

5. Experimentos e Resultados

Inicialmente, utilizando a sequência de montagem dos servomotores na placa base, foi criada uma tabela fazendo a relação entre os servomotores e os elos que cada um aciona. Após esta relação criada, foi criado um programa para a realização da calibração dos tendões, com a finalidade de deixá-los pré-tensionados. Como a fixação dos tendões foi realizada de forma desigual, a variação angular para o pré-tensionamento dos tendões é individual e singular, conforme ilustra a Tabela 7. Isto significa que a montagem dos tendões nas polias dos servos não precisa ser feita com alta acurácia, uma vez que o sistema será calibrado. A escolha das IDs dos servos utilizados pode ser aleatória pois é possível determinar a configuração das IDs através do *software* e por este motivo, as IDs contidas na Tabela 7 não seguem uma sequência lógica definida.

Tabela 5-1 – Pré-tensionamento dos Tendões

Servo ID	Elo ID	Grau (°)	Rotação (0-1023)
1	3	58	400
2	2	41	630
4	1	64	320
8	3	44	400
9	2	63	315
10	1	67	330
11	3	58	590
12	2	60	305
13	1	58	300
14	3	88	530
17	2	66	470
18	1	64	320

A escolha dos valores de tensionamento foi baseada no posicionamento dos três elos na vertical, de forma que eles atingissem uma rigidez final desejada, suficiente para que o sistema pudesse ser disposto na vertical sem possibilidade de flambagem. Já para a velocidade de movimentação dos servomotores, foi criada uma opção de controle de trajetórias a taxas ajustáveis, pois a velocidade padrão dos servos é alta. A taxa de transmissão escolhida como *default* foi a de endereço 0X01, ou seja, 1000000 BPS. Uma interface foi desenvolvida em LabView (Fig. 45) que realiza a calibração e pré-tensionamento dos tendões.

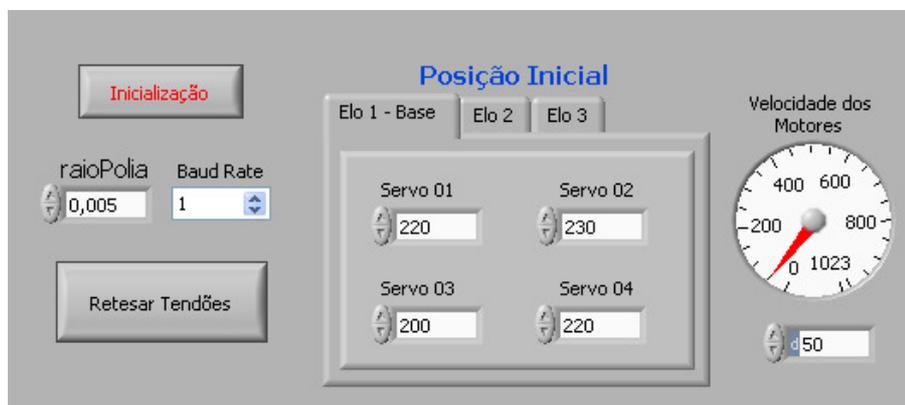


Figura 45 – Interface de Calibração

Na Figura 45 pode se notar que os valores da Tabela 7, referentes à posição inicial do tensionamento dos cabos, podem ser alterados e redefinidos. Após o pré-tensionamento realizado, o manipulador passa a ter um posicionamento inicial definido. O “raioPolia” está em metros, o “Baud Rate” segue a relação apresentada na Tabela 6, a “Posição Inicial” é relativa à posição desejada de cada servo motor (variando de 0 até 1023) e a “Velocidade dos Motores” segue a relação de 1:0,12 em rpm, ou seja, a velocidade máxima de 1023 corresponde a, aproximadamente, 122 rpm. A Figura 46 ilustra o manipulador pré-tensionado, onde pode ser observado que o manipulador possui uma configuração inicial na vertical, sem erros significativos.

