

Projekt robota trójnożnego

Jakub Mazur

16 października 2022

Spis treści

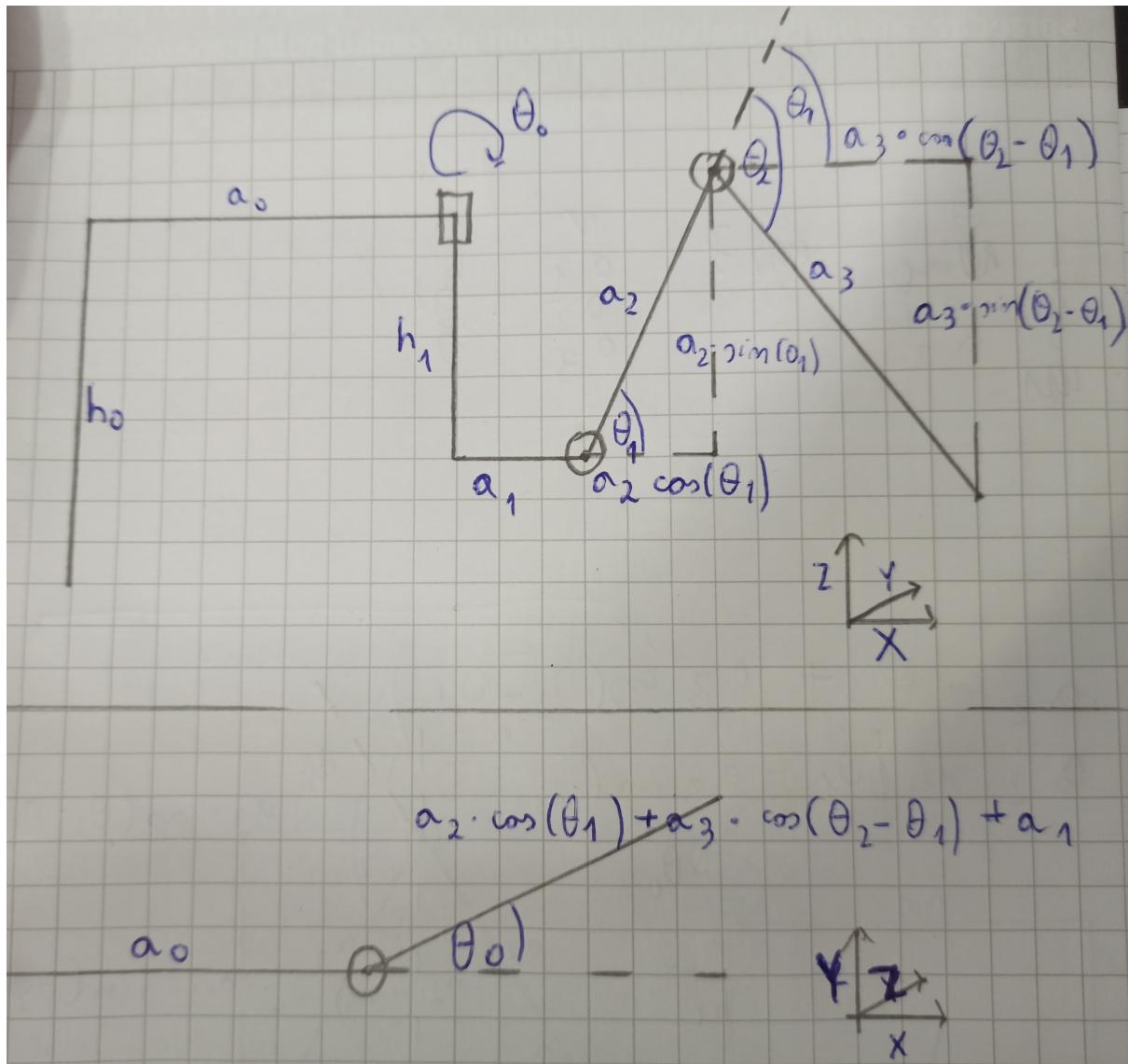
1 Model Matematyczny	1
1.1 Noga robota typu RRR	1
1.1.1 Forward kinematics	2
1.1.2 Forward kinematics - denavit hartenberg [1]	3
1.1.3 Invert kinematics	4
1.2 Cały Robot	5
1.2.1 Matematyka kroku	5
2 Model CAD i druk 3D	5
3 Środowisko ROS	5
3.1 Schemat implementacji	5
4 Algorytm Chodu	5

1 Model Matematyczny

Noga ma 3 stopnie swobody. Wszystkie są typu obrotowego, przy czym dwie obracają się dookoła osi poziomej, a jedna dookoła osi pionowej. Są to te same osi obrotu co w przypadku ramienia robotycznego typu antromorficznego.

