

---

# CINEMÁTICA INVERSA DE TRAJETÓRIAS DE MANIPULADORES ROBÓTICOS REDUNDANTES UTILIZANDO ALGORITMOS GENÉTICOS COM REDUÇÃO PROGRESSIVA DO ESPAÇO DE BUSCA

**Alfranci Freitas Santos\***  
freitas@cpgei.cefetpr.br

**Munif Gebara Junior†**  
munif@cesumar.br

**Heitor Silvério Lopes\***  
hslopes@cpgei.cefetpr.br

\* Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial,  
Centro Federal de Educação Tecnológica do Paraná - CEFET-PR  
Av. 7 de setembro, 3165 – 80230-901 Curitiba (PR)

† Centro Universitário de Maringá - CESUMAR  
Av. Guedner, 1610 – Jd. Aclimação  
87050-390 – Maringá (PR)

---

## ABSTRACT

The problem of inverse kinematics of manipulators is always present in robotics. The solution of this problem, particularly for redundant manipulators, is difficult because it includes nonlinear equations and an extremely large number of feasible solutions for most end-effector locations. Other authors have reported the use of genetic algorithms (GA) to solve the problem of the inverse kinematics for single points. In this work, we present a GA to solve this problem for all points of trajectories. It is introduced a technique named "range reduction" that improves both the convergence speed of the algorithm and its accuracy besides avoiding jumps between multiple solutions. Two case studies are presented, for non-redundant and redundant manipulators. Results for four different test trajectories showed that this methodology can be used efficient for real-world applications.

**KEYWORDS:** Genetic algorithms, robotics, robotic manipulator, inverse kinematics.

## RESUMO

O problema da cinemática inversa de manipuladores está sempre presente em robótica. A solução deste problema, particularmente para o caso dos manipuladores redundantes, é difícil por incluir equações não-lineares e possuir um número muito grande de soluções admissíveis para a maior parte das posições-orientações do efetuador. Alguns autores já utilizaram os algoritmos genéticos para resolver o problema da cinemática inversa para pontos isolados. Neste trabalho é apresentado um algoritmo genético capaz de resolver este problema para todos os pontos de uma trajetória. É ainda introduzida uma técnica chamada "redução progressiva do espaço de busca" – REB

– que melhora tanto a velocidade quanto a precisão da convergência, além de evitar saltos entre soluções múltiplas. São abordados dois estudos de caso: manipuladores redundantes e não-redundantes. Os resultados obtidos para quatro diferentes trajetórias de teste mostram que esta metodologia pode ser utilizada eficientemente em aplicações do mundo real.

**PALAVRAS-CHAVE:** Algoritmos genéticos, robótica, manipulador robótico, cinemática inversa..

## 1 INTRODUÇÃO

A cinemática inversa é um problema complexo: dada a posição-orientação desejada do efetuador, obter o conjunto dos ângulos-de-junta que alcançam esta posição-orientação. A solução deste problema é difícil porque inclui equações não-lineares de funções transcendentais e, em geral, leva a soluções múltiplas. Em certos casos o problema admite somente soluções iterativas. Para a cinemática inversa de trajetórias o problema se torna ainda mais difícil porque o processo precisa ser repetido para cada ponto da trajetória. Assim, o emprego de métodos alternativos, como os Algoritmos Genéticos (AG), para a resolução desta classe de problemas é plenamente justificada. O presente trabalho apresenta a solução da cinemática inversa de trajetórias utilizando o AG para dois casos: manipuladores redundantes e não-redundantes.

Em Da Silveira, Coelho e Campos (1994), o problema da cinemática inversa utilizando algoritmos genéticos foi explorado para pontos isolados. Aquele trabalho é substancialmente distinto do presente, principalmente porque aqui é tratado o mapeamento da trajetória completa, enquanto que o artigo em questão aborda apenas pontos isolados. Ainda mais, são introduzidas novas melhorias na

técnica básica objetivando a computação de trajetórias (no espaço-articular) quase em tempo real.

O problema da cinemática inversa para pontos isolados também foi abordado por Parker, Khoogar e Goldberg (1989). Naquele trabalho, a função de *fitness* combina o erro de posicionamento do braço com um termo adicional baseado nos deslocamentos dos ângulos-de-junta a partir da configuração inicial. O objetivo daquele trabalho era escolher a configuração que coloca o efetuador na posição-orientação correta e minimizar os movimentos (rotações) das juntas.

