

# INFERÊNCIA NO PROBLEMA LINEAR DE SAM COM RESTRIÇÃO DE TOPOLOGIA

ANTONIO HENRIQUE PINTO SELVATICI\*, ANNA HELENA REALI COSTA\*

\**Laboratório de Técnicas Inteligentes — LTI  
Escola Politécnica da USP*

*Av. Prof. Luciano Gualberto, trav.3, n.158 Cidade Universitária  
05508-900 – São Paulo, SP, Brasil*

Emails: antonio.selvatici@poli.usp.br, anna.reali@poli.usp.br

**Abstract**— Robot mapping has been one of the main focus of attention in robotics research. It is specially hard to perform when an accurate robot localization is not available. In this case, it must use the map under construction in order to estimate its localization in the environment. Despite the vast literature addressing this issue, in the last years some graphical approaches have been taking place, concerned in jointly determining the environment map and the robot localization history. This problem is known as Smoothing and Mapping (SAM). This work proposes an algorithm for solving a linear version of the SAM problem with topological maps in an incremental, distributed and non-iterative fashion. Furthermore, map updating is performed exactly and in near-constant time. However, the algorithm can only be applied to a restrict domain of map topologies. Last, but not least, this work brings a new perspective for the interpretation of the SAM problem.

**Keywords**— SLAM, mapping, holonomic robots

**Resumo**— A tarefa de mapeamento é um dos principais focos de atenção das pesquisas em robótica, e se apresenta particularmente difícil quando não se conhece a posição do robô com exatidão, caso em que deve utilizar o próprio mapa que está sendo construído para se localizar no ambiente. Embora haja vasta literatura sobre esse assunto, nos últimos anos vêm surgindo abordagens gráficas que se preocupam em determinar, de forma conjunta, o mapa do ambiente e o histórico de localização do robô, problema conhecido pela expressão *Smoothing and Mapping* (SAM). Este trabalho propõe um algoritmo para resolver a versão linear do problema de SAM com mapas topológicos de forma incremental, distribuída, e não iterativa. Além disso, a atualização do mapa é realizada de forma exata e em tempo quase constante. No entanto, o algoritmo só pode ser aplicado a um domínio restrito de topologias. Como contribuição adicional, o trabalho traz uma nova perspectiva para a interpretação do problema de SAM.

**Palavras-chave**— SLAM, mapeamento, robôs holonômicos

## 1 Introdução

Mapeamento é o nome dado à tarefa de levantar um mapa do ambiente de forma autônoma por um robô móvel, utilizando seus sensores para esse fim. Muitas vezes, para que se possa obter o mapa esperado, é necessário que mapeamento e localização sejam realizados de forma conjunta, o que caracteriza o problema de mapeamento e localização simultâneos, conhecido como SLAM, acrônimo para *Simultaneous Localization and Mapping*. Resolver esse problema é considerado um dos grandes desafios da robótica (Dissanayake et al., 2001; Thrun, 2002; Folkesson et al., 2005).

Na literatura, muito já foi publicado a respeito do problema de SLAM. No entanto, desde o trabalho de Smith et al. (1990), que adaptou o problema de SLAM com mapas topológicos para ser resolvido utilizando o Filtro de Kalman Estendido (EKF), a maior parte dos trabalhos na área tem se baseado nesse filtro, procurando resolver suas limitações. Porém, algumas delas são intrínsecas à representação usada pelo EKF, e por isso necessitam de uma mudança na abordagem para serem superadas. Uma delas é que nenhuma associação entre uma leitura sensorial e um registro no mapa pode ser posteriormente desfeita.

Apesar de haver outras técnicas que resolvem implicitamente esse problema, elas ficam muito aquém do EKF quando o requisito é processamento em tempo real (Thrun et al., 1998). Tentando manter essa característica de baixo custo computacional e superar a limitação de não poder rever decisões tomadas no passado, recentemente surgiram abordagens em que todo o histórico de localização do robô e as associações com os marcos são mantidos em um mapa topológico, abordagens essas conhecidas como SLAM completo, ou ainda, SAM, um acrônimo para *Smoothing and Mapping* (Dellaert, 2005).

O objetivo deste trabalho é propor uma técnica incremental de inferência para o problema de SAM com modelo linear, que atualize o mapa em tempo constante, no caso médio, independentemente do seu tamanho. No entanto, esse tempo pode ser proporcional ao número de entradas do mapa quando o robô revisita áreas já mapeadas. Mesmo assim, a técnica proposta é mais rápida que as técnicas distribuídas propostas até agora para solucionar o problema de SAM, porém apresenta mais restrições para sua aplicação.

Neste artigo, a seção 2 apresenta alguns trabalhos correlatos, focalizando principalmente nas abordagens de SAM. Na seção 3 é apresentado o

modelo do problema tratado neste trabalho, enquanto a seção 4 traz uma interpretação desse modelo que ressalta algumas características de interesse. Com base nessas características, propõe-se um algoritmo de inferência, apresentado na seção 5. A seção 6 traz alguns resultados experimentais obtidos com o algoritmo proposto. Por fim, as conclusões do trabalho estão na seção 7.

