

# Exploração de Ambiente Interno com Múltiplos Veículos e Técnica FastSLAM

Adão de Melo Neto, Paulo Fernando Ferreira Rosa, Thiago Eustáquio Alves de Oliveira and Paulo César Pellanda  
 Instituto Militar de Engenharia - Praça Gal Tibúrcio, Nr 81 , Rio de Janeiro - RJ

**Resumo**—Neste trabalho nós investigamos a exploração integrada de um ambiente com múltiplos veículos usando uma estratégia baseada em grades de ocupação e uma técnica de localização e mapeamento simultâneo (SLAM). A estratégia de exploração usa conceitos de custo e utilidade de células fronteiras. Além disso, o método de SLAM utilizado é baseado no algoritmo FastSLAM com marcos extraídos a partir de sensores visuais e um mapa de características comum. Ambas as atividades são coordenadas por um agente central. Os resultados mostram que quando dois veículos podem comunicar-se com um agente central, a tarefa de exploração torna-se mais eficiente do que a realizada com apenas um veículo pois ela reduz o tempo de exploração mantendo ou melhorando a precisão na posição e orientação dos veículos com um igual número de partículas.

## I. INTRODUÇÃO

A exploração integrada de um ambiente é uma atividade de alto nível em que métodos de exploração, localização, mapeamento e navegação devem ser combinados tal que veículos autônomos sejam capazes de mapear um ambiente maximizando ganhos (por exemplo, a acurácia na posição e orientação - pose - dos veículos) e minimizando custos (por exemplo, o tempo gasto na exploração). Neste trabalho, usamos a técnica FastSLAM [3] para localização dos veículos

O SLAM é uma área de pesquisa em franca atividade que está relacionada com a capacidade de auto-localização do veículo e com a obtenção de um mapa de características do ambiente em que o veículo se encontra. As duas atividades são correlatas, pois, para obter um mapa do ambiente, é necessário que o veículo conheça sua localização, ao passo que para a localização do veículo é necessário que o mesmo tenha um mapa do ambiente. Em muitos casos, para localização do veículo, são utilizados dispositivos GPS (*global positioning system*). No entanto, a qualidade das informações obtidas através deste sistema pode ser afetada pelo número de satélites à vista e por interferências eletromagnéticas.

O problema do SLAM pode ser assim definido: dado um veículo autônomo dentro de um ambiente desconhecido e, usando apenas observações relativas a marcos no ambiente em relação ao veículo, construir um mapa de características destes marcos e computar simultaneamente uma estimativa da localização do veículo baseado neste mapa.

Uma vez que o SLAM está garantindo a localização do veículo, uma grade de ocupação pode ser concomitantemente gerada (Fig.1) para auxiliar a exploração do ambiente. A referida grade consiste de um conjunto de células de pequena dimensão que discretizam o ambiente a ser explorado. Estas

células armazenam a sua probabilidade de ocupação, obtida através do sensoriamento do ambiente com o sensor (laser, por exemplo) embarcado no veículo.

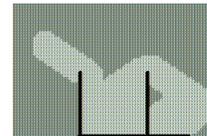


Fig. 1. Grade de ocupação: regiões livres (branco), ocupadas (preto) e inexploradas (cinza).

O objetivo deste trabalho é mostrar que podemos explorar um ambiente interno com dois veículos de forma eficiente quando estamos usando o FastSLAM (com a obtenção de marcos feitas pelo algoritmo SIFT) para localizá-los. Isto é porque a estratégia de exploração coordenada reduz o tempo utilizado na exploração em 30% na média, e porque os veículos mantem ou melhoraram a precisão em sua pose, tudo em relação a operação com um único veículo. No FastSLAM, a precisão na pose é conseguido porque a abordagem utiliza um mapa de características comum aos veículos, pela estrutura típica de um ambiente interno e pela quantidade de marcos que podem ser obtidos com o algoritmo SIFT. Esta precisão é obtida com o uso de um mesmo número de partículas, que poderia ser maior, uma vez que o problema estimado é maior.

A seção II apresenta alguns trabalhos relacionados ao SLAM e a estratégia de exploração. Trataremos do FastSLAM para um e múltiplos veículos, respectivamente, nas seções III e IV. A seção V enfoca estratégia de exploração e a seção VI apresenta a descrição dos experimentos e os resultados obtidos. Teremos nossas conclusões e considerações finais na seção VII .

## II. TRABALHOS RELACIONADOS

Diversos estudos sobre o mapeamento tem sido conduzidos. Algumas abordagens estimam a pose usando somente a hodiometria, o que frequentemente conduz a resultados imprecisos. Outras usam técnicas de SLAM (EKF-SLAM [2] e FastSLAM [3]) com a extração de características a partir de dados brutos de sensores laser e sonar [4].

