

Atividade 2

Controle de posição de robôs de tração diferencial

Aluno: Mateus Soares Marques
Disciplina: Automação Inteligente
Professor: Antonio Marcus Nogueira Lima

Objetivos

- Implementar o controle de posição do robô de tração diferencial usando o modelo uniciclo
- Definir leis de controle baseadas em controle linear (controlador P e PD) e não-linear (à partir do segundo método de Lyapunov)
- Simular computacionalmente no Simulink e no CoppeliaSim os sistemas, considerando o robô Pioneer3DX
- Comparar os resultados obtidos entre as diferentes leis de controle implementadas
- Comparar os resultados entre as diferentes plataformas para simulação computacional

Roteiro

- Introdução
- Desenvolvimento
- Resultados
- Análise e conclusão

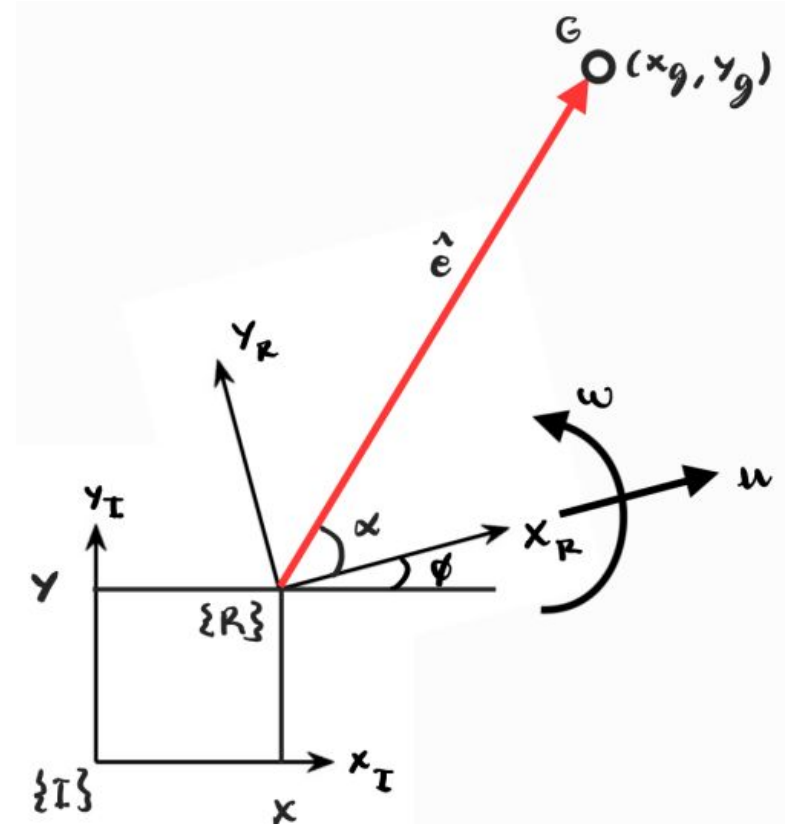
Introdução

Estabilização de ponto:

Procura-se uma lei de controle que leve o robô de uma pose inicial $[x_0 \ y_0 \ \phi_0]^T$ à uma posição alvo $(x_g \ y_g)$ definidas no plano inercial $\{I\}$ no qual o robô se move.

Sendo $\{R\}$ e G os referenciais do robô o ponto alvo, respectivamente; o problema é ilustrado na figura ao lado.

A velocidade linear é desenvolvida na direção x_R , e a velocidade angular é positiva no sentido anti-horário.



Introdução

Estabilização de pose

- Em coordenadas cartesianas

$$\begin{cases} \dot{x} = u \cos \phi \\ \dot{y} = u \sin \phi \\ \dot{\phi} = \omega \end{cases}$$

- Em coordenadas polares

$$\begin{cases} e = \sqrt{\Delta x^2 + \Delta y^2} \\ \alpha = \arctan\left(\frac{\Delta y}{\Delta x}\right) - \phi \end{cases}$$

$$\begin{cases} \dot{e} = -u \cos \alpha \\ \dot{\alpha} = \left(\frac{u}{e}\right) \sin \alpha - \omega \end{cases}$$

$$\begin{cases} \Delta x = x_g - x \\ \Delta y = y_g - y \end{cases}$$

Introdução

Estabilização de pose

- Em coordenadas cartesianas

O ponto de equilíbrio é dado pelo ponto alvo e o problema é definido em termos da pose inicial e da posição final.

