Controle Preditivo Aplicado a um Robô com Tração Diferencial

Alison Tristão

June 22, 2025

Contents

1 Introdução

Este trabalho apresenta o desenvolvimento de um sistema de controle preditivo generalizado (GPC) aplicado a um robô com tração diferencial. O controlador é projetado com base na função de transferência do sistema, otimizando a atuação dos motores para minimizar o erro de rastreamento de trajetoria em um plano bidimensional (x, y).

2 Modelo Cinemático

O sistema modelado consiste em um robô com tração diferencial, que se movimenta em um plano bidimensional (x, y) e é controlado por dois motores de corrente contínua, conforme representado na Figura 1.

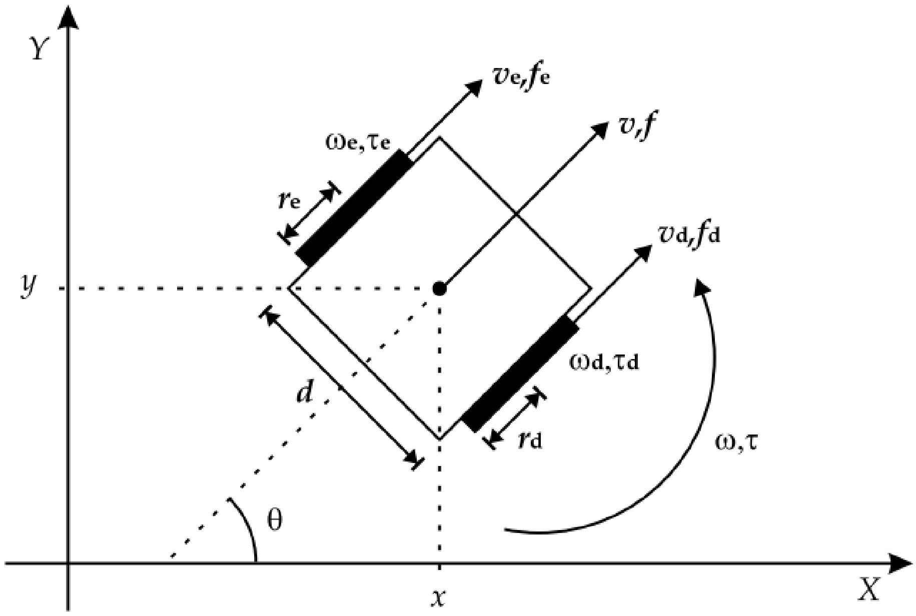

Figure 1: Modelo do robô com tração diferencial

A seguinte nomenclatura é utilizada:

- v, ω: velocidade linear e angular do robô;
- ω_d, ω_e: velocidade angular das rodas;
- v_d, v_e: velocidade linear das rodas;
- r_d, r_e: raio das rodas;
- d: distância entre as rodas;
- (x, y), θ: coordenadas e orientação do robô no plano;
- f, τ: força e torque aplicados sobre o robô;
- f_d, f_e: forças aplicadas nas rodas;
- τ_d, τ_e: torques aplicados nas rodas;

2.1 Relação entre Velocidades, Posição e Orientação

Em [2], o modelo cinemático é descrito por um sistema de equações que relaciona a velocidade linear e angular do robô com a velocidade angular das rodas. Para encontrar o modelo começamos definindo as velocidades tangenciais das rodas:

$$v_e = \omega_e r_e \qquad v_d = \omega_d r_d \tag{1}$$

A partir da velocidade linear e angular do robô, podem-se relacionar as velocidades tangenciais das rodas:

$$v_e = v - \frac{d}{2}\omega \qquad v_d = v + \frac{d}{2}\omega \tag{2}$$

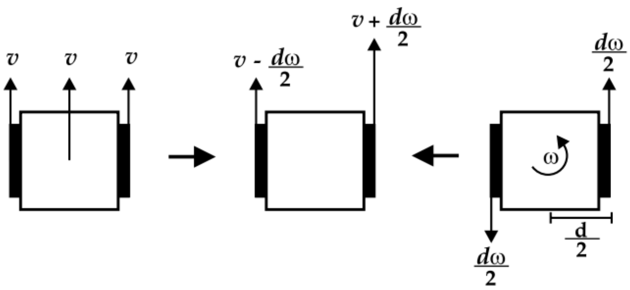

Figure 2: Composição do modelo cinemático

Ou, ao contrário, a partir das velocidades das rodas, podemos relacioná-las com a velocidade do robô:

$$v = \omega_d \frac{r_d}{2} + \omega_e \frac{r_e}{2} \qquad \omega = \omega_d \frac{r_d}{d} - \omega_e \frac{r_e}{d} \tag{3}$$