Em [6] um filtro de partículas Rao-Blackwellized [3] é usado para estimar simultaneamente o mapa e o caminho de um único veículo. No mencionado trabalho, características SIFT são usadas como marcos no ambiente, cujas coordenadas são extraídas com um par de câmeras estéreo.

Abordagens para SLAM com múltiplos veículos podem ser grupadas em duas soluções. Na primeira, cada veículo estima seu próprio mapa usando suas observações e, em um estágio posterior, um mapa comum é formado pela fusão dos mapas individuais dos veículos. Na segunda, a estimação de toda trajetória e do mapa é feita conjuntamente. Um único mapa é computado simultaneamente usando as observações de todos os veículos.

O trabalho exposto em [8] pode ser classificado no primeiro grupo. A idéia é que cada veículo constrói seu próprio mapa e ao mesmo tempo tenta continuamente se localizar no mapa construído pelos outros veículos usando filtros de partículas. A abordagem pode lidar com a situação onde a pose inicial dos veículos são desconhecidas, resolvendo o problema da associação de dados com uma técnica *rendez-vous*. Contudo, a fusão dos mapas individuais é computacionalmente custosa.

A abordagem apresentada em [9] pode ser classificada no segundo grupo. Ela usa um filtro de Kalman estendido(EKF) para estimar um vetor de estado formado pelas poses de todos os veículos e um conjunto de marcos 2D. Os veículos obtêm observações e constroem um único mapa usando equações de atualização do clássico EKF [2]. As poses iniciais dos veículos devem ser conhecidas antecipadamente e a associação de dados é assumida como conhecida. Neste caso, a principal desvantagem reside no fato de que uma única hipótese sobre a pose do veículo é mantida.

[7] apresenta um algoritmo baseado em [3] onde o mapa é comum aos veículos e assume-se que eles tenham pose inicial previamente conhecida. Os autores mostram através de experimentos no MATLAB e sem o uso de técnica de exploração que, para até três veículos, o erro na pose é mantido ou melhorado, embora o problema a ser estimado seja maior. [10] propõe um algoritmo baseado em [3] com múltiplos veículos onde eles podem iniciar de pose previamente desconhecida. Nesta abordagem, cada veículo constrói em mantém seu próprio mapa. Quando eles estão na mesma linha de visada, os mapas são fundidos.

Em relação a estratégias de exploração, destacamos a abordagem descrita em [12] que usa o conceito de custos e utilidade de células fronteiras.

### III. FASTSLAM

O FastSLAM [3] decompõe o problema do SLAM em um problema de localização do veículo, e uma coleção de problemas de estimação de marcos que estão condicionados a estimativa da pose do veículo.

Seja  $u_t$  uma ação de controle responsável pela troca de estado de um veículo no instante  $t$ . Em robótica, a pose  $\chi_t$  do veículo e a observação  $z_t$  de um marco  $\theta_j$  são modelados por leis probabilísticas

$$p(\chi_t|\chi_{t-1}, u_t) \text{ e } p(z_t|\chi_t) \quad (1)$$

com valores amostrados por funções ( $h$  e  $g$ ) com ruído Gaussiano adicionado  $N(\cdot)$  adicionado

$$\chi_t = h(u_t, \chi_{t-1}) + N(0, R_t) \quad z_t = g(\chi_t, \theta_j) + N(0, Q_t) \quad (2)$$

O FastSLAM estima a distribuição de probabilidade posterior sobre o caminho do veículo  $\chi^t = \{\chi_1, \dots, \chi_t\}$  e o mapa  $\Theta$  (Eq.3) considerando as observações  $z^t = \{z_1, \dots, z_t\}$ , as ações de controle  $u^t = \{u_1, \dots, u_t\}$  e as associações  $a^t = \{a_1, \dots, a_t\}$  entre as características dos marcos que foram observados e as características dos marcos no mapa

$$p(\chi^t, \Theta | z^t, u^t, a^t) \quad (3)$$

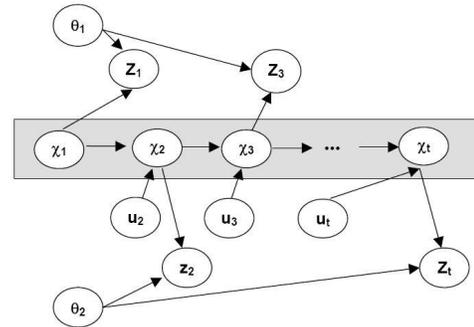


Fig. 2. Fatorando o problema do SLAM - se o caminho verdadeiro do veículo  $\chi^t$  é conhecido (região sombreada), então a posição dos marcos  $\theta_1$  e  $\theta_2$  são condicionalmente independentes [3].

