

## IMPLEMENTASI INVERSE KINEMATICS TERHADAP POLA GERAK HEXAPOD ROBOT 2 DOF

Selamat Muslimin<sup>1</sup>, Kharis Salahuddin<sup>2</sup>, Ekawati Prihatini<sup>3</sup>

### ABSTRAK

Robot adalah alat manipulator yang dirancang dan diprogram menjadi robot. Salah satunya dirancang robot bangun 2 DOF (Gelar Kebebasan) yang diimplementasikan melawan robot berkaki. Dalam hal ini robot hexapod yang terdiri dari dua derajat kebebasan melakukan gerakan kaki robot yang fleksibel. Hal ini berdasarkan sudut pandang motor servo dengan penerapan invers kinematics sebagai analisis sudut servo. Desain robot ini memecahkan metode dasar geometri analitik melalui trigonometri turunan. Metode ini digunakan agar robot menghasilkan output setiap gerakan kaki dengan mengubah servo sudut yang telah dianalisis melalui algoritma kinematika invers pada kaki robot. Keseluruhan total Kaki robot adalah 12 kaki dengan 12 derajat derajat kebebasan.

**Kata Kunci :** *Inverse Kinematics; DOF; Servo*

### ABSTRACT

The robot is a tool manipulator which is designed and programmed to help complete the work of man. There are different types of robots that can be applied in various sectors of society in areas such as legged and wheeled robots. One of them is designed wake robot 2 DOF (Degree Of Freedom) were implemented against legged robot. In this case a hexapod robot which consists of two degrees of freedom to perform the movement of the robot legs flexible. It is on the terms of the angle produced by the servo motor with the implementation of inverse kinematics as servo angle analysis. This robot design solved the basic methods of analytic geometry through derivative trigonometry. This method is used so that the robot produces good output of each movement of the foot by changing the angle servo which has been analyzed through inverse kinematics algorithms on robotic legs. Overall total The robot's legs are 12 feet by 12 angle degrees of freedom.

**KeyWords :** *Inverse Kinematics; DOF; Servo*

### PENDAHULUAN

Banyak jenis robot salah satunya adalah robot beroda dan berkaki. Robot beroda merupakan jenis robot yang banyak digunakan dalam berbagai aplikasi karena kecepatannya dalam melintasi bidang rata dan kemudahan dalam mendesain serta implementasinya. Namun dalam beberapa kasus sering terdapat kondisi yang tidak ideal untuk robot beroda, seperti kondisi bidang yang tidak rata. Hal ini lah yang melatar belakangi untuk membuat robot berkaki yang mampu bergerak di daerah yang halus ataupun kasar, menghindari, dan melangkah di atas halangan (Bekey G. A., *Autonomous Robot*, 2005). Dalam hal ini robot berkaki yang akan dibahas adalah *hexapod* robot, memiliki 2 derajat kebebasan (DOF) dalam 1 kaki terhadap 6 jumlah total kaki yang dimiliki.

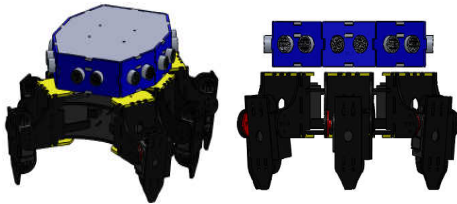
*Hexapod* robot berkaitan erat hubungannya dengan kinematika robot yang

dapat dideskripsikan oleh, kecepatan, akselerasi, dan turunan dari pose badan robot yang menghasilkan mekanisme gerak (Siciliano B, Khatib O ,2008). Berdasarkan kestabilannya, robot berkaki dibedakan menjadi kestabilan dinamis dan statis. Robot berkaki yang memiliki kestabilan dinamis adalah robot dengan titik berat yang tidak mempengaruhi kestabilan robot ketika bergerak. Robot berkaki dengan kestabilan statis adalah robot yang titik beratnya tetap stabil ketika bergerak. Robot yang termasuk dalam jenis ini memiliki jumlah kaki paling sedikit 4 buah, seperti robot hexapod (Thirion B, Thiry L., 2002). Robot berkaki memiliki 2 metode yang dapat diterapkan yaitu, metode *forward* kinematik dan *inverse* kinematik. Metode *forward* kinematik yang sering disebut dengan metode *trial-error*, namun metode ini kurang fleksible dikarenakan jika merubah gerakan robot maka harus melakukan *trial-error* yang lebih banyak sehingga

