

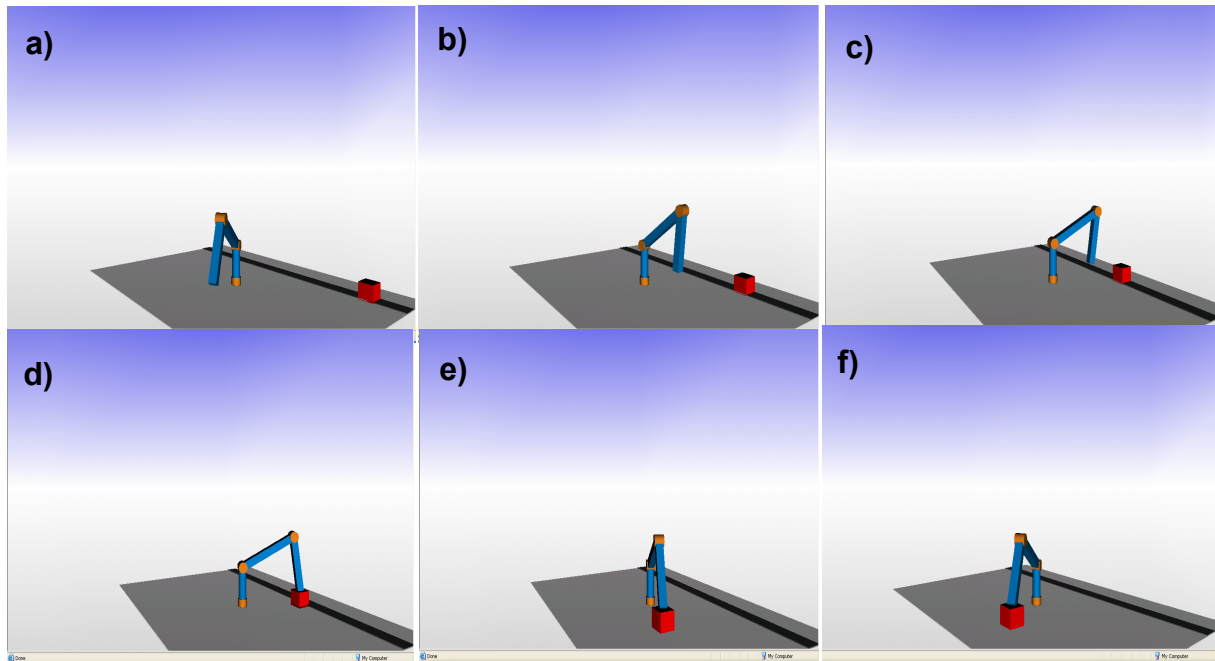


Figura 7.51 - Posição do manipulador e do objeto em diferentes instantes de tempo.

7.6.4 Conclusão

Este estudo apresentou uma proposta para a determinação da trajetória de um robô manipulador serial que tem como tarefa a interceptação de um objeto em movimento. Considera-se, além da interceptação, a otimização da potência mecânica necessária para a realização do movimento.

Foi especificado um critério para a avaliação da distância entre o efetuador e o objeto através do conceito de norma vetorial. A formulação proposta considera a posição e o instante de encontro como variáveis a serem determinadas durante o processo de otimização. A possibilidade de ajuste destes fatores oferece maior liberdade para o algoritmo na busca dos valores que resultam na trajetória ótima.

Em seguida, apresentou-se a formulação de um problema de otimização multicritério, o qual permite a consideração de vários objetivos simultâneos. Esta formulação prevê o uso de fatores de ponderação, o que permite priorizar algum índice de performance previamente escolhido, se o projetista assim o desejar.

O problema de otimização foi resolvido através da formulação de controle ótimo, onde as velocidades das coordenadas de junta representam o vetor de controle e a posição define o vetor de estado.

O exemplo numérico apresentado ilustra o uso da metodologia. A primeira fase da otimização considerou o critério de interceptação do obstáculo. Esta análise leva a um projeto viável. A segunda fase da otimização utiliza este resultado como informação inicial para a determinação de uma trajetória que otimize a potência mecânica durante o movimento.

O uso da potência mecânica como índice de performance leva à otimização da velocidade e do torque simultaneamente, considerando, portanto, tanto aspectos cinemáticos como dinâmicos do movimento.

Destaca-se que esta estratégia pode ser empregada em situações mais complexas, nas quais seja necessária a especificação da velocidade e orientação do manipulador durante o encontro com o objeto, por exemplo. Neste caso, pode-se utilizar a configuração ótima obtida como uma trajetória de referência para a realização dos ajustes requeridos. Pequenos ajustes da trajetória nas proximidades da trajetória ótima, em geral, não alteram de forma significativa o valor dos índices de performance estabelecidos.

Finalmente, destaca-se que, devido à característica altamente não linear das equações envolvidas na formulação proposta, qualquer alteração dos fatores de ponderação leva a diferentes configurações ótimas. Este procedimento permite a determinação de diferentes configurações de projeto possíveis, de acordo com a prioridade dada a cada objetivo.

CAPÍTULO VIII

ANÁLISE DE VIBRAÇÕES E ESTUDO DA TRAJETÓRIA DE ROBÔS MÓVEIS

A seção 8.1 apresenta uma proposta para o aumento da manipulabilidade, redução do efeito de vibração e consideração do erro associado à cinemática inversa durante o planejamento da trajetória. É proposto um novo modelo para a expressão da vibração, inspirado na teoria sobre dinâmica de rotores.

A tarefa a ser desempenhada é definida e descrita através de uma sequência de pontos cartesianos que devem ser atingidos pelo efetuador. O processo de otimização aplica transformações de corpo rígido ao conjunto de pontos no espaço cartesiano, resultando em uma sequência ótima de configurações cinemáticas que minimizam a vibração.

A cinemática inversa é calculada conforme descrito na seção 6.2 e o posicionamento da tarefa é aquele apresentado na seção 2.5 e utilizado posteriormente na seção 7.3.

Os experimentos numéricos confirmam que o posicionamento adequado da tarefa reduz o efeito de vibração. Como alterações no posicionamento da tarefa podem resultar em diferentes índices de manipulabilidade e diferentes perfis de velocidade, a metodologia procura obter uma forma ótima de utilização dos recursos (conforme descrito pelos índices de desempenho).

A seção 8.2 apresenta uma estratégia para determinação da trajetória de robôs móveis. Este estudo representa uma generalização dos estudos anteriores, na medida em que o planejamento *off-line* realizado até então é atualizado durante a realização da trajetória, caracterizando o planejamento *on-line*.

Os experimentos numéricos comprovam a viabilidade da estratégia na obtenção de uma trajetória contínua que prevê o desvio de obstáculos fixos e móveis.

O processo de ponderação dos diferentes objetivos (alvo e obstáculos) durante o movimento provê eficiência ao método para considerar cenários variados, assim como uma grande quantidade de obstáculos e também a existência de alvos móveis.

Inspirada em um método de programação não linear, o Método do Gradiente, esta proposta tem características que permitem sua extensão para o planejamento de trajetória de robôs manipuladores, sugerindo a possibilidade de generalização do conceito de planejamento de trajetória.

8.1 Redução de vibração pelo posicionamento da tarefa

A redução dos custos de produção e o aumento da produtividade estão entre os principais objetivos da automação industrial. Neste contexto, os robôs manipuladores seriais tem demonstrado ser ferramentas úteis.

Com o intuito de aumentar a produtividade, aplicações robóticas industriais têm sido estudadas na literatura a partir do movimento de percurso restrito e tempo ótimo, que consideram as restrições de torque dos atuadores. Nestas formulações, o torque de cada junta é considerado como uma entrada controlada, sendo que o esquema de controle em malha aberta resulta em controle do tipo *bang-bang* ou *bang-singular-bang* (Chen e Desrochers, 1989).

Entretanto, o aumento da velocidade resulta (em alguns casos) no aumento da vibração do efetuador. Isto é observado devido a diversos fatores, como a natureza do controle, flexibilidade das juntas, atrito das engrenagens, ou mesmo uma combinação destes fatores.

A presença da vibração não é desejada, pois pode degradar a performance do sistema e diminuir a vida útil de seus componentes.

Para minimizar o efeito vibracional, alguns trabalhos consideram o controle do movimento de robôs com juntas e/ou elementos flexíveis, com o intuito de melhorar a performance do robô ao realizar tarefas específicas.

Em Benosman *et al.* (2004), considera-se o problema do movimento *rest-to-rest* em tempo fixo para o manipulador planar flexível, onde a vibração residual do efetuador é minimizada. O enfoque consiste no cálculo de um controle com realimentação e não considera o efeito de amortecimento elástico interno. O controle de realimentação é utilizado para estabilizar localmente os estágios do movimento. Conforme mencionado na referência acima, o compensador de realimentação falha ao tentar controlar o sistema em um movimento *rest-to-rest* rápido.

Em Zollo *et al.* (2005) são apresentadas duas formulações teóricas para a lei de controle, considerando o ambiente de trabalho onde acontecem interações inesperadas, sendo ainda proposto um esquema de controle que reduz os efeitos da elasticidade.

Em Dadfarnia *et al.* (2004) é proposta uma estratégia baseada no controle de *Lyapunov* distribuído de dimensão infinita para a movimentação de um robô cartesiano, modelado como uma barra flexível contendo o suporte de uma base translacional. Um excitador de vibrações impõe movimento à base, enquanto uma pastilha piezelétrica (PZT) é utilizada como atuador na superfície da barra flexível para supressão da vibração residual.

A contribuição presente discute uma forma alternativa de reduzir os efeitos da vibração no efetuador de um robô manipulador serial. A metodologia proposta consiste no posicionamento ótimo da tarefa a ser realizada, de forma a minimizar a vibração do efetuador. Para atingir este objetivo, é formulado um modelo que descreve o erro de posicionamento angular de cada junta a partir da manipulabilidade do robô e das velocidades de junta.

A tarefa é definida e descrita através de uma sequência de pontos cartesianos que devem ser atingidos pelo efetuador. O processo de otimização aplica transformações de corpo rígido ao conjunto de pontos no espaço cartesiano, resultando em uma sequência ótima de configurações cinemáticas que minimizam o efeito da vibração.

8.1.1 Formulação

Combinando o índice de manipulabilidade com o efeito da vibração e o erro de cinemática inversa através de uma função escalar, a função objetivo pode ser definida como

$$f_4 = \bar{f}_1 + f_2 + f_3 \quad (8.1)$$

onde $\bar{f}_1 = w_{ref} \cdot f_1$ (Equação 2.14) $f_2 = s_N$ (veja Equação (2.23)), e $f_3 = s_E$ (dado pela Equação (2.25)).

O valor de referência w_{ref} é definido de forma que $f_1=1$ no início do processo de otimização. Por este motivo, no caso ideal o valor da função objetivo é menor que 2, onde $f_1 \leq 1$ (houve o aumento da manipulabilidade), $f_2=0$ (não há erro devido à vibração) e $f_3=1$ (não há erro de cinemática inversa).

Deve ser destacado que a formulação proposta considera também o caso no qual a tarefa está fora do espaço de trabalho do manipulador. Esta situação reflete num grande erro de cinemática inversa, que é considerado a partir de uma penalização na função objetivo.

O problema de programação não linear irrestrito (Equação 8.1) é resolvido através do método modificado de direções factíveis (Vanderplaats, 1984). Os algoritmos são implementados em Fortran, através da biblioteca de otimização DOT (Vanderplaats, 1995).

8.1.2 Experimentos numéricos

Um manipulador do tipo Elbow é usado nos experimentos a seguir, com os parâmetros de *Denavit-Hartenberg* dados na Tabela 8.1.

Tabela 8.1. Parâmetros de *Denavit-Hartenberg*; (*) variável de junta.

<i>Elemento</i>	<i>a (m)</i>	<i>α (rad)</i>	<i>d (m)</i>	<i>θ (rad)</i>
1	0	$\pi/2$	0	θ_1^*
2	a_2	0	0	θ_2^*
3	a_3	0	0	θ_3^*

Os valores dos parâmetros do manipulador são: $a_2=1,5\text{ m}$, $a_3=1,5\text{ m}$, e a origem (0, 0, 0) de referência do sistema não está na base geométrica do manipulador (devido à representação de *Denavit-Hartenberg* utilizada). A posição real da origem é $1,0\text{ m}$ acima da base geométrica do manipulador. Para permitir a descrição de percursos complexos, os dados para o conjunto de pontos de especificação cartesianos (Equação (2.20)) são determinados através da interpolação das coordenadas paramétricas do percurso requerido. Assim, todos os valores que definem a trajetória são obtidos a partir de um conjunto apropriado de pontos cartesianos.

Primeiro experimento

Para este experimento é definido um subconjunto contendo 5 pontos de referência: (0; 1,2; -1,1; 0,3), (1; 1,5; -0,5; 0,7), (2; 1,4; 0,0; 0,5), (3; 1,7; 0,3; -0,1) e (4; 1,8; 0,1; -0,7). Comparando os valores com a Equação (2.20), a tarefa inicial está dentro de uma caixa definida por $1,2 \leq x \leq 1,7$, $-1,1 \leq y \leq 0,3$ e $-0,7 \leq z \leq 0,7$. O intervalo de tempo

correspondente é $0 \leq t \leq 4$ s. Os dados são calculados através da interpolação em t , x , y e z , respectivamente. Os dados interpolados consistem em um conjunto de $N = 27$ pontos (Equação (2.20)) que descrevem o a tarefa a ser desempenhada.

No posicionamento original, os valores das variáveis de projeto são: $\beta = 0$ rad, $\delta_x = 0$ m, $\delta_y = 0$ m e $\delta_z = 0$ m. Estes valores são usados como estimativa inicial para o processo de otimização aplicado à Equação (8.1).

A configuração ótima obtida, apresentada na Tabela 8.2 (para $\varepsilon = 0,002$ m na Equação (2.24)), mostra que o efeito da vibração no manipulador (Equação (2.23)) foi reduzido com sucesso. O posicionamento ótimo da tarefa está dentro do espaço de trabalho, pois o índice de erro da cinemática inversa é $f_3 = s_E = 1$ (Equação (2.25)).

Tabela 8.2 -Valor dos parâmetros iniciais e ótimos.

<i>Parâmetro</i>	<i>Valor inicial</i>	<i>Valor ótimo</i>
Manipulabilidade mínima w	2,708	4,483
Desvio padrão s_N	1,524E-02	1,288E-02
Erro da cinemática inversa s_E	1,000	1,000
Índice de performance f_4	2,006	1,608
Variáveis de projeto (β , δ_x , δ_y , δ_z)	(0, 0, 0, 0)	(-0,907; 0,754; -0,279; -0,164)

A especificação inicial da tarefa, conforme descrito acima, é apresentada na Figura 8.1. As Figuras 8.2 e 8.3 ilustram geometricamente o posicionamento ótimo da tarefa e o movimento do robô, respectivamente.

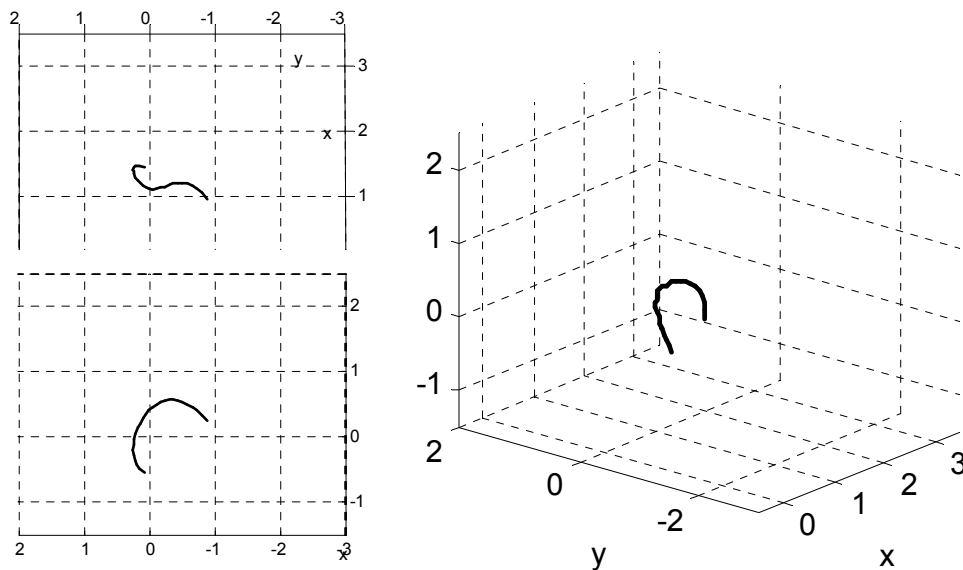


Figura 8.1 - Diferentes perspectivas do posicionamento inicial da tarefa.

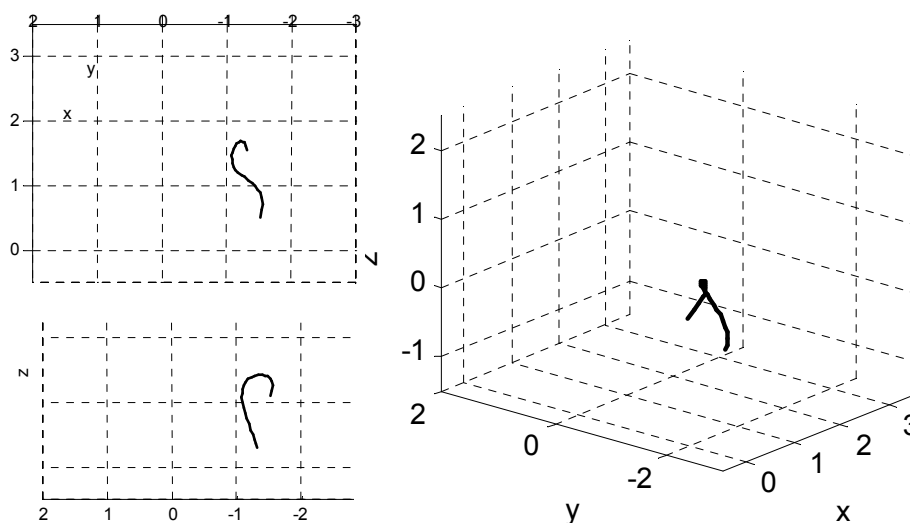


Figura 8.2 - Diferentes perspectivas do posicionamento ótimo da tarefa.

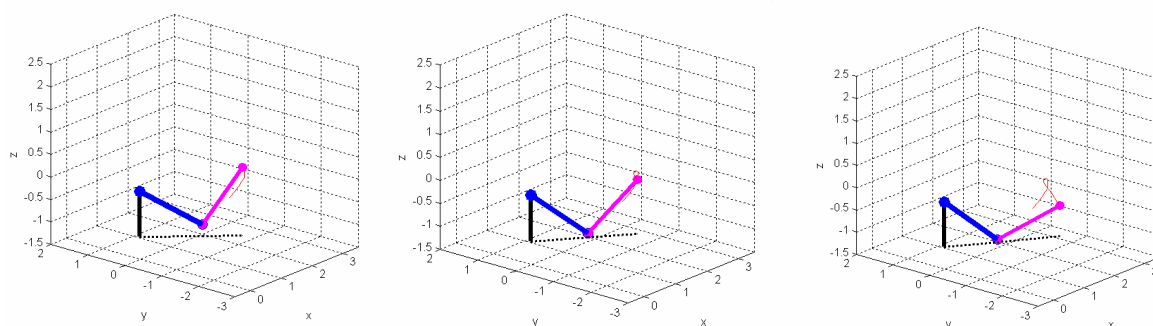


Figura 8.3 - Movimento ótimo do robô.

