

KENDALI POSISI *MOBILE ROBOT* MENGGUNAKAN SISTEM *PROPORTIONAL INTEGRAL DERIVATIVE (PID)* DENGAN METODE *ODOMETRY*

Rijalul Haq¹⁾, Endah Rahmawati²⁾

¹⁾ Mahasiswa Prodi S1-Fisika, FMIPA, Universitas Negeri Surabaya, email : rijalulhaq@mhs.unesa.ac.id

²⁾ Dosen Fisika, FMIPA, Universitas Negeri Surabaya, email : endahrahmawati@unesa.ac.id

Abstrak

Penelitian ini bertujuan untuk konsep fisika untuk kendali posisi menggunakan metode odometry dan kendali PID pada *mobile robot*. Sebuah *mobile robot* telah dibuat dengan model kinematika *differential drive* dengan *sensor magnetic rotary encoder* yang akan menghitung jumlah putaran roda *mobile robot* yang nantinya akan digunakan untuk menghitung posisi *mobile robot*. *Odometry* adalah penggunaan data dari pergerakan aktuator untuk memperkirakan perubahan posisi dari waktu ke waktu. Berdasarkan hasil kalibrasi sensor *rotary encoder* diperoleh nilai kesalahan maksimal sebesar 1,33% untuk roda kanan dan 1,06% untuk roda kiri. Selanjutnya menguji *mobile robot* dengan cara memberikan input berupa koordinat ($X=710$ mm, $Y= 1220$ mm) dan *heading* (θ) = 30° . Pada pengujian diperoleh hasil bahwa; (1) saat $KP=10$ $KI=2$ $KD=0$ sistem tidak dapat langsung menuju set poin dan berosilasi, (2) saat $KP=10$ $KI=0$ $KD=10$ sistem cenderung lebih stabil pada set poin, dan (3) tidak berosilasi dan saat $KP=15$ $KI=0$ $KD=0$ sistem lebih cepat untuk menuju set poin walaupun setelah itu sedikit berosilasi. Dari ketiga konstanta PID tersebut, $KP=10$ $KI=0$ $KD=10$ memiliki respon paling baik.

Kata Kunci : *Mobile robot*, PID, *Odometry*.

Abstract

The purpose of this study is to know the application of physical concept for position control of mobile robot and to know the influence of PID constant for mobile robot position control. This study was conducted by making a mobile robot with a differential drive kinematic model which magnetic rotary encoder sensor was installed to the mobile robot to calculate the amount of rotation of mobile robot wheels that would be used to determine the position of mobile robot. *Odometry* is the use of data from the actuator movement to estimate the change of position over time. The first step in this study was calibrating the sensor. Based on sensor calibration result obtained the maximum error amount 1,33% for right wheel and 1,06% for left wheel. The next step was testing the mobile robot by giving input for mobile robot as destination coordinate ($X=710$ mm, $Y= 1220$ mm) and heading (θ) = 30° . The results show as follows; (1) for $KP=10$ $KI=2$ $KD=0$ system oscillates and can not directly go to set point, (2) for $KP=10$ $KI=0$ $KD=10$ system does not oscillate and tended stable on set point, and (3) for $KP=15$ $KI=0$ $KD=0$ after slight oscillation the system get the set point quicker. From the three PID constants, the $KP=10$ $KI=0$ $KD=10$ has the best response.

Key word : Mobile robot, PID, odometry

PENDAHULUAN

Robot adalah seperangkat alat mekanik yang mampu melakukan tugas fisik, baik dengan pengawasan manusia maupun dengan kecerdasan buatan. Pada awalnya robot diciptakan untuk membantu manusia dalam melakukan pekerjaan yang berat seperti proses produksi pada suatu industri. Namun seiring berkembangnya ilmu pengetahuan dan teknologi, bidang ilmu robotika juga terus berkembang. Robot tidak hanya dimanfaatkan dalam bidang industri tetapi hampir semua aspek dalam kehidupan manusia, seperti dalam bidang kedokteran, pendidikan, dan lain - lain.

Fisika merupakan ilmu pengetahuan yang mempelajari sifat dan gejala yang terjadi pada benda – benda di alam. Salah satu konsep fisika adalah kinematika yang mempelajari tentang gerak dan segala aspek yang berkaitan dengan gerak. Kinematika adalah cabang ilmu fisika yang membahas tentang gerak tanpa memperhatikan penyebabnya yang didalamnya dikaji tentang posisi, jarak, perpindahan, kecepatan, kelajuan, dll. Selain itu Jurusan Fisika UNESA mempunyai konsentrasi bidang ilmu instrumentasi yang mengkaji tentang elektronika dan dasar – dasar sistem kontrol termasuk robotika.

