

ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL DA  
TESE DEFENDIDA POR JOSÉ RIBEIRO  
BRANDÃO NETO E APROVADA PELA  
COMISSÃO JULGADORA EM 11 / 03 / 1999.

  
ORIENTADOR

UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA MECÂNICA

# **Controle Auto-Sintonizável de Sistemas Rotativos Submetidos a Demanda de Torque Variável**

Autor: José Ribeiro Brandão Neto

Orientador: Eurípedes Guilherme Oliveira Nóbrega

**UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA MECÂNICA  
DEPARTAMENTO DE MECÂNICA COMPUTACIONAL**

**DISSERTAÇÃO DE MESTRADO**

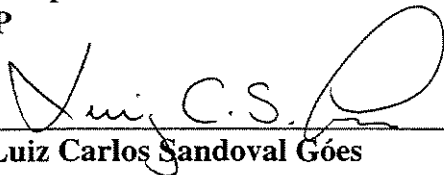
**Controle Auto-Sintonizável de Sistemas  
Rotativos Submetidos a Demanda de Torque  
Variável**

Autor: **José Ribeiro Brandão Neto**

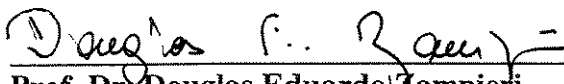
Orientador: **Eurípedes Guilherme Oliveira Nóbrega**



**Prof. Dr. Eurípedes Guilherme Oliveira Nóbrega, Presidente  
UNICAMP**



**Prof. Dr. Luiz Carlos Sandoval Góes  
ITA**



**Prof. Dr. Douglas Eduardo Zampieri  
UNICAMP**

Campinas, 11 de março de 1999

9913820



**UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA MECÂNICA  
DEPARTAMENTO DE MECÂNICA COMPUTACIONAL**

# **Controle Auto-Sintonizável de Sistemas Rotativos Submetidos a Demanda de Torque Variável**

**Autor: José Ribeiro Brandão Neto**

**Orientador: Eurípedes Guilherme Oliveira Nóbrega**

Curso: Engenharia Mecânica

Área de Concentração: Mecânica dos Sólidos e Projeto Mecânico

Dissertação de mestrado apresentada à comissão de Pós Graduação da Faculdade de Engenharia Mecânica, como requisito para a obtenção do título de Mestre em Engenharia Mecânica.

Campinas, 1999  
S.P. – Brasil

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

B733c

Brandão Neto, José Ribeiro

Controle auto-sintonizável de sistemas rotativos submetidos a demanda de torque variável. / José Ribeiro Brandão Neto .-- Campinas, SP: [s.n.], 1999.

Orientador: Eurípedes Guilherme Oliveira Nóbrega  
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Sistemas de controle ajustável. 2. Sistemas de tempo discreto. 3. Velocidade. 4. Controle em tempo real. 5. Mínimos quadrados. I. Nóbrega, Eurípedes Guilherme Oliveira. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

## **Dedicatória**

Para Adriana, companheira e amiga de todas as horas  
de todos os dias  
de todos os meses  
de todos os anos.

## **Agradecimentos**

A todos aqueles que direta ou indiretamente contribuíram para a realização deste trabalho:

Adriana, por sua aplicação na leitura e revisão do texto, pela presença roubada, e pelos fins-de semana na FEM.

Minha família, pela minha formação.

Meu orientador, pela oportunidade de enveredar por novos caminhos.

Caros colegas da FEM, pelas discussões e cafés, muitos cafés.

Demais membros do Departamento de Mecânica Computacional.

*“...eles passarão, e eu, passarinho.”*  
*Mário Quintana*

## Resumo

BRANDÃO NETO, José Ribeiro, *Controle Auto-Sintonizável de Sistemas Rotativos Submetidos a Demanda de Torque Variável*, Campinas, Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 1999. 104 p. Dissertação (Mestrado).

Encontra-se larga aplicação para a teoria de sistemas lineares com parâmetros concentrados e invariantes no tempo quando se estudam métodos de controle para sistemas mecânicos rotativos. Em aplicações para as quais existe um regime de demanda de torque variável, é preciso levar em conta o caráter não estático de alguns parâmetros do sistema, para que se possa projetar controladores que tenham seus desempenhos reais compatíveis com as características de projeto. Fazer o controle da velocidade de rotação em um caso experimental que apresenta um regime de demanda de torque variável é o objeto de estudo utilizado na elaboração deste trabalho. A partir de uma bancada mecânica rotativa, são apresentadas todas as etapas da identificação que serviu como base para o projeto de três tipos de controladores: tacométrico, polinomial e polinomial auto-sintonizável. O projeto dos controladores tinha como objetivo a obtenção de uma velocidade de rotação de saída estabilizada, sendo a avaliação do desempenho feita face a variações na demanda de torque aplicadas aos sistemas controlados. Em especial, deu-se atenção à implementação do método polinomial auto-sintonizável. A estratégia adaptativa resultou na atuação dentro dos limites desejados, enquanto o sistema era submetido a uma demanda de torque variável. Os resultados experimentais obtidos com um sistema reduzido, somados às simulações com o sistema completo, permitem dizer que controladores auto-sintonizáveis são uma melhoria viável para controle de sistemas rotativos.

