

OPTIMIZAÇÃO DINÂMICA PARA PLANEAMENTO DE MOVIMENTO E CONTROLO DE VEÍCULOS AUTÓNOMOS

Sérgio Loureiro Fraga, João Borges de Sousa,
Fernando Lobo Pereira

*Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias s/n, 4200-465 Porto PORTUGAL
{slfraga,jtasso,flp}@fe.up.pt*

Abstract: Este artigo aborda o problema da geração de trajectórias em tempo-real para veículos autónomos e semi-autónomos. Neste trabalho, geração de trajectórias refere-se ao planeamento do movimento do veículo, parametrizado no tempo, e ao cálculo da referência para a entrada do sistema para que esse planeamento seja concretizado. Tempo-real, neste contexto, significa que o cálculo da trajectória é realizado suficientemente rápido de modo que o controlo do veículo em malha fechada possa ser efectuado por este processo. Este trabalho foi desenvolvido tendo em vista a sua utilização na geração de trajectórias em tempo real para um veículo autónomo subaquático existente no Laboratório de Sistemas e Tecnologia Subaquática da Faculdade de Engenharia da Universidade do Porto.

Keywords: Geração de trajectórias em tempo-real, projecto simultâneo de controlo e planeamento, veículos autónomos subaquáticos, sistemas diferencialmente planos, implementação em software.

1. INTRODUÇÃO

Este artigo aborda o problema da geração de trajectórias para veículos autónomos e semi-autónomos em tempo-real. Neste trabalho, geração de trajectórias refere-se ao planeamento do movimento do veículo de modo a incorporar restrições dinâmicas e cinemáticas bem como restrições de estado que surjam devido à presença de obstáculos. Para além disso, a geração de trajectórias deverá calcular a entrada do sistema que permita implementar o planeamento efectuado. Se estas duas tarefas, planeamento e cálculo da entrada, foram executadas em tempos compatíveis com os requisitos de controlo do veículo diz-se que a geração de trajectórias é feita em tempo real. Tempo-real significa então que o cálculo da trajectória é realizado suficientemente rápido

de modo que o controlo do veículo em malha fechada possa ser efectuado por este processo. Esta abordagem permite incorporar dois problemas distintos numa única infra-estrutura, advindo daí vantagens, não só de desempenho do sistema mas também de facilidade de projecto do sistema completo de controlo de veículos autónomos.

A abordagem para a resolução deste problema consiste na transformação de um problema de controlo óptimo num problema de programação não-linear através da aproximação das trajectórias por B-splines e sua discretização. Para além desta aproximação, são apresentados alguns resultados que permitem diminuir significativamente o número de variáveis a calcular no problema de programação não-linear. Esses resultados são baseados na teoria dos sistemas diferencialmente planos que permitem a determinação de variáveis

que parametrizam todo o estado e entrada do sistema, através de um número reduzido de variáveis.

Neste artigo é também apresentada uma aplicação de software que permite implementar os conceitos referidos anteriormente. O software denomina-se por "Nonlinear trajectory generation (NTG)" (Milam, 2003)(Milam *et al.*, 2000) e foi desenvolvido no California Institute of Technology (Caltech). Será apresentado um exemplo da aplicabilidade deste software num veículo subaquático do Laboratório de Sistemas e Tecnologia Subaquática (LSTS) da Faculdade de Engenharia da Universidade do Porto (FEUP), que foi fruto da cooperação existente entre este laboratório e o departamento "Control and Dynamical Systems" no Caltech.

Esta apresentação está organizada conforme se indica de seguida. Na secção 2 descreve-se a técnica que possibilita o cálculo das trajectórias de modo eficiente, na secção 3 é apresentado um exemplo de aplicação desta técnica e na secção 4 são apresentadas as principais conclusões deste trabalho.