1.1 Noga robota typu RRR

Jest to noga o 3 rotacyjnych stopniach swobody, przy czym 1 stopień obraca się dookoła osi z, a 2 stopnie dookoła osi y. Model matematyczny takiej nogi, a co za tym idzie także jej kinematyka, będzie identyczny jak model ramienia robotycznego typu antropomorficznego (zwanego także "angular" bądź "jointed").

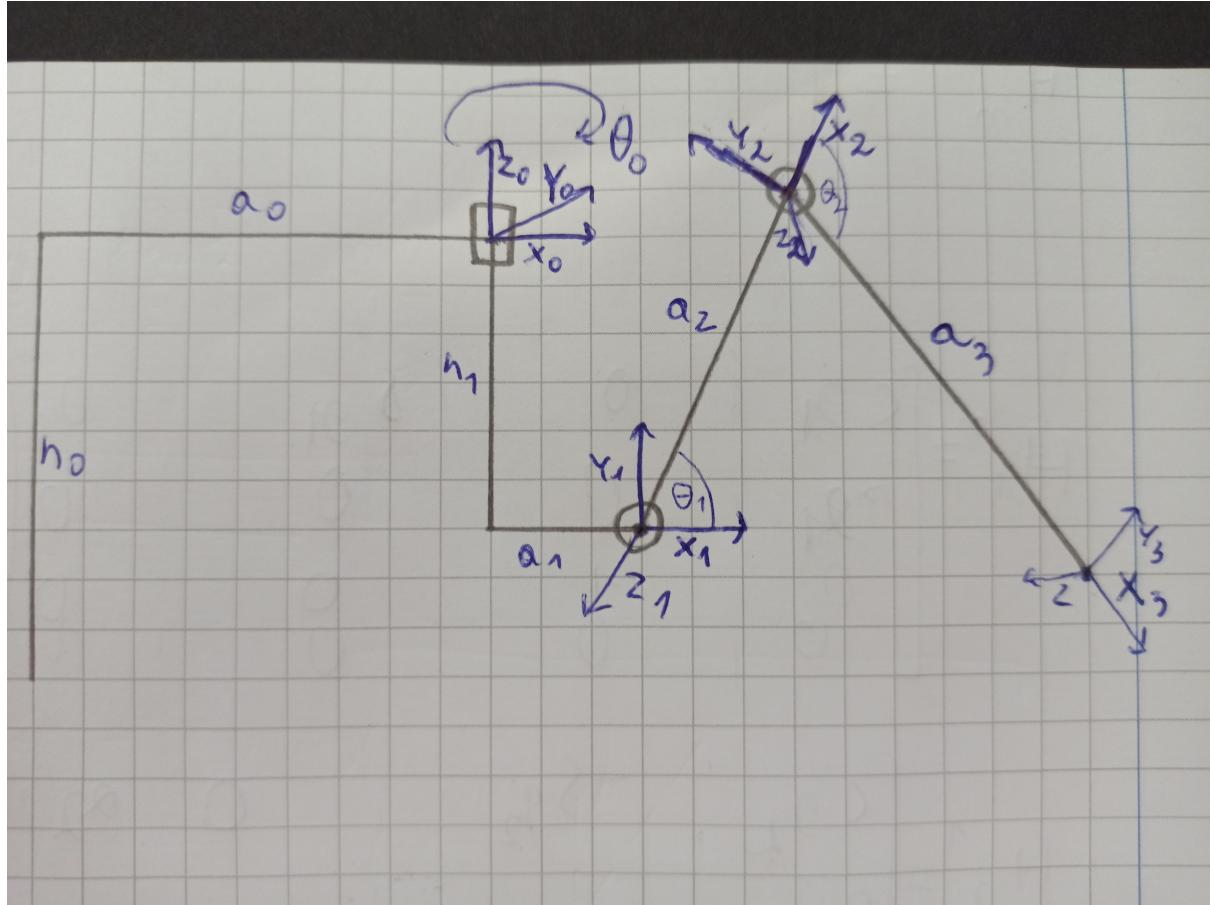


Rysunek 1: Model Matematyczny

1.1.1 Forward kinematics

$$\begin{aligned}
 a_{temp} &= a_2 \cos \theta_1 + a_3 \cos (\theta_2 - \theta_1) + a_1 \\
 Y &= a_{temp} \cdot \sin \theta_0 \\
 X &= a_{temp} \cdot \cos \theta_0 + a_0 \\
 Z &= h_0 - h_1 + a_2 \sin \theta_1 - a_3 \sin (\theta_2 - \theta_1)
 \end{aligned} \tag{1}$$

1.1.2 Forward kinematics - denavit hartenberg [1]



Rysunek 2: Model Denavit Hartenberg

Joint i	θ_i	α_i	r_i	d_i	
1	θ_0	$\frac{\pi}{2}$	a_1	h_1	
2	θ_1	0	a_2	0	
3	θ_2	0	a_3	0	

(2)

Gdzie:

θ_i - angle from x_{n-1} to x_n around z_{n-1}

α_i - angle from z_{n-1} to z_n around x_n

r_i - distance between the origin of the $n-1$ frame and the origin of the n frame along the x_n direction.

d_i - distance from x_{n-1} to x_n along the z_{n-1} direction

Następnie macierze przejść pomiędzy ramkami $n-1$ i n oblicza się zgodnie z następującym wzorem:

$$H_n^{n-1} = \left[\begin{array}{ccc|c} R & & & T \\ \hline 0 & 0 & 0 & 1 \end{array} \right] = \left[\begin{array}{cccc} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{array} \right] [2] \quad (3)$$

Gdzie R oznacza macierz Rotacji a T macierz Transformacji.

A macierz przejścia pomiędzy ramką 0 a ramką 3 zgodnie z następującym wzorem:

$$H_n^{n-1} = H_1^0 \cdot H_2^1 \cdot H_3^2 =$$

