



UNIVERSIDADE FEDERAL DE OURO PRETO - UFOP
DEPARTAMENTO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO -
DECAT
ELEMENTOS DE ROBÓTICA (CAT 181)

TRABALHO DE CONTROLE CINEMÁTICO

RAY DA SILVA BASÍLIO - 17.2.5972

- 1) Considerando o trabalho de controle cinemático de posição, de um robô planar com duas juntas de revolução (RR) – Figura 1, desenvolvido pelos discentes da disciplina CAT 181 no 1º semestre de 2013, implemente o controle cinemático de posição para a execução de uma trajetória retilínea a partir do script “Exemplo2gdlplanar.m” de acordo com os parâmetros da Tabela 1. Sendo: (x_0, y_0) a posição inicial do robô; e (x_i, y_i) o ponto inicial e (x_f, y_f) o ponto final da trajetória retilínea. Analise o comportamento do sinal de controle de cada junta do robô, e dos sinais de erro em X e Y, ao longo da trajetória linear para dois casos distintos. Cada linha associada a um número de matrícula representa uma trajetória a ser executada.

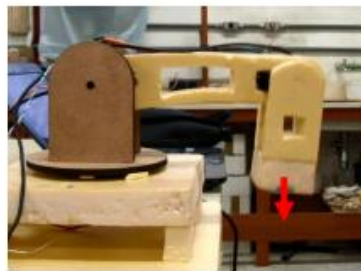


Figura 1 – Robô planar com duas juntas de revolução na posição de repouso. Note que na posição de repouso o antebraço é perpendicular ao solo ($\alpha_2 = -\pi/2$).

Tabela 1 – Parâmetros para a Execução de duas Trajetórias por Aluno(a).

Matrícula	Posição Inicial do Robô		Trajetória Retilínea			
	x_0	y_0	x_i	y_i	x_f	y_f
17.2.5972	-19.75	12.2	-19.75	12.2	18.37	19.07
	20.36	17.3	20.36	17.3	-6.165	12.47

Após a análise dos resultados para as duas situações apresentadas, implemente no script o limite de $80^\circ/s$ para a velocidade angular de cada junta. Repita as duas trajetórias e analise o comportamento do sinal de controle de cada junta do robô em função dos sinais de erro em X e Y. No relatório apresenta a trajetória desejada e executada, bem como o espaço de trabalho. Realize as alterações no script que forem necessárias para implementar as condições supracitadas, bem como para plotar os gráficos da trajetória, posições angulares, velocidades angulares e sinais de erro.

Para implementar o controle cinemático de posição e a executar uma trajetória retilínea a partir do script “Exemplo2gdlplanar.m” de acordo com os parâmetros da Tabela 1. Sendo: (x_0, y_0) a posição inicial do robô; e (x_i, y_i) o ponto inicial e (x_f, y_f) o ponto final da trajetória retilínea. Primeiro foi necessário fazer a cinemática inversa dos pontos (x_0, y_0) . Assim para os pontos propostos os ângulos encontrados levando as considerações dos limites mecânicos foram:

- Limites Mecânicos do Servo do Levantamento do Braço:
 - Mínimo: 0° .
 - Máximo: $+180^\circ$.
- Limites Mecânicos da Servo do Cotovelo do Braço:
 - Mínimo: -50° .
 - Máximo: $+90^\circ$.
- A partir do *script* do matlab temos:
 - $p1(-19.75, 12.2)$:
 - $\theta_1 = 173.6823$ covelo abaixo
 - $\theta_2 = 5.5812$ covelo abaixo
 - $p2(20, 36, 17, 3)$:
 - $\theta_1 = 58.7955$ covelo abaixo
 - $\theta_2 = 32.3133$ covelo abaixo

1.1 SEM A LIMITAÇÃO DAS VELOCIDADES DAS JUNTAS

- PRIMEIRO CASO:

Tabela 1 - Parâmetros para a execução da trajetória 1

Posição inicial do Robô		Trajetória Retilínea			
xo	yo	xi	yi	xf	yf
-19.75	12.2	-19.75	12.2	18.37	19.07

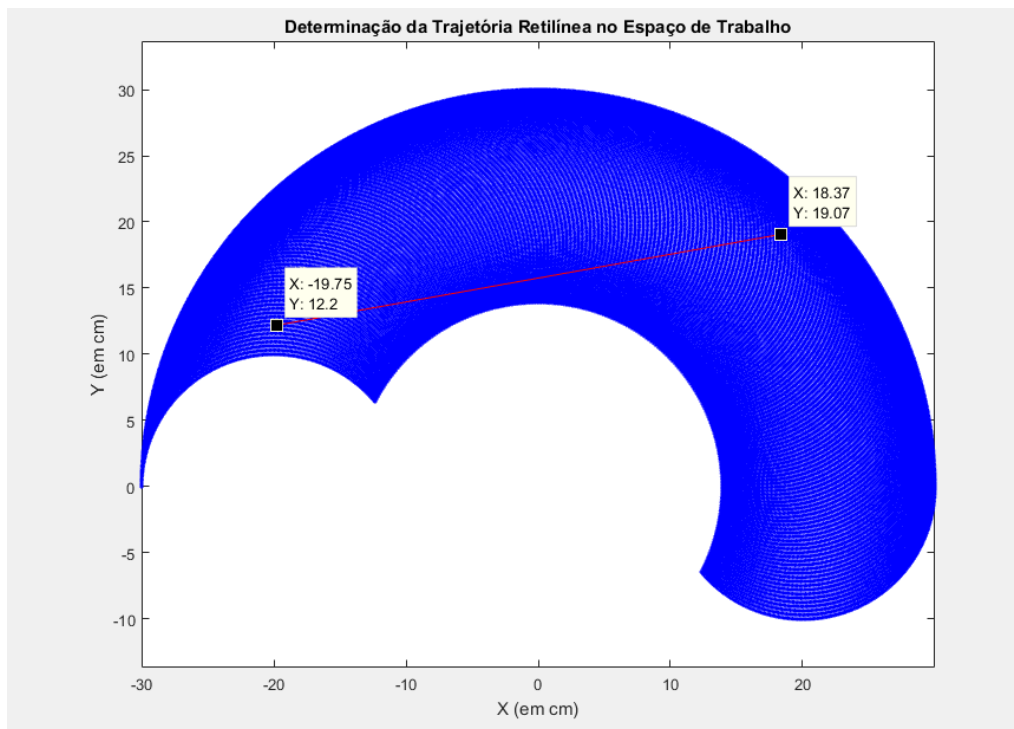


Figura 1 – Determinação da trajetória retilínea no espaço de trabalho 1

Na Figura 1 é possível verificar o espaço de trabalho do robô em azul. Este espaço de trabalho, corresponde ao volume total que pode ser percorrido pelo efetuador (órgão terminal ou ferramenta) dadas todas as possibilidades de execução de movimento do robô, sendo limitado pela geometria do manipulador, bem como pelas restrições físicas das juntas (limites mecânicos). Também é verificado nesta imagem, as características da trajetória determinada, e por meio dela pode-se observar que o movimento é iniciado dentro do espaço de trabalho no ponto $p1_{xyi}(-19.75, 12.2)$ e finaliza no ponto $p1_{xyf}(18.37, 19.07)$, que também está dentro do espaço de trabalho.

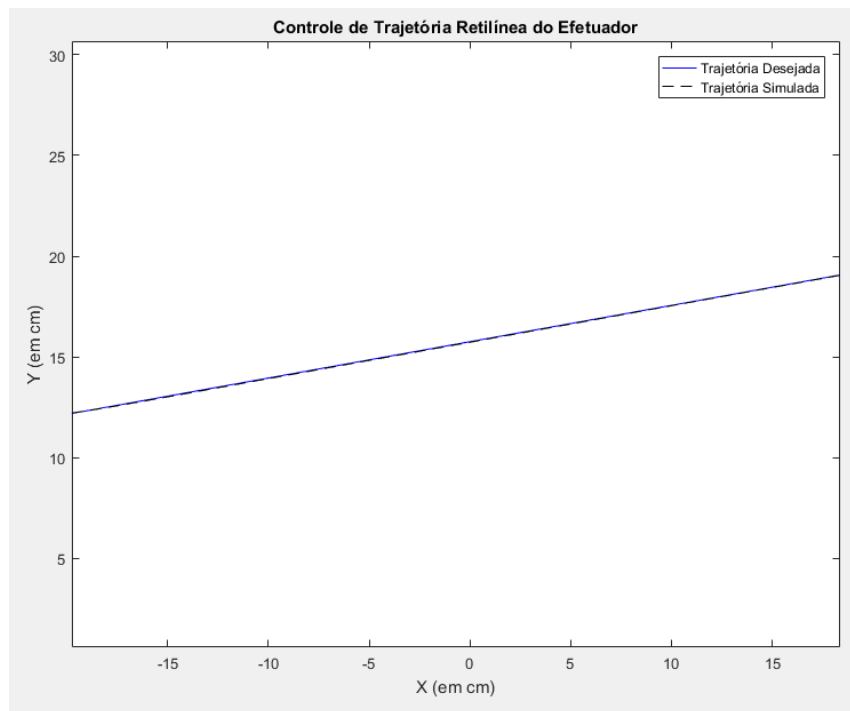


Figura 2 – Gráfico de controle da trajetória 1.

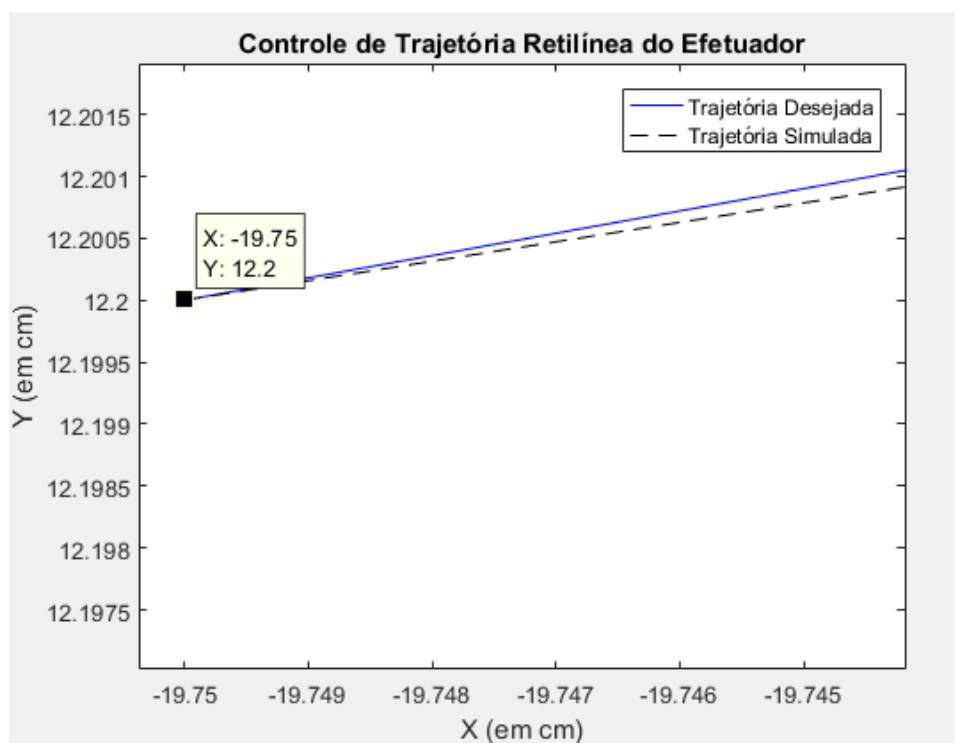


Figura 2.1 – Gráfico de controle da trajetória 1 com zoom no ponto inicial da trajetória.

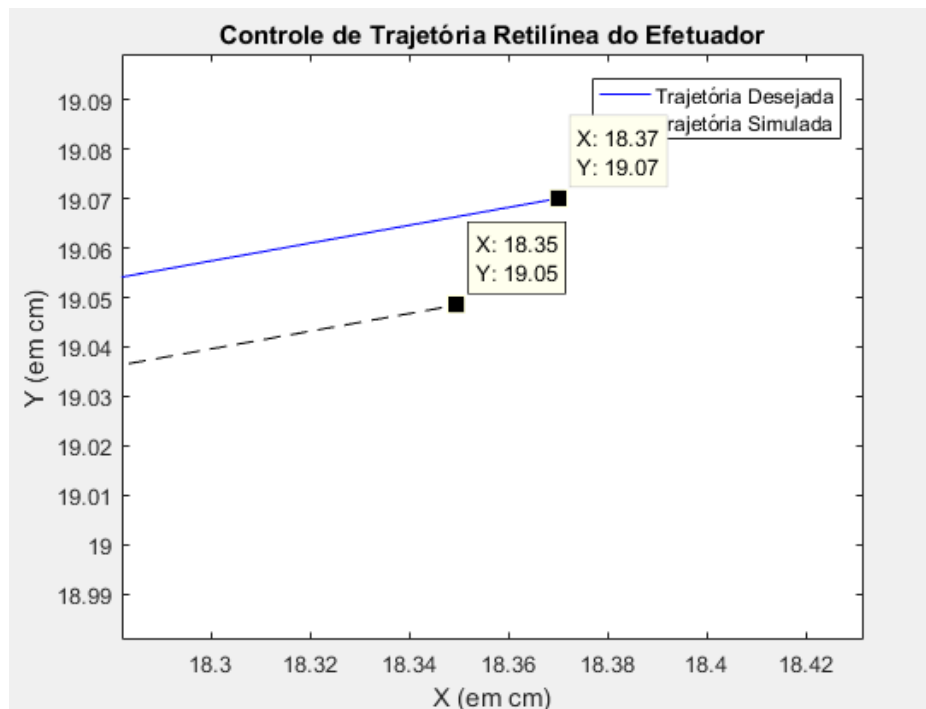


Figura 2.2 – Gráfico de controle da trajetória 1 com zoom no ponto final da trajetória.

Observando a Figura 2, Figura 2.1 e Figura 2.2, é demonstrado como foi feito o controle da trajetória, sendo que a trajetória linear simulada iniciou no ponto $p1s_xyi$ (-19.73; 12.23) e finalizou no ponto $p1s_xyi$ (18.35,19,05). Era desejado que a trajetória linear tivesse início em $p1d_xyi$ (-19.75,12.2) e terminasse no ponto $p1d_xyf$ (18.37,19.07) mas por possíveis arredondamentos feitos durante o código temos esse pequeno erro + ou - 0,02. Apesar disto, o efetuador conseguiu realizar a trajetória sem nenhum distúrbio durante o percurso da trajetória linear.

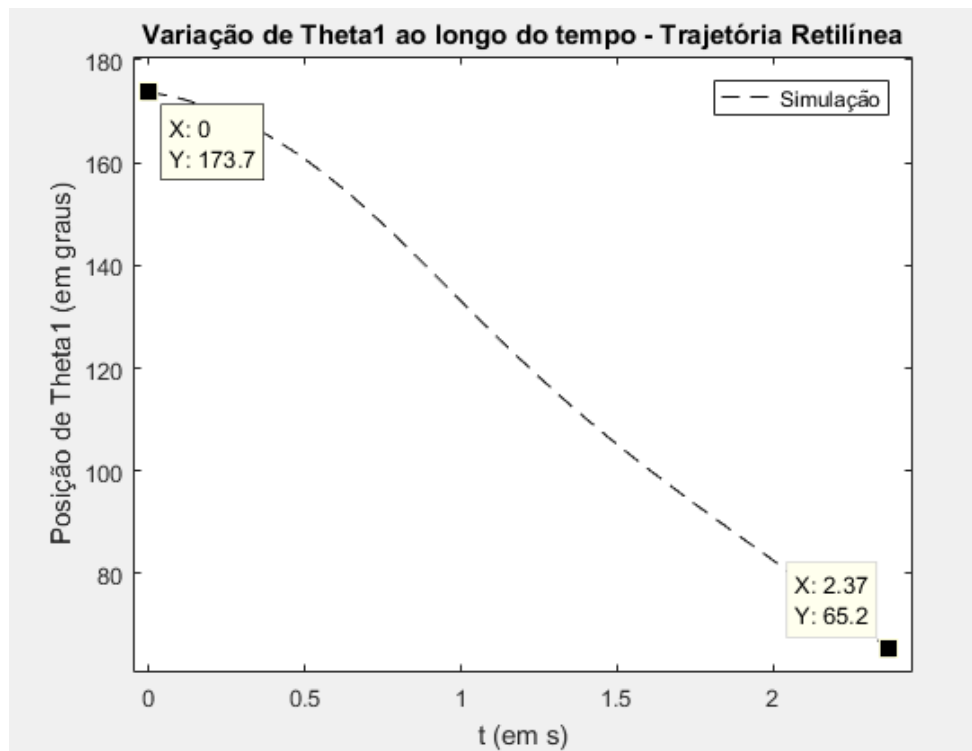


Figura 3 – Variação do ângulo theta1 ao longo do tempo.

Na figura 3, é verificado a variação do theta1 durante a trajetória especificada, tendo um decréscimo de 108,5 unidades em relação ao theta inicial, gastando 2.37s para atingir 65,2°.

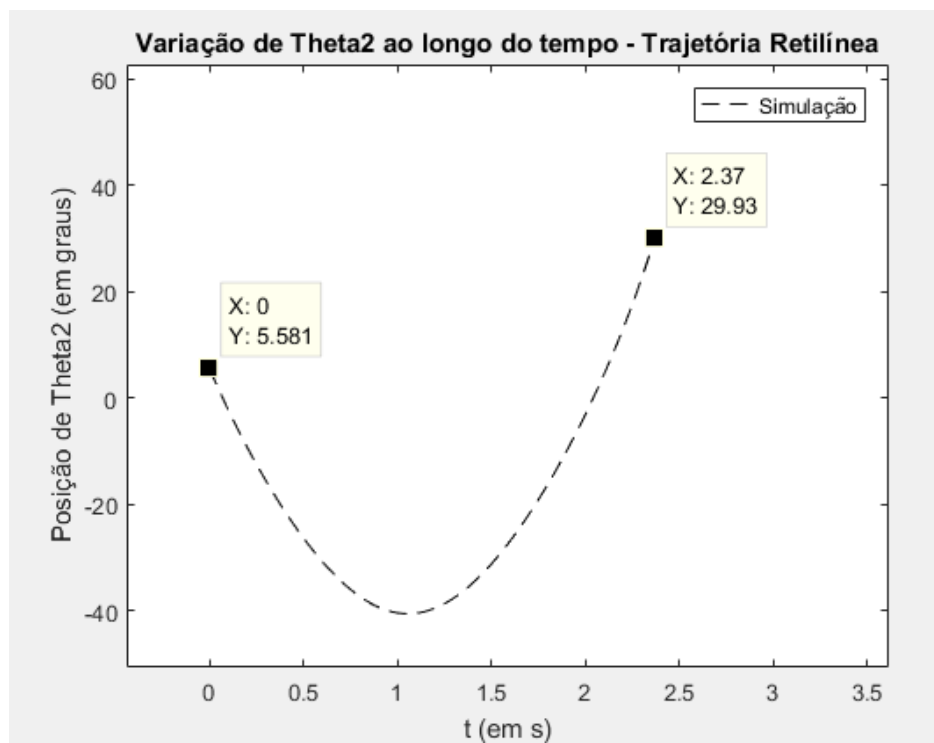


Figura 4 – Variação do ângulo Theta 2 ao longo do tempo.