Dalam dunia *autonomous mobile robot*, robot telah dikembangkan agar mampu bergerak secara otomatis tanpa kontrol dari manusia. Metode yang sering digunakan untuk membuat *mobile robot* adalah *line follower*, di mana robot akan bergerak mengikuti lintasan berupa garis. Namun metode ini dianggap kurang efektif karena area gerak sangat terbatas. Metode lain yang dapat digunakan untuk membuat *mobile robot* adalah *odometry*.

Odometry adalah penggunaan data dari pergerakan aktuator untuk memperkirakan perubahan posisi dari waktu ke waktu. *Odometry* digunakan untuk memperkirakan posisi relatif terhadap posisi awal (Sandi, 2012). Penelitian tentang *mobile robot* dengan metode *odometry* telah dilakukan oleh beberapa peneliti (Ardila, 2011; Sandi, 2012). Pada penelitian Ardila (2011) *mobile robot* menggunakan sensor *rotary encoder* untuk menentukan posisi relatif robot dan target yang akan dicapai. Hasil penelitian tersebut menunjukkan bahwa metode *odometry* dapat digunakan untuk mengontrol posisi *mobile robot*.

Penelitian Sandi (2012) berupa pengujian *mobile robot* untuk mencari koordinat pada *path* dalam suatu bidang datar dengan input berupa koordinat x dan y, hasil penelitian ini menunjukkan bahwa *odometry* dapat bekerja secara efektif pada bidang datar.

Penelitian ini bertujuan untuk mengetahui penerapan kinematika untuk kendali posisi *mobile robot* dengan metode *odometry* dan membahas pengaruh konstanta PID pada kendali posisi *mobile robot*.

METODE PENELITIAN

Salah satu kendali sistem penggerak *mobile robot* yang umum digunakan adalah pengendali differensial. Sistem penggerak ini terdiri dari dua buah roda yang terpasang dalam satu sumbu dan masing – masing roda secara independen dapat dikontrol untuk berputar baik searah jarum jam ataupun berlawanan arah jarum jam.

Odometry adalah metode yang paling sering digunakan untuk menentukan posisi *mobile robot* secara *real time*. *Odometry* adalah penggunaan data dari pergerakan aktuator untuk memperkirakan perubahan posisi dari waktu ke waktu. *Odometry* digunakan untuk memperkirakan posisi relatif terhadap posisi awal (Sandi, 2012).

Untuk memperkirakan posisi relatif robot, diperlukan konstanta peubah yang mampu mengkonversi jumlah *counter* yang dihasilkan oleh sensor *rotary encoder* menjadi panjang lintasan yang ditempuh robot. Konstanta ini didapatkan menggunakan persamaan berikut:

$$C = \frac{K \text{ roda}}{\text{resolusi encoder}} \quad (1)$$

Dimana: C adalah konstanta peubah (mm/counter). $K \text{ roda}$ adalah keliling roda yang digunakan

(mm) dan *resolusi encoder* adalah jumlah counter dalam satu putaran

Jika jarak tempuh roda kanan dan roda kiri masing - masing dimisalkan dengan d_{right} dan d_{left} dan jarak antara kedua roda dimisalkan dengan l , maka jarak tempuh *distance* dan sudut orientasi θ dapat dirumuskan dengan persamaan berikut:

$$d_{right} = \text{counter}_{right} \times C \quad (2)$$

$$d_{left} = \text{counter}_{left} \times C \quad (3)$$

$$\text{distance} = \frac{(d_{right} + d_{left})}{2} \quad (4)$$

$$\theta = \frac{d_{right} - d_{left}}{l} \quad (5)$$

Karena θ masih dalam radian maka untuk mengetahui sudut hadap robot (*heading*) dalam derajat digunakan persamaan

$$\text{heading}(\theta) = \theta \times \frac{180}{\pi} \quad (6)$$

Dengan mengetahui jarak tempuh (*distance*) dan sudut orientasi θ maka dapat ditentukan koordinat posisi robot dengan menggunakan persamaan trigonometri