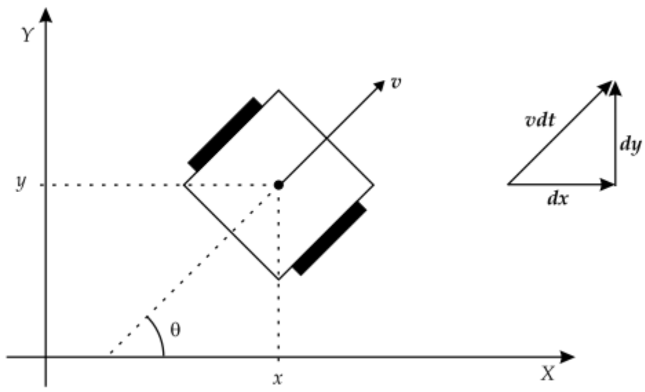

Figure 3: Coordenadas do robô no plano

Ao relacionar os movimentos cinemáticos a deslocamentos incrementais em um plano bidimensional, como na Figura 3, tem-se que a velocidade do robô pode ser expressa como:

$$\begin{cases} \dot{x} = v\cos(\theta) \\ \dot{y} = v\sin(\theta) \\ \dot{\theta} = \omega \end{cases} \tag{4}$$

3 Modelo Dinâmico

Para encontrar o modelo dinâmico resultante entre a tensão aplicada nos motores e as velocidades resultantes sobre o robô, precisamos considerar duas dinâmicas:

A relação entre a tensão aplicada nos motores e o torque resultante sobre as rodas, em conjunto com as resultantes que esses torques aplicados geram sobre o robô, considerando sua massa, momento de inércia e coeficientes de viscosidade.

3.1 Dinâmica dos Atuadores

A dinâmica dos atuadores relaciona a tensão aplicada aos motores com o torque e a velocidade angular nas rodas, resultantes das características do circuito de armadura.

Conforme descrito em [1], as equações diferenciais que representam o circuito de armadura e o equilíbrio de torque em um motor CC são:

$$u = L\frac{di}{dt} + Ri + K_\omega \omega \tag{5}$$

$$t_r = K_t i = J\frac{d\omega}{dt} + \beta\omega \tag{6}$$

A primeira equação descreve o comportamento elétrico do motor, em que a tensão aplicada u se divide entre a indutância L da armadura, a resistência R, e a força contra-eletromotriz (CEMF) $K_\omega \omega$ gerada pela rotação do eixo. Já a segunda equação representa o balanço de torques no rotor, onde o torque gerado $K_t i$ é utilizado para vencer a inércia do motor J e as perdas por atrito viscoso β.

Relacionando as duas equações acima, podemos eliminar a corrente elétrica i e obter uma equação diferencial de segunda ordem que relaciona diretamente a tensão aplicada u à velocidade angular ω:

$$LJ\frac{d^2\omega}{dt^2} + (L\beta + RJ)\frac{d\omega}{dt} + (R\beta + K_\omega K_t)\omega = K_t u \tag{7}$$

Considerando que L é desprezível, podemos simplificar a equação para:

$$RJ\frac{d\omega}{dt} + (R\beta + K_\omega K_t)\omega = K_t u \tag{8}$$

Que, finalmente, pode ser reescrita como:

$$t_r = \rho K_t u - \beta\omega - J\frac{d\omega}{dt} \tag{9}$$

Onde a nomenclatura utilizada é:

- u, i: tensão aplicada e corrente no circuito de armadura;
- L, R, ρ: indutância, resistência e o inverso da resistência do motor;
- K_ω, ω: constante de velocidade angular e velocidade angular;
- K_t, t_r: constante de torque e torque resultante;
- J, β: momento de inércia e coeficiente de viscosidade do motor.

