Baxter Kinematic Modeling, Validation and Reconfigurable Representation

4 authors, including:

Ana Djuric
Wayne State University

44 PUBLICATIONS   **204** CITATIONS

Some of the authors of this publication are also working on these related projects:

Kinematic Modeling Of An Automated Laser Line Scanning System View project

Secondary Electron Yield View project

Baxter Kinematic Modeling, Validation and Reconfigurable Representation

Author, co-author (Do NOT enter this information. It will be pulled from participant tab in MyTechZone)

Affiliation (Do NOT enter this information. It will be pulled from participant tab in MyTechZone)

Abstract

A *collaborative robot* or _cobot_ is a robot that can safely and effectively interact with human workers while performing industrial tasks. The ability to work alongside humans has increased the importance of collaborative robots in the automation industry, as this unique feature is a much needed property among robots nowadays. Rethink Robotics has pioneered this unique discipline by building many robots including the Baxter Robot which is exclusive not only because it has collaborative properties, but because it has two arms working together, each with 7 Degrees Of Freedom. The main goal of this research is to validate the kinematic equations for the Baxter collaborative robot and develop a unified reconfigurable kinematic model for the Left and Right arms so that the calculations can be simplified. The given unified reconfigurable kinematic model represents a base for developing a unified reconfigurable Jacobian matrix, singularity conditions and dynamic model for the Left and Right arms, which will be used for control purposes. The calculation and visualization is done using reconfigurable kinematic modeling methodology and Matalb tools.

Introduction

Today in many industries collaborative robots or cobots have received a lot of attention as they are capable of working with human workers instead of replacing them. As in classic robotic scenario, companies are taking advantage of human ability to perform complex work, especially when it requires combination of multiple sensory inputs and robot's competence of continually repeating the same task to increase productivity and quality of products [3], and safety of the assembly [1]. Peshkin et al. mentioned in [4], it may generate more benefit providing physical orientation than mechanical power to the robot. In this sense, collaborative robots are looking forward to help workers to do their job efficiently, thus reducing certain health hazards such as severe back pain caused by lifting heavy items like wheels of a car in an assembly line [5].

Because collaborative robots (cobots) work by human side, a new technology that has the goal of increasing safety has been developed. In addition, they have rounded surfaces, power limit and speed limit, and collision detection [6]. Today most companies are looking forward to this technology as it saves a lot of money that is spent on building separate cages and isolated workspaces for classic robots [7]. Also the portability and capacity to work in a reconfigurable format makes collaborative robots the best choice for dynamic companies who need to change their assembly lines with ever growing customer expectations.

7/20/2015

Baxter® is the collaborative robot from Rethink. The Baxter® robot arms have seven rotational joints in each arm and these are named s0, s1, e0, e1, w0, w1, w2 starting from the shoulder respectively are shown in Figure 1, [8].
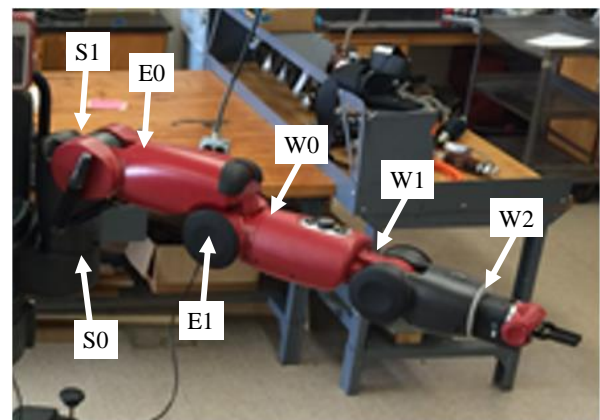

Figure 1. The arms of Baxter® robot with 7 Joints in each arm

The link lengths and link offsets of the Baxter® robot arms are measured in millimeters and are shown in Figure 2, [8].
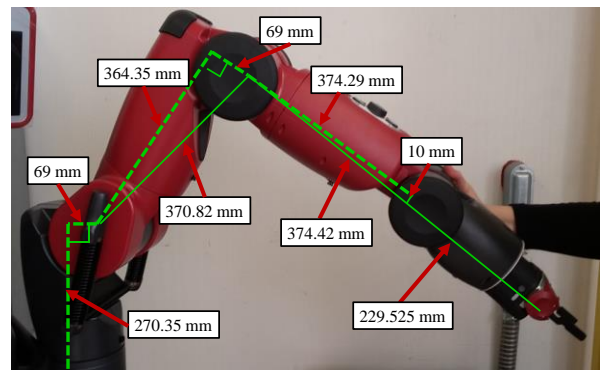

Figure 2. The link lengths and link offsets of the Baxter arms [8]

The Baxter research robot has a file in XML format detailing its parts, joints, dimensions and so on. This file is named as URDF file (Unified Robot Description Format) and is a fundamental robot model file in the Robot Operating System (ROS), [9],[10]. A kinematic model using the left arm as an example is developed in [11]. In [12] authors have used a Baxter robot as an inspector of work in a factory line, successfully using its built in vision system on the

head of the robot. Working in an assembly line, or any other factory floor, the manipulator path of a robot is very important, as it might determine the efficiency of the whole process. There are four main methods of paths generation discussed in [13]. They are,

I. Shortest path in configuration space (i.e., minimizing change in joint angle),

II. Overhead motion frequently appearing in "pick and place" paths.

III. Shortest straight-line path for the end-effector in workspace.

IV. "Curved" path for the end-effector in the workspace to exaggerate intent.

In most industrial applications, overhead motion is utilized [13]. In [14] authors have successfully tested the adaptive impedance of a Baxter robot. Due to its accuracy and sensitivity, Baxter is being suggested as a helper for surgeons during medical procedures [15]. This is a very important task due to the complexity of the process, so the cameras of the arms are used for object detection and recognition and a magnet system fixed to the end of the arm is used in object pick and place procedure [15].

Through research, companies have found out that continuous-repeated work can severely harm a human worked in the long run in a factory. Thus, companies have to address these health issues costing 13 to 20 billion dollars annually [16]. The use of collaborative robots (Cobots) in factory processing lines, helping humans to ease perform their work (Hybrid Automation), can save a great deal of money and human resource to companies [16]. All these factors highlight the importance of Kinematic modeling and validation of cobots. This was applied to a Baxter Research robot in our laboratory.

