

Projekt robota trójnożnego

Jakub Mazur

27 października 2022

Spis treści

1	Wstęp	1
1.1	Istniejące konstrukcje trójnożne	2
1.1.1	STriDER	2
1.1.2	Triped "Martian"	2
1.2	Maker Faire tripod robot	2
2	Model Matematyczny	2
2.1	Noga robota typu RRR	2
2.1.1	Forward kinematics	3
2.1.2	Forward kinematics - denavit hartenberg [2]	3
2.1.3	Invert kinematics	5
2.2	Cały Robot	5
2.2.1	Matematyka kroku	6
3	Model CAD i druk 3D	6
4	Środowisko ROS	6
4.1	Schemat implementacji	6
5	Algorytm Chodu	6

1 Wstęp

Pierwsza idea robota krocącego pojawiła się już pod koniec XV wieku, a narodziła się w głowie nikogo innego jak Leonarda Da Vinci. Od tamtej pory wielu naukowców próbowało tworzyć konstrukcje które używały nóg zamiast kół, lecz pierwsze faktycznie udane konstrukcje datuje się dopiero na początek lat 60 ubiegłego wieku. Pojawiły się wtedy pierwsze działające konstrukcje, na przykład robot czteronożny zbudowany przez Josepha Shingleya oraz roboty sześć i ośmonożne zbudowane przez "Space General Corporation".

Od tamtej pory powstało wiele różnych konstrukcji, które kategoryzuje się w zależności od ilości nóg posiadanych przez robota:

- jednonożne
- dwunożne (Humanoid, chicken-walkers)
- czteronożne (Quadrupedal)
- sześćonożne (Hexapod)
- ośmonożne
- gąsiennicowe

Przy czym można zaobserwować tendencję spadkową, wraz z upływem czasu widać wzrost udanych konstrukcji o mniejszej ilości nóg.

Pojawia się także inna tendencja w ilości nóg robotów. Prawie wszystkie konstrukcje (poza jednonożnymi) opierają się na zwierzętach występujących w naturze. Jest to raczej logiczna tendencja, jako że do takich robotów mamy już algorytmy chodu opracowane przez miliony lat ewolucji, "konstrukcje" które nie mają sensu nie przetrwałyby do dziś.

Co natomiast z konstrukcjami robotów trójnożnych? Można się zastanowić czy konstrukcje takie nie powstają ponieważ faktycznie nie mają sensu, czy może dlatego że temat bardziej "klasycznych", prostszych w implementacji, konstrukcji nie został po prostu jeszcze wyczerpany przez naukowców. Jeżeli rozejrzemy się dookoła siebie możemy zaobserwować wiele przedmiotów codziennego użytku które posiadają właśnie 3 nogi, od wszelakich taboretów, przez wieszaki na ubrania po stoły. Są to jednak konstrukcje statyczne i dla takich konstrukcji 3 nogi są wymaganiem minimum aby dany przedmiot stał stabilnie. Co jednak z konstrukcjami dynamicznymi? Jeżeli robot trójnożny podniesie nogę, straci stabilność, zacznie się przewracać. Czyni to z niego dość ciekawą konstrukcję, gdzie w momencie stania w miejscu zachowuje się bardziej jak roboty o większej ilości nóg, nie przewraca się, a podczas ruchu zachowuje się jak roboty dwunożne, musi odpowiednio szybko odstawić nogę w odpowiednie miejsce aby się nie przewrócić.

1.1 Istniejące konstrukcje trójnożne

Konstrukcje trójnożne pojawiały się w dziełach science fiction już od dawna, od "The War of the worlds" z 1898 aż po "Mroczne Widmo" z 2001 i regularnie pojawiają się naukowcy którzy próbują udowodnić że nie trzeba ograniczać tego typu robotów do dzieł z obszaru science fiction.

