

AUTONOMOUS MOBILE ROBOT MENGUNAKAN METODE FORMAL LOGIKA TEMPORAL LINIER

Irvan Lewenusa, Wisnu Ananta Kusuma

Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam

Institut Pertanian Bogor (IPB)

Jl. Meranti Kampus IPB Darmaga-Bogor Telp/Fax : (0251) 625584

E-mail: lw_wufei@gmail.com, ananta@ipb.ac.id

Abstrak

Logika temporal linier (LTL) dikonstruksi dari formulasi logika tingkah laku sistem yang diharapkan dari waktu ke waktu secara linier. Dengan demikian jika pada bagian atau waktu tertentu terjadi kesalahan pada sistem, kesalahan itu dapat segera diantisipasi. Dalam hal ini penggunaan perencanaan logika temporal linier tidak hanya digunakan pada autonomous mobile robot untuk mencapai sebuah tujuan atau menghindari rintangan, tetapi juga untuk perangkaian, perluasan atau penugasan sementara (temporal ordering) pada sebuah rangkaian tugas yang berbeda dalam satu waktu. Dalam LTL untuk menghasilkan lintasan robot dilakukan tiga tahap aktivitas, yaitu pendefinisian abstraksi diskrit pergerakan robot, perencanaan logika temporal menggunakan model checking, dan implementasi kontinu rencana diskret (continuous implementations of discrete plan).

Pada penelitian ini, untuk membuktikan apakah model perencanaan logika temporal linier yang dibuat sebelumnya memenuhi spesifikasi tingkah laku sebuah mobile robot yang dimaksud, maka dibuatlah sebuah autonomous mobile robot sebagai alat uji implementasi formula logika temporal linier tersebut. Untuk membangun sebuah autonomous mobile robot tersebut diperlukan beberapa perangkat elektronika di antaranya mikrokontroler, sensor, kompas, dan motor penggerak. Selain itu dilakukan pula simulasi dengan perangkat lunak MOBOTSIM versi 1.0.03.

Implementasi formulasi dengan simulasi menggunakan MOBOTSIM menunjukkan hasil yang lebih ideal daripada pembangunan formulasi pada robot sebenarnya. Hal ini disebabkan pergerakan protipe mobile robot banyak dipengaruhi faktor luar yang dalam penelitian ini diabaikan.

Keyword :Autonomous Mobile robot, Logika temporal linier, Metode Formal

1. PENDAHULUAN

Mobile robot adalah tipe robot yang paling populer dalam dunia penelitian robotik. Dari segi manfaat, penelitian tentang berbagai tipe *mobile robot* diharapkan dapat membantu manusia dalam melakukan otomasi dalam transportasi, *platform* bergerak untuk robot industri, eksplorasi tanpa awak, dan lain-lain. Penelitian mengenai teknologi robot telah banyak dilakukan. Berbagai jenis robot telah dikembangkan untuk memenuhi berbagai kepentingan manusia misalnya di bidang industri, robot digunakan untuk melakukan pekerjaan dengan resiko bahaya yang tinggi atau melakukan pekerjaan yang membutuhkan tenaga besar.

Penggunaan *mobile robot* sebagai robot penjelajah (eksplorasi) mendapatkan sebuah perhatian khusus bagi para pengembang *mobile robot*. Hal ini ditunjukkan dengan diadakannya berbagai kompetisi atau perlombaan, diantaranya ABU Robocon, Trinity College, Kontes Robot Cerdas Indonesia dan berbagai perlombaan robot lainnya.

Penelitian ini dilatarbelakangi dari pengembangan *mobile robot* penjelajah sebagai salah satu penelitian yang sedang berkembang saat ini. Kendali navigasi dan waktu yang dibutuhkan untuk menuju ke sebuah tempat pada lokasi tertentu saat ini merupakan salah satu permasalahan yang dihadapi dalam pengembangan *mobile robot* penjelajah.

