* SRF-05 sebagai pengukur jarak robot ke tembok. Penggerak robot menggunakan 18 motor servo yang berfungsi sebagai kaki robot. Pada penggunaan algoritma *backtracking*, jalur yang telah dilewati oleh robot pada saat mencari api akan disederhanakan. Hasil dari penyederhanaan jalur mencari api akan digunakan sebagai jalur kembali pulang. Berdasarkan hasil pengujian, dengan menggunakan algoritma *backtracking*, robot mampu meminimalisasikan jarak yang ditempuh saat kembali pulang, ruangan yang dimasuki hanya 1 ruangan, dan tidak ada mendeteksi boneka.

**Kata Kunci:** Robot Pemadam Api, KRPAI, Algoritma *Backtracking*, Perjalanan Pulang

### Abstract

*Kontes Robot Pemadam Api Indonesia or commonly in short KRPAI is a robot competition that has the main mission of putting out fires. During the competition there are several bonus points that must be taken. One bonus that must be taken is the robot able to return home after holding a fire. In this research the KRPAI Udayana University robot will use the backtracking algorithm to return home. The purpose of using backtracking alogritma is to shorten the distance back home to the robot. In KRPAI University robot Udayana use SRF-05 sensors to measure the distance of the robot to the wall and use 18 servo as the driving leg of the robot. the use of backtracking algorithm, the path that has been passed by the robot when looking for fire will be simplified. the results of the simplification of the fire search path will be used as the return route. Based on the test results, using the robot backtracking algorithm is able to minimize the distance traveled when returning home, the room entered is only 1 room, and there is no detecting dolls.*

**Keyword:** Fire Fighting Robot, KRPAI, Backtracking Algorrithm, Go to Home

<sup>1</sup>Mahasiswa, Program Studi Teknik Elektro Fakultas Teknik Universitas Udayana, jln. Betake gg. Nbeji no.5 dalung Badung 80361. Indoensia ( telp: 087885493133; e-mail: aryasukma24@yahoo.com)

<sup>2,3</sup>Dosen, Program Studi Teknik Elektro Fakultas Teknik Universitas Udayana, Jln. Jalan Kampus Bukit Jimbaran 80361 INDONESIA (telp: 0361-703315; fax: 0361-4321; e-mail: yogadivayana@gmail.com, pratolo@unud.ac.id)

## 1. Pendahuluan

Kontes Robot Indonesia atau yang biasa disingkat dengan KRI merupakan kompetisi yang diadakan oleh Kementerian Riset, Teknologi dan Pendidikan Tinggi Republik Indonesia. KRI merupakan kontes robot yang rutin diadakan setiap tahun. Peserta kompetisi ini adalah seluruh perguruan tinggi di Indonesia. Salah satu kompetisi yang diadakan adalah kontes robot pemadam api Indonesia (KRPAI). Pada kompetisi ini peserta diwajibkan untuk merancang robot otomatis yang memiliki kaki sebagai alat untuk bergerak. Robot yang dirancang harus berhasil melaksanakan misi utama yaitu memadamkan api yang terdapat di sebuah ruangan. Robot akan ditempatkan di sebuah ruangan kemudian robot akan menelusuri lorong berbentuk labirin untuk menemukan ruangan yang berisikan api dan memadamkannya[1].

Pada saat pelaksanaan misi terdapat beberapa bonus nilai yang wajib diambil oleh robot untuk mendapatkan *point* total terbesar. Salah satu dari bonus tersebut adalah robot mampu kembali pulang setelah memadamkan api ke ruangan dimana robot berasal[1]. Pada kompetisi KRPAI yang pernah diikuti tim robot Universitas Udayana robot belum mampu untuk kembali pulang setelah memadamkan api. Hal ini menyebabkan bonus nilai yang didapatkan robot KRPAI Universitas Udayana tidak maksimal dan memperkecil harapan untuk lolos

menuju kompetisi nasional. Berdasarkan permasalahan tersebut, diperlukan sebuah algoritma agar robot mampu kembali pulang.

