

[S2] - FastSLAM com Visual Markers

Daniel Fortunato
81498

daniel.fortunato@tecnico.ulisboa.pt

Daniel Nunes
94034

danielsebastiaonunes@tecnico.ulisboa.pt

David Ribeiro
84027

david.a.c.ribeiro@tecnico.ulisboa.pt

Pedro Fareleira
79074

pedro.miguel.fareleira.dos.santos@tecnico.ulisboa.pt

MEEC

Instituto Superior Técnico, Universidade de Lisboa

Introdução

O princípio do SLAM consiste em usar o método de localização relativa para estimar a posição e num modelo de observação para mapear todo o ambiente à volta em relação à localização do agente em causa. Um dos métodos usados recentemente é o chamado de FastSLAM, que representa uma abordagem estocástica, onde um *particle filter* é implementado para estimar o estado mais provável. As observações dadas ao algoritmo do SLAM podem vir de uma série de sensores geralmente usados em problemas de mapeamento. Neste trabalho foi usada um câmara *Kinect* como sensor, cuja função era capturar *visual markers* e fornecer a localização relativa dos mesmos.

Algoritmo

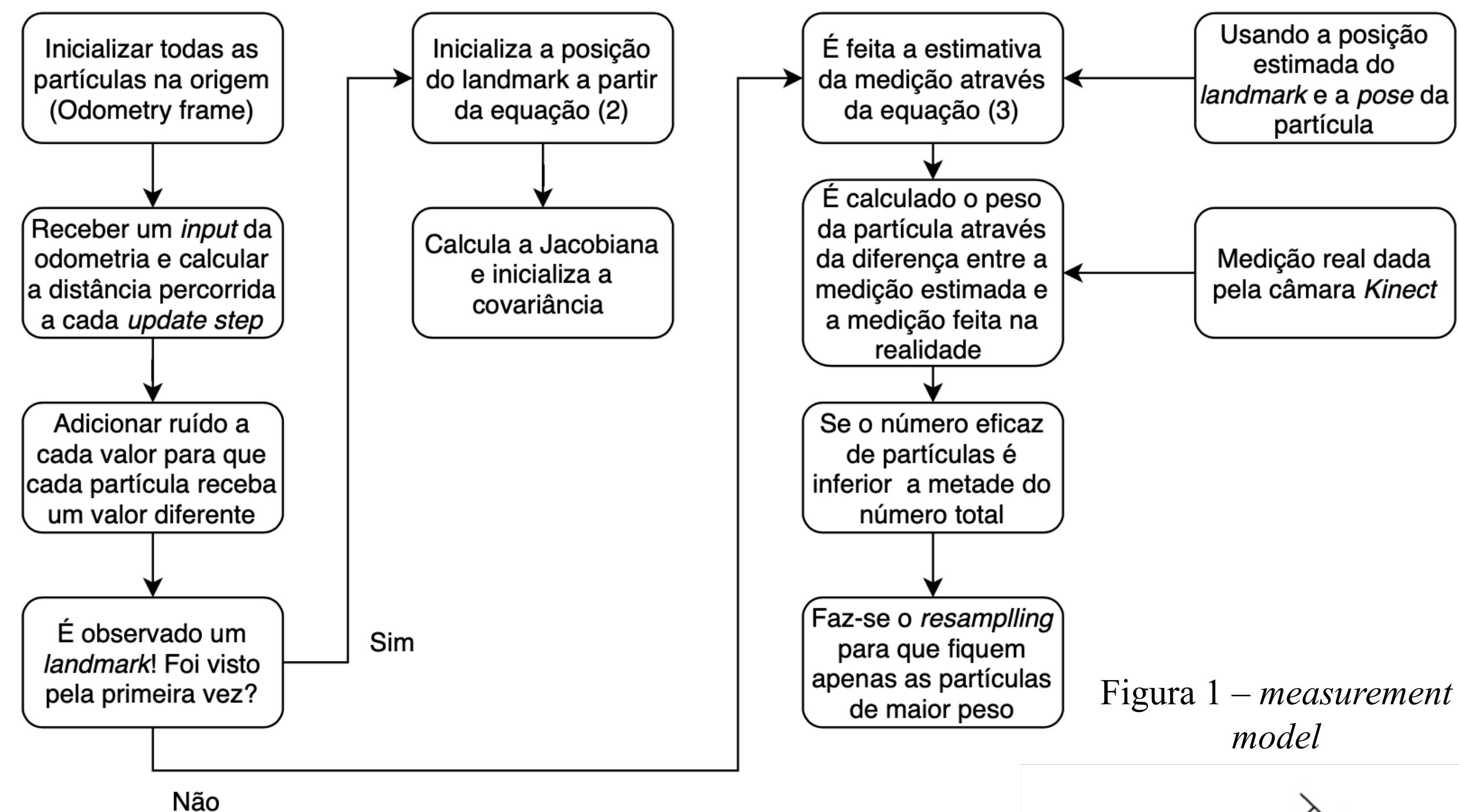
Para cada partícula, a distribuição conjunta de uma *pose* e de um *landmark* dadas as observações e os movimentos é fatorizada, baseada na *Rao-Blackwellization*, e reescrita como o produto da probabilidade da *pose* dadas as observações e os movimentos (*path posterior*) e a probabilidade do mapa dada a posição e as observações (*map posterior*):

