

UMA ABORDAGEM HÍBRIDA PARA LOCALIZAÇÃO E MAPEAMENTO SIMULTÂNEO PARA ROBÔS MÓVEIS COM SONARES ATRAVÉS DO FILTRO DE KALMAN ESTENDIDO

ALAN PORTO BONTEMPO¹, MARLEY VELLASCO¹, KARLA FIGUEIREDO¹.

1. *Laboratório de Inteligência Computacional Aplicada - ICA, Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro*
Rua Marquês de São Vicente 225, Gávea
CEP 22451-900 – Rio de Janeiro - RJ
E-mails: alanbontempo@gmail.com, marley@ele.puc-rio.br, karlafigueiredo@uezo.rj.gov.br

Abstract—This work addresses the problem of Simultaneous Localization and Mapping in structured environments using a mobile robot equipped with sonar, electronic compass and encoders. The proposed method, called H-SLAM (Hybrid - Simultaneous Localization and Mapping), makes use of kinds of environment representation: Occupancy Grid Map and Continuous Representation. The Occupancy Grid Map divides the environment into small equal parts, and classifies it as occupied or empty. The Continuous Representation uses lines to represent detected planes in the environment, forming a two-dimensional map. Each line of the map is considered a landmark. Every time a plan is redetected by the robot the corresponding line to it is rebuild, once the amount of information about the environmental increase. The robot's position is updated through Extended Kalman Filter. The model effectiveness was proved with computer simulations in three virtual environments. Using a prototype developed with LEGO Mindstorms platform three other experiments were also performed in real environments. The results demonstrated the effectiveness and the convergence of the proposed method.

Keywords— Mobile Robotics, SLAM, Sonar, Extended Kalman Filter

Resumo— Este trabalho aborda o problema de Localização e Mapeamento Simultâneos em ambientes estruturados, utilizando um robô móvel equipado com sonares, bússola eletrônica e encoders. O método proposto, denominado de LMS-H (Localização e Mapeamento Simultâneos - Híbrido), faz uso de duas formas de representação do ambiente: Mapa de Ocupação em Grade e Representação Contínua. O Mapa de Ocupação em Grade divide o ambiente em pequenas partes iguais, classificando-as em ocupadas ou vazias. Já a Representação Contínua utiliza retas para representar os planos detectados, formando um mapa em duas dimensões. Cada reta do mapa é considerada um marco. A posição do robô é atualizada via Filtro de Kalman Estendido (FKE). Comprovou-se a eficácia do modelo em uma simulação computacional em um ambiente virtual. Utilizando um protótipo desenvolvido com a plataforma LEGO Mindstorms foi realizado ainda outro experimento em um ambiente real. Os resultados obtidos comprovaram a eficácia e a convergência do modelo proposto.

Palavras-chave— Robótica Móvel, SLAM, Sonar, Filtro de Kalman Estendido

1 Introdução

Simultaneous Localization and Mapping ou SLAM, nome dado ao processo de mapear o ambiente e localizar-se no mesmo simultaneamente, é considerado um dos maiores desafios para a robótica móvel (Thrun, 2005).

Neste trabalho descreve-se um protótipo de robô de baixo custo, equipado com sonares, bússola eletrônica e *encoders* que é capaz de localizar-se e mapear o ambiente simultaneamente.

Utilizando os sonares o robô extrai pontos do ambiente ao mesmo tempo em que o classifica como ocupado ou vazio. Os pontos extraídos dão origem a retas e consecutivamente a um mapa linear, enquanto o mapeamento de áreas vazias ou ocupadas dá origem a um mapa de ocupação em grade.

Utilizando a informação contida nos dois mapas e sua localização estimada via odometria, com o auxílio da bússola, o robô é capaz de localizar-se simultaneamente.

Para verificar o desempenho do robô foram realizados dois experimentos, um em ambiente simulado e outro em ambiente real.

Este trabalho está dividido com como se segue: a seção dois contém uma breve revisão teórica de alguns dos principais métodos utilizados. Na seção três são apresentados alguns dos trabalhos relacionados.

