

ESTABILIZAÇÃO EM TEMPO REAL DE UM ROBÔ MÓVEL UTILIZANDO CONTROLE PREDITIVO BASEADO EM MODELO

JORGE AUGUSTO VASCONCELOS ALVES*, WALTER FETTER LAGES*

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

Emails: jorge@ece.ufrgs.br, fetter@ece.ufrgs.br

Abstract— This paper proposes a model predictive control (MPC) strategy for a differential-drive mobile robot. It is well known that for such a system not any point can be feedback stabilized by a smooth time-invariant control law. By using model predictive control, an appropriate control law is implicitly obtained. Furthermore, the system physical constraints on state and inputs are dealt with in a straightforward way. The optimization problem is solved by sequential quadratic programming (SQP) and experimental results show that the control signal computation can be performed under the real-time requirements.

Keywords— Real-time systems, Predictive control, Nonlinear control systems, Mobile robots, Digital Control

Resumo— Este trabalho apresenta uma estratégia de controle preditivo baseado em modelo (MPC) para um robô de acionamento diferencial. É sabido que tal sistema não pode ser estabilizado assintoticamente por uma lei de controle suave e invariante no tempo. Utilizando controle preditivo baseado em modelo uma lei de controle apropriada é obtida implicitamente. Adicionalmente, as restrições físicas do sistema são tratadas de maneira direta. O problema de otimização é resolvido por um algoritmo de programação quadrática seqüencial (SQP) e resultados experimentais mostram que a lei de controle pode ser computada dentro dos requisitos de tempo real do sistema.

Palavras-chave— Sistemas de tempo real, Controle preditivo, Sistemas de controle não lineares, Robôs móveis, Controle digital

1 Introdução

Apesar da aparente simplicidade do modelo cinemático de um robô móvel, o projeto de leis de controle estabilizantes é um desafio considerável. De acordo com as condições de Brockett (Brockett, 1982), uma lei de controle invariante no tempo e continuamente diferenciável não pode ser obtida. Para contornar estas limitações, estratégias de controle não-suaves ou variantes no tempo foram propostas (Bloch and McClamroch, 1989; Samson and Ait-Abderrahim, 1991; Canudas de Wit and Sørvalen, 1992; Yamamoto and Yun, 1994; M'Closkey and Murray, 1997). Trabalhos recentes sobre controle robusto e adaptativo de robôs móveis podem ser encontrados em (Dixon et al., 2004) e (Dong and Kuhnert, 2005).

Entretanto, em implementações realistas é difícil obter-se um bom desempenho devido a restrições que surgem naturalmente nas entradas e nos estados. Normalmente o controlador é projetado assumindo-se que não há limitações nas entradas ou nos estados e, no máximo, restrições reais são levadas em consideração posteriormente. Ao utilizar controle preditivo baseado em modelo (MPC) restrições podem ser consideradas de maneira direta ao gerar a lei de controle. Para um robô móvel isto é um aspecto importante pois a posição do robô pode ser restringida para que pertença a uma região de operação segura e uma ação de controle dentro dos limites dos atuadores pode ser gerada.

Além disso, transformações do modelo do sistema dinâmico para formas encadeadas ou de potência (Bloch and McClamroch, 1989) não são mais necessárias, o que torna o ajuste de parâmetros para o MPC mais intuitivo.

Trabalhos sobre a aplicação de MPC em robôs móveis são recentes e escassos (Ollero and Amidi, 1991; Normey-Rico et al., 1999; Essen and Nijmeijer, 2001; Gu and Hu, 2005). Uma possível causa é o modelo de um robô móvel ser não-linear e apesar de controle preditivo baseado em modelo não-linear (NMPC) ter sido desenvolvido (Mayne et al., 2000; Allgöwer et al., 1999), o custo computacional é muito maior que na versão linear. No NMPC o problema de programação não-linear a ser resolvido *on-line* é não-convexo e o número de variáveis de decisão é muito maior que na sua variante linear. Neste trabalho é proposta uma estratégia para se contornar alguns dos problemas relacionados ao uso de MPC para sistemas não-lineares não-holonômicos. Um algoritmo de programação quadrática seqüencial (SQP) foi escolhido devido a sua capacidade de encontrar um mínimo exigindo poucas avaliações das funções relacionadas ao problema (Barclay et al., 1997).