Em Toogod, Rao e Wong (1995), os AG foram utilizados para obter um trajeto livre de colisões para um manipulador de 3-GDL (3 graus de liberdade) com juntas rotativas, entre configurações de partida e parada especificadas, através de um espaço contendo obstáculos fixos conhecidos. Além de evitar colisões e de se manter dentro dos limites de junta especificados, o trajeto poderia ser otimizado para a distância mínima, tempo mínimo ou torques-de-junta mínimos.

## 2 FUNDAMENTAÇÃO

### 2.1 Algoritmos genéticos

AGs têm sido largamente utilizados como método de busca heurístico em problemas onde ou o tamanho do espaço de busca é intratável ou os métodos convencionais não são eficientes. AGs têm sido particularmente importantes para a solução de problemas nas áreas de Engenharia (Lopes, 1999). Em essência, AGs são algoritmos de busca baseados na mecânica da seleção natural e da genética natural (Goldberg, 1989), seguindo o princípio da sobrevivência do mais bem adaptado. No contexto da busca no espaço de soluções, possíveis soluções para um problema são considerados indivíduos de uma população. Os melhores candidatos correntes a solução são selecionados para reprodução. Operadores genéticos tais como seleção, cruzamento e mutação são estocasticamente aplicados aos indivíduos de melhor qualidade para gerarem descendentes. Pelo processo de seleção natural, ao longo de várias gerações soluções de qualidade crescente emergem na população, até que um critério de parada pré-definido seja atingido.

Uma das vantagens dos AGs sobre os métodos de busca tradicionais é que os primeiros realizam uma espécie de busca global usando uma população de soluções, em vez de realizar uma busca local, de subida da colina. Os métodos de busca global são menos sujeitos a serem presos pelos máximos locais, quando comparados com os métodos de busca local. Além disto, AGs realizam uma busca paralela intrínseca e extrínsecamente. Isto é, o paralelismo intrínseco se refere à possibilidade de otimizar vários parâmetros do problema em questão ao mesmo tempo, enquanto que o paralelismo extrínseco se refere à capacidade de manipular e testar várias soluções simultaneamente (população de indivíduos).

### 2.2 Cinemática direta e inversa, espaço de trabalho do manipulador

Um manipulador robótico pode ser visto como um conjunto de corpos articulados por juntas. Estes corpos são chamados segmentos. Cada junta forma a conexão entre um par de segmentos vizinhos. A extremidade útil do manipulador é chamada de efetuador. Uma configuração do manipulador é um conjunto de valores (vetor) que representa os ângulos-de-junta entre pares de segmentos consecutivos. A cinemática direta mapeia a configuração (vetor de ângulos-de-junta) para as coordenadas cartesianas (posição-orientação) do efetuador. De forma similar, a cinemática inversa mapeia as coordenadas cartesianas do efetuador (posição-orientação) para a configuração do manipulador (ângulos-de-junta).

A cinemática direta sempre tem solução única. Porém, a cinemática inversa pode resultar em nenhuma solução, uma única solução, várias soluções ou mesmo infinitas soluções. Uma discussão detalhada dos assuntos dessa subseção é encontrada em Craig (1989).

## 3 METODOLOGIA

Neste artigo é tratada a cinemática inversa da posição do efetuador, não da orientação, que será tratada no futuro.

São realizados dois estudos de caso utilizando um manipulador não-redundante bem conhecido, o PUMA-560, e uma versão modificada deste equipamento. No primeiro estudo de caso o objetivo é ajustar experimentalmente o AG para o problema da cinemática inversa de trajetórias. É particularmente centrado na codificação do cromossomo, na avaliação da função de *fitness* e no estabelecimento das restrições. Os resultados deste estudo são comparados com os resultados obtidos pela técnica tradicional proposta por Craig (1998). No segundo estudo de caso a metodologia básica desenvolvida no primeiro é novamente utilizada, mas o manipulador é modificado. A alteração consiste na introdução de um segmento na extremidade do braço que o transforma em um manipulador redundante para posicionamento. O objetivo aqui é investigar a aplicabilidade do AG para este tipo de problema.