- A generalização do problema é complexa
- O resultado de Brockett é aplicável

- Em coordenadas polares

O ponto de equilíbrio é dado pela origem, ou seja, quando todos os erros são nulos. Resolve-se indiretamente a estabilização de pose nos termos cartesianos levando o robô de um erro inicial ao erro nulo.

- A generalização do problema é simples (ponto de equilíbrio na origem do plano de fase)
- O resultado de Brockett não é aplicável

Introdução

Estabilização de pose

Reescreve-se o problema:

Seja um veículo unicycle $\{R\}$ inicialmente posicionado à uma distância não nula de uma ponto alvo G e que os erros de distância e orientação $[e \ \alpha]^T$ possam ser diretamente medidos para qualquer $e > 0$. Procura-se portanto uma lei de controle em malha fechada $[u \ \omega]^T = g(e, \alpha)$ que garanta que os erros serão assintoticamente levados ao ponto limite $[0 \ 0]^T$.

Desenvolvimento

Controle Linear

Controlador Proporcional (P):

$$\begin{cases} u = k_e e \\ \omega = k_\alpha \alpha \end{cases}$$

$$\begin{bmatrix} \dot{e} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -k_e e \cos \alpha \\ k_e \sin \alpha - k_\alpha \alpha \end{bmatrix} \\ \implies \dot{\chi} = \mathbf{f}(\chi)$$

O sistema em malha fechada obtido é não-linear. Para analisar a estabilidade, o primeiro método de Lyapunov diz que, linearizado o sistema, se suas raízes são estáveis, o sistema não-linear original é assintoticamente estável.

$$\delta \dot{\chi} = \left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} \delta \chi \quad \frac{\partial \mathbf{f}}{\partial \chi} = \begin{bmatrix} \frac{\partial f_1}{\partial e} & \frac{\partial f_1}{\partial \alpha} \\ \frac{\partial f_2}{\partial e} & \frac{\partial f_2}{\partial \alpha} \end{bmatrix} = \begin{bmatrix} -k_e \cos \alpha & k_e e \sin \alpha \\ 0 & k_e \cos \alpha - k_\alpha \end{bmatrix}$$

$$\left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} = \begin{bmatrix} -k_e & 0 \\ 0 & k_e - k_\alpha \end{bmatrix}$$

$$\begin{aligned} \text{poly} \left(\left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} \right) &= \det \left(s\mathbf{I} - \left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} \right) = \\ &= \det \left(\begin{bmatrix} s+k_e & 0 \\ 0 & s-k_e+k_\alpha \end{bmatrix} \right) = (s+k_e)(s-k_e+k_\alpha) \end{aligned}$$

Portanto

$$k_e > 0$$

$$k_\alpha > k_e$$

Desenvolvimento

Controle Linear

Controlador Proporcional Derivativo (PD):

$$\begin{cases} U(s) = \left(k_p^e + \frac{k_d^e cs}{s+c} \right) E(s) \\ \Omega(s) = \left(k_p^\alpha + \frac{k_d^\alpha cs}{s+c} \right) A(s) \end{cases}$$

$$\begin{bmatrix} \dot{e} \\ \dot{\alpha} \\ \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -[(k_p^e + ck_d^e)e - c^2k_d^ez_1] \cos \alpha \\ \left[\frac{(k_p^e + k_d^e c)e - c^2k_d^ez_1}{e} \right] \sin \alpha - [(k_p^\alpha + k_d^\alpha c)\alpha - c^2k_d^\alpha z_2] \\ e - cz_1 \\ \alpha - cz_2 \end{bmatrix}$$

$$\Rightarrow \dot{\chi} = \mathbf{f}(\chi)$$

Linearizando

$$\frac{\partial \mathbf{f}}{\partial \chi} = \begin{bmatrix} \frac{\partial f_1}{\partial e} & \frac{\partial f_1}{\partial \alpha} & \frac{\partial f_1}{\partial z_1} & \frac{\partial f_1}{\partial z_2} \\ \frac{\partial f_2}{\partial e} & \frac{\partial f_2}{\partial \alpha} & \frac{\partial f_2}{\partial z_1} & \frac{\partial f_2}{\partial z_2} \\ \frac{\partial f_3}{\partial e} & \frac{\partial f_3}{\partial \alpha} & \frac{\partial f_3}{\partial z_1} & \frac{\partial f_3}{\partial z_2} \\ \frac{\partial f_4}{\partial e} & \frac{\partial f_4}{\partial \alpha} & \frac{\partial f_4}{\partial z_1} & \frac{\partial f_4}{\partial z_2} \end{bmatrix} =$$