To control the Baxter Research robot, a workstation in Linux operating system with ROS-Indigo was used. Basically, the angle values are plugged in radians and saved using the "record and playback" function in the Baxter Research robot [17].

Baxter has a range of work designed to satisfy most industrial robotic applications. Baxter's workspace is shown in Figure 3 and 4 in top and side view.
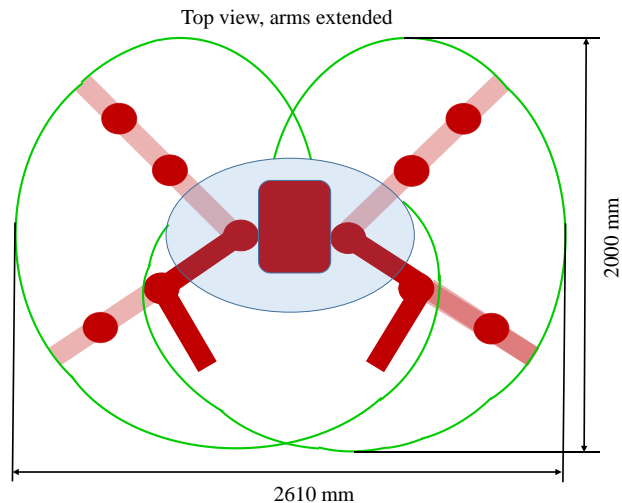

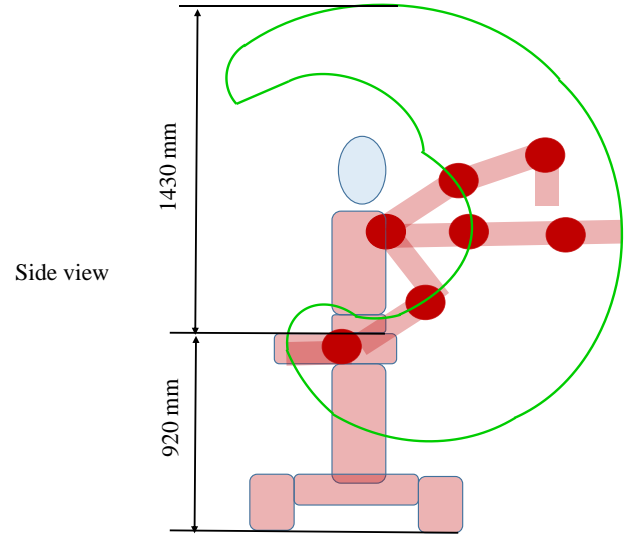Figure 3. Workspace of the Baxter robot – top view (arms extended)

7/20/2015

Figure 4. Workspace of Baxter robot – side view

Robotic kinematics is the field of study that describes the relation between two frames (joints). In order to describe the position and orientation of one frame 'n-1' into another frame 'n', it is necessary to describe the homogenous transformation matrix $^{i-1}A_i$. One way to define the homogeneous transformation matrix $^{i-1}A_i$ is through Denavit-Hartenberg (D-H) [2] parameters as showed in equation (1), where θ, α, a, and d come from D-H parameters [19], [20] and [21].

$$
^{i-1}A_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)
$$

$^{i-1}A_i$ is the link transform for the ith joint; $i = 1,2,..., n$ and n is the number of links. Once one frame 'i' can be described related to two different frames 'i-1' and 'i+1', then 'i-1' and 'i+1' can be related through one homogenous transformation matrix "$^{i-1}A_{i+1} = {}^{i-1}A_i * {}^nA_{i+1}$". The same methodology can be used to relate a robot base to the robot's end-effector as has been done in [20].

In this paper, the complete kinematic model for both arms is presented, validated and unified into one reconfigurable model. Using the unified reconfigurable model, the kinematic solution for Left and Right arms can be automatically generated by determining only the related reconfigurable parameters. The calculation and visualization is done using Matlab tools. Matlab programs are given in Appendix A and B. The reconfigurable kinematic model represents a fundamental base for future dynamic modeling and control purposes.

Baxter Kinematic model for Left and Right arms

In order to accomplish the unified kinematic model for Baxter, the kinematic diagrams for the Left and Right arms has been developed. See Figure 5 and 7.
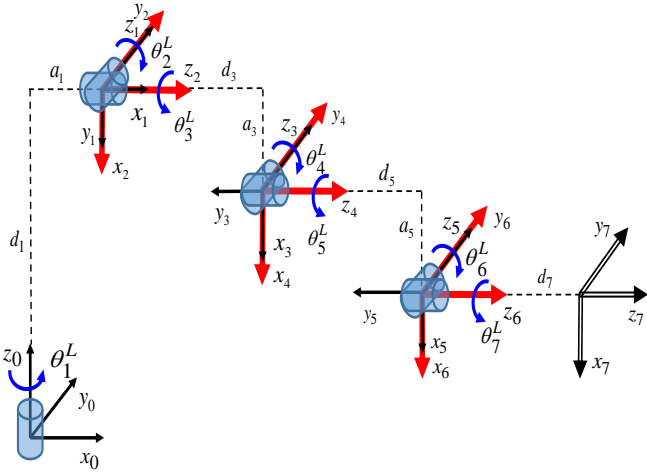
Figure 5. Baxter Left arm configuration

The D-H parameters for the Left arm are generated and presented in Table 1.