## 2 Trabalhos correlatos

O primeiro modelo probabilístico para o problema de SLAM foi apresentado por Smith et al. (1990), segundo o qual consiste em manter uma estimativa do mapa do ambiente através do EKF, tendo como entrada as leituras sensoriais do robô. O estado do sistema é dado pela postura do robô e pelas posições de marcos específicos do ambiente, que formam um mapa topológico. Essa abordagem ganhou muita popularidade dado o modo direto como o problema é tratado e pela garantia teórica de convergência (Dissanayake et al., 2001), mas possui alguns problemas. Um deles é a falta de solução para o problema de correspondência (Thrun, 2002), que se traduz na necessidade de associar uma determinada leitura sensorial com um determinado marco, estando ele já registrado ou não no mapa. No entanto, já surgiram diversas abordagens de solução para esse problema (Zhang et al., 2005).

Uma vez que os algoritmos de filtragem mantêm a estimativa apenas do valor momentâneo do estado, descartando todo o histórico de estimação, decisões a respeito de associações entre leitura sensorial e marcos não podem ser revistas. Além do mais caso alguma associação errônea tenha sido executada, o erro de localização gerado pode levar o filtro a resultados inesperados, pois o EKF utiliza modelos linearizados em torno da localização mais provável. Por isso, mais recentemente surgiram abordagens de solução que trabalham explicitamente com a representação gráfica completa do problema, estimando todo histórico de localização bem como as posições dos marcos, o que caracteriza o problema de SAM. Folkesson et al. (2005) apresentaram uma solução para esse problema, representando-o como um problema de minimização de energia. No entanto, a solução apresentada necessita de reprocessar todo o grafo para obter o mapa.

Dellaert (2005) aprofundou o estudo do problema. Com a linearização dos modelos de movimento e de observação do robô, a modelagem do problema toma a forma de um sistema de equações lineares. Dellaert (2005), então, apresenta uma solução exata e eficiente para o sistema de equações resultante com base na fatoração da matriz de coeficientes  $A$  do sistema. A partir desse estudo, surgiram duas abordagens incrementais de solução para o problema de SAM. A primeira (Kaess

et al., 2007) propõe a solução incremental do sistema inserindo novas entradas na matriz  $R$ , que corresponde ao fator de Cholesky da matriz  $A$ . Então, a cada passo, apenas as variáveis mais recentes são recalculadas, pois são as que foram afetadas mais significativamente pela nova informação, de forma que a atualização do mapa é feita em tempo constante. Isso apenas não ocorre quando há uma associação da observação com um marco já registrado no mapa, tornando o estado corrente muito correlacionado com outros estados do passado. No entanto, esse problema pode ser resolvido com uma reordenação eventual das variáveis no sistema de equações, que é feita em tempo linear com o número de variáveis.

A segunda abordagem de solução (Ranganathan et al., 2007) baseia-se na construção de um campo markoviano aleatório (MRF) a partir das equações do problema linearizado de minimização de energia, utilizando o algoritmo de propagação de crenças em ciclos para calcular a melhor estimativa do mapa. Esse algoritmo possui garantia de encontrar corretamente os valores dos estados em redes probabilísticas — como é o caso de um MRF — desde que as probabilidades constituintes sejam todas gaussianas (Weiss e Freeman, 1999). Além disso, na maior parte das vezes, a atualização das crenças é feita em tempo constante, o que só não ocorre quando um marco é detectado novamente, resultando no fechamento de um ciclo, caso em que a atualização do mapa é realizada em tempo linear.

A principal vantagem do algoritmo de propagação de crença é sua abordagem fortemente distribuída, pois baseia-se na troca de mensagens entre nós de um grafo. No entanto, em comparação com a abordagem centralizada de resolução do sistema de equações, a sua principal desvantagem é ser um algoritmo iterativo. Assim, além da necessidade da elaboração de um critério de parada, o algoritmo pode não convergir se, por exemplo, os modelos envolvidos não tiverem sido linearizados em torno dos pontos adequados. Então, mesmo que a complexidade de ambos seja compatível, o algoritmo centralizado resulta por ser mais rápido por evitar múltiplas iterações, o que pode ser bastante demorado.

Analogamente ao proposto por Ranganathan et al. (2007), a técnica incremental para inferência no problema de SAM proposta neste trabalho se baseia na troca de mensagens entre nós de um grafo, caracterizando uma abordagem distribuída. Além disso, o histórico da localização do robô e as posições dos marcos são determinados de modo direto, sem a necessidade de múltiplas iterações, assim como no trabalho de Kaess et al. (2007). No entanto, a abordagem aqui proposta requer restrições na topologia do mapa, o que, apesar de serem obtidas facilmente através de podas no grafo, representam relevantes perdas de informação.