1.1.1 STriDER

Zbudowany w 2007

<http://www.romela.org/wp-content/uploads/2015/05/Forward-and-Inverse-Displacement-Analysis-of-a-Novel-Three-Legged-Mobile-Robot-Based-on-the-Kinematics-of-In-Parallel-Manipulators.pdf>
3-SRRR

1.1.2 Triped "Martian"

Jest to robot zbudowany przez Yoichi Masudę na uniwersytecie w Osace w 2017 3-RL

1.1.3 Inne konstrukcje

Na internecie można także znaleźć kilka różnych niezbyt dobrze udokumentowanych konstrukcji zakończonych mniejszym lub większym sukcesem:

<https://makerfaire.com/maker/entry/71669/>

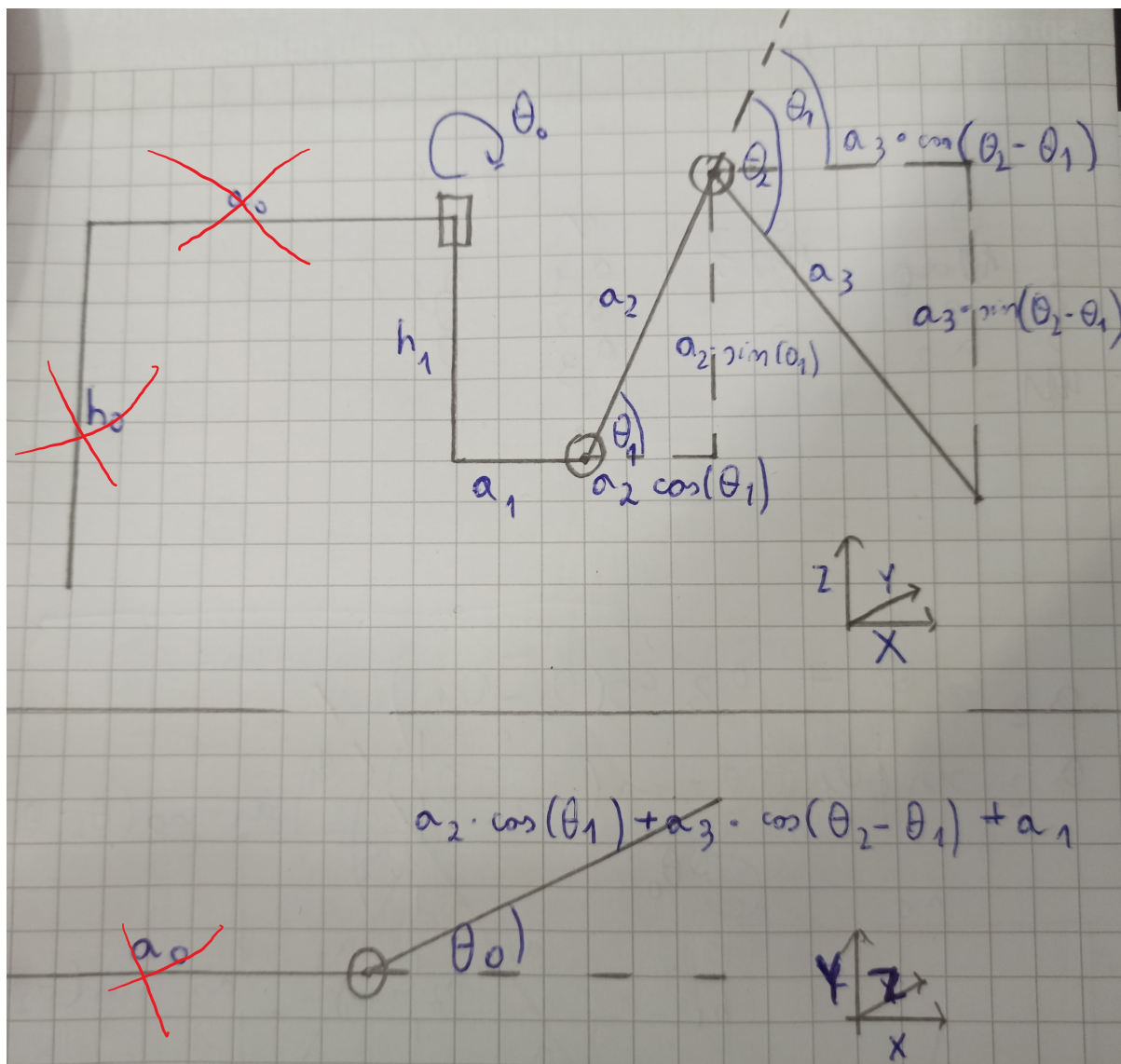
<https://www.youtube.com/watch?v=HGEhCCUgFMg>*channel = driftingswiftly*

2 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 antropomorficznego.