Em um trabalho anterior (Kühne, Lages and Gomes da Silva Jr., 2004), mostrou-se que o esforço computacional necessário para se executar o algoritmo de MPC proposto é tal que uma implementação em tempo real é possível. Entretanto, os resultados foram baseados no número de ope-

rações em ponto-flutuante necessárias (conforme relatado pelo Matlab) para se calcular a lei de controle e em uma estimativa da performance de um processador em Mflops. Neste trabalho, a viabilidade da abordagem é demonstrada pela implementação em tempo real, com medições de tempo feitas no sistema em funcionamento.

Em Bloch and McClamroch (1989) é mostrado que o sistema não-holonômico e não linear relativo a um robô de acionamento diferencial é controlável, i.e., pode-se atingir qualquer estado final a partir de qualquer estado inicial através de entradas com amplitude limitada. É fácil verificar que quando o robô não está se movendo a linearização em torno de um ponto de operação estacionário não é controlável. Apesar de ser possível linearizar o sistema através de realimentação de estados para a posição $([x \ y])$, somente (Kühne, Claro, Suess and Lages, 2004), considera-se aqui o problema mais geral no qual a posição deve ser atingida com uma orientação dada, ou seja, a referência para θ deve ser satisfeita em regime permanente.

2 Modelo do robô móvel

O modelo cinemático de um robô móvel com acionamento diferencial (vide Fig. 1) é dado por (1) (Campion et al., 1996):

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ w \end{bmatrix} = f(\mathbf{x}, \mathbf{u}) \quad (1)$$

onde $\mathbf{x} \triangleq [x \ y \ \theta]^T$ descreve a configuração (posição e orientação) do centro do eixo das rodas, C , com relação a um sistema de coordenadas inercial global $\{O, X, Y\}$. $\mathbf{u} \triangleq [v \ w]^T$ é a entrada de controle, onde v e w são as velocidades linear e angular, respectivamente.

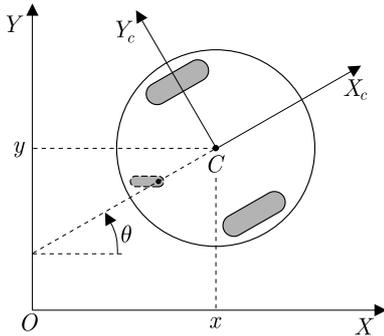


Figura 1: Sistema de coordenadas do robô móvel.

Devido ao fato de que os esquemas MPC apresentados consideram sistemas em tempo discreto, uma discretização de (1) é necessária. Um modelo em tempo discreto foi obtido utilizando-se a aproximação de Euler do modelo original com a forma:

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)) \quad (2)$$

onde k é o passo de amostragem.

3 O algoritmo MPC

A essência do MPC é se otimizar previsões do comportamento do processo a partir de uma seqüência de sinais de controle futuros. Tal previsão é feita com base em um modelo do sistema durante um intervalo de tempo finito, chamado *horizonte de previsão*. A cada passo de amostragem, o controlador MPC gera uma seqüência de controle ótima resolvendo um problema de otimização. O primeiro elemento da seqüência é aplicado no sistema. O problema é resolvido novamente no próximo instante de amostragem utilizando medidas atualizadas do processo e deslocando o horizonte de previsão.

