

ESTIMAÇÃO E CONTROLE DA POSIÇÃO DE UM ROBÔ MÓVEL UTILIZANDO FILTRO DE KALMAN DESCENTRALIZADO

FELIPE KÜHNE*, CARLOS A. H. CLARO*, SÉRGIO R. SUESS*, WALTER F. LAGES*

**Universidade Federal do Rio Grande do Sul
Departamento de Engenharia Elétrica
Av. Oswaldo Aranha, 103 – CEP 90035-190
Porto Alegre, RS, Brasil*

Emails: kuhne@eletro.ufrgs.br, caclaro@eletro.ufrgs.br, suess@eletro.ufrgs.br, fetter@eletro.ufrgs.br

Abstract— This paper proposes a full-connected, decentralized control architecture for a wheeled mobile robot. Three different types of sensors are used. The system is distributed in four nodes, in a way that the controller and the sensors are totally independent from each other, giving the architecture the capability to handle individual sensors' failures. An extended Kalman filter (EKF) is used for state estimation and data fusion. The trajectory tracking problem is solved through a feedback linearization and a linear control law for the robot's position. Simulation results are shown for two different trajectories.

Keywords— Mobile robots, Kalman filtering, linearizing feedback control, decentralized control.

Resumo— Este artigo propõe uma arquitetura descentralizada totalmente conectada para o controle de um robô móvel dotado de rodas. Três tipos diferentes de sensores são utilizados. O sistema é distribuído em quatro nodos, de forma que o controlador e os sensores são totalmente independentes, conferindo ao sistema tolerância a falhas individuais dos sensores. Um filtro de Kalman extendido (EKF) é utilizado para a estimação dos estados e fusão de dados. O problema de rastreamento de trajetória é resolvido através de uma realimentação linearizante e de uma lei de controle linear para a posição do robô. Resultados de simulação são mostrados para dois tipos de trajetórias.

Palavras-chave— Robôs móveis, filtro de Kalman, controle por realimentação linearizante, controle descentralizado.

1 Introdução

No problema de navegação e controle de um robô móvel autônomo, uma das principais informações que deve ser obtida é a atual postura (posição e orientação) deste robô com relação a algum sistema de coordenadas global. Para isso, utilizam-se sensores acoplados ao veículo. De uma forma geral, estes sensores podem ser classificados em sensores de medida *relativa* ou *absoluta*.

No caso de robôs com rodas, é comum o uso de encoders incrementais acoplados aos eixos de rotação das rodas, na técnica denominada *odometria* (Borenstein et al., 1996). Encoders incrementais são bastante utilizados pois são baratos e de fácil implementação. Entretanto, a idéia básica da odometria é a integração da informação da movimentação em um dado período de tempo, o que inevitavelmente leva ao acúmulo ilimitado de erros (Kang et al., 1995; Park et al., 1996). Para limitar estes erros, faz-se uso de sensores de medida absoluta e com domínios de informação sobrepostos, como aparelhos de GPS e bússolas eletrônicas (Medromi et al., 1996; Sandi et al., 1998). A vantagem de sensores de medida absoluta é que os mesmos não sofrem do problema de integração do erro, visto que, em sensores de medida relativa, o erro é polarizado e integrado no tempo. Por outro lado, a informação proveniente destes sensores também pode ser deturpada, seja, por exemplo, por causa de condições atmosféricas adversas (no

caso de aparelhos de GPS) ou por distorções no campo magnético terrestre (no caso de bússolas). No entanto estes erros são absolutos e não são integrados.

No sistema em questão neste trabalho, são utilizados três tipos de sensores, com diferentes tecnologias e com domínios de informação sobrepostos, em uma rede de nodos totalmente conectados. Esta sobreposição dos domínios confere ao sistema a capacidade de *graceful degradation*, ou seja, a falha de um sensor não significa a falha de todo o sistema, pois os outros sensores podem ser utilizados para medir a informação do sensor em falha (Yajnik and Jha, 1997; Mutambara, 1998).

