

# Localização Probabilística de Robôs com Redundância de Faróis

João Santos, 76912<sup>1</sup>

<sup>1</sup>Departamento de Eletrónica, Telecomunicações e Informática, Universidade de Aveiro

Maio 2022

## 1 Introdução

Para navegar eficazmente e atingir objetivos, os robôs móveis devem ser capazes de determinar a sua posição em relação a locais conhecidos no ambiente. Esta é a problemática da localização.

O *dead-reckoning* é o método mais básico para a tentar realizar[1]. Este método utiliza o histórico de velocidades do robô ao longo do tempo para calcular a mudança de posição a partir do ponto de partida. Infelizmente, os métodos de cálculo de *dead-reckoning* puro são propensos a erros que crescem continuamente ao longo do tempo, exigindo a utilização de um segundo método de correção numa base regular. Técnicas adicionais de localização, tais como triangulação de pontos de referência ou correspondência de mapas, são frequentemente combinadas com o *dead-reckoning*, utilizando um filtro Kalman estendido para atualizar probabilisticamente a posição do robô.

## 2 Metodologia

Num ambiente 2D composto por vários faróis com posições conhecidas, tem-se como objetivo que o robô móvel visite, por ordem, todos os faróis, parando no último, percorrendo uma trajetória suave tentando sempre manter uma estimativa da sua pose atual.

Paralelamente, existe a necessidade de verificar que as velocidades a impor ao robô são fisicamente possíveis, recorrendo à cinemática inversa de três modelos móveis.

Este projeto recorreu intensivamente à biblioteca *SymPy*[2] para o cálculo simbólico. Várias equações intermédias foram omitidas.

### 2.1 Filtro de Kalman Estendido

Em 1960, Rudolf Kálmán publicou um artigo[3] em que descrevia uma solução recursiva ao problema da filtragem linear de dados discretos. O filtro de Kalman é um conjunto de equações matemáticas que proporcionam uma computação eficiente para estimar o estado de um processo, de forma a minimizar o erro quadrático médio.

Para casos em que o processo a ser estimado é não linear é frequentemente utilizado o filtro de Kalman estendido (EKF, do inglês) que lineariza em torno de média e da covariância.

Numa primeira instância, com base no estado atual do sistema  $\hat{x}_k$ , num dado controlo  $u_k$  e no ruído associado  $w_k$ , este filtro tenta prever o estado futuro  $\bar{x}_{k+1}$  e a respetiva covariância  $\bar{P}_{k+1}$  (eq. 1).

$$\begin{cases} \bar{x}_{k+1} = f(\hat{x}_k, u_k, w_k) \\ \bar{P}_{k+1} = J_{f_x}(\hat{x}_k, u_k, w_k)P_k J_{f_x}^T(\hat{x}_k, u_k, w_k) + \\ + J_{f_w}(\hat{x}_k, u_k, w_k)Q_k J_{f_w}^T(\hat{x}_k, u_k, w_k) \end{cases} \quad (1)$$

De seguida, a inovação (i.e. é diferença entre as observações obtidos e as esperadas, para um dado estado) são usados em conjunto com o ganho de Kalman  $K$  para produzir uma melhor estimativa do estado atual do sistema  $\hat{x}_{k+1}$ . A covariância do processo  $P_{k+1}$  é também atualizada com recurso ao ga-

nho de Kalman e à covariância da inovação  $S$  (eq. 2).

$$\begin{cases} \hat{x}_{k+1} &= \bar{x}_{k+1} + K(z_{k+1} - h(\bar{x}_{k+1})) \\ P_{k+1} &= \bar{P}_{k+1} - KSK^T \\ S &= J_h(\bar{x}_{k+1})\bar{P}_{k+1}J_h^T(\bar{x}_{k+1}) + R \\ K &= \bar{P}_{k+1}J_h^T(\bar{x}_{k+1})S^{-1} \end{cases} \quad (2)$$

Para ser possível prever o estado futuro do sistema é necessária a existência de uma função de transição de estado  $f$ . No caso de um robô móvel, a eq. 3 define o modelo cinemático genérico do mesmo, onde ao controlo de velocidade linear  $V_k$  e angular  $\omega_k$  são adicionados os respetivos ruídos gaussianos  $\delta V_k$  e  $\delta \omega_k$ . Na etapa de previsão é convencional definir-se que  $w_k = 0$ . Como é também assumido que o intervalo de tempo  $\Delta t$  é definido e constante, este é omitido nos parâmetros do modelo.