3.2 Relação de Esforços nos Referenciais

As forças aplicadas sobre as rodas devido ao torque aplicado nos motores, mostradas na Figura 2, resultam nas velocidades lineares e angulares do robô.
Em [2], essas forças são definidas como:

$$f_d = \frac{t_d}{r_d} \qquad\qquad f_e = \frac{t_e}{r_e} \tag{10}$$

Assim, resultando nos esforços totais aplicados sobre a estrutura do robô:

$$f = f_d + f_e = \frac{t_d}{r_d} + \frac{t_e}{r_e} \qquad\qquad t = t_d\frac{d}{2r_d} - t_e\frac{d}{2r_e} \tag{11}$$

Para encontrar esses esforços, precisamos utilizar a equação 9, que define a relação entre os torques aplicados nos motores e as tensões u_d e u_e.
Ao escrever a equação dos motores em forma matricial, temos:

$$\begin{bmatrix} t_d \\ t_e \end{bmatrix} = \begin{bmatrix} \rho_d K_{td} & 0 \\ 0 & \rho_e K_{te} \end{bmatrix} \begin{bmatrix} u_d \\ u_e \end{bmatrix} - \begin{bmatrix} \beta_d & 0 \\ 0 & \beta_e \end{bmatrix} \begin{bmatrix} \omega_d \\ \omega_e \end{bmatrix} - \begin{bmatrix} J_d & 0 \\ 0 & J_e \end{bmatrix} \begin{bmatrix} \dot{\omega}_d \\ \dot{\omega}_e \end{bmatrix} \tag{12}$$

Utilizando a equação 2, podemos reescrever a relação entre os torques aplicados, eliminando as velocidades angulares das rodas ω_d e ω_e e substituindo-as pelas velocidades lineares v e ω do robô:

$$\begin{bmatrix} \omega_d \\ \omega_e \end{bmatrix} = \begin{bmatrix} \frac{1}{r_d} & \frac{d}{2r_d} \\ \frac{1}{r_e} & -\frac{d}{2r_e} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{13}$$

Resumindo, em forma matricial, a relação entre os torques aplicados sobre o robô e suas velocidades é dada por:

$$\mathbf{t} = \mathbf{K}_{mot}\mathbf{u} - \mathbf{B}_{mot}{}^{\omega}\mathbf{T}_{\mathbf{v}}\mathbf{v} - \mathbf{J}_{mot}{}^{\omega}\mathbf{T}_{\mathbf{v}}\dot{\mathbf{v}} \tag{14}$$

Finalmente, ao aplicarmos a matriz de transformação ${}^{\omega}\mathbf{T}_{\mathbf{v}}{}^{T}$, que projeta os torques das rodas nos eixos do referencial do robô, obtendo a equação que relaciona os esforços totais resultantes $\mathbf{f}$ com os torques aplicados $\mathbf{t}$, encontrando a relação que define as velocidades do robô aos esforços aplicados sobre sua estrutura.

$$\mathbf{f} = {}^{\omega}\mathbf{T}_{\mathbf{v}}{}^{T}\mathbf{t} \tag{15}$$

Onde:

$$\mathbf{t} = \begin{bmatrix} t_d \\ t_e \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} u_d \\ u_e \end{bmatrix} \qquad \mathbf{v} = \begin{bmatrix} v \\ \omega \end{bmatrix} \qquad \mathbf{f} = \begin{bmatrix} f \\ t \end{bmatrix} \qquad {}^{\omega}\mathbf{T}_{\mathbf{v}} = \begin{bmatrix} \frac{1}{r_d} & \frac{d}{2r_d} \\ \frac{1}{r_e} & -\frac{d}{2r_e} \end{bmatrix}$$

$$\mathbf{K}_{mot} = \begin{bmatrix} \rho_d K_{td} & 0 \\ 0 & \rho_e K_{te} \end{bmatrix} \qquad \mathbf{B}_{mot} = \begin{bmatrix} \beta_d & 0 \\ 0 & \beta_e \end{bmatrix} \qquad \mathbf{J}_{mot} = \begin{bmatrix} J_d & 0 \\ 0 & J_e \end{bmatrix} \qquad {}^{\omega}\mathbf{T}_{\mathbf{v}}{}^{T} = \begin{bmatrix} \frac{1}{r_d} & \frac{1}{r_e} \\ \frac{d}{2r_d} & -\frac{d}{2r_e} \end{bmatrix}$$