Para que uma arquitetura descentralizada possa ser vantajosa, é essencial que a informação dos diversos sensores seja combinada de forma a obter-se uma descrição completa, confiável e coerente do sistema. Em um sistema descentralizado, a fusão dos dados ocorre em cada nodo com base na observação local e na informação transmitida pelos outros nodos. Assim, levando em consideração a sobreposição dos domínios de informação dos sensores e realizando a fusão de dados, é possível reduzir a incerteza inerente a cada sensor, possibilitando a obtenção de estimativas mais precisas do que as leituras individuais dos sensores (Vestli, 1995).

Sendo assim, neste trabalho utilizou-se o filtro de Kalman para a estimação dos estados (obten-

ção da estimativa *local*, correspondente apenas à informação de um determinado nodo) e para fusão de dados (obtenção da estimativa *global* dos estados a partir de informações locais).

Três sensores estão presentes no sistema: encoder, GPS e bússola. Apresenta-se uma estrutura descentralizada totalmente conectada composta de um nodo controlador e três nodos sensores, para um robô móvel dotado de rodas com acionamento diferencial. O problema de rastreamento de trajetória é resolvido por meio de uma realimentação linearizante entrada-saída para a posição do robô. Os sensores e o controlador são implementados de forma totalmente independente, sendo a comunicação entre os nodos feita por meio de *sockets*. Os algoritmos foram implementados em linguagem de programação C++ e executados em computadores com sistema operacional Linux.

2 Modelo cinemático do robô

Considera-se aqui o modelo de um robô móvel dotado de rodas e com acionamento diferencial, constituído de um corpo rígido e rodas não deformáveis (Figura 1). A movimentação realiza-se em um plano horizontal. Define-se um sistema de coordenadas global, $\{O, X_o, Y_o\}$, e um sistema de coordenadas móvel, $\{C, X_c, Y_c\}$, associado ao centro de massa, C . A cinemática do robô com relação ao centro de massa é dada por:

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (1)$$

onde u_1 é a velocidade tangencial e u_2 é a velocidade angular.

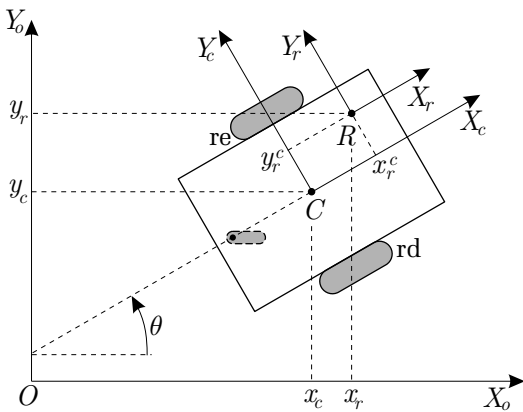


Figura 1: Modelo geométrico do robô móvel.

Ainda, elege-se um ponto, *a priori* arbitrário, da estrutura do robô como ponto de referência, dado pelo sistema de coordenadas $\{R, X_r, Y_r\}$ (Lages and Hemerly, 1998). A posição deste ponto com relação ao sistema global é a saída $\mathbf{y}_s(\mathbf{x})$,

dada por:

$$\mathbf{y}_s(\mathbf{x}) = \begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x_c + x_r^c \cos \theta - y_r^c \sin \theta \\ y_c + x_r^c \sin \theta + y_r^c \cos \theta \end{bmatrix} \quad (2)$$

3 Controle por realimentação linearizante

Para que se possa aplicar uma realimentação linearizante, é necessário que o modelo (1) seja escrito na forma afim (Isidori, 1995), ou seja,

$$\dot{\mathbf{x}} = f(\mathbf{x}) + \sum_{i=1}^p g_i(\mathbf{x}) \mathbf{u}_i, \quad (3)$$

com $p = 2$, $f(\mathbf{x}) = 0$ e

$$\begin{aligned} g_1(\mathbf{x}) &= [\cos \theta \quad \sin \theta \quad 0]^T \\ g_2(\mathbf{x}) &= [0 \quad 0 \quad 1]^T \end{aligned}$$