A seção quatro apresentará o algoritmo LMS-H (Localização e Mapeamento Simultâneo Híbrido), seguidos dos resultados obtidos na seção seis e conclusão na seção sete.

2 Revisão Teórica

2.1 Representação do Ambiente

A representação do ambiente é influenciada pela forma como o robô modela o mundo ao seu redor e como ele “enxerga” os marcos que irá utilizar para sua navegação. Segundo Siegwart & Nourbakhsh (2004) dentre as diversas formas de representação do ambiente duas se destacam: Representação Contínua e Decomposição em Grade.

Na representação contínua o ambiente é descrito por primitivas geométricas como: retas, círculos, quadrados ou outros polígonos. A Figura 1 é um exemplo de ambiente mapeado apenas por retas.

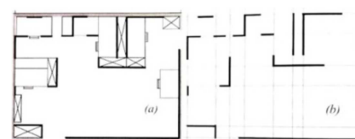


Figura 1 - Representação Contínua (Siegwart & Nourbakhsh, 2004).

Na representação em Grade o ambiente é dividido em partes iguais, classificadas como vazia ou ocupada. A Figura 2 é um exemplo deste tipo de representação, onde os quadrados em branco representam áreas livres e os quadrados em preto representam áreas ocupadas. As áreas em cinza podem ter ainda uma terceira classificação como áreas não exploradas.

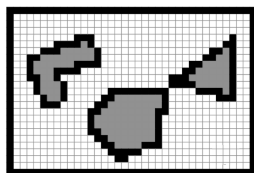


Figura 2 – Exemplo de representação por grade de ocupação.

2.2 Sonar

O SONAR (*S*ONd *N*avigation *A*nd *R*anging) é um dispositivo formado por um transmissor e um receptor de ondas sonoras de alta frequência, cujo princípio de funcionamento é baseado no tempo que o som leva para percorrer o espaço entre o emissor do pulso sonoro e o obstáculo (Ulrich Nehmzow, 2003).

O sonar, todavia, apresenta erros de leitura quando submetidos a situações específicas conhecidas como *specular reflections*. Estas ocorrem quando o som emitido pelo sonar não retorna ao sonar ou quando o som não retorna diretamente para o sonar, dando a impressão de uma área livre à frente que não existe, como mostrado na Figura 3a e Figura 3b, respectivamente.



a) O som não retorna ao sonar. b) O som não retorna diretamente para o sonar

Figura 3 – *Specular Reflections*. Adaptado de (Specifications, 2009).

2.2.1. Extração de pontos do sonar

Não é possível saber exatamente o ponto de contato do som com o obstáculo quando se utiliza sonares, admite-se, portanto, que o ponto detectado está em qualquer lugar ao longo do semicírculo da extremidade do cone do sonar (Figura 4).

Supondo um sonar centrado em um sistema de coordenadas cartesiano, como mostrado na Figura 4, com o seu eixo principal fazendo 90° com o eixo X do sistema de coordenadas, alinhado com o eixo y do sistema de coordenadas do mundo, as coordenadas de um ponto $^S P = ({}^S x, {}^S y)$ qualquer localizado no semi círculo que do sonar no sistema de coordenadas do sonar são dados pelas Equações (1) e (2), respectivamente.

$${}^S x_P = d \cdot \cos(\psi) \quad (1)$$

$${}^S y_P = d \cdot \sin(\psi) \quad (2)$$

Onde ψ é o ângulo que pode variar de $(\frac{\pi}{2} - \frac{\beta}{2})$ até $(\frac{\pi}{2} + \frac{\beta}{2})$ onde β é a abertura do cone do sonar. Usualmente utiliza-se $\psi = \frac{\pi}{2}$ por considerar-se que seja a região de maior probabilidade de ser o local onde o som foi refletido. Todavia, caso já se tenha alguma informação do ambiente, pode-se variar ψ de modo que o ponto detectado fique o mais próximo de um ponto já mapeado.

