

Desenvolvimento de uma estrutura de controle de posição aplicada ao Manipulador Robótico RD5NT

Development of a position control structure applied to the RD5NT Robotic Manipulator

Sérgio Assis Galito Araújo*

Neste trabalho é desenvolvida uma estrutura de controle de posição do tipo RASTRO e aplicada no estudo de caso com uso do manipulador robótico modelo RD5NT do fabricante Didacta Itália. São detalhados a formulação matemática da estrutura de controle assim como os procedimentos adotados para gerar a trajetória desejada e para identificar em tempo real o modelo matemático do manipulador robótico. A seguir, a estrutura de controle apresentada é implementada numericamente e, através de simulações numéricas, é avaliada a qualidade do controlador proposto.

This paper presents a RASTRO position control structure applied to a case study using the RD5NT robotic manipulator produced by Didacta Italia. The study provides details of the mathematical formulation of the control structure, as well as the procedures used to generate the desired trajectory and identify, in real time, the mathematical model of the manipulator. Following, the control structure is numerically implemented and, through numerical simulations, the quality of the controller is assessed.

Palavras-chave: Automação. Controle de Posição. Robótica.

Key words: Automation. Position Control. Robotics.

Introdução

Há muitos séculos a humanidade faz uso de estratégias de controle para a automação de tarefas repetitivas. Como exemplo, pode-se citar o desenvolvimento, em 300 a.C., do mecanismo regulador de boia na Grécia (MAYR, 1970). A necessidade de controlar sistemas dinâmicos cada vez mais sofisticados, tais como sistemas de tráfego urbano, processos químicos industriais, estruturas robóticas, entre outros, impôs enormes desafios à Engenharia de Controle e Automação, o que resultou em um grande desenvolvimento da área.

Um das principais áreas de atuação da Engenharia de Controle e Automação é o desenvolvimento de robôs industriais. Esse desenvolvimento, no início do século XVIII, foi motivado pelo interesse em automatizar as operações dos teares, com o objetivo de diminuir custos de produção, aumentar a qualidade dos produtos, e, desta forma, incrementar a produtividade da indústria têxtil. Apesar desse esforço inicial,

* Mestre em Engenharia Mecânica pela Universidade Federal Fluminense (UFF). Professor do Instituto Federal de Educação, Ciência e Tecnologia Fluminense - Campos dos Goytacazes/RJ - Brasil. E-mail: saraujo@ifff.edu.br

o desenvolvimento de verdadeiros robôs só é possível a partir dos anos 40 com o desenvolvimento dos computadores (MAYR, 1970).

Atualmente, a principal área de atuação dos robôs está na indústria, principalmente nos setores automotivos e bélicos, exercendo tarefas repetitivas, trazendo, como principais vantagens, flexibilidade, alta produtividade e melhor qualidade dos produtos.

Sob o ponto de vista de controle, os robôs são sistemas essencialmente não lineares, multivariáveis e com acoplamentos dinâmicos (ARMSTRONG, 1986). Embora ainda bastante comuns e largamente utilizados na indústria, as técnicas de controle convencionais (PI, PID, etc.) não apresentam um desempenho satisfatório quando aplicados aos robôs industriais (LEWIS, 1993).

Desta forma, tornou-se necessário o desenvolvimento de novas técnicas e ferramentas que permitissem o projeto e a implementação de estruturas de controle cada vez mais eficientes. Assim, novas técnicas de análise no domínio da frequência foram desenvolvidas e as relações entre os domínios do tempo e da frequência foram mais bem compreendidas (OGATA, 1997). Enquanto a teoria de controle clássico propicia condições para estabilidade, a teoria de controle moderno além de garantir a estabilidade proporciona técnicas que otimizam os critérios de desempenho (OGATA, 2010). Além disso, o controle moderno trata de sistemas multivariáveis e traz uma simplificação ao projeto de sistemas de controle em razão de se basear no modelo de um sistema de controle real (DORF e BISHOP, 2009).

Neste trabalho a estrutura de controle de posição desenvolvida do tipo RASTRO aplicada ao robô RD5NT mostrou-se capaz de controlar de maneira satisfatória um sistema não linear.

Estrutura de controle

Será apresentada uma estrutura de controle desenvolvida a partir da proposta formulada por Luiz Carlos Campos Pedroza em 2001 (PEDROZA e PEDROZA, 2001). Essa estrutura de controle apresenta como principais características, a adaptabilidade e robustez, duas características importantes quando se trabalha com robôs. Além disto, é extremamente adequada para ser implementada em códigos computacionais.

Ela foi desenvolvida especificamente para sistema lineares, entretanto, este trabalho mostra que ela pode ser satisfatoriamente utilizada em sistemas não lineares, para isto, sendo necessário apenas linearizá-los por partes. A seguir, é apresentada sua fundamentação teórica, é descrito o problema de rastrear uma desejada trajetória utilizando-a e, finalmente, é proposto um algoritmo para a sua implementação numérica.

Fundamentação teórica

Seja o sistema não linear, de entrada e saída únicas (SISO), descrito pelas equações

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k \\ y_k = C_k^T x_k \end{cases} \quad (1)$$

onde: $u_k, y_k \in \mathbb{R}$; $A_k \in \mathbb{R}_{n \times n}$; $B_k, x_k \in \mathbb{R}_{n \times 1}$ e $C_k \in \mathbb{R}_{1 \times n}$

Esta equação pode ser reescrita, de forma simplificada, como

$$y_{k+1} = \alpha_k + \beta_k u_k \quad (2)$$

onde: $\alpha_k = C_{k+1}^T A_k x_k \in \mathbb{R}$
 $\beta_k = C_{k+1}^T B_k \in \mathbb{R}$

O problema de rastreamento pode ser descrito como: estimar os valores de α_k e β_k , respectivamente, α_k^E e β_k^E , e encontrar o valor de u_k que faça o sistema seguir uma trajetória desejada y_k^D . Como os parâmetros estimados são valores apenas aproximados, existirá um erro E_k no rastreamento da trajetória desejada, definido por

$$E_k = |y_k^D - y_k| \quad (3)$$

Para garantir que o sistema convirja para a trajetória desejada, é necessário garantir que

$$E_{k+1} < E_k \quad (4)$$

Manipulando as Equações (4) e (2), chega-se a

$$\frac{y_{k+1}^D - \alpha_k}{\beta_k} - \left| \frac{E_k}{\beta_k} \right| < u_k < \frac{y_{k+1}^D - \alpha_k}{\beta_k} + \left| \frac{E_k}{\beta_k} \right| \quad (5)$$