É demonstrado que se o caminho  $\chi^t$  é conhecido, então a posição do marco  $\theta_i$  em  $\Theta$  são condicionalmente independentes, o que permite fatorar o problema da estimação da distribuição de probabilidade posterior de  $\chi^t$  e  $\Theta$  como um produto de termos simples (Eq. 4) como descrito na representação (parcial) em rede de Bayes da Figura 2.

$$p(\chi^t, \Theta | z^t, u^t, a^t) = \underbrace{p(\chi^t | z^t, u^t, a^t)}_{\text{Caminho}} \prod_{n=1}^N \underbrace{p(\theta_n | \chi^t, z^t, u^t, a^t)}_{\text{Marco}} \quad (4)$$

A distribuição de probabilidade posterior sobre  $\chi^t$  é estimada usando um filtro de partículas [13] e a distribuição de probabilidade posterior sobre os  $N$  marcos  $\theta_i$  de cada partícula são estimados por meio de  $N$  filtros de Kalman Estendidos [13] condicionados ao caminho  $\chi^t$ .

O filtro de partículas representa a distribuição usando um conjunto  $S_t = \{S_t^{[1]}, \dots, S_t^{[M]}\}$  de partículas

$$S_t^{[m]} = \left[ \chi^{t,[m]}, \underbrace{\psi_{1,t}^{[m]}, \Sigma_{1,t}^{[m]}, d_1^{[m]}, \dots, w_t^{[m]}}_{\Theta^{[m]}} \right] \quad (5)$$

onde  $\psi_{n,t}^{[m]}$  e  $\Sigma_{n,t}^{[m]}$  são a média e a covariância das coordenadas de  $\theta_n$  condicionado ao caminho  $\chi^{t,[m]}$ ,  $d_n^{[m]}$  é o descritor de suas características e  $w_t^{[m]}$  é o peso da partícula.

Quando o veículo realiza uma observação ele deve atualizá-la no seu mapa. Dado que os sensores estão sujeitos a erros, as informações incorporadas no mapa podem ter uma certa quantidade de incerteza. Esta imprecisão pode conduzir a erros de associação de dados e de atualização. O erro de

atualização pode ser minimizado através sucessivas observações e, portanto, para melhorar a precisão na pose do veículo, é necessário *fechar o loop*, ou seja, reobservar um marco previamente observado. O erro na associação de dados causado pela associação errônea de uma característica presente no mapa a uma característica de um marco observado no ambiente pode ser diminuída com a utilização de conjunto de características mais robustos, como é o caso dos descritores SIFT. Estes são extraídos do ambiente através do algoritmo SIFT [15], um algoritmo de visão computacional usado para detectar e descrever características em imagens, cujo descritor (um vetor de dimensão 128) é invariante a escala e orientação da imagem e parcialmente invariante ao ponto de vista e a iluminação.

O algoritmo FastSLAM usado aqui é mostrado na seqüência. Nós consideramos uma observação  $z_t = \{v_t, d_t\}$  onde  $d_t$  é o descritor SIFT de um marco visual e  $v_t = [\theta_x, \theta_y, \theta_z]^T$  são suas coordenadas.

#### FastSLAM Visual ( $S_{t-1}, v_t, d_t, u_t$ )

1.  $S_t = S_{aux} = \emptyset$
2. Para cada partícula  $S_{t-1}^{[m]}$  de  $S_{t-1}$ 
  1. Amostrar  $\chi_t^{[m]} \sim p(\chi_t | \chi_{t-1}^{[m]}, u_t)$
  2. Para cada marco  $\theta_n^{[m]}$  no mapa  $\Theta^{[m]}$  computar
    1.  $E_n^{[m]} = Mahalanobis(d_n^{[m]}, d_t)$
    2.  $p_{n,t}^{[m]} = f(v_t, \theta_n^{[m]})$
  3. Fim Para
  4.  $a_t = argmin_n E_n^{[m]}$
  5. Se  $E_{a_t} < E_0$  (e  $p_{a_t,t}^{[m]} > P_0$ )
    1. Atualizar marco  $\theta_{a_t}^{[m]}$
    2.  $w_t^{[m]} = p_{a_t,t}^{[m]}$
  6. Se  $E_{a_t} \geq E_0$ 
    1. Novo marco em  $\Theta^{[m]}$
  7. Fim Se
  8.  $S_t^{[m]} \Rightarrow S_{aux}$
3. Fim Para
4. Para cada partícula  $S_t^{[m]}$  de  $S_{aux}$ 
  1. Extrair partícula  $S_t^{[i]}$  de  $S_{aux}$  com base em  $w_t^{[i]}$
  2.  $S_t^{[i]} \Rightarrow S_t$
5. Retornar  $S_t$

