

NAVEGAÇÃO DE ROBÔS MÓVEIS COM BASE EM UM SENSOR DE VARREDURA LASER

Flávio Garcia Pereira * Mário Sarcinelli Filho *

* Programa de Pós Graduação em Engenharia Elétrica
Universidade Federal do Espírito Santo
Av. Fernando Ferrari, 514 – 29075-910 – Vitória, ES,
Brasil
E-mail{flaviog, mario.sarcinelli}@ele.ufes.br

Resumo: Este trabalho trata da navegação de robôs móveis em ambientes semi-estruturados. Para isso, foram desenvolvidos controladores baseados na abordagem reativa. Os controladores implementados permitem que o robô navegue de forma segura entre dois pontos e por corredores, evitando os obstáculos que surgirem no seu caminho. São apresentados dois métodos de desvio de obstáculos, um tangencial e outro não tangencial, os quais são adequados para situações em que o robô busca um ponto destino e encontra obstáculos em seu caminho. As informações sobre o ambiente de trabalho do robô são obtidas por um sensor de varredura *laser* instalado a bordo do robô móvel utilizado, o PIONEER 2-DX da ActivMedia. Os controladores desenvolvidos são validados através de experimentos reais, sempre utilizando o robô móvel equipado com o sensor laser de varredura.

Palavras-Chave: *Laser*, Robótica, Corredores, Desvio de Obstáculos.

1. INTRODUÇÃO

Sempre que um robô móvel navega, é necessário garantir que ele não colidirá com obstáculos. Vários métodos são utilizados neste contexto, os quais podem ser classificados, num nível bem genérico, como deliberativos ou reativos. Os métodos deliberativos são aqueles em que a navegação é baseada em um mapa do ambiente, utilizado para planejar a trajetória que o robô deve seguir. Já os métodos reativos não utilizam nenhum conhecimento prévio do ambiente ao redor do robô, exceto por informação bem geral como, por exemplo, que tal ambiente é fechado e não há deslizamento. Portanto não é possível planejar trajetórias. Este tipo de navegação tem como foco a relação de causalidade entre a *percepção* do ambiente pelo robô e a *ação* que ele executará.

Este trabalho trata da navegação segura de robôs móveis em ambientes semi-estruturados, utilizando métodos reativos. Serão implementados controladores para navegação em corredores e de desvio obstáculos, neste último caso utilizando as abordagens tangencial (Ferreira (2004)) e não tangencial (Pereira (2006)). As informações necessárias para a implementação destes algoritmos são fornecidas por um sensor de varredura *laser* instalado a bordo do robô

Para descrever os métodos utilizados e os experimentos realizados, o trabalho está dividido da seguinte forma: na Seção 2 é apresentado um controlador para navegação de robôs móveis em corredores enquanto que na Seção 3 são apresentados controladores para desvio de obstáculos de forma tangencial e não tangencial. Os resultados obtidos com a implementação destes controladores a bordo de um robô PIONEER 2-DX são discuti-

dos na Seção 4. Finalmente, a Seção 5 apresenta uma avaliação conclusiva dos resultados obtidos.

2. NAVEGAÇÃO EM CORREDORES

Quando robôs móveis são postos para navegar em ambientes fechados, eles devem ter a capacidade de seguir corredores, paredes, contornar esquinas, entrar e sair de salas, etc (Freire (2002)).

Considerando que o robô é do tipo monociclo com acionamento diferencial, o seu movimento é descrito pela ação conjunta das velocidades linear v e angular ω . A Figura 1 mostra o robô com o seu sistema de coordenadas.

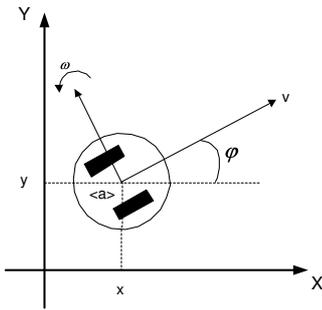


Fig. 1. O robô e o sistema de coordenadas inercial.

Seja $[x, y, \varphi]^T$ o vetor que representa a postura do robô, onde x e y são as coordenadas do seu centro de rotação e φ é o seu ângulo de orientação, em relação a um sistema de coordenadas externo ao robô. O conjunto de equações cinemáticas que descrevem o movimento do veículo é dado por

$$\begin{cases} \dot{x} = v \cos(\varphi) \\ \dot{y} = v \sin(\varphi) \\ \dot{\varphi} = \omega \end{cases}, \quad (1)$$

representado em coordenadas cartesianas, onde v , φ e ω são como indicados na Figura 1.