$$\begin{aligned} f(\hat{x}_k, u_k, w_k) &= \begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} \\ &= \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} \cos(\theta_k) & 0 \\ \sin(\theta_k) & 0 \\ 0 & 1 \end{bmatrix} \left( \begin{bmatrix} V_k \\ \omega_k \end{bmatrix} + \begin{bmatrix} \delta V_k \\ \delta \omega_k \end{bmatrix} \right) \Delta t \end{aligned} \quad (3)$$

Para a previsão da covariância são ainda necessários os Jacobianos da função  $f$  relativamente ao vetor de estado  $J_{f_x}$  e ao vetor de ruídos  $J_{f_w}$ , que estão definidos, respetivamente, nas equações 4 e 5.

$$J_{f_x}(\hat{x}_k, u_k, w_k) = \begin{bmatrix} 1 & 0 & -\Delta t (v_k + \delta V_k) \sin(\theta_k) \\ 0 & 1 & \Delta t (v_k + \delta V_k) \cos(\theta_k) \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$J_{f_w}(\hat{x}_k, u_k) = \begin{bmatrix} \Delta t \cos(\theta_k) & 0 \\ \Delta t \sin(\theta_k) & 0 \\ 0 & \Delta t \end{bmatrix} \quad (5)$$

Já na etapa de atualização, um dos requisitos do filtro de Kalman é que seja possível computacional

as medições esperados dos sensores para um determinado estado. No caso particular em que os sensores detetam a distância  $r^i$  e orientação  $\phi^i$  relativas a cada farol  $i$ , o modelo de observação  $h$  é definido na eq. 6. Como a posição  $[x_L^i, y_L^i]$  de cada farol deverá ser previamente conhecida, esta informação é omitida dos argumentos da função, bem como o ruído associado a estas observações, que se define ser nulo.

$$\begin{aligned} h(\bar{x}_{k+1}) &= \begin{bmatrix} r_{k+1}^i \\ \phi_{k+1}^i \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{(x_L^i - x_{k+1})^2 + (y_L^i - y_{k+1})^2} \\ \arctan(\frac{y_L^i - y_{k+1}}{x_L^i - x_{k+1}}) - \theta_{k+1} \end{bmatrix} \end{aligned} \quad (6)$$

Similarmente à etapa de previsão, o Jacobiano do modelo de observação  $J_h$  é também necessário para que seja realizada a atualização da covariância do processo. Este Jacobiano encontra-se definido na eq. 7. Note-se que a dimensão deste Jacobiano é variável e da forma  $2i \times 3$  onde  $i$  representa o número total de faróis observáveis.

$$\begin{aligned} A &= \frac{-x_L + x_{k+1}}{\sqrt{(x_L - x_{k+1})^2 + (y_L - y_{k+1})^2}} \\ B &= \frac{-y_L + y_{k+1}}{\sqrt{(x_L - x_{k+1})^2 + (y_L - y_{k+1})^2}} \\ C &= \frac{y_L - y_{k+1}}{(x_L - x_{k+1})^2 + (y_L - y_{k+1})^2} \\ D &= \frac{-x_L + x_{k+1}}{(x_L - x_{k+1})^2 + (y_L - y_{k+1})^2} \end{aligned}$$

$$J_h(\bar{x}_{k+1}) = \begin{bmatrix} A & B & 0 \\ C & D & -1 \end{bmatrix} \quad (7)$$

Resta então definir as matrizes (diagonais) de covariância dos ruídos do processo  $Q$  e das observações  $R$ , respetivamente, pelas equações 8 e 9.

$$Q = \text{diag}([\sigma_{V_k}^2, \sigma_{\omega_k}^2]) \quad (8)$$

$$R = \text{diag}([\sigma_{r^1}^2, \sigma_{\phi^1}^2, \dots, \sigma_{r^i}^2, \sigma_{\phi^i}^2]) \quad (9)$$

## 2.2 Cinemática

A cinemática de um robô móvel é o conjunto de equações (modelo) que permite converter um dado *input* de velocidade ou posição angulares das rodas para as correspondentes velocidades linear e angular relativamente a um dado referencial, e vice-versa. Estes modelos podem por isso ser, respetivamente, denominados de cinemática direta ou cinemática inversa.