### 3 Modelo linear do problema de SAM

Seja um robô holonômico que se move no plano cartesiano. A cada instante  $t$ , ele se encontra na posição  $\mathbf{x}_t = (x_t^R, y_t^R)$  e executa um comando de movimento  $\mathbf{u}_t = (\Delta x_t^R, \Delta y_t^R)$ . Possivelmente, também observa um marco  $L_k$ ,  $1 < k \leq M$ , cuja posição é  $\mathbf{l}_k = (x_k^L, y_k^L)$ , a uma posição relativa  $\mathbf{z}_t = (\Delta x_t^L, \Delta y_t^L)$ . Assumindo que o movimento do robô e sua observação são perturbados por ruído branco, tem-se o seguinte modelo para a evolução da posição do robô e para as observações:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{u}_t + \mathbf{v}_t \quad \text{e} \quad \mathbf{z}_t = \mathbf{l}_{k_t} - \mathbf{x}_t + \mathbf{w}_t, \quad (1)$$

onde  $\mathbf{l}_{k_t}$  é a posição do marco observado no instante  $t$ ,  $\mathbf{v}_t$  é o ruído branco com covariância  $Q_t$  que perturba o movimento do robô no instante  $t$ , e  $\mathbf{w}_t$  é o ruído branco com covariância  $R_t$  que perturba a observação do robô no instante  $t$ . A limitação de apenas uma observação por posição é um recurso para facilitar as análises do modelo, pois várias observações feitas ao mesmo tempo podem ser representadas por observações feitas em sucessivos momentos nos quais o movimento do robô tenha sido nulo.

A resolução do problema de inferência em SAM equivale a encontrar os valores das variáveis não observadas,  $\mathbf{x}_t$  e  $\mathbf{l}_{k_t}$ , com  $t = 1, 2, \dots, T$  e  $k = 1, 2, \dots, M$ , de maior probabilidade conjunta. Para o problema de inferência, considera-se que todas as associações entre observações e marcos no ambiente são conhecidas, ou seja, sabe-se previamente qual marco  $L_k$  é observado no instante  $t$ . Embora a solução dos problemas de SLAM do mundo real requeira, na maior parte das vezes, a determinação dessas associações, essa tarefa é realizada através outros métodos, empregados de forma concorrente com a técnica de inferência. Dessa forma, a solução do problema satisfaz:

$$\theta^* = \arg \min_{\theta} \left\{ \frac{1}{2} \|\mathbf{z}_0 - (\mathbf{l}_{k_0} - \mathbf{x}_0)\|_{R_0}^2 + \frac{1}{2} \sum_{t=1}^T \left[ \|\mathbf{x}_t - (\mathbf{x}_{t-1} + \mathbf{u}_{t-1})\|_{Q_{t-1}}^2 + \|\mathbf{z}_t - (\mathbf{l}_{k_t} - \mathbf{x}_t)\|_{R_t}^2 \right] \right\}, \quad (2)$$

onde  $\theta = (\mathbf{x}_{[1:T]}, \mathbf{l}_{[1:M]})$  é o vetor de parâmetros para otimizar. Uma vez que a função a minimizar é quadrática e não possui restrições adicionais, a solução é encontrada quando o gradiente da função a ser minimizada iguala zero, resultando em:

$$-Q_0^{-1}(\mathbf{x}_1 - (\mathbf{x}_0 + \mathbf{u}_0)) - R_0^{-1}(\mathbf{l}_{k_0} - (\mathbf{x}_0 + \mathbf{z}_0)) = 0 \quad (3)$$

$$Q_{t-1}^{-1}(\mathbf{x}_t - (\mathbf{x}_{t-1} + \mathbf{u}_{t-1})) - Q_t^{-1}(\mathbf{x}_{t+1} - (\mathbf{x}_t + \mathbf{u}_t)) - R_t^{-1}(\mathbf{l}_{k_t} - (\mathbf{x}_t + \mathbf{z}_t)) = 0 \quad (4)$$

$$\sum_{t|k_t=k} R_t^{-1}(\mathbf{l}_{k_t} - (\mathbf{x}_t + \mathbf{z}_t)) = 0, \quad (5)$$

para  $t = 1, 2, \dots, T$  e  $k = 1, 2, \dots, M$ .