Para cada partícula  $S_{t-1}^{[m]}$  de  $S_{t-1}$  é amostrado o modelo de movimento do veículo (passo 2.1). Por outro lado, para cada marco  $\theta_n^{[m]}$  no mapa  $\Theta^{[m]}$  é computada a distância de Mahalanobis  $E_n^{[m]}$  entre o descritor SIFT  $d_t$  do marco observado e o descritor  $d_n^{[m]}$  de  $\theta_n^{[m]}$  (passo 2.2.1), bem como a qualidade da associação de  $\theta_n^{[m]}$  com  $v_t$ , feita através de função do FastSLAM (passo 2.2.2). Como associação de dados (passo 2.4), escolhe-se o marco que minimiza a distância de Mahalanobis. Se esta distância estiver abaixo de um certo limite  $E_0$  (e  $p_{a_t,t}^{[m]}$  seja maior que um valor limite  $P_0$ ), consideramos que temos uma associação de dados correta e atualizamos a estimativa das coordenadas do marco  $\theta_{a_t}^{[m]}$  com equações do Filtro de Kalman Estendido (passo 2.5.1). Se for maior ou igual, criamos um novo marco no mapa (passo 2.6.1). O peso da partícula corresponde a qualidade da associação  $p_{a_t,t}^{[m]}$  do marco  $\theta_{a_t}^{[m]}$  que foi associado com  $v_t$  (passo 2.5.2). Na reamostragem (passo 4), partículas com maior peso  $w_t^{[i]}$  são

replicadas.

O algoritmo FastSLAM no caso de um veículo requer tempo  $MN$ . Isto ocorre porque  $M$  partículas no filtro necessitam ser processadas, enquanto que, para cada partícula, a associação de dados necessita iterar sobre  $N$  marcos no mapa. Entretanto, se os marcos forem armazenados usando uma estrutura kd-tree, a pesquisa pelo vizinho nesta estrutura pode ser feita em tempo  $\log N$ , o que acelera a computação da associação de dados, implicando em um tempo  $M \log N$ .

O sucesso obtido com o SLAM tem motivado a pesquisa do SLAM com múltiplos veículos, como é discutido a seguir.

#### IV. FASTSLAM COM MÚLTIPLOS VEÍCULOS

Quando múltiplos veículos tem a possibilidade de comunicar com um agente central é justo que trabalhem juntos uma vez que há uma redução no tempo de exploração, bem como permite uma cooperação na observação de marcos no ambiente.

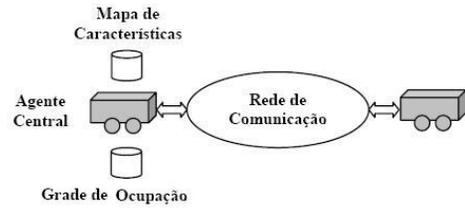


Fig. 3. Esquema da exploração com dois veículos em que um agente central é responsável pela construção do mapa de características e da grade de ocupação.

Se  $k$  veículos exploram o ambiente construindo um mapa comum (Fig. 3) e no instante  $t$  o veículo ( $i$ ) na pose  $\chi_{t,(i)}$  executa uma única observação  $z_{t,(i)}$ , a distribuição de probabilidade posterior sobre o caminho  $\chi_{(1:k)}^t$  dos  $k$  veículos e o mapa  $\Theta$  pode ser estimado a partir da seguinte função [7]

$$p(\chi_{(1:k)}^t, \Theta | z_{(1:k)}^t, u_{(1:k)}^t, a^t) = \underbrace{p(\chi_{(1:k)}^t | z_{(1:k)}^t, u_{(1:k)}^t, a^t)}_{\text{Caminho}} \prod_{n=1}^N \underbrace{p(\theta_n | \chi_{(1:k)}^t, z_{(1:k)}^t, u_{(1:k)}^t, a^t)}_{\text{Marco}} \quad (6)$$

onde  $\chi_{(1:k)}^t = \{\chi_{(1)}^t, \dots, \chi_{(k)}^t\}$ ,  $u_{(1:k)}^t = \{u_{(1)}^t, \dots, u_{(k)}^t\}$  e  $z_{(1:k)}^t = \{z_{(1)}^t, \dots, z_{(k)}^t\}$  são o conjunto de caminhos, ações e observações dos  $k$  veículos e  $a^t = \{a_1, \dots, a_t\}$  é o histórico de associações de dados.

A distribuição de probabilidade posterior sobre  $\chi_{(1:k)}^t$  é estimada usando  $k$  filtros de partículas. Por outro lado, a distribuição de probabilidade posterior sobre os  $N$  marcos  $\theta_i$  de cada partícula são estimados por meio de  $kN$  independentes Filtros de Kalman Estendidos condicionados a  $\chi_{(1:k)}^t$ .

Uma vez que o mapa é comum aos veículos, os  $k$  filtros de partículas produzem um mesmo conjunto  $S_t$  de partículas

$$S_t^{[m]} = [\chi_{(1:k)}^{t,[m]}, \psi_{1,t}^{[m]}, \Sigma_{1,t}^{[m]}, d_{1,t}^{[m]}, \dots, \psi_{N,t}^{[m]}, \Sigma_{N,t}^{[m]}] \quad (7)$$

O estado a ser estimado é composto pela pose  $\chi_{t,(1:k)}$  dos  $k$  veículos. Teoricamente espera-se que com o aumento da dimensão do problema, um número maior de partículas seja necessário para uma boa estimação [13].