<sup>1</sup> *Electrical Engineering Department, State Polytechnic of Sriwijaya, Palembang, South Sumatera, Correspondence : selametmuslimin@gmail.com*

menyebabkan waktu pengerjaan lebih lama dan posisi harus diterapkan pada robot yang mengakibatkan banyak menggunakan memori kontroler. Untuk mengantisipasi permasalahan tersebut diterapkan metode *inverse kinematik* yang merupakan metode analisa untuk melakukan transformasi dari ruang *Cartesian* ke ruang sendi robot. Dari penerapan *inverse kinematik* dapat diterapkan perhitungan trigonometri melalui analisis geometri<sup>[4]</sup>.

**Hexapod Robot.** Hexapod robot adalah jenis *mobile robot*, yang bergerak menggunakan 6 kaki. Robot ini di desain menggunakan perangkat lunak Solidworks terlihat pada Gambar.1.



Gambar 1. Desain Mekanik Robot Hexapod

Kelebihan dan kekurangan robot *hexapod*<sup>[4]</sup>:

Kelebihan :

1. Kestabilan posisi robot diatas permukaan lebih baik daripada *biped* dan *quadrapod*.
2. Dapat bergerak diberbagai permukaan, baik kasar maupun halus.

Kekurangan :

1. Memerlukan biaya riset yang cukup tinggi.
2. Pergerakan relatif lambat dikarenakan rumitnya koordinasi *gait* dari setiap servo.
3. Tingkatan kerumitan dalam mengatur pergerakan servo atau joint theta 1 dan theta 2 terhadap *link*.

Robot hexapod dikembangkan melalui tingkah laku hewan yang dinamakan laba-laba, dari pergerakan dan kinematic struktur. Kaki robot *hexapod* ini di desain menggunakan simulasi inverse kinematics 2 DOF (Degrees Of Freedom) pada setiap kakinya yang berjumlah 6. Dimana dalam hal ini 1 kaki terdiri dari 2 joint (theta 1, theta 2) dan 2 link. Ada pun dimensi robot hexapod ini adalah sebagai berikut:

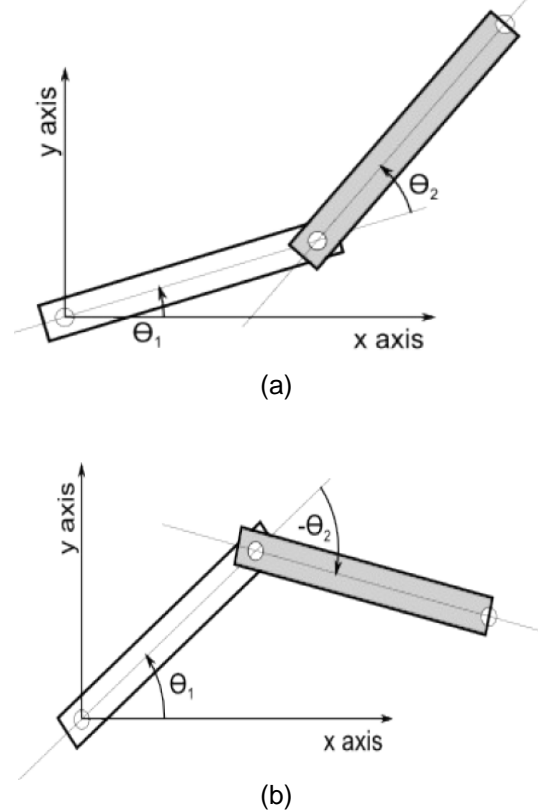
Tabel. 1 Dimensi Robot

Height	108,61 mm
Width	243,43 mm
ink 1	70 mm
Link 2	70 mm

**Inverse Kinematics Problem.** Menentukan posisi dan orientasi *end effector* terhadap

perhitungan dasar semua kemungkinan sudut sendi dan geometri *link* yang bisa digunakan untuk mencapai posisi dan orientasi yang diberikan dari ujung efektor (Reyneri LM., 1997) dan (Haselman, D., Letfield, B, 2001).