Metode formal (*formal method*) merupakan teknik berdasarkan matematis yang dapat digunakan untuk merepresentasikan spesifikasi sistem dengan bahasa spesifikasi tertentu (Tri Supriyo 2003). Dengan demikian diharapkan dapat diwujudkan sistem yang memiliki perilaku tertentu yang sebagaimana telah ditetapkan sebelumnya. Salah satu metode yang banyak digunakan adalah Logika Temporal Linier (LTL). Logika temporal linier (LTL) dikonstruksi dari formulasi logika tingkah laku sistem yang diharapkan dari waktu ke waktu secara linier. Pada penelitian Widiyanto (2008) LTL diterapkan untuk menyusun spesifikasi *Mobile robot* berdasarkan kebutuhan rancangan kendali navigasi robot KRCI (Kontes Robot Cerdas Indonesia) 2006. Pada penelitian ini, akan dibuat sebuah *autonomous mobile robot* sebagai alat uji implementasi formula logika temporal linier yang telah dihasilkan dari penelitian Widiyanto (2008).

2. TINJAUAN PUSTAKA

Robot

Robot adalah perangkat mekanik yang dapat dikendalikan oleh perangkat lunak yang menggunakan sensor untuk memandu satu atau lebih efektor melalui gerakan terprogram dalam suatu ruang kerja dalam hal untuk manipulasi obyek fisik. (Schilling, 2000).

Terdapat 2 (dua) kendali robot yaitu *high-level controller* dan *low-level controller*. *Low level control* digunakan untuk mendefinisikan aksi *primitive* dan mengkomunikasikan status sensor kepada *high level controller* (Shanahan & Witkowski 2000). Dalam *high-level planning*, terdapat 2 (dua) pendekatan yaitu *Robotic-Level Languages*, dan *Task- Level Languages*. *Robotic level languages* memandang rangkaian *task* sebagai rangkaian gerakan robot, dan setiap statement program secara kasar berkaitan dengan aksi robot. *Task level languages* memandang rangkaian *task* sebagai rangkaian posisional *goal* dari obyek lebih.

Kendali navigasi robot, berada pada *high-level controller*. Kendali robot *high-level* digunakan untuk mendefinisikan *task* (*task planner*) berdasarkan sensor yang ada, menghasilkan lintasan robot (*robot trajectory*), dan aksi atau pergerakan robot (*robot motion*) dari suatu tempat ke tempat lain.

Mobile robot atau *mobile robot* adalah robot yang memiliki mekanisme penggerak berupa roda (*wheel*) dan atau kaki (*leg*), untuk dapat berpindah tempat dari suatu tempat ke tempat yang lain.

Metode Formal Logika Temporal Linier

Proses untuk membentuk sebuah model pemeriksaan adalah dengan mempelajari pustaka yang berkaitan secara seksama, kemudian mendapatkan ide yang mendasari lahirnya konsep pemodelan pemeriksaan logika temporal linier dan membuat simulasi, karena ternyata pengujian tradisional seperti testing dan simulasi merupakan fungsi masukan (input) dan keluaran (output), sehingga tidak dapat digunakan untuk pengujian atau pemeriksaan terhadap sistem-sistem reaktif yang beroperasi secara terus menerus. Tingkah laku yang diharapkan dari sistem-sistem reaktif, dapat dinyatakan dalam bahasa logika, kemudian memformulasikannya ke dalam logika proposional sehingga membentuk logika temporal linier (LTL)..

Dalam LTL untuk menghasilkan (lintasan robot) *robot trajectory* terdiri dari 3 tahap : Abstraksi diskrit pergerakan robot, *temporal logic planning* menggunakan *model checking*, dan implementasi kontinu rencana diskret (*continuous implementations of discrete plan*). (Lliston & Wardoyo 2005).

Struktur Kripke

Struktur Kripke merupakan tipe dari nondeterministic *finite state machine* yang digunakan dalam *model checking* untuk merepresentasikan lingkungan dari sebuah sistem. Pada dasarnya sebuah graph yang node-nodenya merepresentasikan *state* yang akan dicapai dari sistem dan *edge-edge* merepresentasikan transisi *state*. Pelabelan fungsi pemetaan tiap node untuk menetapkan properti yang tersedia. Logika temporal digunakan untuk merepresentasikan mekanisme yang merelasikan *state* sistem dalam Struktur Kripke, melalui Büchi Automata. (Clarke, Grumberg and Peled 1999)

Struktur Kripke adalah struktur dengan bentuk tuple $M = (S, S_0, R, L)$ di mana :

S	:	Himpunan semua state dalam sistem
S ₀	:	Himpunan initial state
R	:	Relasi transisi antara state
L	:	Fungsi yang memetakan setiap state dengan proposisi yang bernilai benar pada state tersebut.