De uma forma genérica, a cinemática direta pode ser representado como  $[V_x, V_y, \omega]^T = Rot M \Omega^T$ , onde  $Rot$  representa a matriz de rotação do referencial local do robô para o referencial global (eq. 10),  $M$  é a matriz cinemática característica de uma dada configuração e  $\Omega$  é a discretização do estado das rodas do robô.

$$Rot(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

Nos modelos que adiante se seguem, considera-se que  $r$  e  $L$  são valores constantes, conhecidos e que definem, respetivamente, o raio das rodas do robô e a distância que relaciona a posição relativa das rodas.

### 2.2.1 Robô com Tração Diferencial

Num robô com tração diferencial, duas rodas motrizes ( $\omega_L$  e  $\omega_R$ ) alinhadas no mesmo eixo de rotação providenciam tanto o posicionamento linear como angular. Assim sendo, a eq. 11 define a cinemática direta, em termos de velocidades, deste robô.

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = Rot(\theta(t)) \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ 0 & 0 \\ -\frac{r}{L} & \frac{r}{L} \end{bmatrix} \begin{bmatrix} \omega_L(t) \\ \omega_R(t) \end{bmatrix} \quad (11)$$

Por sua vez, a cinemática inversa obtém-se por integração, i.e.,

$$[x(t), y(t), \theta(t)]^T = \int [\dot{x}(t), \dot{y}(t), \dot{\theta}(t)]^T \delta t.$$

Ora, para que estas integrações tenham solução, dois pressupostos têm de ser impostos: i) num dado intervalo de tempo  $\Delta t$ ,  $\omega_L$  e  $\omega_R$  são constantes e ii) a

velocidade angular  $\dot{\theta}(t)$  nesse período é também constante e, por isso,  $\theta(t) = \dot{\theta} \Delta t$ .

Contudo, mesmo impondo estas restrições, existe ainda uma distinção (aquando da integração) entre trajetos curvilíneos (eq. 12) e trajetos lineares (eq. 13).

$$\begin{cases} \omega_L = \frac{\Delta\theta(-L \sin(\Delta\theta) + 2\Delta x)}{2r\Delta t \sin(\Delta\theta)} \\ \omega_R = \frac{\Delta\theta(L \sin(\Delta\theta) + 2\Delta x)}{2r\Delta t \sin(\Delta\theta)} \end{cases} \quad \text{se } \Delta\theta \neq 0 \quad (12)$$

$$\begin{cases} \omega_L = \frac{\Delta x}{r\Delta t} \\ \omega_R = \frac{\Delta x}{r\Delta t} \end{cases} \quad \text{se } \Delta\theta = 0 \quad (13)$$

De salientar que, por a integração resultar num sistema sobredeterminado, optou-se por determinar as velocidades das rodas em ordem às variações de posição linear na direção  $X$  e angular o que resulta, em última instância, que a posição  $Y$  não possa ser imposta. Caso tal fosse necessário, seria necessário resolver o sistema de equações proveniente da integração em ordem a um dos pares  $x(t), y(t)$  ou  $\theta(t), y(t)$ .

### 2.2.2 Triciclo

No caso do triciclo proposto este pode, na realidade, ser visto como uma bicicleta em que a velocidade angular de tração  $\omega$  é aplicada à roda traseira. Este modelo tem a particularidade da orientação ser definida não só pela velocidade da roda de tração mas também pela posição da roda de direção (dianteira), como demonstrado pela eq. 14.

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = Rot(\theta(t)) \begin{bmatrix} 1 \\ 0 \\ \frac{\tan(\alpha(t))}{L} \end{bmatrix} \omega(t)r \quad (14)$$

Para se obter a cinemática inversa, como já previamente descrito, vários pressupostos têm de ser verdade para que se possam chegar a algumas soluções. Nomeadamente, a velocidade  $\omega(t)$  e posição  $\alpha(t)$  angulares das respetivas rodas serem constantes num dado intervalo de tempo e, consequentemente, a velocidade angular do robô ser também  $\dot{\theta}(t) = \dot{\theta} \Delta t$ .