Outra variável de interesse, a amplitude de oscilação, foi reduzida de $6,337\text{E-}03 \text{ rad}$ para $3,715\text{E-}03 \text{ rad}$. Como este valor está relacionado ao inverso da manipulabilidade (Equação (2.19)), pode-se concluir que quanto maior a manipulabilidade, menor a amplitude da vibração.

Na Figura 8.4 são apresentadas as coordenadas de junta e o respectivo erro angular de cada junta durante o movimento. Estes erros estão associados à vibração do sistema, conforme descrito pela Equação (2.18).

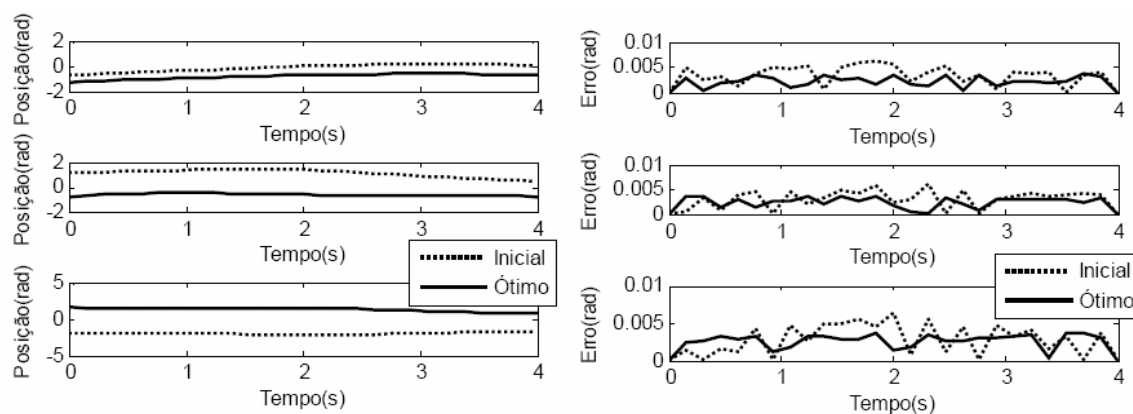


Figura 8.4 - Coordenadas de junta e respectivos erros.

Segundo experimento

Para este experimento é especificado um subconjunto de 15 pontos de referência. Comparando estes valores com a Equação (2.20), a tarefa tem o posicionamento inicial definido dentro do retângulo dado por $x = 1$, $-1,0 \leq y \leq 0,4$ e $0,5 \leq z \leq 1,6$. O tempo correspondente é $0 \leq t \leq 14$ s. Esta especificação simula uma tarefa a ser executada em um plano vertical (em uma parede, por exemplo). A partir destes dados, realiza-se a interpolação para t , y e z , respectivamente. Os pontos interpolados consistem em um conjunto de $N = 54$ pontos, conforme definido pela Equação (2.20), e descrevem o percurso a ser realizado pelo efetuador.

A Figura 8.5 traz a representação geométrica do posicionamento inicial da tarefa. As variáveis de projeto têm os valores $\beta = 0$ rad, $\delta_x = 0$ m, $\delta_y = 0$ m e $\delta_z = 0$ m na configuração inicial, e representam a primeira estimativa para o processo de otimização da Equação (8.1).

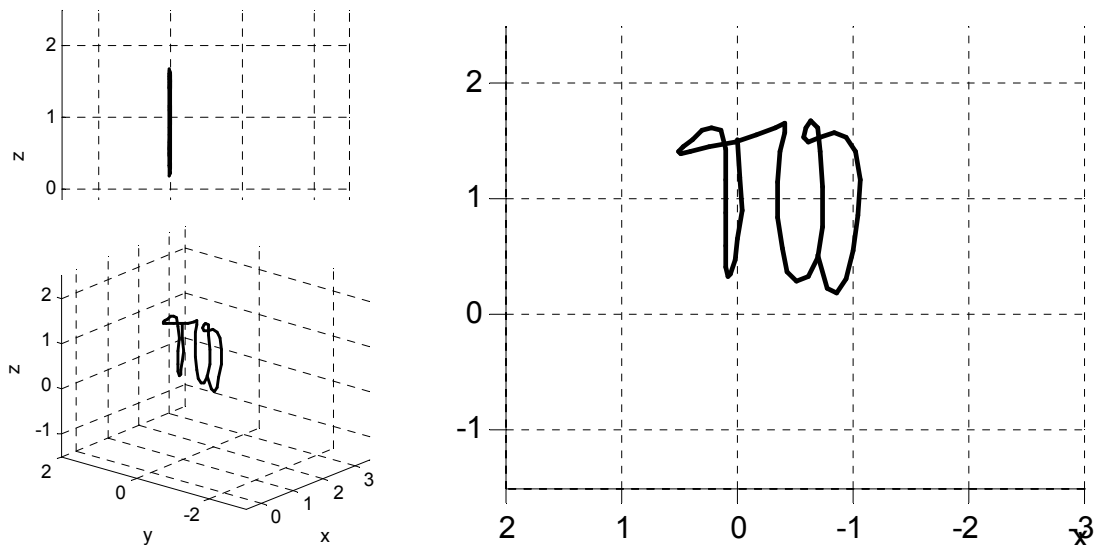


Figura 8.5 - Diferentes perspectivas do posicionamento inicial da tarefa.

O resultado da configuração ótima (para $\varepsilon=0,002$ m) é apresentado na Tabela 8.3, confirmando o sucesso na redução do efeito de vibração em cada elemento (*link*). O posicionamento ótimo da tarefa se localiza dentro do espaço de trabalho, uma vez que o índice de erro de cinemática inversa é $f_3 = 1$ (Equação (2.25)). As Figuras 8.6 e 8.7 apresentam a configuração ótima obtida.

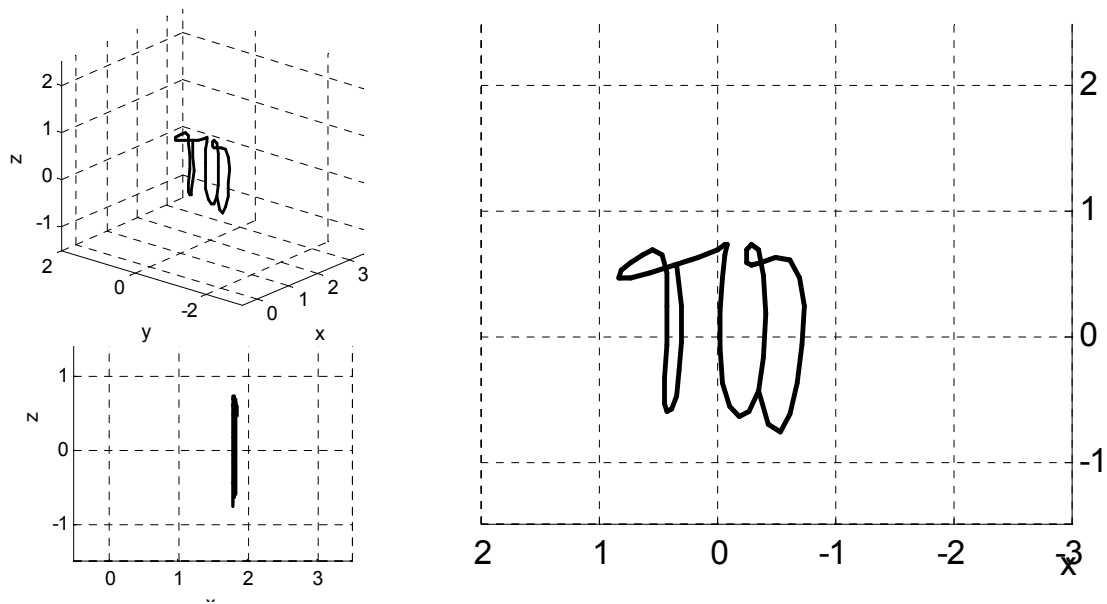


Figura 8.6 - Diferentes perspectivas do posicionamento ótimo da tarefa.

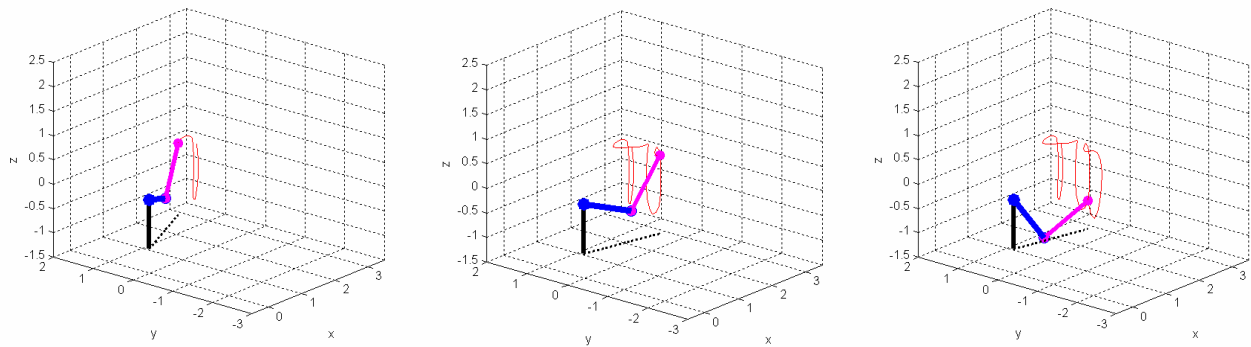


Figura 8.7 - Movimento ótimo do robô.

Tabela 8.3 - Valor dos parâmetros iniciais e ótimos.

<i>Parâmetro</i>	<i>Valor inicial</i>	<i>Valor ótimo</i>
Manipulabilidade mínima w	1,140	2,464
Desvio padrão s_N	1,049E-02	8,010E-03
Erro da cinemática inversa s_E	1,000	1,000
Índice de performance f_4	2,005	1,466
Variáveis de projeto ($\beta, \delta_x, \delta_y, \delta_z$)	(0, 0, 0, 0)	(-4,443E-02; 0,824; 0,378; -0,925)

A amplitude de oscilação foi diminuída de 5,726E-03 *rad* para 2,946E-03 *rad*. A Figura 8.8 apresenta o valor de cada coordenada de junta e o erro no ângulo de cada junta durante a operação do robô.

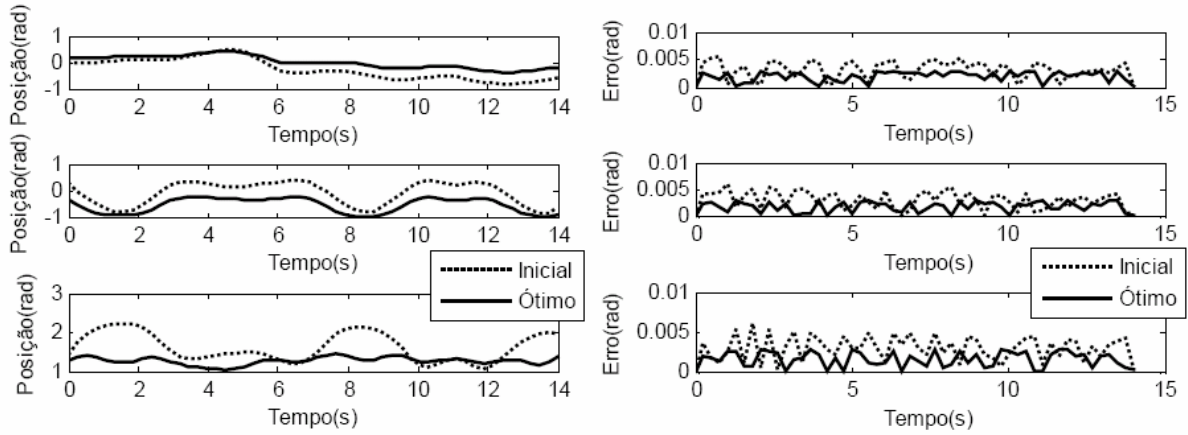


Figura 8.8 - Coordenadas de junta e os respectivos erros.

Terceiro experimento

Para este experimento é especificado um subconjunto contendo 30 pontos de referência. O espaço de trabalho da especificação inicial da tarefa está definido dentro do retângulo $0,3 \leq x \leq 1,7$, $-1,6 \leq y \leq 0,4$ e $z = 0$. O tempo correspondente varia entre $0 \leq t \leq 29$ s. Esta especificação simula uma tarefa posicionada no plano horizontal (no solo, por exemplo). A partir destes dados, realiza-se uma interpolação do tipo *spline* cúbica para cada variável t , x e y , respectivamente. Os dados interpolados constituem um conjunto de $N = 81$ pontos (Equação (2.20)) que descreve o problema de seguimento de trajetória.

A Figura 8.9 apresenta geometricamente a especificação inicial da tarefa, na qual as variáveis de projeto assumem os valores nulos $\beta = 0$ rad., $\delta_x = 0$ m, $\delta_y = 0$ m e $\delta_z = 0$ m. Estes valores são usados como estimativa inicial para o processo de otimização (Equação (8.1)). As Figuras 8.10 e 8.11 apresentam a configuração ótima e o correspondente movimento do robô.

A configuração ótima (para $\epsilon = 0,005$ m), conforme apresentado na Tabela 8.4, confirma que a redução da vibração nos elementos do robô (*links*) foi bem sucedida. O valor inicial do erro de cinemática inversa, $f_3 = 2,006$, significa que o valor inicial das coordenadas de junta não estava apto a satisfazer a precisão requerida para o efetuador (ϵ na Equação (2.24)). Esta exigência foi satisfeita após o processo de otimização, resultando no índice de erro de cinemática inversa, $f_3 = 1$. Isto significa também que o posicionamento ótimo da tarefa está dentro do espaço de trabalho do manipulador.

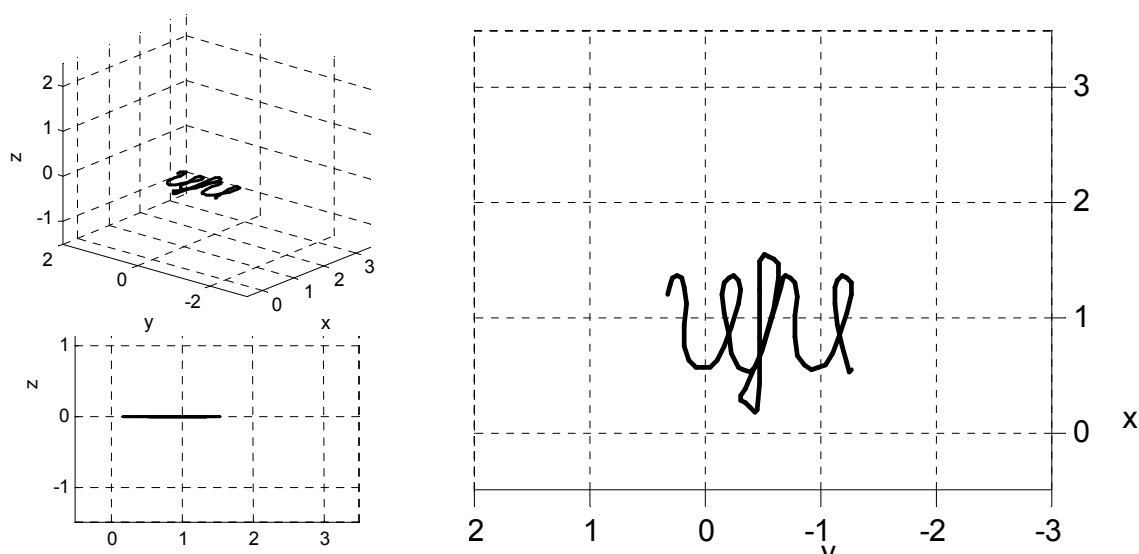


Figura 8.9 - Diferentes perspectivas do posicionamento inicial da tarefa.

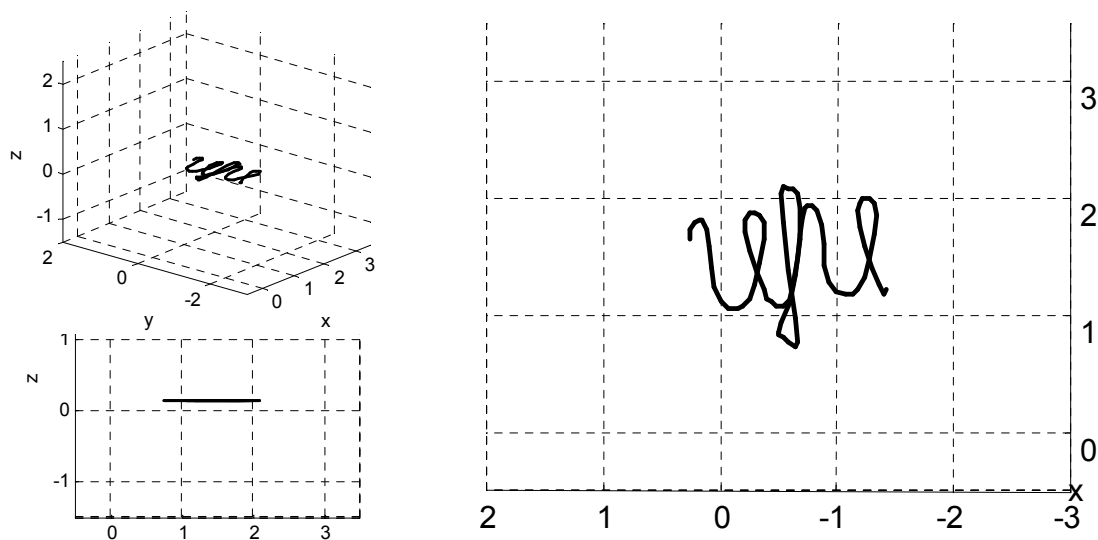


Figura 8.10. - Diferentes perspectivas do posicionamento ótimo da tarefa.