2. GERAÇÃO DE TRAJECTÓRIAS EM TEMPO-REAL

A geração de trajectórias desempenha um papel muito importante num sistema de controlo. Gerar uma trajectória significa planear a trajectória desejada para o estado do sistema e a entrada que permitirá implementar essa mesma trajectória. Normalmente o planeamento das trajectórias é um processo demorado computacionalmente, sobretudo se se pretender planear trajectórias optimizando alguma função de custo sujeita a restrições. Pretende-se então resolver o seguinte problema:

$$\min_{x,u} F_i(t_0, x(t_0)) + \int_{t_0}^{t_f} F_{traj}(s, x(s), u(s)) ds + F_f(t_f, x(t_f)) \quad (1)$$

sujeito a:

$$\begin{aligned} \dot{x}(t) &= f(t, x(t), u(t)) \\ lb_i &\leq A_i \cdot \begin{bmatrix} x(t_0) \\ u(t_0) \end{bmatrix} \leq ub_i \\ lb_{traj} &\leq A_{traj} \cdot \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \leq ub_{traj} \\ lb_f &\leq A_f \cdot \begin{bmatrix} x(t_f) \\ u(t_f) \end{bmatrix} \leq ub_f \\ LB_i &\leq C_i(t_0, x(t_0), u(t_0)) \leq UB_i \\ LB_{traj} &\leq C_{traj}(t, x(t), u(t)) \leq UB_{traj} \\ LB_f &\leq C_f(t_f, x(t_f), u(t_f)) \leq UB_f \end{aligned}$$

onde $x : [t_0, t_f] \rightarrow R^n$, $u : [t_0, t_f] \rightarrow R^m$ e $t \in [t_0, t_f]$, sendo que F representa a função de

custo (inicial, integral e final), f representa as restrições dinâmicas do sistema, as matrizes A representam restrições lineares no instante inicial, na trajectória e no instante final e podem ser vectores linha ou matrizes de dimensão dependente do número de restrições (número de linhas) e número de variáveis de optimização (número de colunas), as funções C representam restrições não-lineares e lb , ub , LB , UB são valores constantes representando os limites superiores e inferiores das restrições lineares e não-lineares das trajectórias. Estas restrições não-lineares são também definidas para os instantes inicial, final e na trajectória das variáveis de optimização.

O objectivo do trabalho apresentado neste artigo será então resolver este problema de um modo eficiente, de tal modo que o controlo do sistema possa ser feito por intermédio desta computação. Assim, poder-se-á em cada instante de amostragem calcular uma nova trajectória que permita a re-optimização com base em novos dados dos sensores e, eventualmente, novas restrições no espaço dos estados. É usual denominar esta metodologia por optimização com horizonte rolante. A implementação destes conceitos permitem grande flexibilidade no projecto do sistema de planeamento de movimento bem como do próprio controlo do veículo, pois permite o sistema adaptar-se com relativa facilidade ao seu meio envolvente.

2.1 Abordagem

Nesta secção mostrar-se-á a técnica que permite a geração de trajectórias optimizadas de modo eficiente. Esta tecnologia constitui o núcleo do software NTG para a geração de trajectórias (*Nonlinear Trajectory Generation - NTG*, 2003). A descrição do núcleo do software NTG é importante para o desenvolvimento de novas aplicações usando este software. Isto ficará claro após apresentação do conteúdo desta secção e da secção onde se apresentará o exemplo de aplicação.

A técnica fundamental para o cálculo de trajectórias optimizadas de um modo eficiente reside na transformação do problema de controlo óptimo apresentado na equação 1 num problema de programação não-linear, que se apresenta a seguir:

$$\begin{aligned} \min_c f(c) \\ C_E(c) &= 0 \\ C_I(c) &\leq 0 \end{aligned} \quad (2)$$

onde c representam as variáveis discretas de optimização, f representa a função de custo e as funções C as restrições do problema. Para fazer essa transformação, parametriza-se as soluções do problema da equação 1 através de B-Splines