O método de linearização entrada-saída consiste em diferenciar a expressão de saída até que a entrada $\mathbf{u}(\mathbf{x})$ apareça na expressão (Isidori, 1995). Percebe-se que isto acontece logo na primeira diferenciação. Assim, tem-se o seguinte modelo:

$$\dot{\mathbf{y}}_s(\mathbf{x}) = \alpha(\mathbf{x}) + \beta(\mathbf{x})\mathbf{u}(\mathbf{x}), \quad (4)$$

que é linearizado aplicando-se o seguinte controle linearizante:

$$\mathbf{u}(\mathbf{x}) = \beta(\mathbf{x})^{-1}\mathbf{v}(\mathbf{x}) - \alpha(\mathbf{x}), \quad (5)$$

onde $\alpha(\mathbf{x}) = 0$, pois $f(\mathbf{x}) = 0$, e

$$\begin{aligned} \beta(\mathbf{x}) &= \begin{bmatrix} \frac{\partial x_r}{\partial \mathbf{x}} g_1(\mathbf{x}) & \frac{\partial x_r}{\partial \mathbf{x}} g_2(\mathbf{x}) \\ \frac{\partial y_r}{\partial \mathbf{x}} g_1(\mathbf{x}) & \frac{\partial y_r}{\partial \mathbf{x}} g_2(\mathbf{x}) \end{bmatrix} \\ &= \begin{bmatrix} \cos x_3 & -(x_r^c \sin x_3 + y_r^c \cos x_3) \\ \sin x_3 & x_r^c \cos x_3 - y_r^c \sin x_3 \end{bmatrix}, \end{aligned} \quad (6)$$

Fica evidente aqui a necessidade de definir um ponto R que não esteja sobre o eixo comum às rodas fixas. Pontos sobre este eixo (como o ponto C) podem mover-se instantaneamente em apenas uma direção, enquanto que os demais pontos do corpo do robô podem mover-se em duas direções (Lages, 1998). Assim, da equação (6), para a inversa de $\beta(\mathbf{x})$ existir, x_r^c precisa ser diferente de zero.

Utilizando (5) em (4), pode-se projetar uma lei linear proporcional para o controle do modelo (4) em malha fechada. Assim:

$$\mathbf{v}(\mathbf{x}) = \mathbf{L}\mathbf{e}_s,$$

onde \mathbf{L} é uma matriz de ganhos positivos e $\mathbf{e}_s = \mathbf{y}_{ref} - \mathbf{y}_s$. Assim:

$$\mathbf{L} = \begin{bmatrix} l_x & 0 \\ 0 & l_y \end{bmatrix} \quad \mathbf{e}_s = \begin{bmatrix} x_{ref} - x_r \\ y_{ref} - y_r \end{bmatrix}$$

Então, representa-se o sistema em malha fechada pelo diagrama de blocos da Figura 2.

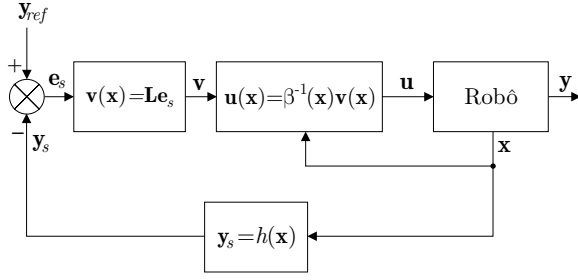


Figura 2: Diagrama de blocos do sistema em malha fechada.

4 Estimação dos estados por filtro de Kalman

O filtro de Kalman (Kalman, 1960) é um método linear recursivo para a estimação de dados e muito utilizado em estudos de sistemas dinâmicos, em especial problemas de navegação aeroespacial, robótica e controle (Leonard and Durrant-White, 1992). A estimativa feita pelo filtro de Kalman é ótima segundo um critério de mínimo erro médio quadrático, ou seja, no sentido de que o mesmo minimiza a covariância do erro da estimativa.

Considerando que o encoder fornece informação de posição e orientação, o GPS fornece informação de posição e a bússola fornece informação de orientação, define-se a expressão de observação como sendo:

$$\mathbf{y}(k) = [x_{enc}(k) \quad y_{enc}(k) \quad x_{gps}(k) \quad y_{gps}(k) \quad \theta_{enc}(k) \quad \theta_{bus}(k)]^T$$

Assim, assumindo a existência de ruídos aditivos no modelo e na observação, pode-se reescrever o modelo (3) na forma discreta como:

$$\mathbf{x}(k+1) = f(\mathbf{x}(k))\mathbf{u}(k) + \mathbf{w}(k) \quad (7)$$

$$\mathbf{y}(k) = h(\mathbf{x}(k)) + \mathbf{v}(k), \quad (8)$$

com $\mathbf{x} = [x_1 \quad x_2 \quad x_3]^T$ e

$$f(\mathbf{x}(k)) = \begin{bmatrix} x_1(k) + \cos x_3(k)T & 0 \\ x_2(k) + \sin x_3(k)T & 0 \\ 0 & x_3(k)T \end{bmatrix},$$

onde T é o período de amostragem e k é o instante amostral. $\mathbf{w}(k)$ e $\mathbf{v}(k)$ são vetores de ruído branco (com distribuição normal e média zero) de processo e de medida, respectivamente, dados pelas seguintes expressões de covariância:

$$E\{\mathbf{w}(k)\mathbf{w}^T(i)\} = \mathbf{Q}(k)\delta(k-i) \quad (9)$$

$$E\{\mathbf{v}(k)\mathbf{v}^T(i)\} = \mathbf{R}(k)\delta(k-i) \quad (10)$$

$$E\{\mathbf{w}(k)\mathbf{v}^T(i)\} = 0, \quad (11)$$

$E\{\cdot\}$ é o operador valor esperado e $\mathbf{Q}(k)$ e $\mathbf{R}(k)$ são matrizes diagonais formadas pelas variâncias de $\mathbf{w}(k)$ e $\mathbf{v}(k)$, respectivamente. A expressão (11) indica que os ruídos de processo e de medida são decorrelacionados.

4.1 Filtro de Kalman para sistemas lineares

Conforme Goodwin and Sin (1984), as expressões recursivas para a utilização do filtro de Kalman para sistemas lineares são as seguintes:

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{C}^T(k) \left(\mathbf{C}(k)\mathbf{P}(k|k-1)\mathbf{C}^T(k) + \mathbf{R}(k) \right)^{-1} \quad (12)$$

$$\mathbf{P}(k|k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{C}(k))\mathbf{P}(k|k-1) \quad (13)$$

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{y}(k) - \mathbf{C}(k)\hat{\mathbf{x}}(k|k-1)) \quad (14)$$

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{A}(k)\hat{\mathbf{x}}(k|k) + \mathbf{B}(k)\mathbf{u}(k) \quad (15)$$

$$\mathbf{P}(k+1|k) = \mathbf{A}(k)\mathbf{P}(k|k)\mathbf{A}^T(k) + \mathbf{Q}(k), \quad (16)$$

onde $\mathbf{K}(k)$ é o ganho do filtro de Kalman, $\mathbf{P}(k|k)$ é a covariância do erro de estimação e $\mathbf{A}(k)$, $\mathbf{B}(k)$ e $\mathbf{C}(k)$ são matrizes que descrevem o modelo linear $\mathbf{x}(k+1) = \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k)$, $\mathbf{y}(k) = \mathbf{C}(k)\mathbf{x}(k)$.

4.2 Extensão do filtro de Kalman para sistemas não lineares

Um dos requisitos para a utilização do filtro de Kalman é que o sistema seja linear. Como este não é o caso, algum tipo de linearização precisa ser usada (Mutambara, 1998). Assim, seja a estimativa do estado dada por:

$$\hat{\mathbf{x}}(k|k) = E\{\mathbf{x}(k)\}$$

Expandindo o modelo (7)–(8) em série de Taylor em torno desta estimativa e desprezando-se os termos de ordem superior, tem-se:

$$\mathbf{x}(k+1) \approx f(\hat{\mathbf{x}}(k|k))\mathbf{u}(k) + \mathbf{F}(k)(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k)) + \mathbf{w}(k)$$

$$\mathbf{y}(k) \approx h(\hat{\mathbf{x}}(k|k)) + \mathbf{H}(k)(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k)) + \mathbf{v}(k),$$

onde $\mathbf{F}(k)$ e $\mathbf{H}(k)$ são os jacobianos de $f(\mathbf{x}(k))$ e $h(\mathbf{x}(k))$, respectivamente, avaliados em $\hat{\mathbf{x}}(k|k)$,

$$\mathbf{F}(k) = \left. \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(k|k)} \quad \mathbf{H}(k) = \left. \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(k|k)}$$

Assim, após algumas manipulações matemáticas, chega-se às expressões do filtro de Kalman estendido:

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{H}^T(k) \left(\mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}^T(k) + \mathbf{R}(k) \right)^{-1} \quad (17)$$

$$\mathbf{P}(k|k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k))\mathbf{P}(k|k-1) \quad (18)$$

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{y}(k) - \mathbf{H}(k)\hat{\mathbf{x}}(k|k-1)) \quad (19)$$

$$\hat{\mathbf{x}}(k+1|k) = f(\hat{\mathbf{x}}(k|k))\mathbf{u}(k) \quad (20)$$

$$\mathbf{P}(k+1|k) = \mathbf{F}(k)\mathbf{P}(k|k)\mathbf{F}^T(k) + \mathbf{Q}(k) \quad (21)$$

Tem-se portanto um método para obter o estado estimado $\hat{\mathbf{x}}(k|k)$ a partir de $\hat{\mathbf{x}}(k|k-1)$,

$\mathbf{P}(k|k-1)$ e da medida $\mathbf{y}(k)$, para um sistema não linear como o considerado nas equações (7)–(8).

Pode ainda ser utilizado aqui o *Princípio da Equivalência à Certeza Presumida* (Goodwin and Sin, 1984). Então, utiliza-se o controle linearizante projetado na Seção 3 para o modelo (7)–(8), considerando agora que a leitura dos estados são fornecidas através do filtro de Kalman estendido.

Na Figura 3, tem-se o diagrama de blocos do sistema completo.

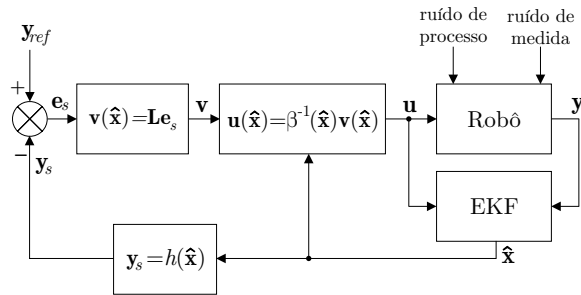


Figura 3: Diagrama de blocos do sistema em malha fechada.

4.3 Filtro de Kalman estendido descentralizado

O emprego de uma estrutura descentralizada em uma rede de nodos totalmente conectados, como a utilizada neste trabalho, faz com que a estimativa global do estado, computada em cada um dos nodos, seja idêntica à estimativa que seria computada em um sistema centralizado. Ainda, como dito anteriormente, esta estrutura de estimação deixa o sistema tolerante a falhas individuais dos sensores. Assim, se um dos nodos ou a comunicação falhar, os outros nodos e o controlador continuarão a realizar a fusão dos dados e a calcular a estimativa global dos estados (Mutambara, 1998).

Por conveniência, escreve-se o vetor de ruídos $\mathbf{v}(k)$ como:

$$\mathbf{v}(k) = [\mathbf{v}_{enc}^T(k) \quad \mathbf{v}_{gps}^T(k) \quad \mathbf{v}_{bus}^T(k)]^T$$

Para o caso onde é feita a estimação descentralizada totalmente conectada, apenas a expressão de observação precisa ser particionada para cada nodo. Neste caso, se todas as inicializações forem iguais, todos os nodos irão gerar estimativas globais idênticas dos estados. Entretanto, é necessário executar o algoritmo do filtro de Kalman estendido em cada um dos nodos, a fim de se calcular todos os parâmetros necessários para, a partir da estimativa local, realizar a fusão e calcular a estimativa global. Assim, como cada nodo sensor possui um domínio de informação diferente,

tem-se que:

$$\mathbf{y}_{enc}(k) = \begin{bmatrix} x_{enc}(k) \\ y_{enc}(k) \\ \theta_{enc}(k) \end{bmatrix} + \mathbf{v}_{enc}(k) \quad (22)$$

$$\mathbf{y}_{gps}(k) = \begin{bmatrix} x_{gps}(k) \\ y_{gps}(k) \end{bmatrix} + \mathbf{v}_{gps}(k) \quad (23)$$

$$\mathbf{y}_{bus}(k) = [\theta_{bus}(k)] + \mathbf{v}_{bus}(k) \quad (24)$$

Considerando que o sistema de acionamento do robô também possui um processador dedicado, de forma que pode ser encarado como um nodo do sistema que possui apenas atuadores e nenhum sensor. Como este nodo aciona as interfaces de potência, por questões de segurança optou-se por utilizar componentes bem mais confiáveis do que os utilizados nos nodos sensores. Assim, a implementação do controle neste nodo, de forma centralizada, não apresenta problemas de confiabilidade. Por outro lado, como os nodos sensores possuem uma menor confiabilidade, o cálculo da lei de controle não pode ser baseado na estimativa obtida por qualquer destes nodos (embora, a princípio, todas as estimativas devam ser iguais). Portanto, o nodo controlador, além de computar a lei de controle, deverá também, antes disto, realizar a fusão das estimativas locais transmitidas pelos nodos sensores. Esta fusão é feita também através do mesmo algoritmo do filtro de Kalman implementado nos sensores.

Na Figura 4, é mostrada a organização dos nodos e suas funções, bem como o fluxo de dados entre eles. Nos nodos sensores, as setas contínua, traço-ponto e tracejada significam, respectivamente, a transmissão de dados do encoder, do GPS e da bússola. Nota-se que no nodo controlador não existem sensores, e portanto, nem a operação de estimação, apenas a de fusão dos dados.

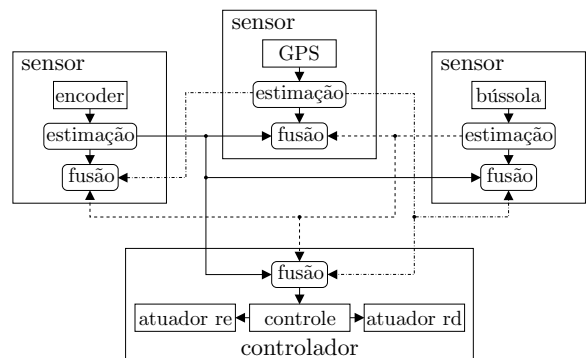


Figura 4: Organização dos nodos e o fluxo de dados.

Levando em conta todas as considerações citadas acima e as expressões de observação (22), (23) e (24), o filtro de Kalman estendido descentralizado, em cada nodo, foi implementado da seguinte maneira:

1. Obedecendo ao critério de minimização do erro médio quadrático, calcula-se o ganho do filtro de Kalman (17);

2. Calcula-se a covariância local:

$$\tilde{\mathbf{P}}(k|k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{H})\mathbf{P}(k|k-1)$$

3. Calcula-se o estado estimado local:

$$\tilde{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{y}(k) - h(\hat{\mathbf{x}}(k|k-1)))$$

4. Calcula-se a seguinte diferença entre as estimativas local e global:

$$\mathbf{e}(k) = \tilde{\mathbf{P}}^{-1}(k|k)\tilde{\mathbf{x}}(k|k) - \mathbf{P}^{-1}(k|k-1)\hat{\mathbf{x}}(k|k-1)$$

5. Calcula-se a seguinte diferença entre as covariâncias local e global:

$$\mathbf{E}(k) = \tilde{\mathbf{P}}^{-1}(k|k) - \mathbf{P}^{-1}(k|k-1)$$

Neste ponto, as grandezas $\mathbf{e}(k)$ e $\mathbf{E}(k)$ de cada nodo são transmitidos para os outros nodos através de *sockets*. Faz-se agora a fusão destes dados, modificando-se as equações (18) e (19) para o caso descentralizado, conforme os passos a seguir:

6. Calcula-se a covariância global:

$$\mathbf{P}(k|k) = (\mathbf{P}^{-1}(k|k-1) + \mathbf{E}_{enc}(k) + \mathbf{E}_{gps}(k) + \mathbf{E}_{bus}(k))^{-1};$$

7. Calcula-se a estimativa global:

$$\hat{\mathbf{x}}(k|k) = \mathbf{P}(k|k)(\mathbf{P}^{-1}\hat{\mathbf{x}}(k|k-1) + \mathbf{e}_{enc}(k) + \mathbf{e}_{gps}(k) + \mathbf{e}_{bus}(k)),$$

8. Calcula-se a predição da estimativa global, $\hat{\mathbf{x}}(k+1|k)$ - equação (20).

9. Calcula-se a predição da covariância global, $\mathbf{P}(k+1|k)$ - equação (21);

E assim repetem-se os passos 1 a 9, até o fim da trajetória.

Os passos 6 a 9 são referentes à operação de fusão de dados. Como o controlador não realiza operações de observação ou de estimação, apenas estes passos são necessários para este nodo, considerando que o mesmo também recebe as diferenças locais $\mathbf{e}(k)$ e $\mathbf{E}(k)$ dos nodos sensores.

5 Resultados de simulação

São mostrados agora os resultados de simulação para duas trajetórias: uma em forma de oito e outra em forma de quadrado. Em todas as figuras, a linha tracejada representa a trajetória de referência.

Nas Figuras 5 e 6 pode-se ver a ação do controle sobre o comportamento do robô. Nota-se que a posição controlada converge para a trajetória de referência.

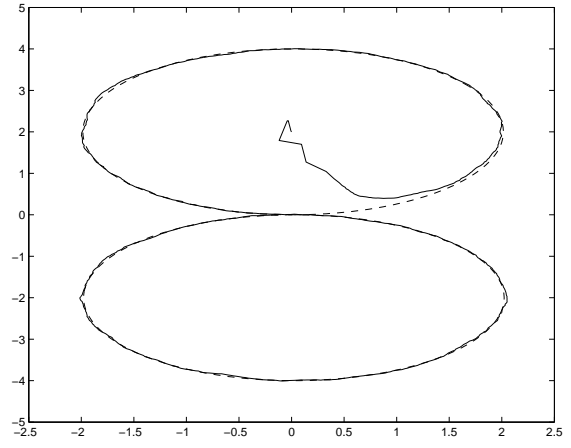


Figura 5: Posição controlada para uma trajetória em forma de oito.

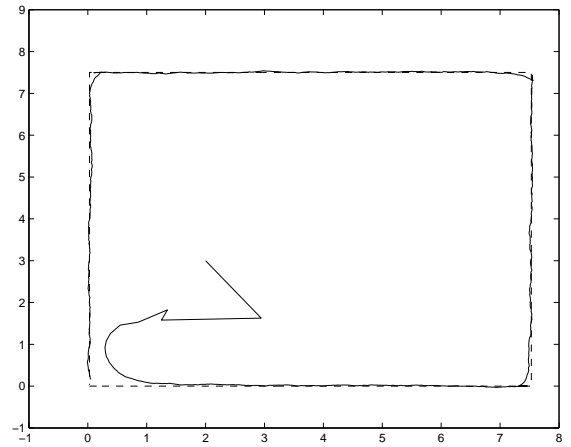


Figura 6: Posição controlada para uma trajetória em forma de quadrado.

Nas Figuras 7 e 8, verifica-se o efeito do aumento do ruído de processo no controle. O desvio padrão do ruído de processo foi aumentado em 20 vezes ($\mathbf{Q} = \text{diag}[0.0004, 0.0004, 0.0006]$). Nota-se que o robô ainda segue a trajetória, apesar de fazê-lo de forma mais errática.

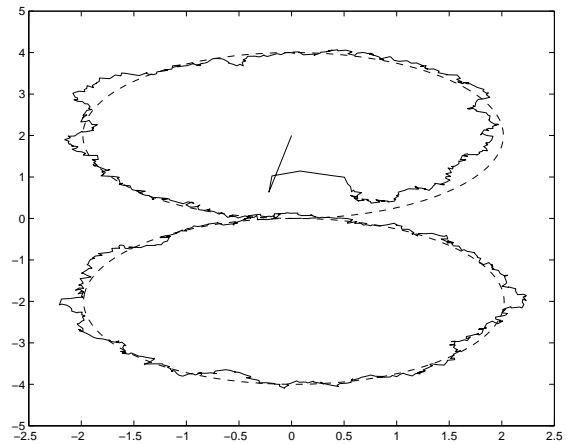


Figura 7: Efeito do aumento do ruído de processo.

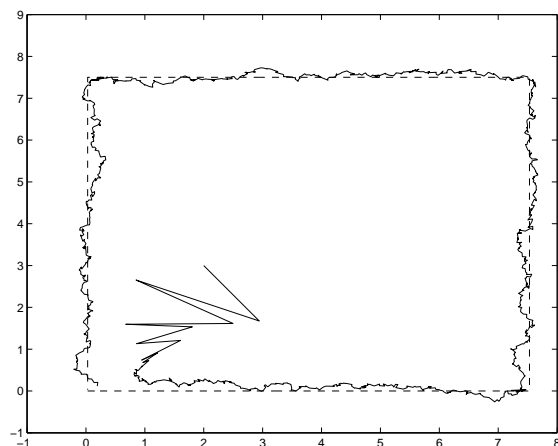


Figura 8: Efeito do aumento do ruído de processo.

6 Conclusões

Neste artigo, foi demonstrado o uso de uma estrutura composta de nodos descentralizados e totalmente conectados, para a estimação ótima da configuração e controle da posição de um robô móvel. Três sensores com domínios de informação sobrepostos foram utilizados.

Verificou-se as vantagens de se utilizar uma arquitetura descentralizada totalmente conectada. Observou-se o desempenho do filtro de Kalman estendido descentralizado como estimador ótimo de dados, e do controle linearizante, que mostrou desempenhos bem satisfatórios para as diversas situações simuladas.

Como trabalhos futuros, pretende-se utilizar esta estrutura para o processamento da mesma em tempo real, preparando a implementação real de uma arquitetura descentralizada para estimação e controle em tempo real de um robô móvel.

Agradecimentos

Agradecimento à CAPES pelo apoio financeiro.

Referências

- Borenstein, J., Everett, H. R. and Feng, L. (1996). Where am i? – sensors and methods for mobile robot positioning, *Technical report*, Univ. Of Michigan.
- Goodwin, G. and Sin, K. (1984). *Adaptive filtering prediction and control*, Prentice-Hall, Englewood Cliffs, NJ.
- Isidori, A. (1995). *Nonlinear control systems*, 3rd. edn, Springer-Verlag, London.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems, *Trans. ASME J. Basic Engineerings* pp. 35–45.
- Kang, D., Hashimoto, H. and Harashima, F. (1995). Position estimation for mobile robot using sensor fusion, *IEEE Intl. Conf. on Intelligent Robots and Systems*, pp. 271–276.
- Lages, W. F. (1998). *Controle e estimação de posição e orientação de robôs móveis*, Tese de doutorado, Inst. Tecnológico da Aeronáutica, São José dos Campos.
- Lages, W. F. and Hemerly, E. M. (1998). Controle em coordenadas polares de robôs móveis com rodas, *XII Congresso Brasileiro de Automática*, pp. 605–610.
- Leonard, J. J. and Durrant-White, H. F. (1992). *Directed sonar sensing for mobile robot navigation*, Kluwer Academic Publishers.
- Medromi, H., Zaafrani, E., Dekneuevel, E. and Thomas, M. C. (1996). Multi-sensors localization for control af autonomous mobile robot, *IEEE/SICE/RSJ Intl. Conf. on Multi-sensor Fusion and Integration for Intelligent Systems*, pp. 373–380.
- Mutambara, A. G. O. (1998). *Decentralized estimation and control for multisensor systems*, CRC Press, Boca Raton, FL.
- Park, K. C., Chung, D., Chung, H. and Lee, J. G. (1996). Dead reckoning navigation of a mobile robot using an indirect kalman filter, *IEEE/SICE/RSJ Intl. Conf. on Multi-sensor Fusion and Integration for Intelligent Systems*, pp. 132–138.
- Sandi, F. A., Hemerly, E. M. and Lages, W. F. (1998). Sistema para navegação e guiagem de robôs móveis autônomos, *Controle e Automação* **9(3)**: 107–118.
- Vestli, S. J. (1995). *Fast, accurate and robust estimation of mobile robot position and orientation*, PhD thesis, Swiss Federal Inst. of Technology, Zurich.
- Yajnik, S. and Jha, N. K. (1997). Graceful degradation in an algorithm-based fault tolerant multiprocessor systems, *IEEE Trans. on Parallel and Distributed Systems* **8(2)**: 137–153.