$$= \begin{bmatrix} -(k_p^e + ck_d^e) \cos \alpha & (k_p^e + ck_d^e)e \sin \alpha & c^2k_d^e \cos \alpha & 0 \\ \frac{c^2k_d^ez_1 \sin \alpha}{e^2} & \frac{u}{e} \cos \alpha - (k_p^\alpha + ck_d^\alpha) & -\frac{c^2k_d^e}{e} \sin \alpha & c^2k_d^\alpha \\ 1 & 0 & -c & 0 \\ 0 & 1 & 0 & -c \end{bmatrix}$$

A substituição do ponto $[0 \ 0]^T$ na matriz acima leva a termos indeterminados e infinitos.

Desenvolvimento

Controle Linear

Controlador Proporcional Derivativo (PD):

Aplicando o PD apenas ao erro de orientação

$$\begin{bmatrix} \dot{e} \\ \dot{\alpha} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -k_e e \cos \alpha \\ k_e \sin \alpha - [(k_p^\alpha + k_d^\alpha c)\alpha - c^2 k_d^\alpha z] \\ \alpha - cz \end{bmatrix}$$
$$\implies \dot{\chi} = \mathbf{f}(\chi)$$

Linearizando

$$\frac{\partial \mathbf{f}}{\partial \chi} = \begin{bmatrix} \frac{\partial f_1}{\partial e} & \frac{\partial f_1}{\partial \alpha} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial e} & \frac{\partial f_2}{\partial \alpha} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial e} & \frac{\partial f_3}{\partial \alpha} & \frac{\partial f_3}{\partial z} \end{bmatrix} =$$
$$= \begin{bmatrix} -k_e \cos \alpha & k_e e \sin \alpha & 0 \\ 0 & k_e \cos \alpha - (k_p^\alpha + c k_d^\alpha) & c^2 k_d^\alpha \\ 0 & 1 & -c \end{bmatrix}$$

$$\left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} = \begin{bmatrix} -k_e & 0 & 0 \\ 0 & k_e - k_p^\alpha - c k_d^\alpha & c^2 k_d^\alpha \\ 0 & 1 & -c \end{bmatrix}$$

$$\text{poly} \left(\left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} \right) = \det \left(s\mathbf{I} - s \left. \frac{\partial \mathbf{f}}{\partial \chi} \right|_{\chi=0} \right) =$$

$$= (s + k_e)[s^2 + (-k_e + k_p^\alpha + c k_d^\alpha + c)s + c(-k_e + k_p^\alpha)]$$

$$\begin{cases} k_e > 0 \\ k_p^\alpha > k_e \\ k_d^\alpha > -1 \end{cases}$$

Desenvolvimento

Controle Não-linear

Dada uma função candidata de Lyapunov, procura-se uma lei de controle que faça tal função ser positiva definida e com derivada no tempo negativa semi-definida, o que, pelo segundo método de Lyapunov, torna o sistema localmente assintoticamente estável.

Função candidata:
$$V(\chi) = \frac{1}{2}\lambda e^2 + \frac{1}{2}h\alpha^2$$

Lei de controle:
$$\begin{cases} u = \gamma e \cos \alpha \\ \omega = k\alpha + \gamma \cos \alpha \sin \alpha \end{cases}$$

Resultados

Requisitos

- As velocidades linear e angular máxima que podem ser desenvolvidas pelo robô são 1,2m/s e 300°/s, respectivamente;
- O robô para quando a raiz da soma dos quadrados dos erros é menor ou igual a 0,05.