Büchi Automata

Büchi Automata adalah pengembangan dari teori automata untuk input yang tak hingga. Suatu runtun yang bertransisi melewati *state final* sebanyak tidak berhingga kali, maka runtun tersebut diterima oleh Automata.

Bila dalam teori otomata memiliki tuple $A = (Q, \Sigma, \delta, I, F)$, maka dalam Büchi Automata memiliki tuple $A = (Q, \Sigma, \delta, I, FT)$ di mana :

Q	:	Himpunan state
Σ	:	Finite Alphabet
δ	:	Relasi Transisi
I	:	Himpunan state awal (Initial State)
FT	:	Himpunan state yang dapat diterima (Acceptance State) di mana $FT = (F_1, F_2, F_3, \dots, F_k)$ dengan $F_i =$ himpunan state yang dapat diterima otomata Büchi.

3. METODE PENELITIAN

3.1 Bahan dan Alat yang Digunakan

Bahan dan alat yang digunakan dalam penelitian ini mencakup perangkat lunak dan perangkat keras.

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut :

- Microsoft Windows XP Profesional
- *MOBILE ROBOTSIM* Ver 1.0.0.3
- SDCC 2.7
- ATMEL AT89ISP Ver 2.2
- Protel DXP 2004

Sedangkan perangkat keras yang digunakan adalah sebagai berikut :

- PC Intel Pentium IV 2.67 GHz
- DDRAM 512 MB
- VGA
- *hardisk* dengan kapasitas 120 GB
- monitor VGA dengan resolusi 1024x768 *pixel*
- *keyboard*
- *mouse*
- Mikrokontroler AT89C51 Min-Sys
- *H-Bridge*
- *Power supply*
- *Driver sensor*
- *Motor DC*
- *Motor Servo*
- Sensor Ultrasonik PING Parallax
- *Driver Power supply*
- *Foto Transistor*
- *Foto Diode*
- Kompas Digital *CMPS03*
- Sensor Inframerah
- *Driver fan*

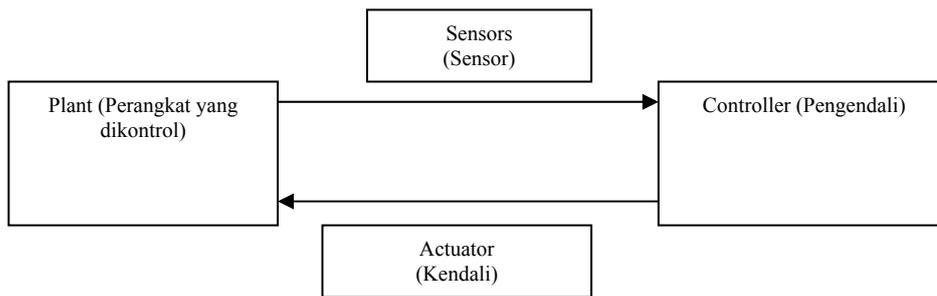
3.2 Metode Penelitian

Berdasarkan hasil formulasi Widiyanto (2008) dilakukan kajian pustaka mengenai konsep robotika, metode formal logika temporal linier, pemrograman simulasi dan implementasi prototipe pada *mobile robot*. Selanjutnya implementasi *mobile robot* dibangun dengan metode *prototyping* (Pressman, 2002) yang terdiri dari tiga tahap, yaitu pendefinisian kebutuhan sistem, konstruksi sistem yang meliputi perancangan dan implementasi, serta evaluasi sistem. Pada tahap pendefinisian kebutuhan dikaji spesifikasi formal dari hasil penelitian Widiyanto (2008). Tahap selanjutnya dilakukan perancangan, meliputi perancangan bentuk kerangka robot yang akan mempengaruhi penempatan posisi sensor, jenis sensor, dan posisi sensor yang optimal, perancangan aktuator yang meliputi arah putaran motor penggerak roda dan efek atau pengaruhnya terhadap arah gerakan dan perancangan proses. Selanjutnya dilakukan pengkodean program untuk mengimplementasikan rancangan proses yang mengendalikan pergerakan robot dari satu tempat ke tempat lain, dan menghasilkan nilai logika atau proposisi "benar" bila gerakan robot sukses. Prototipe yang dihasilkan kemudian dievaluasi dengan mengamati hasil pemrograman dan perilaku sistem, dan akan dilakukan kembali koreksi program hingga perilaku sistem valid atau sesuai dengan yang diharapkan.