Na figura 4, é verificado a variação do Theta 2 durante a trajetória especificada, tendo um valor inicial de 5,5881°, e após 2.37s atinge 29,93°.

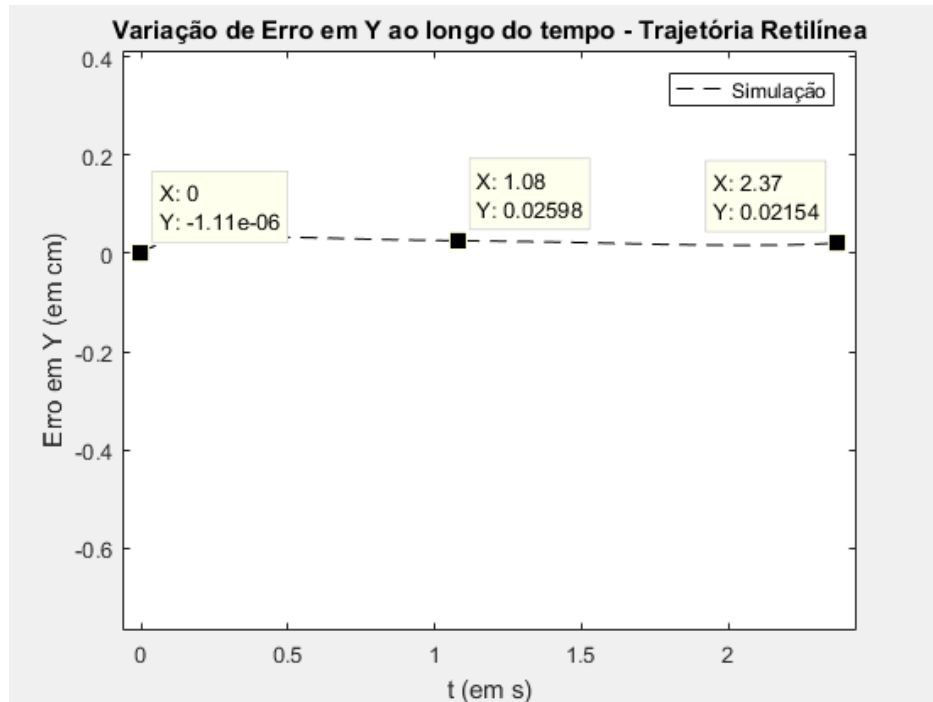


Figura 5 – Variação do erro no eixo Y.

A Figura 5 demonstra o que já se era esperado, como o ponto inicial do robô é coincidente com início da trajetória, o erro inicial em Y é praticamente nulo, tendo pequenas variações devido à aproximação dos thetas utilizando 4 casas decimais. Outro ponto a ser levantado é que como não teve-se distúrbios ou singularidades durante a trajetória simulada no eixo Y, este encontra praticamente a trajetória desejada no eixo Y e o erro se aproxima de zero.

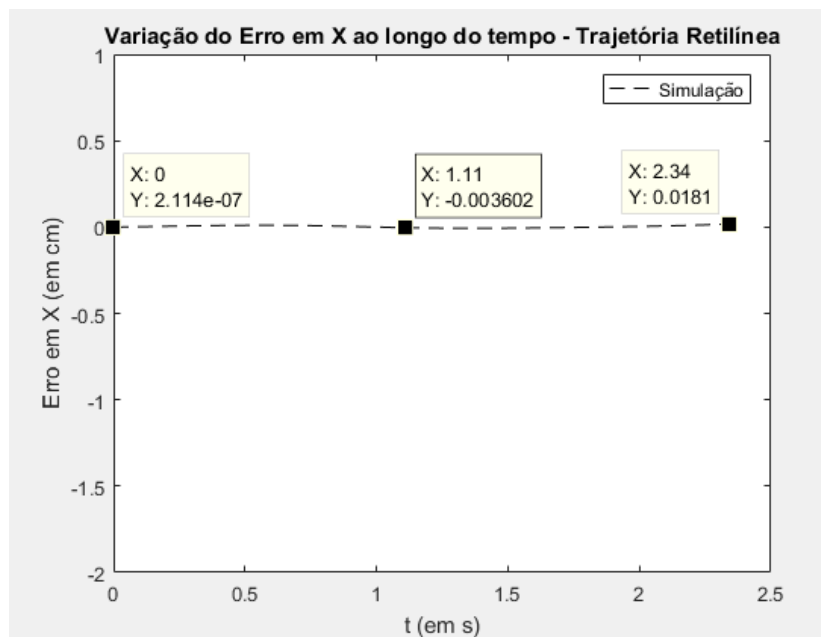


Figura 6 – Variação do erro no eixo X.

A Figura 6 demonstra o mesmo que fora observado na figura anterior , como o ponto inicial do robô é coincidente com início da trajetória, o erro inicial em X é praticamente nulo, tendo pequenas variações devido à aproximação dos thetas utilizando 4 casas decimais. Outro ponto a ser levantado é que como não teve-se distúrbios ou singularidades durante a trajetória simulada no eixo X, este encontra praticamente a trajetória desejada no eixo X e o erro se aproxima de zero .

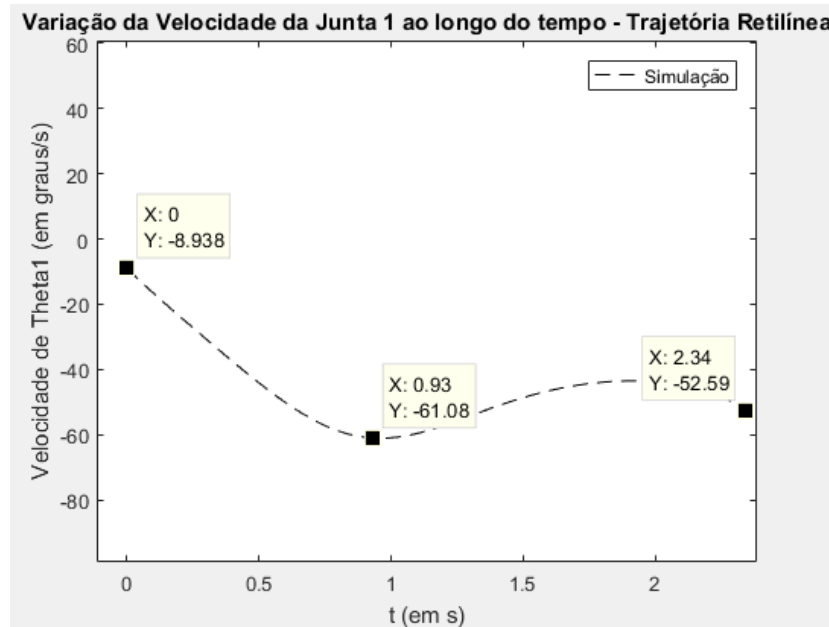


Figura 7 – Variação da velocidade da junta 1 ao longo do tempo .

A Figura 7 demonstra as variações da velocidade da junta 1 à medida que a trajetória simulada é executada. Ao partimos do ponto inicial da trajetória simulada (x_0 , y_0), temos uma alta velocidade aplicada (resultante da aceleração aproximada de $|-61.08 + 8.938|/0.93 = 56.0667^\circ/s^2$), à medida que se aproxima do ponto desejado, a velocidade se estabiliza na faixa de (40 a $52,9^\circ/s$ em módulo).

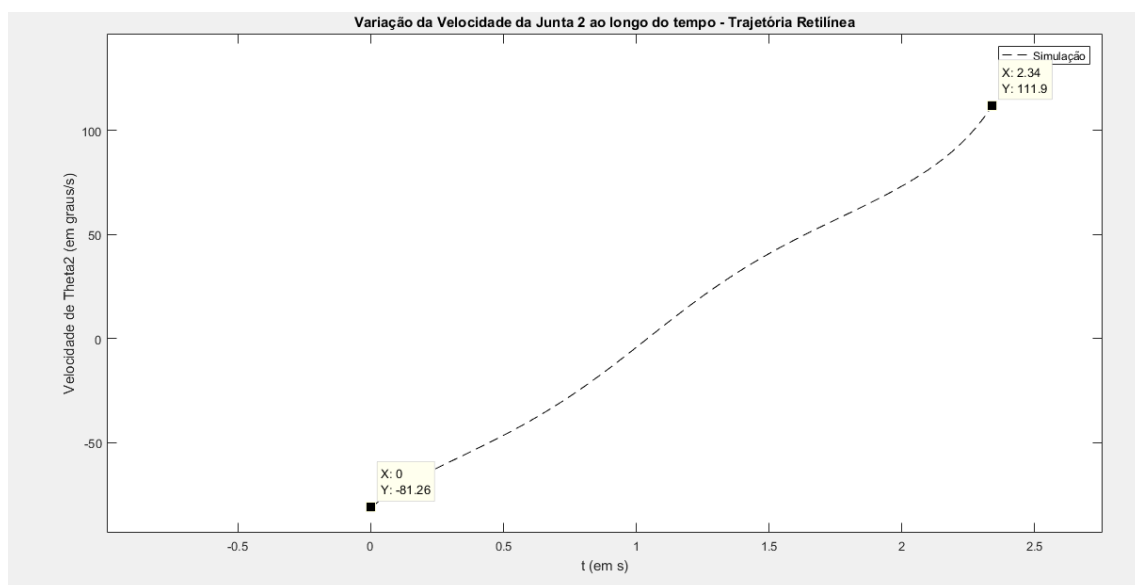


Figura 8 – Velocidade da junta 2 1 ao longo do tempo.

A Figura 8 demonstra o comportamento da velocidade da junta 2 à medida que a trajetória simulada é executada. Ao partimos do ponto inicial da trajetória simulada (x_0 , y_0), temos uma alta velocidade aplicada (resultante da aceleração aproximada de $|111.9-81.6|/2.34 = 12.948^\circ/s^2$) atingindo uma velocidade final de $111.9^\circ/s$.

- **SEGUNDO CASO :**

Tabela 2 - Parâmetros para a execução da trajetória 2

Posição inicial do Robô		Trajetória Retilínea			
x_0	y_0	x_i	y_i	x_f	y_f
20.36	17.3	20.36	17.3	-6.165	12.47

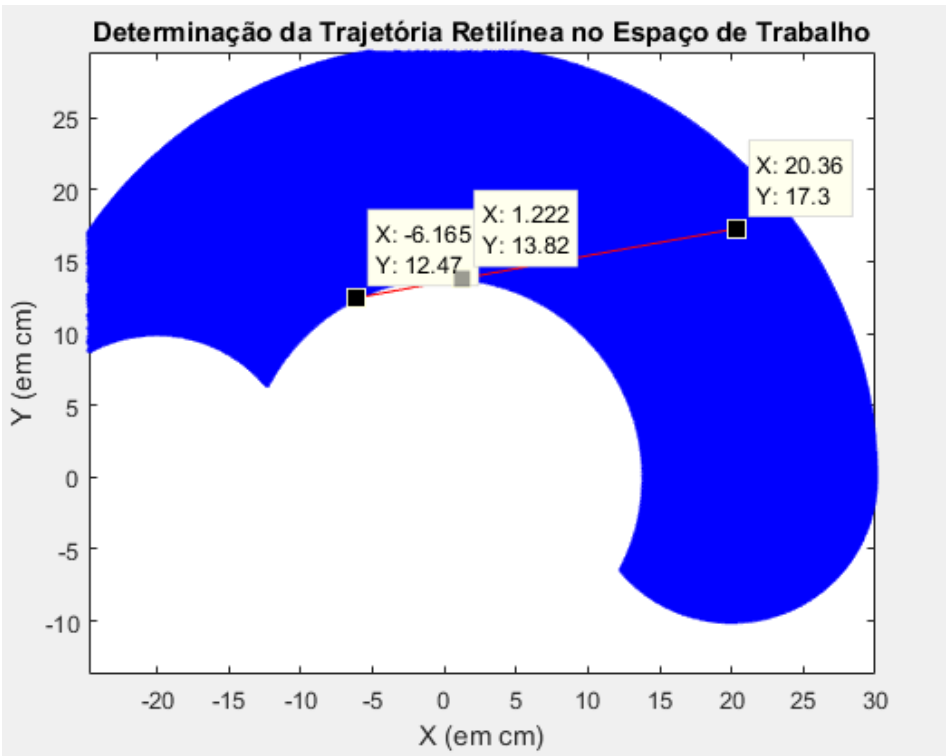


Figura 9 – Determinação da trajetória retilínea no espaço de trabalho 2

Na Figura 9 é possível verificar o espaço de trabalho do robô em azul. Este espaço de trabalho, corresponde ao volume total que pode ser percorrido pelo efetuador (órgão terminal ou ferramenta) dadas todas as possibilidades de execução de movimento do robô, sendo limitado pela geometria do manipulador, bem como pelas restrições físicas das juntas (limites mecânicos). Nesta imagem, as características da trajetória determinada, apresentam os pontos iniciais do dentro do espaço de trabalho no ponto $p1_xyi(20.36,17.3)$ e finaliza no ponto $p1_xyf(-6.165,12.47)$, que também está dentro do espaço de trabalho mas o trecho $pxy(1.22,13.82)$ até $pxy(-6,47,12.47)$ é verificado uma singularidade que corresponde às configurações do mecanismo nas quais a matriz Jacobiana do sistema tem posto deficiente. Em especial, a singularidade apresentada é uma singularidade de fronteira e ocorre quando o efetuador encontra-se na fronteira do espaço de trabalho.

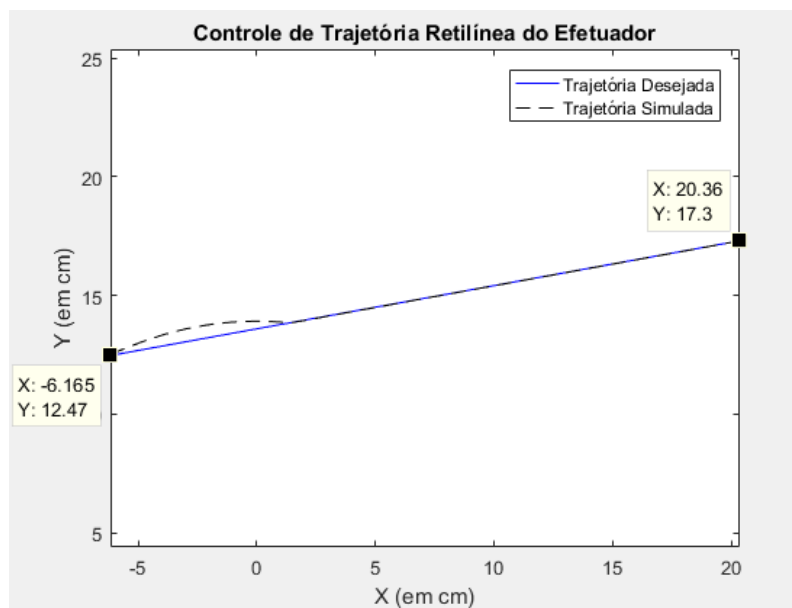


Figura 10 – Gráfico de controle da trajetória 2.

Observa-se na Figura 10 como foi feito o controle da trajetória. Era desejado que a trajetória linear iria iniciar em $p1d_xyi(20.36,17.3)$ e finalizar no ponto $p1d_xyf(-6.165,12.47)$ mas por existir uma singularidade no trecho, o controlador faz com que o sistema durante a simulação tente atingir o valor desejado do trecho linear, não conseguindo ele vai forçando as juntas até que em um determinado período de tempo ele consegue contornar a singularidade e ajustar novamente a trajetória.

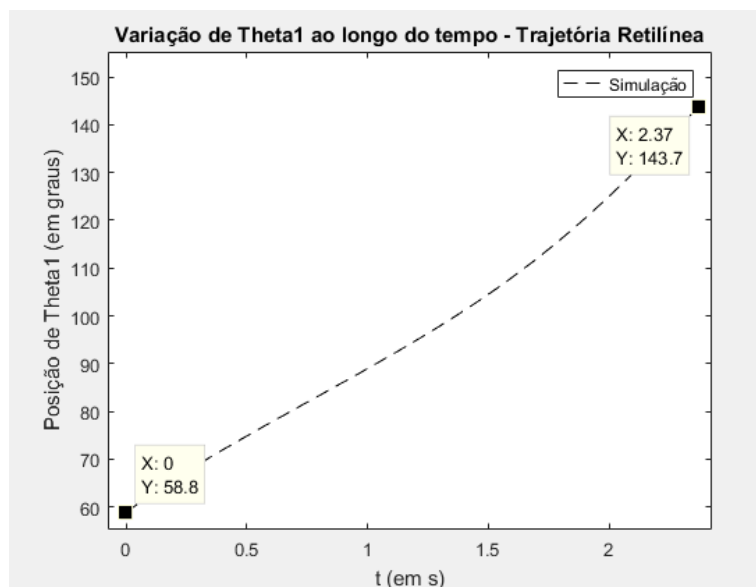


Figura 11 – Variação do ângulo theta1 ao longo do tempo.

Na figura 11, é verificado a variação do theta1 durante a trajetória especificada, tendo um acréscimo de 84.9 unidades em relação ao theta inicial, gastando 2.37s para atingir 143.7°.