3.3 Dinâmica do Robô

Para encontrar a dinâmica do robô, consideramos as leis de Newton e Euler, relacionando a força e o torque aplicados sobre o robô com as acelerações linear e angular resultantes. Assim como em [2], temos:

$$f = m\dot{v} + \beta_l v \qquad\qquad t = J\dot{\omega} + \beta_\theta \omega \tag{16}$$

Onde m é a massa do robô, J é o momento de inércia, β_l e β_θ são os coeficientes de viscosidade linear e angular, respectivamente.
Substituindo as equações de esforço e torque definidas na equação 11, temos:

$$m\dot{v} + \beta_l v = \frac{t_d}{r_d} + \frac{t_e}{r_e} \qquad\qquad J\dot{\omega} + \beta_\theta \omega = t_d \frac{d}{2r_d} - t_e \frac{d}{2r_e} \tag{17}$$

Ou em forma matricial:

$$\begin{bmatrix} m & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \beta_l & 0 \\ 0 & \beta_\theta \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{r_d} & \frac{1}{r_e} \\ \frac{d}{2r_d} & -\frac{d}{2r_e} \end{bmatrix} \begin{bmatrix} t_d \\ t_e \end{bmatrix} \tag{18}$$

Utilizando a equação 14, podemos reescrever a dinâmica do robô em termos da tensão aplicada nos motores:

$$\mathbf{M}_{robo}\dot{\mathbf{v}} + \mathbf{B}_{robo}\mathbf{v} = {}^{\omega}\mathbf{T_v}^T(\mathbf{K}_{mot}\mathbf{u} - \mathbf{B}_{mot}{}^{\omega}\mathbf{T_v}\mathbf{v} - \mathbf{J}_{mot}{}^{\omega}\mathbf{T_v}\dot{\mathbf{v}}) \tag{19}$$

Deixando $\mathbf{v}$ em função de $\mathbf{u}$, temos:

$$(\mathbf{M}_{robo} + {}^{\omega}\mathbf{T_v}^T\mathbf{J}_{mot}{}^{\omega}\mathbf{T_v})\dot{\mathbf{v}} + (\mathbf{B}_{robo} + {}^{\omega}\mathbf{T_v}^T\mathbf{B}_{mot}{}^{\omega}\mathbf{T_v})\mathbf{v} = {}^{\omega}\mathbf{T_v}^T\mathbf{K}_{mot}\mathbf{u} \tag{20}$$

Onde $\mathbf{M}$ é a matriz de massa e inércia do robô e $\mathbf{B}$ é a matriz de viscosidade, representadas por:

$$\mathbf{M}_{robo} = \begin{bmatrix} m & 0 \\ 0 & J \end{bmatrix} \qquad \mathbf{B}_{robo} = \begin{bmatrix} \beta_l & 0 \\ 0 & \beta_\theta \end{bmatrix}$$

Por fim, a dinâmica do robô pode ser expressa como:

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{B}\mathbf{v} = \mathbf{K}\mathbf{u} \tag{21}$$

Onde:

$$\mathbf{M} = \mathbf{M}_{robo} + {}^{\omega}\mathbf{T_v}^T\mathbf{J}_{mot}{}^{\omega}\mathbf{T_v} \qquad \mathbf{B} = \mathbf{B}_{robo} + {}^{\omega}\mathbf{T_v}^T\mathbf{B}_{mot}{}^{\omega}\mathbf{T_v} \qquad \mathbf{K} = {}^{\omega}\mathbf{T_v}^T\mathbf{K}_{mot}$$

O sistema apresenta um comportamento dinâmico linear de primeira ordem. Assumindo que $\mathbf{M}$ e $\mathbf{B}$ são matrizes invertíveis e constantes, a equação pode ser reescrita em espações de estados como:

$$\begin{cases} \dot{\mathbf{v}} = -\mathbf{M}^{-1}\mathbf{B}\mathbf{v} + \mathbf{M}^{-1}\mathbf{K}\mathbf{u} \\ \dot{\mathbf{q}} = {}^{v}\mathbf{T}_{(\mathbf{x},\mathbf{y})}\mathbf{v} \end{cases} \tag{22}$$