$$\begin{bmatrix} c\theta_0 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2) & -c\theta_0 \cdot (c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2) & s\theta_0 & c\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2)) \\ s\theta_0 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2) & -s\theta_0 \cdot (c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2) & -c\theta_0 & s\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2)) \\ c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2 & c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2 & 0 & h_1 + a_2 \cdot s\theta_1 + a_3 \cdot (c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} c\theta_0 \cdot c(\theta_1 + \theta_2) & -c\theta_0 \cdot s(\theta_1 + \theta_2) & s\theta_0 & c\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot c(\theta_1 + \theta_2)) \\ s\theta_0 \cdot c(\theta_1 + \theta_2) & -s\theta_0 \cdot s(\theta_1 + \theta_2) & -c\theta_0 & s\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot c(\theta_1 + \theta_2)) \\ c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2 & c(\theta_1 + \theta_2) & 0 & h_1 + a_2 \cdot s\theta_1 + a_3 \cdot s(\theta_1 + \theta_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Następnie, aby otrzymać właściwe równania, należy z równania 4 wziąć część odpowiedzialną za transformację.

$$\begin{aligned} X &= c\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot c(\theta_1 + \theta_2)) \\ Y &= s\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot c(\theta_1 + \theta_2)) \\ Z &= h_1 + a_2 \cdot s\theta_1 + a_3 \cdot s(\theta_1 + \theta_2) \end{aligned} \quad (5)$$

1.1.3 Invert kinematics

Odwrotną kinematykę można obliczyć poprzez rozwiązywanie równań kinematyki prostej.