2.1 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

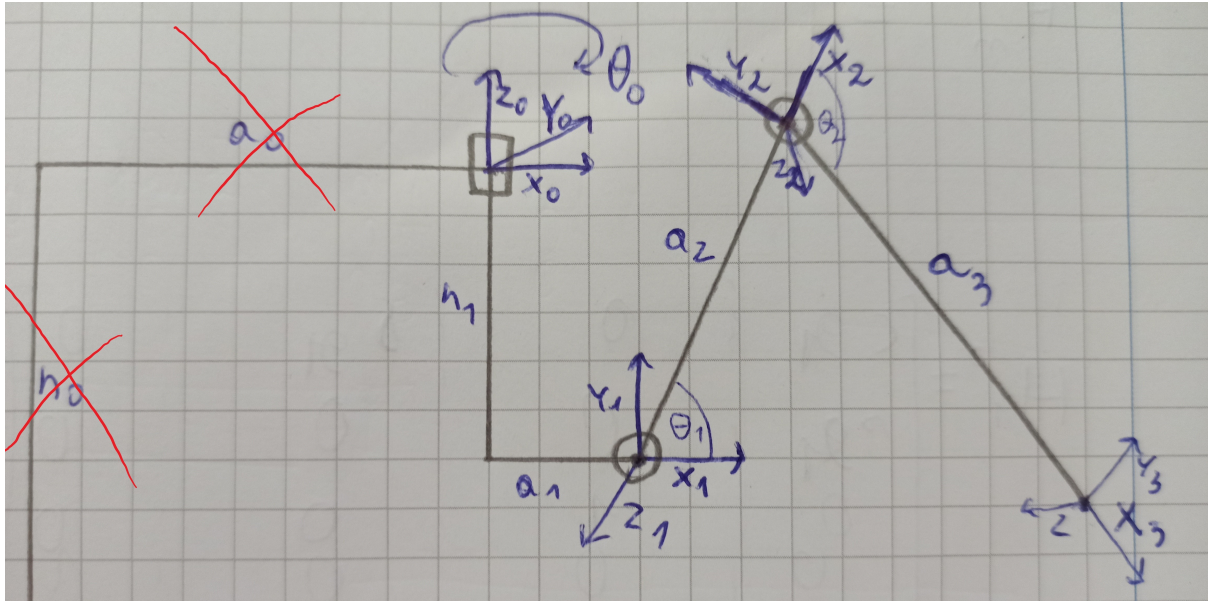
2.1.1 Forward kinematics

Obliczanie kinematyki prostej polega na TODOTODO

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

2.1.2 Forward kinematics - denavit hartenberg [2]

Alternatywą dla zwykłych obliczeń geometrycznych jest metoda Denvaita Hartenberga - TODOTODO-TODO



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:

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

Gdzie R oznacza macierz Rotacji a T macierz Transformacji.

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

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

$$\left[\begin{array}{cccc} c\theta_0 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2) & -c\theta_0 \cdot (c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2) & s\theta_0 & c\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2)) \\ s\theta_0 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2) & -s\theta_0 \cdot (c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2) & -c\theta_0 & s\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot (c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2)) \\ c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2 & c\theta_1 \cdot c\theta_2 - s\theta_1 \cdot s\theta_2 & 0 & h_1 + a_2 \cdot s\theta_1 + a_3 \cdot (c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2) \\ 0 & 0 & 0 & 1 \end{array} \right] =$$

$$\left[\begin{array}{cccc} c\theta_0 \cdot c(\theta_1 + \theta_2) & -c\theta_0 \cdot s(\theta_1 + \theta_2) & s\theta_0 & c\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot c(\theta_1 + \theta_2)) \\ s\theta_0 \cdot c(\theta_1 + \theta_2) & -s\theta_0 \cdot s(\theta_1 + \theta_2) & -c\theta_0 & s\theta_0 \cdot (a_1 + a_2 \cdot c\theta_1 + a_3 \cdot c(\theta_1 + \theta_2)) \\ c\theta_1 \cdot s\theta_2 + s\theta_1 \cdot c\theta_2 & c(\theta_1 + \theta_2) & 0 & h_1 + a_2 \cdot s\theta_1 + a_3 \cdot s(\theta_1 + \theta_2) \\ 0 & 0 & 0 & 1 \end{array} \right] \quad (4)$$