### *Palavras Chave*

Sistemas de controle ajustável; controle de rotação em tempo discreto; controle em tempo real; mínimos quadrados.

## Abstract

BRANDÃO NETO, José Ribeiro, *Self-Tuning Control for Rotating Systems Subject to Torque Load Fluctuations*, Campinas, Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 1999. 104 p. Dissertação (Mestrado).

The theory for time-invariant lumped parameter linear systems (LTI) stands as the basis for control methods applied to rotating mechanical systems. There are some situations, however, for which there is a variable torque load, as in belt conveyors or rolling mills, for example. For those cases it is necessary to design controllers which take into account the non-stationary features of the system, for the real controller performance to cope with the design characteristics. Although feasible in theory and simulation, in the implementation of control schemes which deal with non-LTI systems there are some other constraints involved, as, for instance, the gap between the necessary and the available instrumentation for the real controller to perform as desired. For an experimental case consisting of a rotating system subject to torque load fluctuations, it is presented in this work three rotation regulation control schemes: tachometric, polynomial and self-tuning polynomial. Special attention was given to self-tuning polynomial controllers. The basis for the controller design is the identification of mechanical setup, also developed in the text. The control goal was to regulate the output rotation speed, and the performance index was chosen to be the controlled system output when subjected to torque load variation. The result of the implemented adaptive scheme shows the regulator action to keep the output within the desired levels as the load varies. From all results, including the ones for a reduced experimental system, it is possible to foresee the self-tuning scheme as a feasible improvement for rotating systems' controllers.

### *Key Words*

Self-tuning control; discrete time rotating systems control; real time control; least squares.

## Sumário

|                                                              |     |
|--------------------------------------------------------------|-----|
| <b>1. INTRODUÇÃO</b>                                         |     |
| 1.1. Objetivos                                               | 1   |
| 1.2. Aspectos de Sistemas Rotativos                          | 2   |
| 1.3. Conteúdo do Texto                                       | 8   |
| <b>2. CONTROLE DE SISTEMAS ROTATIVOS</b>                     |     |
| 2.1. Controle Tacométrico                                    | 9   |
| 2.2. Controle Polinomial                                     | 10  |
| 2.3. Identificação Recursiva                                 | 13  |
| 2.4. Controle Polinomial Auto-sintonizável                   | 21  |
| <b>3. IDENTIFICAÇÃO DO SISTEMA</b>                           |     |
| 3.1. Modelos de Primeira Ordem                               | 24  |
| 3.2. Modelos de Segunda Ordem                                | 30  |
| 3.3. Modelo de Sétima Ordem                                  | 42  |
| <b>4. IMPLEMENTAÇÃO DOS MÉTODOS DE CONTROLE</b>              |     |
| 4.1. Instrumentação e Conexão com o Microcomputador          | 52  |
| 4.2. Controle Tacométrico                                    | 54  |
| 4.3. Controle Polinomial                                     | 57  |
| 4.4. Controle Auto-sintonizável                              | 59  |
| <b>5. RESULTADOS E CONCLUSÕES</b>                            |     |
| 5.1. Identificação do Sistema                                | 67  |
| 5.2. Métodos de Controle                                     | 69  |
| 5.3. Conclusões                                              | 70  |
| <b>BIBLIOGRAFIA</b>                                          | 72  |
| <b>APÊNDICE A Instrumentação</b>                             | 74  |
| <b>APÊNDICE B Programa de aquisição e controle</b>           | 77  |
| <b>APÊNDICE C Lugar das raízes para controle tacométrico</b> | 97  |
| <b>APÊNDICE D Controle polinomial com modelo estimado</b>    | 100 |
| <b>APÊNDICE E Simplex não linear</b>                         | 106 |