Em ambos os casos o problema da cinemática inversa é repetido para todos os pontos de quatro trajetórias de teste diferentes. Supõe-se que os pontos consecutivos de uma trajetória são próximos. Assim, para cada ponto ( $p_1, p_2, \dots, p_n$ ) o AG é rodado até que uma precisão pré-definida seja atingida. Depois que o ponto  $p_k$  é alcançado, o AG utiliza o “conhecimento” da população para encontrar o próximo ponto  $p_{k+1}$ . O primeiro ponto da trajetória é buscado no espaço-de-trabalho completo do manipulador, logo se espera que esta busca seja bastante difícil. Após isto, o espaço de busca é restringido para uma região envolvente ao último ponto encontrado. A restrição do espaço de busca é obtida por meio de uma redução temporária nos limites angulares das juntas, que será discutida a seguir. Isto faz com que o AG seja executado mais rapidamente, e um experimento foi montado para verificar esse fato.

### 3.1 O Algoritmo Genético

#### 3.1.1 Codificação dos cromossomos e restrições

As variáveis do problema são os ângulos de junta mostrados na figura 1. A codificação dos cromossomos para o manipulador não-redundante é detalhada na tabela 1, enquanto que para o manipulador redundante, é dada na tabela 2. A precisão requerida é de 0,01 grau, um valor aceitável para problemas reais.



Figura 1- Ângulos de junta

Tabela 1: Codificação de variáveis para o manipulador não-redundante

Variável	Min	Max	Bits	Precisão
$\theta_1$	-170°	170°	16	0,00519°
$\theta_2$	-225°	45°	16	0,00412°
$\theta_3$	-250°	75°	16	0,00496°

Tabela 2: Codificação de variáveis para o manipulador redundante

Variável	Min	Max	Bits	Precisão
$\theta_1$	-170°	170°	16	0,00519°
$\theta_2$	-225°	45°	16	0,00412°
$\theta_3$	-250°	75°	16	0,00496°
$\theta_4$	-135°	135°	16	0,00411°
$\theta_5$	-100°	100°	16	0,00305°

O código de Gray é empregado em vez do código binário natural, a fim de evitar mutações disruptivas nos cromossomos submetidos aos operadores genéticos (Forrest, 1993). Foi empiricamente observado que AGs empregando código de Gray convergem para uma solução aceitável mais rapidamente do que com o uso do código binário natural.

A partir da codificação proposta, define-se o espaço de busca que contém cerca de  $10^{14}$  pontos para o manipulador não-redundante e cerca de  $10^{24}$  para o manipulador redundante. Embora estes espaços de busca não sejam um grande desafio para os AGs, é importante enfatizar que a busca é repetida para cada ponto de uma trajetória. As quatro trajetórias ensaiadas neste trabalho possuem 251 pontos cada.

Para o manipulador não-redundante, os ângulos de junta são obtidos pela equação 1, a seguir.

$$\theta_i = Centro_i + \frac{Valor_i \cdot Faixa_i}{2^{bits} - 1} - \frac{Faixa_i}{2} \quad (1)$$

onde:  $\theta_i$  é o ângulo da  $i$ -ésima junta,  $Centro_i$  é o valor médio de  $\theta_i$ ,  $Valor_i$  é o  $i$ -ésimo valor no cromossomo,  $Faixa_i$  é a faixa de variação de  $\theta_i$  e  $bits$  é o número de bits utilizado para codificar cada gene. Por exemplo, se  $\theta_1$  pode assumir valores entre 20 e 100,  $Centro$  é 60 e  $Faixa$  é 80.

Os limites de rotação das juntas são manipulados diretamente na codificação, de forma que qualquer solução determinada pelo AG é fisicamente admissível.

#### 3.1.2 A técnica de Redução Progressiva do Espaço de Busca (REB)

Uma trajetória é composta de vários pontos. Para cada ponto, o AG busca iterativamente uma solução que satisfaça a precisão requerida. Seria desejável reduzir o número de gerações necessárias para alcançar cada ponto, a fim de diminuir o tempo global de processamento de uma rodada. Isto pode ser obtido por meio de uma redução adaptativa do espaço de busca. Para cada novo ponto da trajetória, a busca é restringida a uma vizinhança desse último ponto obtido. Os parâmetros  $Faixa$  e  $Centro$  da equação 1 são mudados de forma a atingir este objetivo. Experimentalmente o valor de 5° para  $Faixa$  deu bons resultados. Note que este valor é uma fração simples da faixa total de todas as juntas (veja as tabelas 1 e 2). Por outro lado,  $Centro$  é fixado em um ponto obtido por extrapolação linear dos dois últimos pontos. Os limites das juntas são sempre verificados para evitar codificações e valores de junta não admissíveis.