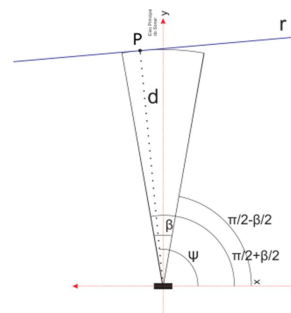


Figura 4 - Exemplo de reta que está no campo de visão do sonar.

2.3 Criação do mapa de ocupação em grade

Para compor o mapa de ocupação em grade, duas matrizes $M \times N$ são utilizadas: a primeira representa a região ocupada $Occ(x, y)$ e a segunda representa a região vazia $Emp(x, y)$. O cone do sonar é modelado por uma função que representa a confiança de uma célula está ocupada ou vazia.

Supondo que o sonar está no centro de um sistema de coordenadas polares e orientado na direção 0° , como mostrado na Figura 5, a região livre, mostrada em branco, pode ser representada pelo intervalo de $r = [d_{min}, d - \varepsilon]$ e $\psi = [\frac{\pi}{2} - \frac{\beta}{2}, \frac{\pi}{2} + \frac{\beta}{2}]$, onde d é a uma medição do sonar, d_{min} o alcance mínimo do sonar, β a abertura do cone do sonar e ε um valor de tolerância ou erro. Já a região ocupada, em cinza, é representada pelo intervalo de $r = [d - \varepsilon, d + \varepsilon]$ e $\psi = [\frac{\pi}{2} - \frac{\beta}{2}, \frac{\pi}{2} + \frac{\beta}{2}]$.

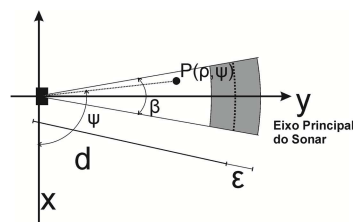


Figura 5 – Representação em 2D da leitura do sonar.

Assim, a funções P_E , que pondera as áreas vazias, e a função P_O , que pondera as áreas ocupadas, são dadas, respectivamente, pelas Equações (3) e (4).

$$P_E(r, \psi) = E_r(r)E_a(\psi) \quad (3)$$

$$P_O(r, \psi) = O_r(r)O_a(\psi) \quad (4)$$

Onde:

$$E_r(r) = \begin{cases} 1 - \left(\frac{r-d_{min}}{r-\varepsilon-d_{min}}\right)^2 & \text{se } d_{min} \leq r \leq d \\ 0 & \text{caso contrário} \end{cases} \quad (5)$$

$$E_r(\psi) = \begin{cases} 1 - \left(2 \cdot \frac{|\frac{\pi}{2} - \psi|}{\beta}\right)^2 & \text{se } \frac{\pi}{2} - \frac{\beta}{2} \leq \psi \leq \frac{\pi}{2} + \frac{\beta}{2} \\ 0 & \text{caso contrário} \end{cases} \quad (6)$$

$$O_r(r) = \begin{cases} 1 - \left(\frac{r-d}{\varepsilon}\right)^2 & \text{se } d - \varepsilon \leq r \leq d + \varepsilon \\ 0 & \text{caso contrário} \end{cases} \quad (7)$$

$$O_a(\psi) = \begin{cases} 1 - \left(2 \cdot \frac{|\frac{\pi}{2} - \psi|}{\beta}\right)^2 & \text{se } \frac{\pi}{2} - \frac{\beta}{2} \leq \psi \leq \frac{\pi}{2} + \frac{\beta}{2} \\ 0 & \text{caso contrário} \end{cases} \quad (8)$$

O perfil das funções de peso P_O e P_E podem ser vistas na Figura 6 e Figura 7, respectivamente, com o sonar posicionado em $P = (0,100)$, orientado na direção 90° e a uma distância $d = 150$ do obstáculo.

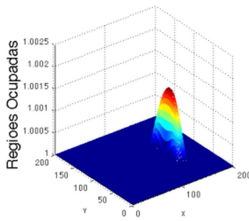


Figura 6 – Perfil da função de peso para regiões ocupadas, P_O .