Konfigurasi *Inverse Kinematics* 2 Sendi (DOF)



Gambar.2. (a) *Elbow up* (b) *Elbow Down*

**Forwad Kinematics**

$$x = l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2)$$

$$y = l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2)$$

**Inverse Kinematics**

$$x^2 + y^2 = l_1^2 \cos^2 \theta_1 + l_2^2 \cos^2 (\theta_1 + \theta_2) + 2 l_1 l_2 \cos \theta_1 \cos (\theta_1 + \theta_2) + l_1^2 \sin^2 \theta_1 + l_2^2 \sin^2 (\theta_1 + \theta_2) + 2 l_1 l_2 \sin \theta_1 \sin (\theta_1 + \theta_2)$$

$$= l_1^2 + l_2^2 + 2 l_1 l_2 [\cos \theta_1 \cos (\theta_1 + \theta_2) + \sin \theta_1 \sin (\theta_1 + \theta_2)]$$

**Persamaan :**

$$\sin (x \pm y) = \sin x \cos y \pm \cos x \sin y$$

$$\cos (x \pm y) = \cos x \cos y \pm \sin x \sin y$$

$$x^2 + y^2 = l_1^2 + l_2^2 + 2 l_1 l_2 [\cos \theta_1 (\cos \theta_1 \cos \theta_2 - \sin \theta_1 \sin \theta_2) + \sin \theta_1 (\sin \theta_1 \cos \theta_2 + \cos \theta_1 \sin \theta_2)]$$

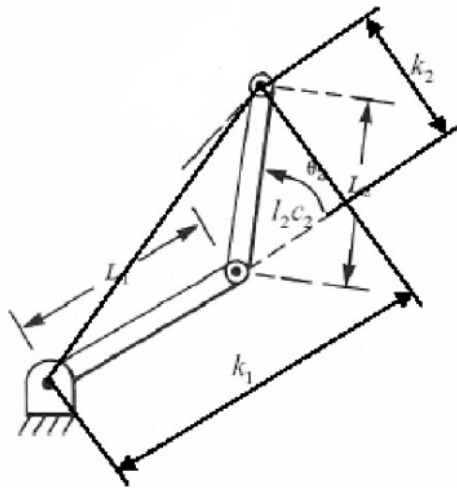
$$\begin{aligned}
 &= l_1^2 + l_2^2 + 2 l_1 l_2 [\cos^2 \theta_1 \cos \theta_2 + \sin^2 \theta_2 \cos \theta_2] \\
 &= l_1^2 + l_2^2 + 2 l_1 l_2 \cos \theta_2 \\
 \cos \theta_2 &= \frac{x^2 + y^2 - l_1^2 - l_2^2}{2 l_1 l_2}
 \end{aligned}$$

Dari sini kita bisa mendapatkan sudut menggunakan fungsi *arc cos*, namun fungsi ini kurang akurat untuk sudut kecil. Cara lain untuk menghindari ketidakakuratan ini adalah dengan mengkonversi terhadap Fungsi *arc tan2* (*atan2*).  
 $\cos^2 \theta_2 + \sin^2 \theta_2 = 1$   
 $\sin \theta_2 = \pm (1 - \cos^2 \theta_2)^{1/2}$

Kedua solusi tersebut sesuai dengan konfigurasi 'Elbow Up' dan 'Elbow Down' seperti yang ditunjukkan pada gambar 1 a) dan b).