$$X_{pos} = \text{distance} \cdot \sin(\theta) \quad (7)$$

$$Y_{pos} = \text{distance} \cdot \cos(\theta) \quad (8)$$

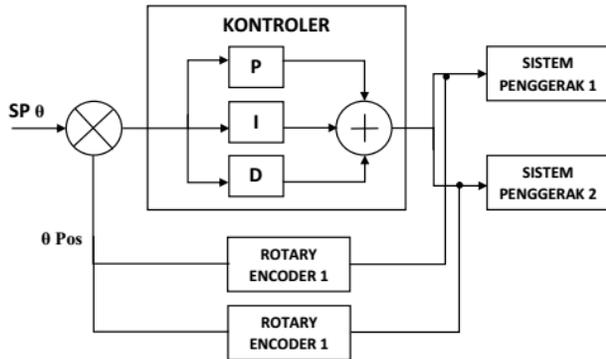
Instrument dalam penelitian ini secara umum terbagi dalam dua bagian, yaitu: perangkat keras dan perangkat lunak. Dalam hal ini perangkat keras yang dimaksud adalah prototype *mobile robot* dengan sensor *magnetic rotary encoder* dan mikrokontroler ATMega 32A serta sistem penggerak berupa motor DC dan roda. Sedangkan perangkat lunak adalah aplikasi (software) CoedvisionAVR yang digunakan untuk membangun logika pemrograman yang selanjutnya akan ditransfer ke mikrokontroler ATMega 32A.

Tahap pertama dari perancangan perangkat keras (protoipe) adalah membuat mikrokontroler *board* yang akan menjadi tempat mikrokontroler ATMega 32A dan komponen elektronika lainnya. Mikrokontroler *board* ini didesain dengan aplikasi *Proteus 8 Profesional* dan kemudian dicetak pada PCB (*Printed Circuit Board*)

Prototipe *mobile robot* ini menggunakan dua buah motor DC sebagai alat geraknya. Motor DC ini terpasang bersebrangan di bagian belakang robot. Perbandingan kecepatan dari kedua motor DC ini yang akan menentukan arah gerak robot. Saat motor kanan dan motor kiri berputar dengan kecepatan yang sama maka robot akan berjalan lurus. Saat kecepatan motor kanan lebih besar dari pada kecepatan motor kiri maka robot akan cenderung berbelok ke kiri. Dan saat kecepatan motor kanan lebih kecil dari kecepatan motor kiri maka robot akan cenderung berbelok ke kanan. Motor tersebut terhubung langsung dengan *rotary encoder*, sehingga saat motor berputar maka *rotary*

encoder juga akan berputar dan posisi robot dapat diketahui.

Selanjutnya adalah perancangan perangkat lunak yang dimulai dengan membangun algoritma atau logika pemrograman sebagai sistem kerja *mobile robot* dalam mencapai target. Algoritma tersebut disajikan dalam diagram alir berikut:



Gambar 1. Diagram blok sistem kontrol *mobile robot*

Target berupa koordinat (x,y) dimasukkan pada mikrokontroler dan akan disimpan pada EEPROM. Dari data (x,y) mikrokontroler akan menghitung sudut (θ) tujuan yang kemudian dianggap sebagai set poin, kemudian mengaktifkan sistem penggerak yang terhubung dengan sensor *rotary encoder*. Data dari *rotary encoder* akan diumpan balik ke mikrokontroler dan dikonversi menjadi data posisi berupa (Xpos, Ypos, θ pos). Data tersebut akan dibandingkan dengan set poin sehingga didapatkan error. Error tersebut akan dimasukkan ke dalam persamaan PID pada mikrokontroler yang kemudian akan berdampak pada bertambah atau berkurangnya kecepatan sistem penggerak.

Pada penelitian ini variabel manipulasinya adalah nilai koordinat input yaitu berupa koordinat (x,y) yang nantinya akan dituju oleh *mobile robot* dan nilai konstanta PID yaitu K_p , K_i , dan K_d . Variabel kontrol yang digunakan adalah diameter roda dan jarak antara kedua roda *mobile robot*. Variabel respon adalah nilai yang akan diperoleh pada suatu penelitian. Variable respon pada penelitian ini berupa koordinat (x',y', θ) yang telah dilalui *mobile robot*.