Table 1. DH parameters for the Left arm

i	d_i	θ_i	a_i	α_i
1	d_1	$\theta_1^L = 0°$	a_1	$-90°$
2	0	$\theta_2^L = 90°$	0	$90°$
3	d_3	$\theta_3^L = 0°$	a_3	$-90°$
4	0	$\theta_4^L = 0°$	0	$90°$
5	d_5	$\theta_5^L = 0°$	a_5	$-90°$
6	0	$\theta_6^L = 0°$	0	$90°$
7	d_7	$\theta_7^L = 0°$	0	$0°$

Using the D-H parameters from Table 1, the related homogeneous matrices are calculated. Equations (2-8).

$$^0A_1^L = \begin{bmatrix} \cos\theta_1^L & 0 & -\sin\theta_1^L & a_1^L\cos\theta_1^L \\ \sin\theta_1^L & 0 & \cos\theta_1^L & a_1^L\sin\theta_1^L \\ 0 & -1 & 0 & d_1^L \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$^1A_2^L = \begin{bmatrix} \cos\theta_2^L & 0 & \sin\theta_2^L & 0 \\ \sin\theta_2^L & 0 & -\cos\theta_2^L & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$^2A_3^L = \begin{bmatrix} \cos\theta_3^L & 0 & -\sin\theta_3^L & a_3\cos\theta_3 \\ \sin\theta_3^L & 0 & \cos\theta_3^L & a_3\sin\theta_3 \\ 0 & -1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$^3A_4^L = \begin{bmatrix} \cos\theta_4^L & 0 & \sin\theta_4^L & 0 \\ \sin\theta_4^L & 0 & -\cos\theta_4^L & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$^4A_5^L = \begin{bmatrix} \cos\theta_5^L & 0 & -\sin\theta_5^L & a_5^L\cos\theta_5^L \\ \sin\theta_5^L & 0 & \cos\theta_5^L & a_5^L\sin\theta_5^L \\ 0 & -1 & 0 & d_5^L \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

$$^5A_6^L = \begin{bmatrix} \cos\theta_6^L & 0 & \sin\theta_6^L & 0 \\ \sin\theta_6^L & 0 & -\cos\theta_6^L & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

$$^6A_7^L = \begin{bmatrix} \cos\theta_7^L & -\sin_7^L & 0 & 0 \\ \sin\theta_7^L & \cos\theta_7^L & 0 & 0 \\ 0 & 0 & 1 & d_7^L \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

Multiplying these seven matrices gives the forward kinematics for the Left arm of the Baxter. See Equation 9.

$$^0A_7^L = {}^0A_1^L\,{}^1A_2^L\,{}^2A_3^L\,{}^3A_4^L\,{}^4A_5^L\,{}^5A_6^L\,{}^6A_7^L \tag{9}$$

Where $^0A_7^L$ is the pose matrix of the Left arm end-effector related to the base frame.

$$^0A_7^L = \begin{bmatrix} n_x^L & s_x^L & a_x^L & p_x^L \\ n_y^L & s_y^L & a_y^L & p_y^L \\ n_z^L & s_z^L & a_z^L & p_z^L \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

The validation and visualization of the forward kinematics given in Equations (1-10), is shown in Figure 6.
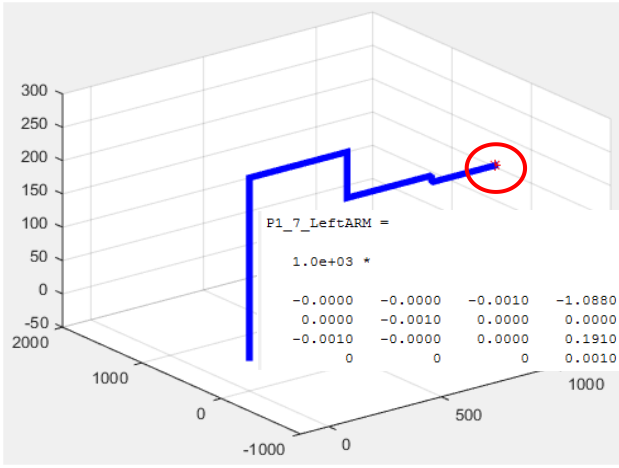
Figure 6. Baxter Left arm Matlab validation

Likewise, the kinematic diagram and D-H parameters for the Right arm are presented in Figure 7 and Table 2, respectively.
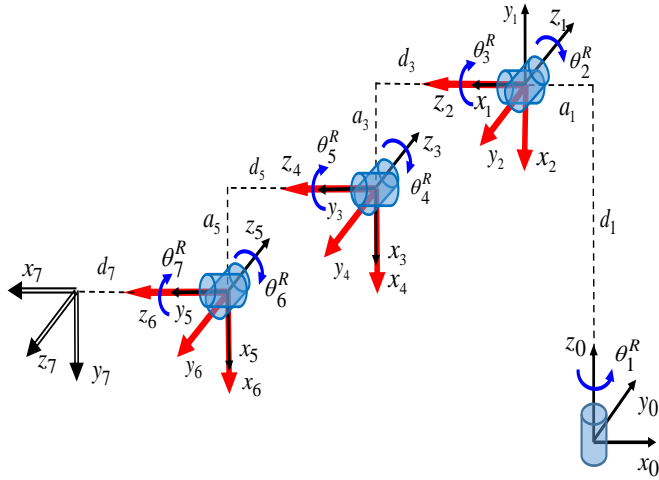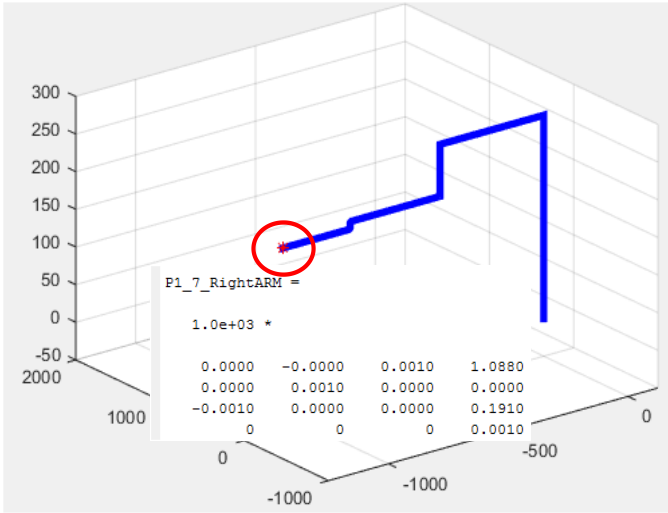

Figure 7. Baxter right arm configuration

Table 2. DH parameters for the right arm

i	d_i	θ_i	a_i	α_i
1	d_1	$\theta_1^R = 180°$	a_1	$90°$
2	0	$\theta_2^R = -90°$	0	$-90°$
3	d_3	$\theta_3^R = 0°$	a_3	$90°$
4	0	$\theta_4^R = 0°$	0	$-90°$
5	d_5	$\theta_5^R = 0°$	a_5	$90°$
6	0	$\theta_6^R = 0°$	0	$-90°$
7	d_7	$\theta_7^R = 0°$	0	$0°$

The Right forward kinematics are calculated using the robot kinematic theory. See Equations (11-19).