Analisando a Equação (5), conclui-se que fazendo o valor de u_k igual a $u_k^R = \frac{y_{k+1}^D - \alpha_k}{\beta_k}$, o erro de rastreamento E_{k+1} tende a zero. Entretanto, como os valores de α_k e β_k são desconhecidos, utilizam-se seus valores estimados α_k^E e β_k^E no cálculo de u_k , levando a um valor u_k^C diferente de u_k^R . Desta forma, tem-se que

$$u_k^C = \frac{y_{k+1}^D - \alpha_k^E}{\beta_k^E} \quad (6)$$

Observa-se, a partir da análise da Equação (5), que a Equação (4) será satisfeita por este valor u_k^C somente se

$$\left| u_k^C - u_k^R \right| < \left| \frac{E_k}{\beta_k} \right| \quad (7)$$

ou, em outras palavras,

$$\left| \frac{y_{k+1}^D - \alpha_k^E}{\beta_k^E} - \frac{y_{k+1}^D - \alpha_k}{\beta_k} \right| < \left| \frac{y_k^D - y_k}{\beta_k} \right| \quad (8)$$

Rastreamento da trajetória desejada

Dos resultados mostrados nas equações 1 a 8, observa-se que exceto na situação em que os erros de estimação são nulos, sempre haverá um erro não nulo no seguimento da trajetória quando se faz uso da Equação (6). Fazendo uso de ferramentas probabilísticas é possível definir os erros máximos de estimação como

$$\left| \alpha_k - \alpha_k^E \right| < \varepsilon \quad (9)$$

e
$$\left| \beta_k - \beta_k^E \right| < \gamma \quad (10)$$

onde: $\varepsilon, \gamma \in \mathbb{R}^+$.

Para garantir que sob essas condições $E_{k+1} < E_k$, propõe-se determinar, para cada interação k , um conjunto dos pontos possíveis de serem alcançados, y_{k+1}^P , e, a partir desse conjunto, definir o ponto alvo y_{k+1}^T . Todos os pontos y_{k+1}^P devem satisfazer a condição dada pela Equação (8) com os valores de α_k e β_k alterados para sua pior estimação. Então, esses pontos devem satisfazer as quatro equações a seguir,

$$E_{k+1} < E_k \quad (11)$$

resultantes das combinações de sinais de ε e γ .

Após escolher o valor para y_{k+1}^T , a entrada u_k^C que deverá ser aplicada ao sistema é calculada como

$$u_k^C = \frac{y_{k+1}^T - \alpha_k^E}{\beta_k^E} \quad (12)$$

trocando o valor de y_{k+1}^D por y_{k+1}^T na Equação (6).

Durante o cálculo de y_{k+1}^P , duas coisas podem ocorrer. A primeira é o ponto desejado y_{k+1}^D não pertencer ao conjunto dos pontos possíveis y_{k+1}^P . Neste caso, o ponto possível y_{k+1}^P mais próximo do ponto desejado y_{k+1}^D deverá ser escolhido como ponto alvo y_{k+1}^T . A segunda é que dependendo dos valores de ε , γ e E_k , o conjunto de Equações (11) pode não ter uma solução real. Neste caso, a parte imaginária deve ser desconsiderada e a parte real dessas soluções utilizadas como y_{k+1}^P . Nos dois casos, o u_k^C provavelmente fará que $E_{k+1} < E_k$.

Observa-se que a solução da Equação (11) exige um esforço computacional significativo. Propõe-se uma simplificação desse cálculo baseada no Teorema a seguir (PEDROZA e PEDROZA, 2001).

Teorema

$$\text{Se: } \begin{cases} \text{sinal}(\beta^{\min}) = \text{sinal}(\beta^{\text{MÁX}}) \\ |\beta^{\min}| < |\beta^{\text{MÁX}}| \end{cases}$$

(a) $\exists \beta^{\min}, \beta^{\text{MÁX}} \in \mathfrak{R}$ tal que $\beta_k, (\beta_k^E \pm \gamma) \in (\beta^{\min}, \beta^{\text{MÁX}})$ (13)

(b) $|y_{k+1}^P - \alpha_k^E| \geq \left| \frac{\beta_k^E}{\beta_k^E \pm 2\gamma} \right|$, com $|\beta_k^E| > 2\gamma$ (14)

então $\left| \frac{y_{k+1}^P - \alpha_k^E}{\beta^{\min}} \right| < \left| \frac{E_k}{\beta^{\text{MÁX}}} \right| \Rightarrow \left| \frac{y_{k+1}^P - \alpha_k^E}{\beta_k^E} - \frac{y_{k+1}^P - (\alpha_k^E \pm \varepsilon)}{(\beta_k^E \pm \gamma)} \right| < \left| \frac{E_k}{(\beta_k^E \pm \gamma)} \right|$ (15)

Na prática, como geralmente o sinal de β_k é conhecido *a priori* e ε e γ são geralmente pequenos, as condições necessárias (a) e (b) não são difíceis de serem satisfeitas.

Do Teorema e da Equação (11), tem-se que, para garantir $E_{k+1} < E_k$, basta obter a solução da inequação quadrática

$$\left(\frac{y_{k+1}^P - \alpha_k^E}{\beta^{\min}} \right)^2 < \left(\frac{E_k}{\beta^{\text{MÁX}}} \right)^2 \quad (16)$$

que exige um esforço computacional menor. Feito isto, basta escolher y_{k+1}^T conforme já explicado acima.

Para simplificar o algoritmo, nos casos em que a condição (b) não está satisfeita, calcula-se y_{k+1}^P pela Equação (16). Neste caso, o erro de rastreamento pode ser aumentado dando maior liberdade para satisfazer as condições do Teorema no próximo incremento do algoritmo.