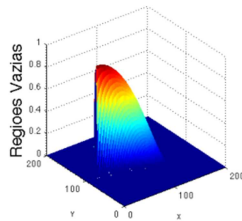


Figura 7 – Perfil da função peso para regiões vazias, P_E .

Para compor as matrizes Occ e Emp , soma-se a leitura atual com as informações anteriores, como segue:

$$Occ = P_O + Occ \quad (9)$$

$$Emp = P_E + Emp \quad (10)$$

As matrizes Occ e Emp acumularão as informações captadas pelos diferentes sonares em diferentes posições. Para reunir esta informação em apenas uma matriz, faz-se:

$$Map = \log_{10} \left(\frac{1 + Occ}{1 + Emp} \right) \quad (11)$$

Dessa forma, células em Map negativas possuem maior chance de serem regiões vazias e células positivas de serem regiões ocupadas. Células não visitadas possuem valor zero, como mostra a Figura 8. De forma heurística, um limiar de decisão dirá se a célula está ocupada ou vazia.

$$\begin{cases} Map_{ij} \geq \text{limiar}_{OCC} & = \text{Região Ocupada} \\ Map_{ij} \leq \text{limiar}_{EMP} & = \text{Região Vazia} \end{cases} \quad (12)$$

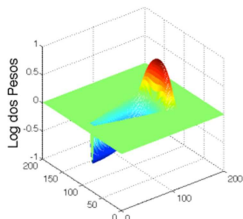


Figura 8 - Mapa logarítmico que engloba as regiões ocupadas e vazias.

2.4 Detecção de Retas

Uma reta r , já mapeada pelo robô, pode ser detectada se estiver tangente ao semicírculo da extremidade do cone do sonar, como mostrado na Figura 4.

A reta $r = (\rho, \alpha)$ descrita no espaço de Hough, onde: ρ é a menor distância entre o centro de sistema coordenadas e a reta r e α é o ângulo formado entre o eixo X do sistema de coordenadas e a reta que liga o centro do sistema de coordenadas ao ponto de menor distância da reta a ele.

Assim, é possível descrever um intervalo no qual a reta pode ser detectada pelo sonar, as inequações (13) e (14) formam este intervalo. .

$$\frac{\pi}{2} + \frac{\beta}{2} + \xi_\alpha \leq \alpha_1 \leq \frac{\pi}{2} - \frac{\beta}{2} - \xi_\alpha \quad (13)$$

$$d - \xi_\rho \leq \rho_1 \leq d + \xi_\rho \quad (14)$$

Onde ξ_ρ e ξ_α são margens de tolerância.

2.5 DBSCAN - Density Based Spatial Clustering of Application with Noise

O DBSCAN é um método de clusterização que procura por conjuntos densos seguidos de áreas de baixa densidade e possui como parâmetros de entrada:

- Raio da Vizinhança - ϕ ;
- Mínimo Número de Pontos - min_p ;
- Conjunto de Dados (pontos).

A saída é um conjunto k de clusters.

Um ponto q_1 qualquer é escolhido e, então, juntam-se ao ponto q_1 , todos os outros pontos que estão a um raio ϕ dele. Se a quantidade de pontos dentro deste subconjunto for maior ou igual a min_p , então um novo agrupamento é criado e o ponto q_1 é rotulado como ponto central. Pontos que pertencem a um subconjunto, mas não são pontos centrais são classificados como pontos fronteiras e pontos que não estão em nenhum agrupamento são considerados pontos *outliers*.

Na Figura 9, $\phi = 20 \text{ cm}$ e $min_p = 4$, temos:

- **Pontos Centrais:** pontos em preto;
- **Pontos Fronteiras:** ponto em cinza;
- **Pontos Outliers:** pontos em branco

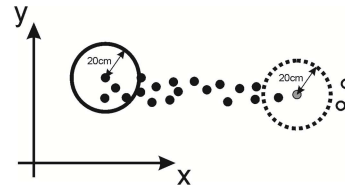


Figura 9 - Exemplo de clusterização pelo DBSCAN, $min_p = 4$.