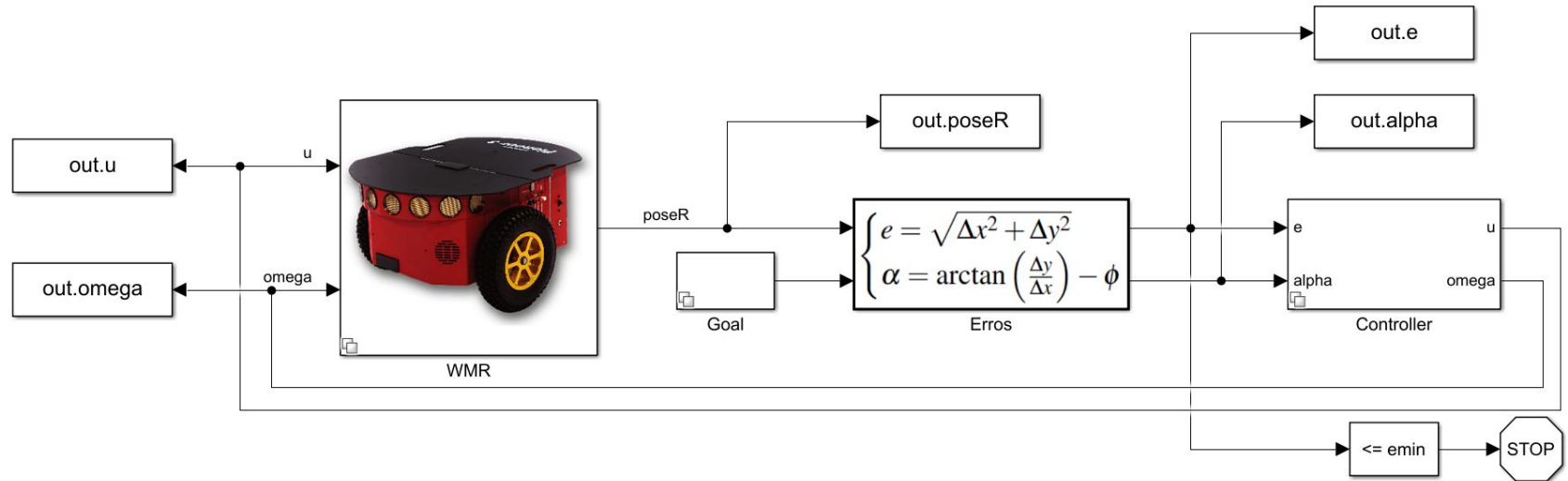
Cenário

O robô parte da posição (2, 2) com orientação inicial de 45° em direção ao alvo em (0, 0).