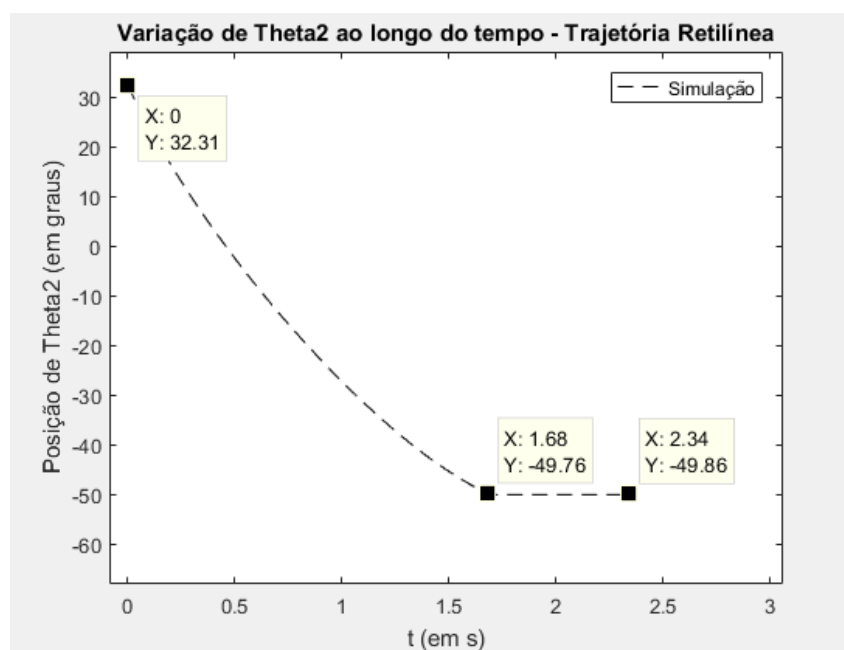


Figura 12 – Variação do ângulo theta 2 ao longo do tempo.

Na figura 12, é verificado a variação do Theta 2 durante a trajetória especificada, tendo um valor inicial de 32.31° , e após 2.37s atinge -49.86° .

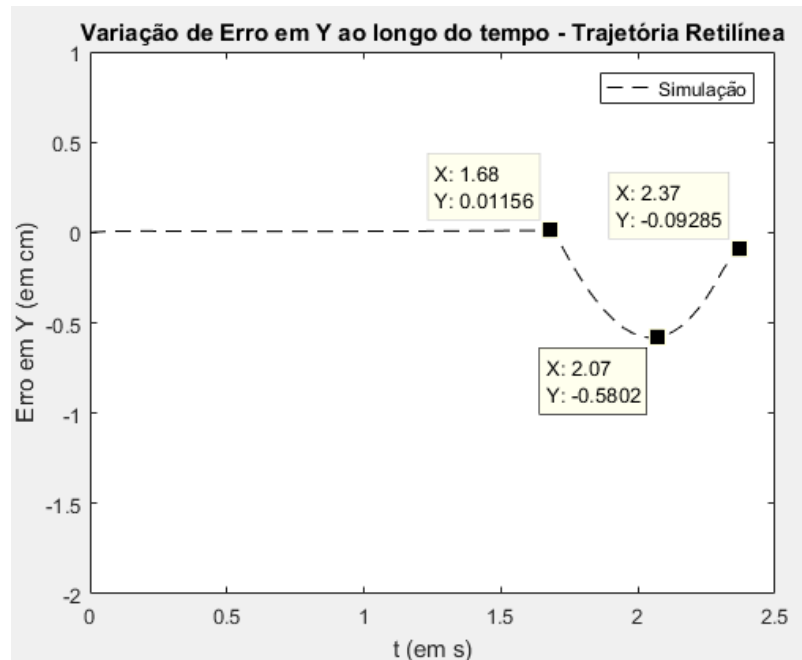


Figura 13 – Variação do erro no eixo Y.

A Figura 13 demonstra o que já era esperado, como o ponto inicial do robô é coincidente com início da trajetória, o erro inicial em Y é praticamente nulo, tendo pequenas variações durante a trajetória. Quando o sistema encontra a singularidade, em 1.68s, os valores na trajetória simulada e desejada começam a distanciar entre si assim, aumenta o erro até que a trajetória simulada retorne ao espaço de trabalho.

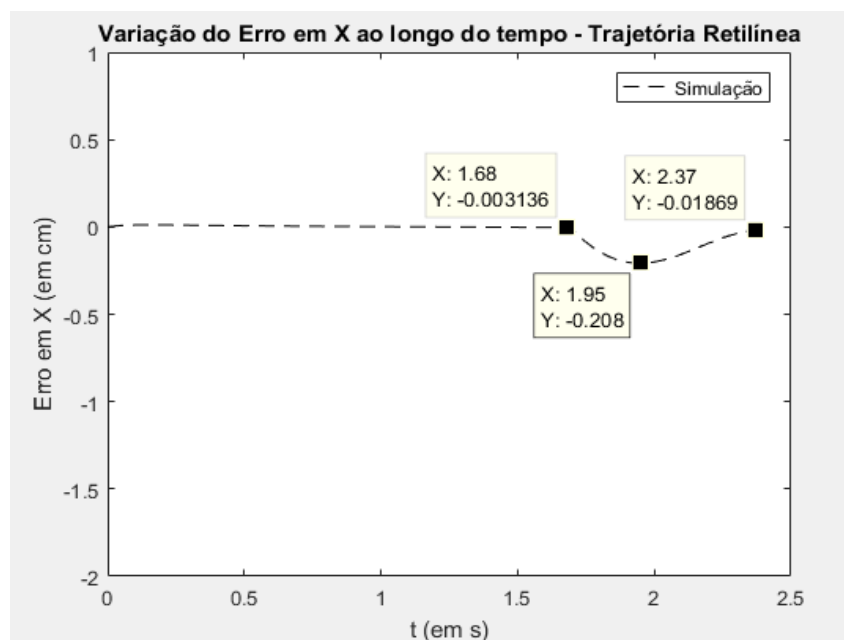


Figura 14– Variação do erro no eixo X.

A Figura 14 demonstra o que já era esperado, como o ponto inicial do robô é coincidente com início da trajetória, o erro inicial em X é praticamente nulo, tendo pequenas variações durante a trajetória. Quando o sistema encontra a singularidade, em 1.68s as trajetórias simulada e desejada começam a distanciar entre si assim, aumenta o erro até que a trajetória simulada retorne ao espaço de trabalho.

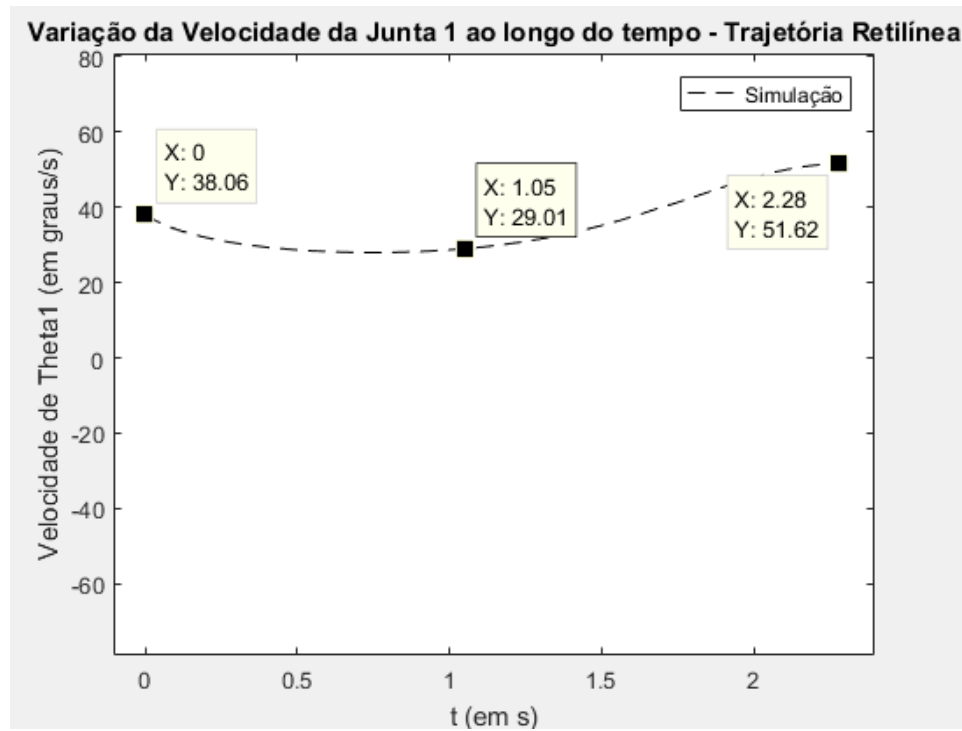


Figura 15– Variação da velocidade da junta 1 ao longo do tempo .

A Figura 15 demonstra as variações da velocidade da junta 1 à medida que a trajetória simulada é executada. Ao partimos do ponto inicial da trajetória simulada (x_0 , y_0), temos baixa variância nos valores apresentados permanecendo em uma faixa estável de 29° /s até 51.62°/s.

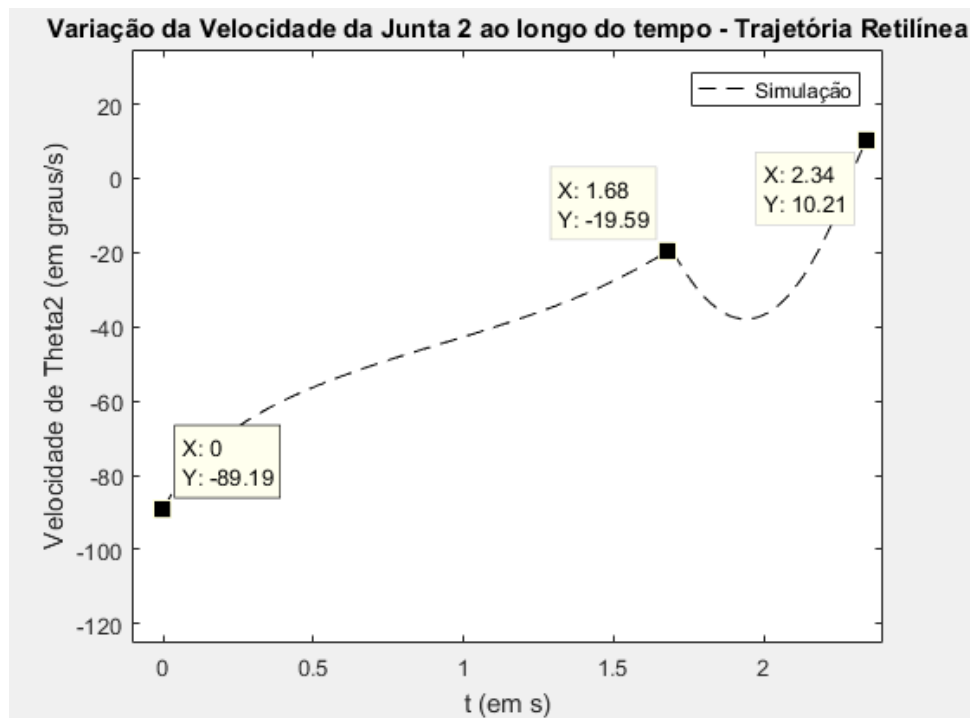


Figura 16 –Variação da velocidade da junta 2 ao longo do tempo.

A Figura 16 demonstra o comportamento da velocidade da junta 2 à medida que a trajetória simulada é executada. Ao partirmos do ponto inicial da trajetória simulada (x_0 , y_0), temos uma alta velocidade aplicada e a medida que chegamos perto singularidade no tempo de 1.68s, temos uma grande variação de modo súbito em seu valor, voltando a se estabilizar após retornar ao espaço de trabalho em 2.34s.

1.2 LIMITANDO A VELOCIDADE ANGULAR DE CADA JUNTA

Em determinadas situações é necessário realizar a limitação das velocidades angulares das juntas, uma vez que, se caso não limitados podem acarretar em esforços excessivos nos atuadores, assim como vibrações ou movimentos que possam comprometer a estrutura mecânica do robô .

- **PRIMEIRO CASO:**

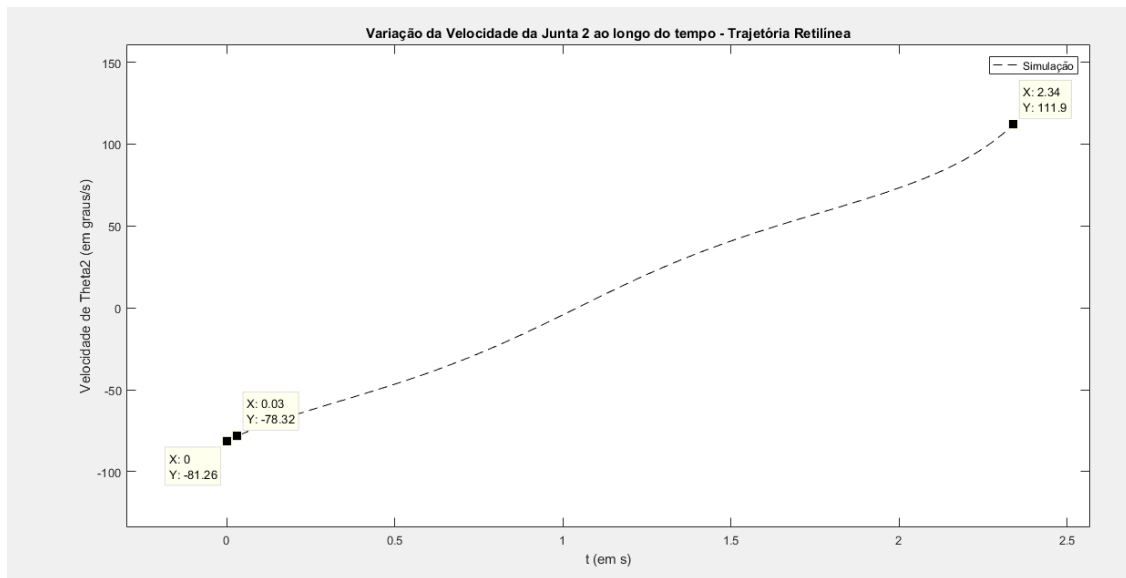


Figura 17 –Variação da velocidade da junta 2 ao longo do tempo sem limitação.

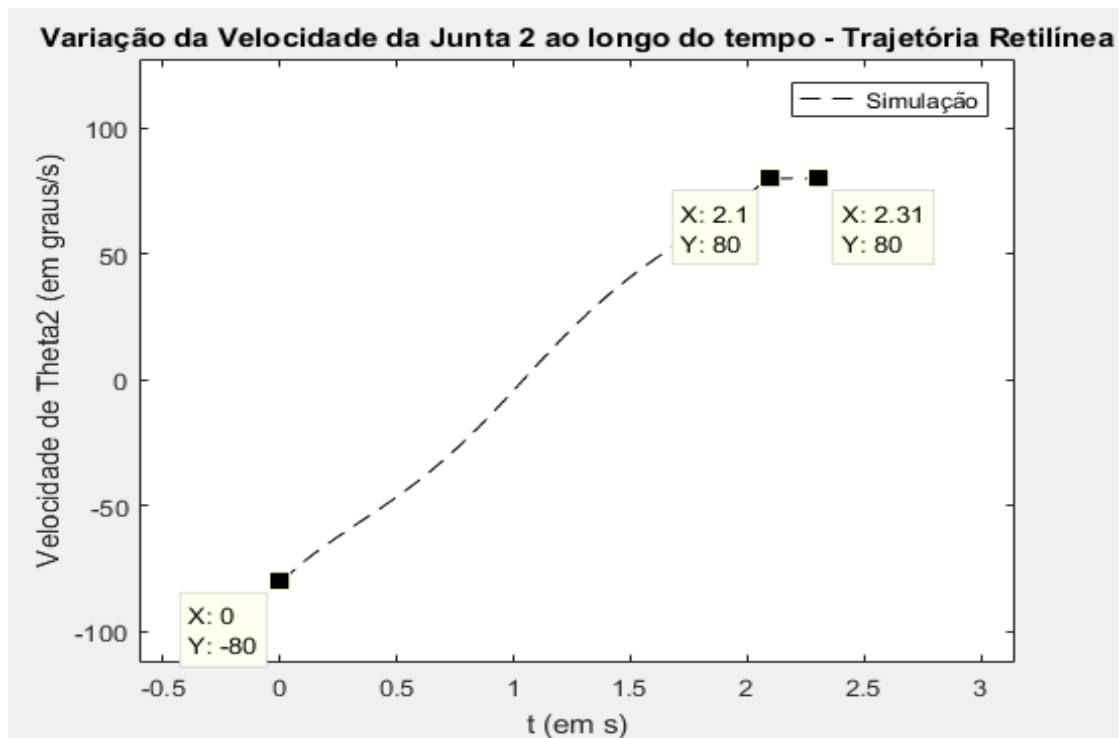


Figura 18 – Variação da velocidade da junta 2 ao longo do tempo com limitação.

As imagens 17 e 18 demonstram as limitações requeridas pelo exercício a $80^\circ/\text{s}$ para a velocidade angular da junta 2. É observado que para a faixa de -80 a $80^\circ/\text{s}$ os valores não foram modificados, já fora deste intervalo ocorre uma saturação em $80^\circ/\text{s}$. As regiões que foram modificadas são compreendidas no intervalo $0 \leq t(s) \leq 0.03$ e $2.1 \leq t(s) \leq 2.31$.