$$\begin{aligned}
 \text{Jadi, } \theta_2 &= \text{atan2}(\sin \theta_2 \cos \theta_2) = \\
 &\text{atan2}(\pm (1 - \cos^2 \theta_2)^{1/2}, \cos \theta_2) = \\
 &\text{atan2}\left(\pm \left(1 - \frac{x^2 + y^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right)^{1/2}, \frac{x^2 + y^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right)
 \end{aligned}$$

Untuk Penyelesaian  $\theta_1$  kita menulis ulang persamaan nonlinear menggunakan perubahan variable sebagai berikut, lihat Gambar.2

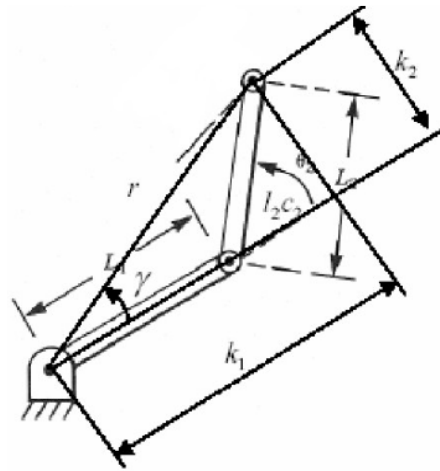


Gambar.3. diambil dari gambar 2

$$\begin{aligned}
 x &= l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2) \\
 y &= l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2) \\
 x &= k_1 \cos \theta_1 - k_2 \sin \theta_1 \\
 y &= k_1 \sin \theta_1 + k_2 \cos \theta_1
 \end{aligned}$$

Dimana,  
 $k_1 = l_1 + l_2 \cos \theta_2$   
 $k_2 = l_2 \sin \theta_2$

Selanjutnya kita tulis konstanta  $k_1$  dan  $k_2$  lihat pada Gambar.3.



Gambar.4. Diambil dari gambar 2

$$\begin{aligned}
 r &= (k_1^2 + k_2^2)^{1/2} \\
 k_1 &= r \cos \gamma ; k_2 = r \sin \gamma
 \end{aligned}$$

Memasukkan ke transformasi sebelumnya dari hasil x dan y

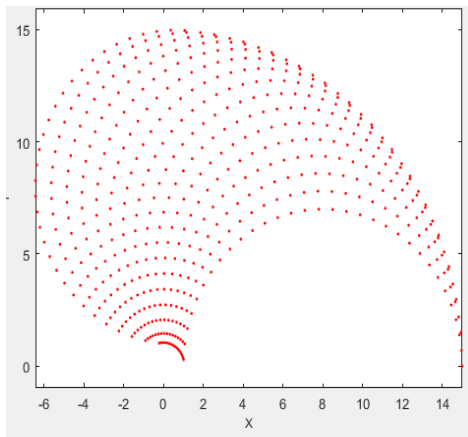
$$\begin{aligned}
 x &= r \cos \gamma \cos \theta_1 - r \sin \gamma \sin \theta_1 ; \frac{x}{r} = \cos \gamma \cos \theta_1 - \sin \gamma \sin \theta_1 ; \frac{x}{r} = \cos (\gamma + \theta_1) \\
 y &= r \cos \gamma \sin \theta_1 - r \sin \gamma \cos \theta_1 ; \frac{y}{r} = \cos \gamma \sin \theta_1 - \sin \gamma \cos \theta_1 ; \frac{y}{r} = \sin (\gamma + \theta_1) \\
 \gamma + \theta_1 &= \text{atan2}\left(\frac{y}{r}, \frac{x}{r}\right) = \text{atan2}(y, x) ; \text{jadi, } \theta_1 = \text{atan2}(y, x) - \text{atan2}(k_2, k_1)
 \end{aligned}$$

**HASIL PENGUJIAN**

Kinematika kaki robot memiliki *workspace area*. Hal ini dapat diketahui melalui konstruksi mekanik kaki robot seperti, tata letak aktuator motor servo dan keterbatasan sudut yang dihasilkan oleh motor servo. Dari Tabel.2 dapat kita lihat *workspace area* robot hexapod 2 DOF (*Degree Of Freedom*).