Najprościej jest wyprowadzić wzór na θ_0 , można to zrobić na podstawie wzoru 1, łącząc wzór na X i Y :

$$\begin{cases} Y = a_{temp} \cdot \sin \theta_0 + a_0 \\ X = a_{temp} \cdot \cos \theta_0 \end{cases} \quad (6)$$

$$\begin{cases} Y = a_{temp} \cdot \sin \theta_0 \\ a_{temp} = \frac{X - a_0}{\cos \theta_0} \end{cases} \quad (7)$$

$$Y = \frac{X - a_0}{\cos \theta_0} \cdot \sin \theta_0 \quad (8)$$

$$\theta_0 = \arctan \frac{Y}{X - a_0} \quad (9)$$

Wzór na θ_2 można wyprowadzić także z równania 1

$$Z = (h_0 - h_1) + a_2 \cdot \sin \theta_1 - a_3 \sin (\theta_2 - \theta_1)$$

$$Z - (h_1 + h_0) = a_2 \cdot \sin \theta_1 - a_3 \sin (\theta_2 - \theta_1)$$

$$\frac{(h_1 + h_0) - Z - a_2 \cdot \sin \theta_1}{a_3} = \sin (\theta_2 - \theta_1)$$

$$\theta_2 = \arcsin \left(\frac{(h_1 + h_0) - Z - a_2 \cdot \sin \theta_1}{a_3} \right) + \theta_1 \quad (10)$$

Równanie 10 jest nadal uzależnione od θ_1 , które należy wyznaczyć.

Pod równanie 1 na a_{temp} podstawiam obliczone równanie 10 i otrzymuję równanie 11

$$a_{temp} = a_2 \cos \theta_1 + a_3 \cos \left(\arcsin \left(\frac{(h_1 + h_0) - Z - a_2 \cdot \sin \theta_1}{a_3} \right) \right) + a_1 \quad (11)$$

Do fragmentu $a_3 \cos \left(\arcsin \left(\frac{(h_1 + h_0) - Z - a_2 \cdot \sin \theta_1}{a_3} \right) \right)$ można zastosować wzór $\cos(\arcsin(x)) = \sqrt{1 - x^2}$. Daje to wzór 12

$$a_{temp} = a_2 \cos \theta_1 + \sqrt{a_3^2 - (a_2 \sin \theta_1 + h_0 - h_1 - Z)^2} + a_1 \quad (12)$$

Niestety wzór 12 nie pozwala jednoznacznie wyznaczyć jawnego θ_1 . Przybliżanie metodą Taylora także się w tym przypadku nie sprawdzi.

1.2 Cały Robot

Zwykle konstruując roboty wzorujemy się na zwierzętach występujących w naturze. W tym przypadku nie mamy jednak tego luksusu, dlatego na początek należy przyjąć jakieś najbardziej intuicyjne założenie. Dlatego też uznałem że najlepszym rozwiązańiem będzie rozmieścić nogi na jednej płaszczyźnie w równych odstępach - co 120 deg.

W poprzednim podrozdziale przyjąłem osie układu relatywne do ułożenia początkowego nogi robota. Oznacza to że robot trójnożny będzie miał 3 niezależne osie x i 3 niezależne osie y , tylko oś z zgadza się między kolejnymi nogami. Idzie za tym konieczność stworzenia pewnej metody "obracania" wszystkich tych osi x i y do jednego zunifikowanego układu współrzędnych. Pozwoli to obliczyć konematykę całego robota.

1.2.1 Matematyka kroku

Aby móc w prosty sposób napisać później algorytm chodu i móc się skupić na kolejności przestawiania nóg przez robota, długości i wysokości kroku i innych tego typu aspektach, należy teraz dobrze ten krok sparametryzować. Na podstawie moich poprzednich doświadczeń z podobnymi projektami, uważam że najbardziej intuicyjne będzie sparametryzowanie kroku do:

- wartości kąta pomiędzy kierunkiem "do przodu" kroku a osią x danej nogi
- długości kroku
- wysokości kroku

2 Model CAD i druk 3D

3 Środowisko ROS

3.1 Schemat implementacji

4 Algorytm Chodu

Literatura

- [1] How to Find Denavit-Hartenberg Parameter Tables, blogpost by Automatic Addison
- [2] Homogeneous Transformation Matrices Using Denavit-Hartenberg, blogpost by Automatic Addison
- [3] Matlab Denavit Hartenberg calculations
- [4] Alexander Wallen Kiessling, Niclas Maatta (2020) Anthropomorphic Robot Arm