A função-objetivo a ser minimizada é:

$$\Phi(k) = \sum_{j=1}^N \mathbf{x}^T(k+j|k) \mathbf{Q} \mathbf{x}(k+j|k) + \mathbf{u}^T(k+j-1|k) \mathbf{R} \mathbf{u}(k+j-1|k), \quad (3)$$

onde N é o horizonte de previsão e $\mathbf{Q} \geq 0$, $\mathbf{R} > 0$ são as matrizes de ponderação. A notação $a(m|n)$ denota a previsão do valor de a no instante m com base em medidas obtidas no instante n .

Assim, o problema de otimização pode ser formalizado como encontrar $\bar{\mathbf{u}}^*$ tal que:

$$\bar{\mathbf{u}}^* = \arg \min_{\bar{\mathbf{u}}} \{\Phi(k)\} \quad (4)$$

com

$$\bar{\mathbf{u}} \triangleq \begin{bmatrix} \mathbf{u}(k|k) \\ \mathbf{u}(k+1|k) \\ \vdots \\ \mathbf{u}(k+N-1|k) \end{bmatrix} \quad \bar{\mathbf{u}}^* \triangleq \begin{bmatrix} \mathbf{u}^*(k|k) \\ \mathbf{u}^*(k+1|k) \\ \vdots \\ \mathbf{u}^*(k+N-1|k) \end{bmatrix}$$

O problema de minimizar (3) é resolvido a cada passo de amostragem k , gerando uma seqüência ótima de controle $\{\bar{\mathbf{u}}^*(k|k), \dots, \bar{\mathbf{u}}^*(k+N-1|k)\}$ e o custo ótimo $\Phi^*(k)$. A lei de controle é implicitamente dada pelo primeiro elemento da seqüência de entrada ótima, ou seja, $\bar{\mathbf{u}}^*(k|k)$.

Controle preditivo baseado em modelo se baseia na premissa de que por um pequeno intervalo de tempo o comportamento do sistema real e do modelo é o mesmo. Para que esta premissa seja verdadeira a diferença entre estes comportamentos deve ser pequena. Obviamente para qualquer sistema real há limitações físicas para as entradas de controle. Conseqüentemente, para evitar grandes discrepâncias, estas limitações devem ser consideradas ao calcular as entradas de controle. Isto pode ser feito de maneira direta definindo-se limites superiores e inferiores para cada entrada de controle. O problema de otimização deve ser resolvido respeitando estes limites.

Entretanto, para um robô de acionamento diferencial os limites são na realidade as velocidades das rodas. Em (Kühne et al., 2005) as restrições consideradas foram relacionadas a v e w diretamente, o que leva a uma região de atuação maior que a fisicamente possível, conforme mostrado na Figura 2. Restrições expressas nestas variáveis não expressam a realidade uma vez que para velocidade linear máxima as duas rodas estão na velocidade angular máxima φ_{max} , portando a velocidade angular w tem que ser nula.

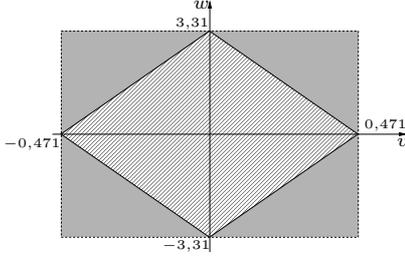


Figura 2: Restrições nas rodas: limitações em φ_{min} e φ_{max} (hachurada) e limitações em v_{max} e w_{max} (cinza).

As velocidades das rodas esquerda e direita φ_l e φ_r podem ser expressas como combinação linear das velocidades linear e angular:

$$\mathbf{u}(k) = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ -\frac{r}{l} & \frac{r}{l} \end{bmatrix} \begin{bmatrix} \varphi_l(k) \\ \varphi_r(k) \end{bmatrix} = \mathbf{P}\varphi(k)$$

onde l é a distância entre as rodas e r é o raio das mesmas. Considerando-se que as duas rodas são iguais e tem os mesmos limites de velocidade máximo e mínimo φ_{min_i} e φ_{max_i} pode-se descrever as restrições relativas às entradas como

$$\varphi_{min} \leq \varphi(k) \leq \varphi_{max}, \quad (5)$$

com $\varphi_{min} \triangleq [\varphi_{min_i} \quad \varphi_{min_i}]^T$ e $\varphi_{max} \triangleq [\varphi_{max_i} \quad \varphi_{max_i}]^T$. Substituindo-se $\varphi(k)$ por $\mathbf{P}^{-1}\mathbf{u}(k)$ obtém-se:

$$\varphi_{min} \leq \mathbf{P}^{-1}\mathbf{u}(k) \leq \varphi_{max}, \quad (6)$$

que é a expressão utilizada neste trabalho para restrições nas entradas.