Considerando, agora, o caso que o robô móvel navega em um corredor, as variáveis de estado são determinadas em relação ao corredor. São elas \tilde{x} e φ , respectivamente a distância do centro do robô à linha média do corredor e a orientação do robô em relação ao eixo do corredor, conforme ilustrado na Figura 2.

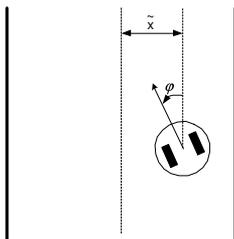


Fig. 2. O robô no corredor.

Para navegação em corredores, pode-se considerar apenas duas variáveis de estado (Carelli and Freire (2003)), e, com isso, a cinemática do robô fica reduzida às equações

$$\begin{cases} \dot{\tilde{x}} = v \sin(\varphi) \\ \dot{\varphi} = \omega \end{cases}. \quad (2)$$

Considerando que o movimento do robô no corredor é descrito pelas equações 2, o objetivo do controlador é fazer com que as variáveis de estado \tilde{x} e φ tendam assintoticamente para zero. Para isso, é considerada uma velocidade linear constante e a ação de controle ω , que representa a velocidade angular do robô, é dada por

$$\omega = -k_1(\varphi) \varphi - k_2(\tilde{x}) \tilde{x} v \frac{\sin(\varphi)}{\varphi}, \quad (3)$$

onde $k_1(\varphi)$ e $k_2(\tilde{x})$ são funções positivas escolhidas para não saturar o sinal de controle ω enviado ao robô móvel utilizado (Freire (2002)), já que há um limite físico para ela.

Visto que para este controlador a velocidade linear foi considerada constante, a única variável de controle é a velocidade angular. Esta, como pode ser observado na Equação 3, é função do erro de posição \tilde{x} e da orientação φ , que são as variáveis de estado do sistema. Para que o controlador funcione de maneira correta, é necessário que ele receba os valores destas variáveis a cada ciclo de controle.

Estas variáveis serão calculadas com base nas informações fornecidas por um sensor *laser* instalado a bordo do robô. Este sensor (Figura 3) fornece, de uma única vez, 181 medidas de distância (de 0° a 180° , em intervalos de 1°), conforme ilustra a Figura 4.



Fig. 3. O sensor *laser*.



Fig. 4. Varredura realizada pelo sensor *laser*.

Para se determinar as variáveis de estado para navegação em corredores, \tilde{x} e φ , foram utilizadas as medidas de 0° e 20° , do lado direito, e de 160° e 180° , do lado esquerdo. A Figura 5 apresenta uma situação em que o robô se encontra num corredor e ilustra as medidas usadas para se determinar as variáveis de estado.

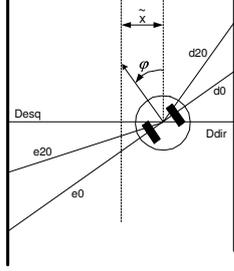


Fig. 5. Medidas tomadas pelo sensor *laser* para se efetuar o cálculo de \tilde{x} e φ .

As variáveis de estado são calculadas com base nas representações geométricas apresentadas na Figura 6, oriundas da situação ilustrada na Figura 5. Os valores de d_0 e d_{20} são conhecidos, pois representam as medidas fornecidas pelo sensor *laser* nas direções 0° e 20° . A Figura 6 (a) mostra a representação geométrica que foi utilizada para se calcular as variáveis de estado para o lado direito do robô, enquanto que a Figura 6 (b) apresenta esta mesma situação, porém rotacionada para facilitar a visualização. Baseado na Figura 6 (b), a primeira variável a ser calculada é o ângulo de orientação do robô.

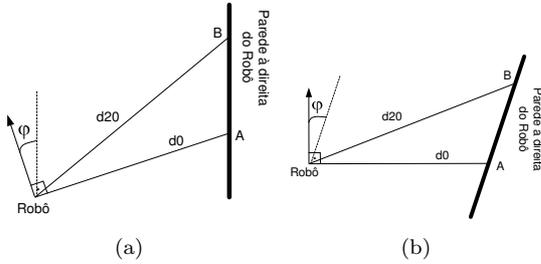


Fig. 6. Geometria para definir \tilde{x} e φ .