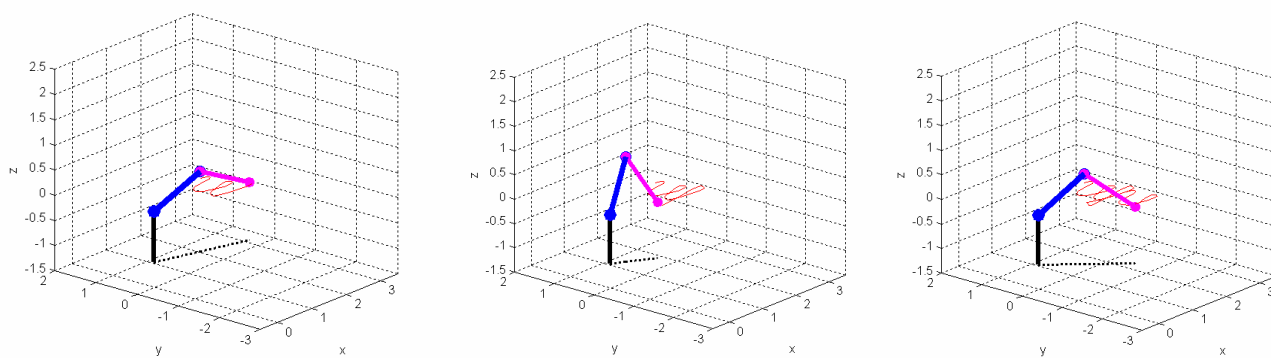


Figura 8.11 - Movimento ótimo do robô.

Tabela 8.4 - Valor dos parâmetros iniciais e ótimos.

<i>Parâmetro</i>	<i>Valor inicial</i>	<i>Valor ótimo</i>
Manipulabilidade mínima w	0,437	2,033
Desvio padrão s_N	5,924E-03	3,482E-03
Erro da cinemática inversa s_E	2,006	1,000
Índice de performance f_4	3,009	1,215
Variáveis de projeto ($\beta, \delta_x, \delta_y, \delta_z$)	(0, 0, 0, 0)	(0,126; 0,626; -0,270; 0,181)

A amplitude de oscilação foi reduzida de 5,218E-03 *rad* para 1,366E-03 *rad*. A Figura 8.12 apresenta as coordenadas de junta durante a operação do manipulador e o correspondente erro em cada ângulo de junta.

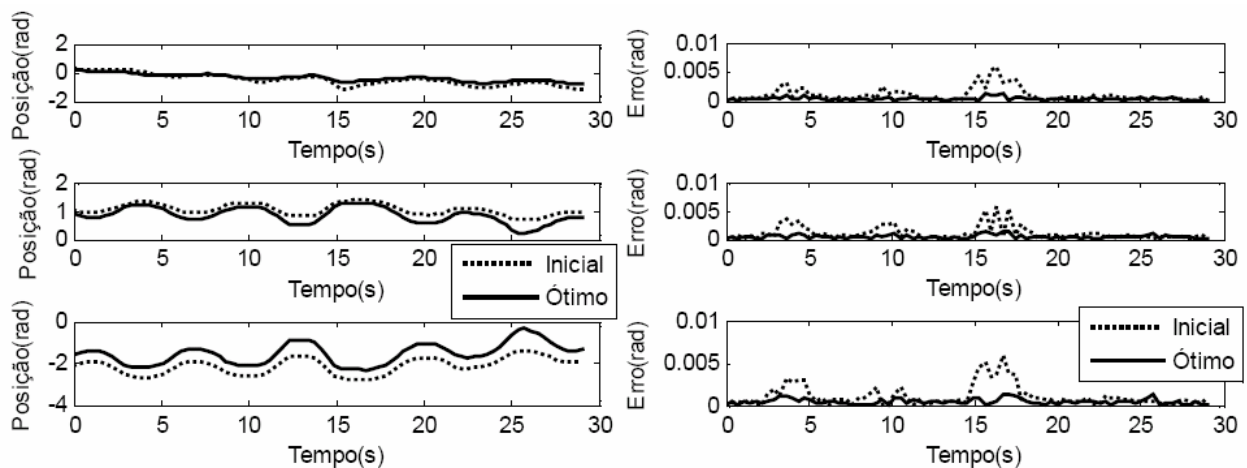


Figura 8.12 - Coordenadas de junta e os respectivos erros.

8.1.3 Conclusão

Este trabalho considerou o problema da redução da vibração de robôs manipuladores. Foi proposta uma alternativa para reduzir os efeitos de vibração em cada elemento (*link*) de um manipulador através do uso de técnicas de otimização.

Como a vibração no efetuador pode acontecer devido a diversos fatores (especificação das leis de controle, flexibilidade das juntas, fricção das engrenagens, etc), foi proposto um modelo geral para descrever os efeitos resultantes de tais vibrações. A equação proposta considerou cada ângulo de junta independentemente, possibilitando o ajuste de múltiplos

fatores para descrever o comportamento distinto de cada articulação. Além disso, foram incluídas informações sobre a velocidade e manipulabilidade de cada junta.

Utilizou-se uma metodologia para redução do erro de movimentação do efetuador que consiste no posicionamento ótimo da tarefa a ser realizada. Usualmente o manipulador serial pode apresentar diferença de desempenho em diferentes regiões do espaço de trabalho.

O processo de otimização procurou adequar esta habilidade à trajetória especificada. Como o processo não considera explicitamente a fonte de distúrbio, pode-se fazer uso desta estratégia para o tratamento de diferentes fontes de perturbação.

Os experimentos numéricos permitiram constatar que o reposicionamento da tarefa é bem sucedido na redução dos efeitos da vibração. Uma vez que alterações no posicionamento da tarefa podem resultar em diferentes índices de manipulabilidade e diferentes perfis de velocidade, a metodologia procura pela melhor forma de utilização da estrutura robótica.

Deve-se destacar que o segmento de trajetórias pelo efetuador é uma tarefa altamente restritiva. Em situações nas quais as especificações físicas do dispositivo robótico não podem ser alteradas, esta estratégia representa uma ferramenta bastante útil.

Devido à abrangência da metodologia proposta para um manipulador serial de complexidade arbitrária, juntamente com a generalidade da fonte de vibração considerada, o autor acredita que esta metodologia representa uma contribuição significativa para o planejamento de trajetória de robôs.

8.2 Planejamento de trajetória de robôs móveis

8.2.1. Introdução

Dentre todas as áreas de pesquisa envolvidas no projeto de um robô móvel, aquela que trata do planejamento de sua trajetória tem recebido grande atenção por parte dos pesquisadores, pois seu desenvolvimento está diretamente relacionado com a maior autonomia dos robôs. A complexidade do problema de planejamento do movimento tem motivado o desenvolvimento dos mais diversos algoritmos. O progresso da pesquisa nesta área tem permitido o uso de robôs móveis em diversas situações, tais como aplicações domésticas, automotivas e de exploração, dentre outras.

Neste trabalho propõe-se uma estratégia para planejamento da trajetória de robôs autônomos a partir de conceitos de programação não linear, estabelecendo uma ferramenta para a determinação da direção durante a realização do movimento, caracterizando-se um planejamento *on-line*.

O estudo da navegação autônoma pode ser realizado em um ambiente estruturado ou não estruturado, de forma que o objetivo é o estabelecimento de uma trajetória livre de colisão.

O planejamento de trajetórias pode ser ainda estático ou dinâmico, dependendo da forma pela qual a informação sobre o obstáculo está disponível. Em um problema estático, toda a informação sobre o obstáculo é conhecida *a priori*, e o movimento do robô é estabelecido a partir desta informação.

No planejamento dinâmico, somente uma informação parcial sobre os obstáculos está disponível, por exemplo, a parte visível dos obstáculos, a partir de uma dada perspectiva.

Para alcançar o alvo estabelecido, o robô planeja a trajetória baseado na informação disponível. Conforme o robô segue seu percurso, descobre mais informações sobre os obstáculos, atualizando sua representação interna do ambiente, o que possibilita um novo planejamento do percurso a partir da representação atualizada obtida.

Neste estudo, considera-se o planejamento dinâmico em um ambiente não estruturado, de forma que o robô não tem informação sobre todo o espaço de trabalho e a trajetória a ser seguida é estabelecida em função da informação obtida pelo sensor sobre a presença de obstáculos em seu percurso.

Quando a dimensão do ambiente ou a posição do obstáculo é variável, define-se um problema variante no tempo. Caso contrário, trata-se de um problema invariante no tempo. O planejamento de trajetórias de qualquer sistema físico é restrito, pois os atuadores (motores) têm potência limitada.

Em geral, os métodos desenvolvidos para o planejamento de trajetórias são adequados para o uso em robôs com arquiteturas distintas, tais como robôs manipuladores seriais, robôs autônomos e robôs de estrutura paralela. Estes métodos geralmente são variações de algumas abordagens gerais: esqueleto, decomposição em células, campo potencial e programação matemática. A maior parte dos problemas de planejamento de percurso pode ser resolvida utilizando estes enfoques (Hwang and Ahuja, 1992). Estas abordagens não são mutuamente exclusivas, e a combinação delas é freqüentemente utilizada no desenvolvimento de uma estratégia para o planejamento de trajetórias.

Na abordagem do esqueleto, o conjunto de movimentos admissíveis é reduzido a uma rede com linhas unidimensionais. Esta abordagem é também chamada de retração. A busca pela solução é limitada à rede, caracterizando um problema de busca em grafo.

Estratégias pertencentes a este conjunto são o grafo de visibilidade e o diagrama de Voronoi (Aurenhammer, 1991). A utilização deste conceito no planejamento de trajetórias pode ser encontrada em Canny e Donald (1987), Canny (1987) e Canny (1988).

As estratégias baseadas na decomposição em células estabelecem a subdivisão do espaço de projeto em um conjunto de células simples, calculando a relação adjacente entre cada célula. Um percurso livre de colisão entre a configuração inicial e final do robô é determinado pela identificação das células iniciais e finais, estabelecendo-se a conexão entre elas através de uma sequência de células conectadas (Keil e Sack, 1985).

A estratégia baseada em programação matemática representa o requisito de desvio do obstáculo através de um conjunto de desigualdades nos parâmetros de configuração. O planejamento do movimento é formulado como um problema de programação matemática que encontra a curva entre as configurações inicial e final minimizando certa grandeza escalar. Como esta otimização é não linear e contém diversas restrições de desigualdade, utiliza-se um método numérico para a determinação da solução ótima (Saramago e Steffen, 1998; Santos *et al.*, 2005b e Santos *et al.*, 2006).

O campo potencial, como estratégia para o desvio de obstáculos, foi aplicado primeiramente por Khatib e Mampey (1978), sendo utilizado posteriormente por Hogan (1985) para o controle de forças.

Este enfoque consiste na construção de uma função potencial, a qual tem valor mínimo quando o robô está na configuração final especificada, e um valor elevado na proximidade dos obstáculos. Em todos os demais lugares, a função tem valor decrescente na direção da configuração objetivo. Assim, o robô pode atingir a configuração objetivo a partir de uma outra configuração, seguindo a direção oposta à do gradiente do potencial. O valor elevado da função potencial na proximidade do obstáculo evita que o robô se mova naquela direção.

A seguir, apresenta-se um modelo para a representação do espaço de trabalho e a estratégia proposta para o planejamento da trajetória.

8.2.2. Representação do espaço de trabalho

Propõe-se a determinação da trajetória a ser seguida pelo robô com uma sequência de passos análogos aos seguidos pelo método do Gradiente. O resultado descrito pelo Teorema 3.1 (Seção 3.2) tem como hipótese o fato de a matriz Hessiana da função quadrática ser definida positiva. Para atender este requisito, propõe-se o uso de uma função quadrática da forma

$$f_{21}(x) = \frac{1}{2} x^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x \quad (8.2)$$

onde $x = [a \ b]^T$ representa a coordenada cartesiana do robô. Com o objetivo de ajustar a concavidade da superfície descrita por esta equação, pode-se multiplicar o valor desta função por uma constante apropriada.

A superfície descrita pelo gráfico desta equação e o respectivo conjunto de curvas de nível são apresentados na Figura 8.13.

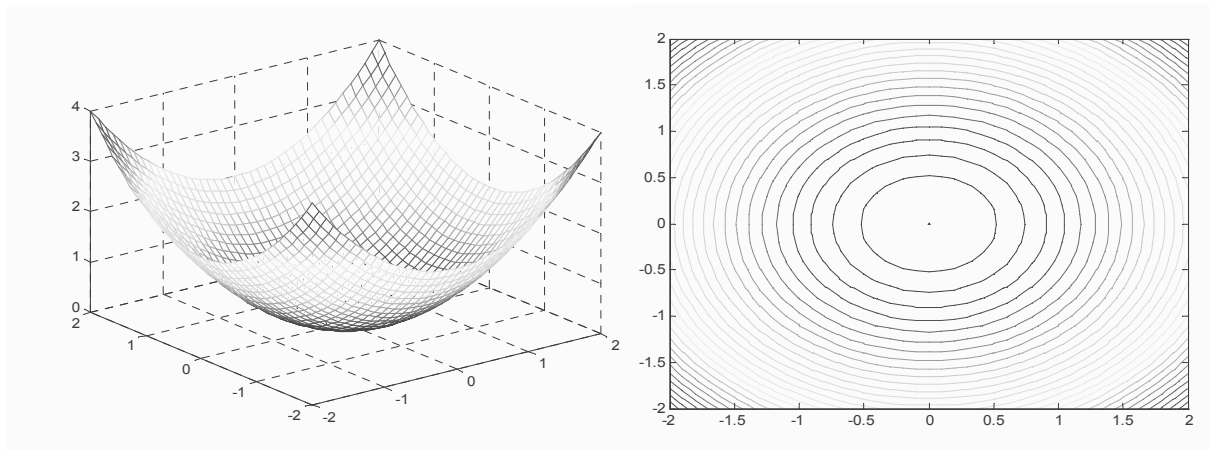


Figura 8.13 - Gráfico da Equação (8.2) e curvas de nível correspondentes.

Ao se posicionar o objetivo no ponto correspondente ao valor mínimo da Equação (8.2), tem-se a modelagem do espaço de trabalho de forma a possibilitar a determinação da trajetória pela direção oposta à do gradiente.

Caso não haja obstáculos no espaço de trabalho, o Teorema 3.1 garante a obtenção do valor mínimo e, conseqüentemente, o estabelecimento da trajetória.

Considera-se agora que a presença de um obstáculo no espaço de trabalho será representada através de uma função potencial que terá seu valor acrescido ao valor da função objetivo.

Propõe-se o uso de uma função Gaussiana para a representação do obstáculo, através da expressão

$$f_{22}(x) = \sum_{j=1}^N \sum_{i=1}^2 e^{\frac{-(x_i - c_i)^2}{2\sigma_i^2}} \quad (8.3)$$

sendo que $j=1, \dots, N$ define o número de obstáculos, $x = [a \ b]^T$ representa a coordenada cartesiana do robô, $c = (c_1, c_2)$ representa a abscissa e a ordenada da posição cartesiana do obstáculo, respectivamente, e σ_i é um parâmetro de suavidade da curva.

A nova superfície, que contém tanto o objetivo a ser atingido como o obstáculo a ser evitado, é representada como a soma das Equações (8.2) e (8.3), resultando no seguinte modelo para a descrição do espaço de trabalho:

$$f_{23}(x) = f_{21}(x) + k f_{22}(x) \quad (8.4)$$

onde k é um fator de ponderação. A Figura 8.14 apresenta o gráfico deste novo modelo e as respectivas curvas de nível.

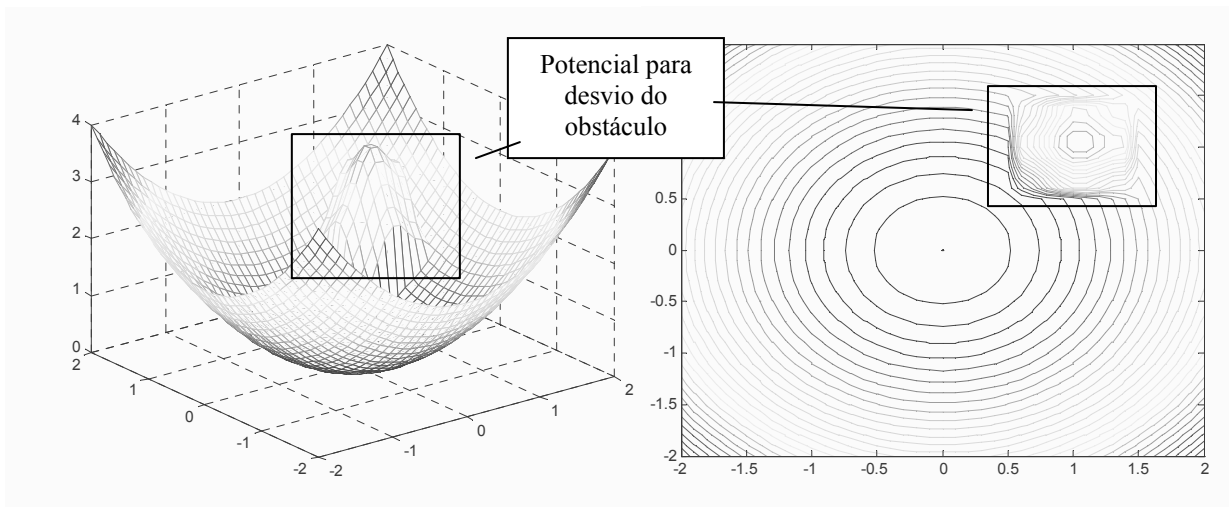


Figura 8.14 - Gráfico da Equação (8.4) e curvas de nível correspondentes.

8.2.3. Formulação da trajetória

Após a definição de uma representação para o espaço livre e a capacidade de identificar configurações viáveis, o planejamento do movimento se reduz a encontrar uma sequência conectada de configurações factíveis entre a posição inicial e a final a partir da representação estabelecida.

O modelo descrito pela Equação (8.4) considera a localização do alvo e a presença do obstáculo, simultaneamente, em um único objetivo. Observa-se que não há necessidade da inclusão explícita do obstáculo como uma restrição na formulação. Por este motivo, a Equação (8.4) representa a função objetivo e a restrição $S = \{x_{\min} < x < x_{\max}\}$ consiste dos valores máximo e mínimo admissíveis para o posicionamento, conforme a formulação geral estabelecida pela Equação (3.4). Pode-se interpretar geometricamente o conjunto S como a especificação das dimensões do espaço de trabalho, estando o alvo posicionado na origem $(0,0)$ do sistema de referência.

Adicionalmente, pretende-se considerar a situação na qual o robô não tem a informação completa sobre o espaço de trabalho ao iniciar o percurso, caracterizando um ambiente dinâmico.

Para tanto, define-se um fator de proximidade p que representa o alcance do sensor de proximidade do robô. Caso a distância entre o obstáculo e o robô seja maior que p , define-se $k = 0$ na Equação (8.4), significando que o robô não tem conhecimento da presença do obstáculo em seu percurso. Caso a distância até o obstáculo seja menor que p , atribui-se um valor positivo a α , de forma a considerar a presença do obstáculo durante o planejamento da trajetória.

Finalmente, calcula-se a nova posição do robô a partir de informações sobre o gradiente da função objetivo no ponto atual, através da expressão:

$$x_{k+1} = x_k - \nabla f_3(x) \cdot v \quad (8.5)$$

onde x_k é a posição cartesiana atual, ∇f_3 é o gradiente da função objetivo e v é a velocidade. Ao se comparar esta proposta com o algoritmo dado pelas Equações (3.5) a (3.7), observa-se que a velocidade v (fixada) corresponde ao parâmetro λ_k da Equação (3.7). Pode-se alternativamente estabelecer um critério para a atualização da velocidade v .