$$^0A_1^R = \begin{bmatrix} \cos\theta_1^R & 0 & \sin\theta_1^R & a_1^R\cos\theta_1^R \\ \sin\theta_1^R & 0 & -\cos\theta_1^R & a_1^R\sin\theta_1^R \\ 0 & 1 & 0 & d_1^R \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (11)$$

$$^1A_2^R = \begin{bmatrix} \cos\theta_2^R & 0 & -\sin\theta_2^R & 0 \\ \sin\theta_2^R & 0 & \cos\theta_2^R & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12)$$

$$^2A_3^R = \begin{bmatrix} \cos\theta_3^R & 0 & \sin\theta_3^R & a_3^R\cos\theta_3^R \\ \sin\theta_3^R & 0 & -\cos\theta_3^R & a_3^R\sin\theta_3^R \\ 0 & 1 & 0 & d_3^R \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

$$^3A_4^R = \begin{bmatrix} \cos\theta_4^R & 0 & -\sin\theta_4^R & 0 \\ \sin\theta_4^R & 0 & \cos\theta_4^R & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

$$^4A_5^R = \begin{bmatrix} \cos\theta_5^R & 0 & \sin\theta_5^R & a_5^R\cos\theta_5^R \\ \sin\theta_5^R & 0 & -\cos\theta_5^R & a_5\sin\theta_5^R \\ 0 & 1 & 0 & d_5^R \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (15)$$

$$^5A_6^R = \begin{bmatrix} \cos\theta_6 & 0 & -\sin\theta_6 & 0 \\ \sin\theta_6 & 0 & \cos\theta_6 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (16)$$

$$^6A_7^R = \begin{bmatrix} \cos\theta_7^R & -\sin\theta_7^R & 0 & 0 \\ \sin\theta_7^R & \cos\theta_7^R & 0 & 0 \\ 0 & 0 & 1 & d_7^R \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (17)$$

The forward kinematics for the right arm can be obtained by multiplying the above matrices. See Equation (18).

$$^0A_7^R = {^0A_1^R} \; {^1A_2^{R\,2}} \; A_3^{R\,3} \; A_4^{R\,4} \; A_5^{R\,5} \; A_6^{R\,6} \; A_7^R \tag{18}$$

Where $^0A_7^R$ is the pose matrix of the Right arm end-effector related to the base frame.

$$^0A_7^R = \begin{bmatrix} n_x^R & s_x^R & a_x^R & p_x^R \\ n_y^R & s_y^R & a_y^R & p_y^R \\ n_z^R & s_z^R & a_z^R & p_z^R \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{19}$$

The validation and visualization of the forward kinematics given in Equations (11-19), is shown in Figure 8.

Figure 8. Baxter Right arm Matlab validation

Figure 9 and 10 shows the visualization of the Baxter kinematic structure in 2D and 3D respectively. The two red stars represent the arms end-effector position and orientation with respect to the Baxter base frame.
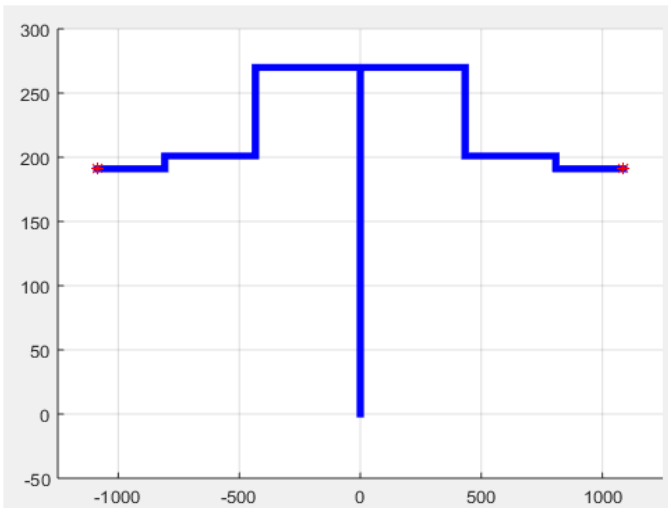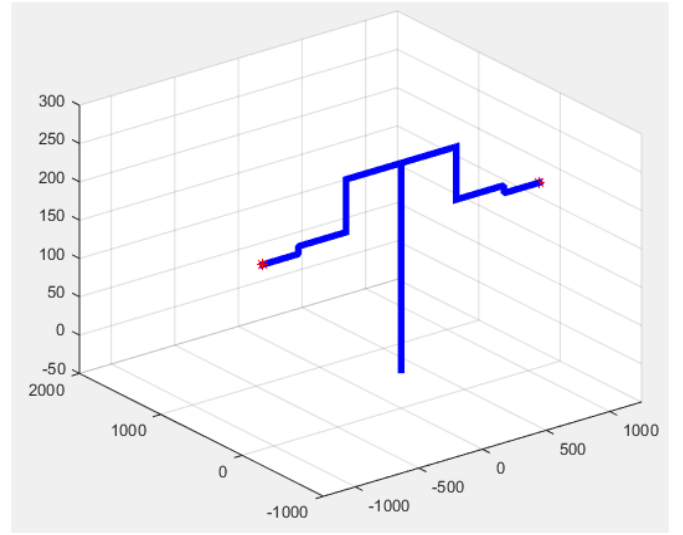

Figure 9. Baxter kinematic structure in 2D

Figure 10. Baxter kinematic structure in 3D

The connection between Baxter as a physical system and kinematic diagram as mathematical representation is given in Figure 11.

This relation is important for understanding how to model Cobots using the kinematic theory, and how to validate the results. The next step is to perform the experimental validation of the Baxter kinematic model.
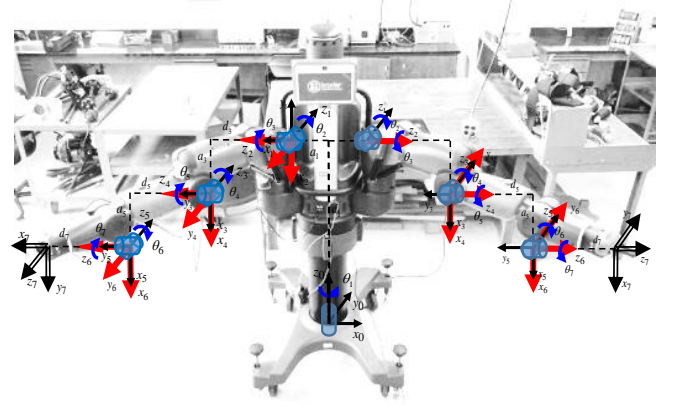

Figure 11. Baxter kinematic structure in 3D

Experimental Validation of the Kinematic Model

To validate the Baxter configurations, the research robot was connected with a workstation and relevant values were sent using the Baxter record and playback command set on python. The following python command was used to transfer the values to the robot.