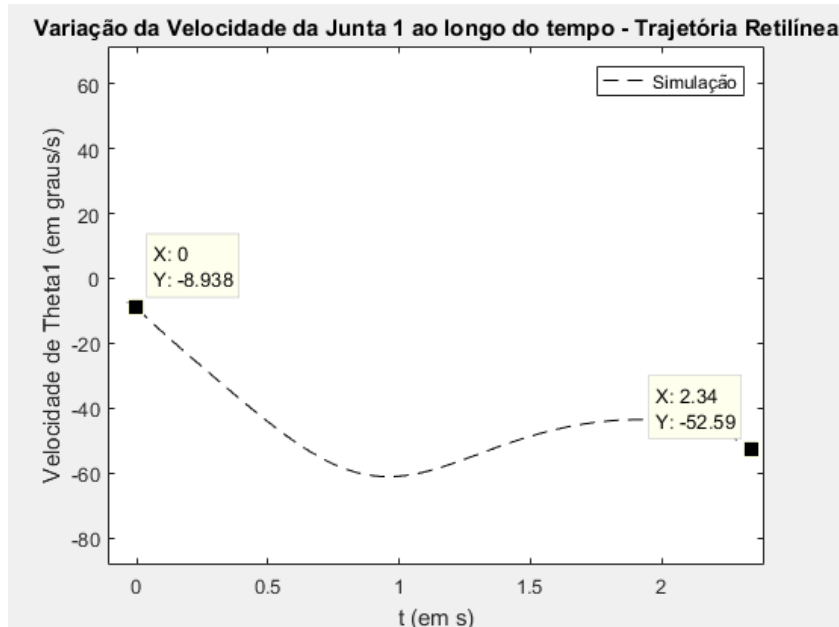


Figura 19 – Variação da velocidade da junta 1 ao longo do tempo sem limitação.

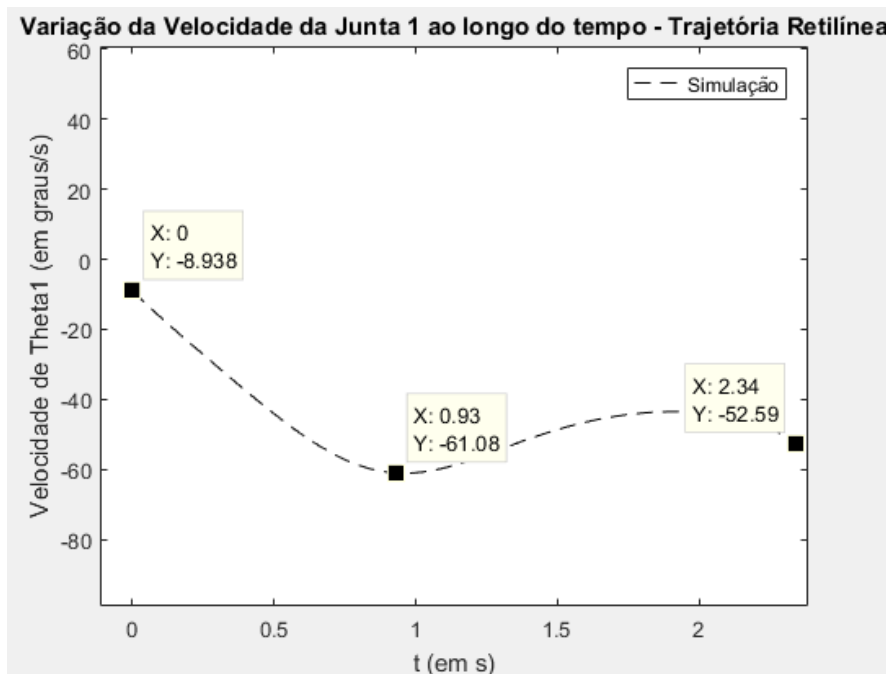


Figura 20 – Variação da velocidade da junta 1 ao longo do tempo com limitação.

As imagens 19 e 20 demonstram as velocidades das juntas antes e depois da limitação para $80^\circ/\text{s}$ na junta 1. É observado que os valores assumidos pela velocidades da junta 1

se encontram na faixa de - 80 a 80 °/s então os valores não sofreram alterações devido a esta restrição.

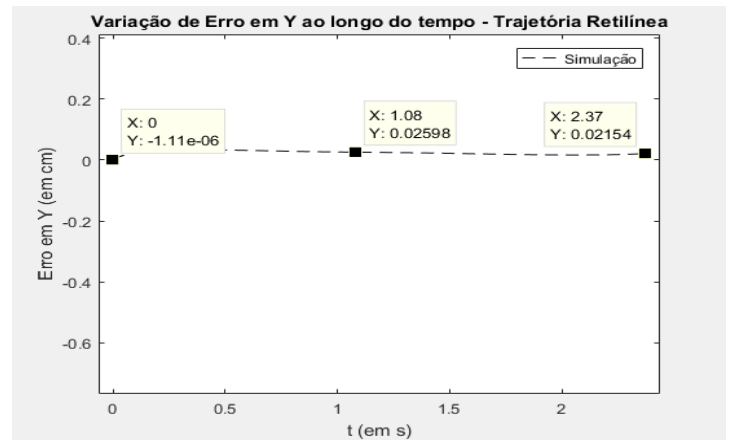
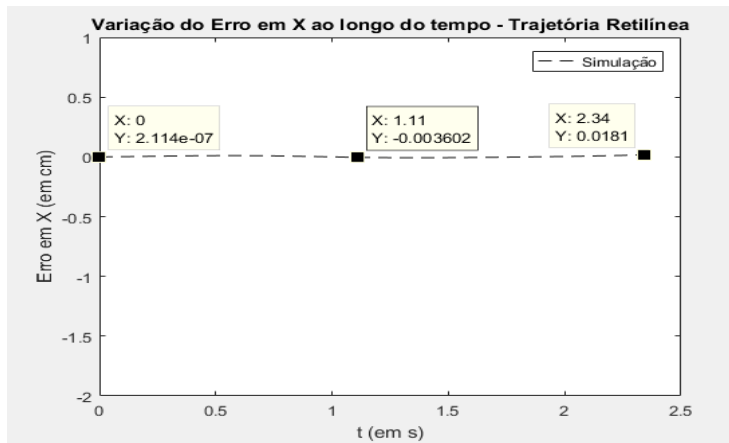


Figura 21 – Variação do erro em X e em Y antes da limitação de 80°/s nas juntas

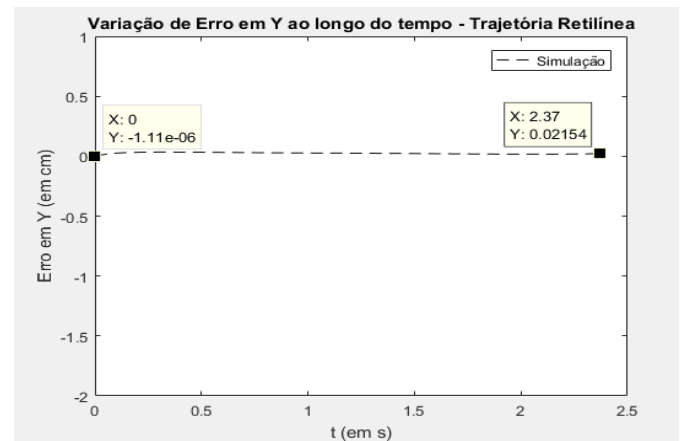
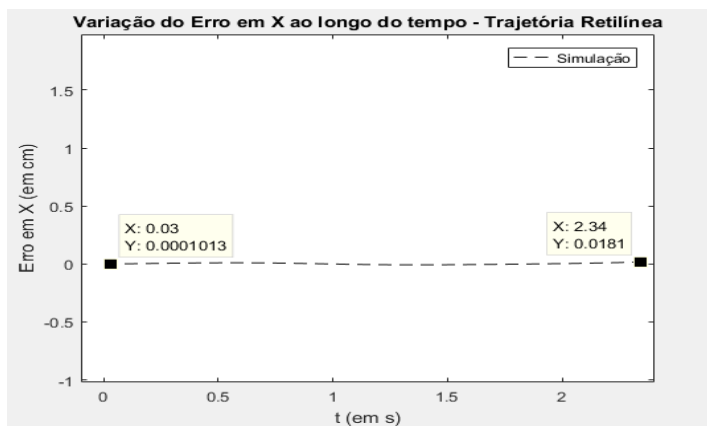


Figura 22 – Variação do erro em X e em Y depois da limitação de 80°/s nas juntas

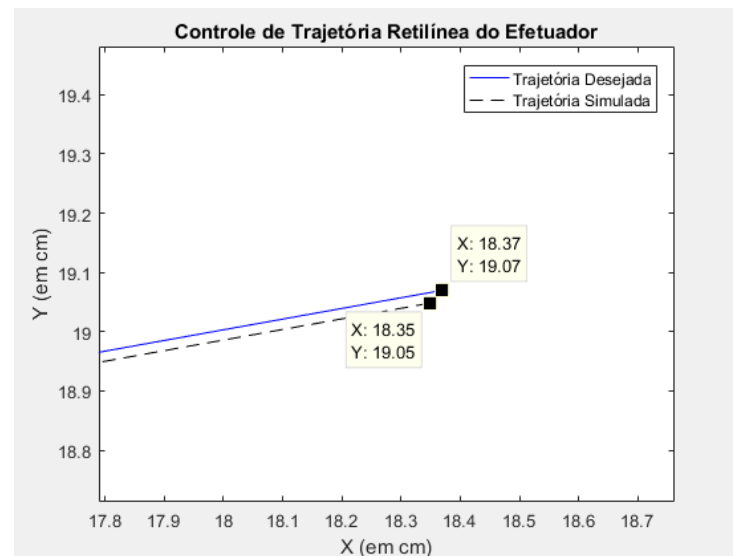
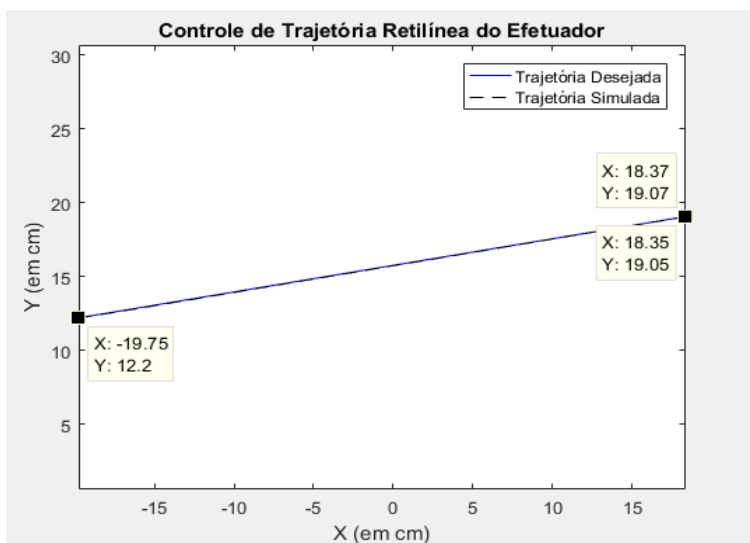


Figura 23- Controle de trajetoria sem a limitação de 80°/s nas juntas

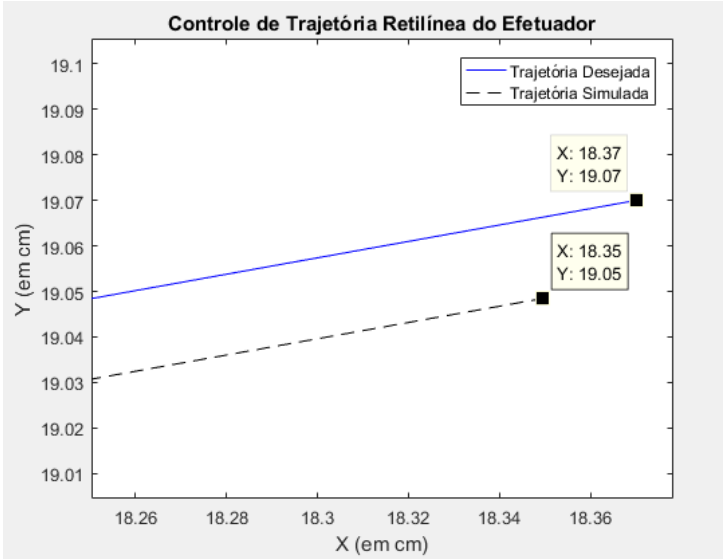
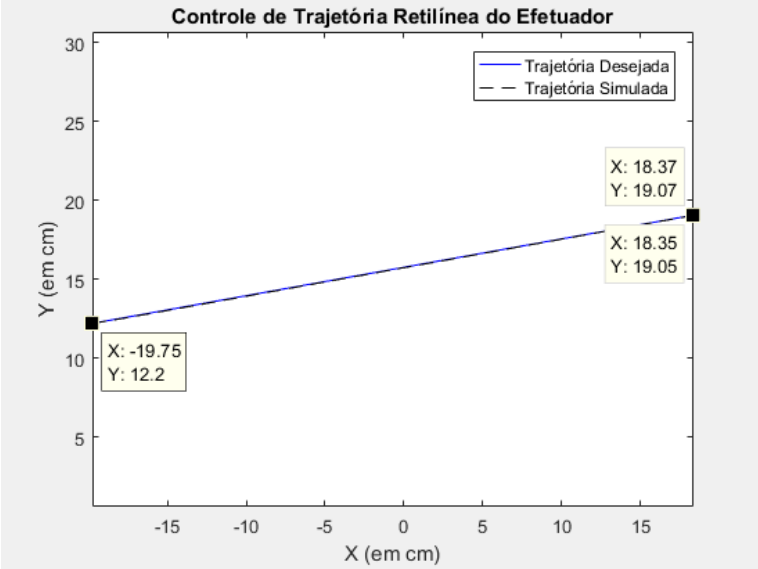


Figura 24 - Controle de trajetória com limitação de 80°/s nas juntas

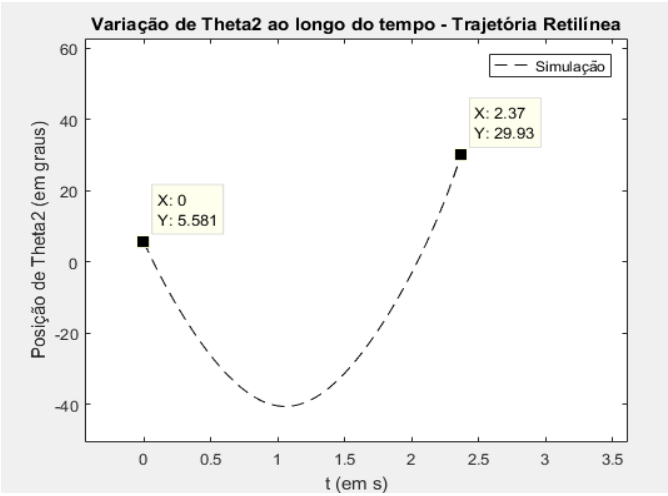
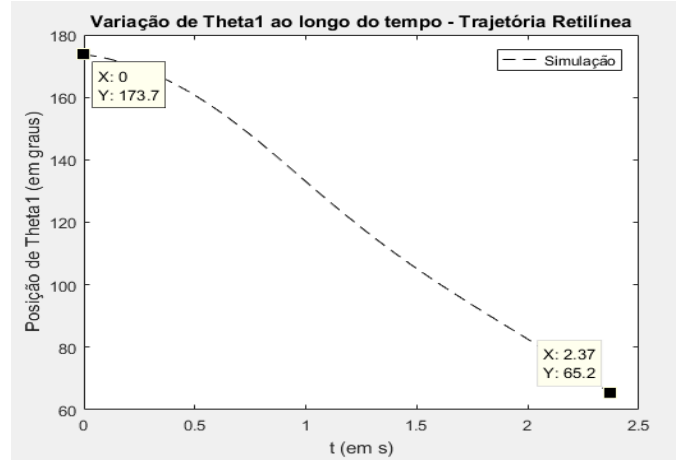


Figura 25 – Variação de theta1 e theta2 sem as limitações de 80°/s nas juntas

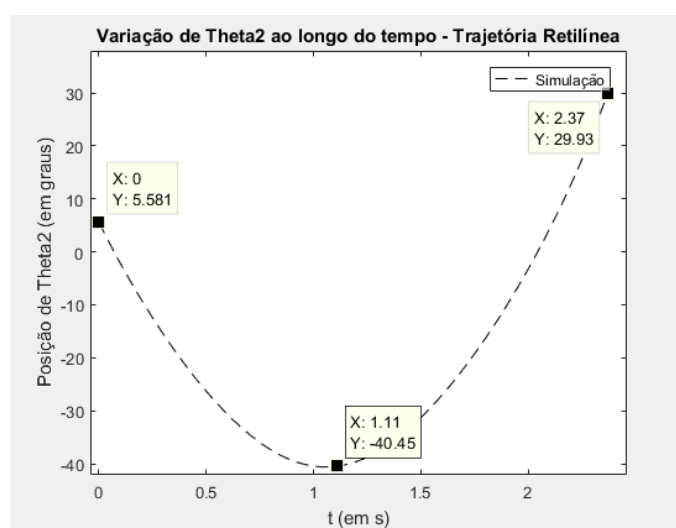
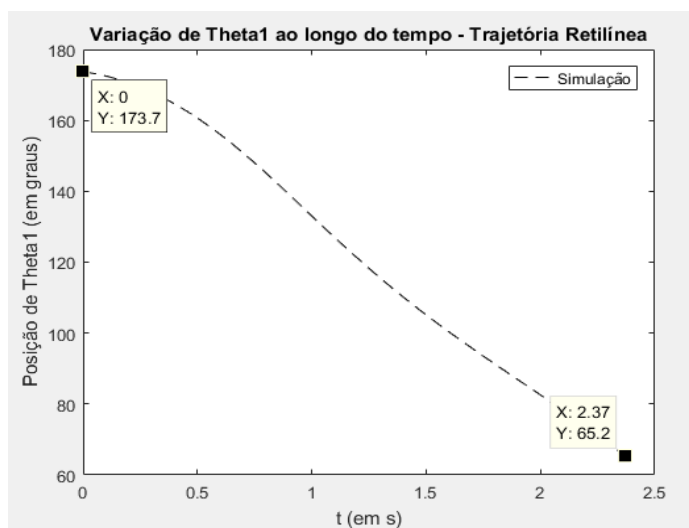


Figura 26 – Variação de theta1 e theta2 com as limitações de 80°/s nas juntas

Observando as imagens do antes e depois da restrição de velocidade pode-se notar que não houve melhora ou piora significativa no erro em regime transiente ou permanente do sistema, pois estes erros foram praticamente os mesmos como demonstrado nas figuras a cima.