Um algoritmo para o FastSLAM Visual com Múltiplos Veículos é apresentado a seguir [7]. A rotina FastSLAM\* corresponde ao algoritmo FastSLAM (página 3) considerando apenas os passos 1 – 3.

#### FastSLAM Visual com Múltiplos Veículos

1.  $S_t = \emptyset$
2. Para  $t = 1$  até Fim fazer
  1.  $[v_{t,(1)}, d_{t,(1)}, v_{t,(2)}, d_{t,(2)}] = \text{Observações}(O)$
  2.  $[S_t, w_{t,(1)}] = \text{FastSLAM}^*(S_{t-1}, v_{t,(1)}, d_{t,(1)}, u_{t,(1)})$
  3.  $[S_t, w_{t,(2)}] = \text{FastSLAM}^*(S_{t-1}, v_{t,(2)}, d_{t,(2)}, u_{t,(2)})$
  4.  $w_t = w_{t,(1)} w_{t,(2)}$
  5.  $S_t = \text{Reamostragem}(S_t, w_t)$
3. Fim Para

Neste ponto, é importante notar que estamos estimando um mapa comum a todos os veículos, e todas observações, independente de qual veículo, são enviadas ao agente central (Fig.3). No passo 4, estamos considerando que os veículos iniciam a exploração a partir de uma pose conhecida pelo agente central. Como um sistema de referência é adotada pelo agente central e a pose inicial dos veículos estão sobre este sistema, as coordenadas dos marcos no mapa estarão também no mesmo sistema de referência. Portanto, para cada partícula  $S_t^{[m]}$  definida na equação 7,  $k$  pesos são calculados e um peso total associado com a partícula  $S_t^{[m]}$  é definido como

$$w_t = \prod_{i=1}^k w_{t,(i)}^{[m]} \quad (8)$$

Cada veículo inclui suas observações em um ordem pré-definida (passo 4). Desde que as observações são independentes e Gaussianas, esta ordem não afeta a estimativa final dos marcos. O algoritmo FastSLAM no caso de  $k$  veículos requer tempo  $kM \log N$ .

Apresentamos a seguir a estratégia de exploração utilizada.

#### V. ESTRATÉGIA DE EXPLORAÇÃO

Em uma atividade de exploração, o caminho a ser seguido pelo veículo deve ser controlado para ser eficiente. Na abordagem que adotamos [12] um destino - célula fronteira de uma grade de ocupação - é escolhida de acordo com uma função que avalia o custo da navegação e a utilidade do destino.

##### A. Custo da Navegação

Para determinar o custo de se alcançar as células fronteiras atuais, computa-se o caminho ótimo da posição atual do veículo para as células fronteiras com base em um variante determinístico do *value iteration*, um conhecido algoritmo de programação dinâmica [16]. Considera-se que o custo para atravessar uma célula da grade  $(x,y)$  seja proporcional a sua probabilidade de ocupação  $P(occ_{x,y})$ . O caminho de custo mínimo é computado usando dois passos:

- 1) **Inicialização.** A célula da grade que contém a localização do veículo é inicializada com 0, e todas as outras com  $\infty$ 

$$V_{x,y} \leftarrow \begin{cases} 0, & \text{se } (x,y) \text{ é a posição do veículo} \\ \infty, & \text{caso contrário} \end{cases}$$
- 2) **Atualização.** Para todas as células  $(x,y)$  fazer

$$V_{x,y} \leftarrow \min\{V_{x+\Delta x, y+\Delta y} + \sqrt{\Delta x^2 + \Delta y^2} P(occ_{x+\Delta x, y+\Delta y}) \mid \Delta x, \Delta y \in \{-1, 0, 1\} \quad P(occ_{x+\Delta x, y+\Delta y}) \in [0, occ_{max}]\}$$

onde  $occ_{max}$  é a probabilidade de ocupação máxima de uma célula que é permitido a um veículo atravessar. Esta técnica atualiza o valor de todas as células da grade pelo valor dos melhores vizinhos delas, mais o custo de mover para este vizinho. Aqui, o custo é a probabilidade  $P(occ_{x,y})$  que a célula esteja ocupada vezes a distância para a célula. A regra de atualização é repetida até a convergência. Então, cada valor  $V_{x,y}$  é o custo cumulativo do veículo se mover de sua posição para  $(x,y)$ .

##### B. Utilidade das células fronteiras

Se já existe um veículo que se move para uma determinada célula fronteira, espera-se que a utilidade daquela célula seja menor para outros veículos.