Novamente, por estar definido um sistema sobredeterminado, optou-se por calcular os parâmetros das

rodas com base na posição linear em  $X$  e angular, o que se divide novamente em movimentos curvilíneos (eq. 15) ou lineares (eq. 16). Tal como anteriormente, esta decisão leva a que a posição em  $Y$  seja sempre calculada e nunca imposta.

$$\begin{cases} \alpha = \arctan\left(\frac{L \sin(\Delta\theta)}{\Delta x}\right) \\ \omega = \frac{\Delta\theta \Delta x}{r \Delta t \sin(\Delta\theta)} \end{cases} \quad \text{se } \Delta\theta \neq 0 \wedge \Delta x \neq 0 \quad (15)$$

$$\begin{cases} \alpha = \arctan\left(\frac{L \Delta\theta}{\Delta x}\right) \\ \omega = \frac{\Delta x}{r \Delta t} \end{cases} \quad \text{se } \Delta\theta = 0 \vee \Delta x = 0 \quad (16)$$

É de realçar a particularidade que este modelo cinemático apresenta e que é tornado evidente pela eq. 16: numa situação em que não exista movimento (i.e.  $w = 0$ ), a direção da roda dianteira  $\alpha$  torna-se indeterminável. Esta situação, de facto, faz sentido uma vez que a locomoção é providenciada pela roda traseira e, neste modelo, não havendo locomoção é impossível alterar a posição angular e, portanto, a posição  $\alpha$  pode tomar uma infinidade de valores sem que o pose do robô se altere.

### 2.2.3 Robô Omnidirecional

Dos modelos estudados, o robô omnidirecional é o único que permite definir as velocidades das suas rodas ( $\omega_1$ ,  $\omega_2$  e  $\omega_3$ ) em função da pose do robô (i.e.,  $X$ ,  $Y$  e  $\theta$ ). Estamos, portanto, perante um sistema definido.

Usando mais uma vez os pressupostos de que as velocidades angulares das rodas são constantes num dado intervalo de tempo e que, por conseguinte, a velocidade angular do robô é também ela constante, é possível integrar a cinemática direta da eq. 17 para se obter a cinemática inversa.

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = Rot(\theta(t)) \begin{bmatrix} 0 & \frac{r}{\sqrt{3}} & \frac{r}{\sqrt{3}} \\ \frac{-2r}{3} & \frac{r}{3} & \frac{-r}{3} \\ \frac{r}{3L} & \frac{r}{3L} & \frac{-r}{3L} \end{bmatrix} \begin{bmatrix} \omega_1(t) \\ \omega_2(t) \\ \omega_3(t) \end{bmatrix} \quad (17)$$

Contudo, a integração traz consigo um par de soluções, caso  $\omega_1 \neq -\omega_2 + \omega_3$  ou  $\omega_1 = -\omega_2 + \omega_3$ . A

restrição de igualdade foi a escolhida para determinar as velocidades a impor às rodas por não necessitar de nenhuma outra restrição e por ser possível resolver as equações em ordem às variáveis desejadas. Assim sendo, chega-se à cinemática inversa definida pela eq. 18.

$$\begin{cases} \omega_1 = \frac{L \Delta\theta - \Delta y}{r \Delta t} \\ \omega_2 = \frac{2L \Delta\theta + \sqrt{3} \Delta x + \Delta y}{2r \Delta t} \\ \omega_3 = \frac{-2L \Delta\theta + \sqrt{3} \Delta x - \Delta y}{2r \Delta t} \end{cases} \quad (18)$$

## 3 Análise de Resultados

Usando a metodologia descrita anteriormente, a implementação em *Matlab* foi avaliada, qualitativamente, para um ambiente com doze faróis e um robô com rodas de 0.15 metros separadas por 1 metros. Os sensores a bordo deste usam uma taxa de leitura de 1 Hz e os atuadores impõe as velocidades pretendidas com uma incerteza de 0.001 m/s e 0.001 rad/s. A velocidade média desejada foi de 5 m/s.

As Figs. 3 e 4 são particularmente interessantes para esta análise.

Na Fig. 3 percebe-se que no ponto inicial existe um desvio relativo ao trajeto planeado (Fig. 1) que pode ser explicado pelo facto da orientação inicial do robô ser  $\theta(0) = 0$  rad e, portanto, a primeira velocidade a impor ser também ela nesta direção (Fig. 2). À parte disto, e apesar da não coincidência dos trajetos planeado e efetuado, as similaridades são patentes o que indica a qualidade da estimativa.