Segundo o Teorema 3.1, considerando-se a ausência de obstáculos, o algoritmo converge para o ponto de mínimo da formulação proposta, atingindo o alvo estabelecido. Mas, conforme poderá ser verificado no exemplo numérico proposto na próxima seção, esta metodologia é viável mesmo na presença de obstáculos. Deve-se ressaltar que a posição do robô e do obstáculo são pontuais, onde não se considera seu volume.

Em resumo, propõe-se a metodologia apresentada na Figura 8.15.

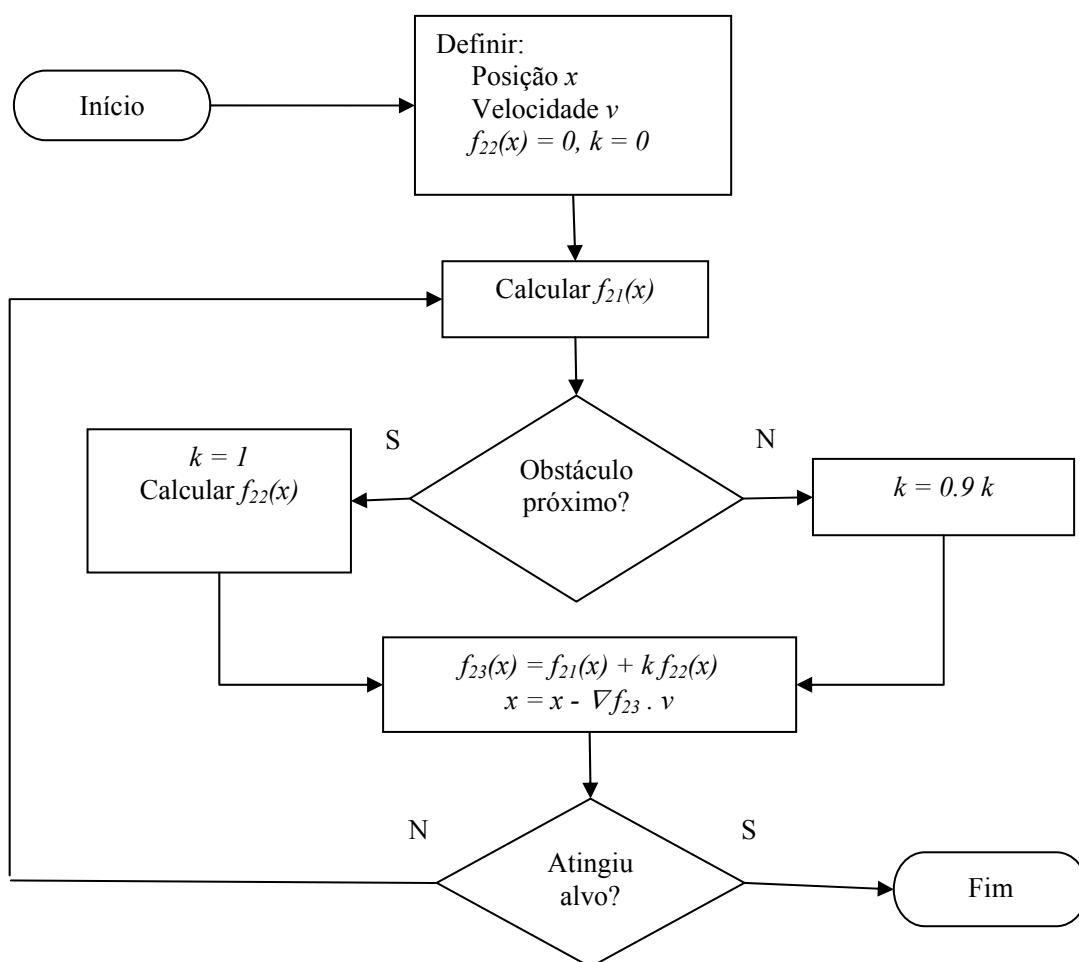


Figura 8.15 - Fluxograma da metodologia proposta.

8.2.4. Experimentos numéricos

Primeiro caso: obstáculo fixo

Com o intuito de avaliar a viabilidade da estratégia proposta, considere o seguinte exemplo numérico: o robô deve alcançar o alvo posicionado na origem $(0,0)$ do sistema cartesiano, iniciando seu movimento a partir da posição $x_0 = (2, -2)$. Há um obstáculo na posição $c = (1, -1,5)$ e adota-se o parâmetro $\sigma = 0,2$ (Equação (8.3)).

A Figura 8.16 apresenta a posição do robô no espaço de trabalho em diferentes instantes de tempo.

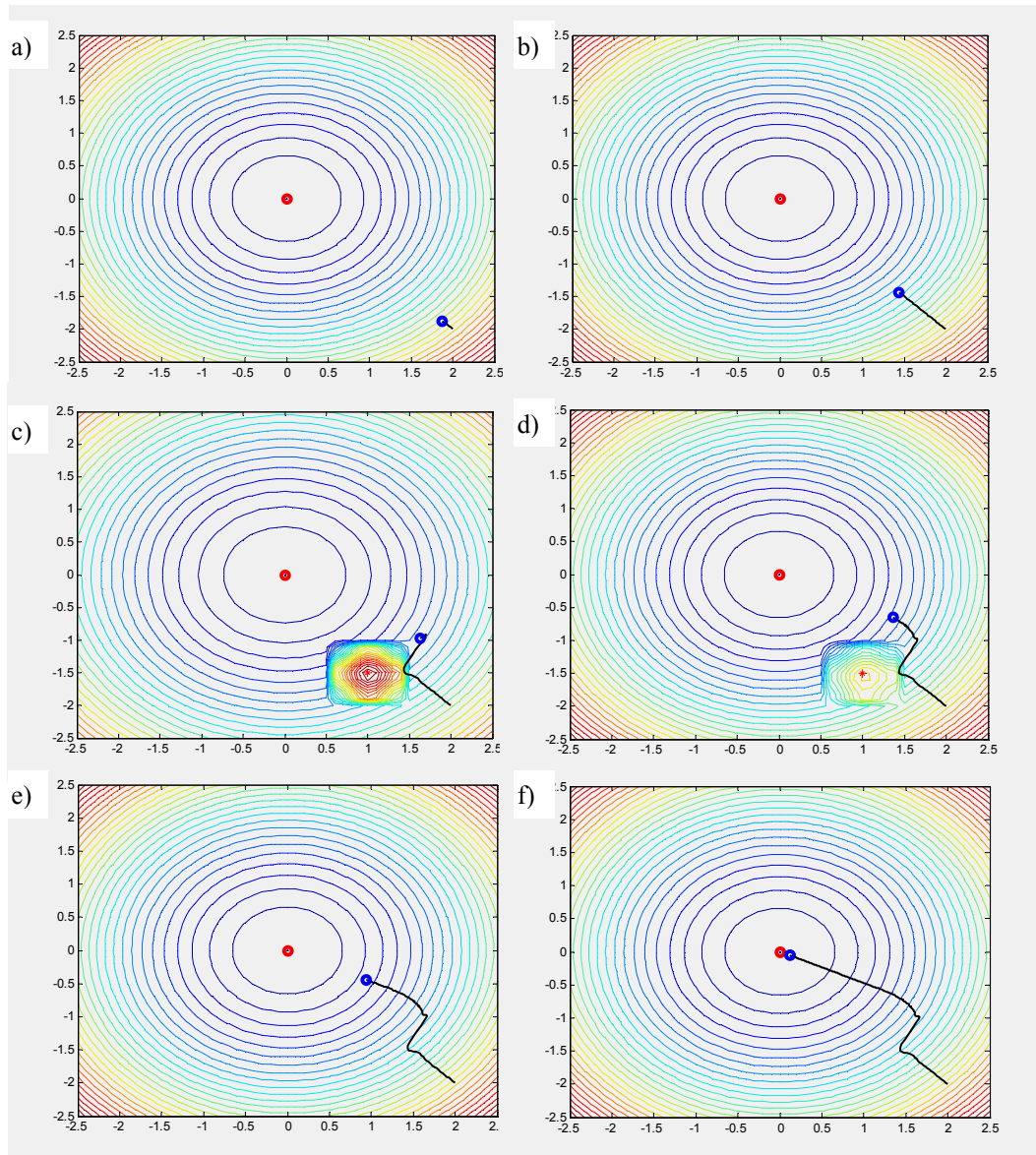


Figura 8.16 - Trajetória do robô na busca do alvo.

Logo após o início do movimento, nas proximidades da posição cartesiana $(2,-2)$, o sensor do robô não detecta a presença de um obstáculo. Assim, ele segue em linha reta na direção do alvo, conforme as Figuras 8.16a e 8.16b. Ao se movimentar pelo espaço de trabalho, o robô se aproxima do obstáculo.

Quando o sensor detecta a presença do obstáculo, a representação da curva de nível é então atualizada. Devido ao valor elevado da função na vizinhança do obstáculo, o algoritmo encontra uma rota de fuga, conforme apresentado na Figura 8.16c.

Após uma manobra evasiva, o método executa algumas iterações bem sucedidas quanto ao desvio do obstáculo, conforme apresentado na Figura 8.16d. As Figuras 8.16e e 8.16f apresentam a trajetória restabelecida na direção do alvo, já distante da área de influência do obstáculo.

Deve-se destacar que somente as Figuras 8.16c e 8.16d consideram a presença do obstáculo, devido ao fator de ponderação k , dado pela Equação (8.4), ter seu valor ajustado durante o processo (Figura 8.15).

Segundo caso: obstáculo móvel – movimento perpendicular

Neste experimento o robô parte da posição $(2,5, -2,0)$ em direção ao alvo $(0, 0)$. O obstáculo se movimenta da posição inicial $(0, -2,5)$ até a posição final $(2,5, 2,0)$. Esta situação é representada na Figura 8.17. Ao atingir a posição final, o obstáculo retorna à posição inicial, sendo este trajeto seguido repetitivamente. O movimento resultante é apresentado na Figura 8.18.

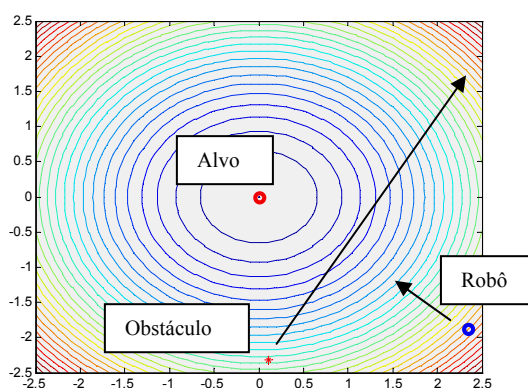


Figura 8.17 - Movimento proposto para o robô e para o obstáculo.

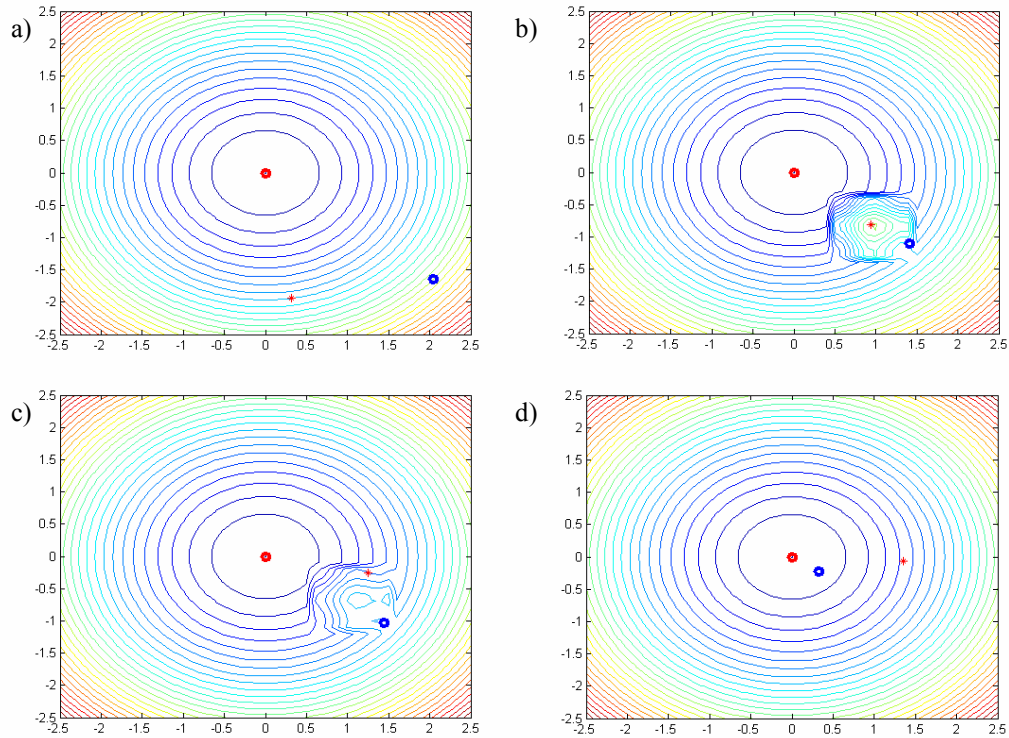


Figura 8.18 - Trajetória do robô na busca do alvo.

Terceiro caso: obstáculo móvel – movimento em direções próximas

Neste experimento o robô parte da posição $(2,5, -2,0)$ em direção ao alvo $(0, 0)$. O obstáculo se movimenta da posição inicial $(2,5, -1,0)$ até a posição final $(-1,0, -0,5)$. Esta situação é representada na Figura 8.19. O movimento resultante desta especificação é apresentado na Figura 8.20.

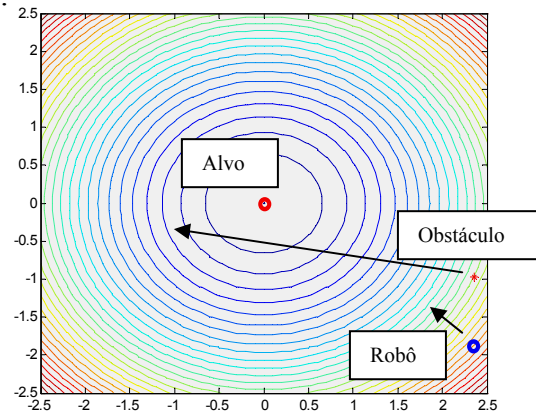


Figura 8.19 - Movimento proposto para o robô e para o obstáculo.

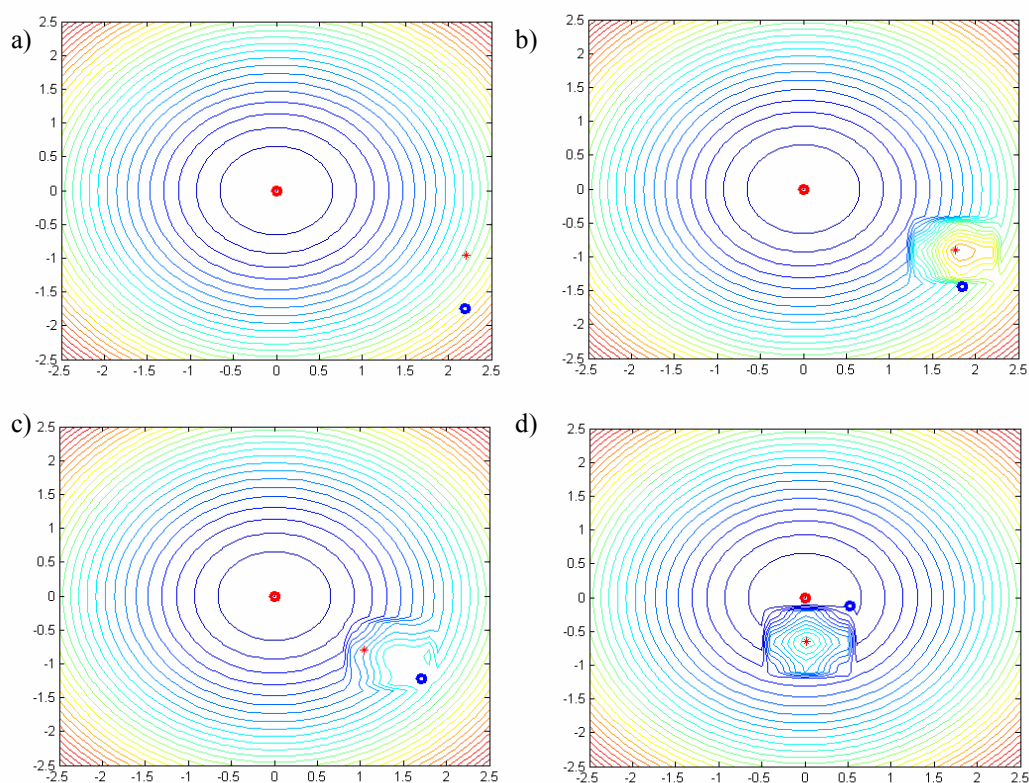


Figura 8.20 - Trajetória do robô na busca do alvo.

Quarto caso: obstáculo móvel – movimento na mesma direção (e sentido oposto)

Neste último experimento, o robô parte da posição $(-2,5, 2,0)$ em direção ao alvo $(0, 0)$. O obstáculo se movimenta da posição inicial $(-0,5, 0,5)$ até a posição final $(-2,5, 2,5)$. Esta situação é representada na Figura 8.21. O movimento resultante desta especificação é apresentado na Figura 8.22.

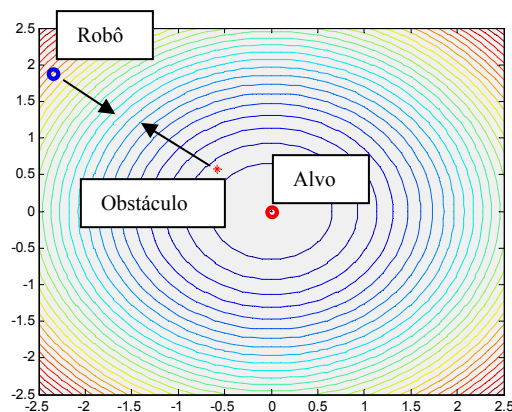


Figura 8.21 - Movimento proposto para o robô e para o obstáculo.