Suponhamos que no início todas as células fronteiras  $t$  tenham uma utilidade  $U_t$  que é igual para todas as referidas células. Então, computamos a utilidade  $U(t_n | t_1, \dots, t_{n-1})$  de uma célula fronteira  $t_n$  dado que as células  $t_1, \dots, t_{n-1}$  já tenha sido atribuída aos veículos  $1, \dots, n-1$  como

$$U(t_n | t_1, \dots, t_{n-1}) = U_{t_n} - \sum_{i=1}^{n-1} \underbrace{P(\|t_n - t_i\|)}_d \quad (9)$$

onde  $P(d)$  é a probabilidade de que o sensor de distância do veículo (sensor a laser em nosso caso) irá cobrir células a uma distância  $d$

De acordo com a equação 9, quanto mais veículos movem-se para um local onde  $t_n$  seja provavelmente visível, menor é a utilidade da  $t_n$ . Computamos  $P(d)$  como

$$P(d) \leftarrow \begin{cases} 1 - \frac{d}{dist\_max}, & \text{se } d < dist\_max \\ 0, & \text{caso contrário} \end{cases} \quad (10)$$

onde  $dist\_max$  é a distância máxima coberta pelo sensor laser. Se existe um obstáculo entre duas células fronteiras  $t_n$  e  $t'$ , fazemos  $P(\|t_n - t'\|)$  igual a zero.

##### C. Seleção dos Destinos

Na seleção dos destinos, consideramos para cada veículo  $i$ , um balanceamento entre o custo  $V_t^i$  de se mover para um destino  $t$  e a utilidade  $U_t$  deste destino.

#### Atribuição de Destinos

1. Determinar o conjunto de células fronteiras.
2. Computar  $V_t^i$  para cada veículo  $i$  e célula fronteira  $t$ .
3. Fazer  $U_t = 1$  para todas células fronteiras  $t$ .
4. Enquanto existir um veículo sem destino fazer
  1. Determinar veículo  $i$  e célula fronteira  $t$  que satisfaça
 
$$(i, t) = \text{argmax}_{(i^*, t^*)} (U_{t^*} - V_{t^*}^{i^*})$$
  2. Reduza a utilidade de cada destino  $t^*$  na área de visibilidade de acordo com
 
$$U_{t^*} \leftarrow U_{t^*} - P(\|t - t^*\|)$$
5. Fim Enquanto

A próxima seção descreve o arranjo experimental e alguns resultados.

## VI. EXPERIMENTOS E RESULTADOS OBTIDOS

Os experimentos foram divididos em duas partes de acordo com o tipo de simulação associado. Na primeira parte, a estratégia de exploração é testada usando o simulador *Player/Stage* [17] (Fig. 4[a]). Na segunda, a abordagem SLAM é verificada pela obtenção de dados de um sensor *Kinect - Xbox* sensor em um percurso pré-definido.

Os resultados foram obtidos considerando o uso de um (caso "a") e dois (caso "b") veículos.

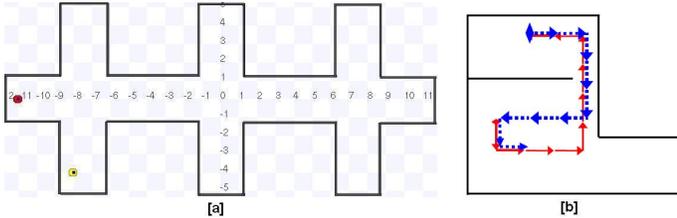


Fig. 4. [a] Um ambiente 23.5m x 10.5m explorado; e [b] Caminho (10 metros) em um ambiente contendo mesas, cadeiras, computadores, etc, percorrido pelos veículos 01 (linha tracejada) e 02 (linha contínua). A coleta de dados com um sensor *Kinect - Xbox* ocorreu a cada 1 metro .

No *Player/Stage*, veículos *Pioneer 3DX* tendo um sensor laser com campo de visão de  $180^\circ$  e alcance de 5m foi simulado. O laser foi utilizado na construção da grade de ocupação (células de tamanho 0.5m x 0.5m) usado na exploração (Fig.4[a]). A seleção de metas foi feita de acordo com o descrito na seção V. Após selecionar um destino, o caminho a ser percorrido pelo veículo foi computado usando o algoritmo *A star* [14] que foi configurado tal que ele escolhe (com maior probabilidade) trajetórias que estão distantes de obstáculos detectado na grade de ocupação. Estas trajetórias consistem de caminho em linha reta e na diagonal.

Na figura 6 temos a trajetória percorrida pelos veículos na tarefa de exploração. No caso (b), dois veículos exploraram o ambiente cerca da metade do número de passos. Os experimentos mostram que há uma redução média de 30% no tempo de exploração [12] (número de passos na grade) quando dois veículos e a estratégia descrita na seção V foi usada, estratégia que não leva em consideração na seleção de possíveis destinos, a proximidade de marcos já detectados, o que poderia melhorar a estimativa de localização dos veículos.