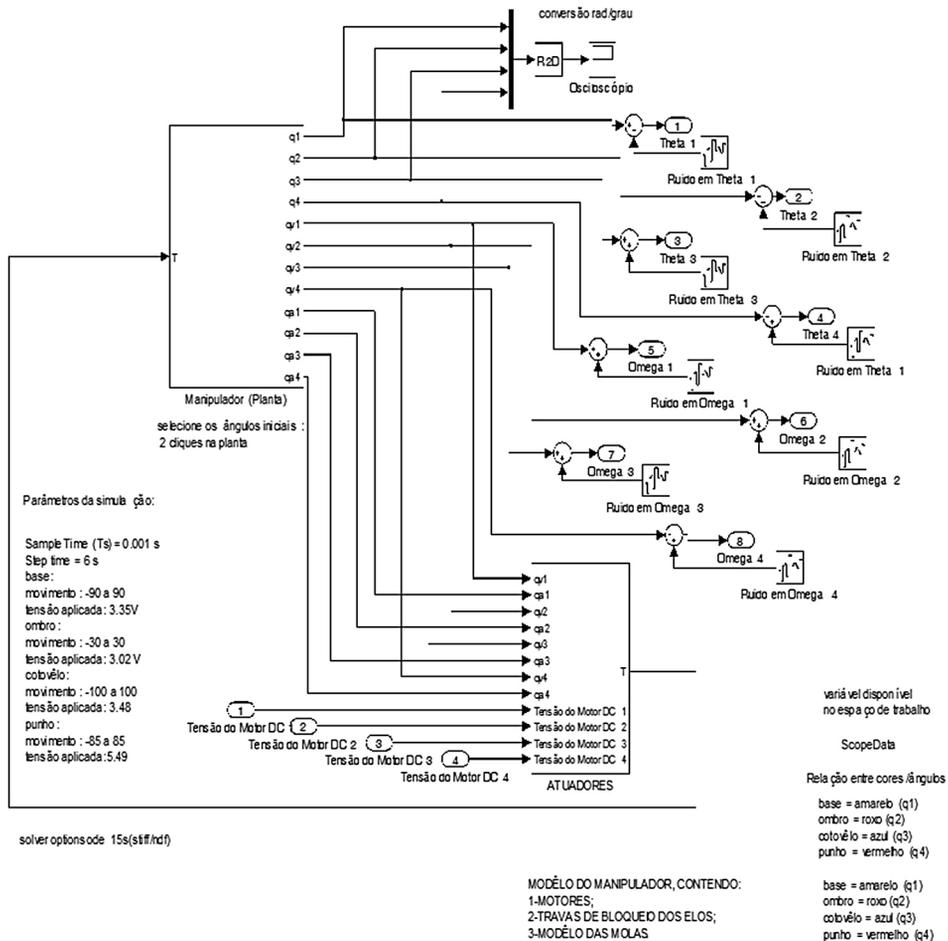
Implementação computacional do robô RD5NT

Para a implementação computacional do robô RD5NT, utilizou-se como ponto de partida a rotina apresentada em Aguiar (2008), desenvolvida em código SIMULINK®. Duas alterações se fizeram necessárias. A primeira está relacionada às condições iniciais das variáveis corrente, velocidade e posição das juntas. Foi necessário definir variáveis tais, que permitissem alterar as condições iniciais do modelo computacional para cada uma das n-iterações a serem executadas. Desta forma, foram criadas as variáveis *corrente_0i*, *omega_0i* e *theta_0i*, que são, respectivamente, a corrente, a

velocidade angular e a posição angular iniciais da *i*-ésima junta do robô RD5NT. A segunda alteração foi a introdução de geradores de ruídos, neste caso um ruído branco com amplitude máxima de 5% dos sinais de saída, velocidade e posição angular das juntas do robô, com o objetivo de simular os erros de aquisição de dados sempre presentes no mundo real.

Nas Figuras 1 a 4 são mostrados os principais blocos utilizados nessa implementação. A Figura 1 apresenta a implementação da estrutura geral do robô com a inserção dos geradores de ruído.

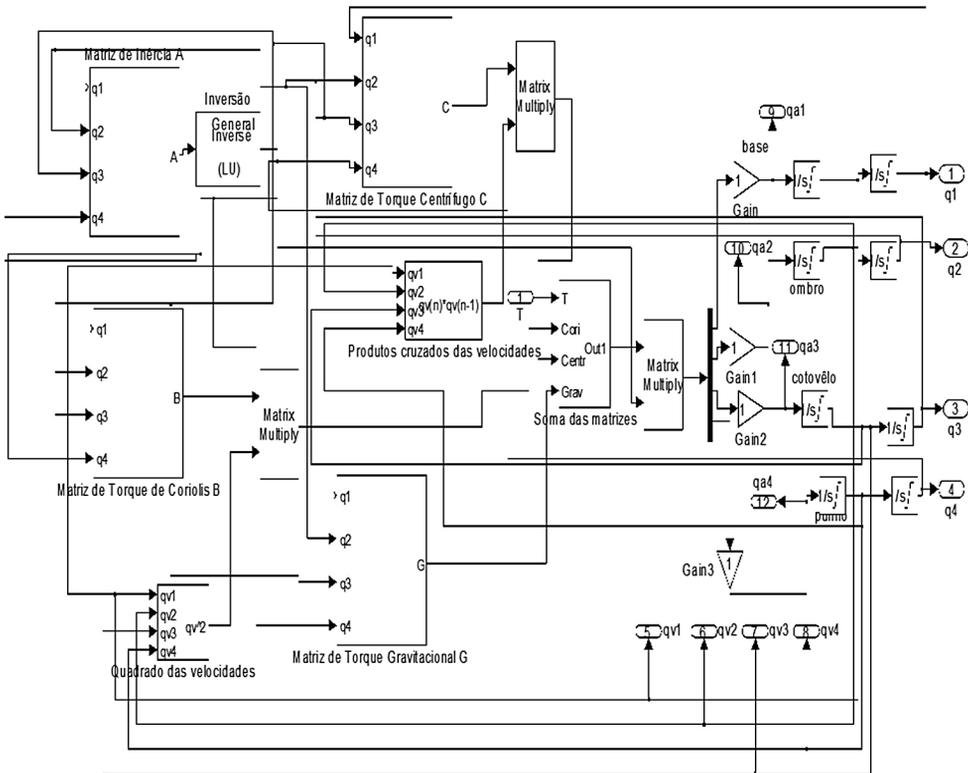
Figura 1: Diagrama em bloco da implementação do Robô RD5NT em código *SIMULINK*[®]



Fonte: Elaboração própria

A Figura 2 apresenta com mais detalhes o manipulador robótico, mostrando os blocos onde são implementados as matrizes de coeficiente de torque inercial, de coeficiente dos torques de Coriolis e centrífugo, o vetor de torque aplicado ao manipulador, vetor de torque gravitacional, vetor de aceleração, vetor das velocidades e o vetor dos ângulos das juntas.

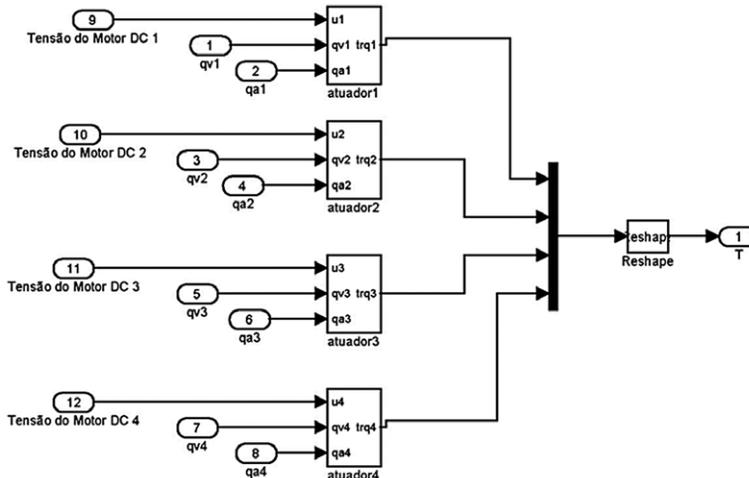
Figura 2: Implementação em código *SIMULINK*® da estrutura mecânica do Robô RD5NT



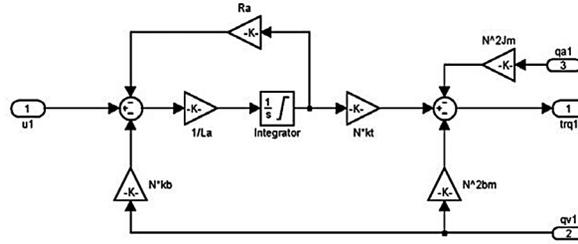
Fonte: Elaboração própria

A Figura 3 representa os atuadores das juntas do manipulador robótico e a Figura 4 a implementação dos motores DC.