Ao final, agrupamentos que compartilham pelo menos um ponto são unidos em um único agrupamento e os *Outliers* são descartados.

3 Trabalhos Relacionados

O SONAR, diferentemente do LRF (*Lazer Range Finder*) não é capaz de captar tanta informação como o LRF, enquanto cada leitura de um LRF possui milhares de pontos, o SONAR faz apenas uma leitura pouco precisa de onde realmente o som foi refletido. A abordagem utilizando sonares, por este motivo, deve ser bastante diferente quando comparado ao LRF.

Crowley (1989), em um dos trabalhos pioneiros na robótica móvel com sonar, extrai um ponto do sonar em cada leitura, usando os parâmetros $(x, y, \sigma_w, \sigma_d)$, onde x e y são as coordenadas do ponto, σ_d a incerteza correspondente à distância do ponto à reta e σ_w a incerteza relativa à posição do ponto dentro do semicírculo do sonar.

Os parâmetros σ_w e σ_d correspondem aos eixos de uma elipse centrada no ponto (x, y) , como mostrada na Figura 10. Entretanto, para facilitar os cálculos, foi considerado que σ_w e σ_d possuem o mesmo valor, extraído de forma experimental, e igual a:

$$\sigma_w = 0.10 + d \cdot \tan^{-1}(5^\circ) \quad (15)$$

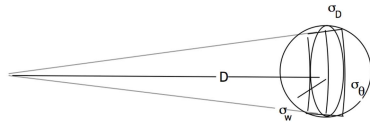


Figura 10 - Modelo da leitura de um sonar e sua incerteza (J L Crowley, 1989).

Utilizando regras bem definidas, três pontos são unidos para criar um seguimento de reta. A reta é criada apenas se os pontos estiverem suficientemente próximos (de forma que não seja possível que o robô passe por entre eles) e se a distância a um dos pontos for menor ou igual a um valor pré-definido ϵ . Um total de onze parâmetros são utilizados para modelar um segmento de reta. A grande quantidade e a redundância dos parâmetros são proposital e utilizados para facilitar a comparação entre as mesmas.

Meng, Sun, & Cao (2000) utilizaram quatro padrões no ambiente para localizar um robô equipado com oito sonares em um ambiente estruturado. Os quatro padrões possíveis são: planos, cantos convexos, cantos côncavos e cilindros. Essas características são traduzidas nas primitivas retas, pontos, pontos e círculos, respectivamente, e um Filtro de Kalman Estendido (FKE) é utilizado para localizar um robô em um ambiente estruturado.

Lee & Song (2010) desenvolveram uma forma bastante eficiente de mapear regiões salientes do ambiente, como: cantos convexos e cilindros. Através de um modelo denominado *Footprint Association (FPA)*, com duas ou mais leituras do sonar busca-se por uma circunferência que tangencia todos os semicírculos dos sonares, como mostrado na Figura 11. Um sistema de equações encontram os valores de q_0 , q_r e ϕ_2 a partir do valor de ϕ_1 . Sendo que o valor de ϕ_1 pode variar de ϕ_{1min} a $\phi_{1min} + \beta$, onde β é a

abertura do sonar. Como não é possível saber exatamente qual o valor de ϕ_1 , é feita uma busca exaustiva, subdividindo o ângulo do cone do sonar. Um círculo válido é aquele cujo valor do seu raio esteja em um intervalo de confiança $p_{min} \leq q_r \leq p_{max}$. Nos experimentos feitos os valores adotados foram 3 e 30cm para p_{min} e p_{max} , respectivamente.

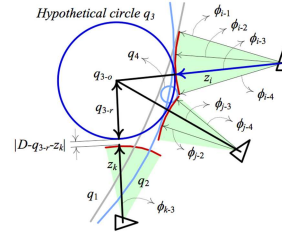


Figura 11 - Exemplo do modelo FPA, (Lee & Song, 2010).