$$p(s^t, L|n^t, z^t, u^t) = p(s^t|n^t, z^t, u^t) \prod_{n=1}^N p(l_n|s^t, n^t, z^t) \quad (1)$$

Cada partícula tem:

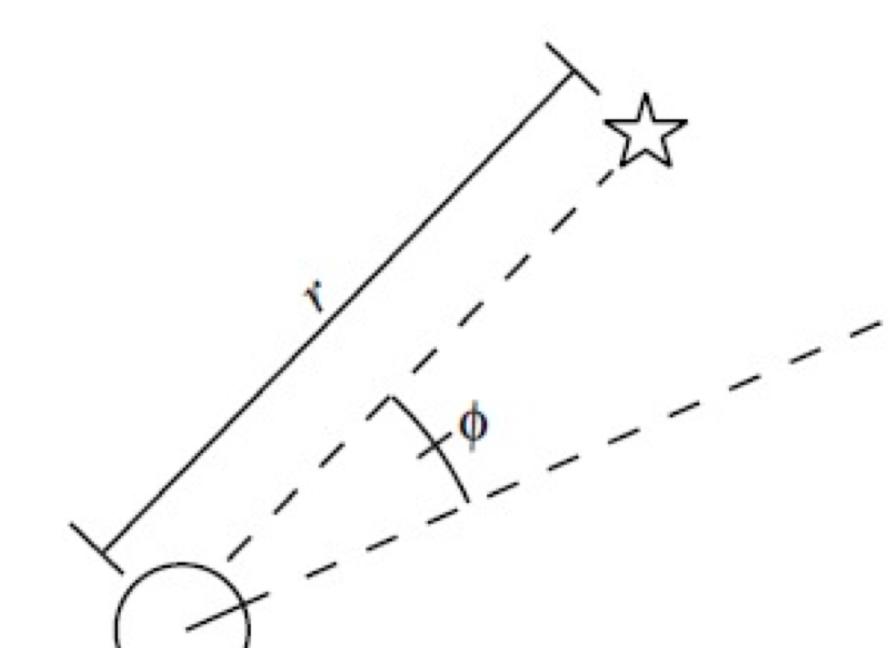
- Uma representação de uma possível *pose* do robô;
- Um *Extendend Kalman Filter* (EKF) para a representação de cada *landmark*.