Como um efeito colateral positivo do emprego da técnica REB a probabilidade de saltos entre soluções múltiplas é reduzida.

#### 3.1.3 Função objetivo

Sob esta abordagem o problema da cinemática inversa se reduz a um problema de minimização. É objetivado minimizar a distância entre um ponto desejado e um ponto obtido com a cinemática direta utilizando os ângulos de junta decodificados do cromossomo. A distância é calculada por meio da equação 2:

$$Distância = \sqrt{(p_x - p_x^*)^2 + (p_y - p_y^*)^2 + (p_z - p_z^*)^2} \quad (2)$$

onde  $p_x^*$ ,  $p_y^*$  e  $p_z^*$  são as coordenadas da posição desejada do efetuador, e  $p_x$ ,  $p_y$  e  $p_z$  são as coordenadas do ponto obtido com a cinemática direta.

Para o manipulador não-redundante a cinemática direta é dada pelo conjunto de equações 3, onde  $d_3$ ,  $d_4$ ,  $a_2$  e  $a_3$  são constantes do manipulador;  $\cos_1$ ,  $\cos_2$ ,  $\cos_{23}$ ,  $\sin_1$ ,  $\sin_2$  e  $\sin_{23}$  são, respectivamente,  $\cos(\theta_1)$ ,  $\cos(\theta_2)$ ,  $\cos(\theta_2 + \theta_3)$ ,  $\sin(\theta_1)$ ,  $\sin(\theta_2)$ , e  $\sin(\theta_2 + \theta_3)$ :

$$\begin{cases} p_x = \cos_1 [a_2 \cos_2 + a_3 \cos_{23} - d_4 \sin_{23}] - d_3 \sin_1 \\ p_y = \sin_1 [a_2 \cos_2 + a_3 \cos_{23} - d_4 \sin_{23}] + d_3 \cos_1 \\ p_z = -a_3 \sin_{23} - a_2 \sin_2 - d_4 \cos_{23} \end{cases} \quad (3)$$

Para o manipulador redundante a cinemática direta é bem mais complexa do que a do manipulador não-redundante. Esta cinemática é dada pelo conjunto de equações 4, onde  $d_3$ ,  $d_4$ ,  $a_2$  e  $a_3$  são constantes do manipulador;  $c_i$  e  $s_i$  são, respectivamente,  $\cos(\theta_i)$  e  $\sin(\theta_i)$ .

$$\begin{cases} p_x = a_2 c_1 c_2 + a_3 c_1 c_2 c_3 - d_3 s_1 - d_4 c_1 c_3 s_2 - d_7 c_1 c_3 c_5 s_2 - \\ d_4 c_1 c_2 s_3 - d_7 c_1 c_2 c_5 s_3 - a_3 c_1 s_2 s_3 - d_7 c_1 c_2 c_3 c_4 s_5 + \\ d_7 c_1 c_4 s_2 s_3 s_5 - d_7 s_1 s_4 s_5 \\ p_y = d_3 c_1 + a_2 c_2 s_1 + a_3 c_2 c_3 s_1 - d_4 c_3 s_1 s_2 - d_7 c_3 c_5 s_1 s_2 - \\ d_4 c_2 s_1 s_3 - d_7 c_2 c_5 s_1 s_3 - a_3 s_1 s_2 s_3 - d_7 c_2 c_3 c_4 s_1 s_5 + \\ d_7 c_4 s_1 s_2 s_3 s_5 + d_7 c_1 s_4 s_5 \\ p_z = -(d_4 c_2 c_3) - d_7 c_2 c_3 c_5 - a_2 s_2 - a_3 c_3 s_2 - a_3 c_2 s_3 + \\ d_4 s_2 s_3 + d_7 c_5 s_2 s_3 + d_7 c_3 c_4 s_2 s_5 + d_7 c_2 c_4 s_3 s_5 \end{cases} \quad (4)$$

A função objetivo é claramente não-estacionária: o algoritmo é executado até que a distância seja menor do que uma dada tolerância (0,002 polegada = 0,0508 mm). Então os valores de  $p_x^*$ ,  $p_y^*$  e  $p_z^*$  são atualizados para o próximo ponto da trajetória. Em consequência, o ambiente no qual o AG tem que otimizar é mudado, forçando-o a adaptar-se estocasticamente à busca do novo objetivo.

