Maestría en Ingeniería en Automatización de Procesos Industriales

Title

Estimation of orientation through quaternion by 6 degrees of freedom robot

Author

Luis Castillo Bermudez

Contributor

B. Benito Salmeron Quiroz

September-December 2013

Third Symposium of the Post-Graduate Department
Universidad Politécnica de Puebla
Engineer Master's in Industrial Processes Automation

Posgrado

Estimation of orientation through quaternion by a 6 degrees of freedom robot

Luis Castillo Bermudez[1], B. Benito Salmeron Quiroz[2]

[1]Engineer Master's in Automation of Industrial Processes, Universidad Politécnica de Puebla, Tercer Carril del Ejido Serrano s/n San Mateo Cuanalá, Juan C. Bonilla, Puebla, México
luis.castillo@uppuebla.edu.mx

[2]Instituto Politécnico Nacional (I.P.N.), SEPI ESIME Azcapotzalco, Av. de la Granjas 682,02250, México, D.F., México

Introduction

Currently there are controllers that use different systems to determine the orientation of a rigid body. However, the usual representations such as Euler angles and rotation matrices, have drawbacks such as loss of one degree of freedom called Gimbal Lock. The use of quaternions has many advantages:

It has better numerical properties, avoids the gimbal lock and provides an easier and faster way for rotational interpolation.

A quaternion is an alternative way of representing rotations through any axis:

$$Q = (q_0\,(iq_1\,jq_2\,kq_3)) = (q_0 + q)$$
$$= \left(\cos\left(\frac{\phi}{2}\right) + \mathrm{sen}\left(\frac{\phi}{2}\right)e\right)$$

General objective

Estimating and modeling through quaternion the orientation of a 6 DOF robot in order to design a mathematical model for the analysis of the direct-inverse kinematics of a robot.

Specific Objectives

1. To develop a mathematical model of rotations and direct and inverse kinematic model with quaternions.

2. To demonstrate using VRML (Virtual Reality Modeling Language), that the mathematical model is an accurate and reliable representation of the direct kinematics of the robot, thus demonstrating that this model can avoid GimbalLock, or at least reducing this problem.

3. Validate the results of the mathematical model and kinematic model of rotations in deferred time, using an IMU in a robot IRB1600.

Method

- By Quaternion algebra is proposed a linear operator that works with vectors $v \in \mathbb{R}^3$

$$w = L_Q v = Q \otimes v \otimes Q^*$$
$$= (q_0^2 - \|q\|^2)v + 2(q \cdot v)q + 2q_0(q \times v)$$

- To construct forward kinematics model by quaternions, programming algorithms in MatLab to calculate the position and analyze how the robot links are rotating.
- To compare between the model based on quaternion and the conventional model based on Denavit – Hartenberg convention showing how the rotation is continuous and avoids "Gimbal – Lock".
- To constructed a VRML model to validate the mathematical model based on quaternions it is showed in the Figure 2 through a GUI in MatLab®.
- To compare time spend between model based on quaternion and the conventional model based on Denavit – Hartenberg convention

DH parameters					Quaternion model	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	Quaternion	Vector
1	θ_1	353 mm	150 mm	$-\pi/2$	$Q1 = [\cos(\frac{\theta_1}{2})\ 0\ 0\ \mathrm{sen}(\frac{\theta_1}{2})]$	$P1 = [0\ 150\ 481]$
2	$\theta_2 - \pi/2$	0	700 mm	0	$Q2 = [\cos(\frac{\theta_2}{2})\ \mathrm{sen}(\frac{\theta_2}{2})\ 0\ 0]$	$P2 = [0\ 0\ 700]$
3	θ_3	0	0	$-\pi/2$	$Q3 = [\cos(\frac{\theta_3}{2})\ \mathrm{sen}(\frac{\theta_3}{2})\ 0\ 0]$	$P3 = [314\ 0\ 0]$
4	θ_4	600 mm	0	$-\pi/2$	$Q4 = [\cos(\frac{\theta_4}{2})\ 0\ \mathrm{sen}(\frac{\theta_4}{2})\ 0]$	$P4 = [286\ 0\ 0]$
5	θ_5	0	0	$\pi/2$	$Q5 = [\cos(\frac{\theta_5}{2})\ \mathrm{sen}(\frac{\theta_5}{2})\ 0\ 0]$	$P5 = [62\ 0\ 0]$
6	θ_6	65 mm	0	0	$Q6 = [\cos(\frac{\theta_6}{2})\ 0\ \mathrm{sen}(\frac{\theta_6}{2})\ 0]$	$P6 = [23\ 0\ 0]$