(Boor, 1978) e discretiza-se essas trajectórias para avaliar as restrições e a função de custo somente em alguns pontos. Daqui pode-se desde já concluir que a solução para este problema é uma aproximação ao problema apresentado na equação 1, dado que as restrições serão só satisfeitas em pontos discretos e a função de custo somente avaliada nesses mesmos pontos. Para além disso, o espaço das soluções está restringido às funções definidas pelas B-Splines e não a todas as funções do espaço. A parametrização das trajectórias por B-Splines permitirá definir uma trajectória através de um número finito de coeficientes viabilizando a transformação eficiente de um problema de dimensão infinita para um de dimensão finita. A escolha da parametrização por B-Splines traz muitas vantagens sobretudo porque o número de coeficientes necessários para descrever uma trajectória com uma determinada suavidade é menor do que com outras parametrizações, como por exemplo parametrizações usando polinómios. Outra vantagem relevante das B-Splines reside no facto destas curvas terem suporte local, ou seja, é possível modificar parte da trajectória sem que outras partes da trajectória sejam afectadas. Esta característica não está presente na parametrização por polinómios, onde modificando um coeficiente toda a trajectória vem afectada.

Assim, parametriza-se as variáveis que se pretende otimizar por B-Splines conforme é mostrado na equação seguinte:

$$z_i^{(d)}(t) = \sum_{j=1}^{N_i^c} B_{j,k_i}^{(d)}(t) \cdot c_j^i \quad (3)$$

Na equação precedente, B representa a base de funções das B-Splines de ordem k , c representa os coeficientes, d refere-se à ordem da derivada da variável e N_i^c representa o número de pontos de controlo da B-Splines ("knot points"). Conforme é possível observar pela equação precedente, a descrição das trajectórias e suas derivadas é feita com recurso a um número discreto e finito de pontos.

Usando esta representação, o problema de controlo óptimo vem então redefinido do seguinte modo:

$$\begin{aligned} & \min_{\tilde{Z}} \tilde{F}_i(t_0, \tilde{z}(t_0)) + \\ & \int_{t_0}^{t_f} \tilde{F}_{traj}(s, \tilde{z}(s)) ds + \tilde{F}_f(t_f, \tilde{z}(t_f)) \\ & G(t, \tilde{z}(t)) = 0 \\ & lb_i \leq \tilde{A}_i \cdot \tilde{z}(t_0) \leq ub_i \\ & lb_{traj} \leq \tilde{A}_{traj} \cdot \tilde{z}(t) \leq ub_{traj} \\ & lb_f \leq \tilde{A}_f \cdot \tilde{z}(t_f) \leq ub_f \end{aligned} \quad (4)$$

$$\begin{aligned} LB_i & \leq \tilde{C}_i(t_0, \tilde{z}(t_0)) \leq UB_i \\ LB_{traj} & \leq \tilde{C}_{traj}(t, \tilde{z}(t)) \leq UB_{traj} \\ LB_f & \leq \tilde{C}_f(t_f, \tilde{z}(t_f)) \leq UB_f \end{aligned}$$

onde $\tilde{z} : [t_0, t_f] \rightarrow R^{N_z}$ e \tilde{z} definido por:

$$\tilde{Z} = \left[z_1^{(0)} \dots z_1^{(d_1)} \dots z_{N_0}^{(0)} \dots z_{N_0}^{(d_{N_0})} \right]^T \quad (5)$$

sendo N_0 o número de variáveis de optimização e (d_i) o número de derivadas de cada uma das variáveis de optimização. Acrescentou-se o símbolo $\tilde{}$ no nome das variáveis para realçar o facto de que nesta nova formulação ser possível definir sistemas de ordem superior a 1 e, assim, incluir derivadas de variáveis de optimização nas funções de custo e nas funções representativas das restrições. A função G representa as restrições dinâmicas do problema num formato em que essas restrições aparecem sob a forma de uma restrição não-linear. Esta representação do problema de controlo óptimo é mais genérica dado ser possível transformá-la numa representação equivalente à da equação 1.