### 4 Representação circuital do problema

As eqs. (3-5) sugerem um modo interessante de representação do problema. Sejam os vetores

$$\mathbf{f}_{\mathbf{x}_t \mathbf{x}_{t+1}} = -Q_t^{-1}(\mathbf{x}_{t+1} - (\mathbf{x}_t + \mathbf{u}_t)) \quad (6)$$

$$\mathbf{f}_{\mathbf{x}_t \mathbf{l}_{k_t}} = -R_t^{-1}(\mathbf{l}_{k_t} - (\mathbf{x}_t + \mathbf{z}_t)) \quad (7)$$

onde  $\mathbf{f}_{\mathbf{ab}}$  é definido como o *vetor força que a aplica em b*. De forma análoga, define-se  $\mathbf{f}_{\mathbf{ba}} = -\mathbf{f}_{\mathbf{ab}}$ . Assim, (3-5) tornam-se equações de equilíbrio de forças, e (6) e (7) correspondem a forças da forma  $\mathbf{f}_{\mathbf{ab}} = -K(\mathbf{b} - (\mathbf{a} + \mathbf{c}))$ , equivalente à força de  $\mathbf{a}$  em  $\mathbf{b}$  quando entre eles há uma mola com constante  $K$  e comprimento e orientação dados pelo vetor  $\mathbf{c}$ . Assim, realizar a inferência nesse modelo corresponde a encontrar as posições de equilíbrio de forças em um circuito de molas onde as posições dos nós representam as variáveis de interesse, conforme ilustrado na figura 1. Em circuitos de molas contendo apenas ciclos aninhados, isto é, ciclos que estão totalmente contidos dentro de outros ciclos, isso pode ser conseguido utilizando-se os conceitos de ligações em série e em paralelo.

Seja uma seqüência de movimentos e observações que geram o grafo representado na figura 2(a), que contém apenas ciclos aninhados. Define-se como nó de início de um ciclo o primeiro que está ligado ao marco que, por ter sido observado em duas posições diferentes pelo robô, gerou aquele ciclo. Assim, no caso do grafo da figura 2(a), o ciclo interno se inicia em  $\mathbf{x}_2$  e termina em  $\mathbf{x}_3$ , enquanto o externo se inicia em  $\mathbf{x}_1$  e termina em  $\mathbf{x}_4$ . Através do conceito de ligação em série e em paralelo, é possível estimar primeiro os valores das variáveis representadas nos nós dos ciclos mais externos, para então calcular os valores das variáveis nos nós dos ciclos internos.

Os passos da solução estão indicados na figura 2. Primeiramente, devem-se eliminar os ciclos mais internos, determinando a ligação equivalente entre seus nós final e inicial, como representado na figura 2(b). Esse procedimento é aplicado recursivamente, até que resultar em um grafo acíclico, como o da figura 2(c). Como o valor de  $\mathbf{x}_0$  é conhecido, as variáveis representadas nesse grafo são diretamente estimadas, calculando-se o comprimento e a orientação da mola equivalente entre

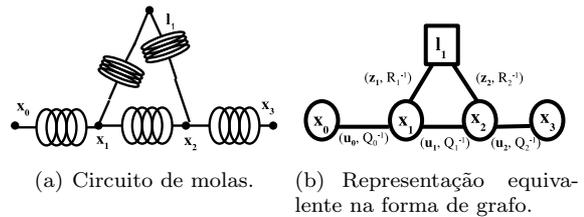


Figura 1: Representação circuital de um problema de SAM linear, no qual o robô percorreu 4 posições. Nas posições 1 e 2 ele observou o marco 1.

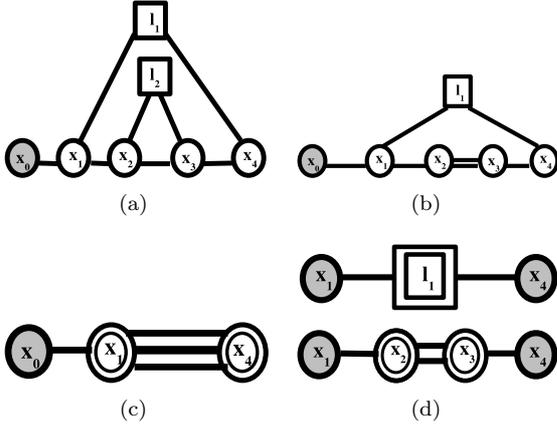


Figura 2: Exemplo de inferência em um grafo com ciclos aninhados. Os nós de cor cinza representam as variáveis de valor já estimado, e os nós desenhados com traço duplo são aqueles que estão tendo o valor estimado.

o nó de  $\mathbf{x}_0$  e o nó de cada variável. Com isso, as variáveis nas extremidades dos ciclos mais externos, cujos valores foram estimados, permitem a estimação das outras variáveis desses ciclos através da equação de equilíbrio de forças (3), (4) ou (5), conforme o caso. A figura 2(d) ilustra esse procedimento, que deve ser executado até que a estimativa de todas as variáveis sejam determinadas. Esse método pode ser generalizado para topologias com qualquer número de ciclos aninhados e ciclos em série.