Penelitian ini akan mengimplementasikan algoritma *backtracking* pada perjalanan pulang robot KRPAI Universitas Udayana. Perancangan akan dilakukan pada robot KRPAI berkaki dengan 6 kaki atau *hexapod*. Pada perancangan ini akan menggunakan Arduino Mega 2560 sebagai pusat pengendalian robot[2].

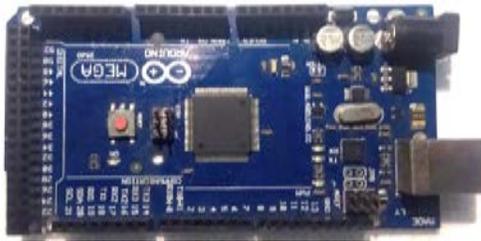
## 2. Kajian Pustaka

### 2.1 Algoritma *Backtracking*

Pada Tahun 1950 Algoritma *backtracking* diperkenalkan oleh D.H Lehmer[3]. Algoritma ini mencari solusi berdasarkan ruang solusi yang ada namun tidak semua ruang solusi diperiksa, hanya pencarian yang mengarah ke solusi yang akan diperiksa. Maka algoritma ini berbasis pada algoritma *depth first search*[4]. Pencarian solusi dilakukan dengan cara menelusuri struktur pohon dari akar hingga daun. Simpul yang dilahirkan dinamakan simpul hidup (*live node*). Simpul hidup yang diperluas dinamakan simpul-E (*expand node*). Sedangkan simpul yang tidak mengarah ke solusi dinamakan simpul mati (*dead node*)[4].

### 2.2 Arduino Mega 2560

Arduino mega adalah salah satu dari *board* mikrokontroler dengan menggunakan IC mikrokontroler ATmega 2560. Arduino mega 2560 memiliki 54 buah pin I/O. Arduino mega 2560 umumnya digunakan pada pengolahan data dan pengendalian yang membutuhkan banyak pin I/O. Bahasa pada arduino adalah bahasa C yang telah dimodifikasi sehingga lebih mudah [2]. Gambar 1 merupakan penampilan dari arduino mega 2560.



Gambar 1 Arduino Mega 2560[2]

### 2.3 Sensor Ultrasonic SRF-05

Sensor ultrasonic SRF-05 merupakan sensor yang dapat digunakan untuk mengukur jarak robot ke tembok. Prinsip kerja dari sensor SRF-05 adalah pemancar gelombang akan mengirimkan gelombang ultrasonic, kemudian diukur waktu yang dibutuhkan gelombang untuk kembali dari pantulan objek dan diterima oleh penerima gelombang. [5]. Gambar 2 merupakan tampilan dari sensor SRF-05.



Gambar 2 Sensor SRF-05[5]

### 2.4 Sensor UV-Tron

Sensor UV-Tron adalah sensor yang mendeteksi sinar ultraviolet. Sensor UV-Tron memiliki spesifikasi pendeteksian spektrum cahaya dari 185nm hingga 260 nm. Jarak deteksi dari sensor UV-tron mencapai 5 meter[6]. Pada penelitian ini sensor UV-tron digunakan untuk mendeteksi adanya sinar ultraviolet yang dihasilkan oleh api. Gambar 3 merupakan tampilan dari sensor UV-Tron.



Gambar 3 Sensor UV-Tron[6]

### 2.5. Motor Servo

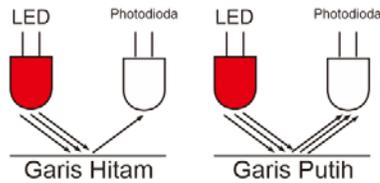
Motor servo adalah aktuator yang dirancang dengan sistem kontrol umpan balik, sehingga motor dapat diatur untuk menentukan sudut dari poros motor servo [7]. Motor servo terdiri dari motor DC, mekanisme gear, potensiometer, dan rangkaian kontrol. Untuk mengontrol sudut dari servo, diperlukan sinyal pulsa (PWM) ke pin sinyal motor servo. Gambar 4 merupakan tampilan dari motor servo.