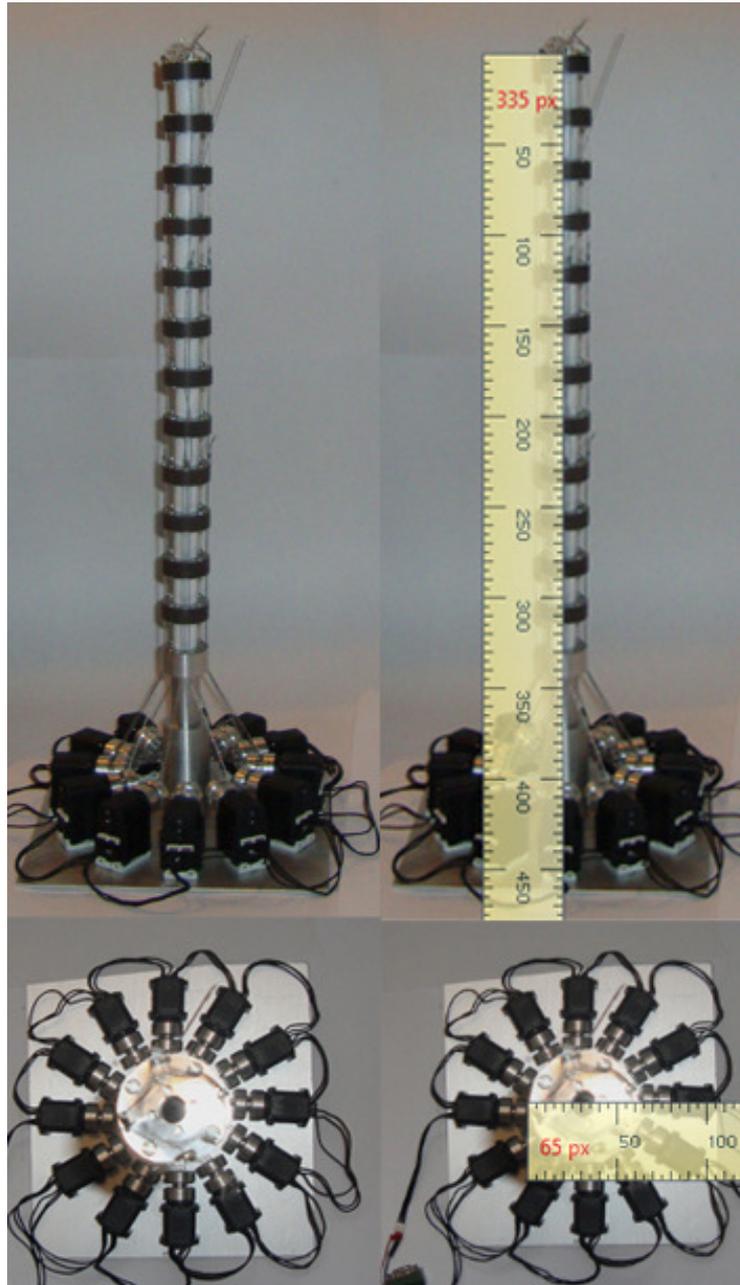


Figura 46 – Posição Inicial do Manipulador

Em seguida, foi desenvolvido um software para realizar o controle de movimentação para os DoF do manipulador na ferramenta *LabView*. O software em questão foi desenvolvido utilizando um recurso do *LabView*, chamado *MatLab Script*, onde é possível inserir linhas de código do *MatLab* no *LabView*. Dessa forma, foi possível inserir o Jacobiano encontrado para um elo com 4 vértebras no *LabView*. Com o intuito de reduzir o custo computacional, ao invés de calcular

o Jacobiano no *LabView*, este foi calculado no *MatLab* e em seguida inserido no *software* na mesma forma que é apresentado na Eq. 3.35. O valor de r é de $1,905 \times 10^{-2} \text{m}$, h_a é $1,267 \times 10^{-2} \text{m}$, h_m é $2,54 \times 10^{-4} \text{m}$, e o valor de h_c é de $2,54 \times 10^{-2} \text{m}$. Após a criação de uma função de cálculo do Jacobiano de 4 vértebras, foi criada outra função para compor o Jacobiano do manipulador, o qual é composto por 3 elos de 4 vértebras cada. Um ponto que deve ser notado é que ao se atuar o elo 1, os tendões dos elos seguintes também devem ser atuados de forma a não serem relaxados ou tensionados excessivamente por motivos de movimentação do manipulador. Isso ocorre pois o sistema é acoplado, e a movimentação de um elo requer que os tendões dos elos seguintes sejam acionados.

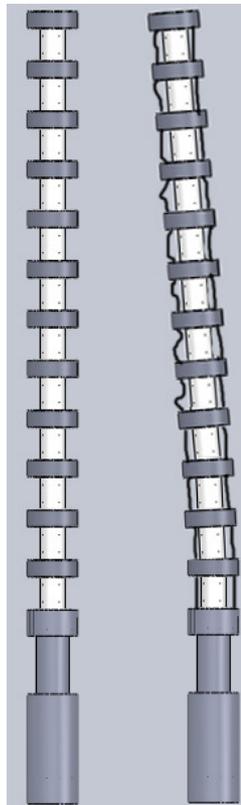


Figura 47 – Tendões nos Elos Consequentes

A Figura 47 ilustra o que ocorreria se apenas os tendões de 1 elo fossem movimentados. Note que os demais tendões em um dos lados ficariam relaxados, e do outro lado sobre-tensionados, inviabilizando o controle ou danificando a estrutura. Para evitar isso, foi criada uma função que calcula o Jacobiano

completo do manipulador, considerando que cada entrada dos DoFs (θ_1 , θ_2 ou d_3) em um elo afetam os DoFs dos seguintes. A função em questão foi desenvolvida para o caso em estudo, ou seja, um manipulador com 3 elos e cada elo contendo 4 vértebras. Entretanto, ela é de simples entendimento e pode ser expandida para qualquer quantidade de elos e vértebras necessária.

Foi desenvolvida uma função (GetID) para selecionar a ID do servo a ser atuado através dos nomes dos elos e dos servos. Dessa forma, foi criada uma tabela com as IDs originais de cada servomotor que relaciona o servomotor com sua posição de atuação no manipulador. A Tabela 8 ilustra as IDs e suas relações. Após ser criada a matriz de IDs, foi definida a nomenclatura utilizada para fazer a distinção dos servomotores. Inicialmente, é descrito o elo e, em seguida, o servo, ou seja, Elo0X – Servo0Y, onde X varia entre 1 e 3 e Y entre 1 e 4.

Tabela 5-2 – Relação entre IDs e Elos

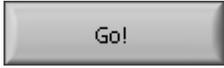
	Servo 01	Servo 02	Servo 03	Servo 04
Elo 01	4	10	13	18
Elo 02	2	9	12	17
Elo 03	8	11	14	1

Após a seleção dos servomotores, foi elaborado um programa que define a posição inicial de pré-tensionamento. Dessa forma, pode-se definir a posição de tensionamento na inicialização do *software* e, em seguida, com um apertar de botão, todos os servomotores são atuados até a posição estabelecida.

Para iniciar o programa foi criado um botão . Com este botão, é ativada a função `dxl_initialize`, onde se verificam se as conexões entre os servomotores e o USB2Dynamixel estão corretas e verifica se existe tensão elétrica em todos os servomotores. Ao pressionar este botão, ele muda seu aspecto para , o qual ao ser clicado, gera a finalização do

programa. A finalização do *software* consiste em levar todos os servomotores à posição zero, sem pré-tensionamento.

No programa também é apresentada a posição atual dos servomotores, bem como a angulação e os deslocamentos aplicados em cada um dos DoFs.

O botão  tem como finalidade ativar as alterações realizadas nas angulações e velocidade. É apresentada uma barra de progressão com o intuito de aguardar todas as instruções a serem enviadas e recebidas entre o sistema. Enquanto ocorre a progressão, o sistema impossibilita a utilização da interface com o intuito de evitar perda de dados ou entradas indevidas durante a comunicação.

O programa apresenta três opções de variação de DoF para cada elo. Os comandos D_theta1 e D_theta2 alteram a angulação dos DoF θ_1 e θ_2 , enquanto que o comando D_prism altera a variação em d_3 .

A Figura 48 apresenta a tela principal do programa, ilustrando os textos explicativos anteriores. O Anexo VII apresenta a programação principal realizada para o desenvolvimento do *software*.