Resultados

Modelos

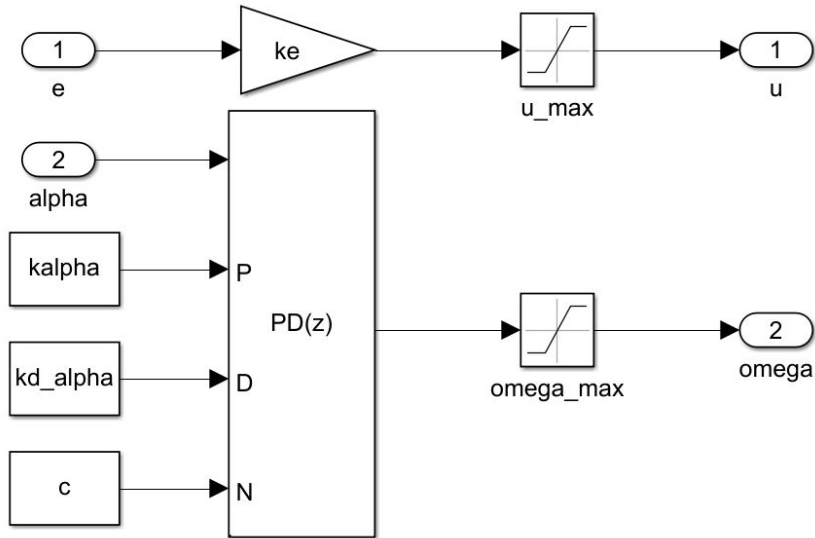
Sistema



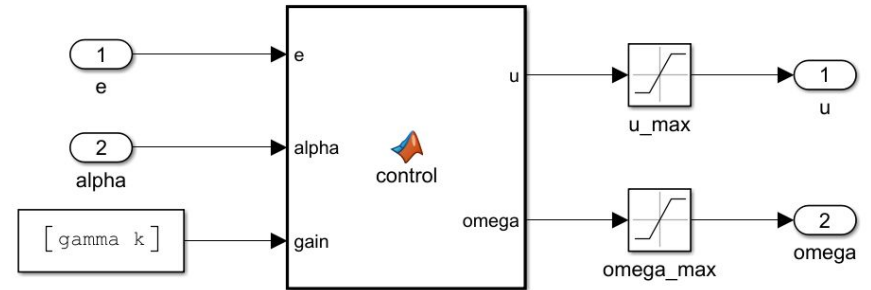
Resultados

Modelos

Controlador Linear

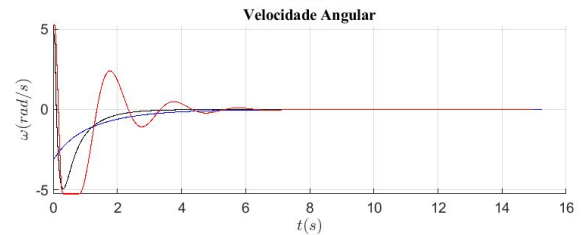
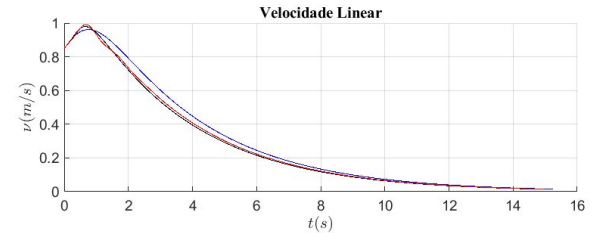
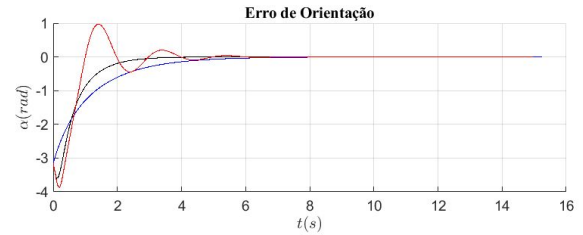
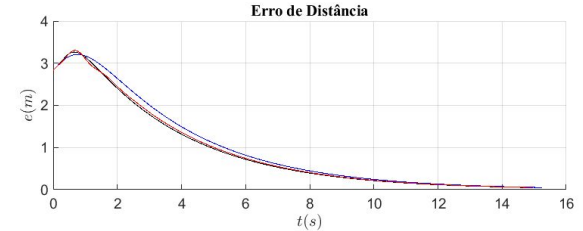
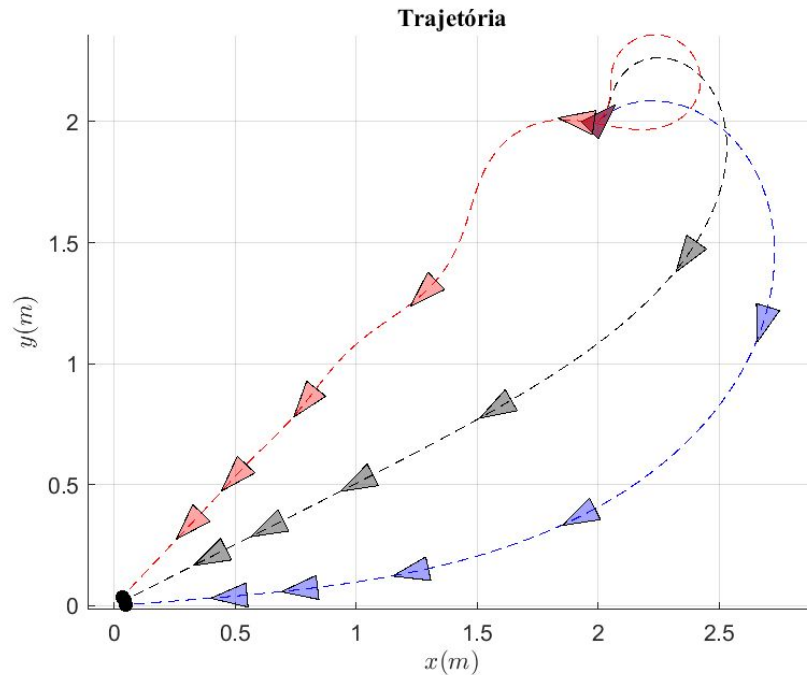