## Lista de Figuras

|        |                                                                             |    |
|--------|-----------------------------------------------------------------------------|----|
| 1.1-1  | Sistema mecânico experimental.                                              | 2  |
| 2.1-1  | Controlador tacométrico implementado.                                       | 9  |
| 2.2-2  | Sistema com controle polinomial.                                            | 10 |
| 2.4-1  | Diagrama do Sistema com Controlador Adaptativo.                             | 22 |
| 3.1-1  | Motor e parâmetros.                                                         | 25 |
| 3.1-2  | Teste de parada do motor em torno da rotação de regime desejada.            | 26 |
| 3.1-3  | Rotor e seus parâmetros.                                                    | 27 |
| 3.1-4  | Teste de parada do rotor em torno da rotação de regime desejada.            | 27 |
| 3.1-5  | Sistema com três graus de liberdade.                                        | 28 |
| 3.1-6  | Resposta do sistema simplificado a perturbação aleatória até 20Hz.          | 29 |
| 3.2-1  | Sistema formado por um motor, uma correia e um rotor.                       | 31 |
| 3.2-2  | Resposta livre do conjunto motor, correia e rotor, <i>medida no rotor</i> . | 35 |
| 3.2-3  | Respostas livres do conjunto, medidas no motor e no rotor.                  | 35 |
| 3.2-4  | Respostas livres do conjunto retirando-se médias móveis.                    | 36 |
| 3.2-5  | Auto-espectros das respostas livres sem média móvel.                        | 36 |
| 3.2-6  | Sistema formado por dois rotores e um eixo.                                 | 37 |
| 3.2-7  | Resposta livre de deformação no eixo – teste 1.                             | 40 |
| 3.2-8  | Resposta livre de deformação no eixo – teste 2.                             | 40 |
| 3.2-9  | Resposta livre de deformação no eixo – teste 3.                             | 40 |
| 3.2-10 | Resposta livre de deformação no eixo – teste 4.                             | 41 |
| 3.2-11 | Resposta livre de deformação no eixo – teste 5.                             | 41 |
| 3.3-1  | Sistema formado por dois motores, dois rotores, duas correias e um eixo.    | 42 |
| 3.3-2  | FRF modelada para $G_{ve}=V_e/I$ .                                          | 46 |
| 3.3-3  | FRF modelada para $G_{sg}/I$ .                                              | 47 |
| 3.3-4  | Localização de pólos e zeros modelados e estimados.                         | 48 |
| 3.3-5  | FRF estimada para $\hat{G}_{ve} = \frac{V_e}{I}$ .                          | 49 |
| 3.3-6  | Localização de pólos e zeros de $\hat{G}_{sg} = \frac{SG}{I}$ .             | 50 |
| 3.3-7  | FRF estimada para $\hat{G}_{sg} = \frac{SG}{I}$ .                           | 51 |
| 4.1-1  | Fluxo de sinais na bancada experimental.                                    | 52 |
| 4.1-2  | Diagramas de Bode para as fontes de corrente.                               | 53 |
| 4.2-1  | Exemplo de torque perturbador aplicado ao rotor.                            | 54 |
| 4.2-2  | Controlador tacométrico sujeito a perturbação.                              | 55 |
| 4.2-3  | Pólos e zeros do modelo paramétrico e do modelo estimado.                   | 55 |
| 4.2-4  | Root locus para controle tacométrico.                                       | 56 |
| 4.2-5  | Respostas em frequência: sistema controlado e não controlado                | 56 |
| 4.3-1  | Resposta a controlador com modelo de referência.                            | 57 |
| 4.3-2  | Resposta ao degrau unitário.                                                | 59 |
| 4.4-1  | Evolução temporal dos parâmetros estimados.                                 | 60 |
| 4.4-2  | Detalhes da evolução temporal dos parâmetros estimados.                     | 60 |

|       |                                                                        |     |
|-------|------------------------------------------------------------------------|-----|
| 4.4-3 | Resposta a controlador com modelo de referência.                       | 61  |
| 4.4-4 | Varição simulada do termo de ordem zero do numerador.                  | 63  |
| 4.4-5 | Evolução temporal dos parâmetros do numerador da planta.               | 64  |
| 4.4-6 | Evolução temporal dos parâmetros do denominador da planta.             | 65  |
| 4.4-7 | Módulo do traço de P.                                                  | 65  |
| 4.4-8 | Simulação de resposta controlada para o sistema.                       | 66  |
| A-1   | Circuito de leitura de deformação no eixo através de extensômetro.     | 74  |
| A-2   | Circuito para fornecimento e leitura de corrente de acionamento.       | 75  |
| A-3   | Circuito para leitura e fornecimento de corrente de perturbação.       | 76  |
| E-1   | Resultado da minimização por simplex refletivo na função $z=x^2+y^2$ . | 107 |

## **Lista de Tabelas**

|                                      |    |
|--------------------------------------|----|
| 3.1-1 Tensão nos terminais do motor. | 25 |
|--------------------------------------|----|