A robustez do filtro de Kalman fica mais uma vez evidente verificando que, mesmo com os erros associados às observações (Fig. 4, especialmente a incerteza angular) as sucessivas poses estimadas são (salvo as exceções de curvas mais apertadas) praticamente coincidentes com a pose real do robô (Fig. 5).

Para efeitos comparativos, realizou-se o mesmo ensaio anterior, mas aumentando a incerteza na velocidade linear (Fig. 6) ou na velocidade angular (Fig. 7) para 0.1 (na respetiva unidade). Fica assim claro como uma maior incerteza na deteção angular tende a piorar substancialmente os resultados.

Com o intuito de avaliar eventuais requisitos excessivos aos atuadores dos robôs, a cinemática inversa foi

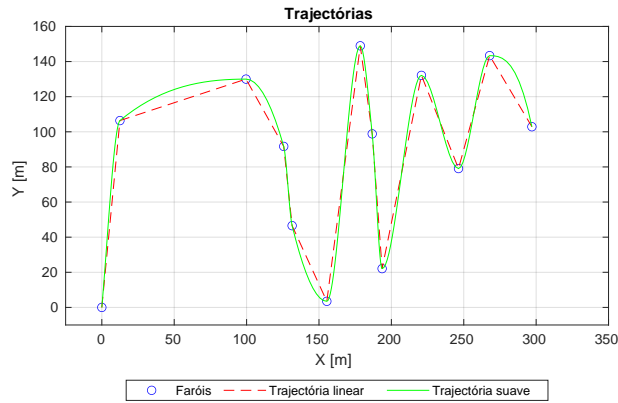


Figura 1: Trajetória para o caso de estudo.

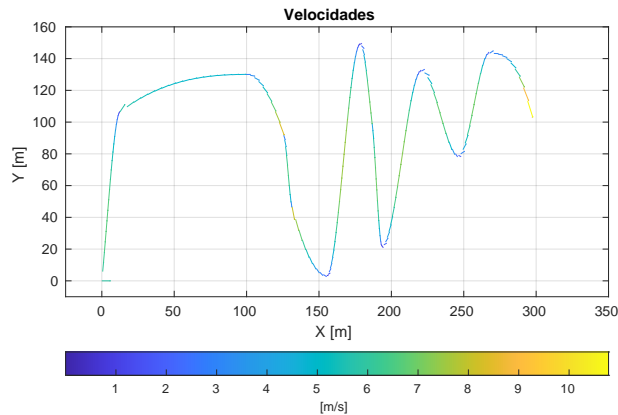


Figura 2: Velocidades calculadas para o caso de estudo.

aplicada a cada conjunto de velocidade requisitada. Os resultados demonstrados apontam para alguns detalhes importantes.

O modelo com tração diferencial apresenta várias discontinuidades no seu perfil de velocidades (Fig. 8). Estas discontinuidades são principalmente nas localizações dos faróis, onde as curvaturas são mais acentuadas.

Repare-se, ainda, na similaridade entre o perfil de velocidade da roda de tração no triciclo (Fig. 9) com o perfil geral das velocidades na tração diferencial. Esta similaridade existe uma vez que para o modelo utilizado para o triciclo, a roda de tração é na reali-

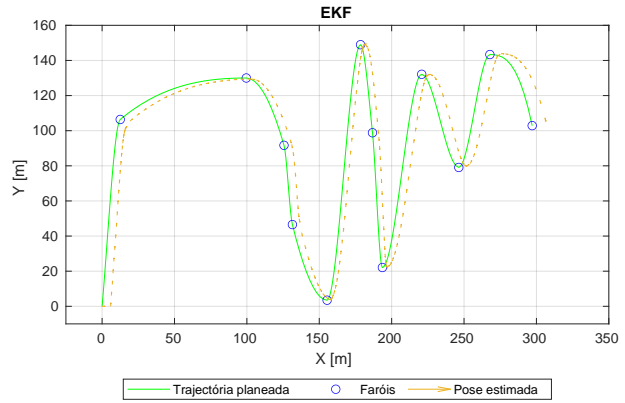


Figura 3: Localização pelo EKF para o caso de estudo.