"rosrun baxter_examples joint_trajectory_file_playback.py -f %filename%.rec"

Each joint has been rotated for +45 degrees and compared with its initial joint zero position. By the comparison of two positions and the Right-Hand-Rule, the positive joint direction has been determined.

Both arms are analyzed and all joint positions and orientations are graphically presented in Tables 3 and 4.

Baxter Home ZERO poison **All Joint variables are equal to ZERO**	
Joint 1 LEFT **Rotation about Z0 for 45°** **Left_s0 = 45°**	
Joint 2 LEFT **Rotation about Z1 for 45°** **Left_s1 = 45°**	
Joint 3 LEFT **Rotation about Z2 for 45°** **Left_e0 = 45°**	
Joint 4 LEFT **Rotation about Z3 for 45°** **Left_e1 = 45°**	
Joint 5 LEFT **Rotation about Z4 for 45°** **Left_w0 = 45°**	
Joint 6 LEFT **Rotation about Z5 for 45°** **Left_w1 = 45°**	
Joint 7 LEFT **Rotation about Z6 for 45°** **Left_w2 = 45°**	

Baxter Home ZERO poison **All Joint variables are equal to ZERO**	
Joint 1 RIGHT **Rotation about Z0 for 45°** **Right_s0 = 45°**	
Joint 2 RIGHT **Rotation about Z1 for 45°** **Right_s1 = 45°**	
Joint 3 RIGHT **Rotation about Z2 for 45°** **Right_e0 = 45°**	
Joint 4 RIGHT **Rotation about Z3 for 45°** **Right_e1 = 45°**	
Joint 5 RIGHT **Rotation about Z4 for 45°** **Right_w0 = 45°**	
Joint 6 RIGHT **Rotation about Z5 for 45°** **Right_w1 = 45°**	
Joint 7 RIGHT **Rotation about Z6 for 45°** **Right_w2 = 45°**	

Development of the Unified Reconfigurable Baxter Kinematic Model

The D-H parameters for both arms are presented in Table 5 and compared according to their similarities and differences. The main difference between Left and Right arms is the positive directions for each joint, which is determined with related twist angles α_i^L and α_i^R.

It is clear form Table 5 that the difference between α_i^L and α_i^R is in its signs.

To unify the Left and Right arm kinematic structures, the differences between them are expressed using seven reconfigurable parameters: K_1, K_2, K_3, K_4, K_5, K_6 and K_7 shown in Table 5.

Table 5. Baxter Reconfigurable Kinematic Model Parameters

i	d_i	θ_i	a_i	α_i^L	α_i^R	Reconfigurable Parameters
1	d_1	$\theta_1^L = 0°$, $\theta_1^R = 180°$	a_1	$-90°$	$90°$	$K_1 = \sin\alpha_1$
2	0	$\theta_2^L = 90°$, $\theta_2^R = -90°$	0	$90°$	$-90°$	$K_2 = \sin\alpha_2$
3	d_3	$\theta_3^{LR} = 0°$	a_3	$-90°$	$90°$	$K_3 = \sin\alpha_3$
4	0	$\theta_4^{LR} = 0°$	0	$90°$	$-90°$	$K_4 = \sin\alpha_4$
5	d_5	$\theta_5^{LR} = 0°$	a_5	$-90°$	$90°$	$K_5 = \sin\alpha_5$
6	0	$\theta_6^{LR} = 0°$	0	$90°$	$-90°$	$K_6 = \sin\alpha_6$
7	d_7	$\theta_7^{LR} = 0°$	0	$0°$	$0°$	$K_7 = \cos\alpha_7$

Using the new Baxter kinematic representation, the homogeneous transformation matrices are generated and presented in Equations (20-28).

$$^0A_1^{LR} = \begin{bmatrix} \cos\theta_1^{LR} & 0 & K_1\sin\theta_1^{LR} & a_1\cos\theta_1^{LR} \\ \sin\theta_1^{LR} & 0 & -K_1\cos\theta_1^{LR} & a_1\sin\theta_1^{LR} \\ 0 & K_1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20)$$

$$^1A_2^{LR} = \begin{bmatrix} \cos\theta_2^{LR} & 0 & K_2\sin\theta_2^{LR} & 0 \\ \sin\theta_2^{LR} & 0 & -K_2\cos\theta_2^{LR} & 0 \\ 0 & K_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (21)$$

$$^2A_3^{LR} = \begin{bmatrix} \cos\theta_3^{LR} & 0 & K_3\sin\theta_3^{LR} & a_3\cos\theta_3^{LR} \\ \sin\theta_3^{LR} & 0 & -K_3\cos\theta_3^{LR} & a_3\sin\theta_3^{LR} \\ 0 & K_3 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

$$^3A_4^{LR} = \begin{bmatrix} \cos\theta_4^{LR} & 0 & K_4\sin\theta_4^{LR} & 0 \\ \sin\theta_4^{LR} & 0 & -K_4\cos\theta_4^{LR} & 0 \\ 0 & K_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (23)$$

$$^4A_5^{LR} = \begin{bmatrix} \cos\theta_5^{LR} & 0 & K_5\sin\theta_5^{LR} & a_5\cos\theta_5^{LR} \\ \sin\theta_5^{LR} & 0 & -K_5\cos\theta_5^{LR} & a_5\sin\theta_5^{LR} \\ 0 & K_5 & 0 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (24)$$

$$^5A_6^{LR} = \begin{bmatrix} \cos\theta_6^{LR} & 0 & K_6\sin\theta_6^{LR} & 0 \\ \sin\theta_6^{LR} & 0 & -K_6\cos\theta_6^{LR} & 0 \\ 0 & K_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (25)$$

$$^6A_7^{LR} = \begin{bmatrix} \cos\theta_7^{LR} & K_7\sin\theta_7^{LR} & 0 & 0 \\ \sin\theta_7^{LR} & K_7\cos\theta_7^{LR} & 0 & 0 \\ 0 & 0 & K_7 & d_7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (26)$$