### 3.1.4 Função de fitness

Os AGs, por definição, tentam maximizar uma dada função. Tendo em vista adaptar a função objetivo para um problema de maximização, a função de *fitness* utilizada é a mostrada na equação 5, onde  $C$  é uma constante maior do que a máxima distância entre quaisquer pontos dentro do espaço-de-trabalho do manipulador. Neste estudo o valor empregado é 100 polegadas (cerca de 254 centímetros).

$$Fitness = C - Distância \quad (5)$$

### 3.1.5 Critérios de parada

Há dois critérios de parada nesta implementação. O primeiro é quando a trajetória completa no espaço-articular é gerada com sucesso. O segundo é quando o AG alcança seu limite superior de gerações. Este limite é fixado em 10.000 para o caso não-redundante e 20.000 para o redundante. Este último critério de parada habilita o AG a parar por *time-out* no caso de não ocorrer convergência. Note que se o ponto procurado não se encontra dentro do espaço-de-trabalho a convergência não ocorrerá.

## 3.1.6 Operadores genéticos e parâmetros de controle

A população inicial é fixada em 100 indivíduos. Os operadores genéticos são o cruzamento (*crossover*) uniforme e a mutação, que são aplicados com probabilidades de 1,00 e 0,01 respectivamente. O método de seleção utilizado é a seleção por torneio estocástico, com tamanho de torneio 15. Este método conserva níveis razoáveis de diversidade genética através das gerações sem excessiva pressão seletiva. Uma diversidade genética elevada na população é necessária porque o AG não para quando encontra um ponto da trajetória, mas quando todos os pontos são obtidos.

## 3.2 Trajetórias de teste

As trajetórias especificadas na tabela 3 são empregadas para avaliar o desempenho dos AGs nos dois estudos de caso.

A trajetória #1 é uma linha reta inclinada em relação aos três planos coordenados. A trajetória #2 é uma hélice de eixo vertical e raio crescente. A trajetória #3 é uma curva reversa no espaço tridimensional. A trajetória #4 é um *zig-zag* no plano vertical  $y=8$ . Estas trajetórias têm diferentes níveis de complexidade e todas são ilustradas na figura 2.