Figura 3: Estrutura *SIMULINK*® do Bloco de Atuadores



Fonte: Elaboração própria

Figura 4: Implementação dos Motores DC em código *SIMULINK*®

Fonte: Elaboração própria

Gerador de trajetória

O passo seguinte foi o desenvolvimento do gerador de trajetória. Para isso algumas considerações foram observadas. O tempo total de simulação foi estabelecido em 20 s, independentemente da tarefa a ser realizada. As trajetórias de posição e velocidade foram divididas em três segmentos: inicialmente, durante um determinado intervalo de tempo, o robô permanece em repouso; em seguida, o robô realiza a tarefa deslocando-se da posição inicial para a posição final; finalmente, o robô permanece em repouso até completar o tempo total de simulação. Os tamanhos desses intervalos de tempo são independentes para cada uma de suas juntas.

Na geração da trajetória desejada para a posição da junta no segundo intervalo de tempo, isto é, quando a junta desloca-se da posição inicial para a posição final, determinou-se que essa trajetória obedecesse a uma função senoidal. Portanto, nesse mesmo intervalo de tempo, a trajetória desejada para a velocidade da junta deverá obedecer a uma função cossenoidal.

A partir dessas considerações, verifica-se que as trajetórias de posição e de velocidade são, respectivamente, descritas matematicamente pelas seguintes equações

$$\theta_D(t) = \begin{cases} \theta_I(t) = \theta_I, & \text{se } 0 \leq t < t_I \\ \frac{\theta_F - \theta_I}{2} \text{sen} \left(\frac{\pi}{t_F - t_I} t - \frac{\pi(t_F + t_I)}{2(t_F - t_I)} \right) + \frac{\theta_F + \theta_I}{2}, & \text{se } t_I \leq t < t_F \\ \theta_D(t) = \theta_F, & \text{se } t_F \leq t < 20 \end{cases} \quad (17)$$

e

$$\omega_D(t) = \begin{cases} \omega_D(t) = 0, & \text{se } 0 \leq t < t_I \\ \frac{\pi(\theta_F - \theta_I)}{2(t_F - t_I)} \cos \left(\frac{\pi}{t_F - t_I} t - \frac{\pi(t_F + t_I)}{2(t_F - t_I)} \right) & \text{se } t_I \leq t < t_F \\ \omega_D(t) = 0, & \text{se } t_F \leq t < 20 \end{cases} \quad (18)$$

onde: $\theta_D(t)$ é a posição desejada no instante t ;

$\omega_D(t)$ é a velocidade desejada no instante t ;

θ_I é a posição inicial da junta;

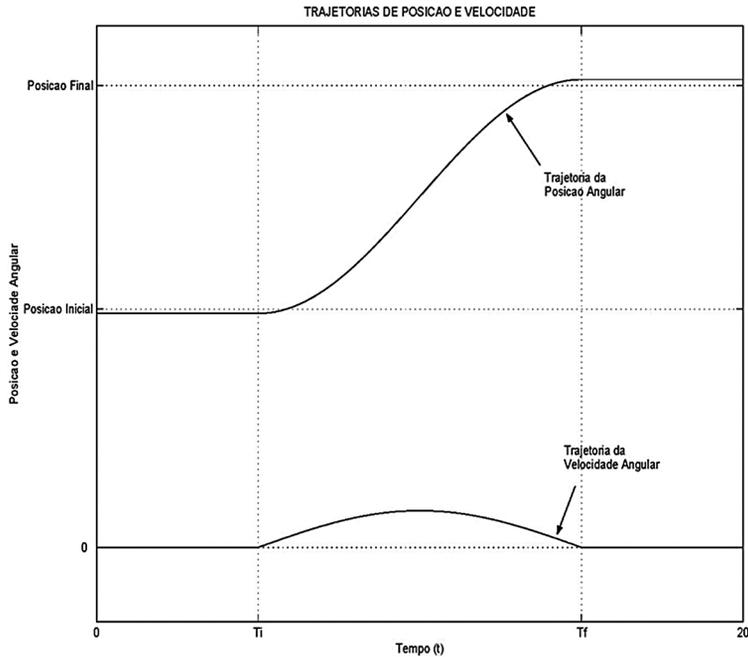
θ_F é a posição final da junta;

t_I é o instante quando a junta inicia seu movimento; e,

t_F é o instante quando a junta termina seu movimento

As curvas propostas para gerar as trajetórias de posição e velocidade estão mostradas na Figura 5.

Figura 5: Perfis da trajetórias de posição e velocidade desejadas



Fonte: Elaboração própria

Para implementação computacional do gerador de trajetórias foi desenvolvida a rotina *trajetoria.m* apresentada na Figura 6.

Figura 6: Rotina para implementação do gerador de trajetórias

```

ROTINA PARA GERAÇÃO DA TRAJETORIA DESEJADA
%
%
%      [thetad_k,omegad_k]=trajetoria(tk1,t_inicial,t_final,
%      theta_inicial, theta_final)
%
%      ESTA ROTINA É UTILIZADA PARA CALCULAR A POSIÇÃO E A
%      VELOCIDADE DESEJADA DA JUNTA NO INSTANTE k.
%ENTRADA:      tk1 - TEMPO NA ITERAÇÃO k+1
%              t_inicial - INSTANTE EM QUE A JUNTA INICIA O MOVIMENTO
%              t_final - INSTANTE EM QUE A JUNTA TERMINA SEU MOVIMENTO
%              theta_inicial - POSIÇÃO INICIAL DA JUNTA
%              theta_final - POSIÇÃO FINAL DA JUNTA
%
%SAÍDA:      thetad_k - POSIÇÃO ANGULAR DESEJADA NO A ITERAÇÃO k+1
%              omegad_k - VELOCIDADE ANGULAR DESEJADA NO A ITERAÇÃO k+1
%
function [thetad_k,omegad_k]=trajetoria(tk1,t_inicial,t_final,
theta_inicial, theta_final)

if tk1 <= t_inicial
thetad_k=theta_inicial;
omegad_k=0;
elseif tk1 >= t_final
thetad_k=theta_final;
omegad_k=0;
else
thetad_k=((theta_final-theta_inicial)/2)*sin((pi/((t_final-
t_inicial)))*tk1-((pi*((t_final+t_inicial)))/(2*(t_final-
t_inicial))))+((theta_final+theta_inicial)/2);
omegad_k=(pi*(theta_final-theta_inicial))/(2*(t_final-
t_inicial))*cos((pi/((t_final-t_inicial)))*tk1-
((pi*((t_final+t_inicial)))/(2*(t_final-t_inicial))));
end

```

Fonte: Elaboração própria

Identificador do modelo matemático

Em seguida, foi desenvolvida a rotina *identificacao.m* para estimar em tempo real os parâmetros do modelo linear discreto adotado pela estrutura de controle na determinação das excitações que deverão ser utilizadas para que o robô RD5NT obedeça à trajetória desejada.