## 5 Algoritmo proposto

O algoritmo de Propagação de Molas é a proposta deste trabalho para a execução incremental e distribuída do procedimento descrito acima, através da propagação de mensagens entre os nós vizinhos do grafo. Seja  $M_\xi^\nu$  a mensagem enviada pelo nó de  $\xi$  ao nó de  $\nu$ , onde  $\xi$  e  $\nu$  são variáveis vizinhas no grafo do problema de SAM, e podem ser tanto posições assumidas pelo robô ( $\mathbf{x}_t$ ) quanto posições de marcos no ambiente ( $\mathbf{l}_k$ ). Cada mensagem é composta por uma tripla:  $M_\xi^\nu = \{\hat{\xi}, \mathbf{c}_\xi^\nu, K_\xi^\nu\}$ , onde  $\hat{\xi}$  é a estimativa do valor da variável de origem da mensagem,  $\xi$ ,  $\mathbf{c}_\xi^\nu$  é o comprimento e orientação da mola equivalente entre os nós de  $\xi$  e  $\nu$ , e  $K_\xi^\nu$  é a constante dessa mesma mola. Os parâmetros da mola são dados por:

- $\mathbf{c}_{x_t}^{x_{t+1}} = -\mathbf{c}_{x_{t+1}}^{x_t} = \mathbf{u}_t$  e  $K_{x_t}^{x_{t+1}} = K_{x_{t+1}}^{x_t} = Q_t^{-1}$
- $\mathbf{c}_{x_t}^{l_{k_t}} = -\mathbf{c}_{l_{k_t}}^{x_t} = \mathbf{z}_t$  e  $K_{x_t}^{l_{k_t}} = K_{l_{k_t}}^{x_t} = R_t^{-1}$ .

O procedimento de atualização dos valores das variáveis do grafo pelo algoritmo é incremental, realizado após cada inserção de novas medidas de deslocamento e de observação no mapa. Considere-se a atualização do mapa no instante  $t$ .

Caso o marco observado nesse instante já tenha sido observado anteriormente, no instante  $t - \tau$ , é formado um ciclo que se inicia no nó de  $\mathbf{x}_{t-\tau}$  e finaliza no nó de  $\mathbf{x}_t$ . A variável representada no nó inicial desse ciclo não sofre modificação de estimativa, pois a mola equivalente entre ele e o nó de  $x_0$  não se modifica. Portanto,  $\mathbf{x}_{t-\tau}$  deve propagar sua estimativa para os demais nós do ciclo. Como os nós dessas variáveis não são sempre vizinhos no grafo, mensagens partindo de  $\mathbf{x}_{t-\tau}$  devem ser propagadas para todos os nós do ciclo. No entanto, como há vários caminhos que uma mensagem pode seguir, ela deve se desdobrar em duas a cada bifurcação encontrada. Do mesmo modo, quando duas mensagens confluem para um mesmo nó, elas devem se unificar, trazendo a informação de em apenas uma mola equivalente.

Quando uma mensagem se desdobra em duas, cada mensagem resultante é constituída de uma nova mensagem empilhada sobre a mensagem original, sendo denominada mensagem composta. Dessa forma,  $\overleftarrow{M}_\xi^{\nu'} = M_\xi^\nu | \overleftarrow{M}_\nu^{\nu'}$  e  $\overrightarrow{M}_\xi^{\nu''} = M_\xi^\nu | \overrightarrow{M}_\nu^{\nu''}$  são as mensagens compostas resultantes do desdobramento de  $M_\xi^\nu$ , onde  $|$  representa a operação de empilhamento e as setas em sobreposição indicam o sentido do fluxo da mensagem no grafo. A mensagem representada mais à direita está no topo da pilha, sendo a única parte da mensagem composta a ser modificada quando ocorre sua propagação de um nó para outro. A propagação de uma mensagem composta  $M_\xi^{\nu'} = M_\xi^\nu | M_\nu^{\nu'}$  de  $\nu'$  para  $\bar{\nu}$  resulta em  $M_\xi^{\bar{\nu}} = M_\xi^\nu | M_\nu^{\bar{\nu}}$ , com  $\mathbf{c}_\xi^{\bar{\nu}} = \mathbf{c}_\nu^{\nu'} + \mathbf{c}_\nu^{\bar{\nu}}$ , e  $K_\xi^{\bar{\nu}} = K_\nu^{\nu'} (K_\nu^{\nu'} + K_\nu^{\bar{\nu}})^{-1} K_\nu^{\bar{\nu}}$ .

Na unificação de duas mensagens compostas confluentes, as únicas partes que diferem nessas mensagens são as mensagens empilhadas que estão nos topos das pilhas. Por isso, apenas as molas representadas por essas mensagens são unificadas em uma mola equivalente, como se fossem duas molas em paralelo. Posteriormente, a mola resultante é unificada com a mola representada pelo restante da mensagem, como em uma ligação em série. A unificação é representada por:

$$\left. \begin{array}{l} \overleftarrow{M}_\xi^{\xi'} = M_\xi^\nu | \overleftarrow{M}_\nu^{\xi'} \\ \overrightarrow{M}_\xi^{\xi'} = M_\xi^\nu | \overrightarrow{M}_\nu^{\xi'} \end{array} \right\} M_\xi^{\xi'} = M_\xi^\nu | M_\nu^{\xi'}, \quad (8)$$