4 Implementação em tempo real

O robô móvel Twil, desenvolvido em nossos laboratórios é considerado neste trabalho. Ele tem formato cilíndrico com 1,35 m de altura e 30 cm de raio. O *software* implementando o controlador NMPC foi escrito em C++. Para assegurar a temporização, uma extensão de tempo real para o kernel do Linux chamada RTAI (Dozio and Mantegazza, 2003) foi utilizada. O problema de otimização foi resolvido utilizando-se a biblioteca donlp2 (Spellucci, 1998).

Todos os resultados consideram as matrizes de peso como sendo $\mathbf{Q} = \text{diag}(1; 1; 0,5)$ e $\mathbf{R} = 0,05\mathbf{I}_{2 \times 2}$. O horizonte de predição utilizado foi $N = 5$, a não ser onde estiver especificado de outra forma. As restrições de velocidade para as rodas são $\varphi_{min_i} = -2\pi$ rad/s e $\varphi_{max_i} = 2\pi$ rad/s.

Conforme (Kühne, 2005), o esquema MPC descrito na Seção 3 gera um sistema realimentado que apresenta erro com relação ou a x ou a y . Isto está relacionado a função-custo (3) onde $\mathbf{x}(k)$ é o estado do sistema, como descrito por (1). Próximo do eixo OY , para y suficientemente pequeno, qualquer ação a ser tomada é penalizada por (3), portanto o robô não se move.

Em (Kühne et al., 2005) foi apresentado um controlador NMPC que utiliza um conjunto de coordenadas polares na função-objetivo. O conjunto $\{x \ y \ \theta\}$ é transformado no conjunto descontínuo $\{e \ \psi \ \alpha\}$, com:

$$\begin{aligned} e &\triangleq \sqrt{x+y} \\ \psi &\triangleq \text{atan2}(y, x) \\ \alpha &\triangleq \theta - \psi. \end{aligned} \quad (7)$$

Pode-se modificar (3) para

$$\begin{aligned} \Phi(k) &= \sum_{j=1}^N \mathbf{x}_p^T(k+j|k) \mathbf{Q} \mathbf{x}_p(k+j|k) \\ &+ \mathbf{u}^T(k+j-1|k) \mathbf{R} \mathbf{u}(k+j-1|k), \end{aligned} \quad (8)$$

com $\mathbf{x}_p \triangleq [e \ \psi \ \alpha]$.

Diferentemente da variante utilizando coordenadas cartesianas, obtém-se um sistema realimentado sem erro em regime permanente. Não obstante, o robô somente se aproxima da origem pelo semiplano direito, ou seja, com $x > 0$ e, conseqüentemente, $v < 0$. Pode-se verificar isto melhor na Figura 3. O robô está inicialmente a uma distância de 3 m da origem com $\theta(0) = 0$ para todas as trajetórias. Observa-se que o robô entra no semiplano direito antes de se aproximar da origem para qualquer condição inicial. É desejável que o robô seja também capaz de atingir o ponto de equilíbrio pelo semiplano esquerdo, com velocidade linear positiva.

O comportamento do sistema realimentado utilizando NMPC e coordenadas polares é também resultado da minimização da função-custo utilizada na sua formulação. Ao minimizar ψ e α faz-se com que o robô tenda a adotar uma posição sobre o lado positivo do eixo OX , com $\theta = 0$. Apesar de se aproximar do ponto $[0 \ 0]^T$ mais rapidamente que outros controladores factíveis quando localizado inicialmente no semiplano direito, o robô percorrerá uma trajetória mais longa que a necessária se posicionado no outro semiplano no início da trajetória. Devido a este problema outro controlador será proposto.