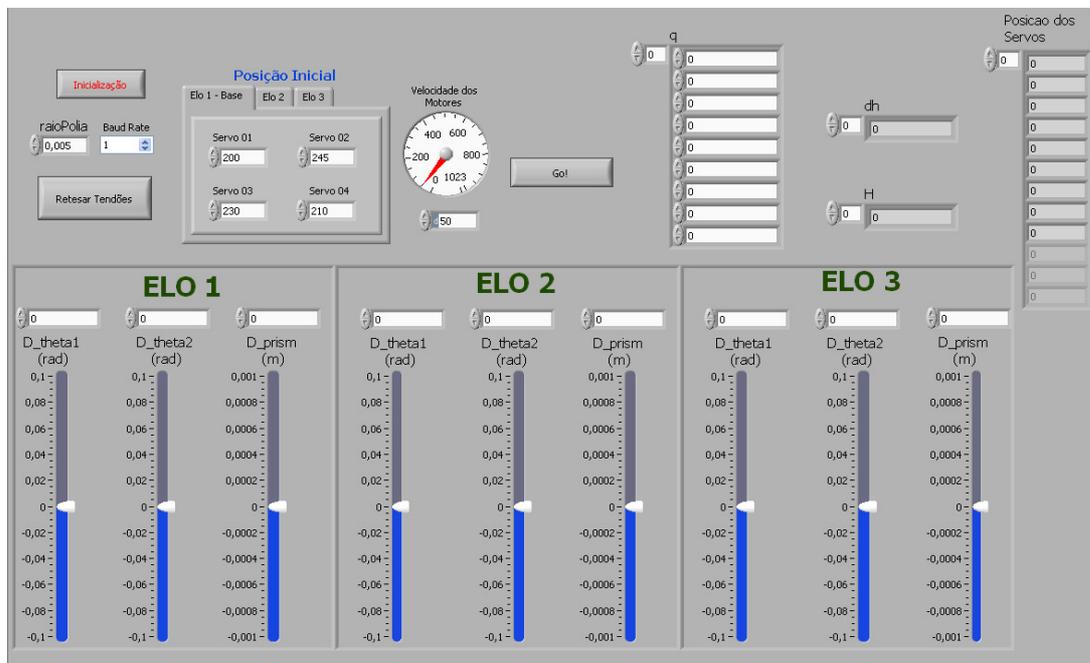


Figura 48 – Tela Principal do *Software* de Controle

Após o desenvolvimento deste programa, o manipulador foi testado para realizar uma comparação entre os DoFs desejados e os obtidos. Foram realizados diversos testes de forma a verificar a repetibilidade do sistema, bem como sua precisão. Para a realização dos testes, foi desenvolvida uma base para suporte de 2 câmeras fotográficas para verificação do posicionamento em 3D. Este procedimento pode ser adotado por ser uma análise estática, e não se fazendo, por isso necessário a utilização de medições em tempo real da extremidade usando câmeras de vídeo. Para uma melhor análise dos resultados, foi inserido um alvo alongado na extremidade do manipulador com o intuito de apresentar uma superfície retilínea e, assim, poder calcular o ângulo gerado pela extremidade do manipulador e conseqüentemente realizar a comparação com o ângulo desejado. Também foi calculado o erro percentual $E_{\%}$ (Eq. 4.1) onde E_{exp} é o valor experimental encontrado e E_{des} é o valor teórico desejado, para cada medida.

$$E_{\%} = \frac{(E_{\text{exp}} - E_{\text{des}})}{E_{\text{des}}} \quad (4.1)$$

As rotações desejadas em cada (D_{Theta1} e D_{Theta2}) foram definidas pelos valores discretos $\pm 0,2\text{rad}$, $\pm 0,3\text{rad}$ e $\pm 0,4\text{rad}$, o que gera um ângulo em cada vértebra por elo de $\pm 0,05\text{ rad}$, $\pm 0,075\text{rad}$ e $\pm 0,1\text{ rad}$, respectivamente. As entradas no manipulador divergem da variação em θ_1 e θ_2 pelo fato de o elo possuir 4 vértebras. Desta forma, o valor de entrada é dividido pelo número de vértebras do elo, fazendo assim com que o ângulo também o seja.

Para realizar a medição dos ângulos gerados, foi utilizado um recurso computacional que simula um transferidor. É um *software* livre chamado Scale2.0³, que permite que o usuário insira uma régua ou transferidor digital na tela do computador, possibilitando o cálculo de medidas e ângulos, vide Figura 49. A régua pode ser utilizada em diversas escalas pré-definidas, tais como pixels, cm, polegadas, e outras. Esta ferramenta também possui uma lupa eletrônica, o

³ Desenvolvido pelo PhD. Ricardo Sgrillo. <http://www.sgrillo.net/scale.htm>

que facilita o correto posicionamento. Com o auxílio do transferidor digital, foi possível calcular as angulações geradas a cada variação de entrada descrita. O erro máximo de medição é 1° , ou seja, $0,01\text{rad}$. Após a realização de 7 testes individuais para as três entradas e para os dois DoFs (D_Theta1 e D_Theta2), conseguiu-se chegar a uma comparação entre os valores desejados e os medidos.

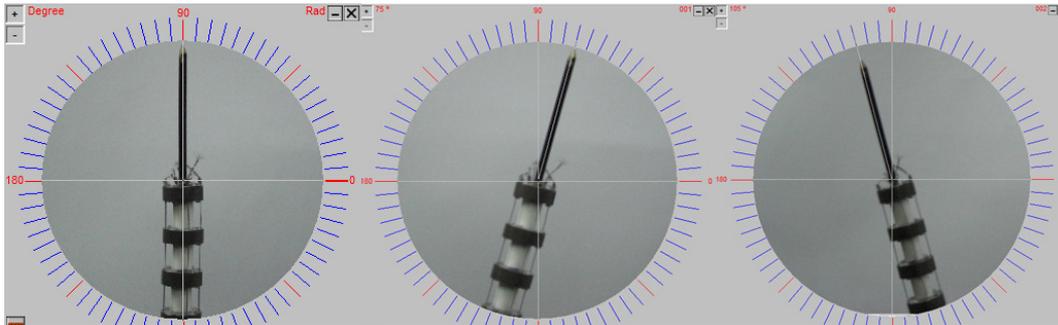


Figura 49 – Medições (Retesado e com $\pm 0,4\text{rad}$ no Elo 01)

As tabelas 5.3 e 5.4 apresentamos resultados encontrados para D_Theta1 e D_Theta2 nos testes realizados. Os valores encontrados para variações de θ_1 são apresentados na tabela 5.3 e os valores para θ_2 , na Tabela 5.4. Os valores das colunas “Posição Desejada” e “Posição Medida” estão em radianos. Pode ser observado que os resultados encontrados para os elos 01 e 02 estão dentro de uma margem de erro aceitável variando entre 1% e 3%, os erros quadráticos médios, enquanto que o para o elo 03 o erro médio fica entre 5% a 11%. Este fato se deve a condição do efeito da gravidade. Uma forma de corrigir este problema é a inserção de um controle de malha fechada no sistema. À frente, será apresentada uma simulação assumindo um controle por torque computado deste manipulador que executa a correção deste erro.