onde  $\mathbf{c}_\xi^{\xi'} = \mathbf{c}_\xi^\nu + \mathbf{c}_\nu^{\xi'}$ ,  $K_\xi^{\xi'} = K_\xi^\nu (K_\xi^\nu + K_\nu^{\xi'})^{-1} K_\nu^{\xi'}$ ,  $\mathbf{c}_\nu^{\xi'} = (K_\nu^{\xi'})^{-1} (\overleftarrow{K}_\nu^{\xi'} \overleftarrow{\mathbf{c}}_\nu^{\xi'} + \overrightarrow{K}_\nu^{\xi'} \overrightarrow{\mathbf{c}}_\nu^{\xi'})$ , e  $K_\nu^{\xi'} = \overleftarrow{K}_\nu^{\xi'} + \overrightarrow{K}_\nu^{\xi'}$ . A unificação só pode ser executada quando ambas as mensagens chegam ao nó de confluência. Quando o nó recebe apenas uma das mensagens, deve guardá-la consigo até receber a outra.

Um caso especial da unificação de mensagens ocorre quando elas não são compostas. Nesse caso, as origens das mensagens podem ser diferentes, e o valor da variável referente ao nó de confluência pode ser diretamente estimado através da equa-

ção:

$$\hat{\nu} = (\overleftarrow{K}_{\xi'}^{\nu} + \overrightarrow{K}_{\xi''}^{\nu})^{-1} (\overleftarrow{K}_{\xi'}^{\nu} (\hat{\xi}' + \overleftarrow{c}_{\xi'}^{\nu}) + \overrightarrow{K}_{\xi''}^{\nu} (\hat{\xi}'' + \overrightarrow{c}_{\xi''}^{\nu})) \quad (9)$$

onde  $\nu$  representa a variável que recebe as mensagens confluentes  $\overrightarrow{M}_{\xi'}^{\nu}$  e  $\overleftarrow{M}_{\xi''}^{\nu}$ , não compostas.

O algoritmo de Propagação de Molas utiliza os mecanismos de desdobramento, propagação e unificação de mensagens para estimar os valores dentro do ciclo formado em um passo de atualização do mapa:

1. **Início:**

- (a) O nó de  $\mathbf{x}_{t-\tau}$  envia  $\overrightarrow{M}_{x_{t-\tau}}^{x_{t-\tau+1}}$  e  $\overleftarrow{M}_{x_{t-\tau}}^{l_{k_{t-\tau}}}$
- (b) O nó de  $\mathbf{x}_{t+1}$  recebe  $\overleftarrow{M}_{x_0}^{x_{t+1}}$ , com  $\overleftarrow{c}_{x_0}^{x_{t+1}} = 0$  e  $\overrightarrow{K}_{x_0}^{x_{t+1}} = 0$

2. **Para cada** variável  $\nu$  que recebeu  $M_{\xi}^{\nu}$  de  $\bar{\nu}$ :

- (a) **Se**  $\nu$  já foi estimada, **então** nada faz
- (b) **Senão**, se  $\nu = \mathbf{x}_i$  está no início de um ciclo, e  $\bar{\nu} = \mathbf{x}_{i-1}$ ,  $i \in \mathbb{Z}$ , **então**:
  - i.  $\nu$  envia  $\overrightarrow{M}_{\xi}^{x_{i+1}} = M_{\xi}^{\nu} | \overrightarrow{M}_{x_i}^{x_{i+1}}$
  - ii.  $\nu$  envia  $\overleftarrow{M}_{\xi}^{l_{k_i}} = M_{\xi}^{\nu} | \overleftarrow{M}_{x_i}^{l_{k_i}}$
  - iii. **Se**  $M_{\xi}^{\nu}$  não for composta, **então**  $\nu$  guarda-a como  $\overrightarrow{M}_{\xi}^{x_i}$
- (c) **Senão**, se  $\nu = \mathbf{x}_i$  está no final de um ciclo, e  $\bar{\nu} = \mathbf{x}_{i+1}$ ,  $i \in \mathbb{Z}$ , **então**:
  - i.  $\nu$  envia  $\overleftarrow{M}_{\xi}^{x_{i-1}} = M_{\xi}^{\nu} | \overleftarrow{M}_{x_i}^{x_{i-1}}$
  - ii.  $\nu$  envia  $\overrightarrow{M}_{\xi}^{l_{k_i}} = M_{\xi}^{\nu} | \overrightarrow{M}_{x_i}^{l_{k_i}}$
  - iii. **Se**  $M_{\xi}^{\nu}$  não for composta, **então**  $\nu$  guarda-a como  $\overleftarrow{M}_{\xi}^{x_i}$
- (d) **Senão**, se  $\nu = \mathbf{x}_i$ ,  $i \in \mathbb{Z}$ , está no final de um ciclo, **então**:
  - i. **Se**  $\bar{\nu} = \mathbf{x}_{i-1}$ , **então**  $\nu$  guarda  $M_{\xi}^{\nu}$  como  $\overrightarrow{M}_{\xi}^{x_i}$
  - ii. **Senão**,  $\nu$  guarda  $M_{\xi}^{\nu}$  como  $\overleftarrow{M}_{\xi}^{x_i}$
  - iii. **Se** existem  $\overrightarrow{M}_{\xi}^{x_i}$  e  $\overleftarrow{M}_{\xi}^{x_i}$ , **então**:
    - A. Unifica  $\overrightarrow{M}_{\xi}^{x_i}$  e  $\overleftarrow{M}_{\xi}^{x_i}$  em  $\overrightarrow{M}_{\xi}^{x_i}$
    - B. Propaga  $\overrightarrow{M}_{\xi}^{x_i}$  para  $\mathbf{x}_{i+1}$
    - C. **Se**  $\overrightarrow{M}_{\xi}^{x_i}$  não for composta, **então**  $\nu$  guarda-a
- (e) **Senão**, se  $\nu = \mathbf{x}_i$ ,  $i \in \mathbb{Z}$ , está no início de um ciclo, **então**:
  - i. **Se**  $\bar{\nu} = \mathbf{x}_{i+1}$ , **então**  $\nu$  guarda  $M_{\xi}^{\nu}$  como  $\overleftarrow{M}_{\xi}^{x_i}$
  - ii. **Senão**,  $\nu$  guarda  $M_{\xi}^{\nu}$  como  $\overrightarrow{M}_{\xi}^{x_i}$
  - iii. **Se** existem  $\overrightarrow{M}_{\xi}^{x_i}$  e  $\overleftarrow{M}_{\xi}^{x_i}$ , **então**:
    - A. Unifica  $\overrightarrow{M}_{\xi}^{x_i}$  e  $\overleftarrow{M}_{\xi}^{x_i}$  em  $\overleftarrow{M}_{\xi}^{x_i}$