- **SEGUNDO CASO :**

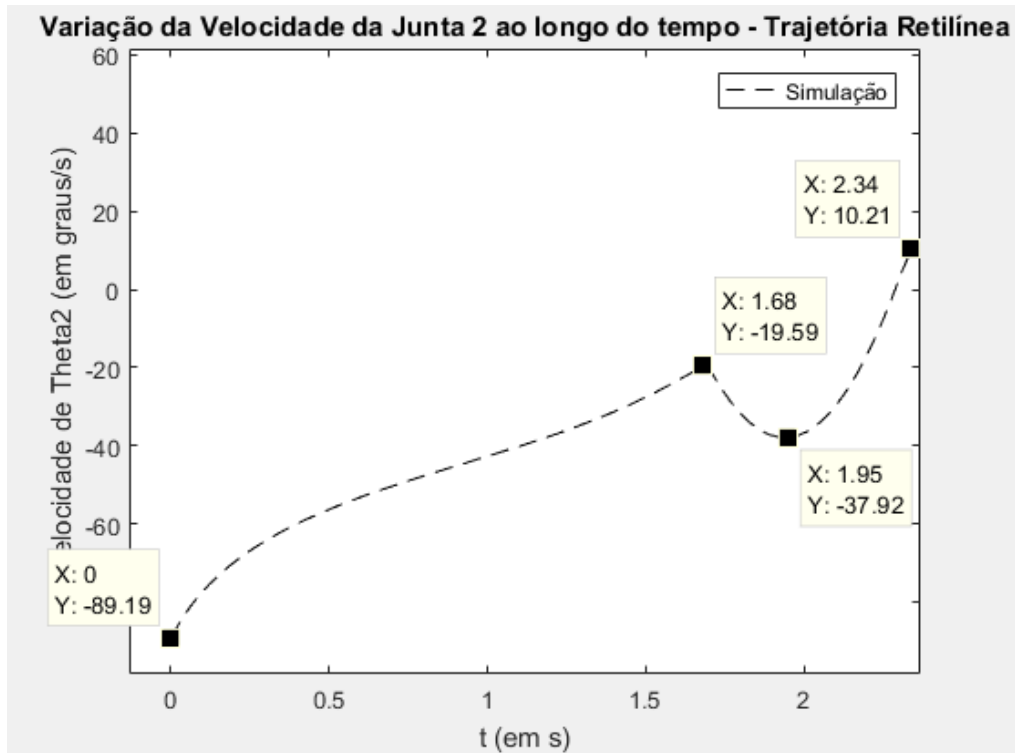


Figura 26 –Variação da velocidade da junta 2 ao longo do tempo sem limitação.

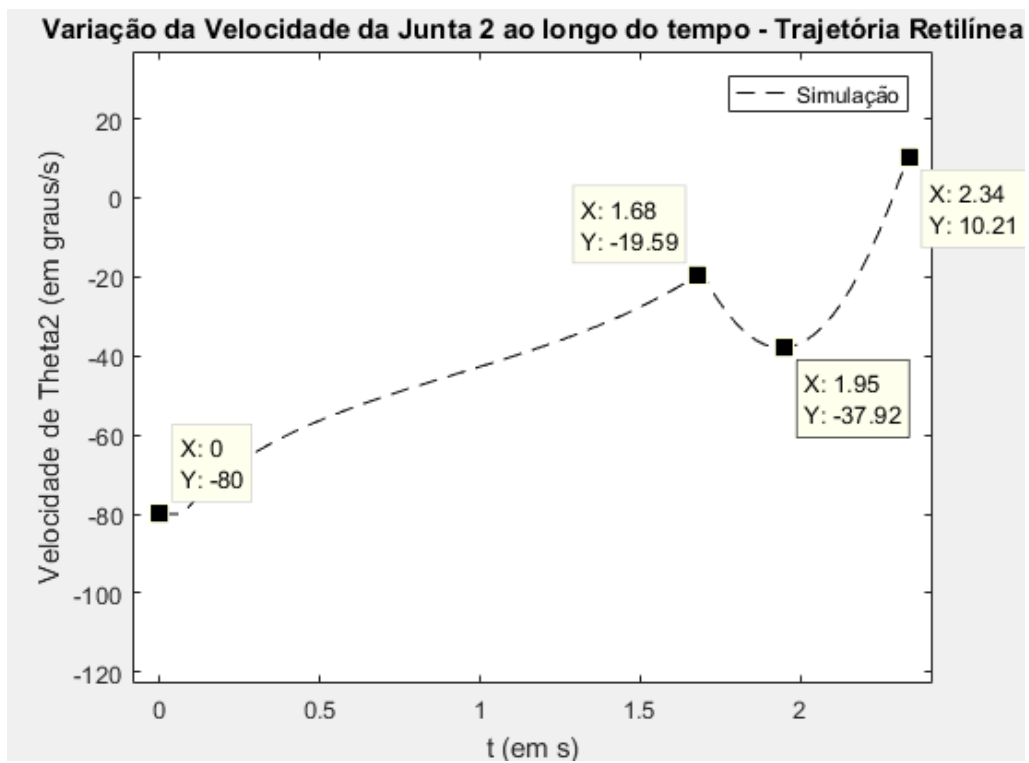


Figura 27 –Variação da velocidade da junta 2 ao longo do tempo com limitação.

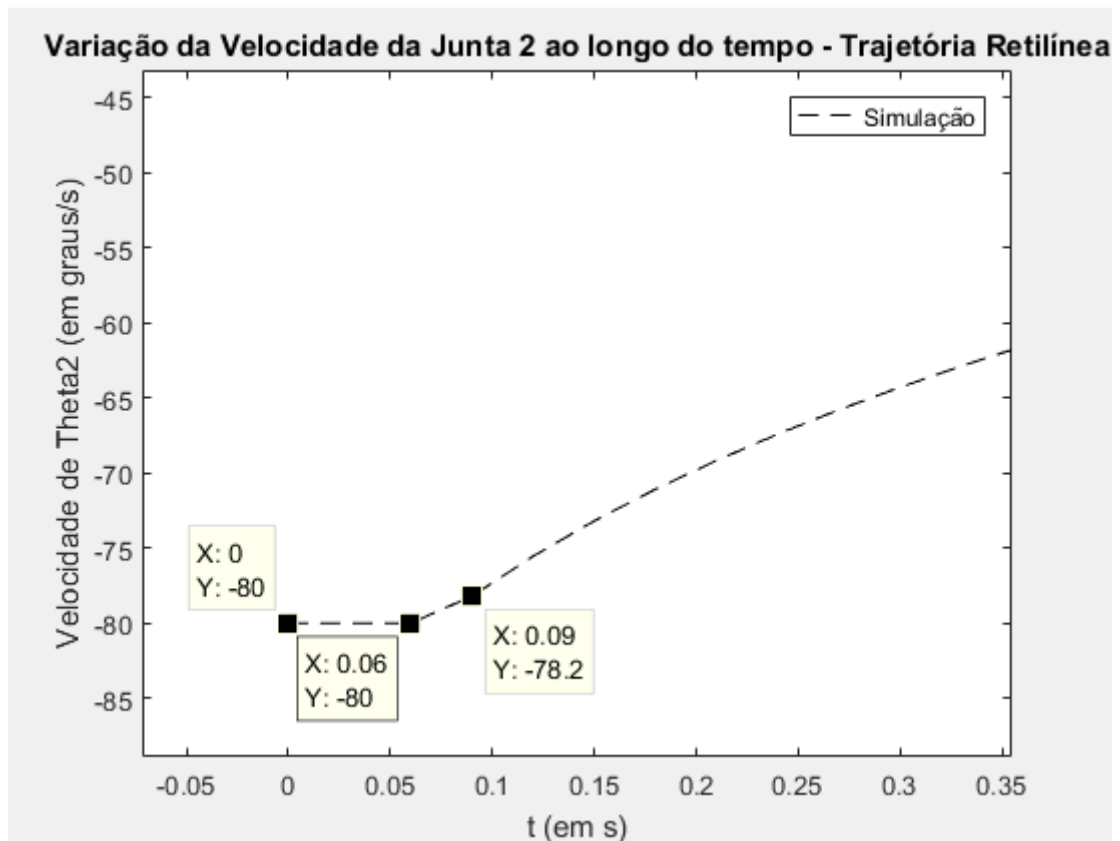


Figura 28–Variação da velocidade da junta 2 ao longo do tempo com limitação e ampliação no início.

As imagens 27 e 28 demonstram as limitações requeridas pelo exercício a $80^\circ/\text{s}$ para a velocidade angular da junta 2. É observado que para a faixa de -80 a $80^\circ/\text{s}$ os valores não foram modificados, já fora deste intervalo ocorre uma saturação em $80^\circ/\text{s}$. Como pode ser verificado na Figura 26, existe uma região onde é necessário limitar a variação de velocidade. Esta região é compreendida em $0 \leq t(\text{s}) \leq 0.06$.

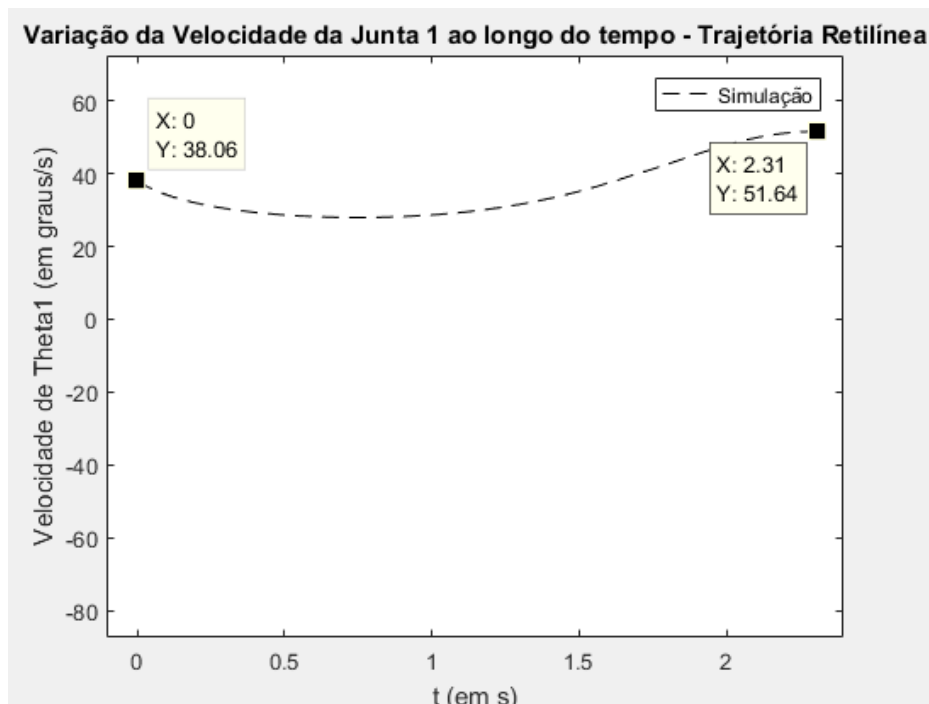


Figura 29– Variação da velocidade da junta 1 ao longo do tempo sem limitação.

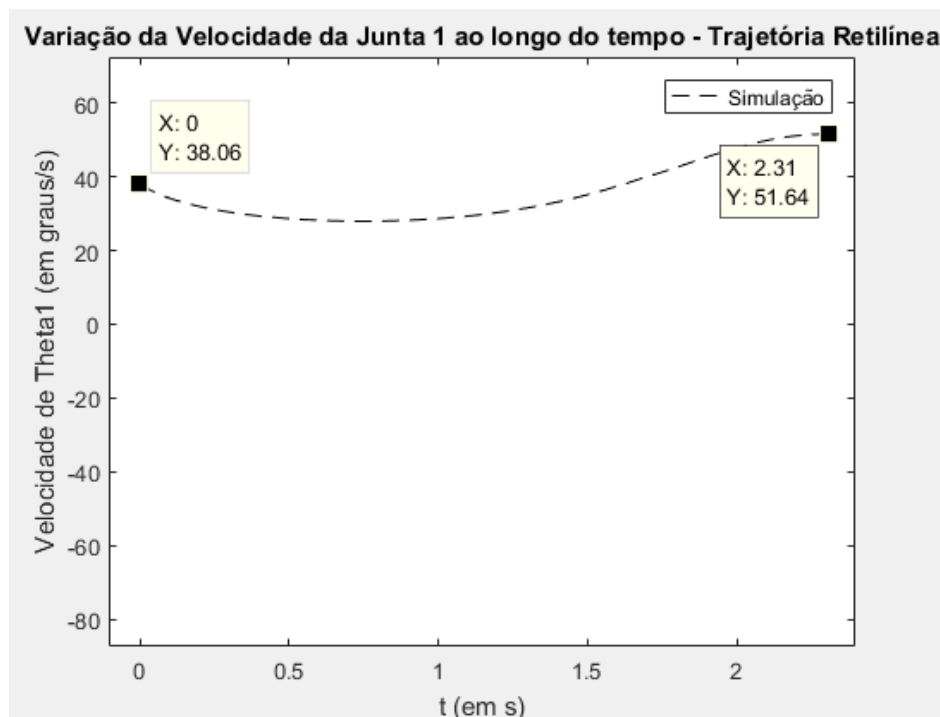


Figura 30 – Variação da velocidade da junta 1 ao longo do tempo com limitação

As imagens 28 e 29 demonstram as velocidades das juntas antes e depois da limitação para $80^\circ/\text{s}$ na junta 1. É observado que os valores assumidos pela velocidades da junta 1 se encontram na faixa de -80 a $80^\circ/\text{s}$ então os valores não sofreram alterações devido a esta restrição.

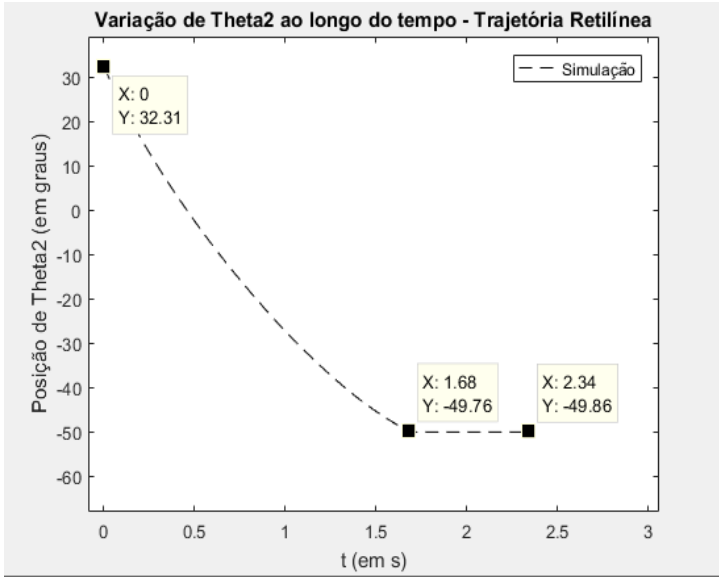
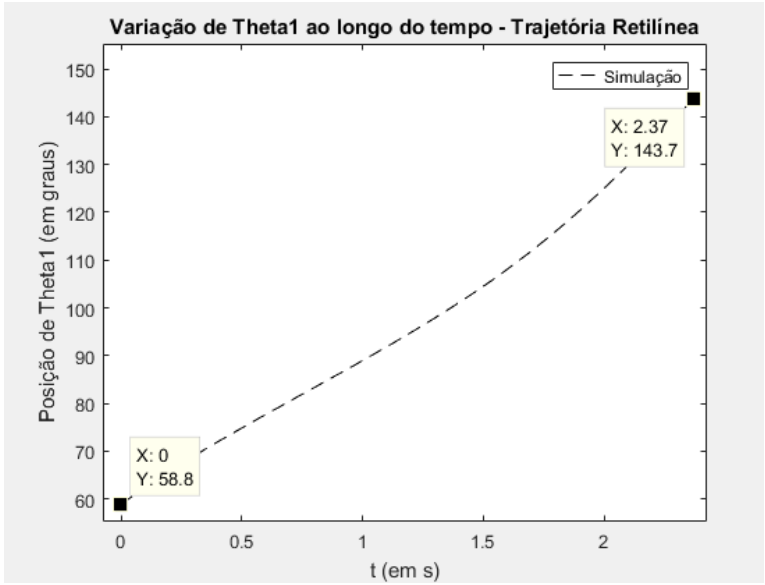


Figura 31– Variação de theta1 e theta2 sem limitação

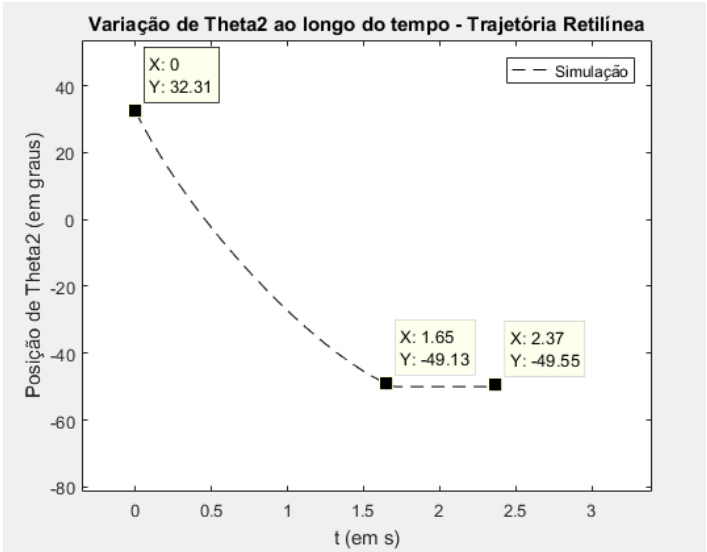
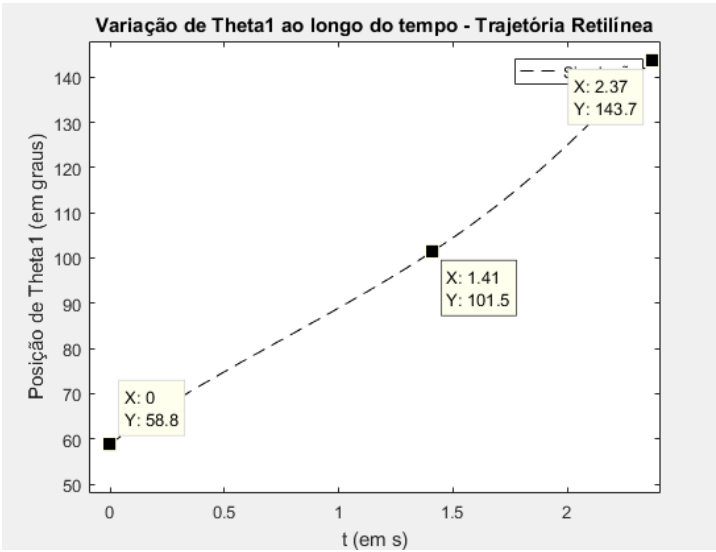