Gambar 4 Motor servo[7]

### 2.6 Sensor Cahaya Photodioda

Sensor photodioda merupakan dioda yang peka terhadap cahaya. Sensor photodioda akan mengalami perubahan resistansi pada saat menerima cahaya dan akan mengalirkan arus listrik secara forward sebagaimana dioda pada umumnya [8]. Pada robot KRPAI photodioda digunakan sebagai pendeteksi garis pintu ruangan. Gambar 5 merupakan ilustrasi pemantulan cahaya dari LED dan dideteksi oleh photodioda.



Gambar 5 Ilustrasi pemantulan cahaya dari LED[8]

### 2.7 Sensor Infrared GY2Y0A21YK

Sensor *Infrared* GY2Y0A21YK merupakan sensor *infrared* yang digunakan untuk mengukur jarak suatu benda berdasarkan intensitas cahaya yang diterima [9]. Penggunaan sensor *infrared* GY2Y0A21YK ini bertujuan untuk mendeteksi boneka yang menjadi rintangan pada lorong. Gambar 6 merupakan tampilan sensor infrared GY2Y0A21YK.

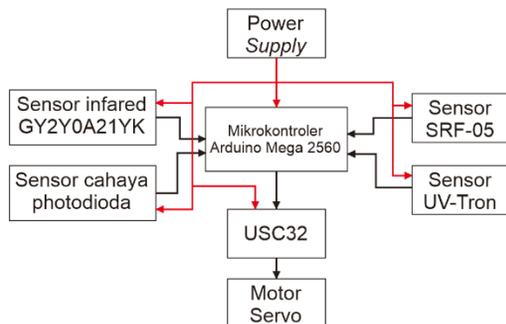


Gambar 6 sensor infrared GY2Y0A21YK [9]

## 3. Metodologi Penelitian

### 3.1 Diagram Blok Hardware

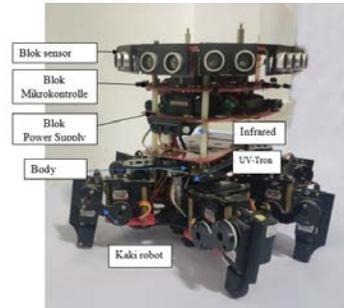
Perangkat keras (*hardware*) pada penelitian ini menggunakan beberapa jenis *hardware* yaitu, mikrokontroler Arduino Mega 2560, sensor SRF-05, sensor UV-Tron, sensor photodiode, sensor *infrared* GY2Y0A21YK, USC32, dan Motor Servo. Gambar 8 merupakan diagram blok *hardware* robot pemadam api.



Gambar 8 Diagram blok *hardware*

### 3.2 Desain Robot

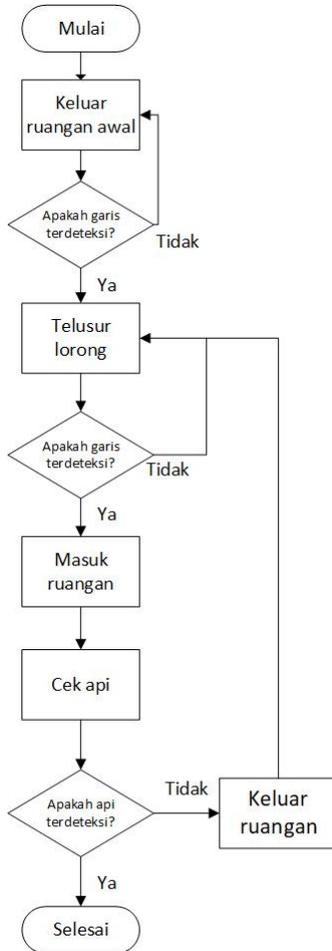
Robot KRPAl Universitas Udayana memiliki desain robot *hexapod*, yaitu memiliki 6 kaki sebagai penggeraknya dengan 18 *Degree of freedom* (DOF). Robot KRPAl Universitas Udayana memiliki dimensi tinggi 30 Cm, panjang 30 Cm, lebar 25 Cm. Bahan yang digunakan untuk membuat rangka robot adalah akrilik. Gambar 9 Desain robot KRPAl Universitas Udayana.