Então, uma terceira leitura é usada e, se a distância entre o semicírculo do sonar z e a extremidade do círculo de raio q_r , dado por $(|D - q_r - z_k|)$, for menor que um limiar (2cm por exemplo), então este círculo é dito como um marco e é gravado no mapa para referência. No exemplo da Figura 11 apenas o círculo de raio q_{3-r} satisfaz todas as restrições. Através destes marcos um Filtro de Kalman Estendido é usado para localização do robô.

Armazenar primitivas como retas, pontos e círculos consomem pouca memória. Além disso, estas primitivas permitem o uso de filtros probabilísticos, como o Filtro de Kalman.

A despeito da pouca memória consumida, se a quantidade de marcos no ambiente for muito grande, outros problemas deverão ser enfrentados. Quando o robô detectar algum marco é imprescindível que o mesmo saiba a qual marco ele está se referindo e, caso a quantidade de primitivas contidas no mapa seja muito grande, fazer a correspondência entre elas será uma tarefa computacionalmente custosa. Ameniza-se o problema diminuindo o espaço de busca, de acordo com a pose estimada, e os possíveis marcos ao alcance do robô.

4 Localização e Mapeamento Simultâneo Híbrido – LMS-H

O modelo desenvolvido neste trabalho, exibido em forma de blocos na Figura 12, foi projetado para um protótipo de robô equipado com *encoders*, sonares e bússola de forma que o robô possa mapear e localizar-se simultaneamente em ambientes que contenham apenas planos.

No bloco “Atualização da Grade de Ocupação” o mapa de ocupação em grade é atualizado de acordo com o modelo apresentado na seção 2.3.

Se os sonares não detectarem nenhuma reta já mapeada, as leituras serão utilizadas para extrair pontos de acordo com o modelo do sonar apresentado na seção 2.2.1. Os pontos extraídos desta forma são armazenados em *pull* geral de pontos.

O bloco de “Filtragem” seleciona os pontos do *pull* geral que estão em áreas consideradas ocupadas, de acordo com o mapa de ocupação em grade, colocando-os em outro *pull* de pontos de locais considerados ocupados.

No bloco de “Clusterização” utiliza-se o DBSCAN para clusterizar os pontos do *pull* de pontos em locais ocupados. Os pontos de cada cluster identificado são transformados em uma reta através da técnica de mínimos quadrados, as retas identificadas dessa forma farão parte de um *pull* de retas, que, na prática, será a representação contínua do mapa.

Caso alguma reta seja detectada com sucesso, como mostrado na seção 2.4, utiliza-se a parte posteriori do Filtro de Kalman estendido para aprimorar a localização do robô. Adiciona-se também o ponto de contato *pc*, mostrado na Figura 4, no cluster que forma aquela reta, e então, a reta é atualizada.

Em todas as transições de estado utiliza-se a parte a priori do Filtro de Kalman Estendido para calcular a posição do robô no tempo *t*.

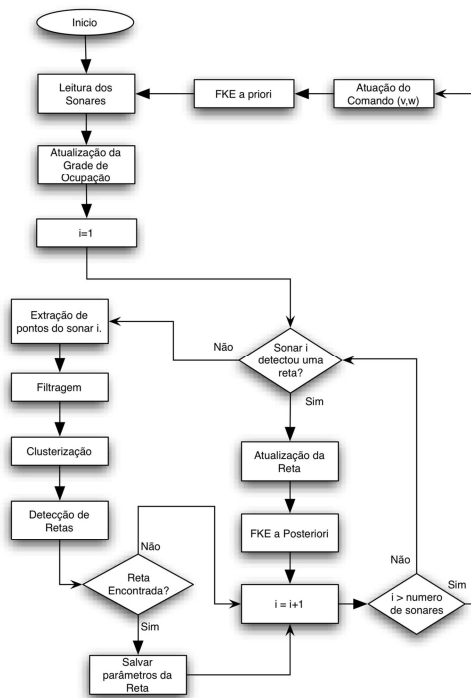


Figura 12 - Algoritmo do LMS-H

5 Resultados

Para comprovar o funcionamento do método, dois experimentos serão apresentados: uma simulação computacional e um ensaio em ambiente real.