## Nomenclatura

### Letras Latinas

|            |                                                                        |                      |
|------------|------------------------------------------------------------------------|----------------------|
| $K$        | – ganho de realimentação do controlador tacométrico.                   | $[A.s.rad^{-1}]$     |
| $K_d$      | – ganho direto do controlador tacométrico.                             | $[A.s.rad^{-1}]$     |
| $G_c$      | – função de transferência da planta controlada.                        | $[rad.s^{-1}A^{-1}]$ |
| $A$        | – polinômio do denominador da planta.                                  |                      |
| $B$        | – polinômio do numerador da planta.                                    |                      |
| $z$        | – variável da transformada Z.                                          |                      |
| $R$        | – polinômio do controlador polinomial.                                 |                      |
| $S$        | – polinômio do controlador polinomial.                                 |                      |
| $T$        | – polinômio do controlador polinomial.                                 |                      |
| $G_m$      | – função de transferência da planta de referência.                     |                      |
| $A_m$      | – polinômio do denominador da planta de referência.                    |                      |
| $B_m$      | – polinômio do numerador da planta de referência.                      |                      |
| $B^+$      | – polinômio com zeros da planta internos ao círculo unitário.          |                      |
| $B^-$      | – polinômio com zeros da planta externos ao círculo unitário.          |                      |
| $B_m'$     | – polinômio componente do numerador da planta de referência estendida. |                      |
| $R'$       | – componente do denominador polinômio estendido do controlador.        |                      |
| $A_o$      | – polinômio observador.                                                |                      |
| $q$        | – operador avanço.                                                     |                      |
| $x_i$      | – variáveis de estado do modelo ARMA.                                  |                      |
| $a_i$      | – coeficientes do denominador do modelo ARMA.                          |                      |
| $b_i$      | – coeficientes do numerador do modelo ARMA.                            |                      |
| $T$        | – intervalo de amostragem.                                             | $[s]$                |
| $k$        | – índice de contagem de tempo.                                         |                      |
| $y_k$      | – variável de saída do modelo ARMA, num instante $kT$ .                | $[rad.s^{-1}]$       |
| $u_k$      | – variável de entrada do modelo ARMA, num instante $kT$ .              | $[A]$                |
| $C$        | – matriz de saída do modelo ARMA.                                      |                      |
| $f$        | – forma genérica da equação de estados do modelo ARMA.                 |                      |
| $e$        | – erro de equação do modelo ARMA.                                      | $[rad.s^{-1}]$       |
| $J$        | – Função erro quadrático do modelo ARMA.                               | $[rad^2s^{-2}]$      |
| $Y$        | – vetor de valores de saída.                                           | $[rad.s^{-1}]$       |
| $J_\theta$ | – Derivada da função erro quadrático com relação aos parâmetros.       | $[rad^2s^{-2}]$      |
| $w$        | – fator de ponderação do método dos mínimos quadrados.                 |                      |
| $W$        | – matriz de ponderação do método dos mínimos quadrados.                |                      |
| $P$        | – inversa da matriz de covariância do estimador recursivo              |                      |
| $\hat{R}$  | – polinômio controlador calculado a partir dos parâmetros estimados.   |                      |
| $\hat{S}$  | – polinômio controlador calculado a partir dos parâmetros estimados.   |                      |

|           |                                                                              |                          |
|-----------|------------------------------------------------------------------------------|--------------------------|
| $\hat{T}$ | - polinômio controlador calculado a partir dos parâmetros estimados.         |                          |
| $\hat{A}$ | - denominador estimado para a planta.                                        |                          |
| $\hat{B}$ | - numerador estimado para a planta.                                          |                          |
| $K_e$     | - constante elétrica do motor.                                               | [V.s.rad <sup>-1</sup> ] |
| $K_t$     | - constante de torque do motor.                                              | [N.m.A <sup>-1</sup> ]   |
| $V_m$     | - diferença de potencial nos terminais do motor com terminais desconectados. | [V]                      |
| $i$       | - corrente elétrica circulante pelo motor de acionamento.                    | [A]                      |
| $J_m$     | - momento de inércia do motor.                                               | [kg.m <sup>2</sup> ]     |
| $a_m$     | - constante de amortecimento do motor.                                       | [s <sup>-1</sup> ]       |
| $c_m$     | - amortecimento do motor.                                                    | [N.m.s]                  |
| $t$       | - tempo.                                                                     | [s]                      |
| $J_R$     | - momento de inércia do rotor.                                               | [kg.m <sup>2</sup> ]     |
| $a_R$     | - constante de amortecimento do rotor.                                       | [s <sup>-1</sup> ]       |
| $c_R$     | - amortecimento do rotor.                                                    | [N.m.s]                  |
| $J_{eq}$  | - momento de inércia equivalente no rotor.                                   | [kg.m <sup>2</sup> ]     |
| $c_{eq}$  | - amortecimento equivalente no rotor.                                        | [N.m.s]                  |
| $K_{ieq}$ | - constante de torque equivalente no rotor.                                  | [N.m.A <sup>-1</sup> ]   |
| $i_p$     | - corrente no motor de perturbação                                           | [A]                      |
| $R_m$     | - raio da polia do motor.                                                    | [m]                      |
| $R_R$     | - raio do rotor.                                                             | [m]                      |
| $k_c$     | - constante elástica linear da correia.                                      | [N.m <sup>-1</sup> ]     |
| $C_c$     | - coeficiente de atrito viscoso introduzido pela correia.                    | [N.m.s]                  |
| $K_{cR}$  | - constante elástica torcional equivalente da correia atuando no motor.      | [Nm]                     |
| $K_{cm}$  | - constante elástica torcional equivalente da correia atuando no rotor.      | [Nm]                     |
| $c_{cm}$  | - coeficiente de atrito equivalente da correia atuando no motor.             | [N.m.s]                  |
| $c_{cR}$  | - coeficiente de atrito equivalente da correia atuando no rotor.             | [N.m.s]                  |
| $n$       | - relação de transmissão entre motor e rotor.                                |                          |
| $\bar{x}$ | - vetor de estados do sistema modelado.                                      |                          |
| $A$       | - matriz de estados do sistema modelado.                                     |                          |
| $B$       | - vetor de entrada do sistema modelado.                                      |                          |
| $J_{mR}$  | - momento de inércia equivalente para o conjunto motor e rotor 1.            | [kg.m <sup>2</sup> ]     |
| $c_{mR}$  | - amortecimento equivalente do conjunto motor e rotor 1.                     | [N.m.s]                  |
| $a_{mR}$  | - constante de amortecimento do conjunto motor e rotor 1.                    | [s <sup>-1</sup> ]       |
| $f_{ntm}$ | - frequência natural do conjunto motor e rotor 1.                            | [Hz]                     |
| $K_e$     | - constante elástica torcional do eixo.                                      | [Nm]                     |
| $c_e$     | - amortecimento do eixo.                                                     | [N.m.s]                  |
| $B_1$     | - vetor de entrada de acionamento do sistema.                                |                          |
| $B_2$     | - vetor de entrada de perturbação do sistema.                                |                          |
| $i_p$     | - corrente de perturbação.                                                   | [A]                      |