Note que no caso "b" o veículo 01 (02) passou perto de um caminho onde o veículo 02 (01) tinha passado (Fig.6). Portanto, a probabilidade de atualização de marcos e a acurácia na pose dos veículos poderia ser potencialmente aumentada, como será discutido abaixo.

No que se refere a abordagem SLAM utilizada, as coordenadas  $[\theta_x, \theta_y, \theta_z]^T$  dos marcos (características SIFT) foram obtidas com um sensor *Kinect - Xbox* (Fig.5). A distância percorrida pelo veículo ( $d$ ) e sua rotação ( $\varphi$ ) a cada passo e as coordenadas  $\theta_i$  dos marcos formam amostradas de acordo com a tabela I.

Para verificar a precisão na pose dos veículos, um caminho em um ambiente contendo cadeiras, mesas, computadores, etc, foi percorrida com um sensor *Kinect - Xbox* (Fig.4[b]). No

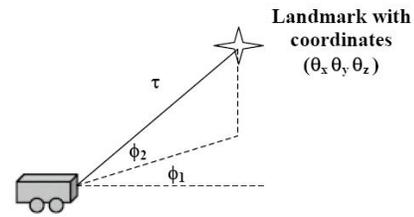


Fig. 5. Observação de um marco visual.

TABELA I

Amostragem dos modelos de movimento e de observação onde  $N(0, \sigma)$  é uma distribuição normal com média zero e desvio padrão  $\sigma$ . Valores estão em metros e radianos

$d = \bar{d} + \bar{d}N(0, \sigma)$	$\bar{d} = \frac{d}{10}$
$\varphi = \bar{\varphi} + \bar{\varphi}N(0, \sigma)$	$\bar{\varphi} = \frac{10\pi}{180}$
$\theta_i = \bar{\theta}_i + \bar{\theta}_iN(0, \sigma)$	$\bar{\theta}_i = 0.15$

caso (b), a precisão na pose dos dois veículos foi similar ou melhor do que o do caso (a). Isto pode ser constatado na tabela II. Note que  $E_0$  permite determinar hipoteticamente se um marco observado é um novo marco ou um marco previamente observado. Por outro lado,  $P_0$  permite refinar a associação de dados, baseado na probabilidade de associação, que é uma função da pose dos veículos e das coordenadas do marco.

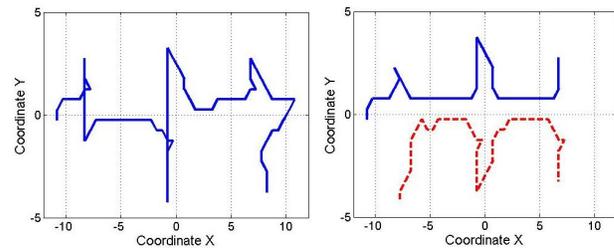


Fig. 6. Trajetória percorrida pelos veículos. Casos (a) e (b) respectivamente.

TABELA II

Erro (RMS) nas coordenadas  $x, y$  e orientação  $\alpha$ . Valores em metros e radianos. Parâmetros: [Partículas=2000,  $\sigma = 0.4$ ,  $E_0 = 0.8$  e  $P_0 = 0.1$ ]

Caso	$x_{min}$	$x_{max}$	$y_{min}$	$y_{max}$	$\alpha_{min}$	$\alpha_{max}$
a - veículo 01	0.02	0.06	0.02	0.04	0.18	0.34
a - veículo 02	0.04	0.10	0.03	0.13	0.39	0.64
b - veículo 01	0.02	0.03	0.03	0.04	0.11	0.31
b - veículo 02	0.02	0.06	0.03	0.06	0.17	0.31

Portanto, a combinação dos resultados - estratégia de exploração e abordagem SLAM - mostram que usando o mesmo número de partículas, a estratégia de exploração conduzida com um ou dois veículo produz uma acurácia similar ou melhor na pose. Era esperado que o aumento no número de veículos usados na exploração implicasse no aumento do número de partículas de forma a se conseguir um erro estimado similar na pose, o que se mostrou desnecessário.

Na exploração com dois veículos, a precisão obtida por cada veículo, em relação a precisão conseguida com somente um veículo, depende do número de atualizações de marcos feitas individualmente por cada veículo (Eq. 8). Este número tende a se manter na mesma proporção devido a quantidade de marcos que pode ser extraído com um algoritmo SIFT e pelo fato de que estas atualizações podem ser feitas em um marco previamente mapeado por qualquer veículo, desde que o mapa é comum a ambos os veículos.

Note que é importante manter o mesmo número de partículas na operação com dois veículos, desde que o algoritmo FastSLAM para dois veículos requer um tempo  $2M \log N$  ao invés de  $M \log N$  e a exploração é coordenada pelo agente central. Embora a adição de mais veículos aumente o custo computacional, a ordem de complexidade desta abordagem com múltiplos veículos permanece  $O(M \log N)$ . Portanto, podemos compensar este aumento com um maior poder computacional no agente central. Em relação a complexidade na extração de marcos com o algoritmo SIFT, devemos considerar que isto é feito individualmente por cada veículo.