Com o problema assim definido é então possível transformá-lo num problema de programação não-linear. Para o fazer será ainda necessário definir os instantes τ onde as trajectórias serão avaliadas nas funções de custo e nas suas restrições. Assim, definindo:

$$\begin{aligned} \tilde{z}(\tau) & = \tilde{B}(\tau) \tilde{C} \\ \tau & \in [\tau_0, \dots, \tau_{N_{\tau-1}}] \end{aligned} \quad (6)$$

sendo C o conjunto de todos os coeficientes das B-Splines das variáveis de optimização, problema de controlo óptimo vem definido do seguinte modo:

$$\begin{aligned} & \min_{\tilde{C}} f(\tilde{C}) \\ LB & \leq \begin{bmatrix} \tilde{C} \\ A\tilde{C} \\ c(\tilde{C}) \end{bmatrix} \leq UB \end{aligned} \quad (7)$$

em que:

$$\begin{aligned} f(\tilde{C}) & = \tilde{F}_i(\tau_0, \tilde{z}(\tau_0)) + \\ & + \sum_{i=1}^{N_{\tau}} \tilde{F}_{traj}(\tau_i, \tilde{z}(\tau_i)) \cdot h_i + \tilde{F}_f(\tau_{N_{\tau-1}}, \tilde{z}(\tau_{N_{\tau-1}})); \\ LB & = [LB_c \ lb_i \ (lb_{traj})^1 \dots \\ & \dots (lb_{traj})^{N_{\tau}} \ lb_f \ LB_i \ (LB_{traj})^1 \dots (LB_{traj})^{N_{\tau}} \ LB_f]^T; \\ UB & = [UB_c \ ub_i \ (ub_{traj})^1 \dots \\ & \dots (ub_{traj})^{N_{\tau}} \ ub_f \ UB_i \ (UB_{traj})^1 \dots (UB_{traj})^{N_{\tau}} \ UB_f]^T; \end{aligned}$$

$$A = \begin{bmatrix} A_i \tilde{B}(\tau_0) \\ A_{traj} \tilde{B}(\tau_0) \\ \dots \\ A_{traj} \tilde{B}(\tau_{N_{\tau-1}}) \\ A_f \tilde{B}(\tau_{N_{\tau-1}}) \end{bmatrix} \quad c(\tilde{C}) = \begin{bmatrix} \tilde{C}_i(\tilde{z}(\tau_0)) \\ \tilde{C}_{traj}(\tilde{z}(\tau_0)) \\ \dots \\ \tilde{C}_{traj}(\tilde{z}(\tau_{N_{\tau-1}})) \\ \tilde{C}_f(\tilde{z}(\tau_{N_{\tau-1}})) \end{bmatrix}$$

A solução deste problema é o conjunto dos coeficientes das B-Splines \tilde{C} . Para se obter as trajectórias que constituem a solução do problema bastará usar a base das funções das B-Splines e aplicar os coeficientes calculados:

$$\tilde{z}(t) = \tilde{B}(t)\tilde{C} \quad (8)$$

2.2 Parametrização do espaço de optimização

Nesta secção ir-se-á mostrar algumas técnicas que permitem simplificar um problema de controlo óptimo permitindo maior eficiência na resolução do mesmo. Assim, vão-se apresentar duas técnicas que permitem diminuir o número de variáveis de optimização, bem como reduzir o número de restrições diferenciais existentes devido à própria dinâmica do sistema.

A primeira técnica baseia-se nas propriedades dos sistemas diferencialmente planos (Petit *et al.*, 2001) (Fliess *et al.*, 1995) (Fraga, 2002). Em sistemas desta classe é possível encontrar variáveis, denominadas por saídas planas, que permitem parametrizar todo o estado do sistema e a sua entrada. A dimensão das saídas planas é igual à dimensão da entrada do sistema pelo que constitui um espaço de optimização menor. Para além disto, com esta mudança de coordenadas o sistema fica equivalente a um sistema sem restrições diferenciais, o que também se traduz numa vantagem em termos de tempo de computação. Assim, denominando as saídas planas por p , tem-se que estas variáveis dependem do estado e da entrada do sistema e suas derivadas do seguinte modo:

$$p = \mathbf{P}(x, u, \dot{u}, \dots, u^{(l)}) \quad (9)$$

Por outro lado, a partir destas variáveis e suas derivadas é possível determinar o estado completo do sistema e a respectiva entrada:

$$\begin{aligned} x &= \mathbf{X}(p, \dot{p}, \dots, p^{(q)}) \\ u &= \mathbf{U}(p, \dot{p}, \dots, p^{(q)}). \end{aligned} \quad (10)$$

No caso das saídas planas serem função apenas da configuração do sistema o sistema denomina-se por sistema de configuração plana.