Tabela 5-3 – Variações de θ_1

Elos	Posição Desajada	Teste 01			Teste 02			Teste 03			Teste 04		Erro Absoluto
		Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	
1	0,2	0,1920	-0,0417	0,0040	0,1920	-0,0417	0,0040	0,2094	0,0451	0,0047	0,1920	-0,0417	0,0040
	0,3	0,2967	-0,0111	0,0016	0,3142	0,0451	0,0071	0,3442	0,0451	0,0071	0,2967	-0,0111	0,0016
	0,4	0,3840	-0,0417	0,0080	0,4189	0,0451	0,0094	0,4363	0,0833	0,0182	0,4014	0,0036	0,0007
	-0,2	-0,2094	0,0451	0,0047	-0,1920	-0,0417	0,0040	-0,1920	-0,0417	0,0040	-0,2094	0,0451	0,0047
2	-0,3	-0,2967	-0,0111	0,0016	-0,3142	0,0451	0,0071	-0,3142	0,0451	0,0071	-0,2967	-0,0111	0,0016
	-0,4	-0,3665	-0,0913	0,0167	-0,4189	0,0451	0,0094	-0,4363	0,0833	0,0182	-0,3840	-0,0417	0,0080
	0,2	0,1920	-0,0417	0,0040	0,2094	0,0451	0,0047	0,1920	-0,0417	0,0040	0,2094	0,0451	0,0047
	0,3	0,2967	-0,0111	0,0016	0,2967	-0,0111	0,0016	0,2793	-0,0743	0,0104	0,2443	-0,2278	0,0278
3	0,4	0,3840	-0,0417	0,0080	0,3142	-0,2732	0,0429	0,316	-0,2062	0,0342	0,3142	-0,2732	0,0429
	-0,2	-0,1920	-0,0417	0,0040	-0,1571	-0,2732	0,0215	-0,1571	-0,2732	0,0215	-0,1571	-0,2732	0,0215
	-0,3	-0,2443	-0,2278	0,0278	-0,2269	-0,3222	0,0366	-0,2443	-0,2278	0,0278	-0,2094	-0,2732	0,0451
	-0,4	-0,3316	-0,2062	0,0342	-0,2793	-0,4324	0,0504	-0,2967	-0,3481	0,0516	-0,2793	-0,4324	0,0504
Teste 05													
Elos	Posição Desajada	Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	Erro Absoluto	Desvio Padrão	Variação	
		0,2	0,2094	0,0451	0,0047	0,1920	-0,0417	0,0040	0,1920	-0,0417	0,0040	0,0085	0,0001
		0,3	0,3142	0,0451	0,0071	0,2967	-0,0111	0,0016	0,2793	-0,0743	0,0104	0,0132	0,0002
		0,4	0,3840	-0,0417	0,0080	0,3840	-0,0417	0,0080	0,3840	-0,0417	0,0080	0,0212	0,0004
1	-0,2	-0,1920	-0,0417	0,0040	-0,1920	-0,0417	0,0040	-0,2094	0,0451	0,0047	0,0093	0,0001	
	-0,3	-0,2793	-0,0743	0,0104	-0,2793	-0,0743	0,0104	-0,2967	-0,0111	0,0016	0,0143	0,0002	
	-0,4	-0,3665	-0,0913	0,0167	-0,3665	-0,0913	0,0167	-0,4014	0,0036	0,0007	0,0282	0,0008	
	0,2	0,1920	-0,0417	0,0040	0,2094	0,0451	0,0047	0,1920	-0,0417	0,0040	0,0093	0,0001	
2	0,3	0,2967	-0,0111	0,0016	0,2967	-0,0111	0,0016	0,2967	-0,0111	0,0016	0,0101	0,0001	
	0,4	0,4189	0,0451	0,0094	0,3840	-0,0417	0,0080	0,3840	-0,0417	0,0080	0,0225	0,0005	
	-0,2	-0,1920	-0,0417	0,0040	-0,1920	-0,0417	0,0040	-0,2094	0,0451	0,0047	0,0085	0,0001	
	-0,3	-0,2967	-0,0111	0,0016	-0,2967	-0,0111	0,0016	-0,2967	-0,0111	0,0016	0,0166	0,0003	
3	-0,4	-0,3840	-0,0417	0,0080	-0,4189	0,0451	0,0094	-0,4014	0,0036	0,0007	0,0194	0,0004	
	0,2	0,1745	-0,1459	0,0127	0,1745	-0,1459	0,0127	0,1745	-0,1459	0,0127	0,0066	0,0000	
	0,3	0,2443	-0,2278	0,0278	0,2618	-0,1459	0,0191	0,2443	-0,2278	0,0278	0,0101	0,0001	
	0,4	0,3142	-0,2732	0,0429	0,3316	-0,2062	0,0342	0,3142	-0,2062	0,0342	0,0132	0,0001	
Elos	Posição Desajada	Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	Erro Absoluto	Posição Medida	Erro Percentual	Erro Absoluto	Desvio Padrão	Variação	
		0,2	-0,1745	-0,0127	-0,0016	-0,1571	-0,2732	-0,0215	-0,1745	-0,1459	-0,0127	0,0093	0,0001
		0,3	-0,2443	-0,2278	-0,0278	-0,2269	-0,3222	-0,0366	-0,2269	-0,3222	-0,0366	0,0132	0,0002
		0,4	-0,2443	-0,2278	-0,0278	-0,2967	-0,3481	-0,0516	-0,3142	-0,2732	-0,0429	0,0222	0,0005

Após as verificações do posicionamento estático terem sido executadas, foi desenvolvida uma simulação de controle para o manipulador. O intuito inicial seria realizar alguns testes aplicando a gravidade para o manipulador na horizontal e a execução simulada de uma determinada tarefa demonstrando suas características. Inicialmente, foi desenvolvido um programa no *MatLab* com o intuito de simular o efeito da gravidade. Utilizando a *toolbox Robot*⁴, foi possível desenvolver todas as análises necessárias para simular o manipulador. A *toolbox* permite a criação de um manipulador a partir de uma matriz contendo o D-H ou de uma forma mais completa, onde se define elo por elo, indicando se o mesmo é prismático ou rotativo, incluindo, também, valores tais como, peso de cada elo, centro de massa, matriz de inércia, redução utilizada entre elos, inércia dos atuadores, atrito viscoso e atrito de Coulomb, entre outros. Além disso, é possível calcular a cinemática direta e/ou inversa com apenas um comando (*fkine* e *ikine*, respectivamente), bem como o Jacobiano em relação à base, a dinâmica direta, a dinâmica inversa, matriz de inércia, força de Coriolis, inserir o efeito da gravidade, entre diversas outras opções – conforme ilustra o Anexo IX.