O robô, tanto simulado como real, possui as seguintes características: 7 sonares com alcance máximo e mínimo de 200cm e 20cm respectivamente, abertura do cone de 20° e resolução de 1°; bússola de 1° de resolução; drive diferencial e *encoders* de 1° de resolução.

A Figura 13 mostra como estão dispostos os sonares no robô, tanto no protótipo quanto no robô utilizado nas simulações.

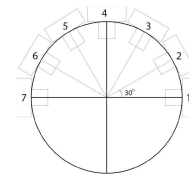


Figura 13 - Modelo do robô.

Tanto no experimento simulado quanto nos experimentos com o protótipo, os parâmetros utilizados pelo LMS-H foram

- **DBSCAN:** mínimo número de pontos em um cluster $\min_p = 5$, raio da vizinhança $\phi = 20\text{cm}$ e erro médio quadrado máximo da reta $\epsilon = 5$;
- **Configurações do Sonar:** Abertura do Cone do Sonar $\beta = 20^\circ$, alcance mínimo $d_{min} = 20\text{ cm}$, alcance máximo $d_{max} = 200\text{ cm}$, erro máximo do sonar $\epsilon = 5\text{ cm}$, n° de pontos extraídos por leituras=1;
- **Configurações do Mapa de Ocupação em Grade:** Tamanho da célula=1, Limiar superior – células ocupadas = 0,5, Limiar inferior – células vazias= 0,5.
- **Configuração de Deteção de Retas:** Erro máximo na coordenada ρ da reta quando detectado pelo sonar no sistema de coordenadas de Hough $\xi_\rho = 20\text{ cm}$, Erro máximo na coordenada α da reta quando detectado pelo sonar no sistema de coordenadas de Hough $\xi_\alpha = 5^\circ$.

$$R = \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,01 \end{bmatrix} \quad Q = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 100 \end{bmatrix}$$

$\sigma_b = 0,001$

Onde: R é a covariância de deslocamento, Q covariância da observação e σ_b a covariância da leitura da bússola.

5.1 Ambiente Simulado

O ambiente simulado pode ser visto na Figura 14. O robô percorre o ambiente passando pelos objetivos na ordem em que estão numerados, em um total de três voltas.

As Figura 15 e Figura 16 mostram o resultado do mapeamento contínuo e em grade de ocupação respectivamente. É possível notar o que o ambiente foi, em grande parte, mapeado. É possível observar como as áreas vazias possuem pesos baixos, enquanto as áreas ocupadas possuem pesos mais elevados.

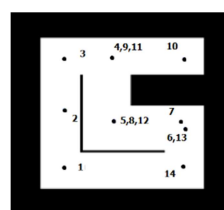


Figura 14 - Ambiente Simulado.

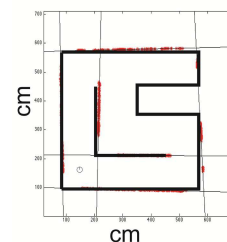


Figura 15 - Resultado do mapeamento via LMS-H no ambiente simulado.

A Figura 16 e Figura 17 mostram os erros nos eixos x , y e na orientação θ . As curvas em mais grossas são as estimações via LMS-H, já as curvas mais finas são as estimações via odometria. É possível observar como o erro do robô estimado via odometria é sempre crescente, enquanto o erro do robô estimado via LMS-H tende a oscilar próximo do zero.

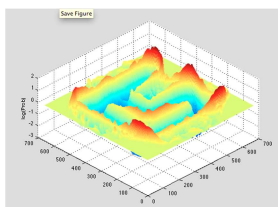


Figura 16 - Mapa de ocupação em grade do ambiente simulado.

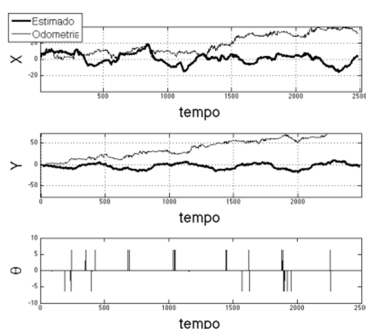


Figura 17 - Erros nos eixos X e Y e na orientação do robô.

5.2 Ambiente Real

O ambiente real consiste de um vão em “L”, Figura 18. Neste ambiente o robô foi controlado manualmente e descreveu a trajetória numerada. O mapeamento, depois de uma volta, pode ser visto na Figura 19 e 20. Novamente, é possível notar que as áreas ocupadas estão com pesos positivos e áreas desocupadas estão com pesos negativos.

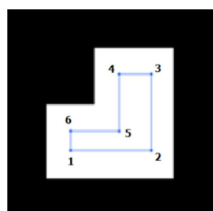


Figura 18 - Ambiente simulado e a trajetória com os respectivos objetivos.

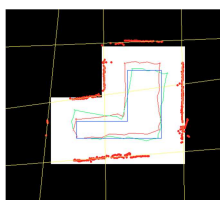


Figura 19 - Resultado do mapeamento via LMS-H no ambiente real.

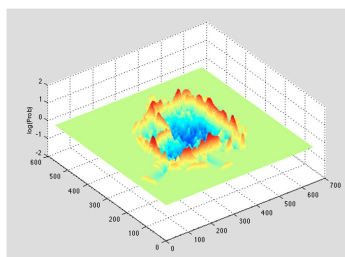


Figura 20 - Grade de ocupação para ambiente real em “L”

Como não foi possível rastrear o robô durante os testes, não foi possível gerar um gráfico de erros nos

eixos X , Y e da orientação θ . Entretanto, a partir da Figura 19, é possível observar que, no final da trajetória, a posição do robô estimada via odometria está bem distante do robô cuja posição foi estimada via LMS-H, mesmo que em todo o resto do trajeto a estimativa via odometria esteja mais próxima da trajetória real.

6 Conclusão

Neste trabalho foi apresentado um protótipo equipado com sonares, bússola e *encoders* capaz de realizar o mapeamento e localização simultâneos via FKE.

Foi desenvolvido um método de detecção e atualização de retas, que permite usá-las como marco natural no processo de localização e mapeamento simultâneos. Foram utilizadas duas formas de representação do mundo: representação contínua e em grade de ocupação. O mapa de ocupação em grade viabilizou a criação de um filtro para a separação das leituras válidas e errôneas do sonar. O LMS-H obteve o desempenho esperado, dentro das limitações que o hardware impõe. O teste em um ambiente real mostrou a eficiência do LMS-H, dada a grande quantidade de ruídos dos sensores, principalmente do sonar. Por fim, o modelo proposto mostrou-se bastante promissor, apresentando bons resultados.

7 Agradecimentos

Ao CNPq, Faperj e a PUC-Rio pelo financiamento da pesquisa e aos auxílios concedidos sem os quais essa pesquisa seria inviável.

8 Referência Bibliográfica

- Crowley, J. L. (1989). World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging. *Robotics* (Vol. 3, pp. 674–680). IEEE Press. doi:10.1109/ROBOT.1989.100062
- Lee, S., & Song, J. (2010). A New Sonar Salient Feature Structure for EKF-Based SLAM. *Robotics*, 5966–5971. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5650169
- Meng, Q., Sun, Y., & Cao, Z. (2000). Adaptive extended Kalman filter (AEKF)-based mobile robot localization using sonar. *Robotica*, 18, 459–473.
- Siegwart, R., & Nourbakhsh, I. R. (2004). *Introduction to autonomous mobile robots*. (R. C. Arkin, Ed.) (p. 321). The MIT Press. Retrieved from http://books.google.com/books?hl=en&lr=&id=gUbQ9_weg88C&pgis=1
- SThrun, S. (2005). *Probabilistic Robotics*. (S. Thrun, W. Burgard, & D. Fox, Eds.) *Communications of the ACM* (Vol. 45, pp. 1999–2000). MIT Press. doi:10.1145/504729.504754

Ulrich Nehmzow. (2003). *Mobile Robotics A Practical Introduction* (2nd ed.). Springer.