4. HASIL DAN PEMBAHASAN

4.1 Deskripsi Umum

Dalam kajian robotika, diungkap robot merupakan sistem yang berdasarkan model sistem kontrol (sistem kendali) yang digambarkan dalam Gambar 1.



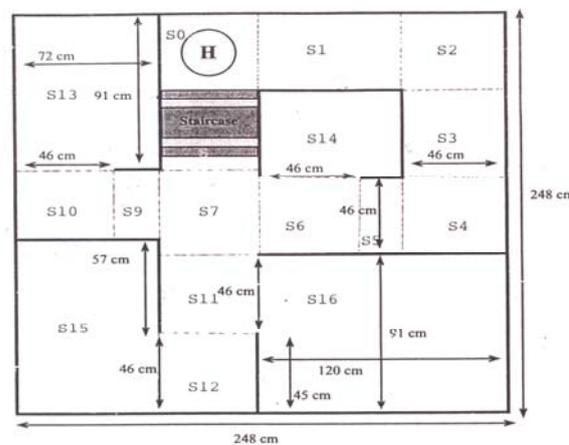
Gambar 1 Sebuah model sistem kontrol.

Plant adalah perangkat yang dikendalikan dalam hal ini adalah pergerakan robot melalui aktuatornya (*actuator*) yang berupa motor penggerak roda. *Mobile robot* memiliki beberapa *sensor* untuk memberikan masukan berupa kondisi kepada pengendali (*controller*).

4.2 Kebutuhan Sistem

Spesifikasi formal hasil penelitian Widiyanto (2008) menginformasikan tiga hal, yaitu representasi graf berdasarkan abstraksi diskrit yang merupakan transformasi dari interpretasi lingkungan yang kontinu, formula proposisi, dan tujuan (*goal*) yang ingin dicapai.

Ruang diskrit dengan menggunakan representasi graf (*graph*) yang merupakan hasil transformasi dari interpretasi lingkungan robot sebagai ruang kontinu. Lingkungan *mobile robot* diambil dari persyaratan yang ditetapkan Kontes Robot Cerdas Indonesia (KRCI) 2007). Abstraksi diskrit dilakukan dengan cara membagi-bagi *hallway* ke dalam ruang semu yang merepresentasikan state. Abstraksi diskrit ini diambil dari data sekunder seperti pada Gambar 2 sebagai berikut :



Gambar 2 Bentuk environment *mobile robot* dalam KRCI 2007 (Widiyanto 2008).

Selanjutnya abstraksi diskrit tersebut direpresentasikan dalam bentuk graf (*graph*) yang dimodelkan dalam bentuk tuple *Mobile robotenv* = (S, I, F, T, G, P) (Widiyanto, 2008).

Dari himpunan *trajectory* optimal dibentuk sebuah formula proposisi \emptyset berupa suatu persamaan matematika logika yang disusun dari beberapa fungsi proposisi dengan menggunakan operator logika \wedge (AND). Hasil dari formula proposisi adalah nilai logika \top (*true*) atau \perp (*false*). Formula proposisi yang diambil yaitu :

$$\emptyset = \diamond p(0, s_{14}) \wedge \diamond p(0, s_{16}) \wedge \diamond p(0, s_{15}) \wedge \diamond p(0, s_{13}) \wedge \diamond p(0, s_0)$$