Controlador Não-linear



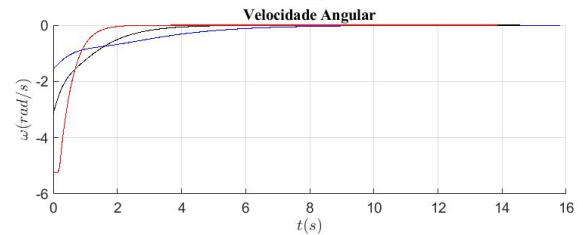
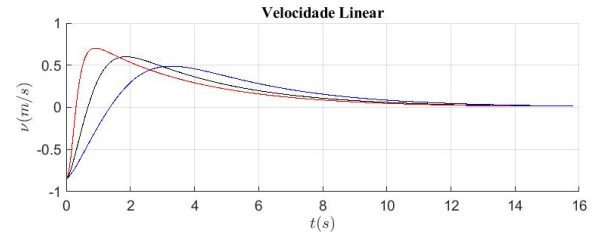
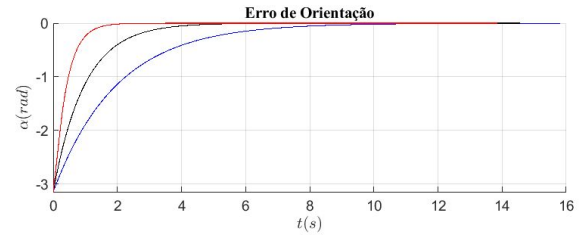
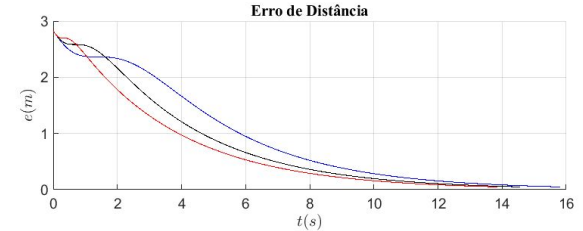
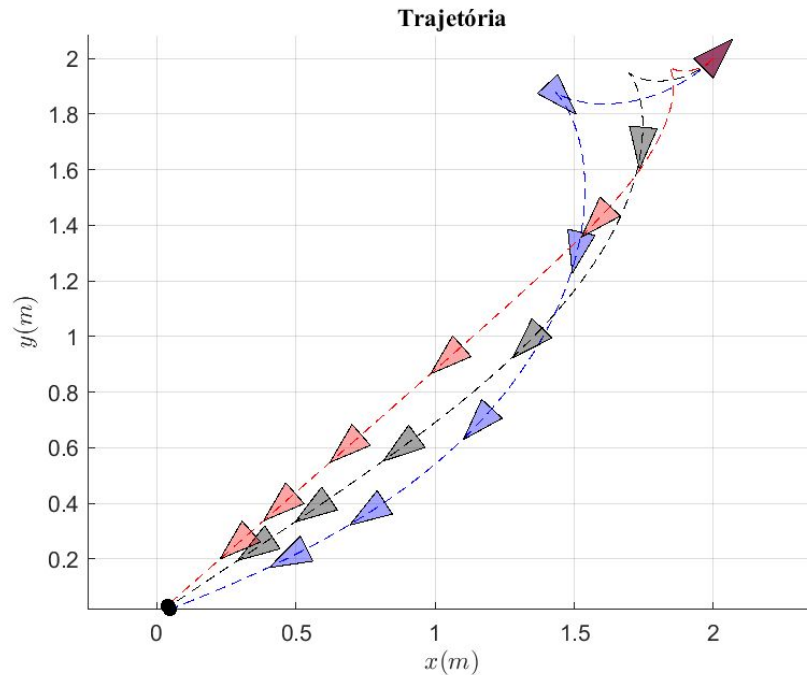
Resultados