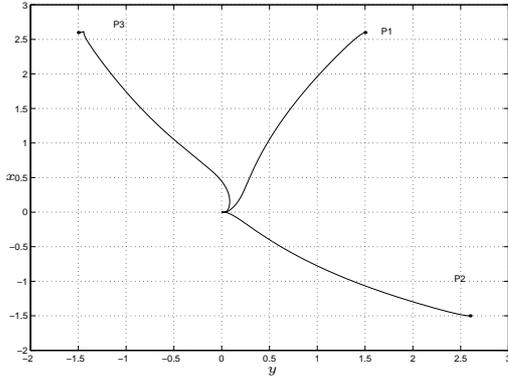


Figura 3: Trajetórias para diferentes condições iniciais, NMPC em coordenadas polares de (Kühne et al., 2005).

A transformação de coordenadas alternativa $\{e \ \psi \ \alpha\}$, definida como

$$e \triangleq \sqrt{x+y}$$

$$\psi \triangleq \begin{cases} \text{atan2}(y, x) & \text{se } x(0) \geq 0, \\ \text{atan2}(-y, -x) & \text{se } x(0) < 0, \end{cases}$$

$$\alpha \triangleq \theta - \psi,$$

é exatamente igual a expressa por (7) quando o robô está no semiplano direito no tempo inicial. Por outro lado, se o robô está no outro semiplano uma transformação similar é utilizada, porém simétrica com relação à origem. Pontos em torno do lado negativo do eixo OX têm valores baixos de ψ enquanto que o máximo valor absoluto desta variável estará relacionado a pontos ao redor do lado negativo do mesmo eixo. A performance difere do sistema realimentado utilizando esta transformação de coordenadas difere da do sistema utilizando o NMPC em coordenadas polares conforme (Kühne et al., 2005) para $x(0) < 0$. Resultados obtidos com o novo controlador podem ser verificados na Figura 4.

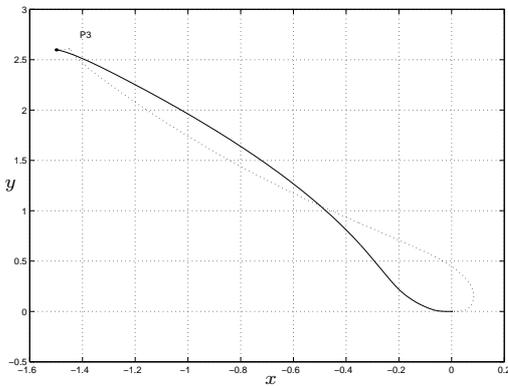


Figura 4: NMPC em coordenadas polares de (Kühne et al., 2005) (linha tracejada) e NMPC em coordenadas polares alteradas (linha contínua).

Pode ser visto na Figura 4 que o estado con-

verge assintoticamente para a origem. Na Figura 5 pode ser visto que as entradas satisfazem as restrições. A entrada em relação a u e v é mostrada na Figura 6.

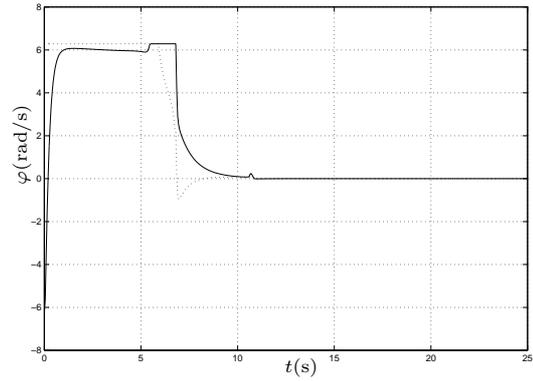


Figura 5: Velocidades das rodas para NMPC em coordenadas polares alteradas. Linha contínua: $\varphi_r(t)$. Linha tracejada: $\varphi_l(t)$.

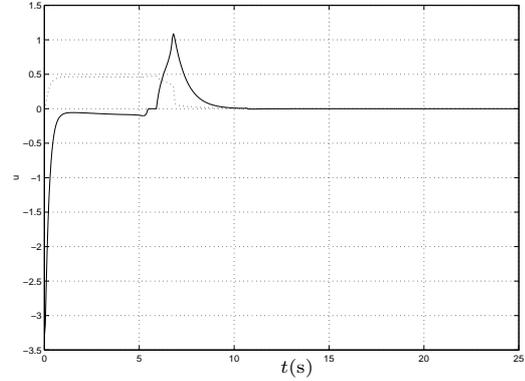


Figura 6: Entradas do sistema para NMPC em coordenadas polares alteradas. Linha contínua: $v(t)$. Linha tracejada: $w(t)$.

Uma medida usual de convergência é o erro médio quadrático (Essen and Nijmeijer, 2001). É definido como

$$\varepsilon \triangleq \frac{1}{K_f} \sum_{k=0}^{K_f} \|\mathbf{x}\|^2,$$

onde K_f é o número de passos necessário para percorrer a trajetória e $\|\cdot\|$ denota a norma euclidiana. A Tabela 1 mostra resultados relacionado o erro e a carga computacional em função do horizonte de predição. Mais especificamente, constam: a) numero de operações em ponto flutuante necessários para calcular a lei de controle a cada passo de amostragem, conforme Matlab, b) tempo estimado de computação, c) tempo real de computação e d) erro médio quadrático. Os tempos estimados de computação foram obtidos considerando uma performance de 895,8203 Mflops para um processador AMD Athlon 64 4000+ índice este obtido através do programa Flops (Aburto, 1992).

Tabela 1: Horizonte de predição \times tempo de computação.

N	flo(médio)	flo(max)	Tempo de computação (s)				ε
			Estimado(médio)	Estimado(max)	Real(médio)	Real(max)	
2	$57,82 \times 10^3$	$1,451 \times 10^6$	$64,55 \times 10^{-6}$	$1,620 \times 10^{-3}$	0,0051	0,0091	1,158
3	$199,1 \times 10^3$	$4,228 \times 10^6$	$222,2 \times 10^{-6}$	$4,719 \times 10^{-3}$	0,0022	0,0103	0,9058
4	$499,1 \times 10^3$	$7,539 \times 10^6$	$577,1 \times 10^{-6}$	$8,416 \times 10^{-3}$	0,0078	0,0317	0,9045
5	$1,143 \times 10^6$	$18,08 \times 10^6$	$1,277 \times 10^{-3}$	$20,18 \times 10^{-3}$	0,0093	0,0410	0,8619
10	$14,94 \times 10^6$	$143,4 \times 10^6$	$16,68 \times 10^{-3}$	$160,0 \times 10^{-3}$	0,0363	0,1644	0,8570
15	—	—	—	—	0,0864	0,3361	0.8834

Foi ajustado o número máximo de iterações permitidas a biblioteca donlp2 com o objetivo de limitar o tempo de computação. Foi verificado que um limite máximo de 40 iterações tem pouca influência sobre a resposta do sistema realimentado para $N = 5$. A alteração foi importante, porém, uma vez que o cálculo da lei de controle muitas vezes demorava tempo superior ao período de amostragem, que é o maior *deadline* possível para a aplicação do sinal de controle. Ao contrário do MPC linearizado (Lages and Alves, 2006), as variações no tempo de computação no NMPC se dão devido ao processo de otimização, tendo o escalonamento pouca influência. A solução do problema de otimização pode tomar mais ou menos tempo dependendo do estado atual.