Modelo matemático adotado

Para obtenção desse modelo discreto, considera-se que no intervalo de tempo k e $k+1$, cada uma das i juntas do manipulador pode ser desacoplada e que seu comportamento dinâmico pode ser adequadamente descrito pelo modelo linear discreto dado por:

$$\begin{cases} x_{k+1}^i = A_k^i x_k^i + B_k^i u_k^i \\ y_k^i = C_k^i x_k^i \end{cases} \quad (19)$$

onde: x_k^i é o vetor de estado da i -ésima junta no instante k ;
 u_k^i é o vetor de excitação da i -ésima junta no instante k ;
 y_k^i é o vetor de saída da i -ésima junta no instante k ;
 A_k^i é a matriz de transição de estado da i -ésima junta no instante k ;
 B_k^i é a matriz de excitação da i -ésima junta no instante k ;
 C_k^i é a matriz de saída da i -ésima junta no instante k ;

Sabendo que as posições θ_k^i e θ_{k-1}^i da junta i nos instantes k e $k-1$ são conhecidas através de medição direta e que através delas pode-se estimar a velocidade ω_k^i da junta, e supondo que cada junta tenha o comportamento aproximado de um sistema de segunda ordem, define-se o vetor de estado como

$$x_k^i = \begin{bmatrix} \omega_k^i & \theta_k^i \end{bmatrix}^T \quad (20)$$

Além disso, sabendo-se que o vetor de excitação é de dimensão 1 e que o vetor de saída é combinação linear dos estados do sistema, pode-se reescrever a Equação (19) como

$$\begin{Bmatrix} \omega_{k+1}^i \\ \theta_{k+1}^i \end{Bmatrix} = \begin{bmatrix} a_{11,k}^i & a_{12,k}^i \\ a_{21,k}^i & a_{22,k}^i \end{bmatrix} \begin{Bmatrix} \omega_k^i \\ \theta_k^i \end{Bmatrix} + \begin{bmatrix} b_{11,k}^i \\ b_{21,k}^i \end{bmatrix} \{u_k^i\} \quad (21)$$

e

$$\{y_k^i\} = \begin{bmatrix} c_{11,k}^i & c_{12,k}^i \end{bmatrix} \begin{Bmatrix} \omega_k^i \\ \theta_k^i \end{Bmatrix} \quad (22)$$

Observa-se que o manipulador em estudo apresenta um comportamento não linear, portanto deve-se ter em mente que o modelo proposto pelas Equações (21) e (22) só é válido como uma aproximação no intervalo de tempo k e $k+1$, ficando claro, desta forma, que as matrizes A_k^i , B_k^i e C_k^i variam à medida que há um incremento de k e, assim, devem ser estimadas toda vez que houver esse incremento.

Para a correta aplicação da estrutura de controle, faz-se necessário apresentar o modelo matemático discreto representado pelas Equações (21) e (22) sob a forma da Equação (2), isto é,

$$y_{k+1} = \alpha_k + \beta_k u_k \quad (2)$$

Portanto,

$$\begin{aligned} \{y_{k+1}^i\} &= \begin{bmatrix} c_{11,k+1}^i & c_{12,k+1}^i \end{bmatrix} \begin{bmatrix} a_{11,k}^i & a_{12,k}^i \\ a_{21,k}^i & a_{22,k}^i \end{bmatrix} \begin{Bmatrix} \omega_k^i \\ \theta_k^i \end{Bmatrix} + \begin{bmatrix} c_{11,k+1}^i & c_{12,k+1}^i \end{bmatrix} \begin{bmatrix} b_{11,k}^i \\ b_{21,k}^i \end{bmatrix} \{u_k^i\} \\ \{y_{k+1}^i\} &= (c_{11,k+1}^i a_{11,k}^i + c_{12,k+1}^i a_{21,k}^i) \{\omega_k^i\} + (c_{11,k+1}^i a_{12,k}^i + c_{12,k+1}^i a_{22,k}^i) \{\theta_k^i\} + (c_{11,k+1}^i b_{11,k}^i + c_{12,k+1}^i b_{21,k}^i) \{u_k^i\} \end{aligned} \quad (23)$$

$$\begin{aligned} \text{onde: } \alpha_{1,k}^i &= c_{11,k+1}^i a_{11,k}^i + c_{12,k+1}^i a_{21,k}^i \in \mathbb{R} \\ \alpha_{2,k}^i &= c_{11,k+1}^i a_{12,k}^i + c_{12,k+1}^i a_{22,k}^i \in \mathbb{R} \\ \alpha_k^i &= \alpha_{1,k}^i \omega_k^i + \alpha_{2,k}^i \theta_k^i \in \mathbb{R} \\ \beta_{1,k}^i &= c_{11,k+1}^i b_{11,k}^i + c_{12,k+1}^i b_{21,k}^i \in \mathbb{R} \end{aligned}$$

Estimando o valor dos parâmetros α_k^i e β_k^i

A seguir será apresentado o procedimento adotado para estimar o valor dos parâmetros α_k^i e β_k^i que serão utilizados pelo controlador. Para tal, pressupõe-se o conhecimento dos valores de ω_k^i , θ_k^i e u_k^i nas últimas m iterações, isto é, a matriz $k \times 3$ definida por:

$$X_{k-1} = \begin{bmatrix} \omega_{k-1}^i & \theta_{k-1}^i & u_{k-1}^i \\ \omega_{k-2}^i & \theta_{k-2}^i & u_{k-2}^i \\ \vdots & \vdots & \vdots \\ \omega_{k-l}^i & \theta_{k-l}^i & u_{k-l}^i \end{bmatrix} \quad (24)$$

Observe que, como não é possível garantir que as medidas de ω_k^i , θ_k^i e u_k^i nas últimas 3 iterações não levará a uma combinação linear entre as linhas da matriz X , deve-se trabalhar com $l > 3$.