Forward kinematics is obtained by multiplying above seven matrices

$$^0A_7^{LR} = {}^0A_1^{LR}\, {}^1A_2^{LR}\, {}^2A_3^{LR}\, {}^3A_4^{LR}\, {}^4A_5^{LR}\, {}^5A_6^{LR}\, {}^6A_7^{R} \quad (27)$$

Where $^0A_7^{LR}$ is the pose matrix of the unified Left and Right arms end-effector related to the base frame.

$$^0A_7^{LR} = \begin{bmatrix} n_x^{LR} & s_x^{LR} & a_x^{LR} & p_x^{LR} \\ n_y^{LR} & s_y^{LR} & a_y^{LR} & p_y^{LR} \\ n_z^{LR} & s_z^{LR} & a_z^{LR} & p_z^{LR} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (28)$$

The reconfigurable kinematic solution is based on the validated Left and Right arms kinematic models.

Conclusion

Any point in the 3D space of the robot workspace can be reached using 6 Degrees of Freedom. Here, the Baxter Robot utilizes 7 DOF giving it one extra joint with redundancy. This redundancy is important in reaching hard to access points making the Baxter robot ideal for many duties such as in medical applications.

The forward Kinematics were obtained using the D-H parameters for each arm. The novel unified reconfigurable kinematic model of the Baxter Left and Right arms can reduce the number of mathematical steps involved in forward kinematics and such parameters. This process is less time consuming, and increasing the efficiency ensures less time consumption. This is very convenient for further calculations that require this as a platform to begin with. Using the Newton Euler Recursive method, the Jacobian matrix and the link's Forces and Torques can be calculated in the reconfigurable manner.

The unified reconfigurable kinematic model for the Baxter Robot has been experimentally and mathematically validated. The Matlab toolbox designed for this work can be utilized for further calculations.

References

1. Akella, P., Peshkin, M., Colgate, E.D., Wannasuphorasit, W., Nagesh, N., Wells, J., Holland, S., Pearson, T. and Peacock, B., 1999, "Cobots for the automobile assembly line", In Robotics and Automation, 1999, Proceedings. 1999 IEEE International Conference on. Vol. 1, pp: 728-733.
2. Denavit J. and Hartenberg R. S., "A Kinematic Notation for Lower-pair Mechanisms Based on Matrices" Journal of Applied Mechanics, 77: 215-221, 1955
3. Ding, H., Heyn, J., Matthias, B., and Staab, H., 2013, "Structured collaborative behavior of industrial robots in mixed human-robot environments", In Automation Science and Engineering (CASE), 2013 IEEE International Conference on, pp: 1101-1106.
4. Peshkin, M., Edward Colgate J., "Cobots (invited)", Industrial Robot, 26 (5), p. 335-341, 1999.
5. Kruger, J., Bernhardt, R., Surdilovic, D., and Spur, G., 2006," Intelligent assist systems for flexible assembly." CIRP Annals-Manufacturing Technology. Vol. 55(1), pp: 29-32.
6. Matthias, B., Kock, S., Jerregard, H., Kallman, M., Lundberg, I., and Mellander, R., 2011, "Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept." In Assembly and Manufacturing (ISAM), 2011 IEEE International Symposium on. pp: 1-6.
7. Waurzyniak, P., 2015, "Fast, lightweight robots help factories go faster." Manufacturing Engineering. Vol. 154(3), pp: 55-+. -
8. Baxter hardware specifications: http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications, Access February 2, 2016.
9. The Unified Robot Description Format (URDF): http://sdk.rethinkrobotics.com/wiki/URDF, Access February 2, 2016.
10. Robot Operating System (ROS): http://wiki.ros.org/urdf/, Access February 2, 2016.
11. Ju Z., Yang C., Ma H., "Kinematics modeling and experimental verification of Baxter robot", 33rd Chinese Control Conference (CCC), 2014.
12. Banh A., Daniel J. Rea, J. E. Y., Ehud Sharlin, "Inspector Baxter: The Social Aspects of Integrating a Robot as a Quality Inspector in an Assembly Line", In Proceedings of the 3rd ACM International Conference on Human-Agent Interaction (HAI '15). 2015.
13. Zhao M., Shome R., Yochelson I., Bekris KE Kowler, "An Experimental Study for Identifying Features of Legible Manipulator Paths", International Symposium on Experimental Robotics (ISER), 2014
14. Liang P., Yang C., Wang N., Li Z., Li R., Burdet E., "Implementation and Test of Human-Operated and Human-Like Adaptive Impedance Controls on Baxter Robot", Advances in Autonomous Robotics Systems, Volume 8717, pp 109-119, 2014.
15. Parida S., Wachs J. P., and Eugenia Cabrera M.," Dynamic Surgical Tool Tracking and Delivery System using Baxter Robot", The Summer Undergraduate Research Fellowship (SURF) Symposium, Paper 51, 2014
16. Akella P., Michael A. Peshkin, Ed Colgate, Wannasuphoprasit W., Nagesh N., Wells J., Holland S., Pearson T., Peacock B., "Cobots for the automobile assembly line**,** International Conference on Robotics and Automation, Detroit MI, 1999
17. Rethink Robotics: Baxter Research Robot Software Development Kit (SDK) Version 0.7.0 (2013)
18. Baxter Robot Hardware Architecture Overview http://www.active8robots.com/wp-content/uploads/Baxter-Hardware-Specification-Architecture-Datasheet.pdf
19. Roh, Ho-Sik, and Jin-Oh Kim. "Manipulator modeling from DH parameters", Industrial Electronics Society, 2004. IECON 2004. 30th Annual Conference of IEEE. Vol. 3. IEEE, 2004.
20. Corke P. "A Simple and Systematic Approach to Assigning Denavit–Hartenberg Parameters", Robotics, IEEE Transactions on 23.3 (2007): 590-594.
21. Manseur R.. "Robot modeling and kinematics", (Boston), Da Vinci Engineering Press, 2006 ISBN 1-58450-851-5.
22. Surdilovic, D., Schreck, G., and Schmidt, U., 2010, "Development of collaborative robots (COBOTS) for flexible human-integrated assembly automation", In Robotics (ISR), 2010 41st international symposium on and 2010 6th German Conference on Robotics (ROBOTIK). pp: 1-8, VDE.
23. Wayland M., Anders M., "Obama administration chooses Southeastern Michigan for manufacturing stimulus program", MLive Business reporter, May 28, 2014