Implementação



$$\begin{cases} x_M = x_P + d_{pM} \cos(\theta + \phi_{pM}) \\ y_M = y_P + d_{pM} \sin(\theta + \phi_{pM}) \end{cases} \quad (2)$$

$$\hat{Z} = [\sqrt{(x_M - x_p)^2 + (y_M - y_p)^2}, \arctg(\frac{y_M - y_p}{x_M - x_p}) - \theta_p] \quad (3)$$



Resultados

Para obter alguns dados sobre o desempenho do nosso algoritmo foram realizadas várias experiências. Operámos em três espaços físicos distintos do piso 5 da Torre Norte do Instituto Superior Técnico: os corredores do piso (que constituem um quadrado quando percorridos), a zona de elevadores e o laboratório de Sistemas Autónomos. Testámos também vários tipos de percurso.

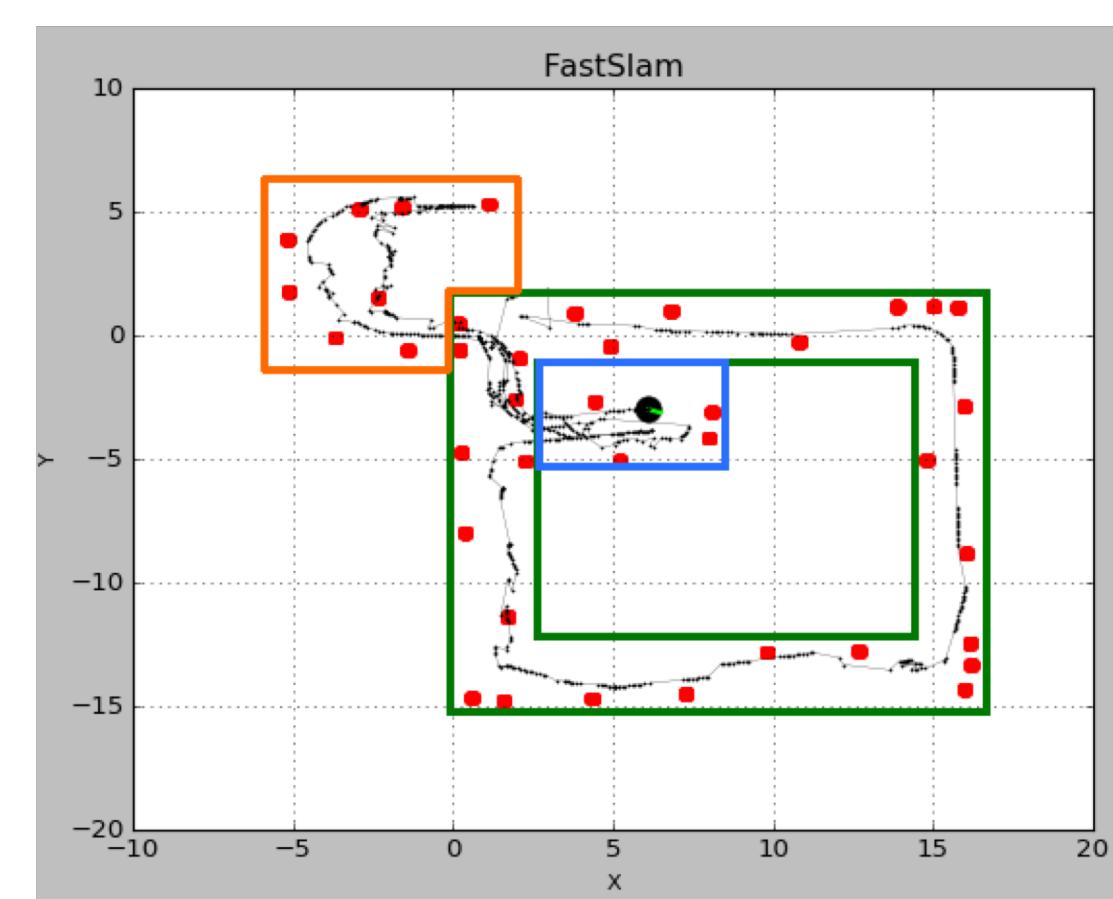


Figura 2 – Todos os locais

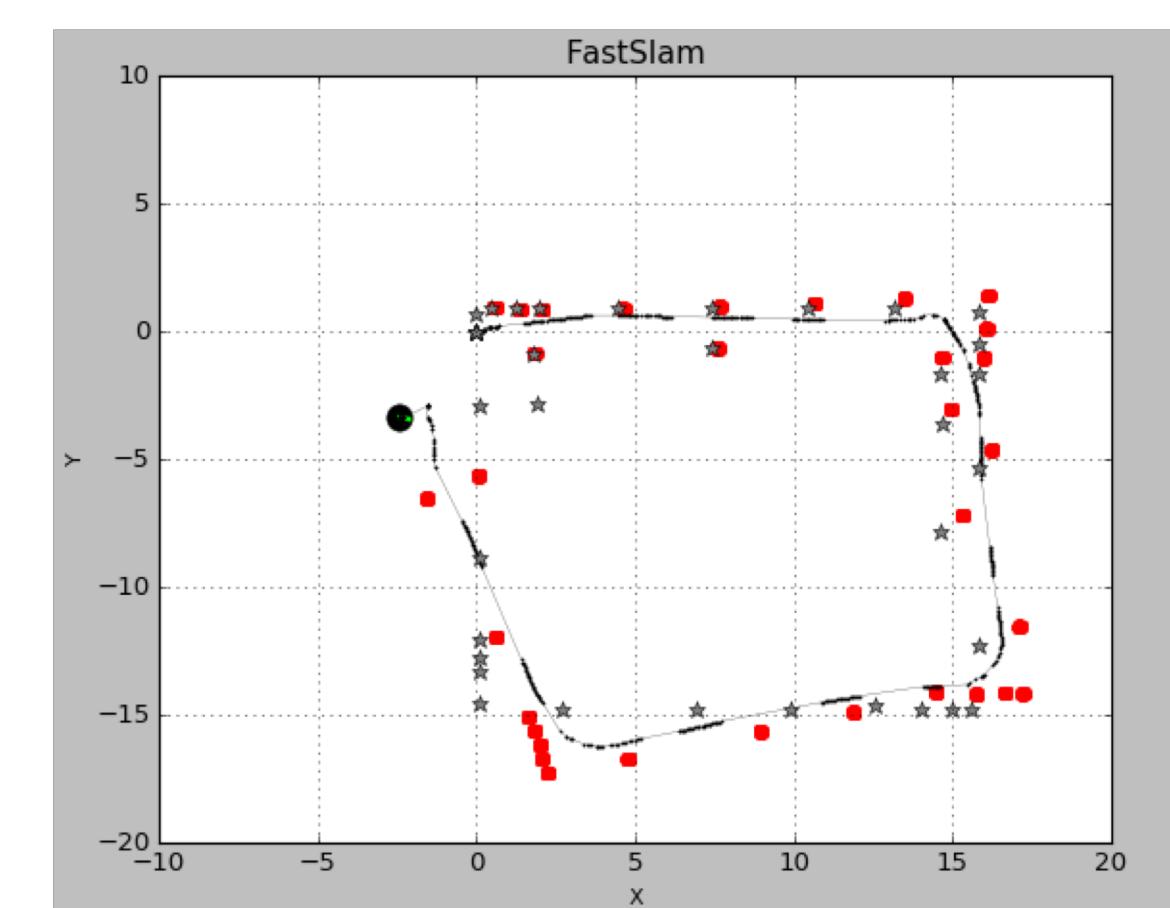


Figura 3 – Corredor sentido horário

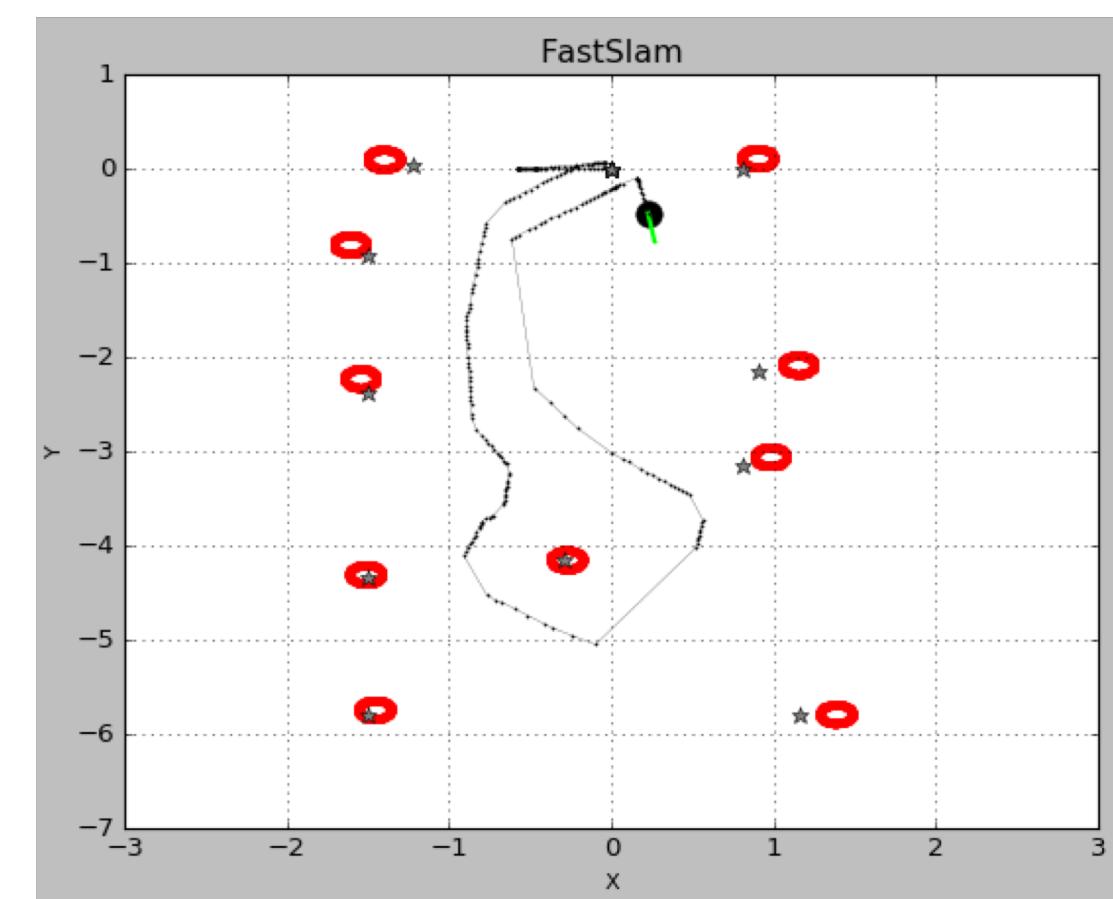


Figura 4 – Elevadores anti-horário

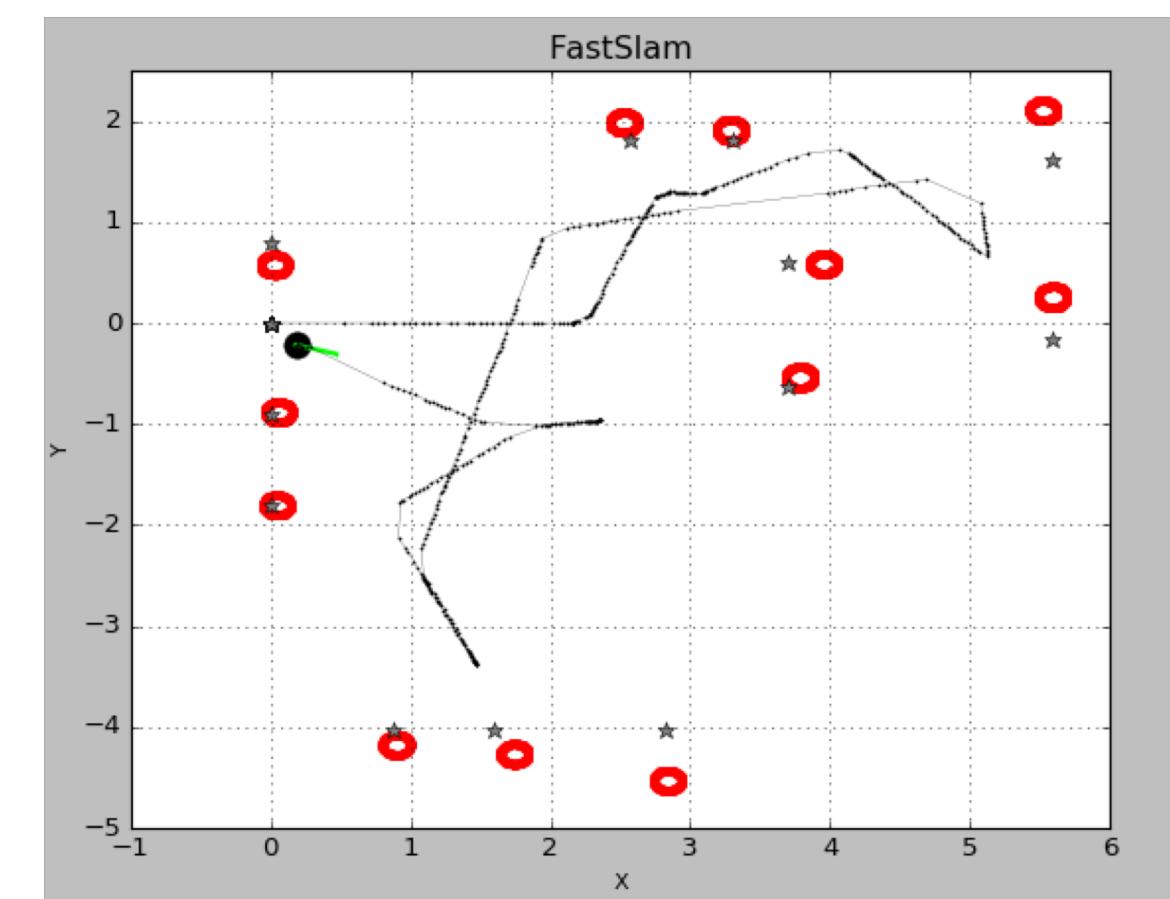


Figura 5 – Laboratório perc. dinâmico

	Anti-horário	Horário
Média (m)	0.1463	0.3603
Desvio Padrão (m)	0.0744	0.2408
Mínimo (m)	0.0152	0.0859
Máximo (m)	0.2491	0.9774

Tabela 1 – Erros na zona de elevadores

	Percuso A	Percuso B
Média (m)	0.2361	0.3425
Desvio Padrão (m)	0.1516	0.1111
Mínimo (m)	0.0439	0.0582