## Letras Gregas

|                         |                                                                                   |                        |
|-------------------------|-----------------------------------------------------------------------------------|------------------------|
| $\omega_{\text{rotor}}$ | - velocidade de rotação do rotor 1.                                               | [rad.s <sup>-1</sup> ] |
| $\omega_{\text{ref}}$   | - velocidade de rotação de referência.                                            | [rad.s <sup>-1</sup> ] |
| $\Omega_{\text{rotor}}$ | - velocidade de rotação do rotor no espaço da transformada Z ou de Laplace.       |                        |
| $\Omega_{\text{ref}}$   | - velocidade de rotação de referência no espaço da transformada Z ou de Laplace.  |                        |
| $\Phi$                  | - matriz de estados do modelo ARMA.                                               |                        |
| $\Gamma$                | - matriz de entradas do modelo ARMA.                                              |                        |
| $\theta^0$              | - vetor de parâmetros reais do modelo ARMA.                                       |                        |
| $\hat{\theta}$          | - vetor de parâmetros estimados do modelo ARMA.                                   |                        |
| $\hat{\theta}_{LS}$     | - vetor de parâmetros estimados por mínimos quadrados.                            |                        |
| $\gamma$                | - escolha usual do fator de ponderação do método dos mínimos quadrados.           |                        |
| $\hat{\theta}_{WLS}$    | - vetor de parâmetros estimados do método de mínimos quadrados ponderado.         |                        |
| $T_m$                   | - torque aplicado pelo motor.                                                     | [Nm]                   |
| $T_i$                   | - torque no motor devido à corrente elétrica circulante.                          | [Nm]                   |
| $T_r$                   | - torque no motor devido às forças dissipativas.                                  | [Nm]                   |
| $\kappa$                | - constante de integração.                                                        |                        |
| $\omega_1$              | - velocidade de rotação do rotor.                                                 | [rad.s <sup>-1</sup> ] |
| $\theta_m$              | - deslocamento angular do motor.                                                  | [rad]                  |
| $\theta_1$              | - deslocamento angular do rotor 1.                                                | [rad]                  |
| $\theta_{im}$           | - deslocamento angular relativo entre motor e rotor 1.                            | [rad]                  |
| $\omega_{nm}$           | - frequência angular natural do acoplamento entre motor e rotor 1.                | [rad.s <sup>-1</sup> ] |
| $\omega_{dtm}$          | - frequência angular natural amortecida do conjunto motor e rotor 1.              | [rad.s <sup>-1</sup> ] |
| $\sigma_{im}$           | - amortecimento do acoplamento entre motor e rotor 1.                             | [N.m.s]                |
| $\zeta_{im}$            | - fator de amortecimento do sistema.                                              |                        |
| $\theta_2$              | - deslocamento angular do rotor 2.                                                | [rad]                  |
| $\theta_i$              | - deslocamento angular relativo entre o rotor 1 e o rotor 2 (deformação do eixo). | [rad]                  |
| $\omega_n$              | - frequência angular natural do acoplamento entre rotores.                        | [rad.s <sup>-1</sup> ] |
| $\zeta$                 | - fator de amortecimento do acoplamento entre rotores.                            |                        |
| $\omega_d$              | - frequência angular natural amortecida do acoplamento entre rotores.             | [rad.s <sup>-1</sup> ] |