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

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

TODO - znak się nie zgadza

2.1.3 Invert kinematics

Kinematyka odwrotna to TODO

Zwykle odwrotną kinematykę można obliczyć poprzez rozwiązanie równań kinematyki prostej. Jest to w tym przypadku także teoretycznie możliwe, jako że mamy do czynienia z trzema niewiadomymi (θ_{0-2}) i układem trzech równań nr. 1 lub 1. (Równanie na a_{temp} jest jedynie pomocnicze, trzeba je traktować jak część równań Y i X).

Jest to jak najbardziej wykonalne w przypadku θ_0 , można to zrobić łącząc wzór na X i Y , dzieląc go obustronnie, zamieniając $\frac{\sin}{\cos}$ na \tan i wyciągając θ_0 na lewą stronę. Daje to wzór 6

$$\theta_0 = \arctan \frac{Y}{X} \tag{6}$$

Większy problem pojawia się jednak w przypadku obliczania θ_1 i θ_2 , ponieważ wartości te da się obliczyć z wyżej wspomnianego równania, ale nie da się wyznaczyć na te wartości równań w postaci jawnej. A bez postaci jawnej nie będzie możliwa poprawna implementacja tych równań.

Można natomiast zastosować pewnego rodzaju trik. Jeżeli wyznaczanie równań θ_1 i θ_2 sprowadzi się do problemu dwuwymiarowego, obliczenia stają się w zasadzie identyczne jak w przypadku obliczeń kinematyki odwrotnej dla robota typu SCARA, co było już robione wielokrotnie.

Aby obliczyć kąt θ_2 można powrócić do rysunku 2 i zastosować na trójkącie utworzonym z a_2 i a_3 twierdzenie cosinusów połączone z twierdzeniem pitagorasa. Daje to wzór 7.

$$(x - a_1)^2 + (z - h_1)^2 = a_2^2 + a_3^2 - 2 \cdot a_2 \cdot a_3 \cdot \cos(180^\circ - \theta_2) \tag{7}$$

Następnie na podstawie równania 7 aby ostatecznie otrzymać θ_2 należy zastosować wzór $\cos(180^\circ - \theta) = -\cos \theta$ i za pomocą prostych przekształceń wyciągnąć z równania θ_2 . Daje to wzór 8

$$\theta_2 = \arccos \left(\frac{(x - a_1)^2 + (z - h_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \right) [5] \tag{8}$$

Natomiast obliczenia dla θ_1 TODO

$$\theta_1 = \arctan \frac{z - h_1}{x - a_1} - \arcsin \left(\frac{a_3 \sin \theta_2}{\sqrt{(x - a_1)^2 + (z - h_1)^2}} \right) [5] \tag{9}$$

Ostatecznie zbierając równania 6, 9, 8 otrzymujemy układ równań który stanowi kinematykę odwrotną opisywanej nogi robotycznej.

TODO - układ równań

TODO - ograniczenia.

2.2 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 rozwiązaniem 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 kinematykę całego robota.

2.2.1 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

3 Model CAD i druk 3D

4 Środowisko ROS

4.1 Schemat implementacji

5 Algorytm Chodu

Literatura

- [1] A Historical Perspective of Legged Robots
- [2] How to Find Denavit-Hartenberg Parameter Tables, blogpost by Automatic Addison
- [3] Homogeneous Transformation Matrices Using Denavit-Hartenberg, blogpost by Automatic Addison
- [4] Matlab Denavit Hartenberg calculations
- [5] Adam Labuda, Janusz Pomirski, Andrzej Rak (2009) Model manipulatora o dwóch stopniach swobody
- [6] Alexander Wallen Kiessling, Niclas Maatta (2020) Anthropomorphic Robot Arm