Gambar 9 Desain robot KRPAl Universitas Udayana

### 3.3 Penelusuran Labirin

Penelusuran labirin bertujuan untuk menemukan ruangan yang terdapat api. Mula-mula robot diletakan di sebuah ruangan yang tidak terdapat api. Kemudian robot dihidupkan dan berjalan secara otomatis untuk mencari api. Pada penelusuran lorong robot akan menggunakan metode telusur kanan, dimana ketika robot menemukan persimpangan maka robot akan memilih belok kanan[10]. Jika dalam perjalanan di lorong robot mendeteksi garis pintu ruangan maka robot dinyatakan memasuki ruangan. Setelah memasuki ruangan, robot akan memeriksa apakah terdapat api di ruangan tersebut. Jika robot tidak mendeteksi adanya api maka robot akan keluar dari ruangan dan kembali menelusuri lorong hingga menemukan ruangan yang terdapat api. Gambar 10 merupakan diagram alir penelusuran labirin.

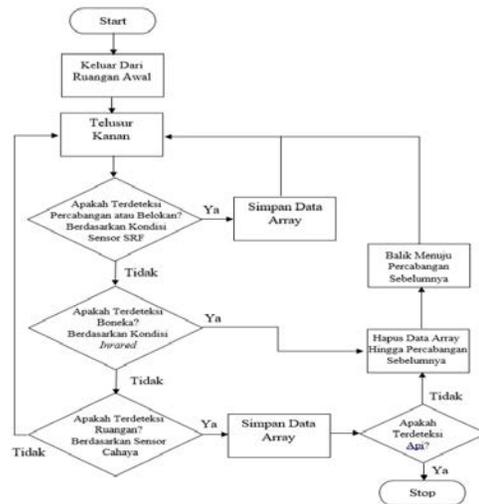


Gambar 10 Diagram alir penelusuran labirin

### 3.4 Algoritma Backtracking

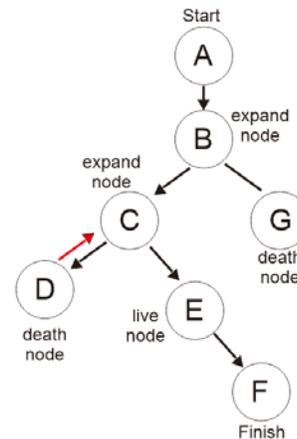
Algoritma *backtracking* pada robot KRPAI Universitas Udayana berfungsi sebagai algoritma mapping pada perjalanan robot. Cara kerja dari algoritma *backtracking* pada robot KRPAI Universitas Udayana adalah robot akan menelusuri labirin dengan struktur berbentuk pohon dari akar menuju daun secara mendalam[11]. Jika pada perjalanan robot terdapat belokan, persimpangan, boneka, maupun ruangan, maka kondisi tersebut akan melahirkan simpul-simpul baru yang dinamakan (*live node*). Jika terdapat persimpangan maka simpul pada persimpangan akan menghasilkan simpul yang diperluas (*expand node*). Setiap simpul akan

diberikan kode sesuai kondisi simpul tersebut. Jika robot memasuki ruangan dan tidak terdapat api maka pencarian solusi pada simpul tersebut akan berakhir dan robot akan kembali menuju simpul *expand* dan menelusuri cabang lainnya. Simpul-simpul yang tidak mengarah ke solusi akan dinamakan simpul mati (*death node*). Pencarian solusi akan berakhir saat robot menemukan api di sebuah ruangan. Gambar 11 merupakan diagram alir algoritma *backtracking*.



Gambar 11 Diagram alir algoritma *backtracking*

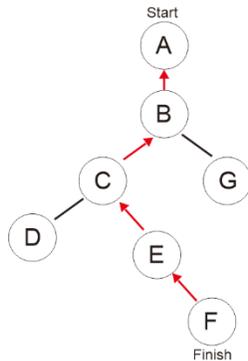
Gambar 12 merupakan pencarian solusi dengan struktur pohon pada algoritma *backtracking*.