B. Propaga  $\overleftarrow{M}_{\xi}^{x_i}$  para  $\mathbf{x}_{i-1}$

C. **Se**  $\overleftarrow{M}_{\xi}^{x_i}$  não for composta, **então**  $\nu$  guarda-a

(f) **Senão**,

- i.  $\nu$  propaga  $M_{\xi}^{\nu}$  para o vizinho, que não  $\bar{\nu}$
- ii. **Se**  $M_{\xi}^{\nu}$  não for composta,  $\nu$  guarda-a como  $\overrightarrow{M}_{\xi}^{\nu}$  ou  $\overleftarrow{M}_{\xi}^{\nu}$ , conforme o caso

(g) **Se** existem  $\overrightarrow{M}_{\xi'}^{\nu}$  e  $\overleftarrow{M}_{\xi''}^{\nu}$ , **então**:

- i. Atualiza  $\nu$
- ii. **Se**  $\nu = \mathbf{x}_i$  está no início de um ciclo, **então**
  - A.  $\nu$  envia  $\overrightarrow{M}_{\nu}^{x_{i+1}}$
  - B.  $\nu$  envia  $\overleftarrow{M}_{\nu}^{l_{k_i}}$
- iii. **Se**  $\nu = \mathbf{x}_i$  está no final de um ciclo, **então**
  - A.  $\nu$  envia  $\overleftarrow{M}_{\nu}^{x_{i-1}}$
  - B.  $\nu$  envia  $\overrightarrow{M}_{\nu}^{l_{k_i}}$

Uma vez que as operações de propagação, desdobramento e unificação possuem complexidade temporal constante, a atualização do mapa possui complexidade temporal linear com relação o número de nós do ciclo que está sendo atualizado. No entanto, considerando que, na maior parte do tempo, os ciclos formados são pequenos, ou não há ciclos, o tempo de atualização é aproximadamente constante com relação ao número de variáveis do grafo, no caso médio.

## 6 Resultados experimentais

Foram realizados experimentos com o algoritmo de SAM apresentado através do simulador de robôs Player/Stage (Gerkey et al., 2003). Foi criado um ambiente no qual está inserido um robô holonômico dotado um detector de marcos e *encoders* nas rodas, que fornecem a sua odometria. No experimento apresentado, o robô executa um movimento cíclico de exploração do ambiente, controlado por um operador humano, enquanto captura as posições relativas dos marcos observados e a leitura dos odômetros. Foi adotado um espaço de observações no qual os marcos são todos distintos entre si. O grafo representando o problema de SLAM foi podado de forma a apresentar apenas ciclos aninhados.

A figura 3 mostra a trajetória real executada pelo robô dentro do ambiente, bem como as posições dos marcos detectados. A figura 4 mostra a trajetória do robô corrigida pelo algoritmo de Propagação de Molas sobreposta à trajetória obtida apenas por odometria. Nota-se que o algoritmo elimina as falsas sobreposições que aparecem na trajetória determinada pela odometria, fornecendo a real topologia do ambiente em que foi realizado o experimento.