## VII. CONCLUSÃO

Neste trabalho, mostramos que podemos explorar um ambiente interno com dois veículos de forma eficiente quando estamos usando o FastSLAM (com a obtenção de marco feitas pelo algoritmo SIFT) para localizá-los. Isto porque a estratégia de exploração coordenada o tempo usado na exploração em 30% [12] na média e porque os veículos mantêm ou melhoram a precisão na pose, tudo sobre a operação com somente um veículo. No FastSLAM, a precisão na pose é obtida porque a abordagem usa um mapa de características comum aos veículos e pela quantidade de marcos que podem ser obtidos com o algoritmo SIFT. Esta precisão obtida com o uso de um mesmo número de partículas, que poderia ser potencialmente maior uma vez que o problema estimado é maior. Dos resultados obtidos por simulação, pretendemos validar a abordagem através da exploração de um ambiente interno com veículos robóticos reais.

## ACKNOWLEDGMENT

Esta pesquisa é realizada sob uma concessão da FAPERJ (Fundação para Pesquisa do Estado do Rio de Janeiro).

## REFERENCES

- [1] A. M. Neto, P. F. F. Rosa and P. C. Pellanda (2010). Mapping semi-structured indoor environment with multiple vehicles with visual sensor embedded. *IV Brazilian Symposium on Inertial Engineering, Vol. 1, Issue 1, pp. 1-6*, Rio de Janeiro, RJ, Brazil.
- [2] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte and M. Csorba (2001). A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation, Vol. 17, Issue 3, pp. 229-241*.
- [3] M. Montemerlo and S. Thrun (2003). Simultaneous localization and mapping with unknown data association. *Proceedings of the IEEE, International Conference on Robotics and Automation, Vol. 2, pp. 1985-1991*.
- [4] C. Fulgenzi, G. Ippoliti, S. Longhi (2009). Experimental validation of FastSLAM algorithm integrated with a linear features based map. *Elsevier: Mechatronics, Vol. 19, Issue 5, pp. 609-616*

- [5] Kristopher R. Beevers (2007). Mapping with limited sensing. *Doctorate thesis of Rensselaer Polytechnic Institute, TROY, NY, USA*.
- [6] A. Gil, O. Reinoso, O. Martínez-Mozos, C. Stachniss, W. Burgard (2006). Improving Data Association in Vision-based SLAM. *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* pp. 2076-2081
- [7] A. Gil, O. Reinoso, M. Ballesta, M. Julia (2009). Multi-robot visual SLAM using rao-blackwellized particle filter. *ACM: Robotics and Autonomous Systems, Vol. 58, Issue 1, pp. 68-80*.
- [8] B. Stewart, J. Ko, D. Fox, K. Konolige (2003). A hierarchical bayesian approach to mobile robot map structure estimation. *Proceedings of the Conference on Uncertainty in AI, UAI,*
- [9] J.W. Fenwick, P.M. Newman, J.J. Leonard (2002). Cooperative concurrent mapping and localization. *IEEE Int. Conf. on Robotics and Automation, ICRA, pp. 1810-1817*
- [10] A. Howard (2006). Multi-robot simultaneous localization and mapping using particle filters. *International Journal of Robotics Research, Vol. 25, Issue 12, pp. 1243-1256*.
- [11] A. A. Makarenko, S. B. Willians, F. Bourgault, H. F. Durrant-Whyte (2002). An Experiment of Integrated Exploration. *Proceedings of the 2002 IEEE-RSJ. Intl Conference on Intelligent Robots and Systems, Vol. 1, pp. 534-539*.
- [12] W. Burgard, M. Moors, C. Stachniss and F. Schneider (2005). Coordinated Multi-Robot Exploration, *IEEE Transactions on Robotics, Vol. 21, Issue 3, pp. 376-386*.
- [13] S. Thrun, W. Burgard and D. Fox (2005). Probabilistic robotics. *Massachusetts Institute of Technology Press, Cambridge, MA, USA, 2005*.
- [14] Hart, P. E., Nilsson, N. J., Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics SSC4, Vol. 4, Issue 2, pp. 100-107*
- [15] David Lowe (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision, Vol. 2, pp. 91-110*
- [16] R. Howard (1960). Dynamic Programming and Markov Process. *MIT Press and Wiley*.
- [17] Brian P. Gerkey, Richard T. Vaughan and Andrew Howard (2003). The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. *In Proceedings of the International Conference on Advanced Robotics (ICAR 2003)*. pp. 317-323
- [18] Pedro Pinés, Juan D. Tardós (2008). Large Scale SLAM Building Conditionally Independent Local Maps: Application to Monocular Vision. *IEEE Transactions on Robotics. Vol. 24, NO. 5*