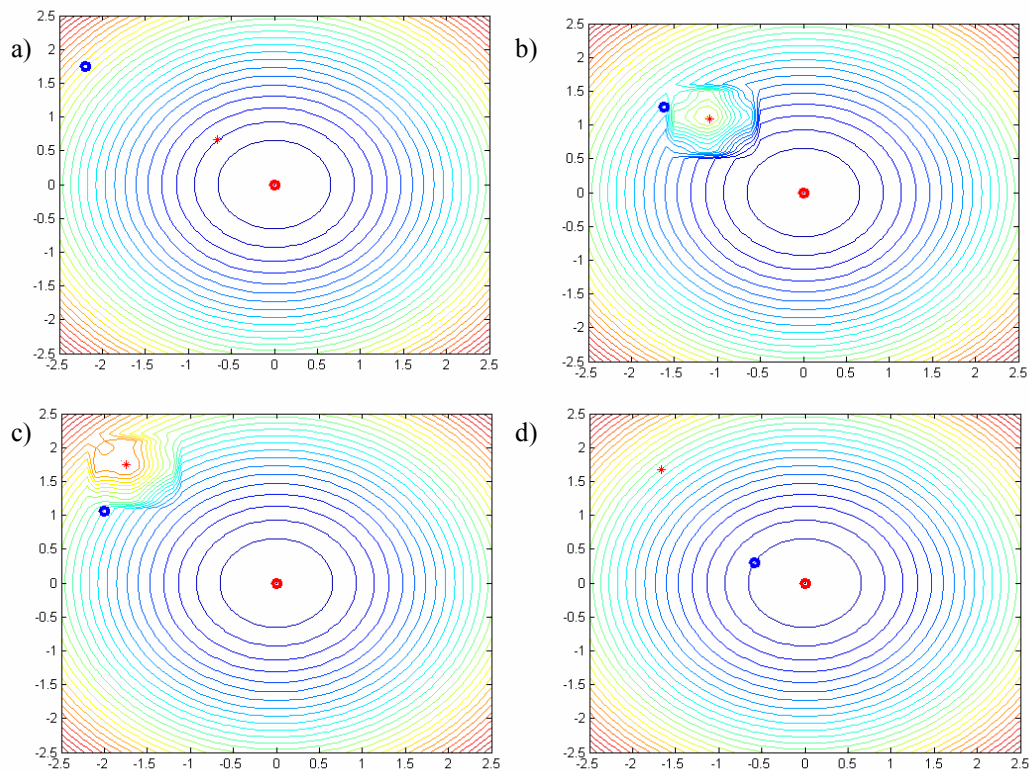


Figura 8.22 - Trajetória do robô na busca do alvo.

8.2.5 Conclusão

Este estudo apresentou uma proposta para a determinação da trajetória de um robô móvel a partir do estabelecimento de campos potenciais, considerando a presença de obstáculos fixos ou móveis no espaço de trabalho.

Foi proposto um modelo de representação do alvo no espaço de trabalho, de forma que a sequência de passos gerada pelo método seja convergente para a solução do problema.

A inclusão de um termo na função objetivo referente à presença do obstáculo no espaço de trabalho mostrou ser uma alternativa viável para a descrição do problema, mesmo não tendo satisfeito a hipótese de convexidade da função objetivo.

O exemplo numérico ilustrou a aplicabilidade da metodologia, apresentando uma trajetória a ser seguida pelo robô, estabelecida pela análise do gradiente da função objetivo. Observou-se a determinação bem sucedida de uma trajetória evasiva quanto à presença do obstáculo e convergente em direção ao alvo.

Cabe destacar que a análise dinâmica do ambiente resulta em diferentes curvas de nível (e, conseqüentemente, diferentes interpretações do espaço de trabalho) durante o processo. Isto equivale a analisar diferentes funções objetivo durante o processo de otimização.

A atualização da representação do ambiente a cada iteração permite considerar situações em que o obstáculo se encontra fixo no espaço ou se movimenta, pois a direção a ser seguida a cada iteração se dá a partir do gradiente da função objetivo, não sendo necessário, portanto, o uso de informações sobre o tipo de movimentação no ambiente.

A generalidade da estratégia e a velocidade na avaliação das informações motivam o autor a considerar sua utilização em problemas de maior complexidade.

A perspectiva de continuidade deste estudo prevê a modelagem da dinâmica do robô, além do uso da velocidade e aceleração como parâmetros de controle.

CAPÍTULO IX

CONCLUSÕES E PERSPECTIVAS

9.1 O trabalho realizado

Neste trabalho foram apresentadas diferentes estratégias *off-line* para determinação da solução da cinemática inversa e planejamento de trajetória de robôs manipuladores, considerando-se o tempo, juntamente com características cinemáticas e dinâmicas, além da presença de obstáculos fixos e móveis, para comporem índices de performance a serem otimizados.

Apresentou-se uma proposta de reposicionamento de tarefa no espaço de trabalho como forma de minimizar a potência mecânica e as vibrações associadas ao movimento do manipulador, assim como uma estratégia de planejamento *on-line* para a determinação de trajetória de robôs autônomos. Foram propostas formulações de funções objetivo para cada caso, explorando-se a resolução através de técnicas de programação não linear, programação multicritério e controle ótimo.

O primeiro capítulo apresentou uma conceituação genérica de robô manipulador e delimitou o objetivo do trabalho, dentro do contexto de uma tese de doutoramento.

No segundo capítulo foram descritas as equações que definem a cinemática direta, cinemática inversa, a manipulabilidade e a dinâmica dos robôs. Houve ainda a proposição de um modelo para representação da vibração ocorrida em cada elemento (*link*), e a proposição de uma especificação de tarefas no espaço de trabalho que foi capaz de permitir seu posicionamento através de movimentos de corpo rígido, conforme verificado posteriormente na sequência da tese.

O terceiro capítulo resumiu os diferentes conceitos de otimização utilizados durante o trabalho. Após a revisão sobre elementos de programação não linear, houve uma breve descrição do Método do Gradiente, utilizado como ponto de partida para a metodologia proposta no oitavo capítulo. Em seguida, foram apresentados elementos de controle ótimo e de programação multicritério, que possibilitaram a extensão das técnicas de otimização para o

tratamento de problemas com mais de um objetivo, de forma eficiente. Houve ainda a descrição de uma estratégia de otimização global, utilizada através de metodologias descritas no sexto e sétimo capítulos desta tese.

O quarto capítulo apresentou o conceito de trajetória, uma estratégia para representação de obstáculos e do manipulador, e a avaliação da distância entre ambos. Esta abordagem foi utilizada para representação do obstáculo em várias situações, conforme descrito no sexto e sétimo capítulo.

A revisão bibliográfica sobre a especificação e otimização da trajetória de robôs manipuladores foi apresentada no quinto capítulo.

No sexto capítulo foram apresentados os resultados obtidos pela pesquisa ao se considerar aspectos cinemáticos durante a otimização da trajetória de robôs manipuladores. Foi feita a proposição de uma estratégia para determinação da cinemática inversa a partir de uma técnica de otimização local (Santos *et al.*, 2004a). Experimentos numéricos realizados confirmaram a viabilidade da estratégia proposta. O enfoque de obter a solução através de técnicas de otimização viabilizou o estudo de cenários variados, havendo a possibilidade de inclusão de novas restrições e ajuste da prioridade no cumprimento das mesmas. A simplicidade de alteração dos parâmetros relevantes, aliado ao bom desempenho computacional verificado nos exemplos implementados, confirmou o potencial de aplicação desta metodologia. Neste capítulo estendeu-se a pesquisa anterior ao se considerar a manipulabilidade durante a determinação da cinemática inversa, além da consideração da presença de obstáculos. A metodologia propôs a utilização de uma técnica de otimização global para a determinação da solução (Santos *et al.*, 2005a). Este enfoque assegurou a obtenção de uma solução para o problema da cinemática inversa, conforme discutido nos experimentos numéricos, e possibilitou a obtenção de diferentes configurações que atendem ao posicionamento e orientação requeridos. Em seguida, foi proposta a otimização do tempo e da derivada da aceleração associados à trajetória (Santos *et al.*, 2004b). Supondo fixados os pontos cartesianos em relação aos quais o manipulador deve se posicionar, fez-se um estudo sobre como o tempo estabelecido para etapas intermediárias do movimento pode ser ajustado para a redução da máxima derivada da aceleração requerida. Adicionalmente considerou-se o ajuste do tempo total, além dos tempos intermediários, de forma a estabelecer o tempo mínimo necessário no qual o valor da derivada da aceleração não excedesse um valor máximo especificado.

Diferentes propostas de otimização dos índices de desempenho relativos à dinâmica do sistema foram apresentadas no sétimo capítulo. Com relação a ambientes irrestritos, apresentou-se uma proposta de otimização do torque através de técnicas de controle ótimo (Santos *et al.*, 2005c). Utilizou-se a formulação para minimizar o valor máximo do torque durante a execução do movimento entre o ponto inicial e final estabelecidos. O resultado numérico comprovou a eficiência da metodologia proposta para a redução do valor máximo do torque necessário durante o movimento. Em seguida foi apresentado um estudo sobre o torque e potência mecânica acumulados em ambiente irrestrito, onde o objetivo multicritério foi otimizado através da teoria de controle ótimo (Santos *et al.*, 2006a). Deste estudo, concluiu-se que a formulação proposta foi apropriada tanto para a redução da potência mecânica e torque associado ao movimento, quanto para a manutenção dos valores de posição e torque em níveis próximos àqueles obtidos a partir da interpolação. Outro ponto favorável na análise dos resultados foi o fato de que a expressão que inclui a energia inibiu o crescimento dos níveis de torque, principalmente nos momentos em que a velocidade apresentou valores próximos de zero. Interessou-se ainda sobre o posicionamento da tarefa a ser realizada, buscando analisar o melhor local para seu posicionamento no espaço de trabalho, considerando-se os requisitos para a potência mecânica do sistema. A posição do efetuador foi fixada em relação a um referencial e descrita como uma seqüência de pontos Cartesianos. O posicionamento foi ajustado com o objetivo de aumentar a manipulabilidade e diminuir a potência mecânica total requerida para o movimento, tendo sido aplicado um método de otimização global para a determinação da solução. Os experimentos numéricos comprovaram a viabilidade da estratégia na análise de trajetórias com alto grau de complexidade. Considerando em seguida a presença de obstáculos fixos no espaço de trabalho, foi apresentada uma estratégia para a determinação da trajetória que considera os requisitos de potência mecânica, velocidade, tempo de operação e posicionamento do robô durante o movimento (Santos *et al.*, 2005b), sendo a trajetória obtida a partir de uma interpolação do tipo *spline*. A principal característica da formulação proposta como um problema de otimização foi a inclusão de nós intermediários para a interpolação das coordenadas de cada junta, estas consideradas como variáveis de projeto. Em cada aplicação numérica, foi considerado o aumento do número de nós intermediários no espaço de coordenadas de junta. Este procedimento ofereceu maior liberdade para o algoritmo obter trajetórias ótimas, contribuindo para a efetiva minimização da potência mecânica associada. Entretanto, deve-se destacar que não é recomendado o aumento arbitrário do número de nós

intermediários de interpolação porque isto pode levar à descontinuidade do movimento ou à especificação de trajetórias não suaves. Foi ainda feita uma proposta de otimização da potência mecânica do manipulador através do uso de controle ótimo, onde se considera a presença de obstáculos moveis tanto num ambiente bidimensional (Santos *et al.*, 2006b) como tridimensional. Este estudo propôs a determinação da trajetória em duas fases: inicialmente considerou-se o desvio do obstáculo, e então, tanto o obstáculo quanto a potência mecânica foram objetivos simultaneamente considerados na segunda fase do processo de otimização. Cabe salientar que a solução da primeira fase da otimização foi utilizada como estimativa inicial para o início da segunda fase. Esta abordagem possibilitou a obtenção de uma solução que minimizou a potência mecânica requerida, enquanto preservou a continuidade do movimento diante da presença de obstáculos em movimento. Encerrou-se o sétimo capítulo com a apresentação de uma proposta para determinação e otimização de trajetórias de robôs, que se preocupa com a interceptação de objetos em movimento (Santos *et al.*, 2006c). Neste caso, tanto o instante como a posição da interceptação são considerados como variáveis de projeto. A primeira fase da otimização considerou um critério de interceptação do obstáculo. Esta análise levou a um projeto viável. A segunda fase da otimização utilizou este resultado como informação inicial para a determinação de uma trajetória que minimiza a potência mecânica requerida durante o movimento. O uso da potência mecânica como índice de desempenho levou à otimização da velocidade e do torque simultaneamente, tendo sido considerados, portanto, aspectos cinemáticos e dinâmicos do movimento.

No oitavo capítulo foi apresentada uma nova proposta para redução de vibrações em robôs manipuladores. Neste sentido, realizou-se a análise através de um índice de desempenho que deve ser otimizado pelo reposicionamento da tarefa a ser executada no espaço de trabalho (Santos *et al.*, 2006e). Considerou-se aspectos como o aumento da manipulabilidade e a redução do efeito de vibração, além do erro de cinemática inversa, durante o planejamento da trajetória. Os experimentos numéricos confirmaram que o posicionamento adequado da tarefa reduz o efeito de vibração. Como alterações no posicionamento da tarefa resultam em diferentes índices de manipulabilidade e diferentes perfis de velocidade, a metodologia procurou por uma forma ótima de utilização dos recursos passíveis de serem utilizados pelo robô manipulador. Em seguida fez-se a proposição de uma estratégia para a determinação da trajetória de robôs móveis (Santos *et al.*, 2006d). Para tanto, considerou-se a presença de obstáculos fixos e móveis no espaço de trabalho, que foram analisados durante o planejamento *on-line* da trajetória. Os experimentos numéricos

demonstraram a viabilidade da estratégia ao promoverem a obtenção de uma trajetória contínua. O processo de ponderação dos diferentes objetivos (alvo e obstáculos) durante o movimento ofereceu robustez ao método no sentido de permitir a consideração de cenários variados, além da possibilidade de inclusão de uma grande quantidade de obstáculos a serem evitados ao longo do movimento do sistema. Esta proposta procurou generalizar o conceito de planejamento de trajetória, confirmando a aplicabilidade dos métodos de otimização para tal tarefa.

Destaca-se que, conforme comentado anteriormente na Introdução desta tese, o resultado obtido através do algoritmo de cálculo da dinâmica do robô manipulador (*Recursive Newton Euler – RNE*), utilizado nas várias propostas, foi comparado com aquele resultante do software NASTRAN, tendo sido observada a concordância entre os resultados. De certa forma, pode-se então dizer que houve preocupação do autor no sentido de validar o programa desenvolvido.

9.2 Principais contribuições

Em resumo, esta tese reuniu diversas contribuições relativas ao planejamento de trajetória de robôs. Em relação à originalidade das propostas, destacam-se as seguintes abordagens:

a) A estratégia de otimização global utilizada para a determinação da cinemática inversa juntamente com a análise da manipulabilidade (seção 6.2). Este estudo considera a utilização de uma estratégia de otimização determinística global para a obtenção da solução do problema de cinemática inversa. No caso em que há infinitas soluções para o problema (o manipulador é cinematicamente redundante), faz-se a opção por aquela configuração que maximiza a manipulabilidade. Este enfoque difere daqueles encontrados na literatura, por considerar uma estratégia global determinística ao mesmo tempo em que se preocupa com a manipulabilidade.

b) A metodologia de avaliação do torque e potência mecânica através de uma função multiobjetivo e a influência de diferentes fatores de ponderação (seção 7.2). Realiza-se a determinação da trajetória através de técnicas de controle ótimo, onde o vetor de controle contém a velocidade, e o vetor de estado contém variáveis de estado que correspondem à posição e ao torque associado. A formulação utilizada da teoria de controle ótimo, aliada a

uma metodologia para determinação do melhor fator de ponderação para os diferentes objetivos, caracterizam a originalidade da proposta.

c) O posicionamento da tarefa a ser realizada como estratégia para aumento de desempenho (apresentado na seção 2.5 e utilizado nas seções 7.3 e 8.1). O posicionamento da tarefa a ser realizada através de sua movimentação por transformações de corpo rígido, juntamente com a análise de vibração (seção 8.1) e a minimização da potência mecânica (seção 7.3) são igualmente aspectos que apontam para a originalidade do trabalho.

d) A forma de utilização de nós intermediários para determinação da trajetória ótima (seção 7.4). O estudo do problema através de uma formulação de programação não linear, onde se considera o desvio de obstáculos, a potência mecânica e o tempo total como índices de desempenho, e a utilização dos nós intermediários da interpolação como variáveis de projeto, representa também uma contribuição significativa.

e) A utilização de técnicas de controle ótimo para especificação da trajetória que otimiza a potência mecânica e permite o desvio de obstáculos em movimento (seção 7.5). A proposição de uma estratégia composta de duas fases para a determinação da trajetória ótima através de uma formulação de controle ótimo, onde é obtida a redução da potência mecânica e o efetivo desvio do obstáculo por todos os elementos constitutivos do manipulador, tem seu caráter de originalidade dado pela escolha dos vetores de estado e controle da formulação. Adicionalmente, a metodologia foi eficiente para prover estabilidade numérica e baixo custo computacional para a obtenção do resultado ótimo, uma vez que estas características não são facilmente obtidas no contexto da formulação do controle ótimo.

f) A otimização da potência mecânica ao interceptar um objeto em movimento (seção 7.6). Segundo a literatura consultada pelo autor, não há até o momento estudos que considerem a redução da potência mecânica ao planejar uma trajetória de interceptação de obstáculos em movimento.

g) A proposição de uma nova formulação para análise de vibrações (apresentado na seção 2.4 e utilizado na seção 8.1). A utilização da teoria clássica de rotores para descrever a vibração dos elementos (*links*) e do efetuador do robô, bem como sua otimização através do posicionamento da tarefa são aspectos não encontrados na literatura consultada.

h) A determinação da trajetória de robôs móveis a partir do conceito de direção de máxima descida juntamente com a ponderação de um campo potencial para a descrição de obstáculos em movimento (seção 8.2). Segundo a literatura consultada pelo autor, pode-se considerar como sendo nova a proposição da estratégia que determina a direção a ser seguida

pelo robô a partir de uma analogia com o Método do Gradiente, para obter o planejamento *on-line* da trajetória.

Assim, o autor acredita que esta tese reúne um conjunto de estratégias úteis e eficientes, aplicáveis a diversas situações concretas que envolvem o projeto e a utilização dos robôs.

Destaca-se que todas as propostas citadas anteriormente podem ser aplicadas a manipuladores de configuração arbitrária, ou seja, hipóteses restritivas sobre características específicas de um dado manipulador não foram utilizadas durante o desenvolvimento das propostas e sua aplicação. Como consequência, durante a realização dos diversos experimentos numéricos, diversas configurações cinemáticas foram consideradas, evidenciando o caráter de generalidade da metodologia desenvolvida.

Finalmente, pode-se considerar esta pesquisa como sendo uma fase de transição com vistas à análise de situações mais complexas, tais como aquela que envolve o planejamento de trajetórias onde a flexibilidade das estruturas robóticas é incluída no modelo, aspecto este que já vem merecendo a atenção da comunidade científica.

9.3 Trabalhos futuros

A satisfação de requisitos conflitantes, como a alta velocidade de operação e alta precisão de posicionamento têm criado novos desafios para o projeto, controle e operação de robôs, aumentando a demanda de estudos dentro deste tema de pesquisa.

Muitos manipuladores industriais apresentam o problema de vibração em seus elementos (*links*), principalmente durante o movimento em alta velocidade. Para aumentar a produtividade industrial, é recomendada a redução do peso dos elementos constitutivos do sistema, o que permite o aumento de sua velocidade de operação. Esta abordagem sugere a consideração de manipuladores robóticos cada vez mais flexíveis.