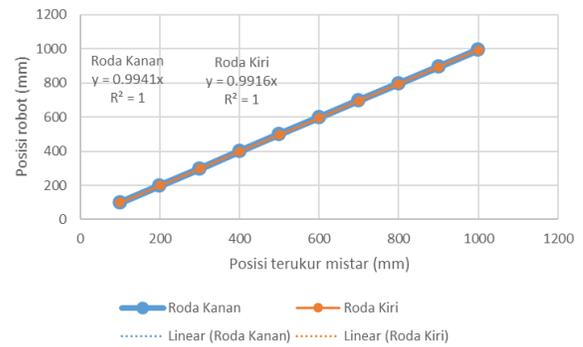
Proses pengambilan data diawali dengan kalibrasi sensor *rotary encoder* dengan menggunakan persamaan (1). Kemudian memberikan input pada *mobile robot* berupa koordinat (x,y). Kemudian *mobile robot* akan bergerak menuju koordinat tersebut secara otomatis. Saat *mobile robot* telah sampai pada koordinat yang dituju maka *mobile robot* akan berhenti dan menampilkan data berupa koordinat (x',y').

HASIL DAN PEMBAHASAN

Pengujian sensor *rotary encoder* dilakukan dengan cara menggerakkan *mobile robot* dengan jarak 10

cm sampai 100 cm dan membandingkan nilai jarak yang terbaca oleh sensor dengan jarak sebenarnya yang terukur dengan mistar kemudian diulangi sebanyak 3 kali pengukuran untuk setiap jarak tempuh *mobile robot*. Pengukuran ini bertujuan untuk menguji keakuratan kepresisian sensor *rotary encoder*

Gambar 2 grafik perbandingan jarak yang terukur dengan mistar dan posisi *mobile robot*



Berdasarkan gambar 2 grafik roda kanan dan roda kiri saling berimpit dan hampir tidak terlihat perbedaannya. Hal ini menunjukkan bahwa sensor *rotary encoder* yang digunakan identik dan memiliki akurasi yang baik.

Selanjutnya Pengujian sistem kendali *mobile robot* dilakukan pada suatu bidang datar yang telah diberi pola digram kartesius untuk mempermudah mengamati posisi *mobile robot*. Mula - mula *mobile robot* diletakan pada titik pusat digram kartesius dengan bagian depan robot sejajar dengan sumbu Y, kemudian peneliti memberikan masukan data berupa koordinat X dan Y yang kan menjadi tujuan akhir *mobile robot*. Dari data ini kontroler akan menghitung sudut (*heading robot*) dan jarak yang akan ditempuh *mobile robot*, kemudian *mobile robot* akan bergerak menuju koordinat (X,Y) tersebut secara otomatis. Dengan memanipulasi konstanta PID, didapatkan hasil sebagai berikut



Gambar 3. Grafik perubahan *heading error* dengan *heading 30°*

Heading eror merupakan kesalahan hadap *mobile robot* yang terjadi, yaitu selisih antara sudut hadap robot pada waktu tersebut dengan heading robot sebagai set poin.

Gambar di atas menunjukkan perubahan nilai heading error tiap 50 ms saat mobile robot mulai bergerak sampai berhenti. Pada saat $KP=0$ $KI=0$ $KD=0$ (tanpa sistem kontrol) nilai heading error berubah secara acak dan tidak dapat mencapai nilai nol ataupun bertahan disekitar nol. Hal ini terjadi karena mobile robot bergerak acak dan tidak dapat mencapai set poin. Sedangkan saat sistem kontrol diaplikasikan pada mobiler robot, awalnya heading error berada pada nilai maskimal yaitu 30 dan terus turun sampai mendekati nilai nol atau melewati nilai nol. Setelah itu nilai haeding eror berada disekitar nol. Hal ini menunjukkan bahwa sistem kendali ini akan berusaha mempertahankan nilai heading robot sesuai set poin sehingga heading error bernilai nol.

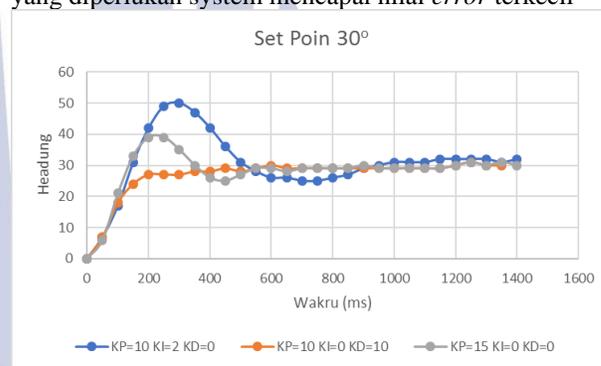
Tabel 1 data posisi akhir *mobile* robot dengan heading 30°