Gambar 12 Struktur pohon pada algoritma *backtracking*

### 3.5 Perjalanan Pulang

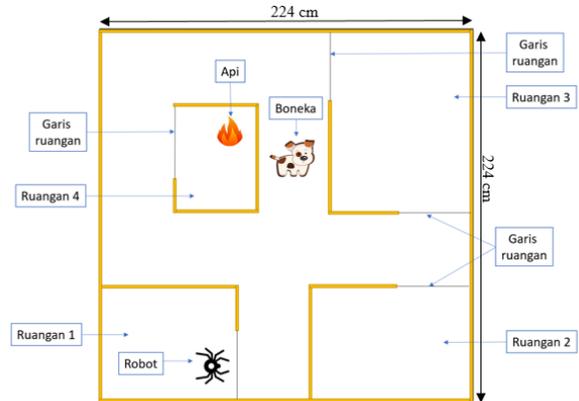
Perjalanan pulang robot KRPAI Universitas Udayana akan dilakukan setelah robot menyelesaikan pencarian api. Perjalanan pulang robot KRPAI Universitas Udayana akan memanfaatkan simpul-simpul yang telah dilahirkan saat penelusuran labirin. Simpul-simpul yang telah dilahirkan akan disederhanakan dengan menghilangkan simpul-simpul mati. Tujuan menghilangkan simpul – simpul mati adalah untuk mempersingkat perjalanan robot dan tidak lagi menelusuri simpul yang tidak mengarah ke solusi. Robot akan berjalan mengikuti simpul yang telah disederhanakan. Gambar 13 merupakan struktur perjalanan pulang robot.



Gambar 13 Struktur perjalanan pulang robot KRPAI

### 3.6 Arena KRPAI

Arena KRPAI terdiri dari 4 ruangan, 1 boneka sebagai halangan pada lorong. Robot ditempatkan pada salah satu ruangan. Posisi api ditempatkan pada ruangan berbeda dengan ruangan asal robot. Arena berukuran 244 cm x 244 cm x 30 cm[1]. pada pintu ruangan ditempatkan garis putih sebagai tanda memasuki ruangan. Gambar 14 merupakan arena KRPAI.



Gambar 14 Arena KRPAI[1]

## 4. Hasil dan Pembahasan

### 4.1 Pengujian Perjalanan Mencari Api

Pengujian dilakukan dengan menjalankan robot dengan posisi ruangan awal yaitu ruangan nomor 4. Api diletakan pada ruangan nomor 3. Perjalanan mencari api menggunakan metode telusur kanan dengan cara pencarian secara mendalam atau *deep fisrt search*. Robot akan mencatat setiap belokan, persimpangan, boneka, garis, dan kondisi api. Tujuan dari disimpannya data pada perjalanan mencari api adalah untuk dapat disederhanakan pada perjalanan pulang. Tabel 1 merupakan hasil dari pengujian robot KRPAI mencari api.

Tabel 1 Perjalanan Mencari Api

NO.	Posisi Robot	Data Tersimpan	Keterangan
1		<pre>COM27     mulai 1 = RA</pre>	Robot berada pada posisi ruangan awal. Maka data yang tersimpan adalah "RA"
2		<pre>COM27     mulai 1 = RA 2 = PI pertigaan = 1</pre>	Keluar dari ruangan, robot mendeteksi pertigaan dan data yang tersimpan adalah "PI"
3		<pre>mulai 1 = RA 2 = PI pertigaan = 1 3 = RA</pre>	Robot mendeteksi adanya belokan kanan, data tersimpan "KA"

4		2 = PI pertigaan = 1 3 = KA 4 = PI pertigaan = 2	Robot mendeteksi pertigaan, data tersimpan "PI"
5		pertigaan = 1 3 = KA 4 = PI pertigaan = 2 5 = BO	Robot mendeteksi adanya boneka dan harus berbalik, data tersimpan "BO"
6		/////////penghap 5 = - 4 = PI 3 = KA 2 = PI 1 = RA /////////	Saat mendeteksi boneka, maka data menuju boneka disederhanakan
7		/////////penghap 5 = - 4 = PI 3 = KA 2 = PI 1 = RA /////////	Robot kembali menuju pertigaan terakhir dilalui.
8		2 = PI 1 = RA ///////// 6 = MR	Robot mendeteksi garis maka dinyatakan memasuki ruangan, data tersimpan "MR"
9		///////// 6 = MR 7 = A <input checked="" type="checkbox"/> Autoscroll <input type="checkbox"/>	Ruangan yang dimasuki terdapat api, data tersimpan "A"