A seguir, reescrevendo a Equação (23) na forma matricial, tem-se

$$\begin{aligned} \{y_k^i\} &= (c_{11,k}^i a_{11,k-1}^i + c_{12,k}^i a_{21,k-1}^i) \{\omega_{k-1}^i\} + (c_{11,k}^i a_{12,k-1}^i + c_{12,k}^i a_{22,k-1}^i) \{\theta_{k-1}^i\} + (c_{11,k}^i b_{11,k-1}^i + c_{12,k}^i b_{21,k-1}^i) \{u_{k-1}^i\} \\ y_k^i &= \alpha_{1,k-1}^i \omega_{k-1}^i + \alpha_{2,k-1}^i \theta_{k-1}^i + \beta_{k-1}^i u_{k-1}^i \\ y_k^i &= \begin{bmatrix} \omega_{k-1}^i & \theta_{k-1}^i & u_{k-1}^i \end{bmatrix} \begin{bmatrix} \alpha_{1,k-1}^i \\ \alpha_{2,k-1}^i \\ \beta_{k-1}^i \end{bmatrix} \end{aligned} \quad (25)$$

Agora, considerando-se que o modelo dinâmico varie de forma lenta e que o vetor Y de dimensão l seja conhecido, isto é,

$$Y_k = \begin{bmatrix} y_k^i \\ y_{k-1}^i \\ \vdots \\ y_{k-(l+1)}^i \end{bmatrix} \quad (26)$$

então, pode-se escrever que

$$\begin{Bmatrix} y_k^i \\ y_{k-1}^i \\ \vdots \\ y_{k-l}^i \end{Bmatrix} = \begin{bmatrix} \omega_{k-1}^i & \theta_{k-1}^i & u_{k-1}^i \\ \omega_{k-2}^i & \theta_{k-2}^i & u_{k-2}^i \\ \vdots & \vdots & \vdots \\ \omega_{k-l}^i & \theta_{k-l}^i & u_{k-l}^i \end{bmatrix} \begin{Bmatrix} \alpha_{1,k}^i \\ \alpha_{2,k}^i \\ \beta_k^i \end{Bmatrix} \quad (27)$$

Essa aproximação só é válida porque supôs que os parâmetros $\alpha_{1,k}^i$, $\alpha_{2,k}^i$ e $\beta_{1,k}^i$ variam muito lentamente e, que seus valores na iteração k são muito próximos dos valores das l últimas iterações. Desta forma,

$$\begin{aligned} Y_k &= X_{k-1} \Lambda_k \\ X_{k-1}^T Y_k &= X_{k-1}^T X_{k-1} \Lambda_k \\ \Lambda_k &= (X_{k-1}^T X_{k-1})^{-1} X_{k-1}^T Y_k \end{aligned} \quad (28)$$

onde: $\Lambda_k = \begin{Bmatrix} \alpha_{1,k}^i \\ \alpha_{2,k}^i \\ \beta_k^i \end{Bmatrix}$.

Para implementação computacional do identificador de parâmetros foi desenvolvida a rotina *identificacao.m* apresentada na Figura 7.

Figura 7: Rotina *MatLAB*[®] utilizada para identificar os parâmetros

```

ROTINA PARA IDENTIFICAÇÃO ON-LINE DO MODELO DISCRETO
%
%      [lambda]=identificacao(j,Yk,X1k)
%
%      ESTA ROTINA É UTILIZADA PARA IDENTIFICAR OS PARÂMETROS ALFA1,
ALFA2 E BETA DOS MODELOS DISCRETOS
%      Yk+!=ALFA1OMEGAK+ALFA2*THETAK+BETA*UK.
%
%      ENTRADA:j - JUNTA j
%                  Yk - VETOR CONTENDO AS ÚLTIMAS 1-AMOSTRAS DA
SAÍDA A SER UTILIZADA NA IDENTIFICAÇÃO
%                  x1k - VETOR CONTENDO AS ÚLTIMAS 1-AMOSTRAS DO
ESTADO (OMEGA E THETA) E DA EXCITAÇÃO
%
%      SAÍDA:      lambda- VETOR COM OS VALORES DA ALFA1, ALFA2 E
BETA ESTIMADOS PELA ROTINA
%
function [lambda]=identificacao(j,Yk,X1k)

lambda=inv(X1k'*X1k)*X1k'*Yk;

```

Fonte: Elaboração própria

Resultados da simulação

A seguir apresentamos algumas simulações, cujo objetivo é validar a estrutura de controle proposta para o robô RD5NT. Analisaremos como a estrutura de controle comporta-se em relação à frequência de amostragem.

Avaliando o comportamento do sistema em relação à frequência de amostragem

Um fator importante nas estruturas de controle diz respeito à frequência de amostragem necessária para garantir uma boa qualidade aos resultados. É fácil verificar que quanto maior for a taxa de amostragem, melhor serão os resultados obtidos. Entretanto, essa melhoria implica um maior esforço computacional. Portanto, torna-se necessário encontrar uma frequência de amostragem que otimize a relação entre qualidade dos resultados obtidos *versus* esforço computacional.

A seguir, são apresentados os resultados obtidos em simulações realizadas com o objetivo de verificar como a variação da taxa de amostragem influencia na qualidade dos resultados obtidos. Realizamos duas simulações para as frequências de amostragens de 10 Hz e 50 Hz mantendo-se inalterados os demais dados de entrada, que são apresentados na Tabela 1.

Tabela 1: Dados utilizados na avaliação da influência da frequência de amostragem

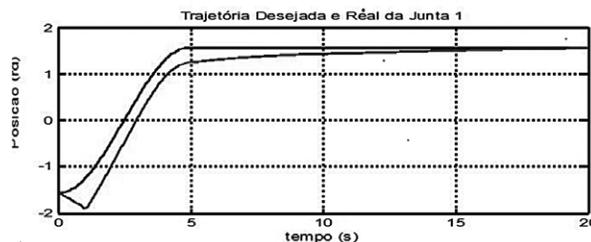
	BASE	OMBRO	COTOVELO	PUNHO
θ_I (rd)	$-\pi/2$	$\pi/2$	$-\pi/2$	$-\pi/2$
θ_F (rd)	$\pi/2$	$-\pi/2$	$\pi/2$	$\pi/2$
t_I (s)	0	5	10	15
t_F (s)	5	10	15	20

Fonte: Elaboração própria

Taxa de Amostragem de 10 Hz

Para uma frequência de amostragem de 10 Hz, foram obtidos os resultados expostos nas figuras 8 a 11.