PID	POSISI AKHIR			KESALAHAN		
	X' (mm)	Y' (mm)	θ'	eX (%)	eY (%)	e θ (%)
KP=0 KI=0 KD=0	412	1340	6	41.97	9.84	6.67
	73	1426	-9	89.72	16.89	10.83
	91	1406	0	87.18	15.25	8.33
	51	1403	-3	92.82	15.00	9.17
	-252	1378	-18	135.49	12.95	13.33
KP=10 KI=2 KD=0	734	1281	33	3.38	5.00	0.83
	763	1272	31	7.46	4.26	0.28
	767	1279	33	8.03	4.84	0.83
	801	1271	32	12.82	4.18	0.56
	754	1252	31	6.20	2.62	0.28
KP=10 KI=0 KD=10	717	1331	30	0.99	9.10	0.00
	718	1330	30	1.13	9.02	0.00
	701	1303	30	1.27	6.80	0.00
	727	1323	30	2.39	8.44	0.00
	725	1332	31	2.11	9.18	0.28
KP=15 KI=0 KD=0	761	1312	30	7.18	7.54	0.00
	766	1308	31	7.89	7.21	0.28
	762	1305	30	7.32	6.97	0.00
	753	1316	30	6.06	7.87	0.00
	773	1318	31	8.87	8.03	0.28

Set poin yang digunakan adalah $X=710$ mm $Y=1220$ mm dan heading (θ) = 30°. Posisi akhir robot adalah posisi saat robot telah berhenti dan tidak mengalami pergerakan lagi. Kesalahan adalah persentase selisih posisi *mobile* robot terhadap set poin, Dengan nilai konstanta PID yang berbeda didapatkan nilai posisi akhir robot yang berbeda pula. Dari tabel 1 terlihat nilai kesalahan yang berbeda -

beda pada tiap konstanta PID. Saat $KP=0$ $KI=0$ $KD=0$ adalah saat dimana *mobile* robot bergerak tanpa adanya sistem kontrol. Hal ini mengakibatkan *mobile* robot bergeak acak dan tidak dapat mencapai tujuan yang telah ditentukan, hal ini terlihat pada posisi akhir robot yang sangat jauh dari target dengan nilai kesalahan mencapai 135,49% untuk X, 16,89 % untuk Y dan 13,33% untuk θ . Sedangkan pada saat sistem kontrol diapliksaikan pada *mobile* robot, nilai rata – rata kesalahan terkecil untuk X terjadi saat $KP=10$ $KI=0$ $KD=10$ yaitu 1,58% , dan terbesar terjadi saat $KP=10$ $KI=2$ $KD=0$ yaitu 7,58%. Sedangkan nilai rata – rata kesalah terkecil untuk Y terjadi saat $KP=10$ $KI=2$ $KD=0$ yaitu 4,18% dan terbesar saat $KP=10$ $KI=0$ $KD=10$ yaitu 8,51%. Dan untuk θ nilai rata – rata kesalahan terkecil terjadi saat $KP=10$ $KI=0$ $KD=10$ yaitu 0,06% , dan terbesar terjadi saat $KP=10$ $KI=2$ $KD=0$ yaitu 0,56%.

Pada umumnya respon dari suatu sistem kontrol diukur berdasarkan karakteristik bentuk gelombang respon. Rising time adalah waktu yang diperlukan sistem melewati 90% nilai set point, *settling time* adalah waktu yang diperlukan system mencapai nilai *error* terkecil



Gambar 4 grafik respon PID untuk sudut 30°

Pada gambar di atas terlihat perbandingan respon perubahan heading robot terhadap waktu untuk tiap PID. Dengan konstanta PID yang berbeda maka didapatkan hasil respon yang berbeda juga. Pada awalnya nilai heading adalah nol dan akan naik menuju set poin yang telah ditentukan. Hal ini menggambarkan *mobile* robot yang awalnya menghadap sejajar sumbu y dan kemudian secara perlahan akan menuju set poin. Terlihat bahwa sistem bekerja cukup cepat dengan rising time kurang dari 200 ms untuk semua nilai konstanta PID. Rising time ini dipengaruhi oleh besarnya KP, semakin besar nilai KP maka rising time akan semakin cepat. Pada gambar 4 rising time yang ditunjukkan kurva dengan $KP=10$ dan $KP=15$ hampir sama, hal ini dikarenakan perbedaan KP yang terlalu kecil dan rentang waktu pengambilan data selama 50 ms sehingga tidak mampu mengetahui perubahan heading antara waktu tersebut.