Tabel.2. Nilai Maksimum dan Minimum Sudut Aktuator Servo

JOINT	Panjang Link Robot	Sudut Minimum (°)	Sudut Maksimum (°)
JOINT 1	7 cm	60	120
JOINT 2	7 cm	60	120



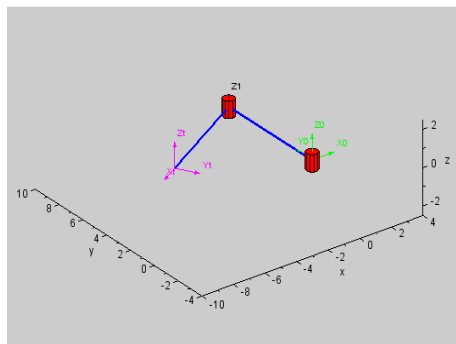
Gambar.5. Workspace Area

Pada Gambar.5 di atas menunjukkan area kerja yang dapat dijelajahi oleh ujung kaki robot dengan besar sudut dan panjang link seperti yang ada di Tabel.2.

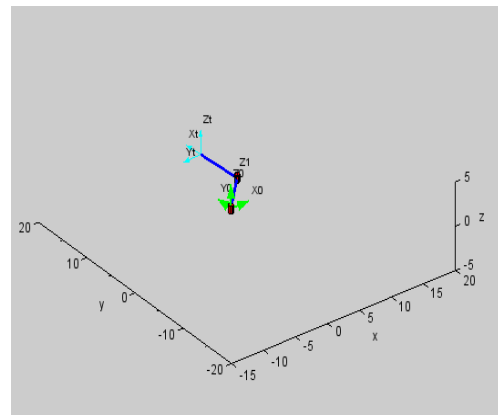
Tabel.3. 2 DOF Manipulator Menggunakan Metode Denavit-Hartenberg

JOINT	Theta	d	a (panjang link)	alpha
JOINT 1	90 <sup>0</sup>	0.00	7 cm	0.00
JOINT 2	120 <sup>0</sup>	0.00	7 cm	0.00

Ilustrasi data ini menggunakan 2 link dan 2 joint dengan a<sub>1</sub>,a<sub>2</sub> = 7 cm ; Theta 1= 90<sup>0</sup> dan Theta 2= 120<sup>0</sup>. Pada titik ini kita fokus pada bagian atas informasi, terutama tabel parameter DH, untuk memverifikasi bahwa sebenarnya manipulator data yang diinginkan. Simulator data ini menggunakan software scilab. Dalam hal ini, hanya menunjukkan cara dasar untuk menggerakkan robot. Dalam prakteknya, kita mungkin ingin mengarahkan pergerakan kaki tersebut menggunakan data dari algoritma generasi perintah yang ada pada scilab yang terdapat pada Gambar.7.



Gambar.6. Plot Robot



Gambar.7. Plot Robot Ketika Disimulasikan untuk bergerak

Tabel.4. Desire Position Terhadap Pendekatan Nilai Inverse Kinematics

No	Theta 1	Theta 2	Set Point Desire Position (cm)		
			X	Y	Z
1	48.34	19.37	0	3	3
2	90.00	23.46	0	7	2
3	90.00	36.68	0	6	3
4	48.34	62.41	0	3	8
5	25.74	31.29	0	2	5

Metode *inverse kinematics* merupakan transformasi posisi pada ruang cartesian menjadi sudut joint dalam ruang joint. Dengan pendekatan *inverse kinematics*, tingkat akurasi dapat diketahui melalui desire position dalam bidang cartesian terhadap sudut actuator servo. Dari pengamatan tabel diatas dapat dilihat *desire position* dengan nilai rerata posisi sudut *inverse kinematics*. Besarnya selisih dari bidang Cartesian yang ditentukan dan nilai persentase *error* terhadap posisi aktual *end-effector* yang dapat dilihat pada Tabel.5.