Considerando um sistema de coordenadas centrado no robô, pode-se calcular as coordenadas dos pontos A e B , mostrados na Figura 6. Estes pontos estão localizados sobre a parede que se encontra à direita do robô. Desta forma, a orientação do robô é igual ao complemento do ângulo que representa a inclinação da reta que une os pontos A e B (ver Figura 7) (Pereira (2006)).

Como o sistema de coordenadas está no robô e as distâncias d_0 e d_{20} são conhecidas, assim como o ângulo entre elas, que é de 20° , as coordenadas dos pontos A e B são

$$\begin{cases} x_A = d_0 \\ y_A = 0 \end{cases} \quad \text{e} \quad \begin{cases} x_B = d_{20} \cos(20^\circ) \\ y_B = d_{20} \sin(20^\circ) \end{cases}. \quad (4)$$

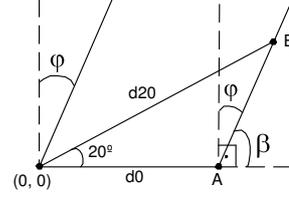


Fig. 7. Determinação do ângulo φ .

Desta maneira, o ângulo φ como se pode observar na Figura 7, é dado por $\varphi = 90 - \beta$, onde β é a inclinação da reta que passa por A e B . Assim,

$$\varphi = 90 - \tan^{-1} \left(\frac{d_{20} \sin(20^\circ)}{d_{20} \cos(20^\circ) - d_0} \right). \quad (5)$$

Pode-se observar que não há restrições no sentido de que φ seja aproximadamente igual a zero. Este ângulo é calculado, analogamente, para o lado esquerdo do robô. Sendo assim, o ângulo φ utilizado como variável de estado é obtido pela média aritmética dos ângulos calculados nos lados esquerdo e direito, ou seja

$$\varphi = \frac{\varphi_{dir} + \varphi_{esq}}{2}. \quad (6)$$

Deve-se, agora, determinar o erro de posição do robô \tilde{x} . Para isso, é necessário conhecer as distâncias do robô às paredes direita e esquerda, D_{dir} e D_{esq} , respectivamente. A Figura 8 ilustra uma situação em que o robô está num corredor, as distâncias D_{dir} e D_{esq} e o ângulo φ já calculado.

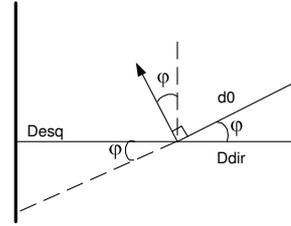


Fig. 8. Cálculo de \tilde{x} .

Para a situação apresentada na Figura 8, a distância até a parede esquerda é a medida fornecida pelo sensor cuja orientação é igual ao suplemento do ângulo que representa a orientação do robô, ou seja,

$$D_{esq} = medida(180 - int(\varphi)), \quad (7)$$

onde $int(\varphi)$ retorna a parte inteira do ângulo φ e $medida(.)$ fornece o valor da distância capturada pelo sensor cuja orientação é passada como parâmetro. Já a distância até a parede direita é calculada como

$$D_{dir} = d_0 \cos(\varphi). \quad (8)$$

Agora que as distâncias entre o robô e as duas paredes são conhecidas, é possível determinar a

segunda variável de estado, o erro de posição do robô, que é dada por

$$\tilde{x} = \frac{D_{dir} - D_{esq}}{2}. \quad (9)$$

Determinadas as variáveis de estado do controlador, basta substituir seus valores na Equação (3) para obter o valor da ação de controle que será enviada ao robô para que o mesmo navegue a fim de alcançar o seu objetivo, ou seja, encontrar o centro do corredor e ali permanecer.

A análise de estabilidade deste controlador é está apresentada em Freire (2002) e Carelli and Freire (2003). Ali se prova que as variáveis de estado tendem assintoticamente para zero quando $t \rightarrow \infty$, ou seja, o controlador de fato faz com que o robô navegue ao longo da linha média do corredor.