Comparados aos manipuladores convencionais, os manipuladores com elementos flexíveis têm a vantagem potencial de menor custo, maior volume de trabalho, maior velocidade operacional, maior razão entre o peso do objeto a ser manipulado e o peso do robô (maior carga útil), atuadores de menor porte, menor consumo de energia e operação mais segura devido à redução da inércia do conjunto. Entretanto, uma grande desvantagem deste tipo de manipulador é o problema da vibração do sistema devido à baixa rigidez.

Por este motivo, uma possibilidade natural de extensão dos resultados aqui apresentados consiste na determinação da trajetória ótima de robôs manipuladores, considerando-se o efeito de flexibilidade e vibração dos elementos constitutivos durante o movimento. Neste contexto, são consideradas informações sobre a vibração do mecanismo, além dos aspectos cinemáticos e dinâmicos do manipulador.

Por outro lado, o enfoque numérico dado até o momento para a resolução dos diferentes problemas motiva a realização de um estudo teórico mais aprofundado, com o objetivo de estabelecer condições necessárias e suficientes para a determinação da trajetória mínima global, bem como a discussão sobre aspectos da estabilidade numérica da solução obtida. Destaca-se que este estudo tem um caráter restritivo, pois irá estabelecer condições necessárias sobre a metodologia (que em geral consiste na inclusão de restrições sobre o problema proposto) de forma a assegurar tais propriedades.

A busca por metodologias de otimização mais eficientes do ponto de vista da velocidade de execução dos códigos computacionais é uma perspectiva igualmente importante, especialmente ao se pensar na incorporação de sistemas *on-line* de otimização da trajetória, desvio de obstáculos e controle de vibrações.

Como consequência do aumento do desempenho, pode-se avaliar as metodologias no sentido de propor sua reformulação para o planejamento *on-line* da trajetória, enriquecendo o universo de estratégias destinadas a este fim, procurando analisar criteriosamente diferentes índices de desempenho.

Finalmente, diante dos resultados obtidos em cada análise, acredita-se que a aplicação das metodologias a problemas industriais seja perfeitamente viável. Este enfoque consiste unicamente no ajuste dos diferentes fatores de ponderação e na descrição apropriada do problema de interesse, para o efetivo aproveitamento da teoria de otimização no contexto da robótica industrial.

Referências bibliográficas

- ALVES, J. B. M. Controle de Robô, Ed. Cartgraf, 1988.
- ARMSTRONG, B.; KHATIB, O.; BURDICK, J. The explicit dynamic model and inertial parameters of the Puma 560 arm, Proceedings of the IEEE Conference on Robotics and Automation, San Francisco, p. 510-518, 1986.
- AURENHAMMER, F. Voronoi diagrams – a survey of a fundamental geometric data structure, ACM Computing Surveys, v. 23, n. 3, p. 345 – 405, 1991.
- BALKAN, T.; ÖZGÖREN, M. K.; ARIKAN, M. A. S.; BAYKURT, H. M. A method of inverse kinematics solution including singular and multiple configurations for a class of robotic manipulators, Mechanism and Machine Theory 35, p. 1221-1237, 2000.
- BARRAQUAND, J.; LATOMBE, J. C. A Monte-Carlo algorithm for path planning with many degrees of freedom. In Proceedings of the IEEE International Conference on Robotics and Automation (Cincinnati May 13-18) IEEE. New York, p. 1712-1717, 1990.
- BENOSMAN, M.; LE VEY, G.; LANARI, L.; DE LUCA, A. Rest-to-Rest Motion for Planar Multi-Link Flexible Manipulator Through Backward Recursion, Journal of Dynamic Systems, Measurement, and Control, v. 126, p. 115-123, 2004.
- BOBROW, J.E.; DUBOWSKY, S.; GIBSON, J. S. Time Optimal Control of Robotic Manipulators Along Specified Paths. International Journal of Robotics Research. v. 4, n. 3, p. 3-17, 1985.
- BOISSIERE, P. T., AND HARRIGAN, R. W. Telerobotic operation of conventional robot manipulators. In Proceedings of the IEEE International Conference on Robotics and Automation (Philadelphia, Apr. 24-29). IEEE, New York, p. 576-583, 1988.
- BOYCHUK, L. M.; OVCHINNIKOV, V. O. Principal Methods of Solution Multicriterial Optimization Problems. Soviet Automatic Control. v. 6, p. 1-4, 1973.
- BROOKS, R. A. 1983. Solving the Findpath problem by good representation of free space. IEEE Trans. Syst., Man, and Cybernetics SMC-13, v. 3 (Mar. /Apr.), p. 190-197.
- BRYSON, A. E. Dynamic optimization. Addison-Wesley Longman, Inc., Reading, MA, 1999.
- CANNY, J. F. The Complexity of Robot Motion Planning, The MIT Press, Cambridge, 1988.

- CANNY, J. F. A new algebraic method for robot motion planning and real geometry, In: Proceedings of the 28th Annual Symposium on Foundations of Computer Science (Los Angeles, Oct 12-14). IEEE, Washington, D. C., p. 39-48, 1987.
- CANNY, J. F.; DONALD, B., Simplified Voronoi diagram, In: Proceedings of the 3rd Annual ACM Symposium on Computational Geometry (Waterloo, Ontario, June 8-10) ACM Press, New York, p. 153-161, 1987.
- CAO, B.; DODDS, G. I.; IRWIN, G. W. Time-Optimal and Smooth Constrained Path Planning for Robot Manipulators. In IEEE International Conference on Robotics and Automation, San Diego, California, p. 1853-1858, 1994.
- CHAPELLE, F.; BIDAUD, P. Closed form solutions for inverse kinematics approximation of general 6R manipulators, Mechanism and Machine Theory 39, p. 323–338, 2004.
- CHEN, P. C.; HWANG, Y. K. 1992. SANDROS: A motion planner with performance proportional to task difficulty. In Proceedings of the IEEE International Conference on Robotics and Automation (Nice, France, May 12-14). IEEE, New York, p. 2346-2353.
- CHEN, Y.; DESROCHERS, A. A. Structure of Minimum-Time Control Law for Robotic Manipulators with Constrained Paths. In: IEEE International Conference on Robotics and Automation. Scottsdale, Arizona. p. 971-976, 1989.
- CHEN, Y.; HUANG, J.; WEN, T. Y. A Continuation Method for Time-Optimal Control Synthesis for Robotic Point-To-Point Motion. In 32nd Conference on Decision and Control, San Antonio, Texas, December, p. 1628-1632, 1993.
- CHEUNG, E.; LUMELSKY, V. Proximity sensing in robot manipulator motion planning: System and implementation issues. IEEE Trans. Robot. Automat. v. 5, p. 740-751, 1989.
- CITRON, S.J. Elements of Optimal Control. Holt, Rinehart and Winston, Inc., New York, 1969.
- CONSTANTINESCU, D.; CROFT, E.A. Smooth and Time-Optimal Trajectory Planning for Industrial Manipulators along Specified Paths. Journal of Robotic Systems. v. 17, n. 5, p. 233-249, 2000.
- CRAIG, J. J. Introduction to robotics: Mechanics & control. Second Edition. Reading, MA: Addison-Wesley, 1989.
- DADFARNIA, M.; JALILI, N.; XIAN, B.; DAWSON, D. M. A Lyapunov-Based Piezoelectric Controller for Flexible Cartesian Robot Manipulators, Journal of Dynamic Systems, Measurement, and Control, v. 126, p. 347-358, 2004.

- DEB, K. Multi-Objective Optimization using Evolutionary Algorithms. New York: John Wiley, 2001.
- DENAVIT, J.; HARTENBERG, R. S. A kinematic notation for lower-pair mechanisms based on matrices, ASME Journal of Applied Mechanisms, p. 215-221, 1955.
- ESCHENAUER, H. ; KOSKI, J.; OSYCZKA, A. Multicriteria Design Optimization. Springer-Verlag, 1990.
- FAVERJON, B. Obstacle avoidance using an octree in the configuration space of a manipulator. In: Proc. IEEE Intern. Conf. Robot. Automat. Atlanta, p. 504-512, 1984.
- FAVERJON, B.; TOURNASSOUD, P. 1987. A local approach for path planning of manipulators with a high number of degrees of freedom. In Proceedings of the IEEE International Conference on Robotics and Automation (Raleigh, N. Carol., Mar. 31-Apr. 3). IEEE, New York, p. 1152-1159.
- FU, K. S.; GONZALES, R. C.; LEE, C. S. G. Robotics: control, sensing, vision and intelligence. McGraw-Hill, 1987.
- GAN, J.Q.; OYAMA, E.; ROSALES, E.M.; GAN, H.H. A complete analytical solution to the inverse kinematics of the Pioneer 2 robotic arm. Robotica. Cambridge University Press, v. 23, p. 123-129, 2005.
- GEERING, H. P.; GUZZELLA, L.; HEPNER, S.A.R.; ONDER, C.H. Time-Optimal Motions of Robots in Assembly Tasks. IEEE Transactions on Automatic Control, AC- v. 31 n. 6, p. 512-518, 1986.
- GENTA, G. Dynamics of Rotating Systems, Springer, 2005.
- GUPTA, K. C.; ROTH, B. Design Considerations for Manipulator Workspace, ASME J. Mechanical Design, 104(4), p. 704-711, 1982.
- HEILIG, J.; McPHEE, J., Determination of Minimum-Time Manoeuvres for a Robotic Manipulator using Numerical Optimization Methods, Mechanics of Structures and Machines , vol.27, no.2, pp.185-201, 1999.
- HEINZINGER, G.; JACOBS, P.; CANNY, J.; PADEN, B. Time-Optimal Trajectories for a Robot Manipulator: A Provably Good Approximation Algorithm. In IEEE International Conference on Robotics and Automation, Cincinnati, Ohio, May, p. 150-156, 1990.
- HIMMELBLAU, D.M. Applied Nonlinear Programming. McGraw-Hill, 1989.
- HOGAN, N. Impedance control: An approach to manipulation: Part III – Application, ASME J. Dynamic Syst. Meas. Control. 107 (Mar.), p. 17-24, 1985.

- HWANG, Y.K.; AHUJA, N. Gross motion planning - A survey. ACM Computing Surveys (CSUR) archive. ACM Press (ISSN:0360-0300). v. 24, n. 3, p. 219-291, 1992.
- KAHN, M.E.; ROTH, B. The Near-Minimum Time Control of Open-Loop Articulated Kinematic Chains. ASME Transactions, Journal of Dynamic Systems, Measurement, and Control, p. 164-172, 1971.
- KHATIB, O. 1985. Real-time obstacle avoidance for manipulators and mobile robots. In Proceedings of the IEEE International Conference on Robotics and Automatton (St. LOUIS, Missouri). IEEE, New York, p. 500-505.
- KHATIB, O. Real-time obstacle avoidance for manipulators and mobile robots. Int. J. Robot. Res. p. 90-98, 1986.
- KHATIB, O.; LE MAITRE, J.F. Dynamic control of manipulators operating in a complex environment. In: Proc. 3rd Int. CISM-IFTToMM Symp., 1978. p. 267-282.
- KHATIB, O.; MAMPEY, L. M. Fonction decision-commande d'un robot manipulateur, Rep. 2/7156. DERA/CERT, Toulouse, France, 1978.
- KEIL, J. M.; SACK, J. R. Minimum decomposition of polygonal objects, In: Computational Geometry. Elsevier Science Publishers, North Holland, Amsterdam, p. 197-216, 1985.
- KIM, D. H.; SHIN, S. Local path planning using a new artificial potential function composition and its analytical design guidelines, *Advanced Robotics*, v. 20, n. 1, p. 115-135, 2006.
- KODITSCHKEK, D. E. Robot planning and control via potential functions. In Robotics Review, v. 1, O. Khatib, J. Graig, and T. Lozano-Pérez, Eds. MIT Press, Cambridge, Mass, 1989.
- KOREN, Y.; BORENSTEIN, J. Potential field methods and their inherent limitations for mobile robot navigation. In Proc. IEEE Int. Conf. Robot. Automat. Sacramento, CA, 1991, v. 2, p. 1398-1404.
- KYRIAKOPOULOS, K.; SARIDIS, G. Minimum Jerk Path Generation. In IEEE International Conference on Robotics and Automation, p. 364-369, Philadelphia, Pennsylvania, April 1988.
- KYRIAKOPOULOS, K.; SARIDIS, G. Minimum Jerk for Trajectory Planning and Control. *Robotica*, v. 12, p. 109-113, 1994.
- LE GRAND, S. M.; MERZ JR., K. M. The Application of the Genetic Algorithm to the Minimization of Potential Energy Functions. *Journal of Global Optimization*. v. 3, n.1, p. 49 - 66, 1993.

- LEVY, A.V.; MONTALVO, A. The Tunneling Algorithm for the Global Minimization of Functions. *SIAM Journal on Scientific and Statistical Computing*. v. 6, n. 1, p. 15-29, 1985.
- LIN, C.S.; CHANG, P.R.; LUH, J.Y.S. Formulation and Optimization of Cubic Polynomial Joint Trajectories for Industrial Robots. *IEEE Transactions on Automatic Control*, AC-28(12), p. 1066-1074, 1983.
- LOZANO-PÉREZ, T. A simple motion-planning algorithm for general robot manipulators. *IEEE J. Robotics Auto. RA-3*, 3 (June), p. 224-238, 1987.
- LOZANO-PÉREZ, T. Spatial planning: A configuration space approach. *IEEE Trans. Comput.* v. C-32, n. 2, p. 102-120, 1983.
- LOZANO-PÉREZ, T.; WESLEY, M A. An algorithm for planning collision-free paths among polyhedral obstacles. *Commun. ACM* 22, 10 (Oct.), p. 560-570, 1979.
- LUENBERGER, D. G. Linear and non-linear programming. Second Edition. Reading, MA: Addison-Wesley, 1984.
- LUENBERGER, D. G. Optimization by vector space methods, John Wiley and Sons, Inc., New York, 1969.
- MACIEJEWSKI, A. A., AND KLEIN, C. A. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *Int. J. Robotics Res.* 4, 3 (Fall), p. 109-117, 1985.
- MANSEUR, R.; DOTY, K. L. Structural kinematics of 6-Revolute-Axis robot manipulators, *Mechanism and Machine Theory* v. 31 n. 5, p. 647-657, 1996.
- MAO, Z.; HSIA, T. C. Obstacle avoidance inverse kinematics solution of redundant robots by neural networks. *Robotica*. Cambridge University Press, v. 15, p. 3-10, 1997.
- MITSI, S.; BOUZAKIS, K. D.; MANSOUR, G. Optimization of robot links motion in inverse kinematics solution considering collision avoidance and joint limits, *Mech. Mach. Theory* v. 30, n. 5, p. 653-663, 1995.
- NAKAMURA, Y. Advanced Robotics: Redundancy and Optimization. Addison-Wesley, 1991.
- NEARCHOU, A. C. Solving the inverse kinematics problem of redundant robots operating in complex environments via a modified genetic algorithm, *Mech. Mach. Theory* v. 33, n. 3, p. 273-292, 1998.
- NEWMAN, W., AND HOGAN, N. High speed robot control and obstacle avoidance using dynamic potential function. In *Proceedings of the IEEE International Conference on*

- Robotics and Automation (Raleigh, N. Carol., Mar.31-Apr. 3). IEEE, New York, pp. 14-24, 1987.
- NOBLE, B.; DANIEL, J.W. Applied Linear Algebra. Prentice-Hall, 1977.
- OLIVEIRA, L.S; SARAMAGO, S.F.P. Otimização Multi-Critérios Aplicada a um Problema Dinâmico. In: 14º Simpósio do Programa de Pós-graduação em Engenharia Mecânica, 2004, Uberlândia. p. 1-8.
- ÖZGÖREN, M. K. Topological analysis of 6-joint serial manipulators and their inverse kinematic solutions, *Mechanism and Machine Theory* 37, p. 511–547, 2002.
- PAUL, R. P. Robot manipulators: Mathematics, Programming, and Control. Cambridge, MA: MIT Press. , 1981.
- PFEIFFER, F.; JOHANNI, R. A Concept for Manipulator Trajectory Planning. *IEEE Journal of Robotics Automation*. v. 3, n. 2, p. 115-123, 1987.
- RAGHAVAN, B.; ROTH, B. Inverse kinematics of the general 6R manipulator and related linkages, *Transactions of the ASME, Journal of Mechanical Design* 115, p.502-508, 1993.
- RICHARD, M. J.; GOSSELIN, C. M. A survey of simulation programs for the analysis of mechanical systems, *Math. Comput. Simul.*, v. 35, n. 2, p. 103-121, 1993.
- RIMON, E.; KODITSCHKE, D. The construction of analytic diffeomorphisms for exact robot navigation on star worlds. In *Proc. IEEE Intern. Conf. Robot. Automat.* Scottsdale, AZ, 1989, v. 1, p. 21-26.
- ROMEIJN, H. E.; SMITH, R. L. Simulated Annealing for Constrained Global Optimization. *Journal of Global Optimization*. v. 5, n. 2, p. 101-126, 1994.
- SALUKVADZE, M. E. On the Existence of Solution in Problems of Optimization under Vector Valued Criteria. *Journal of Optimization Theory and Applications*. v. 12, n. 2, p. 203-217, 1974.
- SANTOS, R.R.; KRESCHEL, M. ; STEFFEN Jr, V.; SARAMAGO, S. F. P. Serial Robot Vibration Analysis Through Task Positioning Optimization. In: *Polimetrica Int.Sci. Pub. (Org.). Advances in Vibration Control and Diagnostics*. Milano: Polimetrica Int.Sci. Pub, 2006e, v. 1, p. 49-64.
- SANTOS, R.R.; SARAMAGO, S. F. P.; STEFFEN Jr, V. Movimentação de Robôs em Ambientes com Obstáculos. In: *Simpósio do Programa de Pós-Graduação em Engenharia Mecânica*, 2004, Uberlândia - MG, 2004a.