Figura 32 – Variação de theta1 e theta2 com limitação

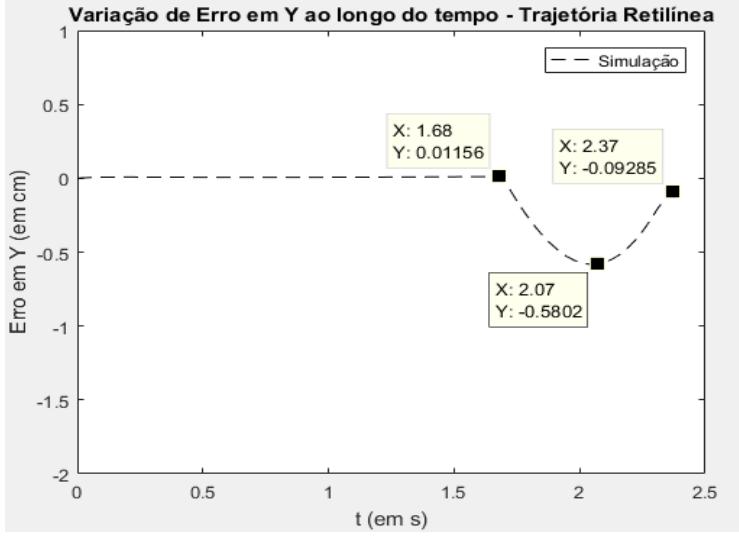
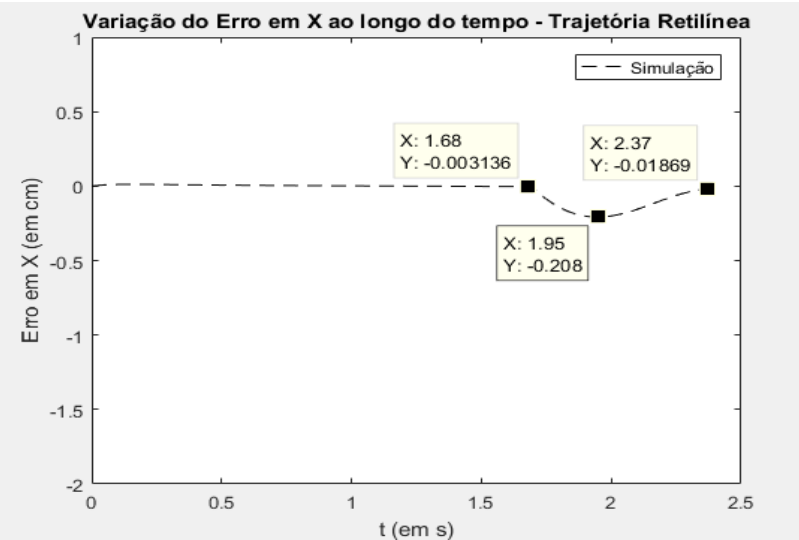


Figura 33 – Variação do erro em X e em Y antes da limitação de 80°/s nas juntas

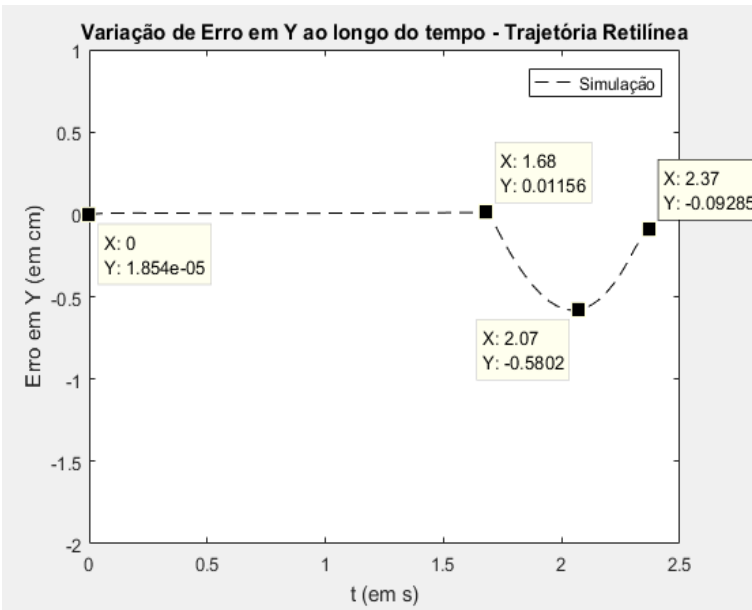
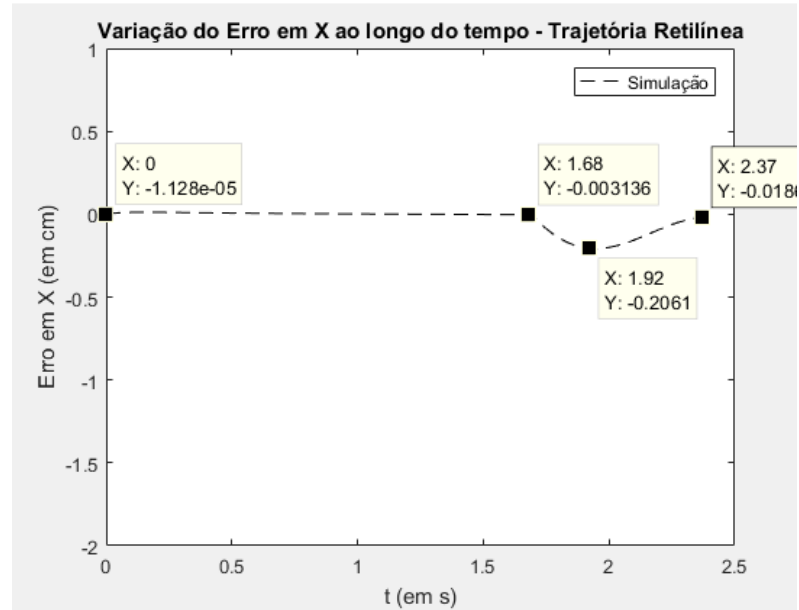


Figura 34 – Variação do erro em X e em Y depois da limitação de 80°/s nas juntas

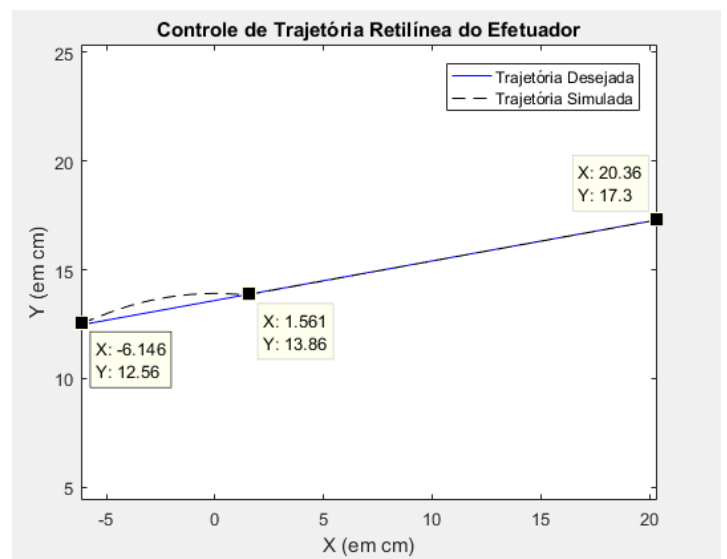
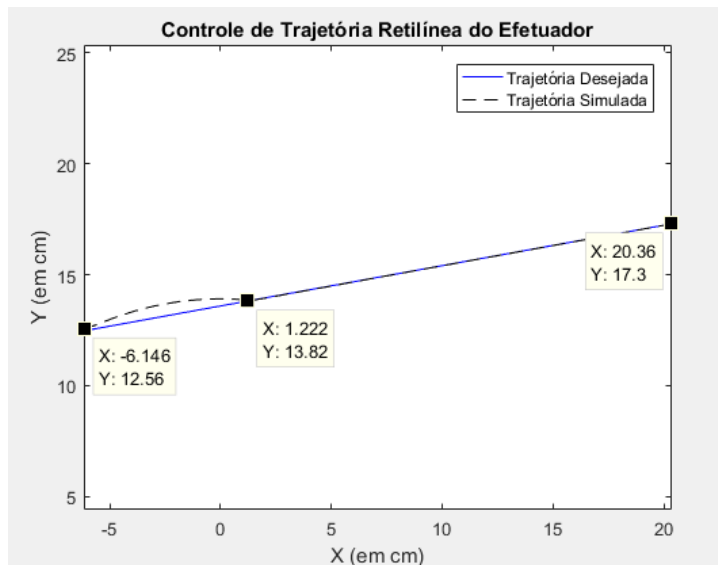


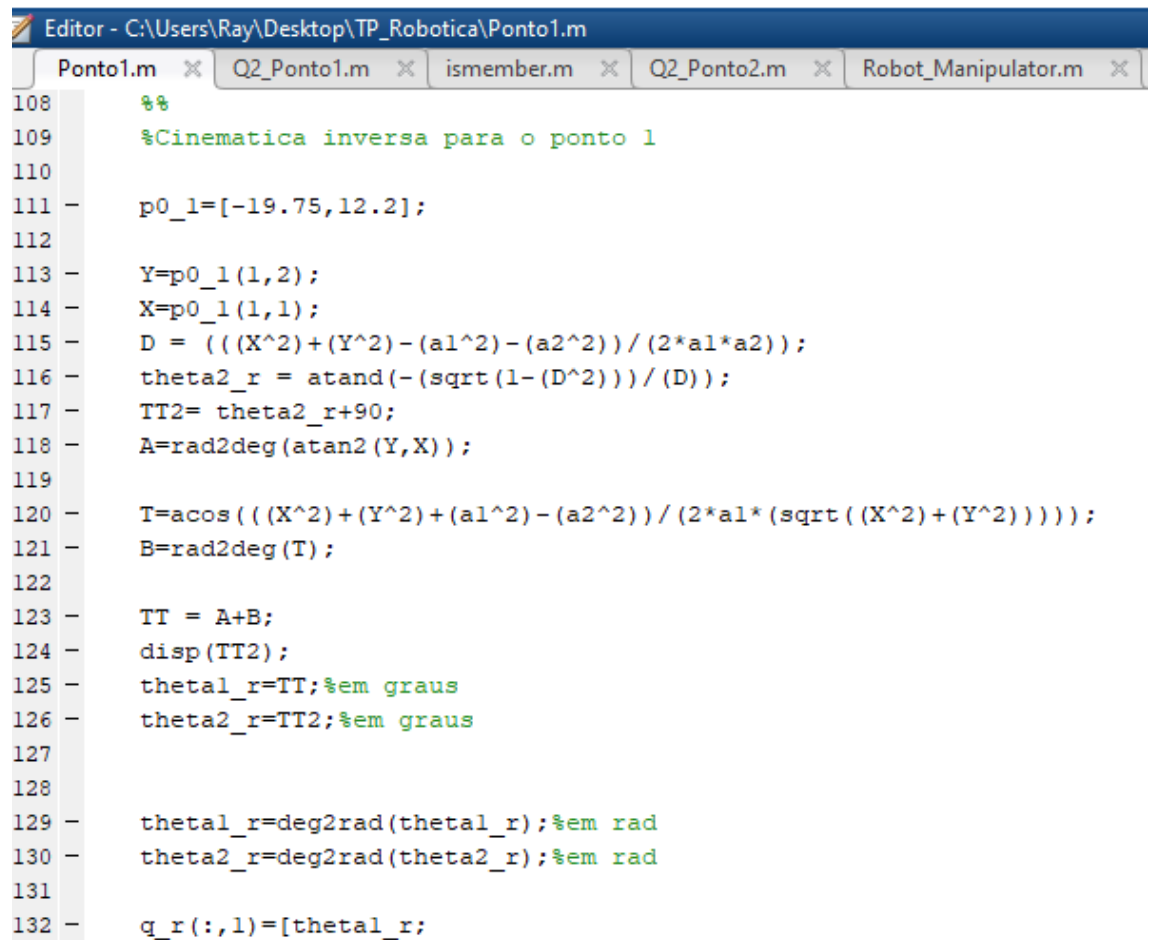
Figura 35 – Controle da trajetória. A esquerda corresponde ao sistema depois da limitação e a imagem a direita corresponde o sistema antes da limitação de $80^\circ/\text{s}$ nas juntas .

Observando as imagens do antes e depois da restrição de velocidade pode-se notar que não houve melhora ou piora significativa no erro em regime transiente ou permanente do sistema, pois estes erros foram praticamente os mesmos como demonstrado nas figuras a cima. Esperava-se que com estas limitações impostas o erro em X e em Y tenderia a aumentar devido contrair as velocidades no início e no término da trajetória de modo que o robô não conseguiria atingir o ponto final desejado. Logo, pode-se dizer que este limitador não é tão restritivo a ponto de causar uma variação significativa no sistema para estas configurações.

1.3 CODIGOS UTILIZADOS:

Dado o script do Matlab concedido pelo professor, foi proposto a mudança dos pontos cartesianos que já estavam nesse mesmo script e inserido os pontos iniciais e finais na trajetória retilínea. Além disso, foi calculado os θ_1 e θ_2 a partir da posição inicial do robô e codificado uma limitação para a velocidade das juntas em $80^\circ/\text{s}$. Abaixo é possível ver onde o código foi alterado.

- **PRIMEIRO CASO:**



The image shows a MATLAB editor window with the title bar "Editor - C:\Users\Ray\Desktop\TP_Robotica\Ponto1.m". The window contains several tabs: "Ponto1.m", "Q2_Ponto1.m", "ismember.m", "Q2_Ponto2.m", and "Robot_Manipulator.m". The code in the "Ponto1.m" tab is as follows:

```
108 %%  
109 %Cinematica inversa para o ponto 1  
110  
111 - p0_1=[-19.75,12.2];  
112  
113 - Y=p0_1(1,2);  
114 - X=p0_1(1,1);  
115 - D = ((X^2)+(Y^2)-(a1^2)-(a2^2))/(2*a1*a2);  
116 - theta2_r = atand(-(sqrt(1-(D^2)))/(D));  
117 - TT2= theta2_r+90;  
118 - A=rad2deg(atan2(Y,X));  
119  
120 - T=acos(((X^2)+(Y^2)+(a1^2)-(a2^2))/(2*a1*(sqrt((X^2)+(Y^2)))));  
121 - B=rad2deg(T);  
122  
123 - TT = A+B;  
124 - disp(TT2);  
125 - theta1_r=TT;%em graus  
126 - theta2_r=TT2;%em graus  
127  
128  
129 - theta1_r=deg2rad(theta1_r);%em rad  
130 - theta2_r=deg2rad(theta2_r);%em rad  
131  
132 - q_r(:,1)=[theta1_r;
```

Figura 36 – Calculo de theta1 e theta2 a partir da posição inicial do robô para o primeiro caso.

```

64
65 %Definição da Trajetória Retilínea
66 %%
67 % Ponto 1
68 p0_r=[-19.75,12.2];
69 p1_r=[18.37,19.07];
70
71
72
73 Dx=p1_r(1)-p0_r(1);
74 Dy=p1_r(2)-p0_r(2);
75 A=Dy/Dx;
76 b=p0_r(1,2)-A*p0_r(1,1);
77 k=0;
78 n_pontos_r=80;
79 for i=p0_r(1,1):(p1_r(1,1)-p0_r(1,1))/(n_pontos_r-1):p1_r(1,1),
80     k=k+1;
81     tx_r(k)=i;
82     ty_r(k)=A*tx_r(k)+b ;
83 end
84 %%
85
86 %Definição do Tempo da Trajetória Retilínea e do Intervalo de Amostragem

```

Figura 37– Pontos inicial e final da trajetoria referentes a minha matricula para o primeiro caso

```

144 q_r(1,i+1)=thetal_limSup;
145 end
146 if q_r(1,i+1)<thetal_limInf, %Limite Mecânico Medido
147     q_r(1,i+1)=thetal_limInf;
148 end
149 if q_r(2,i+1)>theta2_limSup, %Limite Mecânico Medido
150     q_r(2,i+1)=theta2_limSup;
151 end
152 if q_r(2,i+1)<theta2_limInf, %Limite Mecânico Medido
153     q_r(2,i+1)=theta2_limInf;
154 end
155 if dq_r(1,i) >= (80*pi/180), %Velocidade máxima limitada para junta 1 em 80°/s
156     dq_r(1,i) = 80*pi/180;
157 end
158 if dq_r(1,i) <= (-80*pi/180), %Velocidade máxima limitada para junta 1 em -80°/s
159     dq_r(1,i) = -80*pi/180;
160 end
161 if dq_r(2,i) >= (80*pi/180), %Velocidade máxima limitada para junta 2 em 80°/s
162     dq_r(2,i) = 80*pi/180;
163 end
164 if dq_r(2,i) <= (-80*pi/180), %Velocidade máxima para junta em 80°/s (limite inferior)
165     dq_r(2,i) = -80*pi/180;
166 end
167 end
168 end

```

Figura 38: Parte do código referente a limitação de velocidades das juntas

• SEGUNDO CASO:

```

Editor - C:\Users\Ray\Desktop\TP_Robotica\Ponto2.m
Ponto1.m  Q2_Ponto1.m  ismember.m  Q2_Ponto2.m  Robot_Manipulator.m  Ponto2.m
103  %%
104  %Cinematica inversa para o ponto 2
105  - p0_2=[20.36 ,17.3];
106  - Y=p0_2(1,2);
107  - X=p0_2(1,1);
108  - D = ((X^2)+(Y^2)-(a1^2)-(a2^2))/(2*a1*a2));
109  - theta2_r = atand(-(sqrt(1-(D^2)))/(D)); %%
110  - TT2= theta2_r+90;
111  - A=rad2deg(atan2(Y,X));
112  - T=acos(((X^2)+(Y^2)+(a1^2)-(a2^2))/(2*a1*(sqrt((X^2)+(Y^2)))));
113  - B=rad2deg(T);
114
115  - TT = A+B;
116  - disp(TT2);
117  - theta1_r=TT;%em graus
118  - theta2_r=TT2;%em graus
119