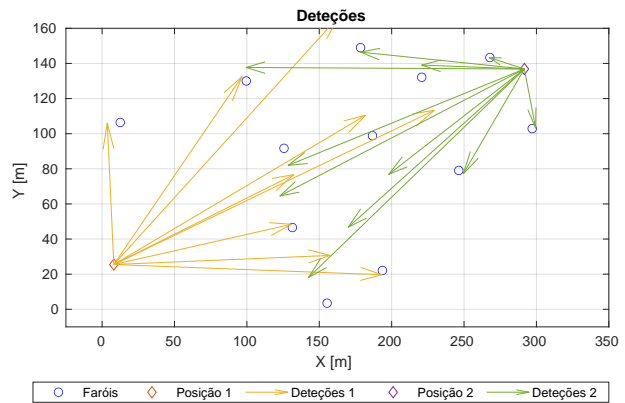


Figura 4: Exemplos de faróis detetados em duas poses distintas.

dade uma roda virtual que acaba por ser a média das rodas esquerda e direita. Ou seja, este triciclo quase que representa um modelo diferencial na traseira responsável apenas pela locomoção ao qual se adiciona uma roda dianteira para impor a orientação.

Já relativamente ao modelo omnidirecional (Fig. 10), este é o único que apresenta velocidades angulares negativas, o que era expectável dado o posicionamento circular das suas rodas e a possível rotação do robô em ambos os sentidos.

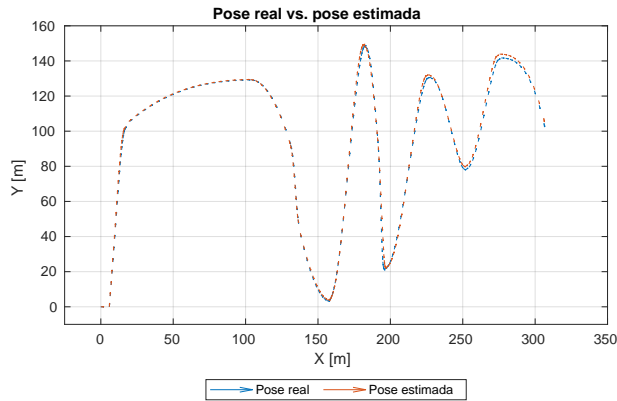


Figura 5: Comparação entre as poses estimadas e reais.

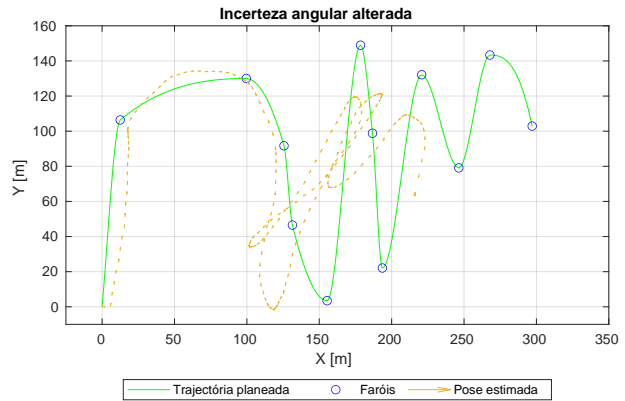


Figura 7: Influência da incerteza na velocidade angular nas sucessivas estimativas para a pose.

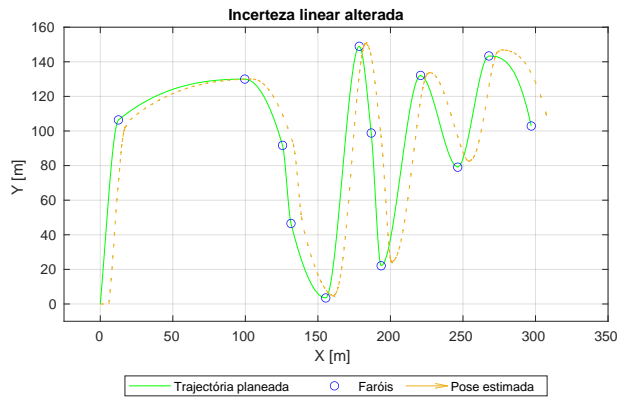


Figura 6: Influência da incerteza na velocidade linear nas sucessivas estimativas para a pose.

## 4 Conclusão

Ficou patente com esta análise a eficácia do filtro de Kalman, em particular na sua versão estendida. Mesmo com observações influenciadas por ruído, assumindo que dentro de limites aceitáveis, o sistema foi capaz de manter uma localização razoável.

A cinemática inversa provou ser a tarefa de maior complexidade devido às complexas equações diferenciais. Graças a ferramentas de cálculo simbólico como o *SymPy* ou o próprio *Matlab* e às respetivas restrições impostas, soluções são possíveis de ser definidas e validadas. Ainda assim, e como desenvolvi-

mento futuro, seria interessante estudar o funcionamento de um planeador genérico para tarefas similares às propostas neste projeto.

## Referências

- [1] C.F. Olson. «Probabilistic self-localization for mobile robots». Em: *IEEE Transactions on Robotics and Automation* 16.1 (2000), pp. 55–66. DOI: 10.1109/70.833191.
- [2] Aaron Meurer et al. «SymPy: symbolic computing in Python». Em: *PeerJ Computer Science* 3 (jan. de 2017), e103. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.103. URL: <https://doi.org/10.7717/peerj-cs.103>.
- [3] Rudolph Emil Kalman et al. «A new approach to linear filtering and prediction problems». Em: *Journal of basic Engineering* 82.1 (1960), pp. 35–45.

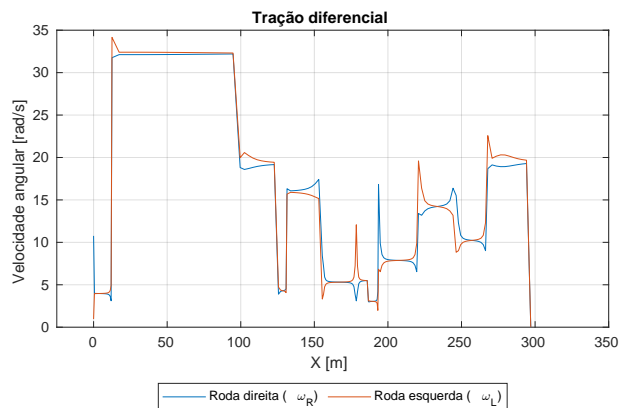


Figura 8: Velocidades das rodas do robô com tração diferencial para o caso de estudo descrito.

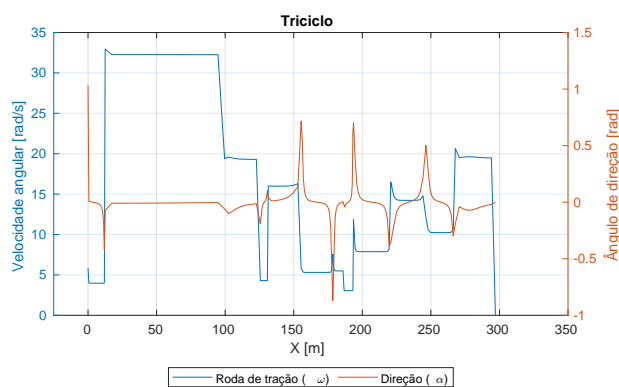


Figura 9: Velocidades das rodas do robô triciclo para o caso de estudo descrito.

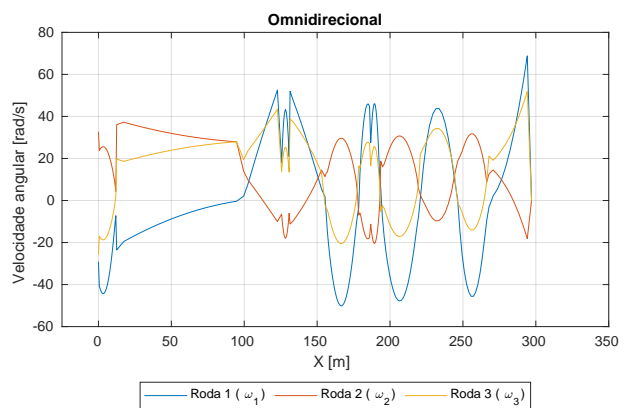


Figura 10: Velocidades das rodas do robô omnidirecional para o caso de estudo descrito.