3. NAVEGAÇÃO COM DESVIO DE OBSTÁCULOS

O movimento do robô é descrito, em coordenadas cartesianas, pelo conjunto de equações 1. Se forem considerados ρ e θ , respectivamente, o erro de posição do veículo e a sua orientação em relação a um referencial inercial $\langle g \rangle$ (ver Figura 9), pode-se escrever as equações cinemáticas do mesmo em coordenadas polares, que são

$$\begin{cases} \dot{\rho} = -v \cos(\theta - \varphi) \\ \dot{\theta} = v \frac{\sin(\theta - \varphi)}{\rho} \\ \dot{\varphi} = \omega \end{cases}. \quad (10)$$

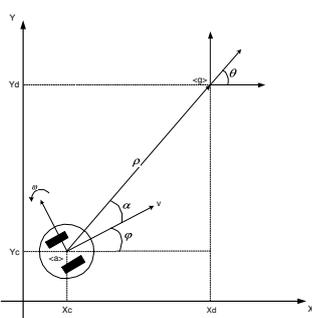


Fig. 9. O robô e o sistema de coordenadas inercial.

Como em Secchi (1998), foi implementado um controlador de posição final baseado nas equações cinemáticas em coordenadas polares. Este controlador tem como objetivo fazer com que o robô navegue entre dois pontos pré-determinados, um inicial e um final. As ações de controle deste controlador são dadas pelas seguintes equações

$$v = v_{\max} \tanh(\rho) \cos(\alpha) \quad (11)$$

e

$$\omega = k_{\omega} \alpha + v_{\max} \frac{\tanh(\rho)}{\rho} \sin(\alpha) \cos(\alpha) \quad (12)$$

onde ρ é o erro de posição do robô em relação ao ponto final, α é o erro de orientação do robô com relação ao ponto destino, v_{\max} é o valor máximo da velocidade linear aplicável ao robô e k_{ω} é escolhida em função do máximo valor desejado para a velocidade angular ω , de forma que $|\omega_{\max}| = k_{\omega} \pi + 0.5 v_{\max}$. A prova da estabilidade deste controlador está em Secchi (1998).

3.1 Desvio Tangencial

Este controlador é o controlador de posição final apresentado em Secchi (1998) com a adição de um *loop* externo de controle. Este *loop* é responsável pela geração de um alvo virtual para o robô quando um obstáculo é detectado. Quando isso ocorre, o controlador gera um ângulo de desvio ϕ , e o alvo real é rotacionado de ϕ graus (ver Figura 10). Esta rotação faz com que o alvo virtual se posicione de forma que o robô evite o obstáculo de forma tangencial (Ferreira (2004)).

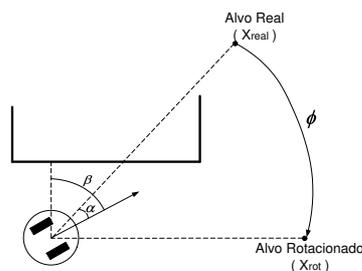


Fig. 10. Determinação do ângulo ϕ .

O ângulo ϕ depende dos ângulos α e β que são, respectivamente, o erro de orientação do robô com relação ao ponto destino e o ângulo entre a orientação do robô com a parte mais próxima do obstáculo detectado (Figura 10). Dessa maneira, o ângulo ϕ é calculado como

$$\phi = \text{sign}(\beta) \cdot (|\beta| - 90) - \alpha. \quad (13)$$

O ângulo β é determinado a partir das leituras do sensor *laser*. Entretanto não foram utilizadas todas as medidas disponíveis. As medidas foram tomadas de 0° a 180° em intervalos de 10° (esta estrutura foi definida a partir da análise de diversas simulações e experimentos que foram realizados).

Após rotacionar o alvo real, este é, ainda, aproximado para 1 m à frente do robô, para que a manobra de desvio de obstáculo se realize com uma velocidade constante quando o robô está em paralelo com a tangente ao obstáculo. A Figura 11 apresenta o diagrama de blocos do controlador para desvio tangencial de obstáculos. O *loop* interno representa o controlador de posição final, enquanto que o externo é responsável pela geração do alvo virtual, para o desvio tangencial.

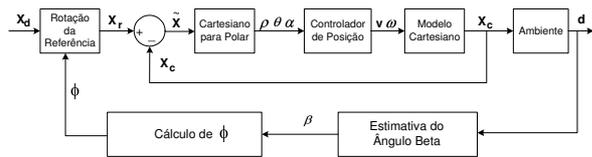


Fig. 11. Diagrama de blocos do controlador de desvio de obstáculo Ferreira (2004).

3.2 Desvio Não Tangencial

Diferentemente do controlador de desvio tangencial de obstáculos, este controlador não utiliza um *loop* externo para gerar uma posição virtual para o robô. Este controlador tem como objetivo fazer com que o robô evite o obstáculo buscando o caminho livre de obstáculos e que possui o menor erro de orientação com relação à posição final desejada.

São tomadas as 181 medidas fornecidas pelo sensor *laser* e é gerado um vetor de 181 posições, formado por 0's e 1's, onde cada 0 representa que a medida fornecida pelo sensor é menor que 70 *cm*, ou seja, há um obstáculo naquela direção. Após determinar as regiões livres de obstáculos, aquela que possui uma abertura suficientemente grande para o robô passar e que tiver o menor erro angular com relação ao ponto destino é escolhida como caminho a ser seguido.

A Figura 12 ilustra uma situação em que o robô encontra um obstáculo à sua frente. Observa-se que o caminho livre à esquerda do robô é maior que aquele à direita. Porém, o caminho à direita possui o menor erro angular em relação ao ponto destino. Logo, o caminho que deve ser escolhido para realizar o desvio deste obstáculo é o que se encontra à direita do robô.

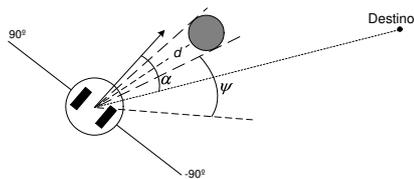


Fig. 12. O ângulo ψ .

Desta forma, o ângulo ψ (ver Figura 12) é escolhido como sendo o menor ângulo que a região livre de obstáculos faz com o vetor erro de posição. Assim, este ângulo e a menor distância a um obstáculo serão utilizados para determinar as variáveis de controle v e ω . O ângulo de desvio é determinado como sendo o ângulo médio do caminho livre.

Assim, se um obstáculo for detectado, as velocidades linear e angular serão dadas por

$$v = k_v \tanh(d) \cos(\psi) \quad (14)$$

e

$$\omega = k_\omega \sin(\psi) \quad (15)$$

onde k_v e k_ω são constantes positivas que determinam, respectivamente, as velocidades linear e angular máximas. É importante mencionar que as equações 14 e 15 foram escolhidas de maneira empírica e não foi realizado nenhum estudo da estabilidade das mesmas.

4. RESULTADOS EXPERIMENTAIS

São apresentados, agora, os resultados obtidos quando os algoritmos de controle descritos nas Seções 2 e 3 foram implementados a bordo do robô rôvel PIONEER 2-DX.

Inicialmente, são apresentados os resultados obtidos quando o robô foi posto para navegar em um corredor de 2 *m* de largura, que, num determinado instante, começa a se estreitar. A Figura 13 mostra a trajetória descrita pelo robô. Já as Figuras 14 e 15 mostram, respectivamente, as velocidades linear e angular do robô, enquanto que o erro de posição e a orientação do robô são apresentados nas Figuras 16 (a) e (b) respectivamente.

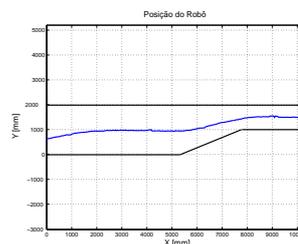


Fig. 13. Trajetória descrita pelo robô.

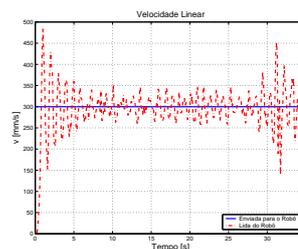


Fig. 14. Velocidade linear.

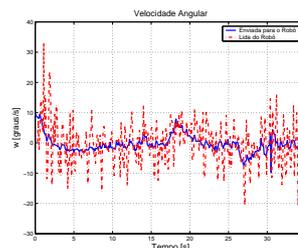


Fig. 15. Velocidade angular.

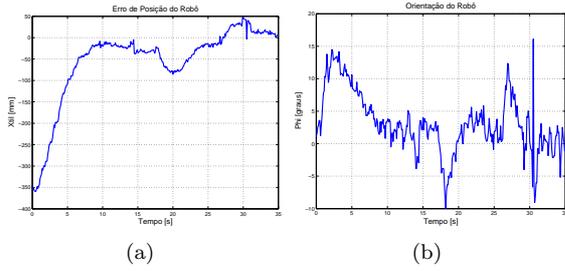


Fig. 16. Erro de posição (a) e a orientação (b) do robô.

As velocidades plotadas em linha contínua, nas Figuras 14 e 15, representam os valores que foram determinados pelo controlador e que foram enviadas ao robô. Já as que estão pontilhadas são os valores das velocidades desenvolvidas pelo robô. Esta diferença acontece devido ao fato do controlador não considerar a dinâmica do robô, e também devido aos erros na leitura da odometria.

Serão apresentados, agora, os resultados obtidos quando o robô móvel foi posto para navegar, partindo da posição (0, 0) em busca do ponto localizado em (1000 mm, 5000 mm) no ambiente ilustrado na Figura 17. Existem, ali, obstáculos circulares e paredes no seu caminho. Este experimento foi realizado utilizando o controlador de desvio tangencial de obstáculos. A Figura 17 mostra o caminho percorrido pelo robô. As velocidades linear e angular podem ser vistas nas Figuras 18 e 19, respectivamente, enquanto que as Figuras 20 (a) e 20 (b) representam, respectivamente, o erro de posição do robô e a sua orientação.

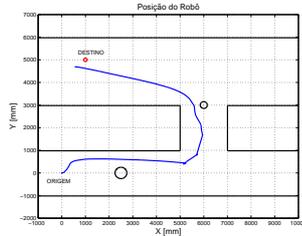


Fig. 17. Trajetória descrita pelo robô.

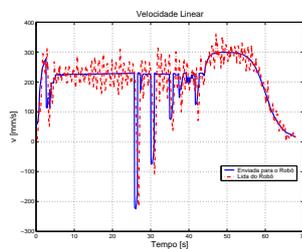


Fig. 18. Velocidade linear.

Os próximos resultados que serão apresentados correspondem a um experimento no qual o robô é posto para navegar em busca do ponto (3000 mm, 4000 mm), e existem dois obstáculos circulares em seu caminho. Neste caso, porém, o algoritmo de desvio de obstáculos é o não tangencial. A

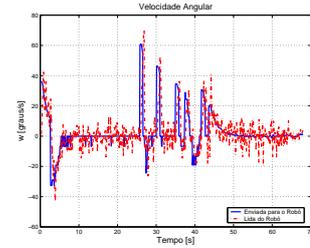


Fig. 19. Velocidade angular.

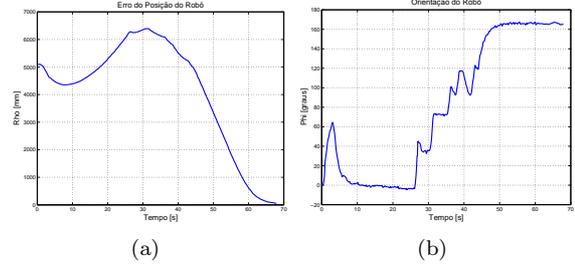


Fig. 20. Erro de posição (a) e a orientação (b) do robô.

Figura 21 mostra a trajetória descrita pelo robô. As velocidades linear e angular podem ser vistas, respectivamente, nas Figuras 22 e 23. O erro de posição e a orientação do robô estão plotados nas Figuras 24 (a) e 24 (b).

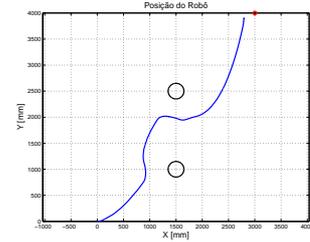


Fig. 21. Trajetria descrita pelo robô.

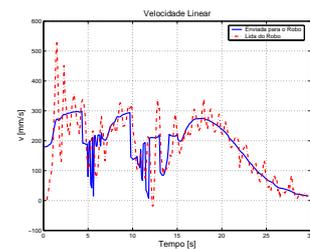


Fig. 22. Velocidade linear.

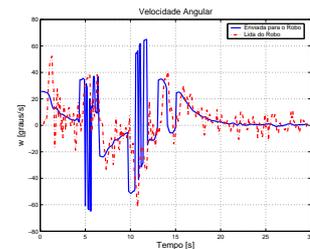


Fig. 23. Velocidade angular.

Observa-se que, nos experimentos apresentados, o objetivo foi alcançado. No caso da navegação em

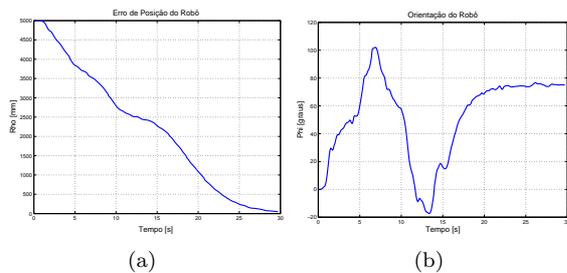


Fig. 24. Erro de posição (a) e a orientação (b) do robô.

corredores, o robô encontrou o centro do corredor e continuou navegando por ele, até se deparar com uma situação inesperada de paredes não paralelas. Mesmo nesta situação, o robô conseguiu navegar por entre estas paredes. Para a navegação com desvio tangencial e não tangencial de obstáculos, o robô atinge o ponto destino com um certo erro de posição. Isto é devido aos erros de odometria, que vão se acumulando à medida que o robô navega.

Analisando os resultados correspondentes aos experimentos envolvendo desvio de obstáculos, em particular, pode-se perceber claramente que o método do desvio tangencial permite que um único controlador guie o robô durante sua navegação em um ambiente complexo, como ilustra o experimento correspondente à Figura 17. Esta é uma vantagem significativa deste método, além do menor tempo gasto para execução da tarefa e da redução do desgaste dos motores devido à maior suavidade na realização das manobras, conforme destacado por Ferreira (2004).

Por outro lado, o desvio não tangencial de obstáculos é um método interessante em situações em que vários obstáculos surgem no caminho do robô e este tem que navegar por entre eles. Neste caso, a definição de um setor angular correspondente ao espaço livre permite ao robô navegar por entre tais obstáculos, como no exemplo da Figura 21. Entretanto, somente com um sensor de varredura preciso é possível utilizar tal método, como é o caso do sensor laser utilizado, para o qual a resolução angular é de um grau.

5. CONCLUSÕES

Neste trabalho foi abordado o problema da navegação de robôs móveis em ambientes semi-estruturados. Foram implementados algoritmos para navegação em corredores e algoritmos de controle de posição final com desvio de obstáculos. Tais controladores foram todos baseados em arquiteturas reativas, e o sensoriamento do ambiente foi feito através de um sensor de varredura *laser* instalado a bordo do robô móvel PIONER 2-DX. Pode-se notar que os controladores implementados obtiveram sucesso nos testes realizados, ou

seja, o robô navegou e atingiu o seu objetivo de controle.

Especificamente no caso de desvio de obstáculos durante a navegação em busca de um ponto destino dado, foram aplicados dois métodos, a saber, o desvio tangencial e o desvio não tangencial de obstáculos. Ambos os métodos se mostraram eficientes em atingir o objetivo de controle, ou seja, levar o robô ao seu ponto destino evitando qualquer colisão. Ambos os métodos se revelaram bastante eficientes, de acordo com experimentos de validação apresentados.

Ainda considerando os experimentos apresentados, mereceu destaque o erro de velocidades linear e angular verificado (considerando os valores medidos pela odometria do robô e o comando de velocidade enviado aos atuadores do robô). Caso fosse considerado o modelo dinâmico do robô, e não somente seu modelo cinemático, a diferença existente entre as velocidades calculadas pelo controlador e as velocidades efetivamente alcançadas pelo robô, obtidas por meio da sua odometria, poderia ser, pelo menos, amenizado.

REFERÊNCIAS

- R. Carelli and E.O. Freire. Corridor navigation and wall-following stable control for sonar-based mobile robots. *Robotics and Autonomous Systems*, 45:235–247, 2003.
- A. Ferreira. *Desvio Tangencial de Obstáculos para Um Robô Móvel Navegando em Ambientes Semi-estruturados*. Master's thesis, UFES, 2004.
- E.O. Freire. *Controle de Robôs Móveis por Fusão de Sinais de Controle Usando Filtro de Informação Descentralizado*. PhD thesis, UFES, 2002.
- F.G. Pereira. *Navegação e Desvio de Obstáculos Usando um Robô Móvel Dotado de Sensor de Varredura Laser*. Master's thesis, UFES, 2006.
- H.A. Secchi. *Control de Vehículos Autoguiados con Realimentación Sensorial*. Master's thesis, Facultad de Ingeniería de la Universidad Nacional de San Juan, San Juan - Argentina, 1998.