```

Figura 39 – Calculo de theta1 e theta2 a partir da posição inicial do robô para o primeiro caso.

```

Editor - C:\Users\Ray\Desktop\TP_Robotica\Ponto2.m
Ponto1.m  Q2_Ponto1.m  ismember.m  Q2_Ponto2.m  Robot_Manipulator.m  Ponto2.m
62  %Definição da Trajetória Retilínea
63  %%
64  %Ponto 2
65  - p0_r=[20.36,17.3];
66  - p1_r=[-6.165,12.47];
67
68  - Dx=p1_r(1)-p0_r(1);
69  - Dy=p1_r(2)-p0_r(2);
70  - A=Dy/Dx;
71  - b=p0_r(1,2)-A*p0_r(1,1);
72  - k=0;
73  - n_pontos_r=80;
74  - for i=p0_r(1,1):(p1_r(1,1)-p0_r(1,1))/(n_pontos_r-1):p1_r(1,1),
75  -     k=k+1;
76  -     tx_r(k)=i;
77  -     ty_r(k)=A*tx_r(k)+b ;
78  - end
79  %%

```

Figura 40– Pontos inicial e final da trajetoria referentes a minha matricula para o segundo caso.

```
Editor - C:\Users\Ray\Desktop\TP_Robotica\Ponto2Limitado.m
Ponto1.m  Q2_Ponto1.m  ismember.m  Q2_Ponto2.m  Robot_Manipulator.m  Ponto2.m  Ponto2Limitado.m

141 -         if q_r(1,i+1)>thetal_limSup, %Limite Mecânico Medido
142 -             q_r(1,i+1)=thetal_limSup;
143 -         end
144 -         if q_r(1,i+1)<thetal_limInf, %Limite Mecânico Medido
145 -             q_r(1,i+1)=thetal_limInf;
146 -         end
147 -         if q_r(2,i+1)>theta2_limSup, %Limite Mecânico Medido
148 -             q_r(2,i+1)=theta2_limSup;
149 -         end
150 -         if q_r(2,i+1)<theta2_limInf, %Limite Mecânico Medido
151 -             q_r(2,i+1)=theta2_limInf;
152 -         end
153 -         if dq_r(1,i) >= (80*pi/180), %Velocidade máxima limitada para junta 1 em 80°/s
154 -             dq_r(1,i) = 80*pi/180;
155 -         end
156 -         if dq_r(1,i) <= (-80*pi/180), %Velocidade máxima limitada para junta 1 em -80°/s
157 -             dq_r(1,i) = -80*pi/180;
158 -         end
159 -         if dq_r(2,i) >= (80*pi/180), %Velocidade máxima limitada para junta 2 em 80°/s
160 -             dq_r(2,i) = 80*pi/180;
161 -         end
162 -         if dq_r(2,i) <= (-80*pi/180), %Velocidade máxima para junta em 80°/s (limite inferior)
163 -             dq_r(2,i) = -80*pi/180;
164 -         end
165 -     end
<
```

Figura 41: Parte do código referente a limitação de velocidades das juntas

Os *scripts* recorridos para a execução dessa etapa do trabalho estão disponíveis em :

https://github.com/RayBasilio123/tp_Robotca/archive/refs/heads/main.zip

Foram utilizados os arquivos .m :

- QUESTÃO 1 :
 - SEM A LIMITAÇÃO DAS VELOCIDADES:
 - PRIMEIRO CASO:
 - Ponto1
 - SEGUNDO CASO :
 - Ponto2
 - COM A LIMITAÇÃO DAS VELOCIDADES:
 - PRIMEIRO CASO
 - Ponto1Limitado
 - SEGUNDO CASO :
 - Ponto2limitado

2) Implemente o controle cinemático de posição sob as mesmas condições da posição inicial do robô e de trajetória para o robô planar de três graus de liberdade com juntas de revolução – Figura 2. Considere o comprimento do terceiro elo (a_3) igual a 5cm, $\theta_{3_mín} = -135^\circ$ e $\theta_{3_máx} = +135^\circ$. Manter no script a pose do robô em repouso com $\alpha_2 = -\pi/2$ (alfa) – antebraço perpendicular ao solo – não incluir α na cinemática direta relacionada ao movimento da terceira junta de revolução. Para reduzir o esforço computacional, podem reduzir a razão de 199 por 100 a fração do incremento do ângulo nos três laços de repetição com a função FOR para plotar o espaço de trabalho. Analise o comportamento do sinal de controle de cada junta do robô, e dos sinais de erro em X e Y, ao longo da trajetória linear para cada uma das situações. No relatório apresente a trajetória desejada e executada, bem como o espaço de trabalho. Realize as alterações no script que forem necessárias para implementar as condições supracitadas, bem como para plotar os gráficos da trajetória, posições angulares, velocidades angulares e sinais de erro. No livro do Siciliano a cinemática direta desse robô é apresentada na p. 69 e o Jacobiano na p. 113.

- Limites Mecânicos do Servo para o Theta 3:
 - Mínimo: 0° .
 - Máximo: $+180^\circ$.
- Limites Mecânicos do Servo para o Theta 2:
 - Mínimo: -50° .
 - Máximo: $+90^\circ$.
- Limites Mecânicos do Servo para o Theta 3:
 - Mínimo: -135° .
 - Máximo: $+135^\circ$.

- A partir do toolbox e pelo script do matlab temos:

- p1 (-19.75 ,12.2) :

- theta1_r = 158.4
- theta2_r = 28.4
- theta3_r = 83.7

- p2 (20,36 ,17,3) :

- theta1_r = 41.4
- theta2_r = 59.2
- theta3_r = 54.5

ROBÔ PLANAR DE TRÊS GRAUS DE LIBERDADE

PRIMEIRO CASO:

Tabela 1 - Parâmetros para a execução da trajetória 1

Posição inicial do Robô	Trajetória Retilínea				
-------------------------	----------------------	--	--	--	--

xo	yo	xi	yi	xf	yf
-19.75	12.2	-19.75	12.2	18.37	19.07

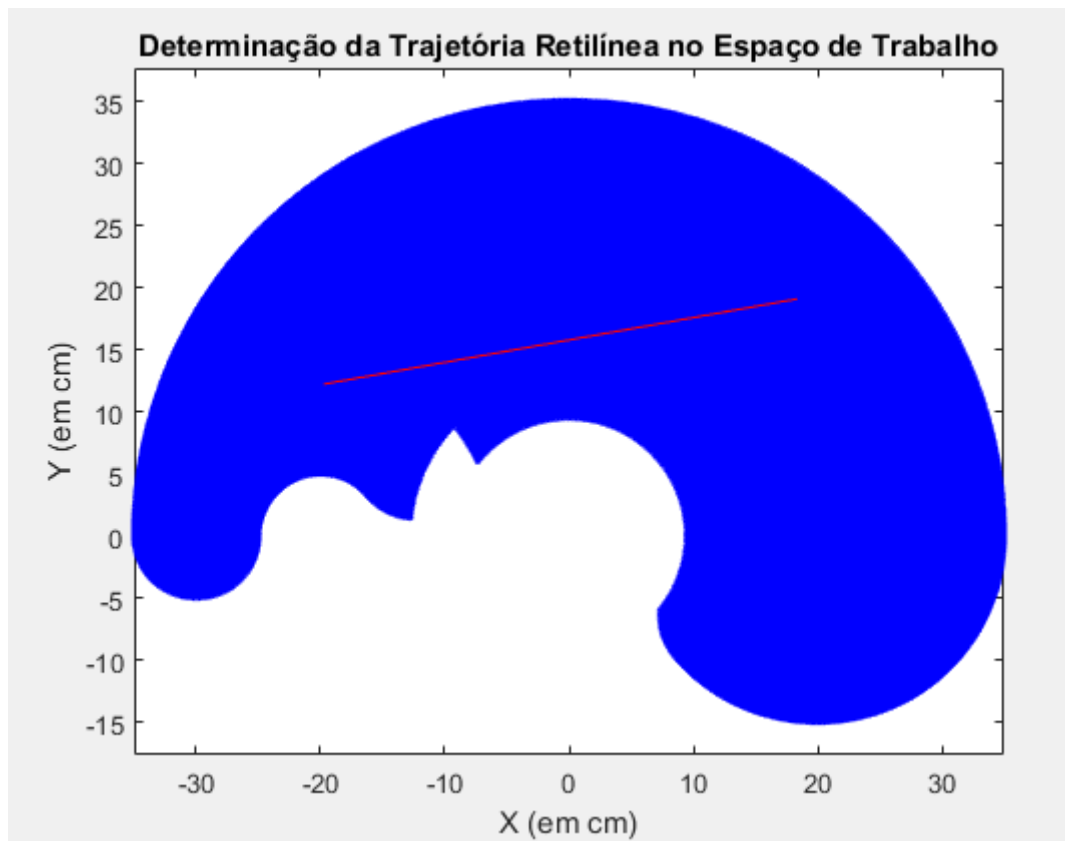


Figura 41 – Determinação da trajetória retilínea no espaço de trabalho 1

Na Figura 41 é possível verificar o espaço de trabalho do robô com três graus de liberdade em azul. Comparando com o espaço de trabalho da figura 1 (robô com 2 graus de liberdade) é visível que aumentando os graus de liberdade aumentou-se o espaço de trabalho consideravelmente e assim como na figura 1 a trajetória se encontra totalmente neste espaço.

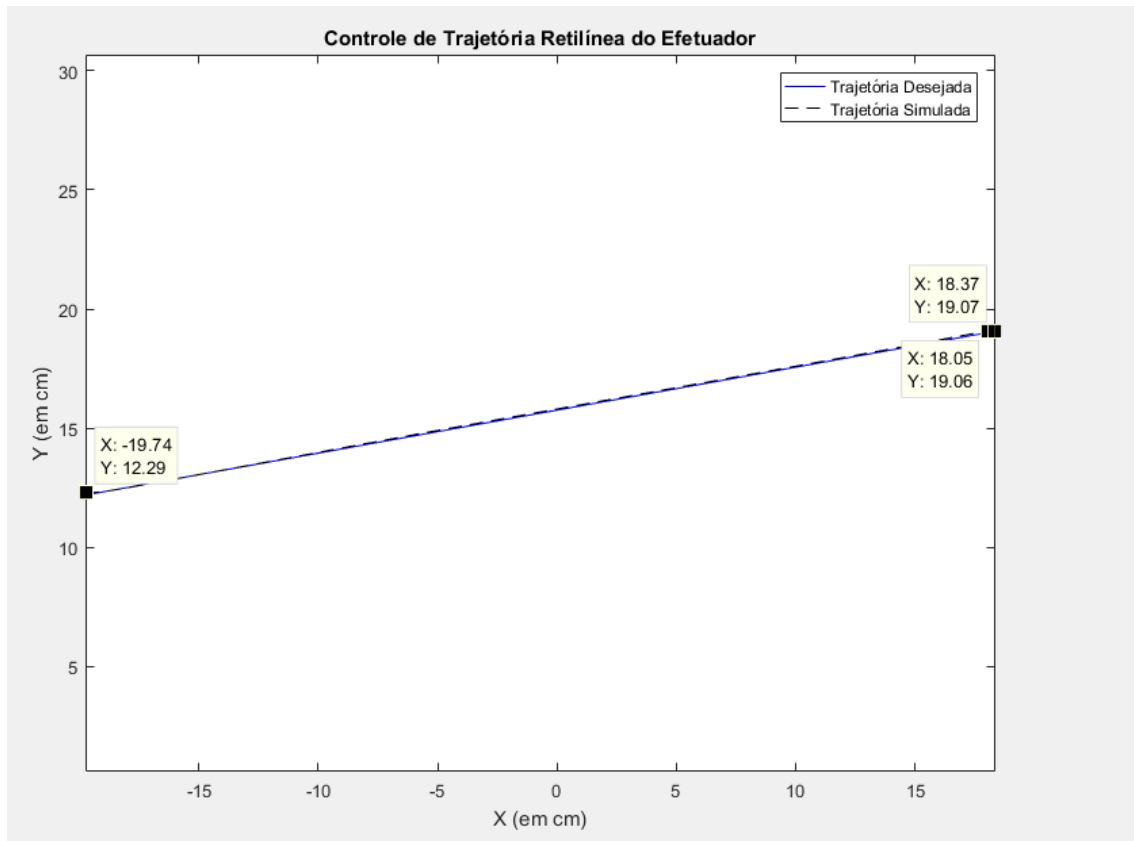


Figura 42 – Controle da trajetória retilínea do efetuador

Pela figura 42 , é demonstrado como foi feito o controle da trajetória, sendo que a trajetória linear simulada iniciou no ponto $p1s_xyi$ (-19.74; 12.29) e finalizou no ponto $p1s_xyi$ (18.35,19,06).Era desejado que a trajetória linear tivesse início em $p1d_xyi$ (-19.75,12.2) e terminasse no ponto $p1d_xyf$ (18.37,19.07) mas por possíveis arredondamentos feitos durante o código temos esse pequeno erro. Apesar disto, o efetuador conseguiu realizar a trajetória sem nenhum distúrbio durante o percurso da trajetória linear.

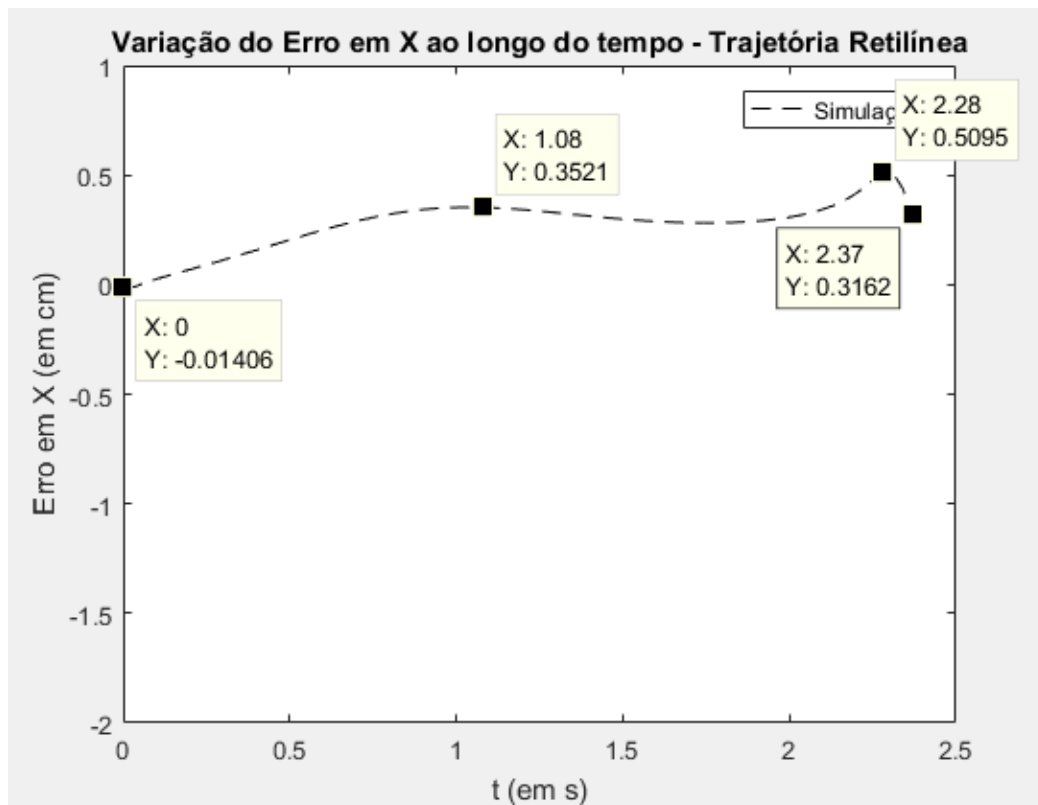


Figura 43 – Variação do erro no eixo X.

A Figura 43 demonstra o que já era esperado, como o ponto inicial do robô é coincidente com início da trajetória, o erro inicial em X é praticamente nulo, tendo pequenas variações durante a trajetória. Quando o sistema está no término de seu movimento o erro começa a aumentar terminando com o valor de 0.50.

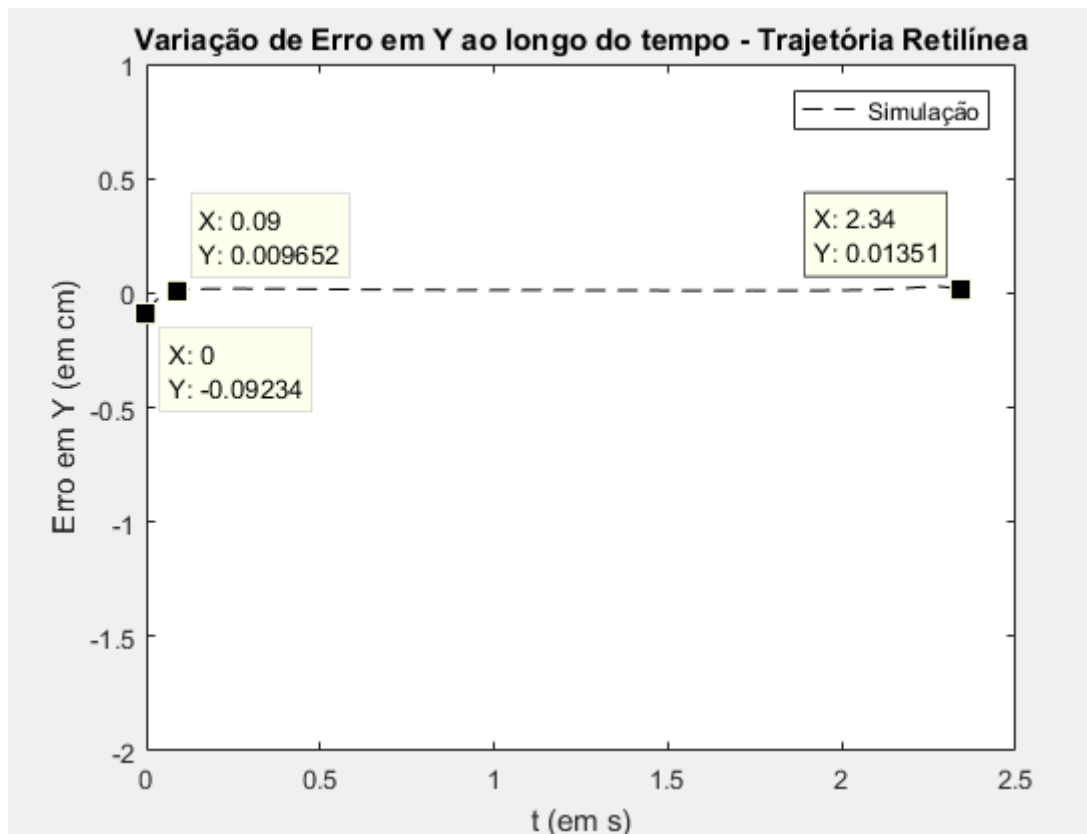


Figura 44 – Variação do erro no eixo Y.