## Siglas

|     |                                        |
|-----|----------------------------------------|
| CC  | - corrente contínua.                   |
| PE  | - excitação persistente.               |
| PID | - proporcional, integral e derivativo. |

# Capítulo 1

## Introdução

No estudo dos métodos de controle para sistemas mecânicos rotativos encontra-se larga aplicação para a teoria de sistemas lineares com parâmetros concentrados e invariantes no tempo. Em aplicações nas quais existe um regime de demanda de torque variável, como por exemplo em esteiras rolantes para transporte ou em laminadores siderúrgicos, é preciso levar em conta o caráter não estático de alguns parâmetros do sistema, para que se possa projetar controladores que tenham seus desempenhos reais compatíveis com as características de projeto.

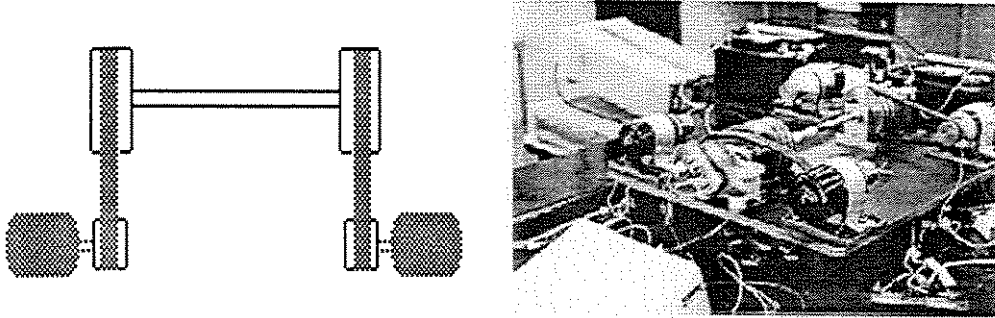
Embora em teoria e simulação seja possível prever a existência de controladores que permitam trabalhar com tais sistemas, a implementação real dessas estratégias envolve outros fatores, como a instrumentação necessária e disponível, para que se alcance de fato o desempenho de projeto para o sistema controlado. Um elemento chave é o tempo de amostragem do sistema controlador, pois ele deve ter no máximo a mesma ordem de grandeza que os tempos característicos do sistema a ser controlado. A existência de saturação no atuador de controle também faz parte do conjunto de limitantes, assim como a velocidade de processamento do computador utilizado para controle, pois deve ser rápido o suficiente para conseguir entregar de forma integral o sinal a ser aplicado no sistema controlado. A aplicação real deve levar em conta não só o conjunto matemático de projeto, mas também a estrutura operacional envolvida na realização de um sistema de controle.

Fazer o controle da velocidade de rotação desse caso experimental, o qual apresenta um regime de demanda de torque variável, é o objeto de estudo utilizado na elaboração deste trabalho.

### 1.1. Objetivos

A partir de uma bancada mecânica rotativa, que permite a simulação de uma situação de torque variável e que também possibilita a monitoração da deformação no eixo, serão apresentadas neste trabalho todas as etapas de identificação desse sistema rotativo.

A disposição mecânica do sistema é apresentada na Figura 1.1-1.



**Figura 1.1-1** – Sistema mecânico experimental.

São dois motores e dois rotores conectados por correias, e um eixo conectando os dois rotores. O sistema identificado será a base para o projeto de três tipos de controladores: tacométrico, polinomial e polinomial auto-sintonizável. O critério de projeto dos controladores foi a obtenção de uma rotação de saída estabilizada, e será feita a avaliação do desempenho desses controladores face a variações na demanda de torque dos sistemas utilizados. Em especial desejou-se dar atenção à implementação do método polinomial adaptativo utilizando mínimos quadrados recursivo.

As bases teóricas para a implementação das estratégias de controle apresentadas tiveram ponto de partida no final da década de 60, e os livros de ÅSTROM e WITTENMARK (1984), ÅSTROM e WITTENMARK (1989), FRANKLIN, POWELL et al (1990) e OGATA (1995), editados duas décadas depois, foram utilizados como ponto de referência. Nesses livros, além de toda a teoria de controle não adaptativo convencional, encontram-se os fundamentos e algoritmos para a implementação da estratégia de controle adaptativo indireto por modelo de referência.

## 1.2. Aspectos de Sistemas Adaptativos

Métodos de controle para sistemas lineares com parâmetros invariantes no tempo exigem do projetista o conhecimento de um modelo linear representativo do sistema a ser controlado. Essa exigência é o fator que simplifica a teoria, mas também é o ponto que pode introduzir a perda de robustez da estratégia adotada. Para sistemas variantes no tempo, é possível adotar estratégias de controle adaptativas, as quais vão fazer com que o controlador leve em conta alterações sofridas pelo sistema controlado. Esta seção traz um breve histórico do estudo de controladores adaptativos, além de uma coletânea de pontos relacionados ao projeto de

---

controladores adaptativos, mais especificamente os controladores adaptativos indiretos polinomiais por modelo de referência.