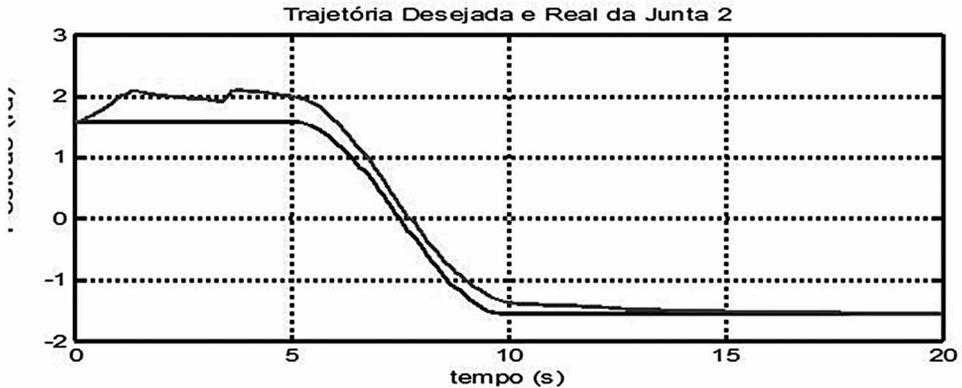
Figura 8: Gráfico trajetória desejada e trajetória real da base do robô



Fonte: Elaboração própria

Desse gráfico observa-se que a junta não segue com precisão adequada a trajetória desejada. A maior diferença no início da trajetória está relacionada ao início do processo de identificação. Deve-se chamar atenção para o fato de que as primeiras dez amostras utilizadas pelo identificador são valores arbitrados aleatoriamente.

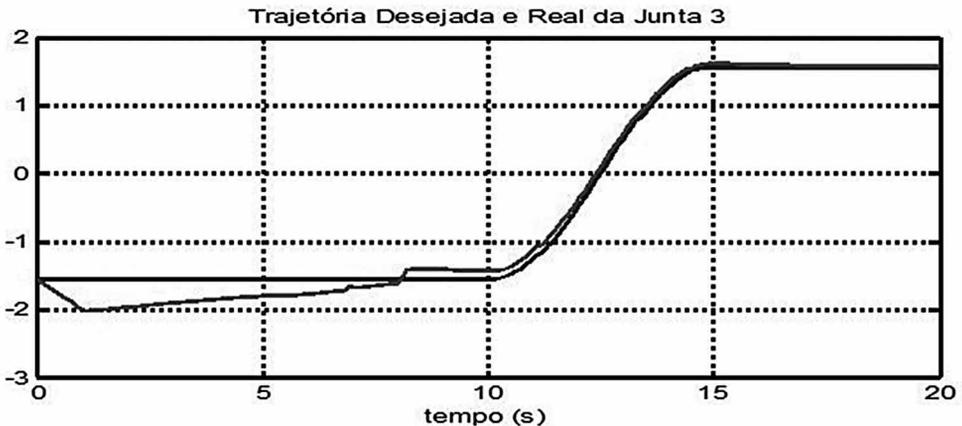
Figura 9: Gráfico trajetória desejada e trajetória real do ombro do robô



Fonte: Elaboração própria

Analisando-se o gráfico, verifica-se que inicialmente a junta não consegue rastrear a trajetória desejada. Observa-se que, a partir de cinco segundos, a junta tende para a trajetória; mas não consegue atingi-la.

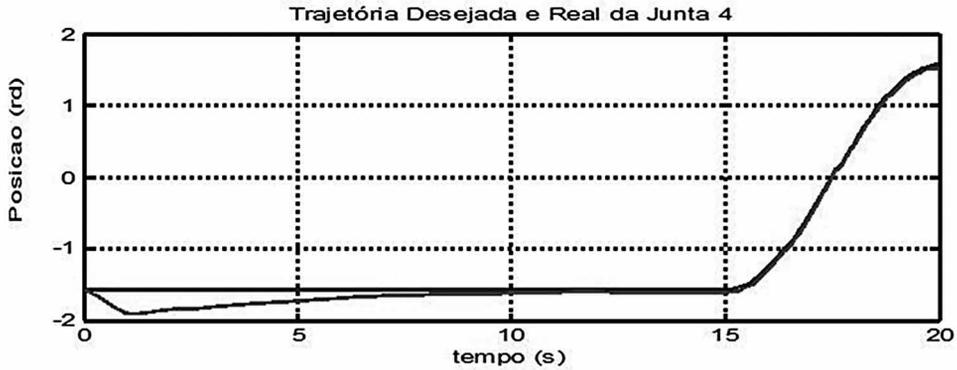
Figura 10: Gráfico trajetória desejada e trajetória real do cotovelo do robô



Fonte: Elaboração própria

Observa-se que, nos primeiros sete segundos, de forma idêntica ao que acontece com a base, o cotovelo não acompanha a trajetória desejada. Isso se dá por problemas no procedimento adotado para identificação do modelo. Finalmente, observa-se que, a partir do instante de onze segundos, o cotovelo acompanha adequadamente a trajetória desejada.

Figura 11: Gráfico trajetória desejada e trajetória real do punho do robô



Fonte: Elaboração própria

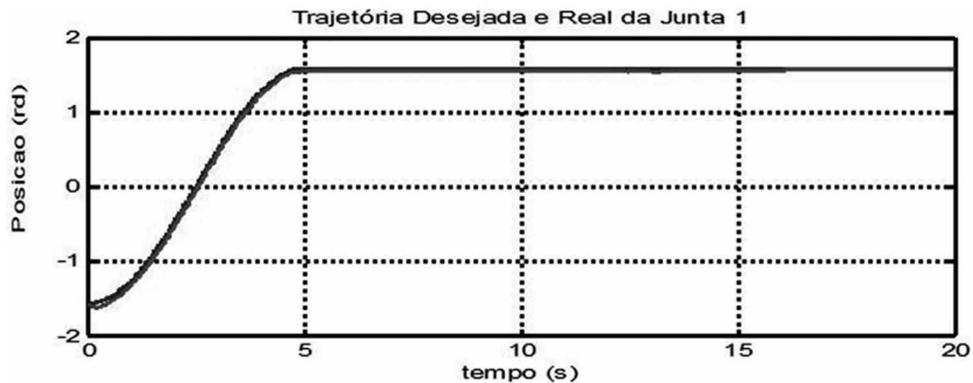
Observa-se que o punho tem comportamento similar ao comportamento observado no cotovelo. Observa-se ainda que, após o instante dez segundos, o punho acompanha adequadamente a trajetória desejada.

Finalmente, dessas experiências, conclui-se que a estrutura proposta, com a frequência de amostragem de 10 Hz, não conseguiu controlar adequadamente a base e o ombro.