Onde $\mathbf{q}$ é o vetor de estados do robô.

$${}^{v}\mathbf{T}_{(\mathbf{x},\mathbf{y})} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{q} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \qquad \dot{\mathbf{q}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

4 Linearização do Modelo

A implementação de controle preditivo em hardware embarcado exige soluções computacionalmente menos custosas, devido ao baixo poder de processamento dos microcontroladores. Isso favorece a utilização de modelos lineares, dada a simplicidade na minimização de funções quadráticas.

Para reduzir o custo computacional do controle, utilizaremos a linearização por meio de séries de Taylor em torno de ω_0 e v_0.

Assumiremos que o centro do referencial está localizado no centro geométrico do robô, e que sua orientação permanece fixa em 0 graus em relação ao eixo x.

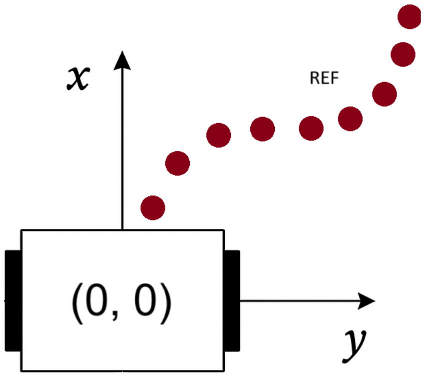

Figure 4: Referencial do robô

Utilizando a aproximação por séries de Taylor:

$$x = x_0 + \left.\frac{\partial x}{\partial v}\right|_{v_0} \Delta v + \left.\frac{\partial x}{\partial \omega}\right|_{\omega_0} \Delta\theta$$

$$y = y_0 + \left.\frac{\partial y}{\partial v}\right|_{v_0} \Delta v + \left.\frac{\partial y}{\partial \omega}\right|_{\omega_0} \Delta\theta$$

Assim, o modelo cinemático definido na Equação 4 pode ser representado em (x, y) como:

$$\begin{cases} x \approx v_0 \cos(\omega_0) + \cos(\omega_0)\Delta v - v_0 \sin(\omega_0)\Delta\theta \\ y \approx v_0 \sin(\omega_0) + \sin(\omega_0)\Delta v + v_0 \cos(\omega_0)\Delta\theta \end{cases} \tag{23}$$

Considerando os valores numéricos com $\omega_0 = 0$, temos:

$$\begin{cases} \dot{x} \approx v \\ \dot{y} \approx v_0\theta \\ \dot{\theta} = \omega \end{cases} \quad \Rightarrow \quad \begin{cases} x \approx s \\ y \approx s_0\theta \end{cases} \tag{24}$$

É possível observar que ao linearizar o modelo, acontece um desacoplamento entre as variáveis de estado, resultando em em uma relação direta entre a velocidade linear v e a posição x, e entre a velocidade angular ω e a posição y. Simplificando significativamente a relação entre a posição $\mathbf{q}$ e as entradas de controle $\mathbf{u}$.

5 Controle Preditivo Generalizado (GPC)

O Controle Preditivo Generalizado (GPC) é uma técnica avançada de controle preditivo (MPC) que adota a função de transferência do sistema como modelo. No qual, nesse trabalho a função de transferência é proveniente da equação descrita em 21.

Nota-se que o sistema descrito possui um comportamento dinâmico linear de primeira ordem, onde é possivel encontrar as matrizes que contêm as constantes de tempo e ganhos estáticos do sistema:

$$\boldsymbol{\tau} = \mathbf{B}^{-1}\mathbf{M} \qquad \mathbf{K}_u = \mathbf{B}^{-1}\mathbf{K} \tag{25}$$