Trajectory $s_0-s_{14}-s_{16}-s_{15}-s_{13}-s_0$

Posisi *Start* dan *Finish Mobile robot* sesuai dengan persyaratan KRCI 2007, bahwa *mobile robot* harus dapat *start* maupun *finish* di posisi s_0 . *mobile robot* harus mengunjungi (*visiting*) ke-empat ruangan (s_{13} , s_{14} , s_{15} , dan s_{16}), di mana setiap ruangan dikunjungi tepat 1 (satu) kali, dan berakhir (*finish*) di tempat yang sama dengan tempat keberangkatan (misal s_0). Sehingga akan terbentuk *robot trajectory* yaitu $s_0, s_{13}, s_{14}, s_{15}$,

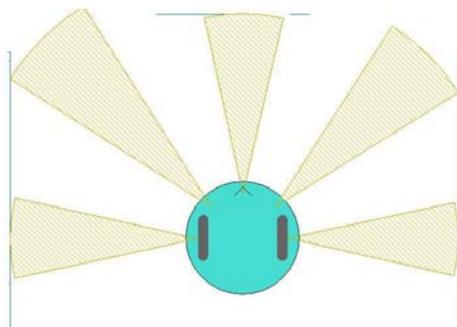
S₁₆, S₀. Pergerakan robot yang benar yaitu robot mampu menginterpretasikan posisinya dengan benar dan menghindari halangan (*obstacle avoidance*). Selain itu robot dapat mencari cahaya lilin dan memaatikannya.

4.3 Perancangan dan Impelementasi Model *Mobile robot* untuk Simulasi dan Prototipe *Mobile robot*

Perancangan simulasi dituangkan dalam pemrograman simulator dengan menggunakan *MOBILE ROBOTSIM* versi 1.0.03. Untuk perancangan simulasi *mobile robot*, aktuator dikendalikan langsung melalui perilaku roda kiri dan kanan serta pengaruhnya terhadap gerakan. Sedangkan pada perancangan kerangka fisik prototipe *mobile robot*, aktuator diwujudkan dengan menggunakan 2 (dua) buah motor DC yang dihubungkan dengan sebuah *driver h-bridge* sebagai perantara motor DC dengan sistem kendali (*mikrokontroler*). Logika pengaturan aktuator pada prototipe *mobile robot* dapat diuraikan seperti dibawah ini.

fungsi maju
Soket 1 (satu) pada motor DC 1 diberi pulsa 1 oleh mikrokontroler
Soket 2 (dua) pada motor DC 2 diberi pulsa 1 oleh mikrokontroler
Soket lainnya diberi pulsa 0
fungsi belok kiri
Soket 2 (dua) pada motor DC 1 diberi pulsa 1 oleh mikrokontroler
Soket 2 (dua) pada motor DC 2 diberi pulsa 1 oleh mikrokontroler
Soket lainnya diberi pulsa 0
fungsi belok kanan
Soket 1 (satu) pada motor DC 1 diberi pulsa 1 oleh mikrokontroler
Soket 1 (satu) pada motor DC 2 diberi pulsa 1 oleh mikrokontroler
Soket lainnya diberi pulsa 0

Rancangan kerangka tampak atas robot ini berbentuk lingkaran, dan rancangan kerangka robot yang sebenarnya berbentuk segi 6, desain kerangka ini mempengaruhi penempatan sensor yang optimal. Dengan penempatan sensor yang optimal, maka abstraksi lingkungan robot berdasarkan penerimaan sensor tersebut akan optimal pula. Rancangan kerangka robot simulasi dapat dilihat pada Gambar 3 dan desain kerangka robot sebenarnya dapat dilihat pada Gambar 4.



Gambar 3 Kerangka robot simulasi tampak atas.



Gambar 5 Kerangka robot tampak atas .

MOBOTSIM telah menyediakan sensor jarak sebagai simulasi dari sensor. Sensor jarak akan mendeteksi penghalang serta jarak antar sensor dengan penghalang tersebut. Sedangkan untuk prototype *mobile robot*, dipakai sensor PING *ultrasonic* untuk mendeteksi penghalang serta jarak penghalang tersebut dengan *mobile robot*. Jarak sensor adalah 3 cm – 3 m. Slain sensor PING *ultrasonic* dipakai juga sensor fototransistor dan fotodiode. Fototransistor dipakai untuk mendeteksi adanya cahaya pada sebuah ruangan tertentu pada lingkungan robot, sedangkan fotodiode dipakai untuk mendeteksi garis pada sebuah ruangan yang akan memberikan tanda jika robot tersebut sudah berada pada ruangan tertentu pada lingkungan robot. Tabel 1 menjelaskan kerarakteristik robot simulasi dan Tabel 2 menjelaskan karakteristik protipe robot sebenarnya.