O estudo de controladores adaptativos é tópico de grande interesse desde o início dos anos 50, quando a indústria aeronáutica iniciou o desenvolvimento de mecanismos de guiagem automática para dar maior desempenho às aeronaves que surgiam àquela época, pois as estratégias de controle linear por realimentação não eram compatíveis com grandes mudanças nas condições de operação desses sistemas. Nos anos 60 a teoria de espaço de estados e a teoria de estabilidade dariam novas ferramentas para o tratamento de sistemas lineares, para que nos anos 70 a combinação entre métodos de estimação de parâmetros e técnicas de projeto resultasse no desenvolvimento e aplicação de várias estratégias adaptativas, embora houvesse falta de resultados teóricos servindo de base para a utilização desses controladores nas aplicações propostas. Ao final dos anos 70 e início dos anos 80 começaram a surgir os primeiros trabalhos referentes à estabilidade dos controladores adaptativos. Paralelamente, o desenvolvimento de equipamentos microeletrônicos possibilitou a implementação de controladores mais sofisticados de forma simples e barata, tendência esta que continua até hoje. O embasamento teórico continua avançando na direção da obtenção de controladores com convergência global garantida.

O trabalho que marca o início da utilização de controladores polinomiais auto-ajustáveis para sistemas lineares é o artigo de ÅSTRÖM e WITTENMARK (1973), citado como referência em grande parte dos trabalhos sobre controle polinomial adaptativo a partir de então. Os autores apresentam um algoritmo para o projeto de um regulador para plantas com parâmetros invariantes no tempo, embora sejam os valores dos parâmetros desconhecidos. Supõe-se um sistema de fase mínima, com atraso conhecido para aplicação do sinal de controle, e sendo conhecidos limites para a ordem do sistema. O mérito do trabalho é apresentar uma estratégia de controle com baixo custo computacional, baseada na estimação de parâmetros por método de mínimos quadrados seguida por projeto do controlador baseando-se nos parâmetros estimados. O sucesso dos resultados simulados e experimentais, decorrente das propriedades de convergência assintótica do algoritmo, tornaram o método promissor para aplicações industriais. Após 25 anos, com interesse sempre renovado por melhorias no algoritmo, o método de controle polinomial baseado em estimação por mínimos quadrados pode ser colocado como uma técnica de referência da teoria de controle de sistemas lineares.

---

KIDD (1985) faz uma comparação entre o desempenho em simulação de um sistema de controle com modelo de referência não adaptativo e um controlador adaptativo indireto com modelo de referência. A planta não controlada estudada é formada por um motor CC sujeito a uma demanda de torque variável, e o desempenho dos controladores é avaliado para algumas situações de operação não ideais, a saber: saturação na entrada da planta, mudanças no momento de inércia da carga do motor, variação dos parâmetros extrapolando hipóteses de projeto e perturbações externas aplicadas na saída. Para saturação na entrada e mudanças descontínuas na carga do motor, observa-se pouca diferença no desempenho dos controladores. Entretanto, o sistema não adaptativo mostrou estar melhor preparado para situações de extrapolação de hipóteses de projeto e para perturbações aplicadas ao sistema. Esse trabalho mostra um estágio inicial de preocupação com implementações reais de sistemas adaptativos, e fica clara a necessidade de buscar uma metodologia adequada ao projeto de controladores adaptativos, pois caso contrário, a utilização desse tipo de controle não acrescenta vantagens em implementações práticas.

GOODWIN e SIN (1981) fazem o estudo da convergência de um algoritmo para controle adaptativo indireto de um sistema linear discreto com parâmetros invariantes no tempo. A abordagem adotada visa a convergência do algoritmo adaptativo quando aplicado a sistemas de fase não mínima, pois até aquele momento os estudos de convergência tinham como hipótese básica a condição de estabilidade da função inversa do sistema (fase mínima). Nesse artigo é mostrado que tanto a entrada quanto a saída do sistema mantêm-se limitadas se os parâmetros de partida do estimador estiverem dentro de uma faixa relacionada com os parâmetros reais, sendo que não são necessárias hipóteses a respeito da convergência dos parâmetros ou da natureza da excitação externa.