Simulink: PD - K_p fixo, K_d entre 0,225 e 0,9



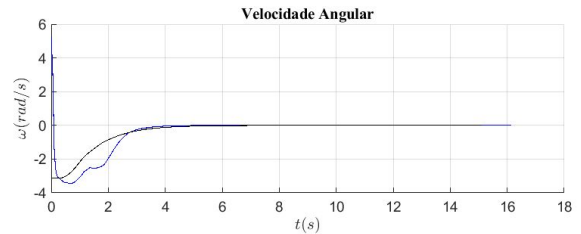
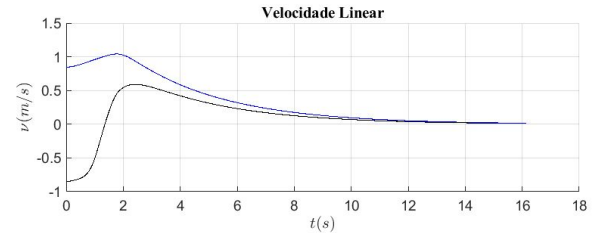
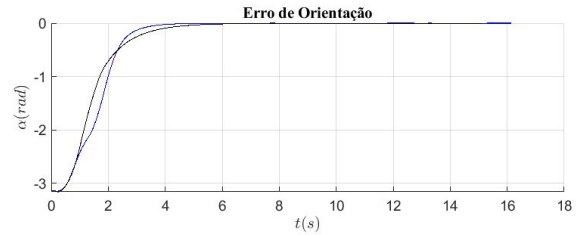
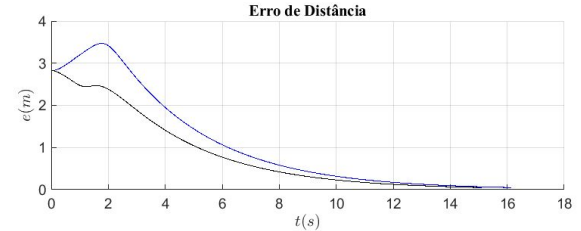
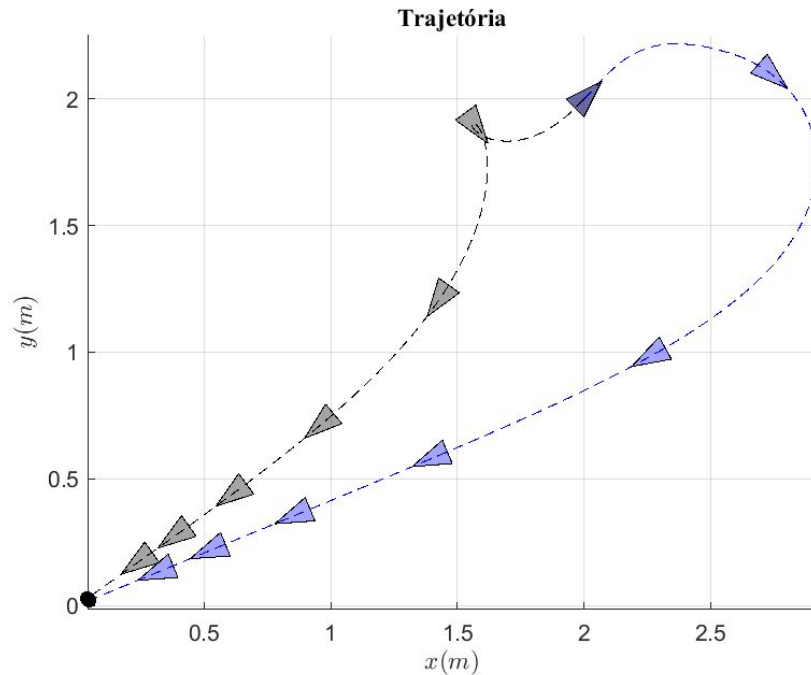
Resultados

Simulink: NL- γ fixo, k entre 0,5 e 2,5



Resultados

Coppelia: $k_e=0,3$; $k_\alpha=1$; $k_d=-0,5$; $\gamma=0,3$; $k=1$



Análise e Conclusão

Entre as leis de controle

- A adição do controle PD torna a resposta ao erro de orientação mais rápida quando em comparação com o controlador P;
- A lei de controle não-linear permite manobra de ré, o que não é possível com o controle linear;
- A lei de controle não-linear leva a movimentos mais suaves e naturais, com curvas menores;
- O controle não linear torna o sistema estável para qualquer ganho (contanto que positivo), diferente do controle linear.

Entre as simulações

- O modelo do robô considerado no Simulink é o modelo uniclo, que é puramente cinemático e abstrai até mesmo o tipo de robô móvel com rodas
- O CoppeliaSim simula a dinâmica de corpos rígidos do Pioneer3DX, considerando atritos, torques e as inércias de todas as partes móveis do robô
- A diferença dos modelos é o maior motivo para a diferença entre os resultados.

Análise e Conclusão

Conclui-se que o controle não-linear é preferível na solução do problema de estabilização de pose, e pode ser facilmente empregado no seguimento de caminho; como apresentado pelo artigo de Aicardi.

O uso de uma modelagem cinemática considerando o modelo uniciclo facilita o projeto de controle em todos os casos tratados neste trabalho, porém não permite observar o impacto da dinâmica de corpos rígidos específica do robô utilizado. Portanto, faz-se necessário estender a compreensão do problema para a dinâmica do robô, de forma a buscar soluções de controle mais robustas.

Obrigado!