Tabel.5. Set Point Terhadap Posisi Aktual dan Persentase Error

Set Point (cm)			Actual (cm)			Selisih (cm)			Persentase Error		
x	y	z	x	y	z	x	y	z	x	y	z
0	3	3	0	3.1	3.1	0	0.1	0.1	0	3.3%	3.3%
0	7	2	0	7.1	2	0	0.1	0	0	1.4%	0%
0	6	3	0	5.8	3.2	0	0.2	0.2	0	3.3%	6.6%
0	3	8	0	3.1	8.2	0	0.1	0.2	0	3.3%	2.5%
0	2	5	0	2	5.2	0	0	0.2	0	0%	4%

Untuk input yang sama dapat terjadi nilai error yang sama dapat terjadi nilai error yang berbeda hal ini terjadi akibat pergerakan kaki robot terhadap penentuan titik gerak. Dengan nilai dari set point x, y, atau z yang tidak sama akan menyebabkan seluruh actuator atau servo akan bergerak sehingga letak atau pun posisi kaki robot dapat terjadi perubahan.

Rata-rata error pergerakan kaki robot dapat dihitung melalui perbandingan dari jumlah error terhadap jumlah data yang diambil.

$$\text{rata error (cm)} = \frac{\text{jumlah error}}{\text{jumlah data}}$$

Rata-rata error disumbu x :

$$\text{rata error (cm)} = \frac{0}{5} = 0.0 \text{ cm}$$

Rata-rata error disumbu y :

$$\text{rata error (cm)} = \frac{0.5}{5} = 0.1 \text{ cm}$$

Rata-rata error disumbu z :

$$\text{rata error (cm)} = \frac{0.7}{5} = 0.14 \text{ cm}$$

## KESIMPULAN

Dapat disimpulkan bahwa implementasi *inverse kinematics* terhadap pergerakan kaki robot lebih efisien dan efektif dikarenakan pergerakan kaki robot terhadap bidang *Cartesian* lebih tepat dan hanya mengalami sedikit *error*. Rata – rata *error* terhadap sumbu x adalah 0.06 cm, sumbu y 0.12 cm dan sumbu z 0.1 cm.

## DAFTAR PUSTAKA

- Bekey G. A., *Autonomous Robot: From Biological Inspiration to Implementation and Control*, MIT Press, 2005.
- Siciliano B, Khatib O 2008. *Handbook of Robotics* [bibliografi]. Berlin (DE): Springer.
- Thirion B, Thiry L. 2002. Concurrent programming for the control of hexapod walking. *ACM SIGAda Ada Letters*. 22 (1):17-28. doi: 10.1145/507548.507551.
- Arefi, R & Sadiqh, M J 2011, *Fuzzy Inverse Kinematics Algotihm For Man and*

*Machine Cooperation*, University of Technology Isfahan, Iran.

- Reyneri LM. 1997. An introduction to fuzzy state automata. Di dalam *International Work-Conference on Artificial and Natural Neural Networks*. 1997 Jun 4-6; Canary Islands, Spanyol. Canary Islands (ES). hlm: 273-283.
- Arif, A, *Kontrol Sliding Mode Dengan Tebal Bounday Layer Yang Bervariasi Pada Manipulator Planar*, Thesis Magister, Program Studi Instrumentasi dan Kontrol Program Pasca Sarjana ITB, Bandung, 2000.
- Koivo, A., J., *Fundamentals for Control of Robotic Manipulators*, John Wiley & Sons. Inc., New York, 1989.
- Hamel, R., W., Haley, C., D., *Advanced Robotics for Decontamination and Dismantlement, in Intelligent Automation and Soft Computing, Trends in Research, Development, an Aplications*, Vo1.1. TSI Press Series, Albuquerque, New Mexieo, USA, 1994.
- Haselman, D., Letfield, B., *Matlab, Bahasa Komputasi Teknis*, Penerbit Andi Yogyakarta, 2001.