Usando esta mudança de coordenadas é necessário exprimir as restrições do problema e as funções de custo em função destas variáveis planas. Normalmente, este procedimento faz com que a função de custo e as restrições fronteira e de trajectória se tornem mais complexas. Este facto pode-se tornar numa desvantagem e poderá prejudicar o tempo de computação do problema. Muitas vezes é preferível usar parametrizações parciais por se verificar que o tempo computacional é menor (Petit *et al.*, 2001).

O desafio actual nesta área reside na verificação se um sistema é diferencialmente plano e caso o seja encontrar as saídas planas desse sistema. Existem já alguns resultados interessantes que permitem, por exemplo, fazer o paralelismo entre sistemas linearizáveis por realimentação e sistemas diferencialmente planos (Nieuwstadt, 1997). Existe também um catálogo de sistemas mecânicos (Murray *et al.*, 1995) onde foram encontradas as saídas planas de alguns sistemas mecânicos.

Para além desta técnica, existe também outra metodologia que permite diminuir a dimensão do espaço de optimização do problema de controlo óptimo, mas sem eliminar totalmente as restrições diferenciais do sistema. Trata-se pois de uma parametrização parcial conforme referido atrás. Esta técnica denomina-se por inversão da dinâmica (Petit *et al.*, 2001) e consiste em definir, caso seja possível, as entradas do sistema em função das variáveis de estado do sistema. Deste modo, não será necessário calcular as trajectórias das entradas pois estas serão uma função das outras variáveis de optimização. Este procedimento, para além de diminuir o espaço das variáveis de optimização, permite também a eliminação de algumas restrições diferenciais provocadas pela dinâmica do sistema. Esta técnica será usada no exemplo da secção seguinte.

3. EXEMPLO

Nesta secção será apresentado um exemplo onde mostrar-se-á a aplicabilidade dos conceitos referidos ao longo deste artigo. Serão mencionados aspectos importantes de projecto, tais como a correcta definição dos parâmetros das B-Splines e a transformação das variáveis de optimização noutras que aumente o desempenho do processo de optimização. Para além destes conceitos irá ser também apresentado o modo de definição do problema de optimização no software NTG. A problemática da definição de uma função de custo adequada para o planeamento de movimento será também abordada. O exemplo apresentado refere-se ao planeamento de movimento de um Veículo Subaquático Autónomo (VSA) existente no Laboratório de Sistemas e Tecnologia Subaquática da FEUP (figura 1).

Para simplificar a apresentação dos conceitos de geração de trajectórias, ir-se-á usar um modelo simplificado do plano horizontal do veículo subaquático do LSTS.

A definição das variáveis de estado e entrada para este veículo é a seguinte:



Fig. 1. Veículo do projecto PISCIS (LSTS-FEUP)

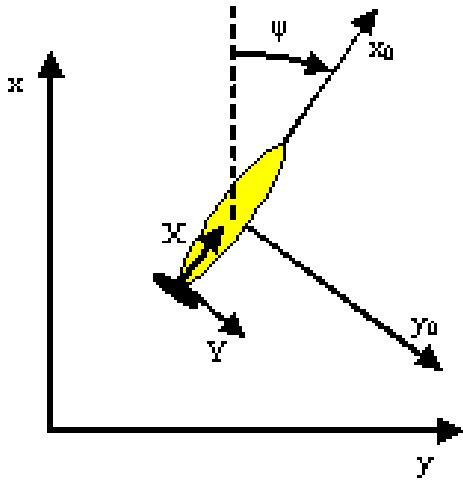


Fig. 2. Referencial inercial e referencial do corpo

$[x, y]$	posições do centro de massa (m)
ψ	orientação do veículo (rad)
X	actuação (direcção x do ref. do corpo)
Y	actuação (direcção y do ref. do corpo)
u	velocidade (direcção x do ref. do corpo)
v	velocidade (direcção y do ref. do corpo)

O sistema de coordenadas utilizado para a definição do movimento do VSA está de acordo com a notação de "The Society of Naval Architects and Marine Engineers (SNAME)". Para o plano horizontal este sistema de coordenadas é apresentado na figura 2.