Contact Information

Ana M. Djuric
e-mail: ana.djuric2@wayne.edu
Phone: (313)-577-5387

Definitions/Abbreviations

α_i	twist angle along X axis
α_i^L	twist angle of the Left Arm
α_i^R	twist angle of the Right Arm
d_i	Linear displacement in Z axis
a_i	Linear displacement in X axis

θ_i	twist angle along X axis	**e0, e1**	elbow 0 (Baxter joint)
			elbow1 (Baxter joint)
θ_i^L	twist angle of the Left Arm	**w0, w1, w2**	wrist 0 (Baxter joint)
			wrist 1 (Baxter joint)
θ_i^R	twist angle of the Right Arm		wrist 2 (Baxter joint)
$K_1, K_2, K_3, K_4,$ K_5, K_6, K_7	*Constants assigned related to twist angle values* $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7$.	**XML**	Extensible Markup Language
		URDF	Unified Robot Description Format
s0, s1	shoulder 0 (Baxter joint)	**ROS**	Robot Operating System
	shoulder 1 (Baxter joint)		

Appendix

<u>Appendix A:</u> **Matlab program for Baxter Left arm.**

```matlab
clc
clear all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Kinematics Baxter 7DOF Left Arm
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% D-H Parameters

a1 = 69;    % length of first arm
a2 = 0;     % length of second arm
a3 = 69;    % length of third arm
a4 = 0;     % length of fourth arm
a5 = 10;    % length of fifth arm
a6 = 0;     % length of sixth arm
a7 = 0;     % length of seventh arm

d1 = 270;   % offset of first arm
d2 = 0;     % offset of second arm
d3 = 364;   % offset of third arm
d4 = 0;     % offset of fourth arm
d5 = 375;   % offset of fifth arm
d6 = 0;     % offset of sixth arm
d7 = 280;   % offset of seventh arm

X=[0  0  -a1  -(a1+d3)  -(a1+d3)   -(a1+d3+d5)  -(a1+d3+d5)   -(a1+d3+d5+d7)];
Y=[0  0  0        0          0          0           0               0      ];
Z=[0 d1  d1       d1     d1-a3      d1-a3       d1-a3-a5        d1-a3-a5 ];

Baxter = plot3(X,Y,Z,'b','LineWidth',4);

axis([-1250 1250 -1000 2000 -50 300]);

grid on
hold('all')

ALPHA_1 = (90*pi)/180;
ALPHA_2 = (-90*pi)/180;
ALPHA_3 = (90*pi)/180;
ALPHA_4 = (-90*pi)/180;
ALPHA_5 = (90*pi)/180;
ALPHA_6 = (-90*pi)/180;
```

```
ALPHA_7 = (0*pi)/180;


theta1_0 =  180;
theta2_0 =  -90;
theta3_0 =  0;
theta4_0 =  0;
theta5_0 =  0;
theta6_0 =  0;
theta7_0 =  0;

theta1_Pendent = 0;
theta2_Pendent = 0;
theta3_Pendent = 0;
theta4_Pendent = 0;
theta5_Pendent = 0;
theta6_Pendent = 0;
theta7_Pendent = 0;

theta1 = (theta1_Pendent+theta1_0)*pi/180;
theta2 = (theta2_Pendent+theta2_0)*pi/180;
theta3 = (theta3_Pendent+theta3_0)*pi/180;
theta4 = (theta4_Pendent+theta4_0)*pi/180;
theta5 = (theta5_Pendent+theta5_0)*pi/180;
theta6 = (theta6_Pendent+theta6_0)*pi/180;
theta7 = (theta7_Pendent+theta7_0)*pi/180;

P1 = [cos(theta1), -cos(ALPHA_1)*sin(theta1), sin(ALPHA_1)*sin(theta1), a1*cos(theta1);
sin(theta1), cos(ALPHA_1)*cos(theta1), -sin(ALPHA_1)*cos(theta1), a1*sin(theta1); 0, sin(ALPHA_1),
cos(ALPHA_1), d1; 0, 0, 0, 1 ];

P2 = [cos(theta2), -cos(ALPHA_2)*sin(theta2), sin(ALPHA_2)*sin(theta2), a2*cos(theta2);
sin(theta2), cos(ALPHA_2)*cos(theta2), -sin(ALPHA_2)*cos(theta2), a2*sin(theta2); 0, sin(ALPHA_2),
cos(ALPHA_2), d2; 0, 0, 0, 1 ];

P3 = [cos(theta3), -cos(ALPHA_3)*sin(theta3), sin(ALPHA_3)*sin(theta3), a3*cos(theta3);
sin(theta3), cos(ALPHA_3)*cos(theta3), -sin(ALPHA_3)*cos(theta3), a3*sin(theta3); 0, sin(ALPHA_3),
cos(ALPHA_3), d3; 0, 0, 0, 1 ];

P4 = [cos(theta4), -cos(ALPHA_4)*sin(theta4), sin(ALPHA_4)*sin(theta4), a4*cos(theta4);
sin(theta4), cos(ALPHA_4)*cos(theta4), -sin(ALPHA_4)*cos(theta4), a4*sin(theta4); 0, sin(ALPHA_4),
cos(ALPHA_4), d4; 0, 0, 0, 1 ];

P5 = [cos(theta5), -cos(ALPHA_5)*sin(theta5), sin(ALPHA_5)*sin(theta5), a5*cos(theta5);
sin(theta5), cos(ALPHA_5)*cos(theta5), -sin(ALPHA_5)*cos(theta5), a5*sin(theta5); 0, sin(ALPHA_5),
cos(ALPHA_5), d5; 0, 0, 0, 1 ];

P6 = [cos(theta6), -cos(ALPHA_6)*sin(theta6), sin(ALPHA_6)*sin(theta6), a6*cos(theta6);
sin(theta6), cos(ALPHA_6)*cos(theta6), -sin(ALPHA_6)*cos(theta6), a6*sin(theta6); 0, sin(ALPHA_6),
cos(ALPHA_6), d6; 0, 0, 0, 1 ];

P7 = [cos(theta7), -cos(ALPHA_7)*sin(theta7), sin(ALPHA_7)*sin(theta7), a7*cos(theta7);
sin(theta7), cos(ALPHA_7)*cos(theta7), -sin(ALPHA_7)*cos(theta7), a7*sin(theta7); 0, sin(ALPHA_7),
cos(ALPHA_7), d7; 0, 0, 0, 1 ];

P1_7_LeftARM = P1*P2*P3*P4*P5*P6*P7
HOME = plot3(P1_7_LeftARM(1,4), P1_7_LeftARM(2,4), P1_7_LeftARM(3,4), 'r*');
```