Setelah berhasil menemukan api maka robot dapat kembali ke ruangan asal robot. Pada perjalanan pulang akan diuji sebanyak 2 kali. Yang pertama perjalanan pulang menggunakan algoritma *backtracking*. Yang kedua menggunakan telusur kiri.

#### 4.2 Perjalanan Pulang dengan Algoritma *Backtracking*

Pada perjalanan mencari api terdapat data yang telah disederhanakan karena data tersebut bukan jalur yang mengarah ke solusi. Data yang telah disederhanakan akan dijadikan panduan untuk robot dapat kembali pulang dengan jalur tercepat.

Tabel 2 merupakan hasil dari pengujian perjalanan pulang menggunakan algoritma *backtracking*.

Tabel 2 Perjalanan pulang dengan algoritma *backtracking*

NO.	Posisi Robot	Data	Keterangan
1		ayo pulang ///////// 7 = A 6 = MR 5 = -	Robot keluar ruangan menuju pertigaan.
2		///////// 7 = A 6 = MR 5 = - 4 = PI	Setelah mendeteksi pertigaan robot menuju belokan
3		7 = A 6 = MR 5 = - 4 = PI 3 = KA	Setelah berbelok robot berjalan menuju pertigaan
4		5 = - 4 = PI 3 = KA 2 = PI	Robot mendeteksi pertigaan dan menuju ruangan awal
5		3 = KA 2 = PI 0 1 = RA	Saat data ada pada urutan 1 dan garis dideteksi maka robot memasuki ruangan awal

Berdasarkan data yang telah disederhanakan saat perjalanan mencari api, perjalanan pulang mampu ditempuh dengan jalur terpendek menuju ruangan awal. Robot akan berjalan mengikuti data yang telah disederhanakan dari urutan terbesar hingga ke terkecil. Pada perjalanan pulang robot tidak lagi mendeteksi boneka karena robot mampu menentukan arah yang paling tepat pada pertigaan agar langsung menuju ruangan awal robot.

#### 4.3 Perjalanan Pulang dengan Telusur Kiri

Perjalanan pulang menggunakan telusur kiri tidak lagi memanfaatkan data yang telah dicatat saat mencari

api. Perjalanan pulang dengan telusur kiri adalah kebalikan dari perjalanan mencari api dengan telusur kanan. Pada perjalanan pulang dengan telusur kiri robot akan menghitung ruangan yang dimasuki saat mencari api, kemudian pada perjalanan pulang robot harus memasuki ruangan sebanyak ruangan yang dimasuki saat mencari api. Tabel 3 merupakan hasil dari pengujian perjalanan pulang menggunakan telusur kiri.

**Tabel 3** Perjalanan pulang dengan telusur kiri

NO.	Posisi Robot	Keterangan
1		Robot berada di ruangan api. Menuju keluar ruangan
2		Robot mendeteksi pertigaan dan berbelok ke kiri
3		Robot mendeteksi boneka dan berbalik ke pertigaan terakhir
4		Robot kembali ke pertigaan terakhir dan berbelok ke kiri
5		Robot mendeteksi belokan
6		Robot mendeteksi pertigaan dan berbelok ke kiri
7		Robot mendeteksi garis dan memasuki ruangan awal

Pada perjalanan pulang menggunakan telusur kiri, jalur yang

ditempuh sama dengan jalur perjalanan mencari api. Pada perjalanan mencari api ruangan yang dimasuki hanya 1 maka pada perjalanan pulang dengan telusur kiri, ruangan pertama yang dimasuki adalah ruangan awal robot.