Assume-se que o único propulsor do veículo exerce uma força constante no veículo X que equilibrará a força de arrastamento de tal modo que a velocidade do veículo na direcção x possa ser considerada constante $u = const.$ Dado considerar-se $X = const.$ a única actuação no sistema refere-se ao leme do veículo que afecta a orientação ψ e as variáveis x e y . Neste exemplo, desprezar-se-ão as forças hidrodinâmicas como o

arrastamento e a massa acrescentada bem como as forças de Coriolis.

O modelo utilizado neste exemplo é o seguinte:

$$\begin{aligned}\dot{x} &= \cos \psi \cdot u - \sin \psi \cdot v \\ \dot{y} &= \sin \psi \cdot u + \cos \psi \cdot v \\ J_z \cdot \ddot{\psi} &= -d \cdot Y \\ \dot{u} &= 0 \Rightarrow u(t) = u(0) \\ \dot{v} &\simeq \frac{Y}{m} \Rightarrow \dot{v} = -\frac{J_z}{d \cdot m} \cdot \ddot{\psi} \Rightarrow v = -\frac{J_z}{d \cdot m} \cdot (\psi - \psi(0))\end{aligned}$$

Neste exemplo considerou-se $u(0) = 1.5m \cdot s^{-1}$ e $d = 0.5m$ representa a distância entre o centro de massa do veículo e o seu leme. A formulação do problema de controlo óptimo é agora possível. Neste exemplo, optou-se por um problema de tempo mínimo já que esta situação corresponde à forma mais agressiva para a geração de trajectórias, em termos de exigência do sistema. Assim, o problema de controlo óptimo é formulado do seguinte modo (considerando $\dot{\psi}(0) = 0$):

$$\begin{aligned}\min t_f^2 & \quad (11) \\ \dot{x} &= 1.5 \cos \psi + \frac{J_z}{d \cdot m} \sin \psi \cdot \dot{\psi} \\ \dot{y} &= 1.5 \sin \psi - \frac{J_z}{d \cdot m} \cos \psi \cdot \dot{\psi} \\ x(0) &= y(0) = 0 \\ \psi(0) &= \dot{\psi}(0) = \ddot{\psi}(0) = 0 \\ x(t_f) &= y(t_f) = 0 \\ \psi(t_f) &= \pi; \dot{\psi}(t_f) = \ddot{\psi}(t_f) = 0 \\ -5 &\leq -\frac{J_z}{d} \ddot{\psi} \leq 5.\end{aligned}$$

A última equação refere-se à restrição na actuação, que neste caso é o valor máximo que o leme do veículo pode exercer (5N (Prester, 2001)). Como se pode observar na formulação do problema de tempo mínimo, foi possível usar a técnica descrita na secção 2.2 no que se refere à determinação da actuação a partir das variáveis de estado e respectivas derivadas ($Y = -\frac{J_z}{d} \ddot{\psi}$). Deste modo foi possível reduzir o espaço das variáveis de optimização numa variável. Observe-se também que houve uma restrição diferencial que foi eliminada. Assim, para este problema as variáveis de optimização são as seguintes:

$$z = [x \quad \dot{x} \quad y \quad \dot{y} \quad \psi \quad \dot{\psi} \quad \ddot{\psi} \quad t_f]. \quad (12)$$

Para a resolução deste problema, os parâmetros das B-Splines de cada trajectória são os seguintes:

	x	y	ψ	t_f
Suavidade (C^{s-1})	3	3	4	0
Nº de intervalos (l)	3	3	3	1
Ordem dos polinómios (k)	5	5	5	1
Nº de derivadas	1	1	2	0