Tabel 1 Karakteristik Robot Simulasi.

Karakteristik	Keterangan
Bentuk Kerangka	Bentuk kerangka berupa lingkaran (tampak atas)
Jenis sensor	Berupa sensor pengukur jarak (<i>range sensor</i>)
Jumlah sensor	5 buah (sensor0, sensor1, ..., sensor4)
Penjelasan penempatan posisi sensor	s0 adalah sensor kanan, berturut-turut berlawanan arah jarum jam, s2 adalah sensor depan dan s4 adalah sensor kiri
Sudut antara 2 sensor	45 ⁰ (45 derajat)
Jarak / <i>Range Sensor</i>	Secara skala adalah 11 meter atau 80 % lebar hallway

Tabel 2 Karakteristik Prototipe Robot Sebenarnya.

Karakteristik	Keterangan
Bentuk kerangka	Bentuk kerangka berupa segi 6 (enam) (tampak atas) terdiri dari 2 lantai, lantai dasar berisi motor DC, <i>driver power supply</i> , <i>h-bridge</i> , <i>fotodiode</i> , sumber tegangan. lantai 2 berisi mikrokontroler, motor servo, dinamo kipas, <i>fototransistor</i> , kompas digital. sensor PING ultrasonik.
Jenis sensor	Sensor PING ultrasonik, sensor garis <i>fotodiode</i> , sensor cahaya <i>fototransistor</i> .
Jumlah sensor	5 buah sensor PING ultrasonik, 3 buah sensor <i>fototransistor</i> , 2 buah sensor <i>fotodiode</i> .
Penjelasan penempatan posisi sensor	s0 adalah sensor kanan PING ultrasonik berturut-turut berlawanan arah jarum jam, s2 adalah sensor depan PING ultrasonik, dan s4 adalah sensor kiri PING ultrasonik, FT1,FT2,FT3 adalah <i>fototransistor</i> berada pada sebelah dinamo kipas GRS1,GRS2 adalah fotodiode berada pada bagian bawah <i>mobile robot</i> .

Untuk kompas, sesuai dengan kompas pada umumnya, derajat arah adalah dalam modulo 360 derajat. pada umumnya sudut 0⁰ (nol derajat) menunjukkan arah utara, tetapi arah utara pada simulasi mengacu pada sudut 90⁰ (90 derajat). Tabel 3 Karakteristik Kompas Robot dan Perbedaan Dengan Kompas Umumnya.

Tabel 3 Karakteristik Kompas Robot dan Perbedaan Dengan Kompas Umumnya.

Karakteristik	Keterangan	
	Kompas Umumnya	Kompas Robot Simulasi
Derajat Arah	Utara : 0 Timur : 90 Selatan : 180 Barat : 270	Utara : 90 Timur : 0 Selatan : 270 Barat : 180
Putaran	Dilihat dari derajat arah di atas maka untuk putaran (belok) ke kiri akan memiliki sudut putar negatif dan sebaliknya untuk putaran (belok) ke kanan memiliki sudut putar positif	Dilihat dari derajat arah di atas maka untuk putaran (belok) ke kiri akan memiliki sudut putar positif dan sebaliknya untuk putaran (belok) ke kanan memiliki sudut putar negatif

Untuk menghubungkan kompas sebenarnya dengan mikrokontroler, digunakan sebuah protokol yang bernama protokol I²C. protokol I²C digunakan untuk memberitahu mikrokontroler posisi arah *mobile robot* melalui SDA (Serial Data). Kodel program berikut adalah fungsi proses pembacaan arah pada *mobile robot* :

```
volatile unsigned char cmps()
{volatile unsigned char hasil=0xff;
  iicstart(); //start
  iicwxd(0xc0); //ambil data
  iicwxd(0x01); //ambil data
  iicstart();
  iicwxd(0xc1); //ambil data
  hasil=iicrxd(); //baca data
  iicstop(); //stop
  return(hasil);
}
```

Karena faktor lingkungan mempengaruhi arah kompas pada *mobile robot*, maka dilakukan perhitungan galat pada setiap arah kompas. Galat ditetapkan pada penelitian ini adalah 5 (lima) derajat sebelum dan sesudah arah kompas.