Permitindo que encontremos a função de transferência das variáveis de controle para cada uma das variáveis de saída do sistema. Que ao utilizarmos u_d, u_e, v e ω, no domínio discreto, usando a transformada z, elas são dadas por:

$$\mathbf{H}_{dv}(z^{-1}) = \frac{b_{dv}z^{-1}}{1 - a_d z^{-1}} \qquad \mathbf{H}_{d\omega}(z^{-1}) = \frac{b_{d\omega}z^{-1}}{1 - a_d z^{-1}}$$

$$\mathbf{H}_{ev}(z^{-1}) = \frac{b_{ev}z^{-1}}{1 - a_e z^{-1}} \qquad \mathbf{H}_{e\omega}(z^{-1}) = \frac{b_{e\omega}z^{-1}}{1 - a_e z^{-1}}$$

Onde os parâmetros b e a são obtidos a partir da discretização da constante de tempo e ganho estático extraídos das matrizes $\mathbf{K}$ e $\boldsymbol{\tau}$.

Integrando os valores das velocidades para obter as posições (sendo T o período de amostragem) e utilizando os incrementos de controle Δu, conforme a implementação típica do GPC, obtêm-se as funções de transferência discretas para cada variável de controle e saída do sistema $\frac{Y(z)}{\Delta U(z)}$.

$$\mathbf{H}_{ds}(z^{-1}) = \frac{b_{dv}z^{-1}}{1 - a_d z^{-1}} \frac{T}{(1 - z^{-1})^2} \qquad \mathbf{H}_{d\theta}(z^{-1}) = \frac{b_{d\theta}z^{-1}}{1 - a_d z^{-1}} \frac{T}{(1 - z^{-1})^2}$$

$$\mathbf{H}_{es}(z^{-1}) = \frac{b_{ev}z^{-1}}{1 - a_e z^{-1}} \frac{T}{(1 - z^{-1})^2} \qquad \mathbf{H}_{e\theta}(z^{-1}) = \frac{b_{e\theta}z^{-1}}{1 - a_e z^{-1}} \frac{T}{(1 - z^{-1})^2}$$

Por fim, a partir das funções de transferência do sistema, podemos definir o modelo que será utilizado no GPC. Nesse modelo, a matriz G representa a resposta ao degrau para cada variável de controle, enquanto F é a matriz de polinômios responsável pelo cálculo da resposta livre a partir de $y[k]$.
Assim podemos definir a predição futura como:

$$\hat{\mathbf{y}} = \mathbf{G}\Delta\mathbf{u} + \mathbf{F}(z^{-1})\mathbf{y}[k] \tag{26}$$

Resultando na função de custo do GPC, em que $\mathbf{w}$ representa o valor de referência, e $\boldsymbol{\delta}$ e $\boldsymbol{\lambda}$ são os pesos associados, respectivamente, à ação de controle e ao erro de predição."

$$J_u = \boldsymbol{\delta}(\hat{\mathbf{y}} - \mathbf{w})^2 + \boldsymbol{\lambda}\boldsymbol{\Delta}\mathbf{u}^2 \tag{27}$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_{ds} & \mathbf{G}_{es} \\ \mathbf{G}_{d\theta} & \mathbf{G}_{e\theta} \end{bmatrix} \quad \boldsymbol{\Delta}\mathbf{u} = \begin{bmatrix} \boldsymbol{\Delta}\mathbf{u_d} \\ \boldsymbol{\Delta}\mathbf{u_e} \end{bmatrix} \quad \mathbf{F}(z^{-1}) = \begin{bmatrix} \mathbf{F}_s(z^{-1}) & 0 \\ 0 & \mathbf{F}_\theta(z^{-1}) \end{bmatrix} \quad \mathbf{y}[k] = \begin{bmatrix} s[k] \\ \theta[k] \end{bmatrix}$$

Utilizando o modelo linearizado descrito na equação 24, podemos relacionar s e θ diretamente com (x, y).

6 Implementação

A implementação da teoria descrita foi realizada em ambiente simulado, utilizando python e a biblioteca pygame para interface gráfica e simulação. Ademais, foi utilizando GNU octave para extrair os coeficientes da resposta ao degrau e matriz de polinômios F.

O link para o repositório contendo o código-fonte se encontra disponível no GitHub, `https://github.com/alison-tristao/line_follower_simulator`.