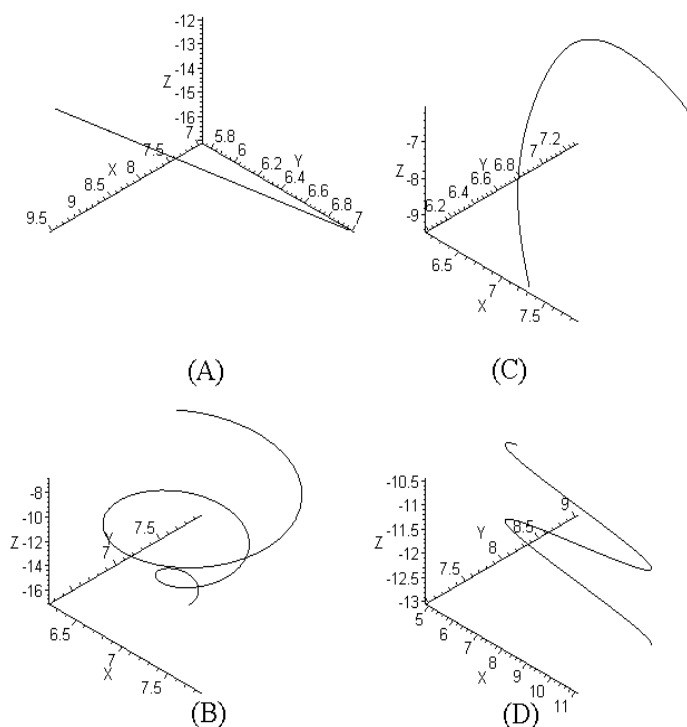
Tabela 3: Trajetórias de teste

Trajeto	Num.	Duração	Equações
		pontos e Passo	
#1	251	0 a 2,5 s 0,01 s	$P_x=7+t$ $P_y=7-t/2$ $P_z=-17+2*t$
#2	251	0 a 5 s 0,02 s	$P_x=7+\cos(t*3)*t/5$ $P_y=7+\sin(t*3)*t/5$ $P_z=-17+2*t$
#3	251	0 a 2,5 s 0,01 s	$P_x=7+\sin(\cos(t*2))$ $P_y=7+t*\cos(t)/2$ $P_z=-7+2*t*\cos(t)*\sin(t)$
#4	251	0 a 2,5 s 0,01 s	$P_x=8+3*\cos(t*4)$ $P_y=8$ $P_z=-13+t$

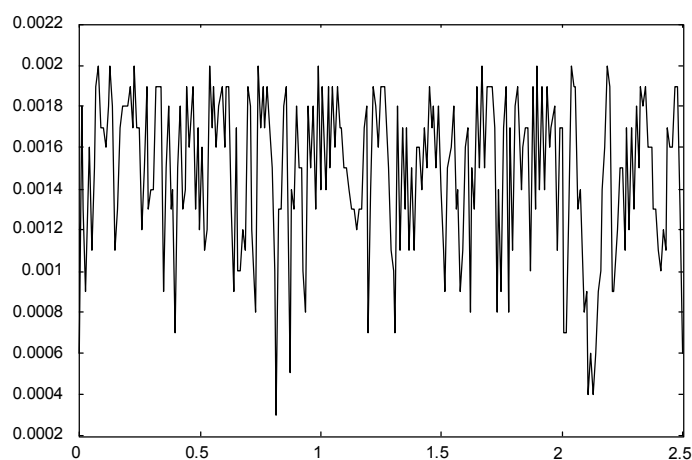
## 4 RESULTADOS

A figura 3 mostra a distância entre cada ponto da trajetória #1 e o ponto correspondente gerado pelo AG para o manipulador não-redundante, na medida em que as gerações avançam. Resultados similares são encontrados para as outras trajetórias e também para o manipulador redundante.

A tabela 4 apresenta os resultados das rodadas para o manipulador não-redundante, com e sem redução progressiva do espaço de busca - REB. O número de gerações necessárias é indicado por  $N^\circ$ Gerações, o indicador *Compl.* indica se a trajetória foi completada ou não, e *Ger/Pt* indica o número médio de gerações para alcançar cada ponto. REB indica o emprego da técnica de redução progressiva de espaço de busca.



**Figura 2:** (A) Trajetória de teste #1, (B) Trajetória de teste #2, (C) Trajetória de teste #3, (D) Trajetória de teste #4.



**Figura 3:** Distância (polegadas) versus tempo (s).

A tabela 5 apresenta os resultados para o caso do manipulador redundante onde as mesmas quatro trajetórias são alcançadas com sucesso com a técnica de redução do espaço.

Todas as rodadas reportadas aqui foram realizadas utilizando um computador PC-compatível de baixo desempenho (Pentium 300 MHz). Para cada uma das quatro trajetórias do caso não-redundante o tempo de execução ficou em menos do que um minuto, e para o caso do manipulador redundante cada rodada completa tomou menos do que três minutos.

**Tabela 4: Resultados para o manipulador não-redundante**

		NºGerações	Compl.	Ger/Pt
Trajetória #1	sem REB	10.000*	18,7%	210
	com REB	2.000	100%	7
Trajetória #2	sem REB	10.000*	16,3%	240
	com REB	6.678	100%	26
Trajetória #3	sem REB	10.000*	19,9%	198
	com REB	4.245	100%	16
Trajetória #4	sem REB	10.000*	15,5%	255
	com REB	8.587	100%	34

\*Não alcançou o resultado após 10000 gerações.

**Tabela 5: Resultados para o manipulador redundante**

		Nº Gerações	Compl.	Ger/Pt
Trajetória #1		7.309	100%	29
Trajetória #2		13.137	100%	52
Trajetória #3		10.100	100%	40
Trajetória #3		13.817	100%	55

Para o manipulador não-redundante o primeiro ponto das trajetórias de teste #1 e #2 foi obtido com 359 gerações; para a trajetória #3 foram necessárias 335 gerações e para a #4 foram realizadas 356 gerações. Para o manipulador redundante o primeiro ponto das trajetórias #1 e #2 foi obtido com 689 gerações; para a trajetória #3 foram usadas 902 gerações e para a #4, 402 gerações. A coincidência do número de gerações para obter o primeiro ponto das trajetórias #1 e #2 em ambos os manipuladores deve-se ao fato de que estas duas trajetórias partem do mesmo ponto (7; 7; -17).