- SANTOS, R.R.; SARAMAGO, S. F. P.; STEFFEN Jr, V. Especificação de Trajetória de Robôs em Tempo Ótimo . In: Simpósio do Programa de Pós-Graduação em Engenharia Mecânica, 2004, Uberlândia, 2004b.
- SANTOS, R.R.; SARAMAGO, S. F. P.; STEFFEN Jr, V. Otimização do Torque Aplicado pelos Atuadores de Robôs Usando Técnicas de Controle Ótimo. In: Simpósio do Programa de Pós-Graduação em Engenharia Mecânica, 2005, Uberlândia - MG, 2005c.
- SANTOS, R.R.; SARAMAGO, S. F. P.; STEFFEN Jr, V. Trajetória ótima de robôs manipuladores ao interceptar objetos em movimento. In: Simpósio do Programa de Pós-Graduação em Engenharia Mecânica, 2006, Uberlândia - MG, 2006c.
- SANTOS, R.R.; SARAMAGO, S. F. P.; STEFFEN Jr, V. Planejamento de trajetória de robô autônomo através do conceito de campo potencial. In: Simpósio do Programa de Pós-Graduação em Engenharia Mecânica, 2006, Uberlândia - MG, 2006d.
- SANTOS, R.R.; STEFFEN Jr, V.; SARAMAGO, S. F. P. Solving the Inverse Kinematics Problem Through Performance Index Optimization. In: XXVI Iberian Latin-American Congress on Computational Methods in Engineering (CILAMCE), 2005, Guarapari - ES, 2005a.
- SANTOS, R.R.; STEFFEN Jr, V.; SARAMAGO, S. F. P. Robot Path Planning: Avoiding Obstacles. In: 18th International Congress of Mechanical Engineering (COBEM), 2005, Ouro Preto - MG, 2005b.
- SANTOS, R. R.; STEFFEN Jr, V.; SARAMAGO, S. F. P. Técnicas de Controle Ótimo Multicritério Aplicadas à Otimização de Aspectos Dinâmicos no Planejamento de Trajetórias de Robôs Manipuladores. In: VII Simpósio de Mecânica Computacional, 2006, Araxá. Anais do VII Simpósio de Mecânica Computacional, 2006a. p. 1-10.
- SANTOS, R.R.; STEFFEN Jr, V.; SARAMAGO, S. F. P. Robot Path Planning in a Constrained Workspace by using Optimal Control Techniques, in: III European Conference on Computational Mechanics Solids, Structures and Coupled Problems in Engineering, Lisboa, 2006b.
- SARAMAGO, S. F. P. Otimização de trajetórias de robôs manipuladores na presença de obstáculos. 1998. Tese de Doutorado - Universidade Federal de Uberlândia, Uberlândia.
- SARAMAGO, S. F. P. ; STEFFEN Jr, V. Trajectory Modeling of Robot Manipulators in the Presence of Obstacles. Journal of Optimization Theory and Applications. USA, v. 110, n. 1, p. 17-34, 2001.

- SARAMAGO, S. F. P. ; STEFFEN Jr, V. Optimal Trajectory Planning of Robot Manipulators in the Presence of Moving Obstacles. *Machine and Mechanism Theory*. Oxford, v. 35, n. 8, p. 1079-1094, 2000.
- SARAMAGO, S.F.P.; STEFFEN Jr, V. Considerações Sobre Funções Custo a serem utilizadas na Otimização de Trajetórias de Robôs Manipuladores, In: V Congresso de Engenharia Mecânica Norte - Nordeste, 1998, Fortaleza, Anais do V CEM-NNE, 1998. v. 1, p. 415-422, 1998.
- SARAMAGO, S. F. P; STEFFEN Jr, V. Optimization of the Trajectory Planning of Robot Manipulators Taking Into-Account The Dynamics of the System. *Mechanism and Machine Theory*. v. 33, n. 7, p. 883-894, 1998.
- SERAJI, H.; BON, B. Real-Time Collision Avoidance for Position-Controlled Manipulators, *IEEE Transactions on Robotics and Automation*. v. 15, n. 4, 1999.
- SERAJI, H.; STEELE, R.; IVLEV, R. Sensor-based collision avoidance: Theory and experiments. *J. Robot. Syst.* v. 13, n. 9, p. 571-586, 1996.
- SHILLER, Z. Time-Energy Optimal Control of Articulated Systems with Geometric Path Constraints. In *IEEE International Conference on Robotics and Automation*, p. 2680-2685, San Diego, California, 1994.
- SHILLER, Z.; DUBOWSKY, S. Global time optimal motions of robotic manipulators in the presence of obstacles. In *Proceedings of the IEEE International Conference on Robotics and Automation (Philadelphia, Apr 24-29)* IEEE, New York, pp. 370-375, 1988.
- SHILLER, Z.; LU, H.H. Computation of Path Constrained Time-Optimal Motions with Dynamic Singularities. *ASME Transactions, Journal of Dynamic Systems, Measurement, and Control*. v. 114, n. 1, p. 34-40, March 1992.
- SHIN, K. G.; MCKAY, N. D. Open-loop minimum-time control of mechanical manipulators and its application. In *Proceedings of the American Control Conference (San Diego, June 6-8)* IEEE, New York, p. 1231-1236, 1984.
- SHIN, K. G.; MCKAY, N. D. Minimum-Time Control of Robotic Manipulators with Geometric Path Constraints. *IEEE Transactions on Automatic Control*, AC-30(6), p. 531-541, June 1985.
- SIMON, D. The Application of Neural Networks to Optimal Robot Trajectory Planning. *Robotics and Autonomous Systems*, v. 11, p. 23-34, 1993.
- SPONG, M. W.; VIDYASAGAR, M. *Robot Dynamics and Control*. John Wiley and Sons, 1989.

- STEFFEN Jr, V.; SARAMAGO, S.F.P. Optimization Techniques for Off - line Trajectories Planning of Robot Manipulators. Non - linear Dynamics, Chaos, Control and their Applications in Engineering Sciences. v. 1, p. 363-368, 1997.
- STEINBACH, M.C.; BOCK, H.G.; LONGMAN, R.W. Time-optimal extension or retraction in polar coordinate robots: a numerical analysis of the switching structure. In: Proceedings of the 1989 AIAA Guidance, Navigation and Control Conference. p. 895-910, 1989.
- STOER, J.; BULIRSCH, R. Introduction to Numerical Analysis, Springer, New York, 1993.
- STORN, R.; PRICE, K. Differential Evolution - a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization. v. 11, p. 341 - 359, 1997.
- VANDERPLAATS, G. N. Numerical Optimization Techniques for Engineering Design. 3rd edition, VRD, Inc., 1999.
- VANDERPLAATS, G. N. An Efficient Feasible Direction Algorithm for Design Synthesis. *AIAA Journal*, v. 22, n. 11, 1984.
- VANDERPLAATS, G. N. DOT - Design Optimization Tools Program, Vanderplaats Research & Development, Inc, Colorado Springs, 1995.
- VON STRYK, O.; BULIRSCH, R. Direct and Indirect Methods for trajectory optimization. In: Annals of Operations Research. v. 37, p. 357-373, 1992.
- VUKOBRATOVIC M.; KIRCANSKI, M. A Method for Optimal Synthesis of Manipulator Robot Trajectories. ASME Transactions, Journal of Dynamic Systems, Measurement, and Control, v. 104, p. 188-193, June 1982.
- WANG, D.; HAMAM, Y. Optimal Trajectory Planning of Manipulators with Collision Detection and Avoidance. The International Journal of Robotics Research, v. 11, n. 5, p. 460-468, October 1992.
- WAPPENHANS, H.; HOLZL, J.; STEINLE, J.; PFEIFFER, F. Optimal Trajectory Planning with Application to Industrial Robots. International Journal of Advanced Manufacturing Technology, v. 9, p. 49-55, 1994.
- WIKMAN, T. S.; NEWMAN, W. S. A fast, on-line collision avoidance method for a kinematically redundant manipulator based on reflex control. CAISR Tech. Rep. 91-160, Case Western Reserve Univ., Cleveland, OH, 1991.
- WU, C. H.; PAUL, R. P. Resolved motion force control of robot manipulator, IEEE Transactions on Systems, Man and Cybernetics SMC v. 12, n. 3, p. 266-275, 1982.

- YOSHIKAWA, T. Foundations of Robotics: Analysis and Control, Cambridge : Mit, 1990.
- ZLAJPAH, L.; NEMEC, B. Force strategies for on-line obstacle avoidance for redundant manipulators. *Robotica*. Cambridge University Press, v. 21, p. 633-644, 2003.
- ZHAO, X.; PENG, S. A successive approximation algorithm for the inverse position analysis of serial manipulators. *Robotica*. Cambridge University Press, v. 17, p. 487-489, 1999.
- ZOLLO, L.; SICILIANO, B.; DE LUCA, A.; GUGLIELMELLI, E.; DARIO, P. Compliance Control for an Anthropomorphic Robot with Elastic Joints: Theory and Experiments, *Journal of Dynamic Systems, Measurement, and Control*, v. 127, p. 321-328, 2005.
- ZOUTENDIJK, G. Methods of feasible directions: a study in linear and non-linear programming. Amsterdam: Elsevier, 1960.

Apêndice A

Descrição cinemática de um manipulador planar no padrão VRML.

```
#VRML V2.0 utf8

WorldInfo {
  title "Planar Manipulator"
}

# LINK SHAPE

PROTO PLink1 [ ] {
  Transform {
    children [
      Transform {
        rotation 1 0 0 1.5708
        children DEF Joint Shape {
          geometry Cylinder { radius 0.15 height 0.25 }
          appearance Appearance { material Material { diffuseColor 0.9
0.5 0.1 } }
        }
      }
      Transform {
        translation 0.7 0 0
        children DEF Link Shape {
          geometry Box { size 1.4 0.2 0.2 }
          appearance Appearance { material Material { diffuseColor 0.1
0.9 0.0 } }
        }
      }
    ]
  }
}

PROTO PLink2 [ ] {
  Transform {
    children [
      Transform {
        rotation 1 0 0 1.5708
        children DEF Joint Shape {
          geometry Cylinder { radius 0.15 height 0.25 }
          appearance Appearance { material Material { diffuseColor 0.9
0.5 0.1 } }
        }
      }
      Transform {
        translation 0.7 0 0
        children DEF Link Shape {
          geometry Box { size 1.4 0.2 0.2 }
          appearance Appearance { material Material { diffuseColor 0.1
0.9 0.0 } }
        }
      }
    ]
  }
}
```

```

# ROBOT KINEMATICS

DEF Robot Transform {
  children DEF Link1 Transform {
    children [
      DEF Link2 Transform {
        translation 1.4 0 0
        children [
          PLink2 {}
        ]
      }
      PLink1 {}
    ]
  }
}

# VIRTUAL WORLD SPECIFICATIONS

NavigationInfo {
  type "EXAMINE"
  avatarSize 1
}

Viewpoint {description "Upper Right" position 4 2 7 orientation 0 1 0 0.7}
Viewpoint {description "Lower Left" position -4 0 7 orientation 0 1 0 -0.7}
Viewpoint {description "Center" position 0 1 5 }

Background {
  skyColor [ 1 1 0 0.3 0.3 0.9 1 1 1 ]
  skyAngle [ 1.1 1.57 ]
  groundAngle [ 1.57 ]
  groundColor [ 0.1 0.1 0.1 1 1 1 0.1 0.1 0.1 ]
}

DEF Workspace Transform {
  children [
    Transform {
      translation -1.2 -0.5 0
      children DEF Floor Shape {
        geometry Box { size 3 0.1 3 }
        appearance Appearance { material Material { transparency 0.5
diffuseColor 0.9 0.9 0.9 } }
      }
    }
    Transform {
      translation 0 -0.25 0
      children DEF Beam Shape {
        geometry Box { size 0.05 0.5 0.05 }
        appearance Appearance { material Material { transparency 0.5
diffuseColor 0.9 0.9 0.9 } }
      }
    }
  ]
}

# MOVEMENT SPECIFICATION

DEF Energy TimeSensor {
  loop TRUE
  enabled TRUE
}

```

```

    cycleInterval 10
}

DEF PowerSource ScalarInterpolator {
    keyValue [0, 1]
    key [0, 1]
}

DEF Joint1 OrientationInterpolator {
    keyValue [0 0 1 0, 0 0 1 0]
    key [0, 1]
    # the lines below are used by the software to update trajectory.
    #Joint1
    #Time
}

DEF Joint2 OrientationInterpolator {
    keyValue [0 0 1 0, 0 0 1 0]
    key [0, 1]
    # the lines below are used by the software to update trajectory.
    #Joint2
    #Time
}

ROUTE Energy.fraction_changed TO PowerSource.set_fraction
ROUTE PowerSource.value_changed TO Joint1.set_fraction
ROUTE PowerSource.value_changed TO Joint2.set_fraction
ROUTE Joint1.value_changed TO Link1.set_rotation
ROUTE Joint2.value_changed TO Link2.set_rotation

```

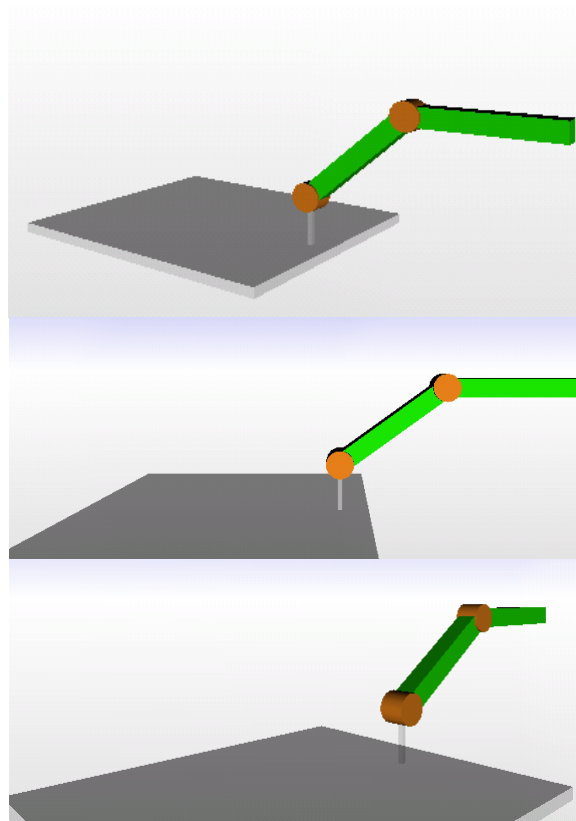


Figura 1. Diferentes perspectivas de visualização.

Apêndice B

SMART PATH-PLANNER

Descrição do software para planejamento ótimo de trajetória de robôs.

Descrição

O sistema determina a trajetória de robôs manipuladores seriais. A partir da especificação da posição inicial e final, é determinada uma especificação ponto a ponto do movimento, com a resolução requerida.

A trajetória inicial é especificada a partir de um polinômio interpolador tipo *spline*. Pode-se optar entre duas estratégias de otimização [1, 2] para a obtenção da trajetória ótima.

A informação gerada pelo sistema consiste na especificação sobre tempo, posição, velocidade, aceleração e torque de cada junta. O relatório é gerado de acordo com o padrão *HyperText Markup Language (HTML)*.

A trajetória ótima pode ser visualizada através de uma representação de realidade virtual no formato *Virtual Reality Modeling Language (VRML)*. Este arquivo pode ser atualizado pelo usuário para representar diferentes ambientes virtuais.

O sistema executável é composto do arquivo principal (*spp.exe*) que contém a interface gráfica (Figura 1), o arquivo que contém os métodos de otimização (*simulation.exe*) e o arquivo para visualização em ambiente em realidade virtual (*planar.wrl*).

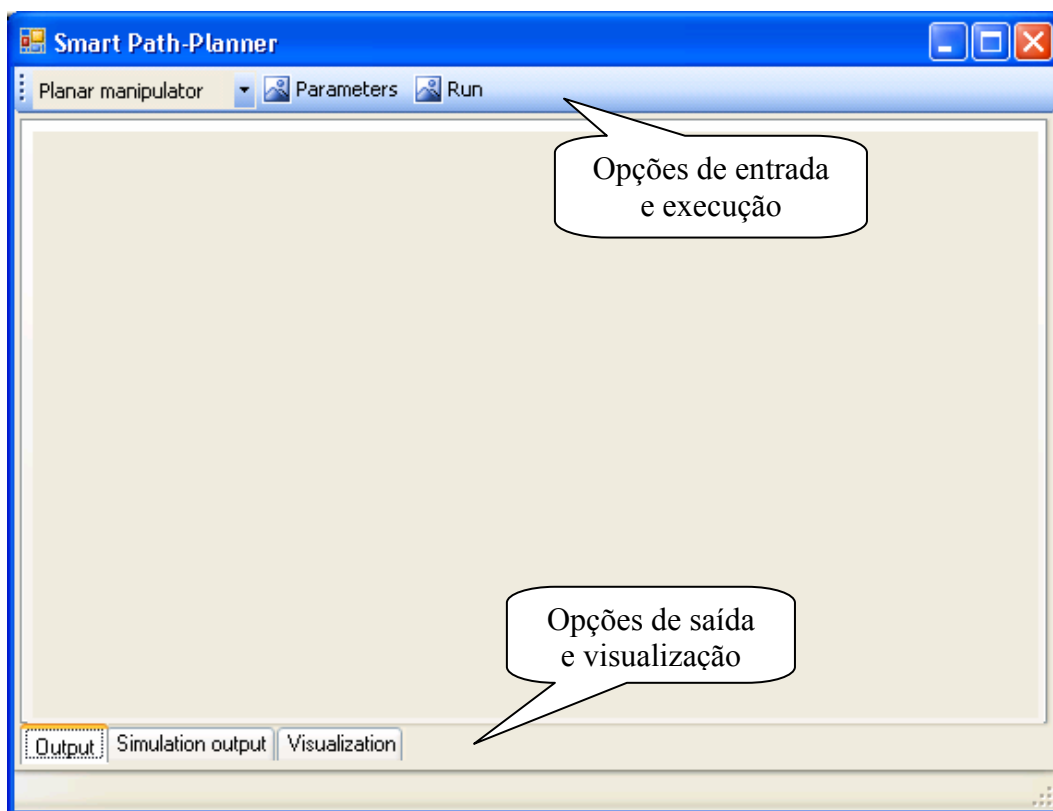


Figura 1. Tela principal do sistema.

O botão *Parameters* acessa a janela de especificação da trajetória (Figura 2), permitindo o estabelecimento de diversos parâmetros do movimento. O botão *Run* executa os processos de determinação da trajetória e otimização contidos no programa *simulation.exe*.

Este programa analisa os parâmetros especificados pelo usuário de acordo com o seguinte critério: a primeira estratégia de otimização [1] executa a rotina correspondente (Listagem 1), fazendo uso da respectiva função objetivo (Listagem 2). Caso contrário, a segunda estratégia de otimização [2] executa a rotina correspondente (Listagem 3), fazendo uso da respectiva função objetivo (Listagem 4).

Independentemente da estratégia escolhida, calcula-se a dinâmica do manipulador (Listagem 5) e a potência mecânica associada à trajetória (Listagem 6).

As informações calculadas pela simulação são automaticamente atualizadas nas telas de relatório do sistema (Figuras 3, 4 e 5).

O botão *Output* apresenta os dados numéricos obtidos antes e após a otimização, em formato compatível com o padrão *HTML* (Figura 3). O botão *Simulation output* apresenta os passos executados pelo módulo *simulation.exe*, bem como o valor das variáveis de projeto (Figura 4). O botão *Visualization* exibe o movimento do robô em um ambiente de realidade virtual (*VRML*), embutido em um *browser HTML* (Figura 5).

A execução do sistema requer a instalação do software *Microsoft .Net Framework 2.0* (www.microsoft.com) e de um visualizador *VRML* embutido em *browser* (www.blaxxun.com).