4.4 Evaluasi hasil Simulasi dan Protipe Mobile Robot

Tujuan simulasi menggunakan MOBOTSIM dan prototipe mobile robot adalah membuktikan bahwa proposisi

$$\emptyset = \diamond p(0, s_{14}) \wedge \diamond p(0, s_{16}) \wedge \diamond p(0, s_{15}) \wedge \diamond p(0, s_{13}) \wedge \diamond p(0, s_0) \text{ bernilai benar.}$$

Trajectory $s_0-s_{14}-s_{16}-s_{15}-s_{13}-s_0$ merupakan urutan rangkaian *state* yang akan dikunjungi oleh simulator *mobile robot* dan prototipe *mobile robot*.

Pada status awal s_0 arah hadap *mobile robot* menuju ke selatan (270^0 pada simulasi dan 180^0 pada robot sebenarnya).

Eksekusi proposisi $\diamond p(0, s_{14})$:

Maju hingga persimpangan, belok ke timur menuju ke depan gerbang status 14, setelah tiba di depan gerbang status 14, belok ke utara hingga tiba di mulut gerbang. Untuk selanjutnya *mobile robot* berjalan mengelilingi ruangan status 14 sebagai hasil eksekusi *visiting* 14. Pada prototipe *mobile robot* sebenarnya, sensor fotodiode menginformasikan bahwa robot itu sudah ada di dalam sebuah ruangan tertentu dan untuk selanjutnya robot akan mengelilingi ruangan dan mencari cahaya lilin. Lilin ini kemudian akan dimatikan oleh dinamo kipas yang terdapat pada robot. Eksekusi ini disebut eksekusi *visiting*.

Eksekusi proposisi $\diamond p(0, s_{16})$:

Mula-mula *mobile robot* di gerbang status 14 menghadap selatan dan telah siap berjalan ke status 16. *Mobile robot* berjalan menghadap selatan keluar dari status 14, menuju *hallway* atau lorong. Bila sensor depan (sensor 3) mendeteksi adanya dinding, maka *mobile robot* telah tiba di *hall way*, selanjutnya *mobile robot* akan belok ke arah barat dan siap berjalan menuju perempatan, *Mobile robot* berjalan lurus menuju perempatan. Bila sensor 4 tidak mendeteksi dinding, maka *mobile robot* telah tiba di perempatan. Selanjutnya *mobile robot* akan belok ke arah selatan. *Mobile robot* berjalan lurus ke arah selatan hingga tiba di depan gerbang status 16, bila sensor 0 mendeteksi adanya dinding maka *mobile robot* sudah berada pada depan gerbang 16. *Mobile robot* belok ke arah timur untuk siap masuk (*visiting*) ke ruangan status 16. Melakukan *visiting* ruang 16.

Eksekusi proposisi $\diamond p(0, s_{15})$:

Posisi *mobile robot* ada di gerbang status 16, dengan arah hadap ke barat dan siap untuk melangkah keluar. *Mobile robot* berjalan lurus ke arah barat (arah 180) ke luar status 16. Bila sensor depan (sensor 3) mendeteksi adanya dinding, maka *mobile robot* telah tiba di *hallway*. *Mobile robot* membelok ke arah selatan (270) untuk menuju status 15. *Mobile robot* berjalan lurus ke selatan hingga sensor depan mendeteksi adanya dinding, yang menandakan *mobile robot* telah tiba di depan gerbang status 15. *Mobile robot* belok ke arah barat (arah 180) untuk siap masuk ke ruang status 15. Melakukan *visiting* status 15.