Máximo (m)	0.5083	0.5068

Tabela 2 – Erros no laboratório

Número de partículas	10	100	1000
Média (m)	0.3217	0.2338	0.4140
Desvio Padrão (m)	0.1113	0.0954	0.1203
Mínimo (m)	0.1454	0.0659	0.2194
Máximo (m)	0.4750	0.4042	0.6557

Tabela 3 – Erros para diferente nº de partículas

	Tempo
Média (ms)	23.8
Desvio Padrão (ms)	8.7
Mínimo (ms)	15.3
Máximo (ms)	62.1

Tabela 4 – Tempo das iterações

Conclusão

O algoritmo mostra-se robusto em certas situações, apresentando erros mínimos, contudo, em boa parte dos casos apresenta algum erro e não se mostrou capaz de se localizar em situações de maior movimentação independente. Ponderam-se que possíveis causas para este problema sejam erros de odometria elevados entre iterações relacionados com os sensores do Pioneer, possíveis erros na leitura da posição dos *markers* ou uma má estimativa da evolução do erro ao longo do percurso. A nível aplicacional, o projeto desenvolvido mostra-se muito interessante, pois, apesar de termos considerado algumas simplificações na nossa abordagem acabámos por desenvolver a base/essência do FastSLAM, que pode ser aplicado a projetos mais ambiciosos, como, por exemplo, na localização e mapeamento simultâneo para um robô da NASA, em Marte.