Para a criação da simulação no *Simulink*, foi utilizado o controle de torque computado. O controle de torque computado é baseado na compensação dinâmica do manipulador e tem como objetivo, descobrir a dinâmica linear e desacoplada em um sistema de malha fechada para compensar as variações não levadas em conta pelo modelo dinâmico, devidas aos erros de estimação, atrito, variações na carga, entre outras. Assim, é possível controlar cada junta independentemente. Além da implementação da linearização e do desacoplamento dinâmico, é inserido um controlador linear com realimentação Proporcional-Derivativa, com o intuito de filtrar perturbações e corrigir desvios nas trajetórias escolhidas. A Figura 50 exemplifica a lei de controle por torque computado, onde o símbolo de * tem o significado de “desejado”.

⁴ Desenvolvida por Peter Corke. http://www.petercorke.com/Robotics_Toolbox.html

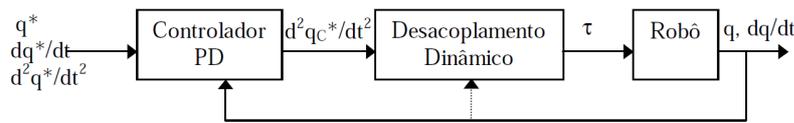


Figura 50 – Malha de Controle por Torque Computado

Para o desenvolvimento de uma simulação do manipulador no *MatLab*, foi utilizada a *toolbox Robot* e, a partir dela, pode-se construir um robô desenvolvido através dos parâmetros de D-H apresentados na Tabela 4, onde cada elo indica um DoF da vértebra. Foram atribuídas as respectivas massas relativas a cada componente (vide capítulo 2), de forma que para θ_1 , θ_2 e θ_4 foram assumidas metade da massa de uma membrana e para θ_5 , foi atribuído o valor de metade da membrana mais a massa do anel, enquanto que para o d_3 , foi atribuído o valor da massa da coluna. O valor de metade da massa da membrana é inserido devido aos DoFs, θ_1 e θ_5 , pertencerem a apenas uma membrana e possuírem a massa distribuída uniformemente. Da mesma forma para os θ_2 e θ_4 . O centro de massa e a matriz de inércia, para cada DoF, foram inseridos na construção do manipulador. Seus respectivos valores foram obtidos, através de análises realizadas no *SolidWorks*. A transmissão, bem como a inércia dos servomotores, também, foram adotadas. Também foi inserida uma pequena taxa de atrito seco para indicar o atrito entre os tendões e os componentes do manipulador. Além da criação do manipulador, foi criada outra função que gera a matriz de rigidez do sistema. Como a rigidez em d_3 é muito superior à encontrada para os outros DoFs, na ordem de 2 casas acima, ela será assumida como tendendo a infinito, para que seja possível realizar uma simulação coerente.

Logo, foi possível desenvolver um programa para realizar a simulação do sistema atuado apenas com a força da gravidade. Para que as respostas possuíssem uma melhor visualização, foram feitos testes com o manipulador na direção de z e, também, com ele na direção de x. A Figura 51 ilustra o manipulador sobre efeito da gravidade com a base rotacionada em 90° , em direção ao eixo x (realizada no *MatLab*) e, uma análise de deslocamento realizada no *Simulation*.

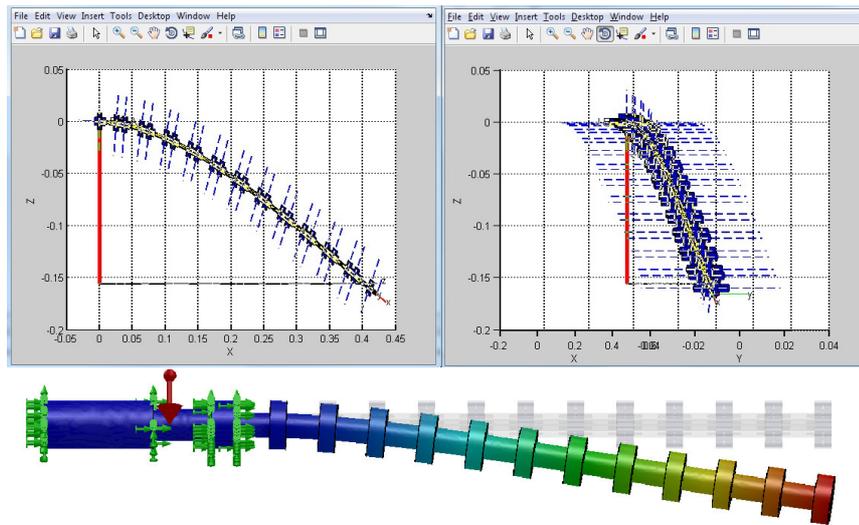


Figura 51 – Gravidade

Esta análise pode ser comparada com o manipulador real. A base do manipulador real foi colocada na direção do eixo x. Assim, foi possível fazer uma comparação entre a angulação da simulação e a angulação do experimento. A simulação apresentou uma curvatura de aproximadamente 29° , enquanto que, para o experimento, a curvatura encontrada foi de 25° . O erro é considerado dentro do admissível, pois o erro da resposta gerada pelo programa Scale2.0 é de 1° .

Para o controle de torque computado, foi utilizado o *Simulink* com as funções da *toolbox Robot* e o programa desenvolvido para a criação do modelo do manipulador no *MatLab*. A Figura 52 apresenta o diagrama de blocos utilizado para a geração da simulação.