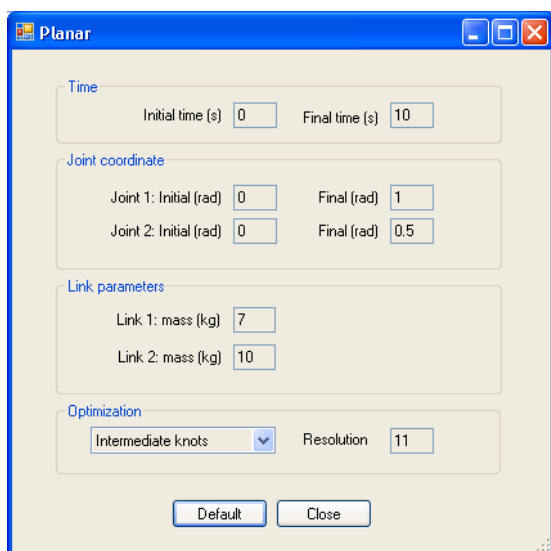


Figura 2. Parâmetros para estabelecimento da trajetória inicial e ótima.

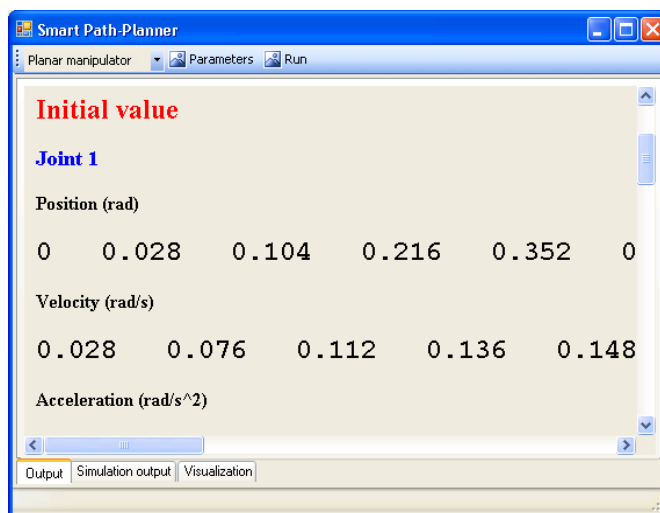


Figura 3. Especificação da trajetória inicial e ótima.

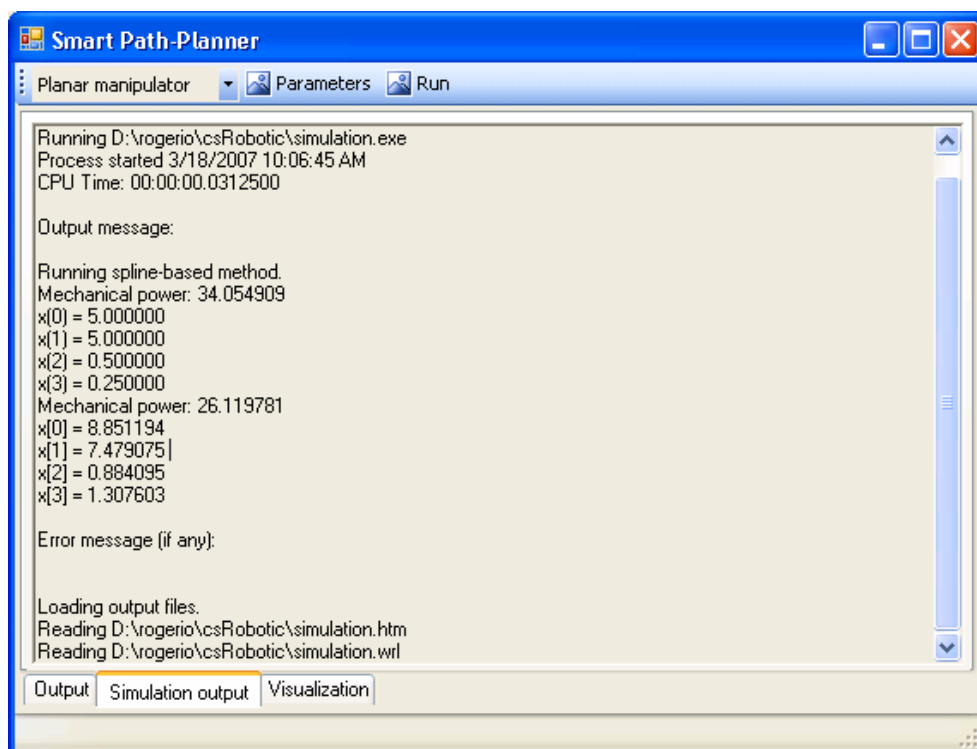


Figura 4. Valor inicial e ótimo das variáveis de projeto.

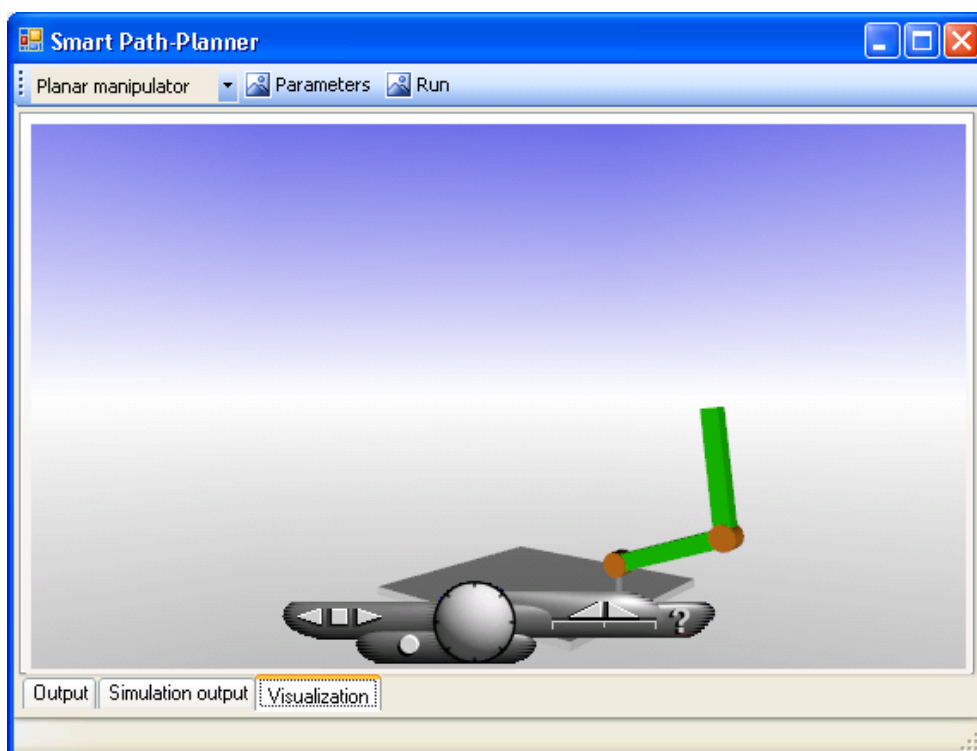


Figura 5. Visualização do movimento do robô.

```

void OptimumSpline(float *xIni)
{
    int nIter;
    float fTorque;
    float x[4], xl[4], xu[4];
    Dot bf2(4, 0, 0);

    printf("\nRunning spline-based method.");
    robo.SetTrajectory('a', 3, xIni);

    x[0] = xIni[1];
    x[1] = xIni[4];
    x[2] = xIni[7];
    x[3] = xIni[10];

    xl[0] = 0.1;
    xl[1] = 0.1;
    xl[2] = -7;
    xl[3] = -7;

    xu[0] = xIni[2] - 0.1; // final time
    xu[1] = xIni[5] - 0.1; // final time
    xu[2] = 7;
    xu[3] = 7;

    // before optimization
    fTorque = SolveSystem();
    printf("\nMechanical power: %f", fTorque);
    robo.SaveData('l', "output01.txt");

    for(int i=0; i<4; i++)
        printf("\nx(%d) = %f ", i, x[i]);

    // optimize
    bf2.SetFunction(MinimizeTorque);
    bf2.SetBound(xl, xu);
    bf2.GetMinimum(x, &nIter, &fTorque);

    // optimal result
    fTorque = SolveSystem();
    printf("\nMechanical power: %f", fTorque);
    robo.SaveData('l', "output02.txt");
    for(int i=0; i<4; i++)
        printf("\nx[%d] = %f ", i, x[i]);
}

```

Listagem 1. Otimização através da estratégia [1].

```

float MinimizeTorque(float knot[])
{
    float fTorque;
    float x[12];

    x[1] = knot[0]; // time link 1
    x[4] = knot[1]; // time link 2
    x[7] = knot[2]; // pos link 1

```

```

x[10] = knot[3]; // pos link 2

robo.SetTrajectory('i', 3, x);
fTorque = SolveSystem();

return fTorque;
}

```

Listagem 2. Função objetivo otimizada pela estratégia [1].

```

void OptimumControl(float *xIni)
{
    int nIter, i;
    float fTorque;
    float g[4];
    float *x = new float[iRes*2];
    Dot bf2(iRes*2, 4, 1);

    printf("\nRunning control-based method.");
    robo.SetTrajectory('a', 3, xIni); // spline interpolation
    for(i=0; i<iRes; i++)
    {
        x[i] = robo.GetVel(0,i);
        x[i+iRes] = robo.GetVel(1,i);
    }
    fTorque = SolveSystem();
    printf("\nMechanical power: %f", fTorque);
    robo.SaveData('l', "output01.txt");

    // optimize
    bf2.SetFunction2(MinimizeTorque2);
    bf2.GetMinimum2(x, g, &nIter, &fTorque);

    // optimized result
    fTorque = SolveSystem();
    printf("\nMechanical power: %f", fTorque);
    robo.SaveData('l', "output02.txt");
    for(i=0; i<4; i++) printf("\ng[%d] = %f ", i, g[i]);
    delete x;
}

```

Listagem 3. Otimização através da estratégia [2].

```

float MinimizeTorque2(float x[], float g[])
{
    float fTorque;
    float posini[2], posfim[2];
    robo.SetTrajectory('v', x);
    fTorque = SolveSystem();

    posini[0] = xIni[6];
    posini[1] = xIni[9];
    posfim[0] = xIni[8];
    posfim[1] = xIni[11];
}

```

```

g[0] = 0.0;
g[1] = 0.0;
g[2] = 0.0;
g[3] = 0.0;

for(int j=0; j<2; j++)
{
    g[0] += pow(robo.GetPos(j,iRes-1)-posfim[j], 2);
    g[1] += pow(robo.GetPos(j,0)-posini[j], 2);
    g[2] += pow(robo.GetVel(j,0), 2);
    g[3] += pow(robo.GetVel(j,iRes-1), 2);
}
return fTorque;
}

```

Listagem 4. Função objetivo otimizada pela estratégia [2].

```

float Planar::Dynamics(float x[])
{
    float q1, dq1, ddq1, q2, dq2, ddq2;
    float m1, m2, l, g;

    m1 = 1.0;
    m2 = 1.0;
    l = 1.0;
    g = -9.81;

    q1 = x[0];
    dq1 = x[1];
    ddq1 = x[2];
    q2 = x[3];
    dq2 = x[4];
    ddq2 = x[5];

    x[12] = 0.3333*m1*l*l*ddq1 + 1.3333*m2*l*l*ddq1 + 0.3333*m2*l*l*ddq2 +
m2*cos(q2)*l*l*ddq1
    + 0.5*m2*l*l*cos(q2)*ddq2 - m2*sin(q2)*l*l*dq1*dq2 -
0.5*m2*sin(q2)*l*l*dq2*dq2 + 0.5*m1*g*l*cos(q1)
    + 0.5*m2*g*l*cos(q1+q2) + m2*g*l*cos(q1);
    x[13] = 0.3333*m2*l*l*ddq1 + 0.3333*m2*l*l*ddq2 + 0.5*m2*l*l*cos(q2)*ddq1
    + 0.5*m2*g*l*cos(q1+q2) + 0.5*m2*l*l*sin(q2)*dq1*dq1;

    return 0;
}

```

Listagem 5. Dinâmica do manipulador planar.

```

float SolveSystem()
{
    int i;
    float fMin;
    float fTorque, fDelta;
    float *x;

    x = new float[4*iRes];
}

```

```

fDelta = (robo.GetTime(0,iRes-1) - robo.GetTime(0,0)) / (iRes-1);
for(i=0; i<iRes; i++)
{
    x[i] = robo.GetPos(0,i);
    x[i+iRes] = robo.GetPos(1,i);
    x[i+iRes+iRes] = 0;
    x[i+iRes+iRes+iRes] = 0;
}

fMin = robo.DynamicsError(x);

fTorque = 0;
for(i=0; i<iRes; i++)
{
    fTorque += pow(robo.GetTorque(0,i) * robo.GetVel(0,i) * fDelta, 2) +
    pow(robo.GetTorque(1,i) * robo.GetVel(1,i) * fDelta, 2);
}
delete x;
return fTorque;
}

```

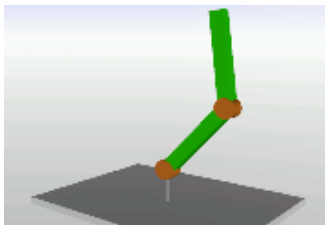
Listagem 6. Potência mecânica associada ao movimento.

Referências Bibliográficas

- [1] R.R. Santos, V. Steffen, Jr. and S. F. P. Saramago, “Robot Path Planning: Avoiding Obstacles”, in: 18th International Congress of Mechanical Engineering (COBEM), 2005, Ouro Preto - MG, 2005.
- [2] R.R. Santos, V. Steffen, Jr. and S. F. P. Saramago, “Robot Path Planning in a Constrained Workspace by using Optimal Control Techniques”, in: III European Conference on Computational Mechanics Solids, Structures and Coupled Problems in Engineering, Lisboa, 2006.

Apêndice C

Manual do usuário.



Smart Path-Planner
A system for optimal robot path planning

Introduction

The system generates the trajectory for a serial robot manipulator. Given the initial and final joint coordinates, the system computes a point-to-point trajectory with the required resolution.

If no optimization is required, a cubic spline-based interpolation is performed to compute the intermediate values. It is possible to choose between two different strategies [1,2] to obtain the optimal path planning.

The output provides information regarding time, position, velocity, acceleration and torque of each joint. It is generated according to the *HyperText Markup Language (HTML)* standard and can be readily used to exchange data.

The optimal robot trajectory is visualized through a *Virtual Reality Modeling Language (VRML)* file, enabling interaction with the robot workspace. The file can be updated to describe a virtual environment, as required.

The user can freely modify the robot parameters and the corresponding VRML file (according the ISO/IEC 14772-1:1997 specification).

Instalation

The system is provided as a “ready to run” set of files:

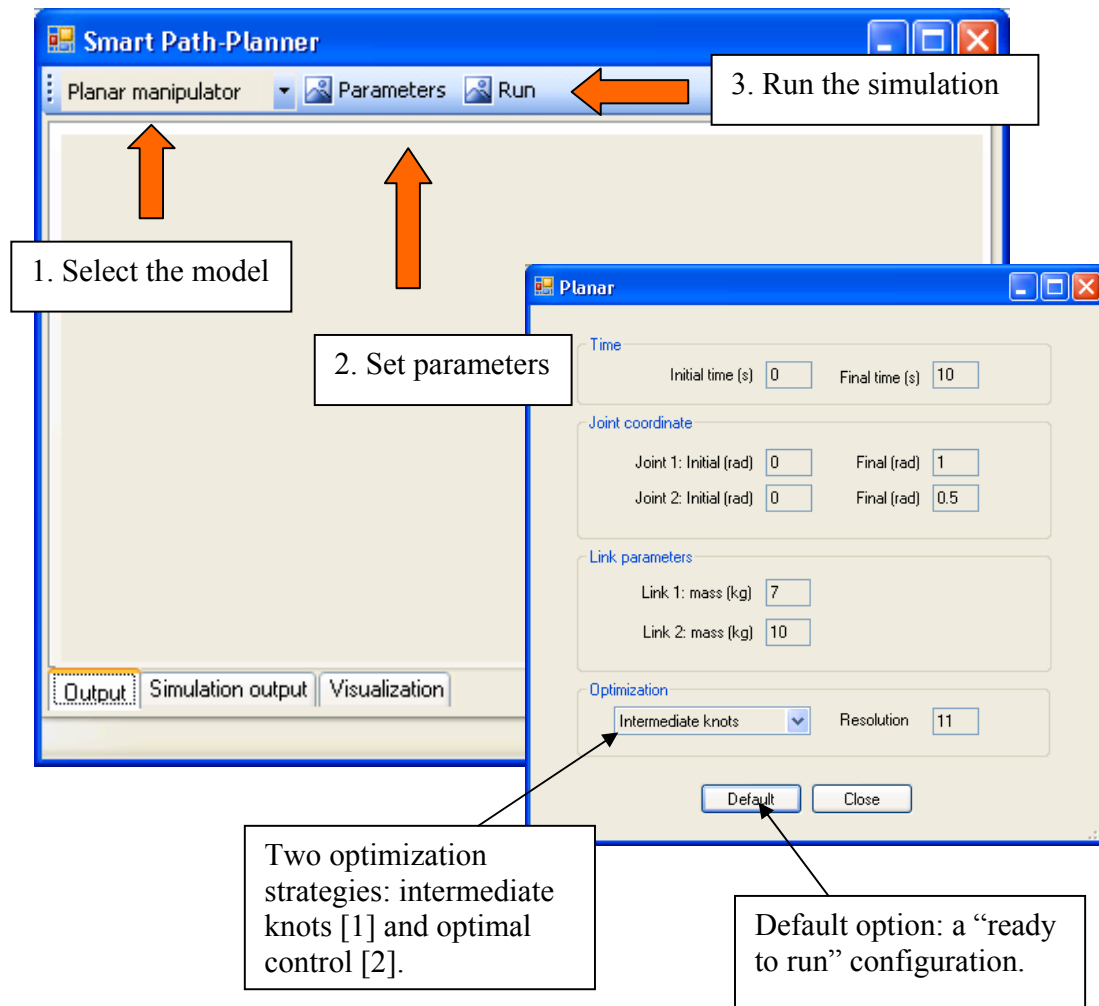
Spp.exe	- Main program.
Simulation.exe	- Optimization library.
Planar.wrl	- Robotic VRML model.

Therefore, put all the files in the same directory and invoke *spp.exe* to start the system. The following files have also to be installed to prepare the system to run properly:

Dotnetfx.exe	- Microsoft .Net Framework 2.0. (http://www.microsoft.com)
BlaxxunContact44.exe	- VRML Viewer. (http://www.blaxxun.com)

Input

Run the simulation through 3 steps:



[1] R.R. Santos, V. Steffen, Jr. and S. F. P. Saramago, "Robot Path Planning: Avoiding Obstacles", in: 18th International Congress of Mechanical Engineering (COBEM), 2005, Ouro Preto - MG, 2005.

[2] R.R. Santos, V. Steffen, Jr. and S. F. P. Saramago, "Robot Path Planning in a Constrained Workspace by using Optimal Control Techniques", in: III European Conference on Computational Mechanics Solids, Structures and Coupled Problems in Engineering, Lisboa, 2006.

Output

Visual and numerical information:

