

Uma proposta de SLAM para robótica subaquática

Felipe A. Guth and Silvia S. C. Botelho and Paulo Drews Jr¹

Abstract—The solution for the Simultaneous Localization and Mapping (SLAM) problem is considered a key prerequisite for making a robot fully autonomous. Although the SLAM problem has solutions that solve the problem almost completely for outdoors environments, it remains an open problem in large environments and especially in underwater environments. Among the many difficulties encountered in underwater environments, we can highlight the inability to use GPS, the participatory characteristics of the water, inaccuracy of sensors, dynamic of the environment and currents, high pressure and the restrict use of artificial lights. This paper proposes a SLAM approach using multi-sensorial inputs with focus in subaquatic high depth environments with manmade structures found in areas of oil and gas extraction. The methodology proposes the use of an Extended Kalman Filter (EKF) to maintain robot position and landmarks estimations along a Hough transform technique to extract line features of the acoustic data provided by a mechanically scanned imaging sonar.

Keywords: Underwater SLAM, Multi-sensorial, EKF.

I. INTRODUÇÃO

Dentro dos vários problemas existentes na robótica, um dos mais importantes refere-se à possibilidade de um robô móvel mapear e estimar a sua localização no ambiente navegado. O mapeamento robótico tem sido uma área bastante explorada em robótica e inteligência artificial há pelo menos duas décadas. A técnica de SLAM, (Localização e Mapeamento Simultâneos) é usada para construir mapas em ambientes desconhecidos, ou para atualizar um mapa em um ambiente antes percorrido, mantendo sua localização corrente no mapa que é simultaneamente construído, este é um dos problemas mais difíceis encontrados na robótica. Embora existam técnicas de SLAM bem estabelecidas que obtenham resultados satisfatórios, pode-se dizer que o mapeamento e localização simultâneos em ambientes de grande extensão é ainda, um problema não resolvido. Isso devido ao grande custo computacional para com as leituras de dados massivos advindos de diversas fontes sensoriais e a imprecisão dos dados obtidos através dos sensores.

Neste contexto, abordagens para a realização de SLAM subaquático encontra-se ainda, em uma fase inicial e um limitado número de abordagens tem sido apresentado. O meio subaquático apresenta muitas dificuldades como a impossibilidade de se usar sinal de GPS, a dificuldade de imageamento devida turbidez da água, a influência de correntes marítimas e do próprio movimento dos propulsores. Esses fatores dificultam a modelagem do comportamento dinâmico do robô e seus atuadores para aplicação de técnicas de localização e mapeamento associadas a presença do meio

aquoso, o qual, interfere na capacidade de aquisição de informações e atuação dos seus sensores e atuadores.

No contexto histórico de SLAM em robótica subaquática os trabalhos de [1], foram precursores, com o uso de dados adquiridos através de um cargueiro da Marinha Americanana. O uso de um dos primeiros robôs para tarefa de SLAM ocorreu com o trabalho de [3] para monitoramento de corais. Já [4] executou testes tanto em tanques de água como em oceano aberto usando dados apenas de sonares para performance de SLAM. O trabalho de [5] propôs uma nova técnica no tratamento de incertezas de dados de um sonar mecânico obtidos em um ambiente estruturado. Recentemente [6] propôs uma nova abordagem através do uso de peixes robóticos para mapeamento subaquático através de múltiplos robôs.

Este trabalho apresenta uma proposta para o problema de SLAM em ambientes subaquáticos através da utilização de diversas fontes sensoriais. A abordagem proposta trabalha com extração de *features* de sonar em formas de linhas usando transformada de Hough e usa o Filtro de Kalman Estendido para fornecer uma estimativa de posição do robô e dos marcos do ambiente, resolvendo o problema de SLAM *online*. Os resultados parciais apresentados correspondem aos primeiros testes realizados usando apenas dados de sensores inerciais. Na próxima fase do trabalho, além de integrar dados do sonar, será realizado a integração de *features* de imagem no sistema e o uso de técnicas de *bag of words* para associação de dados.

O artigo tem continuidade com a apresentação do problema de SLAM e consequente relação de trabalhos na área de SLAM aplicados a ambientes subaquáticos. A seguir é apresentada a metodologia do trabalho, abordando a atual proposta para resolução do problema de SLAM subaquático, bem como a explicação das principais técnicas e sensores utilizados. Por fim são apresentados os primeiros resultados da aplicação e as conclusões atuais do trabalho que ainda encontra-se em desenvolvimento.

II. O PROBLEMA DE SLAM

Simultaneous Localization and Mapping (SLAM) é um termo que surgiu na conferência de robótica em São Francisco no ano de 1986 [7], tempo onde métodos probabilísticos estavam apenas começando a ser introduzidos na robótica e Inteligência Artificial em geral. Ao longo dos anos conseguintes Smith e Cheesman [8] e Durrant-Whyte [9] estabeleceram a base estatística para descrever relações entre pontos de referência e para manipulação de incerteza geométrica. O problema de SLAM também é designado como Concurrent Mapping and Localization (CML) por alguns autores.

OS autores agradecem o apoio de CAPES, CNPQ e FINEP.

¹NAUTEC - Centro de Ciências Computacionais C3, Universidade Federal do Rio Grande, Rio Grande, Brasil felipeguth@furg.br

A localização de um robô móvel é o problema de determinar a pose de um robô relativa a um dado mapa. A busca por técnicas de localização de um robô móvel é uma instância do problema geral de localização, o qual apresenta como desafio básico tratar a capacidade de robôs perceberem o ambiente [10]. A localização de um robô móvel é um fator essencial para dotar um veículo de características autônomas.

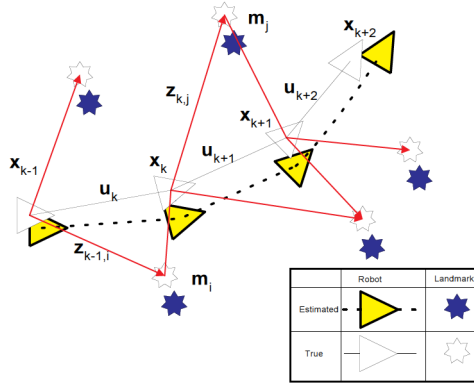


Fig. 1. [Durrant-Whyte and Bailey, 2006]. O problema essencial de SLAM. Um estimativa simultânea tanto do robô como das referências é necessária. As conexões reais do ambiente nunca são conhecidas ou medidas diretamente. Observações são feitas entre robôs e localização de referências.

Devido a dificuldade de modelagem e as restrições de operação dos sensores, a informação resultante da sua operação tem associada uma componente de incerteza, a qual acarreta ao longo do tempo a possibilidade de existência de erros de estimação associados a movimentação do robô. Assim, prejudicando tanto o mapeamento quanto a localização, uma vez que as duas tarefas são correlacionadas. Como resultado o veículo eventualmente irá manter uma representação errônea e se perderá. As técnicas de SLAM são aplicadas para atenuar esse efeito. A observação de pontos de referência já existentes no mapa são processadas junto a um algoritmo de associação de dados levando a atualização das estimativas de posição e mapa. Assim sendo, quando um lugar é revisitado pode-se melhorar as representações através da diminuição da incerteza de localização.

III. TRABALHOS RELACIONADOS

Uma das primeiras implementações de SLAM com veículos marinhos usando dados reais de sensores ocorreu no ano de 1997, quando um cargueiro da Marinha Americana foi utilizado para adquirir uma base de dados de sensores como sonares, INS, DVL e DGPS. Os dados adquiridos foram usados para o teste em duas abordagens de SLAM. O primeiro [11] apresentou uma implementação simplificada utilizando EKF, onde filtros independentes são iniciados para cada novo ponto de referência encontrado no mapa, essa abordagem não possui estimativa de incerteza e a correlação entre os marcos do ambiente é ignorada. Na segunda implementação em parceria com os pesquisadores do MIT [1] foi apresentado um EKF utilizando os dados dos sensores de *dead-reckoning* para melhorar a estimação do estado, além de produzir um mapa estocástico completo, com representação de incerteza.

Nesta implementação correlações dos marcos foram levadas em conta como também a representação de incerteza. Marcos em formato de ponto extraídos dos centroides dos dados acústicos do sonar foram utilizados para construir o mapa. O processo de associação de dados foi realizado através da aplicação de um algoritmo de *Nearest Neighbor* (NN).

Outro trabalho introdutório na área de SLAM underwater foi realizado por [2] onde uma abordagem de SLAM para larga escala foi apresentada utilizando Mapeamento Estocástico Dissociado (DSM). Essa proposta reduz o custo computacional dividindo o ambiente em múltiplos mapas sobrepostos representando uma dada região, cada sua com seu mapa estocástico.

Experimentos foram executados em recifes e corais da barra de Sidney, Austrália [3] usando uma implementação clássica de EKF junto a um mapa estocástico. Marcos artificiais foram introduzidos no ambiente a fim de produzir um conjunto de *features* de pontos confiáveis e mesmo alguns marcos naturais sendo detectados, muitos foram descartados por serem muito instáveis para serem incorporados ao mapa.

Em [4] uma implementação clássica de EKF com mapa estocástico foi realizada com dados reais obtidos em La Spezia, Italia, com o AUV Remus. Nesta versão foi utilizado o filtro Rauch-Tung-Striebel (RTS) para suavizar o caminho percorrido usando dados de um sonar do tipo sidescan. As *features* do mapa foram extraídas manualmente, da mesma forma foi realizada a associação de dados. Mais tarde esse trabalho foi estendido, definindo um processo automático para extração de *features*, usando Campo Randômico de Markov (MRF) [12] e associação de dados com um algoritmo de teste de compatibilidade de ligações (JC) [13].

Utilizando filtro de partículas surgiu em 2007 um novo AUV [15], [16] com uma suíte sensorial com 56 transdutores de sonar além de sensores inerciais. O núcleo de SLAM nesse trabalho é um filtro de partículas Rao-Blackwellized onde cada partícula representa a posição do veículo e o mapa [17].

Em 2008 [18] apresentou um sistema de EKF SLAM para ambientes parcialmente estruturados, utilizando um novo AUV para a criação de uma base de dados tendo como sensor principal um sonar mecânico, além de dados de sensores inerciais e de DGPS para *ground truth* [5]. Foi proposta uma nova abordagem para extração de *features* em formato de linhas como para representação da incerteza dos dados acústicos do sonar, construindo múltiplos mapas locais ao longo da trajetória. [19] apresentou uma aplicação de SLAM para AUV em tempo real, demonstrando uma nova solução para fusão de dados onde sete diferentes fontes sensoriais foram fundidas em diferentes escala de tempo usando cinco estimadores diferentes com múltiplas aplicações de EKF. [20] expôs uma nova proposta usando visão computacional com descritores de *features* SIFT para extração em sequências de imagens subaquáticas e o desenvolvimento de mapas topológicos baseados em redes de Kohonen visando as tarefas de localização e mapeamento. A integração dos dois módulos permite simultaneamente localizar o robô e mapear o ambiente.

[23], [24] propôs uma solução baseada em poses para

resolver o problema de SLAM completo (*full SLAM*) para navegação de um AUV em ambiente parcialmente estruturado. A solução usa dois EKF's para estimativa de posição e registro de poses, utilizando dados de um sonar mecânico como sensores de *dead-reckoning*. O algoritmo foi testado em AUV percorrendo um caminho de 600m em uma marina, confirmando a viabilidade do projeto. Também no mesmo ano [25] apresentou uma solução de SLAM aplicada a inspeção de casco de um porta-aviões de 30x5m usando uma câmera monocular. Para detecção de *features* foram usados extratores SIFT e Harris. Para manutenção e atualização do estado e mapa, foi usado um EKF.

[26] apresentou uma nova abordagem de SLAM com EKF usando sonar mecânico (MSIS). Esta começa processando os dados do MSIS, obtendo os dados de distância do scan enquanto considera também a movimentação do robô. Então as relativas movimentações entre scans são armazenadas no vetor de estados, esses estados são usados para fazer o SLAM com EKF. A novidade é que essa sequência não é representada com respeito a um fator de coordenadas fixado no mundo, mas sim por um fator de coordenadas fixas com respeito ao robô. Com isso, os erros de linearização do EKF são reduzidos.

Aulinas e colegas [27] apresentaram uma implementação de SLAM para o AUV SPARUS apresentando uma abordagem de sub mapas. O principal sensor dessa aplicação foi uma câmera apontada para baixo. Experimentos mostraram uma limitação dos erros de linearização graças ao uso de sub-mapas, estimativa de trajetórias precisa e reconstrução tridimensional do mapa. Além disso, o modelo de observação para uma câmera apontada para baixo foi introduzida. Experimentos realizados em ambientes reais estruturados demonstraram que SLAM melhorou a trajetória em relação aos sensores de *dead-reckoning*. Além do mais, SLAM combinado com parametrização inversa de profundidade foi capaz de produzir um mapa consistente.

[6] propôs uma nova abordagem para localização e mapeamento 3D de um ambiente utilizando um cardume de peixes robóticos subaquático com conexão sem fio. É baseado em localização cooperativa usando filtro de partículas (CLPF) e um grid de ocupação. Usando um *framework* probabilístico, a formulação detalhada de CLPF é apresentada, incluindo a inicialização, a previsão, atualização e reamostragem. O resultado da localização é então utilizado para mapeamento 3D no grid de ocupação onde os peixes robóticos compartilham o mesmo conjunto de células no grid. Experiências por meio de simulação foram realizadas para avaliar o desempenho dos algoritmos propostos e comprovaram o potencial da solução.

IV. METODOLOGIA

Esse trabalho tem como objetivo principal desenvolver uma técnica de SLAM usando Filtro de Kalman Estendido, e usar dados de múltiplos sensores, aplicada a localização e mapeamento de ambientes na robótica subaquática.

O sistema opera em dois módulos simultaneamente, o primeiro busca continuamente por *features* do ambiente, através de dados obtidos por um sonar mecânico. Outro

módulo consiste na implementação do Filtro de Kalman Estendido que continuamente funde informações de sensores (DVL, bússola, MRU) e as observações do módulo de extração de características para manter a estimativa de posição do robô e construir simultaneamente uma mapa. No trabalho é utilizado o data set de [5] o qual foi obtido em uma marina abandonada. Este ambiente em particular é composto por várias paredes. Assim sendo, o algoritmo de extração de características (*features*) faz busca continuamente por linhas, representativas das paredes, através de uma abordagem que implementa a transformada de Hough por meio de um esquema de votos, identificando as *features* que têm maior probabilidade de representarem elementos no ambiente. A Figura 2, descreve em alto nível a abordagem de SLAM proposta.

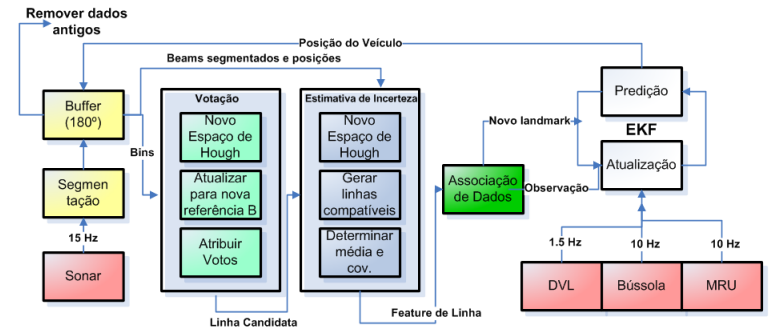


Fig. 2. Metodologia de SLAM proposta.

A. Sensores utilizados

A seguir apresentam-se de forma sucinta os sonares utilizados atualmente conforme a metodologia proposta.

Sonar Mecânico de Imagem - MSIS: Um sonar mecânico de imagem (MSIS) trabalha obtendo scans 2D do ambiente, em um determinado setor que pode variar de alguns graus para uma leitura completa de 360°. O sonar produz uma onda acústica em formato de leque para cada ângulo, quando a onda encontra um objeto, parte da energia é refletida de volta ao transdutor. Medindo o tempo de viagem da onda, assumindo um valor conhecido para velocidade do som na água, permite determinar a distância a qual o sinal foi originado.

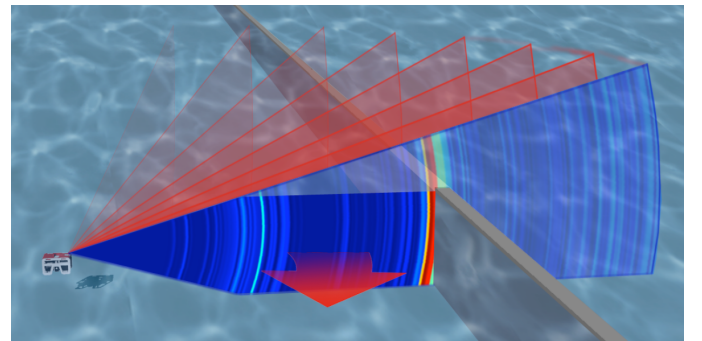


Fig. 3. Representação de um processo de scan do MSIS [18]

Analisando-se o sinal por um período de tempo permite a produção de vários valores de medida de amplitude versus

distância. Esses valores são designados com *bin*, enquanto o conjunto de *bins* obtidos para uma única onda emitida do sonar é denominada como *beam*. Portanto a emissão de um *beam* é composta por um conjunto de *bins*, cada um representando um determinado valor de intensidade para um lugar específico ao longo do eixo do transdutor.

DVL: O sensor de DVL (*Doppler Velocity Log*) é um sensor altamente importante capaz de aferir a velocidade do sensor em relação ao ambiente, usando o efeito doppler de partículas suspensas na água. Esse sensor possui um erro de medição semelhante ao encontrado em sensores de odometria de robôs terrestres. Geralmente pode fornecer tanto medidas de velocidade em relação à água, que é prejudicada consideravelmente em caso de ambientes com correnteza, quanto medidas de velocidade em relação ao fundo, que podem ser prejudicadas por reverbações em ambientes de baixa profundidades ou relevos muito irregulares no fundo do mar.

MRU: Uma Unidade de Referência de Movimento é outro sensor frequentemente utilizado em aplicações subaquáticas, este fornece orientação em 3D (atitude e orientação), taxa de giros em 3D como também medidas de aceleração em 3D.

B. Filtro de Kalman Estendido - EKF

O Filtro de Kalman Estendido trata de sistemas não lineares, conforme a grande maioria dos problemas tratados pelo SLAM. O estado pode ser representado por um vetor x e uma matriz de covariância P que representa a incerteza do estado, marcos e suas correlações. O Filtro de Kalman Estendido consiste de uma solução recursiva através de passos de predição onde o estado e sua incerteza é propagado no tempo de acordo com seu modelo de movimento, e correção onde o estado e sua incerteza são atualizados com os dados aferidos pelos sensores e seu modelo de observação, no final do processo é fornecido uma nova estimativa do estado e sua matriz de covariância que servem de entrada para a próxima iteração [10].

Em aplicações de SLAM com EKF utiliza-se uma abordagem onde o vetor de estados e matriz de covariância são expandidos quando encontra-se uma novo marco no ambiente. Assim para cada tempo k o sistema representa a corrente localização do veículo e dos marcos, bem como suas incertezas e correlações. Adicional a isso, é necessário um módulo de associação de dados, para que quando um marco do ambiente for detectado, se identifique que o mesmo representa uma elemento já contido no estado ou se representa uma novo elemento, se assim for, o vetor de estado e matriz de covariância devem ser expandidos para alocar uma nova representação.

A Figura 4 apresenta de forma sucinta o funcionamento recursivo do Filtro Estendido de Kalman.

1) Inicialização do sistema: Quando inicializa um novo mapa estocástico, deve-se selecionar uma base de referência L no passo 0. Define-se assim a posição inicial do veículo para localização desta base no tempo 0 com completo conhecimento e sem incerteza do vetor de estado. Pode-se representar o estado do veículo x_V :

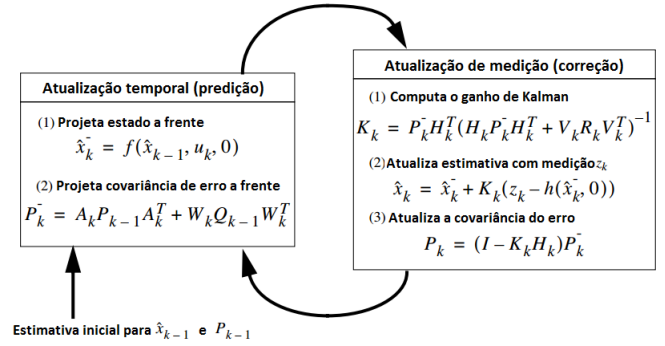


Fig. 4. Ciclo sequencial do Filtro Estendido de Kalman.

$$x_V = [x \ y \ z \ \psi \ u \ v \ w \ r \ \psi L_0]^T \quad (1)$$

O termo ψL_0 funciona como um bias para representar a orientação inicial do veículo no tempo 0 em relação ao norte magnético da terra aferido pela bússola. Isso torna possível o uso da bússola para determinação da orientação do veículo. Assumindo que ainda não existem marcos (*landmarks*) no estado, e que o mesmo encontra-se em uma pose estacionária, o estado do veículo no tempo 0 é igual a:

$$\begin{aligned} \hat{x}(0) &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \hat{\psi} L_0]^T; \\ P(0) &= P_V(0) = \begin{bmatrix} 0_{8 \times 8} & 0_{8 \times 1} \\ 0_{1 \times 8} & \sigma_{\psi L_0}^2 \end{bmatrix} \end{aligned} \quad (2)$$

onde $\hat{\psi} L_0$ tem o valor da primeira medida da bússola e $\sigma_{\psi L_0}^2$ é inicializado conforme a precisão do sensor.

2) Predição: O vetor de estados e a predição do estado completo para um robô AUV usando um Filtro de Kalman Estendido em um tempo k pode ser representada como:

$$x(k) = f(x(k-1), s(k-1)),$$

$$\begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \\ \psi L_0 \\ p \\ \theta \end{bmatrix}_{(k)} = \begin{bmatrix} x + (uT + s_u T^2/2) \cos(\psi) \\ - (vT + s_v T^2/2) \sin(\psi) \\ y + (uT + s_u T^2/2) \sin(\psi) \\ + (vT + s_v T^2/2) \cos(\psi) \\ z + wT + s_w T^2/2 \\ \psi + rT + s_r T^2/2 \\ u + s_u T \\ v + s_v T \\ w + s_w T \\ r + s_r T \\ \psi L_0 \\ 0 \\ 0 \end{bmatrix}_{(k-1)} \quad (3)$$

Onde p e θ representam parâmetros de um marco (*landmark*). Como os dados do sonar são referenciados em polar, o tamanho do vetor de estados final é igual a 9 ($x, y, z, \psi, u, v, w, r$) na discretização dos eixos, orientação *yaw* e correspondentes velocidades lineares e angulares mais $2 \times n$ marcos do ambiente, uma vez que cada marco é representado por θ^B e p^B .

A variável $s = [s_u \ s_v \ s_w \ s_r]^T$ representa um ruído branco de aceleração aditivo a velocidade com média zero e covariância Q . Em relação as *features*, presume-se que as mesmas são estáticas e portanto sua localização não possui modificações no passo de predição, elas tendem a permanecer no mesmo lugar. Assim sendo, todo estado pode ser predito como:

$$\hat{x}(k|k-1) = [f(\hat{x}_V(k-1)) \ \hat{x}_1(k-1) \dots \hat{x}_n(k-1)]^T \quad (4)$$

E a matriz de covariância atualizada como:

$$P(k|k-1) = \begin{bmatrix} F_v(k) & 0_{9 \times 2n} \\ 0_{2n \times 9} & I_{2n \times 2n} \end{bmatrix} P(k-1) + \begin{bmatrix} F_v(k) & 0_{9 \times 2n} \\ 0_{2n \times 9} & I_{2n \times 2n} \end{bmatrix}^T + \begin{bmatrix} W_v(k) \\ 0_{2n \times 4} \end{bmatrix} Q \begin{bmatrix} W_v(k) \\ 0_{2n \times 4} \end{bmatrix}^T \quad (5)$$

Onde F_V e W_V são as matrizes Jacobianas de derivadas parciais do modelo não-linear da função f com respeito ao mesmo estado x_V e ao ruído n respectivamente:

$$F_V = \frac{\partial f}{\partial x_V}(\hat{x} + V(k), 0) = \begin{bmatrix} 1 & 0 & 0 & -\hat{u}T \sin \psi & -\hat{v}T \cos \psi & T \cos \psi & -T \sin \psi & 0 & 0 & 0 \\ 0 & 1 & 0 & \hat{u} \cos \psi & -\hat{v}T \sin \psi & T \sin \psi & T \cos \psi & 0 & 0 & 0 \\ & & & 0 & 0 & 1 & 0 & 0 & 0 & T \\ & & & 0 & 0 & 0 & 1 & 0 & 0 & T \\ & & & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ & & & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ & & & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ & & & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$W_v = \frac{\partial f}{\partial s}(\hat{x}_V(k), 0) = \begin{bmatrix} \frac{T^2}{2} \cos \psi & -\frac{T^2}{2} \sin \psi & 0 & 0 \\ \frac{T^2}{2} \sin \psi & \frac{T^2}{2} \cos \psi & 0 & 0 \\ & & 0 & 0 & \frac{T^2}{2} & 0 \\ & & 0 & 0 & 0 & \frac{T^2}{2} \\ & & T & 0 & 0 & 0 \\ & & 0 & T & 0 & 0 \\ & & 0 & 0 & T & 0 \\ & & 0 & 0 & 0 & T \\ & & 0 & 0 & 0 & 0 \end{bmatrix} \quad (7)$$

3) *Atualização Sensorial*: Os dados providos pela DVL, MRU, bússola são usados como observação direta em seus correspondentes componentes no vetor de estados. As medidas dos diferentes sensores são integradas assim que adquiridas usando-se as equações clássicas do filtro de Kalman para modelos lineares.

Para a atualização da velocidade usam-se os dados do sensor DVL. Esta fornece medidas em 3DOF, as medidas de velocidades podem ser tomadas como observação direta

das velocidades do veículo. Define-se assim, a matriz de observação H como:

$$H_D = [0_{3 \times 4} \ I_{3 \times 3} \ 0_{3 \times 2} \ 0_{3 \times 2n}], \quad (8)$$

E a matriz de covariância do ruído de medidas da DVL como:

$$R_D = \begin{bmatrix} \sigma^2 D_u & 0 & 0 \\ 0 & \sigma^2 D_v & 0 \\ 0 & 0 & \sigma^2 D_w \end{bmatrix} \quad (9)$$

Os dados de profundidade são obtidas facilmente por um sensor de profundidade e podem ser inseridas diretamente na observação da profundidade, resultando na matriz H :

$$H_P = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0_{1 \times 2n}] \quad (10)$$

A variância da profundidade é representada simplesmente como:

$$R_P = \sigma_P^2 \quad (11)$$

A bússola mede o ângulo do veículo em relação ao norte magnético. Assim a orientação do veículo ψ é referenciado a base de referência da terra, deve ser subtraída ao primeiro ângulo aferido para produzir uma medida compatível com a base de referência do veículo, produzindo a seguinte matriz H :

$$H_C = [0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0_{1 \times 2n}] \quad (12)$$

O ruído da orientação é representada pela variância da bússola R_C .

$$R_C = \sigma_C^2 \quad (13)$$

C. Extração de Feature de Sonar e Estimação de incerteza

Os dados acústicos advindos do sonar podem ser afetados por uma série de fatores que produzem ruídos, fantasmas entre outros. Assim, o primeiro passo no tratamento dos dados acústicos é executado na onda emitida (*beam*) para uma determinada orientação. Esse processo consiste em selecionar apenas os retornos de distância (*bins*) com uma intensidade maior do que um valor pré-estabelecido, estes retornos (*bins*) são os máximos locais da leitura. Os dados descartados são armazenados para serem usados posteriormente na determinação de incerteza de *features*. Também devem ser eliminados *bins* redundantes através da exclusão de máximos locais que estão muito próximos, pois estes provavelmente representam os mesmos objetos.

O processo de *threshold* dos dados permite que as próximas etapas do processo de extração de *features* trabalhem apenas com os dados mais consideráveis e que, portanto tem maior probabilidade de representar um marco do ambiente, uma vez que grande parte dos dados redundantes e ruídos são excluídos nesta primeira fase.

Para armazenar a representação do ambiente é criado um buffer de leituras, o qual representa os 180° mais recentes, esse valor advém do fato que uma linha pode cobrir no

máximo 180°. Neste buffer são armazenados os valores de ângulo e distância dos máximos locais, conforme descrito anteriormente.

1) *Espaço de Hough e determinação de features*: A fim de determinar as *features* existentes no ambiente, periodicamente é executado uma abordagem de voto usando os dados armazenados no buffer. Isso é feito com uma aplicação modificada da clássica transformada de Hough. Em uma tabela bidimensional, para as coordenadas de orientação e distância perpendicular, são geradas todas as posições possíveis de localizações de *features* (p^B e θ^B de acordo com uma base de referência B). As *features* que receberem o maior número de votos de acordo com os dados do buffer são as prováveis descritoras de objetos do ambiente.

A base de referência B pode ser definida arbitrariamente, porém uma boa abordagem é considerar como referência a posição do sonar. Neste caso permite-se integrar uma feature diretamente no sistema de SLAM, uma vez que já está representada com relação ao sensor, como uma observação de uma feature já presente no mapa, ou incorporada como uma nova feature depois de ser ajustada com a posição corrente do veículo.

O fato de definir a base de referência como posição do sonar requer que toda vez que o veículo mude de posição se recalcule a base de referência B, uma vez que esta não é fixa. O espaço de Hough é discretizado em coordenadas polares, de acordo com a resolução de leitura do sonar, no caso do data set da marina (1.8° e 0.1m). Uma resolução maior não aumenta necessariamente a qualidade da detecção, uma vez que a resolução do sonar limita sua precisão.

O algoritmo para extração de *features* inicia com a chegada de um novo *beam*, o espaço de Hough é referenciado para posição corrente do sonar como nova base de referência. Depois, todos os *bins* armazenados no buffer são referenciados para a base de referência B para serem usados em um espaço de voto, essa transformação para nova referência distorce os dados. Uma vez que os dados estejam distorcidos, os votos correspondentes a cada *bin* são atribuídos para as *features* candidatas na forma de um modelo do sonar. Então uma busca pelas *features* com maior número de votos é executada, a fim de selecionar as *features* candidatas com maior número de votos.

2) *Selecionando features candidatas*: Devido a incerteza produzida pela largura do feixe da onda de sonar, uma medição não pode ser atribuída a um único ponto no espaço. Trabalhos [29], [30] propuseram uma abordagem comum, bem estabelecida para sonares usados ao ar livre, considerando a medida na forma de um arco cuja abertura representa a incerteza da largura de medida. Uma intensidade alta de retorno é produzida quando a onda acústica atinge uma superfície perpendicularmente, pode-se inferir que todas as superfícies tangentes ao arco podem explicar o retorno de alta intensidade. Enquanto essa abordagem é bem aceita em ambientes ao ar livre, [5] propõem uma abordagem onde em ambientes subaquáticos a capacidade de detecção de objetos não é limitada as superfícies tangentes ao arco, mas aos beams que fazem intersecção a superfície, dentro

de limites definidos por um ângulo máximo de incidência também produzem um retorno discernível. Basicamente, dado um ângulo de largura de feixe horizontal α e um ângulo de incidência β , o conjunto de linhas é composto não apenas de linhas tangentes ao arco definido por α , mas também por todas as linhas que interseccionam o arco com um ângulo de incidência menor que $\pm \beta$. Antes da votação, esse conjunto de linhas deve ser determinado para cada *bin* armazenado no buffer de dados.

Tanta a transformação e os valores de distância podem ser obtidos do buffer. Para emular o efeito da largura de feixe horizontal, o conjunto de valores i são tomados a uma dada resolução dentro de uma abertura de $\pm \alpha/2$ sobre a direção a qual o transdutor está orientada, também referido como θ_S^B :

$$\theta_S^B - \alpha/2 \leq \theta_i^B \leq \theta_S^B + \alpha/2. \quad (14)$$

Cada valor de θ_i^B representa o parâmetro de orientação para uma linha tangente com o arco o qual modela a largura do feixe horizontal. Tanto linhas tangente ao arco, quanto as dentro do ângulo máximo de incidência $\pm \beta$ são candidatas. Por esta razão, k valores são tomados em conta para uma dada resolução para cada valor de θ_i^B e dentro de uma abertura $\pm \beta$:

$$\theta_i^B - \beta \leq \theta_{i,k}^B, k \leq \theta_i^B + \beta. \quad (15)$$

O resultado dessa operação são a geração de $i \times k$ valores diferentes de $\theta_{i,k}^B$. Essas são as orientações de um conjunto de linhas a qual são uma representação de todas as possíveis candidatas compatíveis com o *bin*. O passo final é determinar o parâmetro de distância $p_{i,k}^B$ correspondendo a cada valor obtido de orientação $\theta_{i,k}^B$. Dada a geometria do problema, são calculados como:

$$p_{i,k}^B = x_S^B \cos(\theta_{i,k}^B) + y_S^B \sin(\theta_{i,k}^B) + p^S \cos(\theta_i, k) \quad (16)$$

Uma vez gerado o conjunto de linhas, estas podem ser usadas para receber votos de acordo com os *bins* de máximos locais armazenados no buffer. A resolução escolhida durante a geração das $i \times k$ linhas deve ser suficiente para assegurar uma exploração correta das células do grid do espaço de Hough. Cada célula do grid pode receber apenas um voto de um *bin* particular, células contendo múltiplos votos portanto representam linhas compatíveis com diferentes *bins* individuais.

Toda vez que um novo *beam* é adquirido, um novo espaço de voto é gerado para se buscar uma linha candidata vencedora. Uma linha vencedora deve ser detectada uma vez que seja totalmente observada. No espaço de voto, a zona onde essas linhas candidatas podem existir é completamente determinada pelo subconjunto de linhas candidatas contidas nos 180° mais recentes, que não intersectam com o último *beam*. Qualquer linha candidata com um número suficiente de votos encontrada dentro desta zona é declarada vencedora. Realizando a detecção nesta forma garante que o algoritmo detecte as linhas assim que as mesmas sejam completamente

visíveis. Assim que uma linha é detectada todos os *bins* envolvidos na eleição de uma linha são removidos no buffer para não interferir com a detecção das próximas *features*.

V. RESULTADOS PARCIAIS

O resultado apresentado na Figura 5 constitui um dos primeiros testes do sistema. Este usa dados do *data set* obtido em uma marina abandonada [5]. O Mesmo foi gerado utilizando-se apenas dados dos sensores DVL que proporciona medidas de velocidades nos eixos x, y, z e MRU que proporciona dados da atitude do robô *roll, pitch, yaw* e suas respectivas velocidades.

Para a geração da estimativa de posição do estado do robô foram usadas as equações do Filtro de Kalman Estendido. Nota-se uma grande discrepância se comparado com o trajeto gerado por GPS (*ground truth*) que apresenta o caminho real percorrido pelo robô, conforme é exibido na Figura 6. Este resultado será aprimorado ao longo das próximas etapas do trabalho, utilizando dados do sonar.

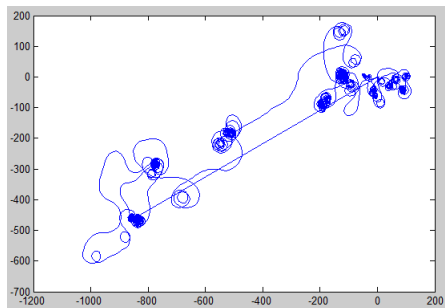


Fig. 5. Caminho gerado com uso do Filtro de Kalman Estendido usando dados de sensores DVL e MRU.



Fig. 6. Trajeto real realizado pelo robô aferido através de DGPS (*ground truth*)

VI. CONCLUSÃO

Para um robô ser considerado autônomo antes de tudo deve ser capaz de se localizar em seu ambiente de operação. Durante os últimos anos diferentes trabalhos foram propostos para a resolução do problema de Localização e Mapeamento Simultâneos (SLAM). Apesar de ser amplamente explorado o problema de SLAM é considerado um problema não resolvido na robótica principalmente em ambientes de difícil acesso e sem sinal de GPS, como o subaquático. O ambiente subaquático é um dos mais desafiadores de ser trabalhado devido às propriedades do meio que dificultam a operação e sensoramento do robô. Neste contexto, este trabalho busca propor uma abordagem de SLAM para ambientes subaquáticos usando dados de vários sensores como sonar, DVL, MRU. Os primeiros resultados foram apresentados, utilizando apenas dados de sensores inerciais. Como trabalhos futuros busca-se a integração do sonar para a melhoria dos resultados. Além disso, serão integrados ao sistema de SLAM dados sensoriais de câmera juntamente com uma nova abordagem usando *bag of words* junto ao módulo de associação de dados para localização de *features* visuais em ambientes estruturados.

REFERENCES

- [1] J. L. Leonard, R. N. Carpenter, and H. J. S. Feder, "Stochastic mapping using forward look sonar," *Robotica*, vol. 19, no. 5, pp. 467–480, Aug. 2001.
- [2] J. J. Leonard and H. J. S. Feder, "Decoupled stochastic mapping [for mobile robot & auv navigation]," *Oceanic Engineering, IEEE Journal of*, vol. 26, no. 4, pp. 561–571, 2001.
- [3] S. B. Williams, P. Newman, J. Rosenblatt, G. Dissanayake, and H. Durrant-Whyte, "Autonomous underwater navigation and control," *Robotica*, vol. 19, no. 5, pp. 481–496, Aug. 2001.
- [4] S. Tena I., Y. Petillot, and D. Lane, "Concurrent mapping and localization using sidescan sonar," *IEEE JOURNAL OF OCEANIC ENGINEERING*, vol. 29, no. 2, pp. 442–456, APR 2004.
- [5] D. Ribas, P. Ridao, J. Tardos, and J. Neira, "Underwater slam in man-made structured environments," *Journal of Field Robotics*, vol. 25, no. 11–12, pp. 898–921, October 2008.
- [6] S. Wang, L. Chen, H. Hu, Z. Xue, and W. Pan, "Underwater localization and environment mapping using wireless robots," *Wireless Personal Communications*, vol. 70, no. 3, pp. 1147–1170, 2013. [Online]. Available: <http://dx.doi.org/10.1007/s11277-013-1106-z>
- [7] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (slam): Part i the essential algorithms," *IEEE ROBOTICS AND AUTOMATION MAGAZINE*, vol. 2, p. 2006, 2006.
- [8] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainty," *Int. J. Rob. Res.*, vol. 5, no. 4, pp. 56–68, Dec. 1986. [Online]. Available: <http://dx.doi.org/10.1177/027836498600500404>
- [9] H. Durrant-Whyte, "Uncertain geometry in robotics," *Robotics and Automation, IEEE Journal of*, vol. 4, no. 1, pp. 23–31, 1988.
- [10] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.
- [11] R. N. Carpenter, "Concurrent mapping and localization with fls," *Workshop on Autonomous Underwater Vehicles, Cambridge, MA, USA*, pp. 133–148, 1998.
- [12] S. Reed, Y. Petillot, and J. Bell, "Automated approach to classification of mine-like objects in sidescan sonar using highlight and shadow information," *IEE PROCEEDINGS-RADAR SONAR AND NAVIGATION*, vol. 151, no. 1, pp. 48–56, FEB 2004.
- [13] J. Neira and J. D. Tardos, "Data association in stochastic mapping using the joint compatibility test," 2001.
- [14] C. N. Roman and H. Singh, "Improved vehicle based multibeam bathymetry using sub-maps and slam," in *IROS. IEEE*, 2005, pp. 3662–3669.

- [15] N. Fairfield, D. Jonak, G. Kantor, and D. Wettergreen, "Field results of the control, navigation, and mapping systems of a hovering auv," in *Proceedings of the Unmanned Untethered Submersible Technology Conference (UUST)*. Citeseer, 2007, pp. 20–22.
- [16] N. Fairfield, G. Kantor, and D. Wettergreen, "Real-time slam with octree evidence grids for exploration in underwater tunnels," *Journal of Field Robotics*, vol. 24, no. 1-2, pp. 03–21, 2007.
- [17] G. Grisetti, G. Tipaldi, C. Stachniss, W. Burgard, and D. Nardi, "Fast and accurate SLAM with Rao-Blackwellized particle filters," *Robotics and Autonomous Systems, Special issue on Simultaneous Localization and Map Building*, vol. 55, no. 1, pp. 30–38, 2007.
- [18] D. Ribas, P. Ridao, and J. Neira, *Underwater SLAM for Structured Environments Using an Imaging Sonar*, ser. Springer Tracts in Advanced Robotics. Springer, 2010. [Online]. Available: <http://books.google.com.br/books?id=R72Q2-IN3D0C>
- [19] J. Folkesson, J. Leederkerken, R. Williams, A. Patrikalakis, and J. Leonard, *A Feature Based Navigation System for an Autonomous Underwater Robot*, ser. Springer Tracts in Advanced Robotics. Springer Berlin Heidelberg, 2008, vol. 42.
- [20] P. Drews, S. Botelho, and S. Gomes, "Slam in underwater environment using sift and topologic maps," in *Robotic Symposium, 2008. LARS '08. IEEE Latin American*, 2008, pp. 91–96.
- [21] L. Jaulin, "A nonlinear set membership approach for the localization and map building of underwater robots," *Trans. Rob.*, vol. 25, no. 1, pp. 88–98, Feb. 2009. [Online]. Available: <http://dx.doi.org/10.1109/TRO.2008.2010358>
- [22] S. Barkby, S. Williams, O. Pizarro, and M. Jakuba, "An efficient approach to bathymetric SLAM," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, Oct. 2009, pp. 219–224. [Online]. Available: <http://dx.doi.org/10.1109/iros.2009.5354248>
- [23] A. Mallios, P. Ridao, E. Hernández, D. Ribas, F. Maurelli, and Y. Petillot, "Pose-based slam with probabilistic scan matching algorithm using a mechanical scanned imaging sonar," in *OCEANS 2009-EUROPE*. IEEE, 2009, pp. 1–6.
- [24] A. Mallios, P. Ridao, D. Ribas, F. Maurelli, and Y. R. Petillot, "EKF-slam for auv navigation under probabilistic sonar scan-matching," in *IROS*. IEEE, 2010, pp. 4404–4411. [Online]. Available: <http://dblp.uni-trier.de/db/conf/iros/iros2010.htmlMalliosRRMP10>
- [25] A. Kim and R. Eustice, "Pose-graph visual slam with geometric model selection for autonomous underwater ship hull inspection," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2009.
- [26] A. Burguera, Y. Gonzalez, and G. Oliver, "Underwater slam with robocentric trajectory using a mechanically scanned imaging sonar," in *Intelligent Robots and Systems (IROS), 2011 IEEE-RSJ International Conference on Robotics*, 2011, pp. 3577–3582.
- [27] J. Aulinas, Y. Petillot, X. Llado, J. Salvi, and R. Garcia, "Vision-based underwater slam for the sparus auv", proceedings of the 10th international conference on computer and it applications in the maritime industries," in *COMPIT 11*, 2011, pp. 171–181.
- [28] A. Kim, "Active visual slam with exploration for autonomous underwater navigation," Ph.D. dissertation, MICHIGAN UNIVETSISTY ANN ARBOR DEPT OF MECHANICAL ENGINEERING, 2012.
- [29] J. J. Leonard, H. F. Durrant-Whyte, and O. Pj, "Directed sonar sensing for mobile robot navigation," 1992.
- [30] J. D. Tardos, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *Int. J. Robotics Research*, vol. 21, pp. 311–330, 2002.