Destes valores é possível concluir que o número de coeficientes das B-Splines a calcular no problema de programação não-linear é $\sum_{i=1}^n l_i(k_i - s_i) + s_i = 26$ (Boor, 1978). Quanto menor for o número de coeficientes a calcular maior será a eficiência computacional do problema. No entanto, um número reduzido de coeficientes significa também que o espaço de optimização estará bastante restringido o que poderá implicar que a solução encontrada tenha um custo mais elevado. A resolução deste compromisso permitirá a obtenção dos melhores resultados. A título de exemplo note-se que para a variável t_f é uma constante pelo que a ordem e o número de intervalos da B-Spline é igual a 1. Após definição do problema de controlo óptimo é então possível mostrar o resultado obtido quando se define este problema no NTG. Basicamente, a manobra que se pretende implementar é inverter a direcção do movimento do VSA em tempo mínimo com condições iniciais e finais bem definidas, satisfazendo as restrições dinâmicas do sistema e evitando que a actuação ultrapasse a sua saturação. A solução calculada por esta aplicação é representada na figura 3 onde são mostradas as trajectórias das variáveis de optimização (para simplificar a apresentação omitiram-se os resultados referentes às derivadas destas variáveis). O valor do tempo mínimo foi 17.6s. A partir da solução $\ddot{\psi}(t)$ é possível determinar a actuação no leme do veículo para inverter o sentido de navegação do veículo de modo que este regresse à posição inicial. Na figura 4 é apresentada a actuação Y do sistema e a trajectória do caminho solução deste problema. O cálculo destas trajectórias foi realizado num processador Intel Pentium 4 a 2.4GHz e teve um tempo de computação de 0.08s. O tempo de cálculo para este problema pode ser considerado suficientemente baixo de tal modo é possível fechar a malha de controlo através deste processo. Isto é o mesmo que dizer que será possível implementar um sistema de geração de trajectórias optimizadas em tempo-real.

4. CONCLUSÃO

Este artigo mostrou uma técnica para geração de trajectórias optimizadas. Foram apresentados os fundamentos desta técnica e mostrou-se como esta pode ser implementada efectivamente em veículos autónomos. O baixo tempo de computação poderá permitir o controlo do sistema por este processo. As principais desvantagens residem no facto de só se garantir que as restrições do sistema sejam