Setelah mencapai heading yang sesuai dengan set poin, sistem akan mengalami *overshoot*. Besarnya nilai *overshoot* dipengaruhi oleh nilai KP, namun pada nilai KP

yang sama nilai KI dan KD juga mempengaruhi. Terlihat pada gambar 4 *overshoot* terbesar dicapai saat $KP=10$ $KI=2$ $KD=0$.

Waktu yang dibutuhkan untuk mencapai *error* terkecil berbeda – beda untuk setiap konstanta PID. *Settling time* ini dipengaruhi oleh KI dan KD. Kombinasi KI dan KD yang tepat akan menghasilkan respon sistem yang stabil. Pada set poin 30° dengan $KI=2$ $KD=0$ diperoleh *settling time* 900 ms, saat $KI=0$ $KD=10$ diperoleh *settling time* yang cukup cepat yaitu 400 ms, sedangkan saat $KI=0$ $KD=0$ diperoleh *settling time* sebesar 600 ms.

PENUTUP

Simpulan

Mobile robot yang dirancang pada penelitian ini menggunakan model kinematik differential drive dengan sensor rotary encoder yang terpasang pada motor penggerak. Mobile robot memanfaatkan data input berupa koordinat X dan Y kemudian memproses data dengan pertinguan vektor sehingga didapatkan sudut, jarak dan koordinat posisi mobile robot. Sistem kendali posisi mobile robot mengarahkan mobile robot menuju set poin yang ditentukan. Saat mobile robot bergerak tanpa sistem kendali posisi, mobile robot akan bergerak acak dan tidak dapat mencapai set poin dengan *error* terbesar 135,49% untuk heading 30° . Sedangkan pada saat menggunakan sistem kendali posisi diperoleh *error* terbesar 12,82% untuk heading 30° .

Sistem kontrol mobile robot akan memberikan respon yang berbeda untuk nilai konstanta PID yang berbeda. Komponen KP mempercepat sistem menuju set poin. Komponen KI memungkinkan sistem untuk mengakumulasi kesalahan yang terjadi, namun pemilihan KI yang tidak sesuai akan menyebabkan sistem berosilasi secara berlebihan dan membutuhkan waktu lama untuk mencapai kesetabilan. Komponen KD berpengaruh pada kecepatan perubahan sinyal output sistem. Dengan adanya KD sistem akan lebih stabil dan mencegah sistem berosilasi

Saran

Berdasarkan penelitian yang telah dilakukan, mobile robot dapat bergerak sesuai koordinat yang telah ditentukan namun kadang tidak dapat berhenti tepat pada koordinat tersebut. Hal ini disebabkan karena terjadi slip pada rotary dan juga roda robot. Hal ini dapat diatasi dengan menggunakan roda dari bahan karet yang lebih kuat dan juga dapat menggunakan sensor rotary encoder yang terpisah dari motor sehingga saat motor slip, sensor rotary tidak slip.

DAFTAR PUSTAKA

- li, M. 2004. Pembelajaran perancangan sistem kontrol PID dengan software matlab. Jurnal Edukasi@Elektro. Vol 1. No.1 pp 1-8.
- Ang, K.H and Chong, G.C.Y., and Li, Y. 2005. PID Control system analysis, design and technology. IEEE Transaction on Control Systems Technology 13. Vol 4 pp 559-576.
- Ardilla, F., Marta, B.S., Besari, A.R.A. 2011. Path tracking mobile robot dengan umpan balik odometry. The 13th Industria Electronic Seminar 2011. ISBN: 978-979-8689-14-7.
- David, A. 2006. IMU Odometry. Article, Oktober 2006. Dallas: Texas.
- Dudek, G., Jenkin, M. 2010. Computational principles of mobile robotics second edition. Cambridge University Press: New York.
- Kariyantp, J.D., Alasiry, A.H., Ardilla, A. dan Hanafi, N. 2012. Navigasi mobile robot berbasis trajectory dan odometry dengan pemulihan jalur secara otomatis. Paper. Teknik Elektro PENS
- Martinelli, A. 2002. The odometry error of a mobile robot with a synchronous Drive System. IEEE Transactions on Robotics and Automation Vol. 18 No. 8 pp 1-8
- Villarreal, G. M and Alvarez, J. 2016. Off-line PID tuning for planar parallel robot using DE variants. Expert System and Application. Vol 64, pp 444-454.
- Williamson, T.A. 2007. Modeling and implementation of PID control for autonomus robots. Thesis Master of Science Applied Physic. Naval Postgraduat School.