Os dados contidos na Tabela 1 mostram que um computador recente é capaz de rodar um controlador NMPC para um robô móvel. De acordo com (Kühne et al., 2005), períodos de amostragem da ordem de 50 ms são adequados, e o algoritmo MPC proposto pode ser calculado para $N = 5$ em cerca de 40 ms. Entretanto, o cálculo toma 80% do período de amostragem, o que é significativo. Ainda assim, não foram feitas tentativas de otimizar os cálculos relacionados a minimização, como ajustar os parâmetros do algoritmo ou eliminar possíveis cálculos desnecessários presentes nas funções relacionadas às restrições e ao custo. Além disso, um processador de 64 bits foi utilizado, porém operando no modo 32 bits.

Similarmente ao que acontece com MPC Linearizado (Lages and Alves, 2006), uma estimativa baseada no número de operações em ponto flutuante mostra-se pouco confiável. Os tempos de computação foram muito mais longos que o estimado, como pode ser verificado na Tabela 1. Por outro lado, o tempo de computação apresenta grandes variações dependendo do valor do estado no instante de amostragem em questão. Para o LMPC um processador Athlon 3200+ resolve o problema de minimização em média em menos de 1,0 ms, sendo que há poucas variações no tempo de computação.

5 Conclusões

Este trabalho apresentou aplicações de MPC para o problema de estabilização em um ponto de um

robô móvel não-holonômico. Um novo controlador, considerando uma nova função-custo, foi apresentado, melhorando a resposta do sistema realimentado para condições iniciais no semiplano esquerdo. A relação entre a função-custo e o desempenho do sistema realimentado foi analisada para controladores MPC considerando o problema em questão. Foram propostas modificações para outros controladores de modo a considerar restrições mais realistas para as entradas. Os sinais de controle obtidos foram tais que satisfaziam as restrições.

A escolha de um controlador MPC é bastante justificada devido a algumas vantagens: a maneira direta na qual se pode considerar restrições nas entradas e estados; transformações de coordenadas para a forma encadeada ou de potência não são necessárias; o controlador MPC implicitamente gera um sinal de controle que lida com as condições de Brockett.

A viabilidade do esquema proposto foi demonstrada por uma implementação em tempo real, e dados da performance mostraram que processadores recentes e disponíveis no mercado possuem capacidade de processamento suficiente para satisfazer os requisitos de tempo. Particularmente, foi possível resolver o problema de minimização para um horizonte de predição igual a 5, adequado para o problema, para um período de amostragem de 50,0 ms.

Referências

- Aburto, A. (1992). FLOPS C program (double precision) v2.0 18 dec 1992.
*ftp://ftp.nosc.mil/pub/aburto/flops
- Allgöwer, F., Badgwell, T. A., Qin, J. S., Rawlings, J. B. and Wright, S. J. (1999). Non-linear predictive control and moving horizon estimation: an introductory overview, in P. M. Frank (ed.), *Advances in Control: Highlights of ECC'99*, Springer-Verlag, New York, pp. 391–449.
- Barclay, A., Gill, P. E. and Rosen, B. (1997). Sqp methods and their application to numerical optimal control, *Technical Report NA 97-3*, Dept of Mathematics, University of California, San Diego, USA.