#### 4.4 Data Hasil Pengujian

Data hasil pengujian dilakukan sebanyak 6 kali. Posisi robot dan api akan diubah di setiap pengujian. pengujian dilakukan dengan menaruh robot pada ruangan yang berbeda dengan api.

Gambar 15 adalah grafik jarak perjalanan antara jarak mencari api, jarak jalur terdekat, perjalanan pulang dengan algoritma *backtracking* dan perjalanan pulang dengan telusur kiri.



**Gambar 15** Grafik jarak perjalanan

Gambar 16 adalah grafik jumlah memasuki ruangan antara jarak mencari api, jarak jalur terdekat, perjalanan pulang dengan algoritma *backtracking* dan perjalanan pulang dengan telusur kiri.



**Gambar 16** Grafik Jumlah memasuki ruangan

Gambar 17 adalah grafik banyaknya boneka dideteksi antara jarak mencari api, jarak jalur terdekat, perjalanan pulang dengan algoritma *backtracking* dan perjalanan pulang dengan tulusur kiri.



**Gambar 17** Grafik Jumlah memasuki ruangan

Berdasarkan hasil perbandingan grafik pada gambar 15 dengan gambar 16 dan gambar 17 maka, semakin banyak ruangan yang dimasuki, dan boneka yang dideteksi maka perbedaan jarak yang ditempuh antara perjalanan pulang menggunakan *backtracking* dengan perjalanan mencari api dan telusur kiri semakin besar. Dengan menggunakan algoritma *backtracking* mampu meminimalisir jarak perjalanan, persimpangan, jumlah ruangan, jumlah boneka yang dideteksi dalam mencari jalan pulang.

## 5. Kesimpulan

Dengan menggunakan telusur kanan robot telah mampu menelusuri lorong, memasuki ruangan dan menemukan api di salah satu ruangan. Pada perjalanan pulang menggunakan telusur kiri, jarak perjalanan, ruangan yang dimasuki, jumlah persimpangan, dan jumlah boneka yang dideteksi sama dengan perjalanan mencari api. Pada perjalanan pulang menggunakan algoritma *backtracking*, jarak perjalanan, ruangan yang dimasuki, jumlah persimpangan, dan jumlah boneka yang dideteksi sama dengan

perjalanan mencari api. Dengan menggunakan algoritma *backtracking*, robot mampu meminimalisasikan jarak perjalanan, ruangan yang dimasuki, jumlah persimpangan, dan jumlah boneka yang dideteksi pada perjalanan pulang.

## Daftar Pustaka

- [1] (2019). Kontes Robot Pemadam Api Indonesia. Tersedia di: <https://ristekdikti.go.id>
- [2] (2019). Arduino Mega 2560. Tersedia di: <https://www.arduino.cc/>
- [3] Rusmini, Setiawardhana, M. Iqbal Nugraha. Pencarian Jalur Terpendek Untuk Robot Micromouse Dengan Menggunakan Algoritma *Backtracking*. Politeknik Elektronika Negeri Surabaya. pp 41-48. 2011
- [5] (2019). SRF05. Tersedia di: <https://www.robot-electronics.co.uk/html/srf05tech.htm>
- [6] (2014). Flame sensor (UV TRON). <https://www.hamamatsu.com/>
- [7] (2019) HS-645MG High Torque, Metal Gear Premium Sport Servo. Tersedia di: <https://hitecrd.com/>
- [8] (2010) 3mm Photodiode PD204-6C/L3. Tersedia di: [www.everlight.com](http://www.everlight.com)
- [9] (2019) GP2Y0A21YK0F. Tersedia di: <https://global.sharp/>
- [10] Faela Shofa, Tatyantoro Andrasto. Penerapan Meote Simple Maze Pada Robot Wall Follower Untuk Menyelesaikan Jalur Dalam Menelusuri Sebuah Labirin. Universitas Negeri Semarang. Vol 4, pp 35-43. 2015
- [11] Fikri Fauzan, Dahnial Syauqy, Rizal Maulana. Implementasi Algoritme Mapping Dengan *Backtracking* Pada Mobile Robot Dalam Maze. Universitas Brawijaya. Vol 3, pp. 2459-2464. 2019