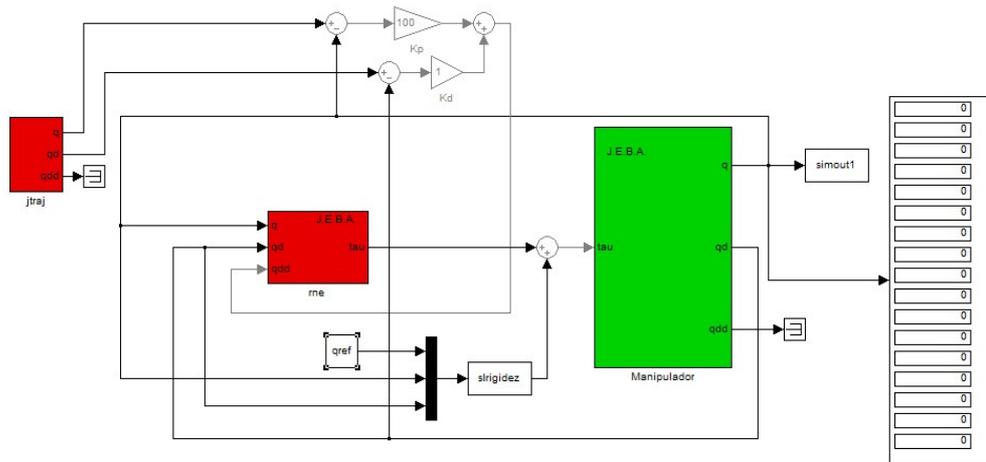


Figura 52 – Controle por Torque Computado

A função “jtraj” possui parâmetros que incluem os valores iniciais e finais de cada elemento, no caso, os DoFs e o tempo de movimentação. As velocidades inicial e final são determinadas como zero. A função “rne” – *Recursive Newton-Euler* – computa a dinâmica inversa do manipulador via algoritmo recursivo de Newton-Euler. A função “s/rigidez” foi uma função desenvolvida para inserir a matriz de rigidez no sistema, enquanto a função “manipulador” é o robô criado no *MatLab*, com suas respectivas características. Também são apresentados dois blocos, onde podem ser observados em tempo real, os valores que o programa atribui aos DoFs e, também, uma saída com todos os dados da simulação (simout1).

Uma primeira simulação foi realizada de forma a conferir as medidas obtidas pelo experimento prático. Inicialmente, foi simulada uma aplicação com entrada de θ_1 de 0,4rad no elo 01, segundo a Figura 53, que ilustra a resposta encontrada.

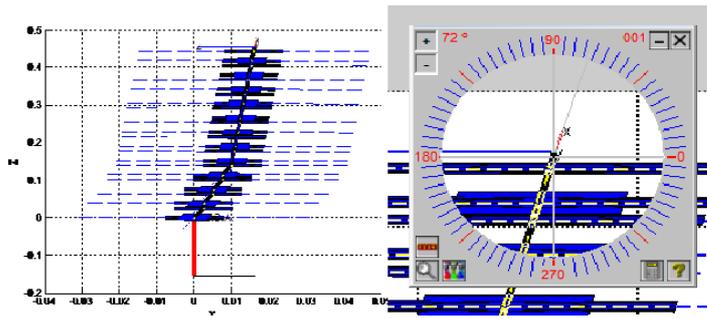


Figura 53 – Resposta Simulação 0,4rad

Pode ser observado que a angulação gerada pela simulação (18°) esta dentro da faixa encontrada experimentalmente (variando entre 21° e 23°) – Anexo VIII.

Uma segunda análise foi realizada ao aplicar $0,4\text{rad}$ em θ_2 e pode-se observar que a resposta é similar ao teste anterior, porém, no sentido do eixo $-x$, o que verifica que os DoFs de θ_1 e θ_2 estão desacoplados.

Com o intuito de apresentar uma funcionalidade prática, foi pensada uma tarefa em que a utilização de um manipulador tradicional fosse limitada, devido a sua estrutura. Imaginou-se uma caixa fechada, com alguns objetos a serem retirados, em seu interior, dentro de uma piscina de rejeitos radioativos, o que inviabiliza a entrada de seres humanos no local. Existem outros objetos sobre a caixa, impossibilitando uma retirada da caixa do ambiente radioativo. A caixa apresenta uma abertura em uma de suas extremidades e, no seu interior, existem algumas porcas que precisam ser retiradas, por algum motivo qualquer. A Figura 54 ilustra a caixa, objetos e porcas.

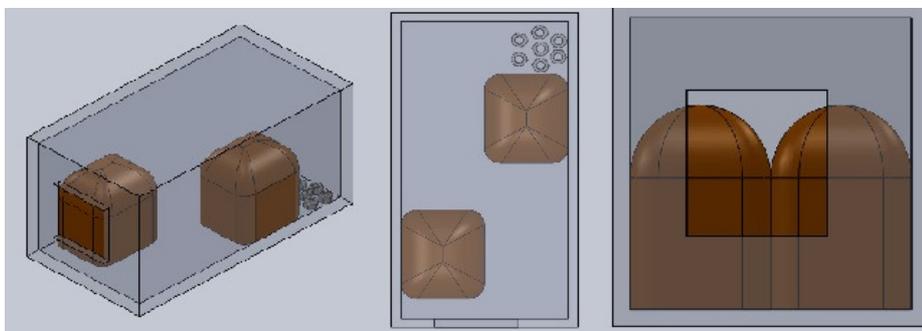


Figura 54 – Ilustração do Problema

Assumindo que as dimensões e posicionamentos dos objetos são idênticos aos da Figura 54, pode-se traçar uma trajetória para que um manipulador possa capturar as porcas. Um manipulador tradicional não consegue executar a tarefa devido ao pequeno espaço de trabalho, e, também, não consegue obter as angulações necessárias para realização das curvas para que se chegue até as porcas. Assim, um manipulador contínuo pode ser utilizado no lugar de um manipulador tradicional. Para a realização da tarefa, pode-se utilizar um ROV submarino com um manipulador contínuo remoto, atuado por tendões, acoplado em sua estrutura. Todo o sistema de controle e os atuadores do manipulador contínuo, podem estar afastados da piscina radioativa, gerando menos trabalho na fase de descontaminação e, também, menor número de possíveis rejeitos.