Taxa de Amostragem de 50 Hz

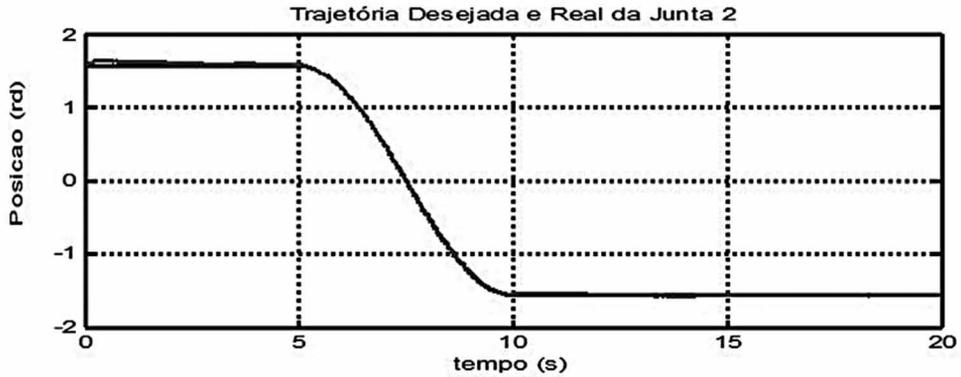
Para uma frequência de amostragem de 50 Hz, foram obtidos os resultados expostos nas figuras 12 a 15.

Figura 12: Gráfico trajetória desejada e trajetória real da base do robô



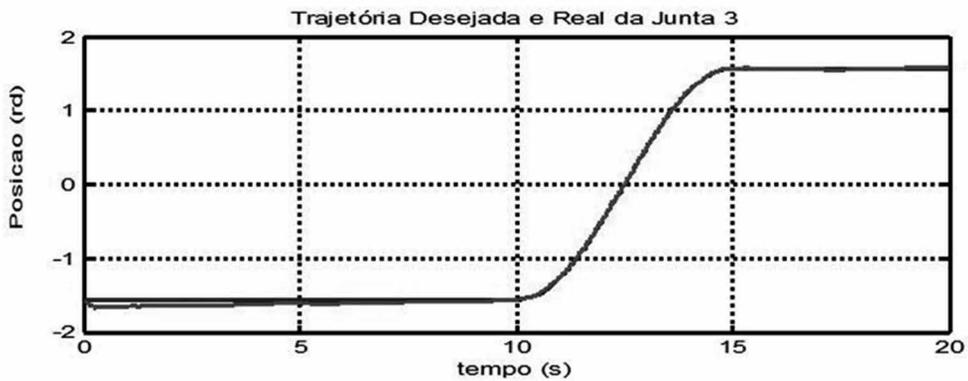
Fonte: Elaboração própria

Figura 13: Gráfico trajetória desejada e trajetória real do ombro do robô



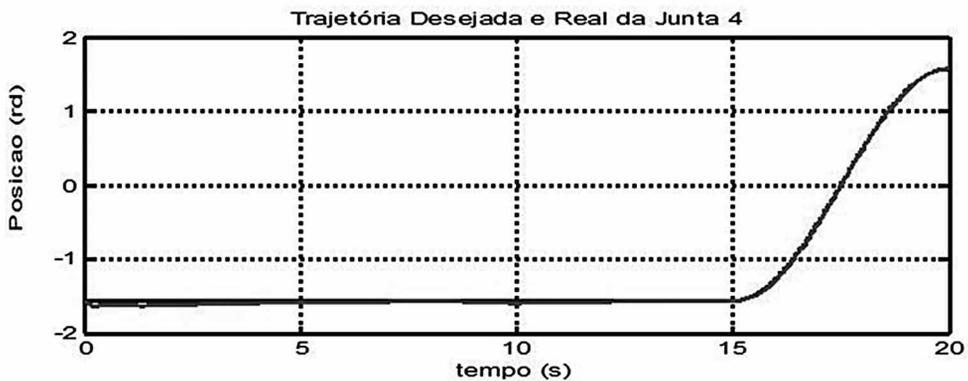
Fonte: Elaboração própria

Figura 14: Gráfico trajetória desejada e trajetória real do cotovelo do robô



Fonte: Elaboração própria

Figura 15: Gráfico trajetória desejada e trajetória real do punho do robô



Fonte: Elaboração própria

Da análise dos gráficos das figuras 12 a 15, observa-se que as juntas conseguiram acompanhar a trajetória desejada satisfatoriamente com pequenos erros, cujos valores são apresentados na Tabela 2.

Tabela 2: Tabela dos erros de posição (em radianos) para frequência de 50 Hz

	BASE	OMBRO	COTOVELO	PUNHO
ERRO MÉDIO	0,0013	-0,0011	0,0002	0,0007
DESVIO PADRÃO	0,0320	0,0331	0,0322	0,0311
ERRO MÁXIMO	0,09	0,038	0,11	0,07
ERRO MÍNIMO	-0,01	-0,055	-0,028	-0,028

Fonte: Elaboração própria

Conclusões

Das diversas simulações executadas, verificou-se que a estrutura de controle proposta apresentou resultados bastante animadores quanto à tarefa de seguir uma trajetória segundo um perfil de posição e de velocidade, como mostra a Tabela 2, indicando ser bastante viável sua aplicação a esse tipo de problema.

Simulações com frequência de amostragem maior que 50 Hz foram realizadas; mas, embora obtidos resultados ainda mais precisos, implicaram maiores esforços computacionais.

Todas as simulações foram executadas com a inserção de ruídos, e foi observado que a estrutura de controle respondeu adequadamente quanto a esse aspecto.

Finalmente, deve-se esclarecer que o comportamento inadequado das juntas no início da simulação está relacionado ao procedimento adotado para identificar os parâmetros do modelo discreto.

Referências

- AGUIAR, E. M. *Desenvolvimento do Modelo Dinâmico do Manipulador Robótico RD5NT*. 2008. Dissertação (Mestrado) – Universidade Federal Fluminense, Niterói. 2008.
- ARMSTRONG, B. et al. The Explicit Dynamic Model And Inertial Parameters of The Puma 560 Arm. Proc. IEEE Conf. Robot. Autom., New York, v. 2, 1986.
- DORF, R.C.; BISHOP, R. H. *Sistemas de controle modernos*. Rio de Janeiro: Livros Técnicos e Científicos, 2009.
- LEWIS, F. L. et al. *Control of Robot Manipulators*. New York: Macmillan, 1993.
- OGATA, K. *Discrete-Time Control Systems*. 2. ed. Upper Saddle River: Prentice-Hall, 1995.
- OGATA, K. *Engenharia de Controle Moderno*. 5ª ed. São Paulo: Prentice-Hall, 2010.
- O. MAYR. *The Origins of Feedback Control*. Cambridge, Mass.: MIT Press, 1970.

PEDROZA, L. C. ; PEDROZA, B. C. . A Study in Discrete-Time Robust Control. In: THE IASTED INTERNATIONAL CONFERENCE IN MODELING, IDENTIFICATION AND CONTROL, 2001, Innsbruck. *Proceedings ...*, 2001. v. 1. p. 12-16.

Artigo recebido em: 19 nov. 2013

Aceito para publicação em: 08 jul. 2014