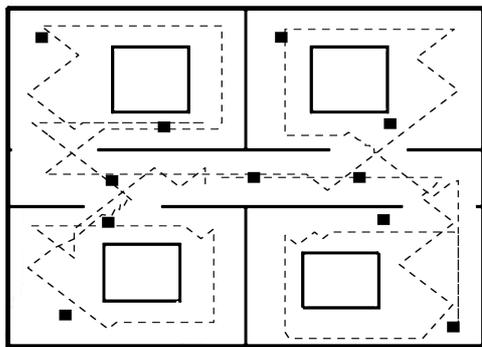


Figura 3: Cenário no qual foi realizada a simulação, com a trajetória do robô indicada. Os quadrados em preto correspondem às posições dos marcpos posicionados no ambiente.

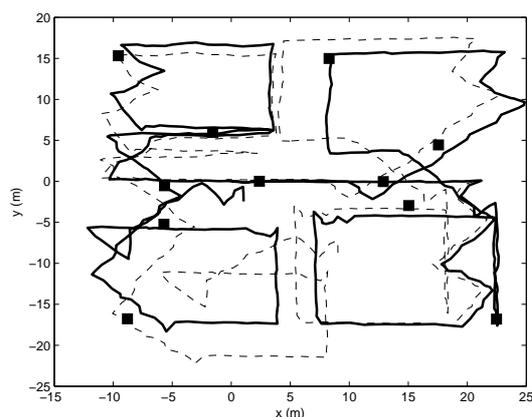


Figura 4: Resultado do alinhamento da trajetória pelo algoritmo de SAM proposto. O traço contínuo indica a trajetória obtida pelo algoritmo de SAM, enquanto o traço tracejado indica a trajetória determinada apenas a partir da odometria.

## 7 Conclusões

Este artigo propôs um algoritmo incremental que, a cada passo, resolve o problema de SAM de forma ótima e em tempo constante com relação ao tamanho do mapa, no caso médio, evitando o reprocessamento de todos os nós do grafo resultante. Apesar de restritiva, a restrição de topologia exigida pelo algoritmo de propagação de molas permite que o robô reduza consideravelmente o erro de sua trajetória ao observar novamente marcpos do início da execução. Esse retorno ao início da trajetória para observar marcpos cuja posição é conhecida com mais acurácia é a base da tarefa localização em SLAM. No entanto, é possível relaxar a restrição de topologia do algoritmo caso se admita obter uma trajetória sub-ótima, o que deverá ser investigado em trabalhos futuros. Além disso, propõe-se também a adaptação do algoritmo para o caso de robôs não-holonômicos, quando o problema de otimização de trajetória torna-se não-linear.

## Agradecimento

Antonio Henrique P. Selvatici agradece o apoio do CNPq, proc. n. 1408682005-4.

## Referências

- Dellaert, F. (2005). Square root SAM, *Proceedings of Robotics: Science and Systems*, Cambridge, USA.
- Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H. F. e Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem, *IEEE Trans. on Robotics and Automation* **17**(3): 229–241.
- Folkesson, J., Jensfelt, P. e Christensen, H. I. (2005). Graphical SLAM using vision and the measurement subspace, *Proc. of the IEEE/RSJ Int. Joint Conf. on Intelligent Robots and Systems (IROS)*, pp. 325–330.
- Gerkey, B., Vaughan, R. T. e Howard, A. (2003). The Player/Stage Project: Tools for multi-robot and distributed sensor systems, *Proc. of the 11th Int. Conf. on Advanced Robotics (ICAR'03)*, Coimbra, Portugal, pp. 317–323. Página na Internet: <http://playerstage.sourceforge.net>.
- Kaess, M., Ranganathan, A. e Dellaert, F. (2007). Fast incremental square root information smoothing, *Proc. of the 20th. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Hyderabad, India, pp. 2129–2134.
- Ranganathan, A., Kaess, M. e Dellaert, F. (2007). Loopy SAM, *Proc. of the 20th. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Hyderabad, India, pp. 2191–2196.
- Smith, R., Self, M. e Cheeseman, P. (1990). Estimating uncertain spatial relationships in robotics, *Autonomous Robot Vehicles* pp. 167–193.
- Thrun, S. (2002). Robotic mapping: A survey, *Exploring Artificial Intelligence in the New Millennium*, Morgan Kaufmann.
- Thrun, S., Burgard, W. e Fox, D. (1998). A probabilistic approach to concurrent mapping and localization for mobile robots, *Machine Learning* **31**: 29–53.
- Weiss, Y. e Freeman, W. T. (1999). Correctness of belief propagation in Gaussian graphical models of arbitrary topology, *Advances in Neural Information Processing Systems (NIPS)* pp. 673–679.
- Zhang, S., Xie, L. e Adams, M. (2005). An efficient data association approach to simultaneous localization and map building, *Int. Journal of Robotics Research* **24**(1).