Como o objetivo é exemplificar o trabalho do manipulador desenvolvido em uma tarefa prática, serão simulados os passos do manipulador até que se chegue aos objetos desejados. Inicialmente, somente o elo 3 será atuado positivamente de forma a evitar a colisão com o primeiro objeto. Em seguida, o elo 2 será atuado com a mesma angulação que o elo 3 utilizou na primeira etapa, enquanto que o elo 3 passa a receber uma atuação negativa, com o intuito de evitar colidir tanto com a parede da caixa quanto com o novo objeto a sua frente. A última etapa é a atuação do elo 1, que recebe a mesma angulação utilizada pelo elo 3 na etapa 01, o elo 2 recebe a angulação que o elo 3 estava utilizando na etapa 02 e, finalmente, o elo 3 recebe uma angulação positiva, chegando até as porcas. Nesta terceira etapa, o elo 1, também, recebe uma pequena inclinação negativa, perpendicular às angulações anteriores, para que o manipulador possa chegar na altura das porcas. Para realizar a retirada das porcas do local, basta realizar o procedimento inverso. Foi utilizado, para calcular os ângulos necessários para executar cada etapa, o programa Scale2.0. As Figuras 55 a 57 ilustram os ângulos necessários.

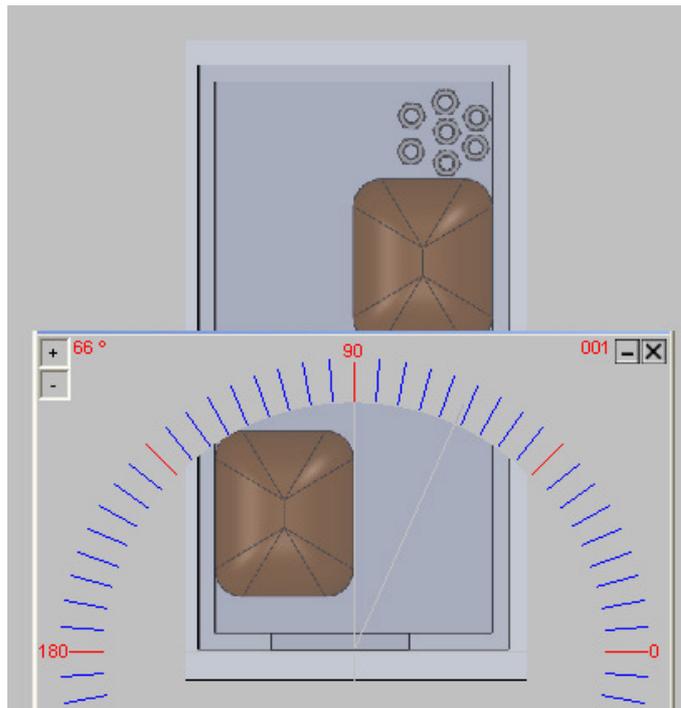


Figura 55 – Ângulo da Etapa 01

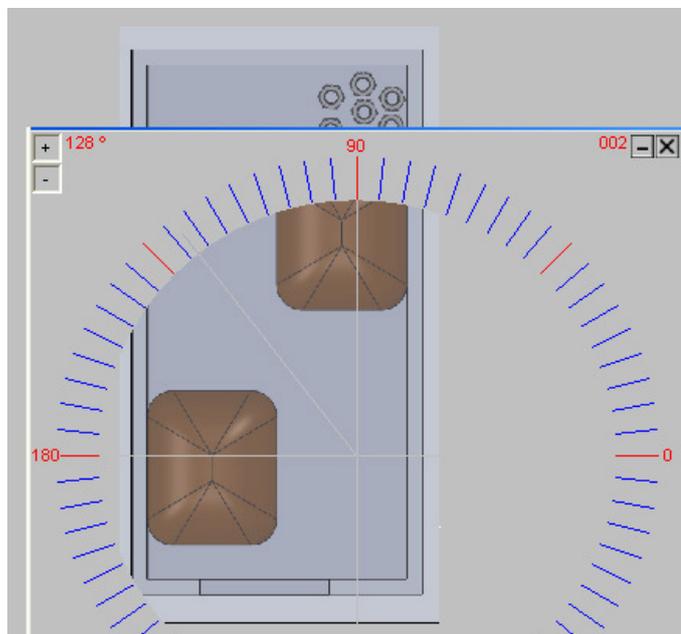


Figura 56 – Ângulo da Etapa 02

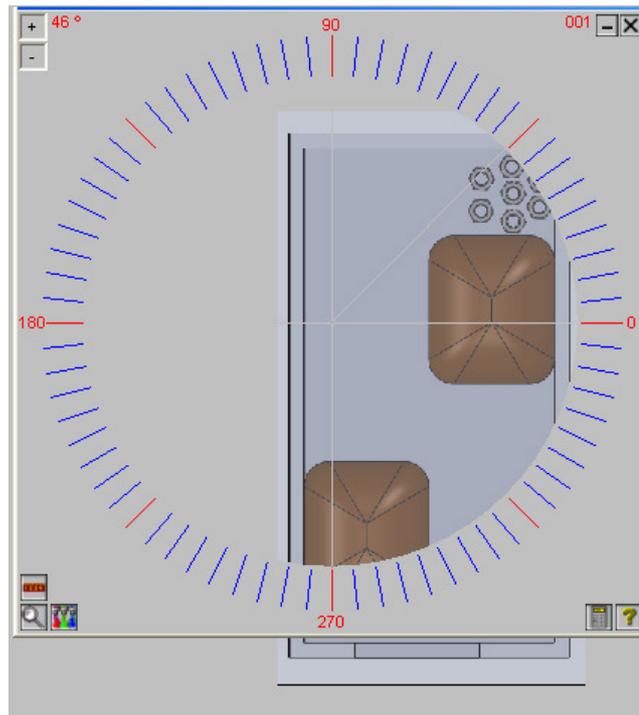


Figura 57 – Ângulo da Etapa 03

Logo, foram realizadas as conversões necessárias de graus para radianos e, de acordo os dados da Tabela 9, que apresenta as angulações das entradas para cada DoF, cada etapa foi simulada no programa desenvolvido no *MatLab*.

Tabela 5-5 – Entradas para Realização das Etapas

	Etapa 01			Etapa 02			Etapa 03		
	Elo03	Elo02	Elo 01	Elo03	Elo02	Elo 01	Elo03	Elo02	Elo 01
θ_1	0,95993	-	-	-0,66323	0,97738	-	-0,78540	-0,66323	0,97738
θ_2	-	-	-	-	-	-	-	-	-0,08726
d_3	-	-	-	-	-	-	-	-	-

As Figuras 58, 59 e 60 apresentam os resultados obtidos pelas simulações.