A Figura 44 demonstra o que já era esperado, como o ponto inicial do robô é coincidente com início da trajetória, o erro inicial em Y é praticamente nulo, tendo pequenas variações durante a trajetória e terminando com o valor de 0.01351.

SEGUNDO CASO:

Tabela 2 - Parâmetros para a execução da trajetória 2

Posição inicial do Robô	Trajetória Retilínea				
-------------------------	----------------------	--	--	--	--

xo	yo	xi	Yi	xf	yf
20.36	17.3	20.36	17.3	-6.165	12.47

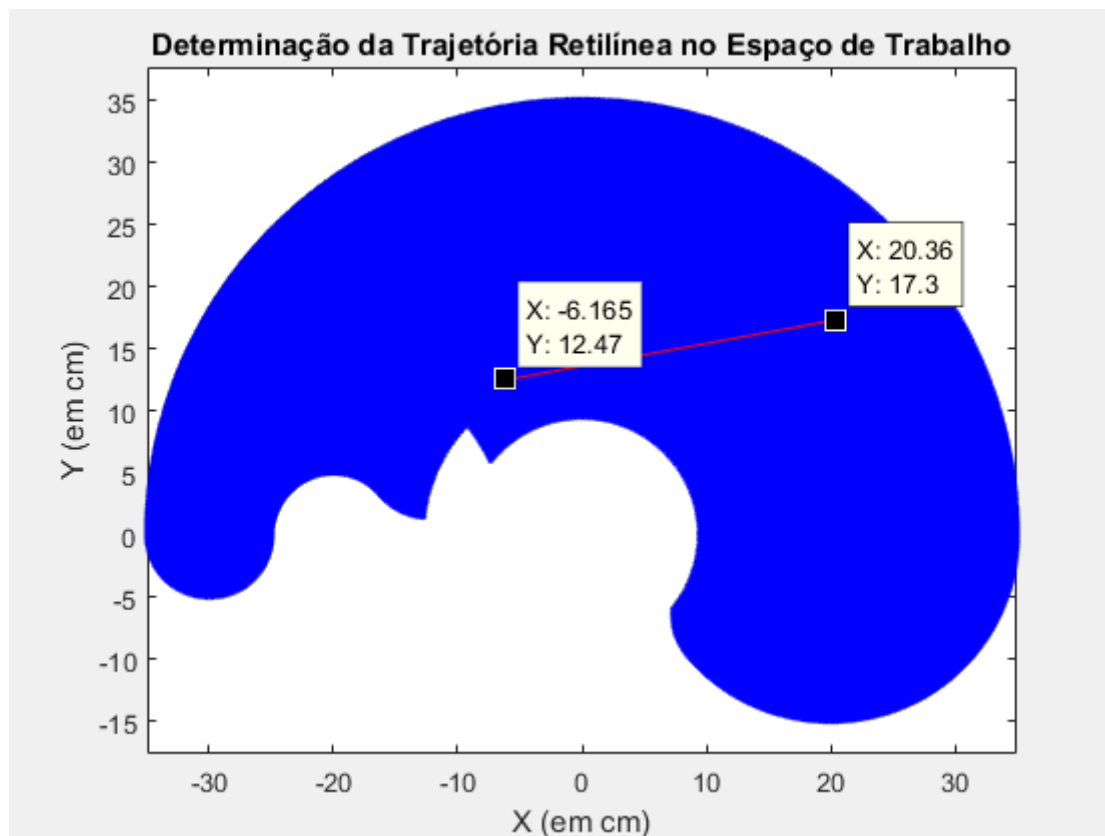


Figura 45 – Determinação da trajetória retilínea no espaço de trabalho 2

Como observado pela Figura 45, diferentemente de quando tínhamos apenas dois graus de liberdade na Figura 9, para a mesma trajetória, não passamos em uma singularidade, pois o espaço teve um aumento e com isso englobando toda a trajetória.

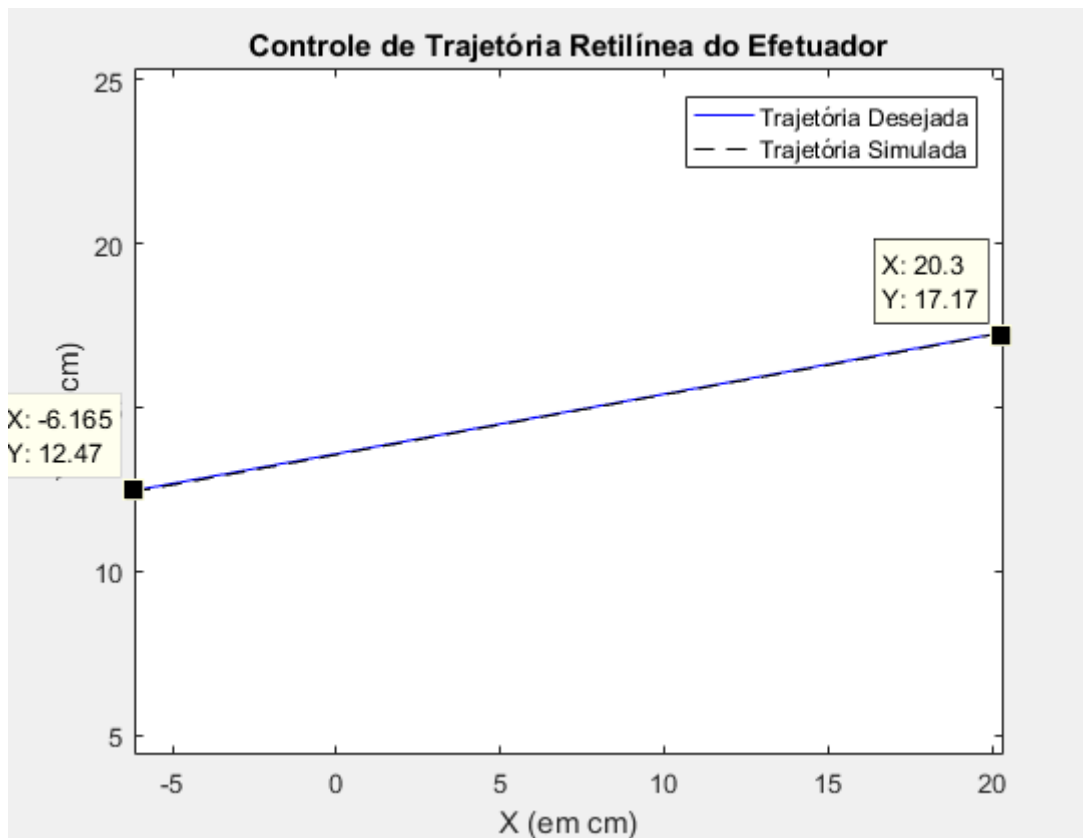


Figura 46 – Controle da trajetória retilínea do efetuador

Observa-se na Figura 46 como foi feito o controle da trajetória. É observado que não houve nenhum distúrbio crítico durante a trajetória de modo a permanecer durante o percurso de forma constante.

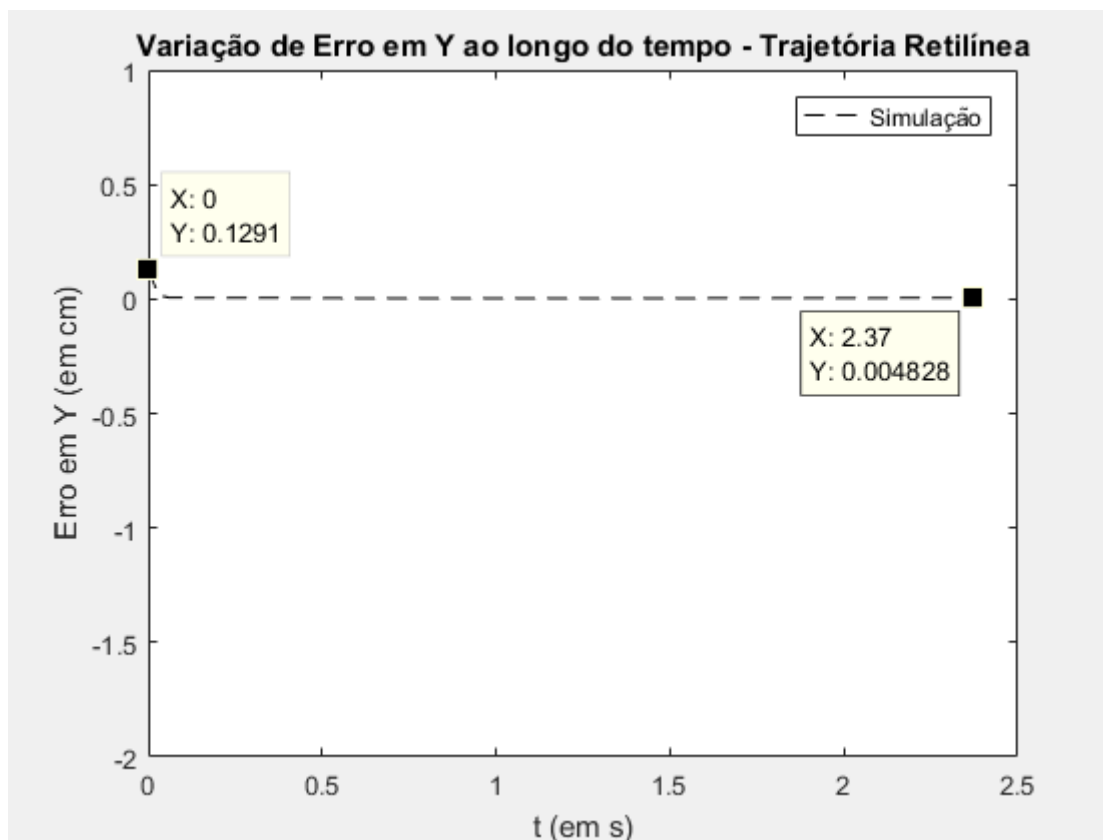


Figura 47 – Variação do erro em Y

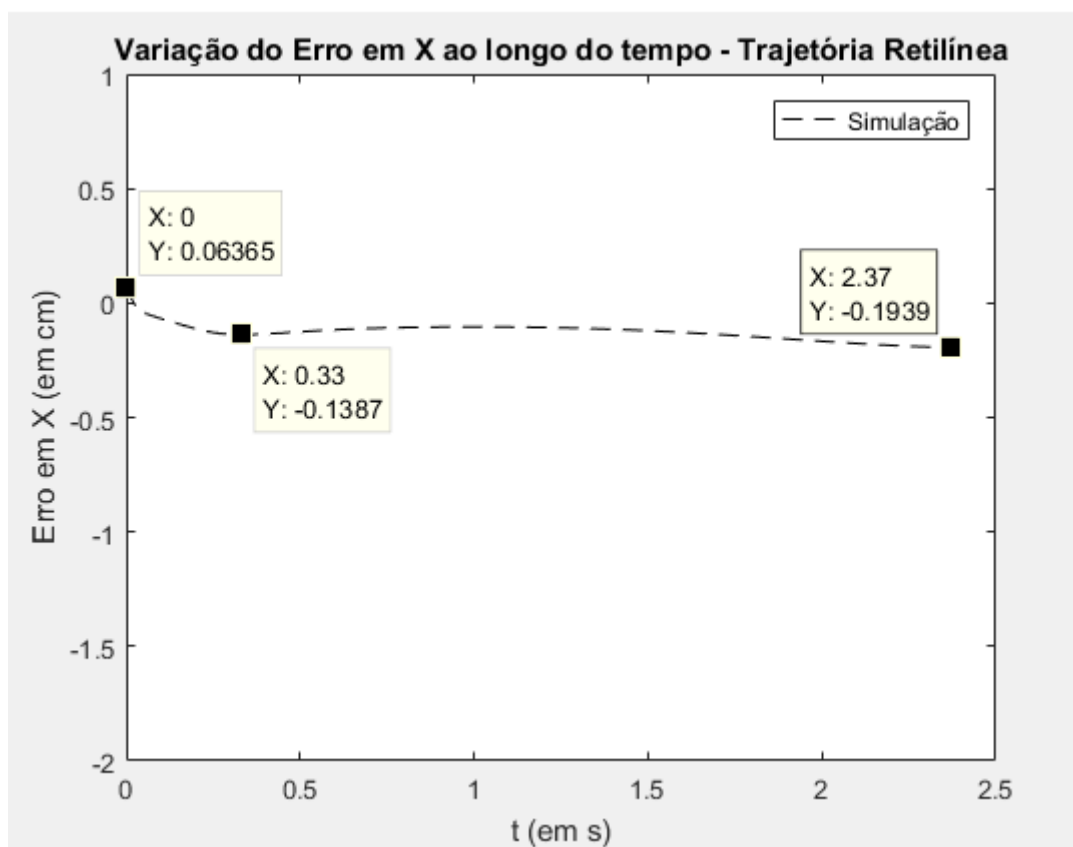


Figura 47 – Variação do erro em X

2.1 CODIGOS UTILIZADOS:

Para implementar esse tipo de robô, foi realizado algumas modificações durante o código, primeiramente achando o espaço de trabalho, que foi dado por DH (Denavit-Hartenberg), posteriormente o cálculo do jacobiano e pseudo jacobiano. No livro do Siciliano a cinemática direta desse robô é apresentada na p. 69 e o Jacobiano na p. 113.

$$T_3^0(q) = A_1^0 A_2^1 A_3^2 = \begin{bmatrix} c_{123} & -s_{123} & 0 & a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ s_{123} & c_{123} & 0 & a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 48 : Matriz DH para robô RRR

```

21 %Espaço de Trabalho
22 %%
23 i=0;
24
25 - theta1_limInf=0;%limite inferior de theta 1 em graus
26 - theta1_limSup=180;%limite superior de theta 1 em graus
27 - theta2_limInf=-50;%limite inferior de theta 2 em graus
28 - theta2_limSup=90;%limite superior de theta 2 em graus
29 - theta3_limInf=-135;%limite inferior de theta 3 em graus
30 - theta3_limSup=135;%limite superior de theta 3 em graus
31
32 - theta1_limInf=deg2rad(theta1_limInf);
33 - theta1_limSup=deg2rad(theta1_limSup);
34 - theta2_limInf=deg2rad(theta2_limInf);
35 - theta2_limSup=deg2rad(theta2_limSup);
36 - theta3_limInf=deg2rad(theta3_limInf);
37 - theta3_limSup=deg2rad(theta3_limSup);
38
39

```

Figura 48 : Adicionando os limites mecanicos dados pelo exercicio

Foi realizado um laço de repetição (for) para testar os ângulos para teta3, e também colocar as formulas para X(i) e Y(i), no qual estão na última coluna da matriz de DH:

```

40 - for thetal=thetal_limInf:(thetal_limSup-thetal_limInf)/100:thetal_limSup,
41 -     for theta2=theta2_limInf:(theta2_limSup-theta2_limInf)/100:theta2_limSup,
42 -         for theta3=theta3_limInf:(theta3_limSup-theta3_limInf)/100:theta3_limSup,
43 -             i=i+1;
44 -             %Matriz DH
45 -             x(i)=a1*cos(thetal)+a2*cos(thetal+theta2+alfa)+a3*cos(thetal+theta2+theta3);
46 -             y(i)=a1*sin(thetal)+a2*sin(thetal+theta2+alfa)+a3*sin(thetal+theta2+theta3);
47 -             if(x(i)>= 20.30 && x(i)<= 20.36 && y(i)>=17.2 && y(i)<=17.3)
48 -
49 -                 disp(x(i));
50 -                 disp(y(i));
51 -                 disp('Valor de theta 1 para o ponto (x,y):')
52 -                 disp(rad2deg(thetal))
53 -                 disp('Valor de theta 2 para o ponto (x,y):')
54 -                 disp(rad2deg(theta2))
55 -                 disp('Valor de theta 3 para o ponto (x,y):')
56 -                 disp(rad2deg(theta3))
57 -             end
58 -         end
59 -     end

```

Figura 49 : Espaço de trabalho

```

139 - J = [-a1*sin(q_r(1,i))-a2*sin(q_r(1,i)+q_r(2,i)+alfa)-a3*sin(q_r(1,i)+q_r(2,i)+alfa+q_r(3,i)), -a2*sin(q_r(1,i)+q_r(2,i)+alfa)-a3*sin(q_r(1,i)+q_r(2,
140 - J_inv = J'*inv((J'*J')));
141 - dq_r(:,i)=J_inv*u_r(:,i);
142 - q_r(:,i+1)=q_r(:,i)+dq_r(:,i)*dT_r(:,i);
143
144
145 - if q_r(1,i+1)>thetal_limSup, %Limite Mecânico Medido
146 -     q_r(1,i+1)=thetal_limSup;
147 - end
148 - if q_r(1,i+1)<thetal_limInf, %Limite Mecânico Medido
149 -     q_r(1,i+1)=thetal_limInf;
150 - end
151 - if q_r(2,i+1)>theta2_limSup, %Limite Mecânico Medido
152 -     q_r(2,i+1)=theta2_limSup;
153 - end
154 - if q_r(2,i+1)<theta2_limInf, %Limite Mecânico Medido
155 -     q_r(2,i+1)=theta2_limInf;
156 - end
157 - end
158 - end

```

Figura 50 : Cálculo do Pseudo-Jacobiano.

Foram utilizados os arquivos .m :

- QUESTÃO 2 :
 - PRIMEIRO CASO:
 - Q2_Ponto1
 - SEGUNDO CASO :
 - Q2_Ponto2