Eksekusi Proposisi $\diamond p(0, s_{13})$:

Posisi *mobile robot* berada di gerbang status 15 menghadap ke arah timur dan siap keluar status 15. *Mobile robot* berjalan lurus ke timur hingga ke *hallway*, bila sensor depan (sensor 3) mendeteksi adanya dinding, maka *mobile robot* telah sampai di *hall way*. *Mobile robot* belok ke arah utara. *Mobile robot* berjalan lurus menuju perempatan, bila sensor 0 dan sensor 4 tidak mendeteksi dinding, maka *mobile robot* telah sampai di perempatan. *Mobile robot* belok ke arah barat untuk siap menuju ke status 13. *Mobile robot* berjalan lurus ke timur hingga tiba di depan gerbang status 13, bila sensor depan mendeteksi adanya dinding, kurang dari 15 cm maka *mobile robot* telah tiba di gerbang 13. *Mobile robot* belok ke utara dan segera mengunjungi status 13 (*visiting* 13)..

Eksekusi Proposisi $\diamond p(0, s_0)$:

Mobile robot berada di gerbang status 13 menghadap arah selatan dan siap keluar status 13. *Mobile robot* berjalan lurus ke selatan menuju *hallway*. Bila sensor depan mendeteksi adanya dinding, maka *mobile robot* telah tiba di *hall way*. *Mobile robot* belok ke arah timur. *Mobile robot* berjalan menuju perempatan, bila sensor kiri (sensor 0) dan sensor 4 tidak mendeteksi dinding, maka telah tiba di perempatan. *Mobile robot* belok ke arah utara menuju status 0 (*home*). *Mobile robot* berjalan lurus ke utara, hingga tiba di home, bila sensor depan (sensor 3) mendeteksi dinding, maka *mobile robot* telah tiba di home. *Mobile robot* mengembalikan arah hadapnya sebagaimana arah hadap saat start yaitu ke arah selatan dengan cara putar (balik) kanan. Dengan demikian formula proposisi $\emptyset = \diamond(p(0, s_{14}) \wedge \diamond(p(0, s_{16}) \wedge \diamond(p(0, s_{15}) \wedge \diamond(p(0, s_{13}) \wedge \diamond p(0, s_0))))))$ dapat dibuktikan kebenarannya.

5. KESIMPULAN

Implementasi formulasi dengan simulasi menggunakan MOBOTSIM dan Prototipe mobile robot menunjukkan bahwa spesifikasi yang diformulasikan menggunakan Logika Temporal Linier dapat dibuktikan

kebenarannya. Selain itu, dengan adanya formulasi navigasi *Mobile robot* implementasi pembangunan prototipenya menjadi lebih mudah dan akurat.

Namun demikian implementasi formulasi dengan menggunakan simulasi program hasilnya lebih ideal dari pada implementasi formulasi dengan menggunakan prototipe *mobile robot* sebenarnya. Hal ini disebabkan pergerakan protipe *mobile robot* banyak dipengaruhi faktor luar, seperti pengaruh gesekan dengan lantai, *kekutan power supply*, dan lainnya yang dalam penelitian ini diabaikan.

6. DAFTAR PUSTAKA

- Clarke, Grumberg and Peled: "*Model Checking*", page 14. The MIT Press, 1999.
- Pressman.2002. *Software Engineering : Practitional Aproach 6-rd edition*.New York.Sommerville
- Schilling, J Rober. 2000.*Fundamentals of Robotics Analysis and Control*. NJ.Prentice Hall
- Sihite Liston & Wardoyo Retantyo 2005. *Pemodelan Logika Pemeriksaan Temporal Linier dengan Buchi Automata*. Yogyakarta. ILKOM UGM.
- Supriyo, Tri Prapto.2003. *Model Formal Sistem Pengendali Lalulintas Kereta Api di suatu Statsiun*.Jakarta. Fusilkom UI
- Widiyanto, D., Supriyo, P.T, Kusuma, W. A .2008. *Formalisasi Navigasi Mobile robot*. Prosiding Seminar Nasional Teknologi dan Rekayasa Industri. Fakultas Teknologi Industri, Institut Teknologi Indonesia.
- Widodo Edi & Prasetia Retna 2004. *Interfacing prot parallel dan port serial komputer dengan VB 6.0*.Semarang.ANDI.