## 5 DISCUSSÃO

A figura 3 mostra que a distância entre o ponto desejado e o ponto obtido pelo AG oscilou entre a tolerância especificada e valores muito pequenos (da ordem de  $10^{-4}$ ), dando assim uma precisão extremamente satisfatória no posicionamento.

As trajetórias geradas para o manipulador não-redundante foram comparadas com os resultados exatos obtidos por meio do método convencional da cinemática inversa (solução em forma fechada) e se mostraram indistinguíveis exceto pelo valor da tolerância.

Como esperado, o número de gerações para obter o primeiro ponto da trajetória foi muito maior do que o número de gerações necessário para alcançar os pontos sucessivos com o emprego de REB. Um resultado interessante é que no caso do manipulador não-redundante aproximadamente 25% dos pontos foram obtidos com uma única geração, o que indica que este ponto já estava presente na população da geração anterior.

Conforme apresentado na tabela 4, o emprego da técnica de redução progressiva do espaço de busca (REB) causou uma sensível aceleração no AG, encontrando pontos da trajetória em cerca de um décimo do número de gerações necessário sem a utilização desta técnica. Nenhum salto entre soluções múltiplas foi registrado, assegurando a continuidade da trajetória no espaço-articular.

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Utilizando AGs, trajetórias articulares perfeitas foram geradas para ambos manipuladores, redundante e não-redundante, em quatro diferentes trajetórias de teste. O primeiro estudo de caso, utilizando um manipulador para o qual a solução da cinemática inversa em forma fechada é conhecida, foi importante para ajustar convenientemente o AG e comparar resultados.

A hipótese adotada de que o mesmo conjunto de parâmetros do AG que funcionou muito bem para o primeiro estudo de caso funcionaria da mesma forma para o segundo, foi constatada ser válida. Por extrapolação, pode-se inferir que esta metodologia de emprego dos AGs poderia também ser válida para outros problemas similares. Para manipuladores (em particular para os redundantes) para os quais a cinemática inversa é desconhecida ou excessivamente complexa para deduzir, o uso de AGs na forma mostrada neste trabalho é certamente uma alternativa válida. Além disto, foi mostrado que com a utilização da técnica REB o AG é dramaticamente acelerado.

Se o sistema tivesse rodado em um computador atualizado, muito mais rápido do que o utilizado, é perfeitamente possível afirmar que é factível gerar trajetórias em tempo real. Assim, trabalhos futuros podem visar outras melhorias de velocidade no algoritmo básico, como operadores específicos, por exemplo. Trabalhos futuros poderão cobrir problemas mais complexos de manipuladores, incluindo a geração otimizada de trajetórias segundo algum critério, tais como a minimização do movimento das juntas ou o desvio de pontos singulares.

## REFERÊNCIAS

- Craig, J.J (1989). *Introduction to Robotics Mechanics and Control*. Addison-Wesley, Reading, MA.
- Da Silveira, C.H., Coelho, L.S. e Campos, M.F.M. (1994). The use of genetic algorithms for the evaluation of inverse kinematics of manipulators. In: *Anais do 1º Congresso Brasileiro de Redes Neurais*, Itajubá, Brasil, [s.p.].
- Lopes, H.S. (1999). Algoritmos genéticos em projetos de engenharia: aplicações e perspectivas futuras. In: *Anais do SBAI'99 - 4o. Simpósio Brasileiro de Automação Inteligente*. São Paulo, 08-10/setembro, pp. 64-74.
- Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA.
- Parker, J.K., Khoogar, A.R. e Goldberg, D.E. (1989). Inverse kinematics of redundant robots using genetic algorithms. In: *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 271-276.
- Forrest, S. (1993). Genetic algorithms: principles of natural selection applied to computation. *Science*, Vol. 261, pp. 872-878.
- Toogood, R., Hao, H. e Wong, C. (1995). Robot path planning using genetic algorithms. In: *Proceedings of IEEE Systems, Man and Cybernetics Conference*, Vol. 1, pp. 489-494.