

Faculdade de Engenharia da Universidade do Porto



Localização de Robôs Móveis Baseada em Marcos

Sandro Augusto Costa Magalhães
Tiago José Ferreira Mendonça

VERSÃO FINAL

Trabalho realizada no âmbito da unidade curricular de
Sistemas Robóticos Autónomos

Docentes: Vitor Pinto e António Moreira

6 de dezembro de 2017

1 Introdução

No presente trabalho laboratorial pretende-se desenvolver um algoritmo de localização para um robô móvel de tração diferencial baseado na detecção de paredes, isto é, o robô deve seguir uma determinada trajetória, neste caso um quadrado, sendo que as medições obtidas por recurso à odometria serão corrigidas através de dois sensores localizados na sua lateral, aquando da passagem pelos marcos. Assim, num primeiro instante, deve-se fazer com o que o robô siga o caminho pretendido, seguidamente obter-se as medidas dos sensores e, pela média das mesmas, proceder à atualização da sua posição. No sentido de recriar mais fidedignamente as condições de operação em contexto industrial, posteriormente, será adicionado ruído aleatório às medidas e retirar-se-ão as vantagens de utilização de um filtro de Kalman estendido para suprimir as perturbações introduzidas.

2 *FollowPath* – Definição da trajetória

Num primeiro momento, o principal propósito prende-se com a definição de uma trajetória em quadrado para o veículo. Para o efeito, tirou-se partido da função *followLine* previamente concebida nas aulas laboratoriais, que permite ao robô aproximar-se de um segmento de reta e seguir o mesmo, entre dois limites estabelecidos e na orientação pretendida. Neste caso particular, o trajeto pode ser decomposto em quatro linhas, dadas pelas seguintes coordenadas de origem e destino, respetivamente:

$$A_1 = (0.5, 0.5) \rightarrow B_1 = (0.5, -0.5)$$

$$A_2 = (0.5, -0.5) \rightarrow B_2 = (-0.5, -0.5)$$

$$A_3 = (-0.5, -0.5) \rightarrow B_3 = (-0.5, 0.5)$$

$$A_4 = (-0.5, 0.5) \rightarrow B_4 = (0.5, 0.5)$$

Deduz-se, assim, que a origem do referencial foi centrada no quadrado e que o robot inicia a trajetória em $(0.5, 0.5)$.

Posto isto, concebeu-se uma máquina de estados, em que cada um estava associado a uma linha a definir, sendo que no estado i eram estabelecidos os pontos limites da reta i (A_i e B_i) e, quando se atingia o extremo de um segmento, transitava-se para o estado seguinte, que define a nova linha a percorrer. O código implementado encontra-se na função *FollowPath*.

3 Algoritmo de Localização

Para percorrer o percurso definido, é imperativo conhecer a localização absoluta do robô em cada instante. Nesse sentido, fez-se uso do procedimento de cálculo com base na odometria proveniente de aulas laboratoriais anteriores. Contudo, esta técnica revela-se bastante falível dado os constantes incrementos de erros no decorrer do movimento, visto que nem todo o movimento rotacional das rodas é de facto convertido em deslocamento linear. Por consequência, observa-se que, ao descrever o circuito estabelecido, a posição do veículo diverge gradualmente das coordenadas pretendidas. Para colmatar este erro é necessário usufruir dos sensores dispostos lateralmente, nomeadamente das medições das suas distâncias aos marcos presentes no ambiente de simulação. Estes obstáculos estão colocados estrategicamente no terreno, em posições pré-definidas, de forma a que, quando detetados, seja possível, indiretamente, retirar a posição atual do robô e proceder à correção das estimativas obtidas por odometria. Adicionalmente, as paredes estão dispostas paralelamente a um dos eixos cartesianos, pelo que, embora seja sempre possível proceder à atualização da orientação do robô (θ), apenas se pode retificar a coordenada cartesiana (x ou y) correspondente ao eixo perpendicular. Por último, também se assume que as medições provenientes dos sensores se encontram afetadas por ruído gaussiano de média nula e desvio padrão variável, pelo que se procede à implementação de um filtro de Kalman estendido para estimação das observações.

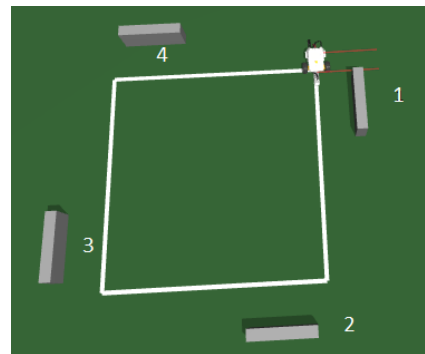
De forma a conceber um algoritmo ajustado ao raciocínio descrito, dividiu-se o programa desenvolvido em três módulos:

- *AssociationModule*;
- *ObservationModule*;
- *EstimationModule*.

4 *AssociationModule*

Dado que existem diversos marcos dispostos no terreno, quando se deteta uma observação proveniente dos sensores tem de se proceder à associação de um obstáculo à mesma. Só assim é possível saber quais as coordenadas suscetíveis de serem atualizadas e a magnitude das mesmas.

Uma vez que estes alvos estão suficientemente afastados e cada um disposto num quadrante diferente do referencial cartesiano, a sua relação com a medição obtida é facilmente decifrável pela estimativa atual da posição do robô facultada pela odometria. Assim, se ambas as coordenadas cartesianas forem simultaneamente positivas estamos perante o *marco 1*, se apenas a ordenada for negativa *marco 2*, ambas negativas *marco 3* e ordenada positiva e abcissa negativa *marco 4*.

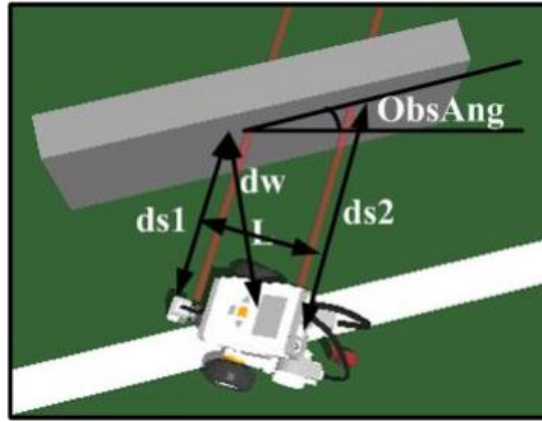


5 ObservationModule

Quando identificada qual a parede a ser detetada, avança-se para uma fase de cálculo, nomeadamente da distância do centro do veículo ao marco (dw) e do desvio angular face à orientação pretendida ($ObsAng$). Para esse fim, deve-se obter a medição de cada um dos sensores ($ds1$ e $ds2$) e, sabendo a distância que os separa ($L = 0.1\text{ m}$), obtêm-se as seguintes relações:

$$ObsAng = \text{atan} \left(\frac{ds1 - ds2}{L} \right)$$

$$dw = \frac{ds1 + ds2}{2} \cos(ObsAng)$$



Obtidos estes valores, é possível proceder ao cálculo da suposta posição atual do robô, implicando, no entanto, o conhecimento *a priori* da distância do marco à linha de trajetória. Este valor foi de antemão medido e é de $(0.2-0.025)\text{ m}$ (distância à parede interior). Tendo estas premissas por base, resultam as seguintes correções para as diferentes paredes ($Lw = 0.5+0.2-0.025$):

$$Z_1 = \begin{bmatrix} x \\ \theta \end{bmatrix} = \begin{bmatrix} Lw - dw \\ -\frac{\pi}{2} + ObsAng \end{bmatrix}$$

$$Z_2 = \begin{bmatrix} y \\ \theta \end{bmatrix} = \begin{bmatrix} -(Lw - dw) \\ -\pi + ObsAng \end{bmatrix}$$

$$Z_3 = \begin{bmatrix} x \\ \theta \end{bmatrix} = \begin{bmatrix} -(Lw - dw) \\ \frac{\pi}{2} + ObsAng \end{bmatrix}$$

$$Z_4 = \begin{bmatrix} y \\ \theta \end{bmatrix} = \begin{bmatrix} Lw - dw \\ ObsAng \end{bmatrix}$$

6 EstimationModule

Os resultados provenientes do procedimento anterior encontram-se afetados por várias sucessões de erros. Primeiramente, tal como anteriormente mencionado, o processo de odometria introduz erros nas estimativas de distância percorrida e de variação angular, refletindo-se em perturbações nas variáveis de processo. Por outro lado, o processo de medição com recurso a sensores também introduz erros nas saídas obtidas, pelo que estas não reproduzem os verdadeiros valores. Posto isto, é necessário proceder a técnicas de compensação das perturbações introduzidas ao longo desta cadeia de forma a retirar estimativas tão perto quanto possíveis dos valores reais. Nesse sentido, surge o previsor e filtro de Kalman. Este método modela os sinais de ruído como gaussianos de média nula e permite obter estimativas que minimizam a matriz associada às covariâncias dos erros de estimação das variáveis do processo (P), neste caso as coordenadas (x, y, θ) .

Na ausência de observações (longe dos marcos), recorre-se ao previsor, isto é, realiza-se uma etapa de predição em que as estimativas de posição são fornecidas com base na odometria e, por isso, a covariância cresce continuamente, visto que a técnica de odometria introduz erros que se acumulam de forma contínua e que, consequentemente, aumentam a incerteza associada à posição. Nesta fase, a posição é então facultada pela odometria e a covariância estimada é dada por:

$$P(k+1) = \nabla f_X P(k) \nabla f_X^T + \nabla f_q Q \nabla f_q^T$$

Sendo que:

$$\nabla f_X = \begin{bmatrix} 1 & 0 & -d \cdot s \left(\hat{\theta}(k) + \frac{\Delta\theta}{2} \right) \\ 0 & 1 & d \cdot c \left(\hat{\theta}(k) + \frac{\Delta\theta}{2} \right) \\ 0 & 0 & 1 \end{bmatrix} \quad \nabla f_q = \begin{bmatrix} c \left(\hat{\theta}(k) + \frac{\Delta\theta}{2} \right) & -\frac{1}{2}d \cdot s \left(\hat{\theta}(k) + \frac{\Delta\theta}{2} \right) \\ s \left(\hat{\theta}(k) + \frac{\Delta\theta}{2} \right) & \frac{1}{2}d \cdot c \left(\hat{\theta}(k) + \frac{\Delta\theta}{2} \right) \\ 0 & 1 \end{bmatrix}$$

No momento de observação, isto é, quando há a deteção de um marco, a covariância do erro diminui, de forma a refletir a maior certeza associada a estimativa da posição do robô. É neste instante que ocorre, então, a correção da estimativa da odometria. Nesse sentido, implementam-se as seguintes fórmulas.

$$\hat{X}(k+1) = \hat{X}(k) + W(k) \cdot V(k)$$

Em que $W(k)$ é o ganho do filtro de Kalman e dado por:

Os gradientes considerados são dados por ($C_1 = 1$ e $C_2 = 0$ se marco 1 ou marco 3 e $C_1 = 0$ e $C_2 = 1$ se marco 2 ou marco 4):

$$\nabla h_X = \begin{bmatrix} C_1 & C_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T \\ r \end{bmatrix}^{-1}$$

$$\nabla h_r = \begin{bmatrix} \text{signal} \cdot (a + b) & \text{signal} \cdot (a - b) \\ \frac{d\text{ObsAng}}{dd_{s1}} & -\frac{d\text{ObsAng}}{dd_{s1}} \end{bmatrix}$$

$$a = \frac{1}{2}c(\text{ObsAng}) \quad \frac{d\text{ObsAng}}{dd_{s1}} = \frac{L}{L^2 + (d_{s1} - d_{s2})^2}$$

$$b = -\frac{d_{s1} + d_{s2}}{2}s(\text{ObsAng})\frac{d\text{ObsAng}}{dd_{s1}}$$

Por último, a matriz de covariância da estimativa do erro deve ser inicializada. No instante inicial, esta deve ser diagonal com os elementos da mesma o mais elevado possível de forma a reproduzir a elevada nuvem de incerteza inerente a este estado inicial. Contudo, devem assumir valores plausíveis no contexto do problema. Posto isto, considerou-se:

$$P_0 = \begin{bmatrix} 0.5^2 & 0 & 0 \\ 0 & 0.5^2 & 0 \\ 0 & 0 & \left(\frac{\pi}{2}\right)^2 \end{bmatrix}$$

Também as matrizes de covariância associadas ao estado (Q) e às saídas do sistema (R) devem ser definidas, tendo-se considerado:

$$Q = \begin{bmatrix} 0.0008 & 0 \\ 0 & 0.001 \end{bmatrix} R = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}, \text{ em que } \sigma \text{ corresponde ao desvio padrão do ruído que é } 0.002.$$

6.1 Previsão

Calculando cada um dos parâmetros, para a previsão da posição do robô quando não existem marcos para a correção, temos que a matriz P resulta em:

$$\begin{pmatrix} Q11 \, dfq11^2 + Q22 \, dfq12^2 + Pxx + Pxo \, dfx13 + dfx13 \, (Pox + Poo \, dfx13) & Pxy + Pxo \, dfx23 + dfx13 \, (Poy + Poo \, dfx23) + Q11 \, dfq11 \, dfq21 + Q22 \, dfq12 \, dfq22 & Pxo + Poo \, dfx13 + Q22 \, dfq12 \\ Pyx + Pyo \, dfx13 + dfx23 \, (Pox + Poo \, dfx13) + Q11 \, dfq11 \, dfq21 + Q22 \, dfq12 \, dfq22 & Q11 \, dfq21^2 + Q22 \, dfq22^2 + Pyy + Pyo \, dfx23 + dfx23 \, (Poy + Poo \, dfx23) & Pyo + Poo \, dfx23 + Q22 \, dfq22 \\ Pox + Poo \, dfx13 + Q22 \, dfq12 & Poy + Poo \, dfx23 + Q22 \, dfq22 & Poo + Q22 \end{pmatrix}$$

6.2 Correção

Por outro lado, quando estamos perante um marco temos primeiramente que decidir qual é o marco que estamos a observar e, de seguida, calcular as expressões associadas.

Portanto, se o marco for dedicado à correção da abcissa, conclui-se que W , o ganho do filtro de Kalman, é igual a

$$\begin{pmatrix} \frac{p11 \sigma_2}{\sigma_1} - \frac{p13 \sigma_4}{\sigma_1} - \frac{p13 \sigma_2}{\sigma_1} - \frac{p11 \sigma_5}{\sigma_1} \\ \frac{p21 \sigma_2}{\sigma_1} - \frac{p23 \sigma_4}{\sigma_1} - \frac{p23 \sigma_2}{\sigma_1} - \frac{p21 \sigma_5}{\sigma_1} \\ \frac{p31 \sigma_2}{\sigma_1} - \frac{p33 \sigma_4}{\sigma_1} - \frac{p33 \sigma_2}{\sigma_1} - \frac{p31 \sigma_5}{\sigma_1} \end{pmatrix}$$

where

$$\sigma_1 = p11 \ p33 - p13 \ p31 + h11^2 \ p11 \ r11 + h11^2 \ p33 \ r11 + h22^2 \ p11 \ r22 + h12^2 \ p33 \ r22 - h11 \ h21 \ p13 \ r11 - h12 \ h22 \ p13 \ r22 - h11 \ h21 \ p31 \ r11 - h12 \ h22 \ p31 \ r22 + h11^2 \ h22^2 \ r11 \ r22 + h12^2 \ h21^2 \ r11 \ r22 - 2 \ h11 \ h12 \ h21 \ h22 \ r11 \ r22$$

$$\sigma_2 = r11 \ h21^2 + r22 \ h22^2 + p33$$

$$\sigma_3 = r11 \ h11^2 + r22 \ h12^2 + p11$$

$$\sigma_4 = p31 + h11 \ h21 \ r11 + h12 \ h22 \ r22$$

$$\sigma_5 = p13 + h11 \ h21 \ r11 + h12 \ h22 \ r22$$

E a matriz das covariâncias é igual a

$$\begin{pmatrix} p31 \ \sigma_6 + p11 \ \sigma_1 & p32 \ \sigma_6 + p12 \ \sigma_1 & p33 \ \sigma_6 + p13 \ \sigma_1 \\ p21 + p31 \ \sigma_5 + p11 \ \sigma_4 & p22 + p32 \ \sigma_5 + p12 \ \sigma_4 & p23 + p33 \ \sigma_5 + p13 \ \sigma_4 \\ p31 \ \sigma_2 + p11 \ \sigma_3 & p32 \ \sigma_2 + p12 \ \sigma_3 & p33 \ \sigma_2 + p13 \ \sigma_3 \end{pmatrix}$$

where

$$\sigma_1 = \frac{p13 \ \sigma_8}{\sigma_9} - \frac{p11 \ \sigma_7}{\sigma_9} + 1$$

$$\sigma_2 = \frac{p31 \ \sigma_{11}}{\sigma_9} - \frac{p33 \ \sigma_{10}}{\sigma_9} + 1$$

$$\sigma_3 = \frac{p33 \ \sigma_8}{\sigma_9} - \frac{p31 \ \sigma_7}{\sigma_9}$$

$$\sigma_4 = \frac{p23 \ \sigma_8}{\sigma_9} - \frac{p21 \ \sigma_7}{\sigma_9}$$

$$\sigma_5 = \frac{p21 \ \sigma_{11}}{\sigma_9} - \frac{p23 \ \sigma_{10}}{\sigma_9}$$

$$\sigma_6 = \frac{p11 \ \sigma_{11}}{\sigma_9} - \frac{p13 \ \sigma_{10}}{\sigma_9}$$

$$\sigma_7 = r11 \ h21^2 + r22 \ h22^2 + p33$$

$$\sigma_8 = p31 + h11 \ h21 \ r11 + h12 \ h22 \ r22$$

$$\sigma_9 = p11 \ p33 - p13 \ p31 + h21^2 \ p11 \ r11 + h11^2 \ p33 \ r11 + h22^2 \ p11 \ r22 + h12^2 \ p33 \ r22 - h11 \ h21 \ p13 \ r11 - h12 \ h22 \ p13 \ r22 - h11 \ h21 \ p31 \ r11 - h12 \ h22 \ p31 \ r22 + h11^2 \ h22^2 \ r11 \ r22 + h12^2 \ h21^2 \ r11 \ r22 - 2 \ h11 \ h12 \ h21 \ h22 \ r11 \ r22$$

$$\sigma_{10} = r11 \ h11^2 + r22 \ h12^2 + p11$$

$$\sigma_{11} = p13 + h11 \ h21 \ r11 + h12 \ h22 \ r22$$

Se, por outro lado, o marco que estamos a observar, nos permitir corrigir a ordenada, temos, respetivamente, as matrizes W e P iguais a

$$\begin{pmatrix} \frac{p12 \ \sigma_2}{\sigma_1} - \frac{p13 \ \sigma_4}{\sigma_1} - \frac{p13 \ \sigma_2}{\sigma_1} - \frac{p12 \ \sigma_5}{\sigma_1} \\ \frac{p22 \ \sigma_2}{\sigma_1} - \frac{p23 \ \sigma_4}{\sigma_1} - \frac{p23 \ \sigma_2}{\sigma_1} - \frac{p22 \ \sigma_5}{\sigma_1} \\ \frac{p32 \ \sigma_2}{\sigma_1} - \frac{p33 \ \sigma_4}{\sigma_1} - \frac{p33 \ \sigma_2}{\sigma_1} - \frac{p32 \ \sigma_5}{\sigma_1} \end{pmatrix}$$

where

$$\sigma_1 = p22 \ p33 - p23 \ p32 + h21^2 \ p22 \ r11 + h11^2 \ p33 \ r11 + h22^2 \ p22 \ r22 + h12^2 \ p33 \ r22 - h11 \ h21 \ p23 \ r11 - h11 \ h21 \ p32 \ r11 - h12 \ h22 \ p23 \ r22 - h12 \ h22 \ p32 \ r22 + h11^2 \ h22^2 \ r11 \ r22 + h12^2 \ h21^2 \ r11 \ r22 - 2 \ h11 \ h12 \ h21 \ h22 \ r11 \ r22$$

$$\sigma_2 = r11 \ h21^2 + r22 \ h22^2 + p33$$

$$\sigma_3 = r11 \ h11^2 + r22 \ h12^2 + p22$$

$$\sigma_4 = p32 + h11 \ h21 \ r11 + h12 \ h22 \ r22$$

$$\sigma_5 = p23 + h11 \ h21 \ r11 + h12 \ h22 \ r22$$

e a

$$\begin{pmatrix} p_{11} + p_{31} \sigma_6 + p_{21} \sigma_4 & p_{12} + p_{32} \sigma_6 + p_{22} \sigma_4 & p_{13} + p_{33} \sigma_6 + p_{23} \sigma_4 \\ p_{31} \sigma_5 + p_{21} \sigma_1 & p_{32} \sigma_5 + p_{22} \sigma_1 & p_{33} \sigma_5 + p_{23} \sigma_1 \\ p_{31} \sigma_2 + p_{21} \sigma_3 & p_{32} \sigma_2 + p_{22} \sigma_3 & p_{33} \sigma_2 + p_{23} \sigma_3 \end{pmatrix}$$

where

$$\sigma_1 = \frac{p_{23} \sigma_8}{\sigma_9} - \frac{p_{22} \sigma_7}{\sigma_9} + 1$$

$$\sigma_2 = \frac{p_{32} \sigma_{11}}{\sigma_9} - \frac{p_{33} \sigma_{10}}{\sigma_9} + 1$$

$$\sigma_3 = \frac{p_{33} \sigma_8}{\sigma_9} - \frac{p_{32} \sigma_7}{\sigma_9}$$

$$\sigma_4 = \frac{p_{13} \sigma_8}{\sigma_9} - \frac{p_{12} \sigma_7}{\sigma_9}$$

$$\sigma_5 = \frac{p_{22} \sigma_{11}}{\sigma_9} - \frac{p_{23} \sigma_{10}}{\sigma_9}$$

$$\sigma_6 = \frac{p_{12} \sigma_{11}}{\sigma_9} - \frac{p_{13} \sigma_{10}}{\sigma_9}$$

$$\sigma_7 = r_{11} h_{21}^2 + r_{22} h_{22}^2 + p_{33}$$

$$\sigma_8 = p_{32} + h_{11} h_{21} r_{11} + h_{12} h_{22} r_{22}$$

$$\sigma_9 = p_{22} p_{33} - p_{23} p_{32} + h_{21}^2 p_{22} r_{11} + h_{11}^2 p_{33} r_{11} + h_{22}^2 p_{22} r_{22} + h_{12}^2 p_{33} r_{22} - h_{11} h_{21} p_{23} r_{11} - h_{11} h_{21} p_{32} r_{11} - h_{12} h_{22} p_{23} r_{22} - h_{12} h_{22} p_{32} r_{22} + h_{11}^2 h_{22}^2 r_{11} r_{22} + h_{12}^2 h_{21}^2 r_{11} r_{22} - 2 h_{11} h_{12} h_{21} h_{22} r_{11} r_{22}$$

$$\sigma_{10} = r_{11} h_{11}^2 + r_{22} h_{12}^2 + p_{22}$$

$$\sigma_{11} = p_{23} + h_{11} h_{21} r_{11} + h_{12} h_{22} r_{22}$$

Com base em tudo o que foi dito, implementou-se o procedimento *EstimationModule*.

7 Resultados

No sentido de reduzir o risco associado a uma implementação completa, optou-se por uma incremental que nos permitiu calibrar e testar cada parte do trabalho iterativamente.

Tal como referido, começou-se então por fazer o robot seguir a trajetória pretendida, utilizando a localização exata robô, obtidas através das funções próprias do simulador. Desta primeira implementação, conseguiu-se obter uma trajetória precisa do robô sem a necessidade de qualquer ajuste, tal como demonstra a figura 7.1. Nota-se, no entanto um ligeiro *overshoot* na trajetória do robô e uma pequena deformação da mesma quando o robô segue para a sua quarta linha. Esta é provocada essencialmente devido a um ajuste que o corre no ângulo do robô, desorientando-o momentaneamente.

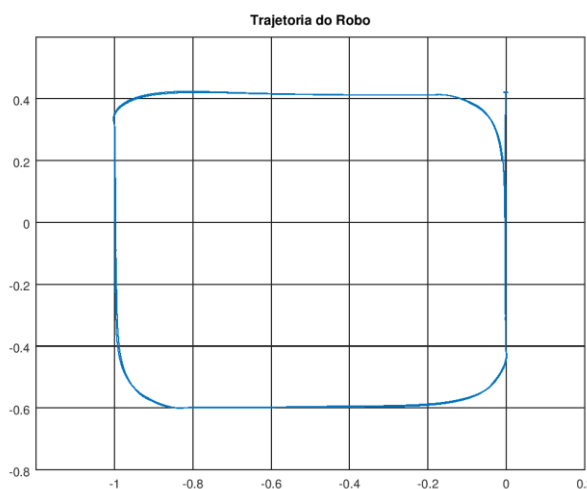


Figura 7.1 Trajetória do robô usando a sua posição real

Numa segunda fase da implementação, substituímos a localização real do robô por uma estimada através da odometria e que é corrigida periodicamente em marcos colocados estrategicamente no mapa, tal como referido na secção 4. Desta alteração no modo de localização do robô é imediatamente notável o aumento da sua desorientação ao longo do tempo, tal como demonstra a figura 7.2.

Nesta imagem observa-se o movimento do robô ao longo de três trajetórias. Numa fase inicial, verifica-se que a diferente forma de funcionamento dos motores e eventuais derrapagens que possam ocorrer fazem, desde logo o robô desviar-se da trajetória pretendida, mas quando este encontra um marco, ele é capaz de corrigir a sua posição, assim como a sua trajetória.

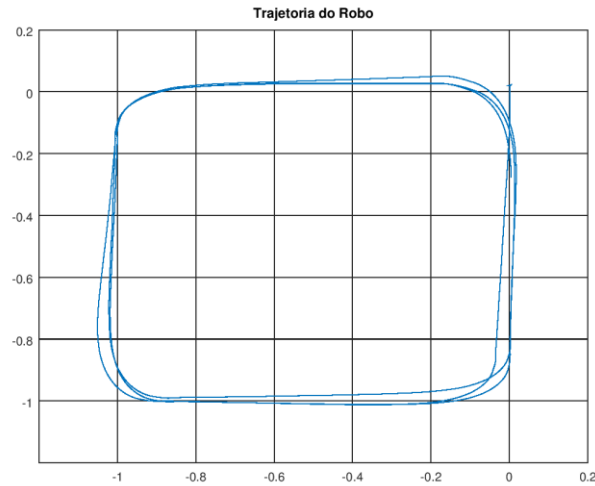


Figura 7.2 Trajetória do robô, tendo por base a odometria e a localização baseada em da em marcos

Nesta fase, consta-se ainda que embora a percurso do robô não seja sempre igual em cada rota, esta circula sempre a trajetória pretendida, sem que se perca totalmente.

Por fim, por forma a atingir uma simulação próxima do real, adicionou-se aos sensores um ruído branco gaussiano tal como referido na secção 3.

Da figura 7.3, consegue-se concluir que quando sujeito a ruído nos sensores exteroceivos, o seguimento de uma estratégia baseada numa correção determinística, não é a solução mais adequada, uma vez que estamos a submeter erros esporádicos no sistema que podem levar à perda de localização do robô.

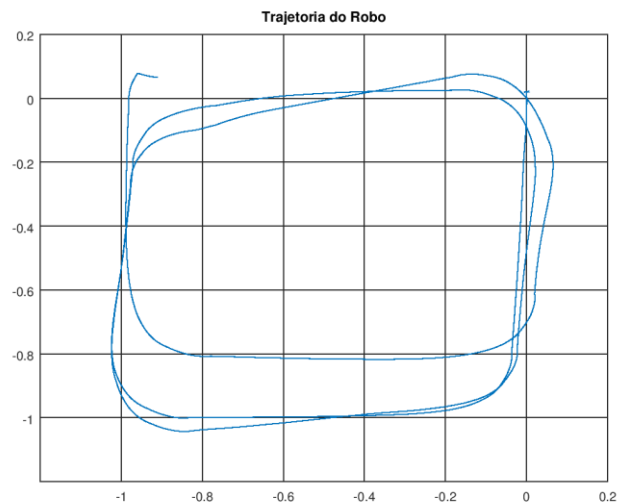


Figura 7.3 Trajetória do robô localizado através da odometria e de marcos com ruído

Como tal, no sentido de contornar esta situação, procurou-se implementar um filtro de Kalman. Contudo, os resultados obtidos não foram satisfatórios, tornando-se mesmo piores que os obtidos com o modelo determinístico.

Analisando mais proximamente a evolução dos dados verifica-se que o filtro está a demorar muito tempo a convergir, não sendo suficiente o tempo em que o robô está perante um marco. Em suma, pode-se concluir que não se conseguiu atingir uma calibração satisfatória dos parâmetros do filtro.

8 Generalização da disposição dos marcos

No problema considerado, a parede a detetar encontrava-se paralela a um dos eixos cartesianos, não possibilitando, desta forma, a correção simultânea de todas as coordenadas do robô no momento de observação. Contudo, tal aspeto pode ser colmatado alterando a orientação do marco. Desta forma, com uma orientação tal que o seu eixo seja oblíquo a ambos os eixos cartesianos, as coordenadas de posição seriam todas atualizadas aquando da passagem por este, conduzindo à redução do número de obstáculos necessários no ambiente de simulação para a localização global do veículo.

Assim, considere-se a situação hipotética em que o centro geométrico de um marco apresenta as coordenadas (x_m, y_m) no referencial global e um ângulo α entre o eixo que o atravessa e a direção vertical. Assumindo que todas as observações ocorrem de forma a que o ponto médio entre os sensores coincida com o centro geométrico da parede (tal como sugerido na figura), as coordenadas corrigidas do veículo serão dadas por:

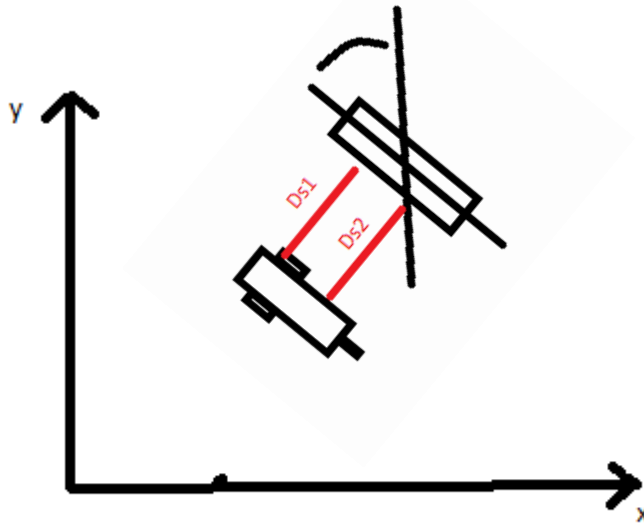


Figura 4 - Posição hipotética de um marco e deteção pelos sensores do veículo

$$x = x_m - dw \cdot \cos(ObsAng + Alpha)$$

$$y = y_m - dw \cdot \sin(ObsAng + Alpha)$$

$$\theta = \beta + ObsAng + Alpha, \beta - \text{orientação pretendida para a trajetória do veículo}$$

Sendo que:

$$ObsAng = \text{atan}\left(\frac{Ds1 - Ds2}{L}\right), L - \text{distância entre sensores}$$

$$dw = \frac{Ds1 - Ds2}{2} \cdot \cos(ObsAng)$$

9 Utilização de *scan laser*

Um *scan laser* apresenta um funcionamento análogo a um sensor IR, isto é, detecção do obstáculo num determinado alcance. Contudo, enquanto que no caso de estudo tínhamos apenas dois raios de luz isolados, cada um associado a um sensor, na aplicação de um *scan laser* surge um varrimento contínuo num determinado intervalo angular. Como tal, resulta um número drasticamente superior de medições realizadas, sendo que a distância à parede corresponderá à de menor amplitude. Com o intuito de proceder à localização do veículo, poder-se-á considerar o ângulo entre esta medição e a orientação do laser tendo em conta a sua disposição no robô. Em virtude da larga densidade de medidas utilizadas, esta técnica revela-se mais robusta e poderá conduzir a um menor erro na estimação da pose do veículo. Adicionalmente, também torna a solução mais imune à presença de *outliers*.

Finalmente, uma estratégia desta natureza também legitimaria o recurso a outros algoritmos, como poderá ser o caso da localização por ajuste de mapas, permitindo, através de um varrimento da zona a percorrer, a definição de um mapa do percurso e de todos os elementos neste presente e que condicionariam a trajetória.