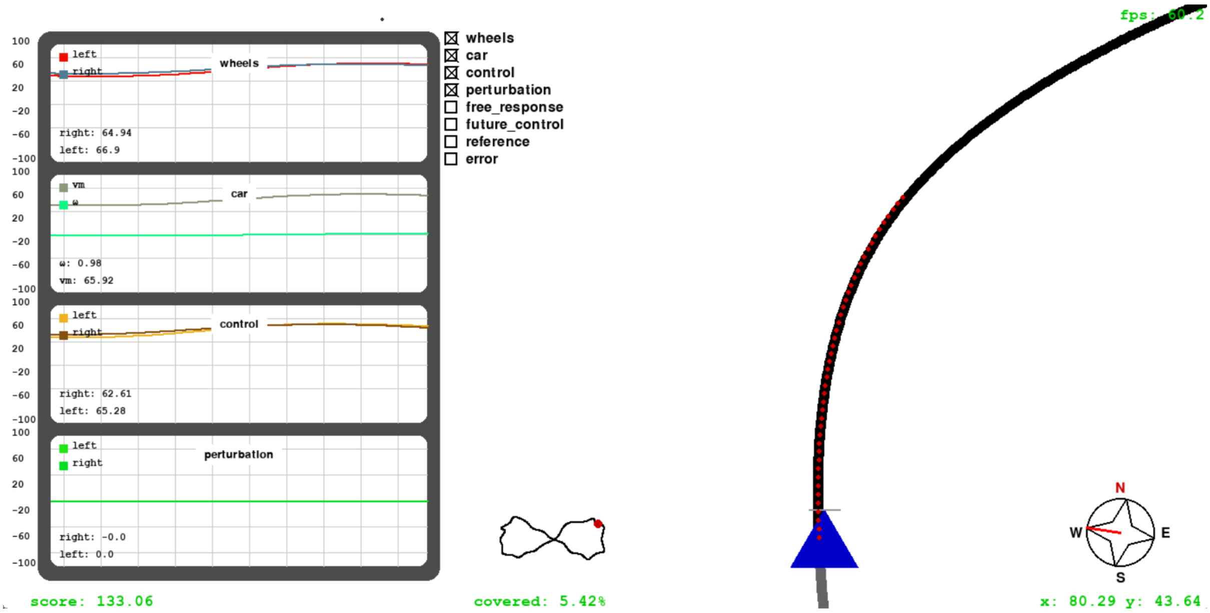

Figure 5: Interface do simulador

Utilizando valores normalizados entre 0 a 100% para $\mathbf{u}$, a implementação do GPC considerou os seguintes parâmetros:

- Período de amostragem $T = 0.0125$s (80fps);
- Horizonte de predição $N_{ss} = 5\tau$ do motor mais lento;
- Horizonte de controle $M = 5$;
- Peso da ação de controle $\boldsymbol{\delta_e}$ e $\boldsymbol{\delta_d} = 0.001$;
- Peso do erro $\boldsymbol{\lambda_s} = 1$ e $\boldsymbol{\lambda_\theta} = 0.5$;

Os parâmetros do robô foram considerando:

- Raio das rodas $r = 0.04$m;
- Distância entre as rodas $L = 0.15$m;
- Tempo de acomodação $\tau_e = 0.62$s e $\tau_d = 0.58$s;
- Ganho estático para v: $K_{ev} = 0.042$ e $K_{dv} = 0.042$;
- Ganho estático para ω: $K_{e\omega} = 0.55$ e $K_{d\omega} = 0.55$;

7 Conclusão

A implementação do controle preditivo generalizado (GPC) para um robô com tração diferencial demonstrou ser uma abordagem eficaz. Através da linearização do modelo dinâmico e minimização da função custo em caso irrestrito, foi possível obter uma ação de controle sem esforço computacional significativo, demonstrando a viabilidade da aplicação para sistemas embarcados.

Além disso, a implementação em ambiente simulado permitiu uma validação preliminar do desempenho do controlador, com resultados satisfatórios na trajetória do robô.

References

[1] K. Ogata. *Modern Control Engineering*. Prentice Hall, 3rd edition, 1997.

[2] F. C. Vieira. Controle dinâmico de robôs móveis com acionamento diferencial. Dissertação de mestrado, Universidade Federal do Rio Grande do Norte, Natal–RN, Brasil, Feb. 2005. Disponível em PDF.