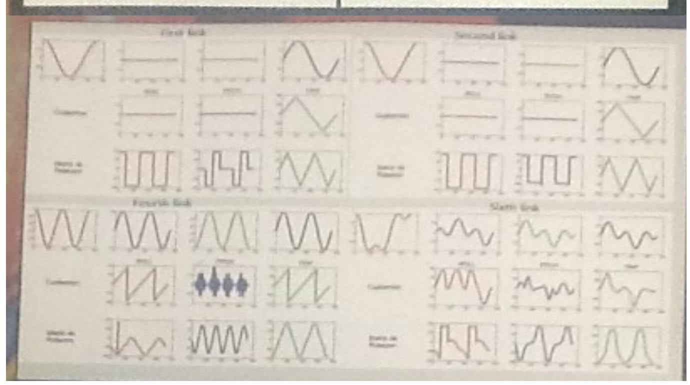

Figure 1: Euler angles corresponding to quaternion and rotation matrices

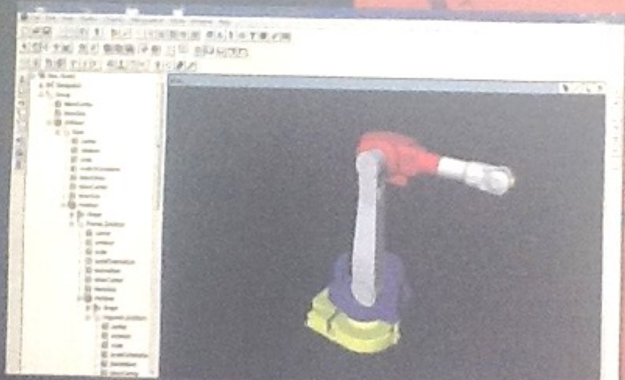

Figure 2: VRML model construction and union the kinematic chain to implement the mathematical model.

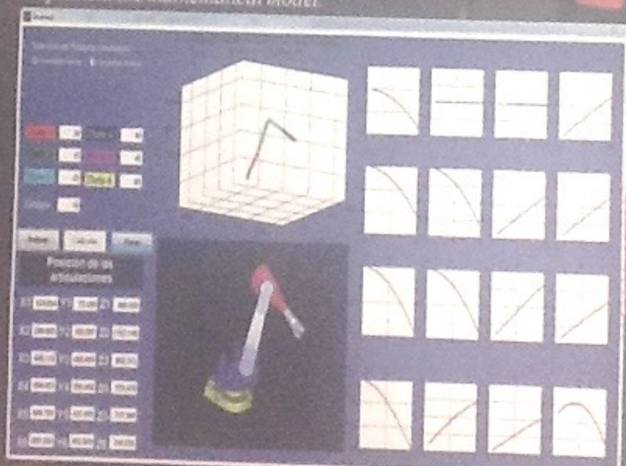

Figure 3: MatLab GUI that show the VRML model and all parameters obtained by the forward kinematics model the graphics showed

Results

Results obtained so far are the following:

Has been showed that the model based on quaternions is simpler than the model based on Denavit-Hartenberg parameters, which uses rotation matrices. The model based on quaternion requires less computational resources, we can say that for rotational models based on quaternions are more efficient when the mathematical processing is implemented.

Figure 1 shows that the model based on quaternions have lesser indetermination point than the conventional models and avoids the problem of "Gimbal – Lock" as remain constant.

Figure 3 shows a graph model made in VRML which by programming in MatLab GUI takes the values of the rotations calculated using the rotational operator proposed. Hence, it is of great importance in the validation of this study as it represents the functionality of the mathematical model.

Profile Summary

Generated 10-Mar-2014 10:02:09 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Prueba_tiempo	1	0.836 s	0.504 s	
CRQ	1001	0.282 s	0.173 s	
Qrot	6006	0.058 s	0.058 s	
[illegible]	6006	0.053 s	0.053 s	
xlabel	2	0.027 s	0.024 s	
ylabel	2	0.006 s	0.006 s	
zlabel	2	0.006 s	0.006 s	
axis	1	0.005 s	0.003 s	
view	1	0.005 s	0.002 s	
[illegible]	1	0.003 s	0.003 s	
[illegible]	6	0.003 s	0.003 s	
grid	1	0.002 s	0.002 s	
[illegible]	1	0.001 s	0.001 s	
[illegible]	1	0.001 s	0.001 s	
[illegible]	1	0 s	0.000 s	
deg2rad	2	0 s	0.000 s	

Self time is the time spent in a function excluding the time spent in its child functions. Self time also includes overhead resulting from the process of profiling.

Profile Summary

Generated 10-Mar-2014 10:01:09 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Robot	1	1.378 s	0.471 s	
DH	1001	0.435 s	0.283 s	
[illegible]	1	0.428 s	0.378 s	
[illegible]	7007	0.044 s	0.044 s	
[illegible]	1	0.043 s	0.028 s	
[illegible]	7007	0.042 s	0.042 s	
[illegible]	7007	0.037 s	0.037 s	
axis	1	0.036 s	0.015 s	
[illegible]	7007	0.029 s	0.029 s	
[illegible]	1	0.020 s	0.019 s	
setdiff	1	0.017 s	0.004 s	
[illegible]	1	0.013 s	0.013 s	
view	1	0.005 s	0.003 s	
[illegible]	1	0.004 s	0.004 s	
grid	1	0.004 s	0.004 s	
[illegible]	1	0.002 s	0.002 s	
[illegible]	1	0.001 s	0.001 s	
[illegible]	1	0.001 s	0.001 s	
[illegible]	1	0.001 s	0.001 s	
[illegible]	1	0.001 s	0.001 s	
[illegible]	1	0 s	0.000 s	
[illegible]	1	0 s	0.000 s	
[illegible]	1	0 s	0.000 s	

Self time is the time spent in a function excluding the time spent in its child functions. Self time also includes overhead resulting from the process of profiling.

This report shows that the number of operations and the time spend by each one each [illegible]matical model, it is simply note that the mathematical [illegible] based on quaternions is simpler and faster than the [illegible]nal model based on Denavit – Hartenberg parameters.

Conclusion

In this work it has been described a rotational model of a 6 DOF robot based on quaternions, and it is compared to the traditional model based on Denavit – Hartenberg parameters (using rotation matrices).

The most important results are:

- ✓ **Lesser use of computational resources that the model based on rotation matrices due to smaller number of operations.**
- ✓ **Easier modeling.**
- ✓ **Avoids most of indetermination related to rotation matrices as "Gimbal - Lock".**

2013