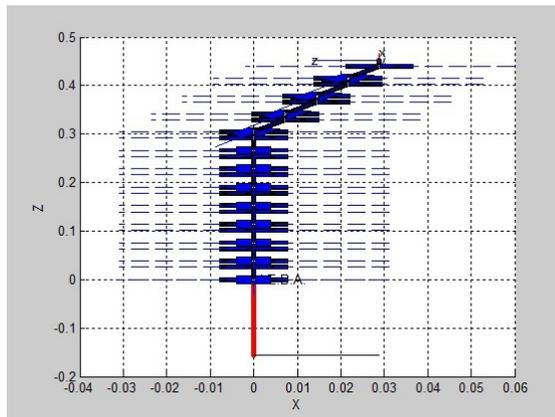


Figura 58 – Etapa 01

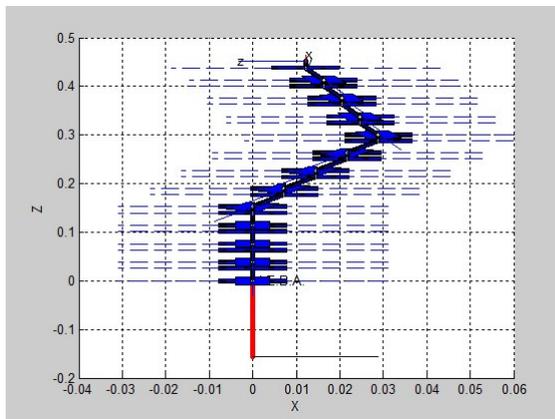


Figura 59 – Etapa 02

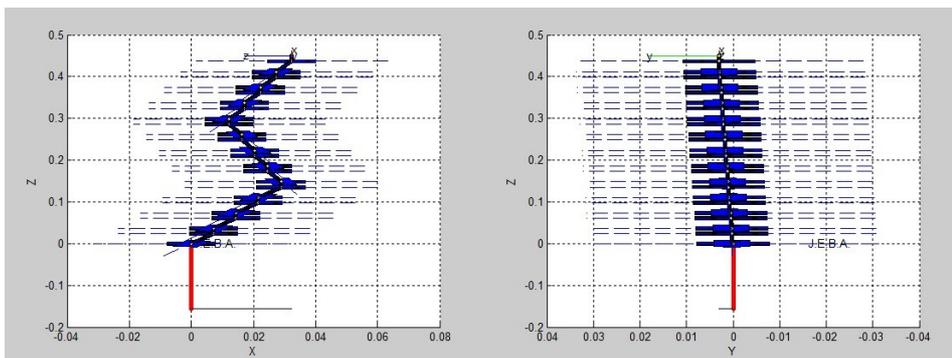


Figura 60 – Etapa 03

A Figura 61 ilustra as angulações geradas em cada etapa da simulação. A primeira imagem (esquerda superior) é relativa à etapa I, enquanto que, ao seu lado, a etapa II e as duas imagens na parte inferior, ilustram os ângulos apresentados na Figura 60.

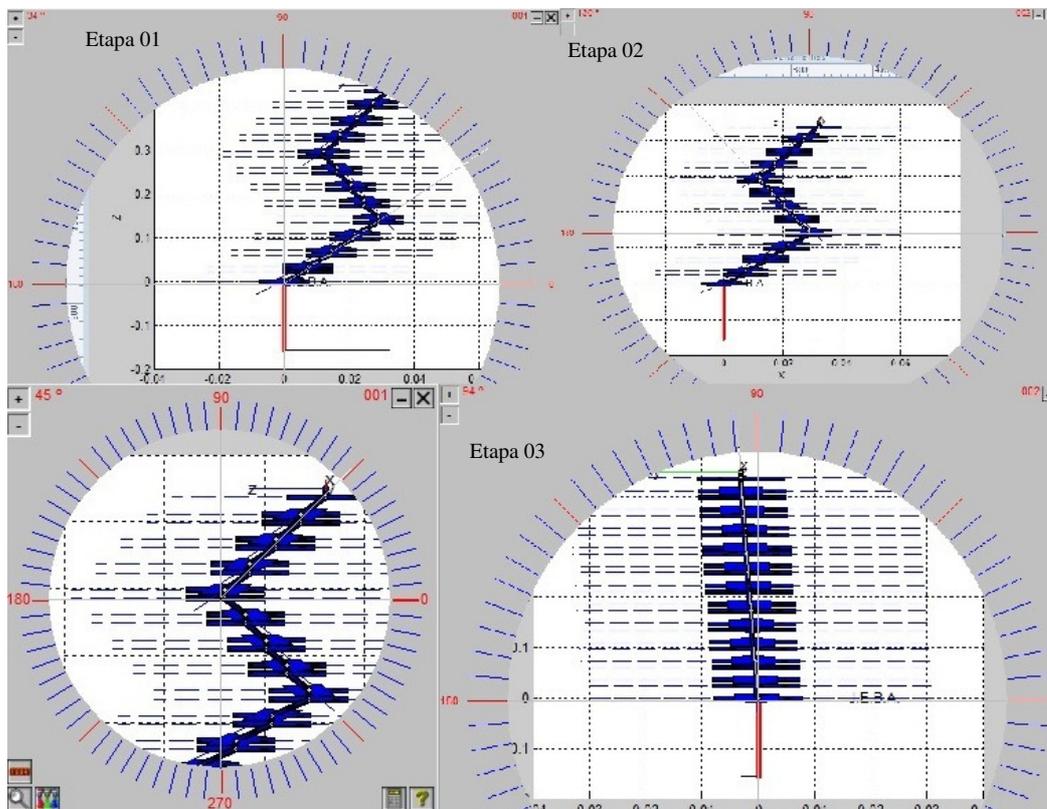


Figura 61 – Ângulos Medidos

Os ângulos medidos na simulação foram os apresentados na Tabela 10.

Tabela 5-6 – Ângulos da Simulação

	Etapa 01			Etapa 02			Etapa 03		
	Elo03	Elo02	Elo01	Elo03	Elo02	Elo01	Elo03	Elo02	Elo01
θ_1	0,9774	-	-	-0,6981	0,9774	-	0,7854	-0,6981	0,9774
θ_2	-	-	-	-	-	-	-	-	-0,0698
d3	-	-	-	-	-	-	-	-	-

Pode-se observar que com as angulações escolhidas anteriormente (Tab. 9), estão bem próximas das angulações encontradas na simulação (Tab. 10), logo a tarefa, teoricamente, pode ser executada. Vale salientar que este procedimento é uma simulação e que no procedimento experimental, o resultado pode ser diferente devido as forças externas que não foram analisadas nesta simulação.