Appendix B:

Matlab program for Baxter Right arm.

```matlab
clc
clear all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Kinematics Baxter 7DOF Left Arm
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% D-H Parameters

a1 = 69;     % length of first arm
a2 = 0;      % length of second arm
a3 = 69;     % length of third arm
a4 = 0;      % length of fourth arm
a5 = 10;     % length of fifth arm
a6 = 0;      % length of sixth arm
a7 = 0;      % length of seventh arm

d1 = 270;    % offset of first arm
d2 = 0;      % offset of second arm
d3 = 364;    % offset of third arm
d4 = 0;      % offset of fourth arm
d5 = 375;    % offset of fifth arm
d6 = 0;      % offset of sixth arm
d7 = 280;    % offset of seventh arm

X=[0   0   a1   a1+d3    a1+d3    a1+d3+d5    a1+d3+d5    a1+d3+d5+d7];
Y=[0   0   0     0        0         0           0            0       ];
Z=[0  d1  d1    d1      d1-a3     d1-a3      d1-a3-a5     d1-a3-a5  ];

Baxter = plot3(X,Y,Z,'b','LineWidth',4);

axis([-1250 1250 -1000 2000 -50 300]);

grid on
hold('all')

ALPHA_1 = (-90*pi)/180;
ALPHA_2 = (90*pi)/180;
ALPHA_3 = (-90*pi)/180;
ALPHA_4 = (90*pi)/180;
ALPHA_5 = (-90*pi)/180;
ALPHA_6 = (90*pi)/180;
ALPHA_7 = (0*pi)/180;

theta1_0 =  0;
theta2_0 =  90;
theta3_0 =  0;
theta4_0 =  0;
theta5_0 =  0;
theta6_0 =  0;
theta7_0 =  0;

theta1_Pendent = 0;
theta2_Pendent = 0;
theta3_Pendent = 0;
theta4_Pendent = 0;
theta5_Pendent = 0;
```

7/20/2015

```
theta6_Pendent = 0;
theta7_Pendent = 0;

theta1 = (theta1_Pendent+theta1_0)*pi/180;
theta2 = (theta2_Pendent+theta2_0)*pi/180;
theta3 = (theta3_Pendent+theta3_0)*pi/180;
theta4 = (theta4_Pendent+theta4_0)*pi/180;
theta5 = (theta5_Pendent+theta5_0)*pi/180;
theta6 = (theta6_Pendent+theta6_0)*pi/180;
theta7 = (theta7_Pendent+theta7_0)*pi/180;

P1 = [cos(theta1), -cos(ALPHA_1)*sin(theta1), sin(ALPHA_1)*sin(theta1), a1*cos(theta1);
sin(theta1), cos(ALPHA_1)*cos(theta1), -sin(ALPHA_1)*cos(theta1), a1*sin(theta1); 0, sin(ALPHA_1),
cos(ALPHA_1), d1; 0, 0, 0, 1 ];

P2 = [cos(theta2), -cos(ALPHA_2)*sin(theta2), sin(ALPHA_2)*sin(theta2), a2*cos(theta2);
sin(theta2), cos(ALPHA_2)*cos(theta2), -sin(ALPHA_2)*cos(theta2), a2*sin(theta2); 0, sin(ALPHA_2),
cos(ALPHA_2), d2; 0, 0, 0, 1 ];

P3 = [cos(theta3), -cos(ALPHA_3)*sin(theta3), sin(ALPHA_3)*sin(theta3), a3*cos(theta3);
sin(theta3), cos(ALPHA_3)*cos(theta3), -sin(ALPHA_3)*cos(theta3), a3*sin(theta3); 0, sin(ALPHA_3),
cos(ALPHA_3), d3; 0, 0, 0, 1 ];

P4 = [cos(theta4), -cos(ALPHA_4)*sin(theta4), sin(ALPHA_4)*sin(theta4), a4*cos(theta4);
sin(theta4), cos(ALPHA_4)*cos(theta4), -sin(ALPHA_4)*cos(theta4), a4*sin(theta4); 0, sin(ALPHA_4),
cos(ALPHA_4), d4; 0, 0, 0, 1 ];

P5 = [cos(theta5), -cos(ALPHA_5)*sin(theta5), sin(ALPHA_5)*sin(theta5), a5*cos(theta5);
sin(theta5), cos(ALPHA_5)*cos(theta5), -sin(ALPHA_5)*cos(theta5), a5*sin(theta5); 0, sin(ALPHA_5),
cos(ALPHA_5), d5; 0, 0, 0, 1 ];

P6 = [cos(theta6), -cos(ALPHA_6)*sin(theta6), sin(ALPHA_6)*sin(theta6), a6*cos(theta6);
sin(theta6), cos(ALPHA_6)*cos(theta6), -sin(ALPHA_6)*cos(theta6), a6*sin(theta6); 0, sin(ALPHA_6),
cos(ALPHA_6), d6; 0, 0, 0, 1 ];

P7 = [cos(theta7), -cos(ALPHA_7)*sin(theta7), sin(ALPHA_7)*sin(theta7), a7*cos(theta7);
sin(theta7), cos(ALPHA_7)*cos(theta7), -sin(ALPHA_7)*cos(theta7), a7*sin(theta7); 0, sin(ALPHA_7),
cos(ALPHA_7), d7; 0, 0, 0, 1 ];

P1_7 = P1*P2*P3*P4*P5*P6*P7
HOME = plot3(P1_7(1,4), P1_7(2,4), P1_7(3,4), 'r*');
```

7/20/2015