- Bloch, A. M. and McClamroch, N. H. (1989). Control of mechanical systems with classical nonholonomic constraints, *Proceedings of the 28th IEEE American Conference on Decision and Control*, Tampa, FL, pp. 201–205.
- Brockett, R. W. (1982). *New Directions in Applied Mathematics*, Springer-Verlag, New York.
- Campion, G., Bastin, G. and D’Andréa-Novel, B. (1996). Structural properties and classification of kinematic and dynamical models of wheeled mobile robots, *IEEE Transactions on Robotics and Automation* **12**(1): 47–62.
- Canudas de Wit, C. and Sørvalen, O. J. (1992). Exponential stabilization of mobile robots with nonholonomic constraints, *IEEE Transactions on Automatic Control* **37**(11): 1791–1797.
- Dixon, W. E., de Queiroz, M. S., Dawson, D. M. and Flynn, T. J. (2004). Adaptive tracking and regulation of a wheeled mobile robot with controller/update law modularity, *IEEE Control Systems Technology* **12**(1): 138–147.
- Dong, W. and Kuhnert, K.-D. (2005). Robust adaptive control of nonholonomic mobile robot with parameter and nonparameter uncertainties, *IEEE Transactions on Robotics* **21**(2): 261–266.
- Dozio, L. and Mantegazza, P. (2003). Linux real time application interface (RTAI) in low cost high performance motion control, *Proceedings of the Motion Control 2003 Conference*, Associazione Nazionale Italiana per l’Automazione, ANIPLA, Milano, Italy.
- Essen, H. V. and Nijmeijer, H. (2001). Non-linear model predictive control of constrained mobile robots, *Proceedings of the European Control Conference*, EUCA, EUCA, Porto, Portugal, pp. 1157–1162.
- Gu, D. and Hu, H. (2005). A stabilizing receding horizon regulator for nonholonomic mobile robots, *IEEE Transactions on Robotics* **21**(5): 1022–1028.
- Kühne, F. (2005). *Controle preditivo de robôs móveis não-holonômicos*, Tese de mestrado, Universidade Federal do Rio Grande do Sul, Porto Alegre, RS. Orientador: João Manoel Gomes da Silva Jr. and Walter Fetter Lages.
- Kühne, F., Claro, C. A., Suess, S. and Lages, W. F. (2004). Estimação e controle da posição de um robô móvel utilizando filtro de kalman descentralizado, *Anais do XV Congresso Brasileiro de Automática*, Gramado, RS.
- Kühne, F., Lages, W. F. and Gomes da Silva Jr., J. M. (2004). Model predictive control of a mobile robot using linearization, *Proceedings of the Mechatronics and Robotics 2004*, IEEE Press, Aachen, Germany, pp. 525–530.
- Kühne, F., Lages, W. F. and Gomes da Silva Jr., J. M. (2005). Point stabilization of mobile robots with nonlinear model predictive control, *Proceedings of the IEEE International Conference on Mechatronics & Automation*, IEEE Press, Niagara Falls, Canada, pp. 1163–1168.
- Lages, W. F. and Alves, J. A. V. (2006). Real-time control of a mobile robot using linearized model predictive control, *4th IFAC Symposium on Mechatronic Systems*, Elsevier, Heidelberg, Germany, pp. 968–973.
- Mayne, D. Q., Rawlings, J. B., Rao, C. V. and Scokaert, P. O. M. (2000). Constrained model predictive control: stability and optimality, *Automatica* **36**(6): 789–814.
- M’Closkey, R. T. and Murray, R. M. (1997). Exponential stabilization of driftless control systems using homogeneous feedback, *IEEE Transactions on Automatic Control* **42**(5): 614–628.
- Normey-Rico, J. E., Gómez-Ortega, J. and Camacho, E. F. (1999). A smith-predictor-based generalised predictive controller for mobile robot path-tracking, *Control Engineering Practice* **7**(6): 729–740.
- Ollero, A. and Amidi, O. (1991). Predictive path tracking of mobile robots: application to the CMU navlab, *Proceedings of the 5th IEEE International Conference on Advanced Robotics*, Pisa, Italy, pp. 1081–1086.
- Samson, C. and Ait-Abderrahim, K. (1991). Feedback control of a nonholonomic wheeled cart in cartesian space, *Proceedings of the IEEE International Conference on Robotics and Automation*, IEEE Press, Sacramento, CA, pp. 1136–1141.
- Spellucci, P. (1998). A new technique for inconsistent qp problems in the sqp method, *Mathematical Methods of Operations Research* **47**(3): 355–400.
- Yamamoto, Y. and Yun, X. (1994). Coordinating locomotion and manipulation of a mobile manipulator, *IEEE Transactions on Automatic Control* **39**(6): 1326–1332.