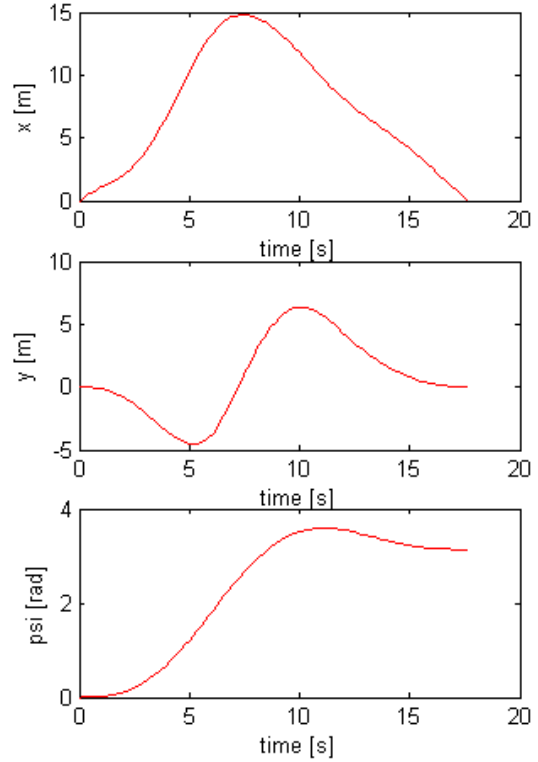


Fig. 3. Solução do problema de tempo mínimo

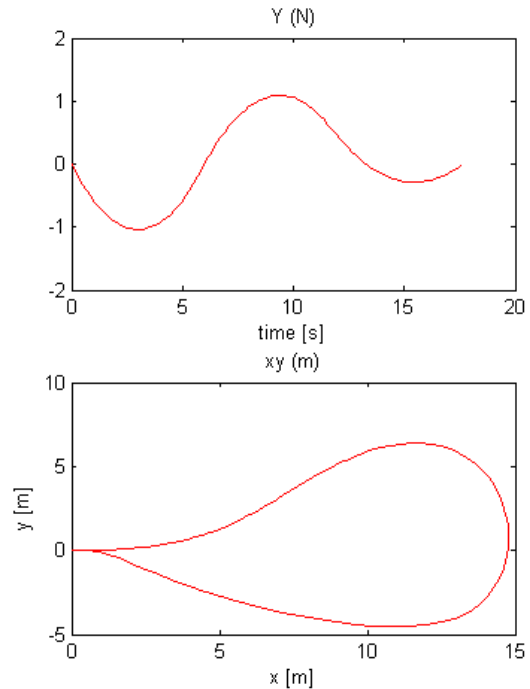


Fig. 4. Actuação e caminho do sistema para o problema de tempo mínimo

satisfeitas em pontos discretos das trajetórias. A possibilidade de não convergência da otimização traduz-se também numa desvantagem. A cada computação é dada como condição inicial a última trajetória calculada, que por vezes pode não ser suficiente. Se a otimização não convergir, usa-se a última trajetória calculada para controlar o sistema. Se este problema ocorrer repetidamente, o tempo de amostragem aumenta consideravelmente implicando perda de desempenho. Para se evitar que esta situação ocorra na prática é necessário simular o problema e analisar o desempenho desta otimização. A complexidade do modelo influencia o desempenho da otimização podendo haver necessidade da sua simplificação para a aplicação desta técnica. Futuramente, ir-se-á implementar um modelo mais aproximado do veículo o que permitirá a implementação deste software no computador de bordo do veículo. Em relação à evolução do algoritmo de otimização, está-se a estudar a aplicação de novas parametrizações que permitam baixar o tempo computacional da otimização.

5. AGRADECIMENTOS

O autor Sérgio Fraga gostaria de agradecer o apoio financeiro concedido pela Fundação Gulbenkian para o seu estágio no Caltech em 2003 e agradecer o apoio concedido pelo Professor Richard Murray para a realização desse estágio. Os autores gostariam de agradecer aos revisores deste artigo pelos seus comentários e sugestões.

REFERENCES

- Boor, C. De (1978). *A Practical Guide to Splines*. Springer-Verlag.
- Fliess, M., J. Levine, P. Martin and P. Rouchon (1995). Flatness and defect of nonlinear systems: Introductory theory and examples. In: *International Journal of Control*. pp. 61(6):1327–1360.
- Fraga, Sérgio Loureiro (2002). Trajectory generation - application to underwater vehicles. Master's thesis. Faculdade de Engenharia da Universidade do Porto.
- Milam, M., K. Mushambi and R. Murray (2000). A new computational approach to real-time trajectory generation for constrained mechanical systems. In: *Control and Decision Conference*.
- Milam, Mark (2003). Real-Time Optimal Trajectory Generation for Constrained Dynamical Systems. PhD thesis. Caltech.
- Murray, Richard, Muruhan Rathinam and Willem Sluis (1995). Differential flatness of mechanical control systems: A catalog of prototype systems. In: *ASME International Mechanical Engineering Congress and Exposition*.
- Nieuwstadt, Michiel J. Van (1997). Trajectory Generation for Nonlinear Control Systems. PhD thesis. California Institute of Technology, Pasadena, California.
- Nonlinear Trajectory Generation - NTG* (2003). www.cds.caltech.edu/ntg.
- Petit, N., M.B. Milam and R.M. Murray (2001). Inversion based constrained trajectory optimization. In: *5th IFAC Symposium on Nonlinear Control Systems*.
- Prestero, Timothy (2001). Verification of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle. Master's thesis. Massachusetts Institute of Technology.