Algoritmos para controle adaptativo tiveram estudos de convergência global feitos para aplicações em sistemas lineares com parâmetros invariantes no tempo, e os resultados mostram que esses algoritmos funcionam para as condições ideais estudadas. Entretanto, as motivações reais para o uso de controladores adaptativos é a utilização em sistemas variantes no tempo, e para esses casos uma teoria que demonstre a convergência global apresenta muitas dificuldades devido à complexidade da abordagem variante no tempo. GOODWIN, ELLIOT et al (1983) estudam a convergência de um algoritmo adaptativo com estimação por mínimos quadrados recursivo e reajuste da inversa da matriz de covariância. Esse algoritmo introduz uma alteração

simples no algoritmo original, e permite trabalhar com sistemas variantes no tempo. O estudo mostra que para o caso ideal com parâmetros invariantes no tempo, o algoritmo com reajuste da matriz é convergente, servindo como referência para a utilização em sistemas variantes no tempo.

A utilização de modelos de referência para controle polinomial sempre passa pela possibilidade ou pela viabilidade de cancelamento de pólos e zeros, vez que se o sistema real for um sistema de fase não mínima, o controlador terá pólos instáveis como resultado do cancelamento dos zeros localizados fora do círculo unitário na planta não controlada. ÅSTRÖM e WITTENMARK (1984) e ÅSTRÖM e WITTENMARK (1989) propõem que os zeros do modelo de referência sejam os zeros que tornariam o sistema controlado instável, o que exclui a necessidade de cancelamento desses zeros, mas reduz a capacidade de alocação de pólos e zeros que o método polinomial permite.

O algoritmo para controle auto-sintonizável proposto originalmente por ÅSTRÖM e WITTENMARK (1973) (Å-W), embora tenha sido utilizado em aplicações importantes e atraído a atenção para o estudo teórico, não tinha suas condições de convergência e atuação ótima provadas de forma rigorosa. GUO e CHEN (1991) retornam ao estudo do algoritmo original, provando a validade do algoritmo adaptativo original Å-W, e no decorrer do trabalho estendem a validade do algoritmo para um caso associado.

UNBENHAUEN e KEUCHEL (1992) abordam o problema do controle de sistemas de fase não mínima para aplicações em máquinas elétricas. A solução apresentada consiste em se utilizar uma planta estendida para contornar possíveis alocações indesejadas de pólos. A planta estendida é composta pela planta a ser controlada, mais uma malha de correção que basicamente faz a predição da perturbação à qual está sujeito o sistema, levando isso em conta através da malha de realimentação. O algoritmo proposto inclui a adaptação direta dos parâmetros do controlador, assim como a atualização dos parâmetros da rede de correção a partir dos parâmetros estimados para a planta. Esta abordagem agrega robustez ao método, pois garante que o sistema estendido vai sempre satisfazer as restrições de projeto. Partindo de modelos linearizados para duas plantas reais, a estratégia adaptativa aplicada em testes experimentais foi eficaz no controle de velocidade de um motor CC sujeito a uma demanda de torque variável, e também no controle da tensão de fase de um gerador síncrono sujeito a uma carga resistiva variável.

SHAFIQ, LU et al (1996) propõem que no modelo de referência estejam os zeros que levariam o sistema controlado à instabilidade, mas para manter a flexibilidade do método, sugerem a inclusão de pólos auxiliares de forma que os zeros indesejados e os pólos auxiliares

sejam vistos na planta controlada como um filtro “passa tudo”, ou seja, o modelo de referência pode ser escolhido a menos de uma defasagem constante introduzida pelo filtro, defasagem essa que pode ser compensada pela aplicação adiantada da entrada de referência. Os resultados de simulação com um sistema de fase não mínima mostram que o método proposto tem desempenho satisfatório.

Fazer o controle de sistemas com ordem elevada é uma tarefa que exige a utilização de ferramentas robustas e ao mesmo tempo implementáveis, pois a ordem está relacionada com o custo computacional. Para o projeto de controladores para sistemas lineares com ordem elevada, é proposto por BANDYOPADHYAY, UNBENHAUEN et al (1998) um método de redução de ordem que permite o projeto de um controlador por realimentação de saída a partir do sistema reduzido. Supondo-se um sistema de ordem  $n$  e um modelo reduzido de ordem  $r$ , sendo ambos SISO, de fase mínima e com pólos reais e distintos, e construindo um modelo modal de estados para o sistema escolhe-se um modelo reduzido. É feito o projeto de um controlador por realimentação de estados para o modelo reduzido segundo algum critério de desempenho, e para o sistema controlado calcula-se a função de transferência a partir da qual se obtém a função de transferência do controlador. Os resultados apresentados mostram que a garantia de controle no sistema completo não é equivalente ao sistema reduzido, mas que, por outro lado, o método é simples e não exige medida de estados, por trabalhar somente com entradas e saídas.

A escolha do modelo a ser utilizado para representar um sistema é um ponto crucial para o desenvolvimento da análise de desempenho de um controlador. A representatividade do modelo influencia o sucesso da estratégia de controle a ser utilizada. O modelamento deve ser feito de forma sistemática para que se obtenha a ordem do modelo adequada. Pode-se encontrar no artigo de WILSON e TAYLOR (1998) um algoritmo para a dedução da ordem de um modelo linear no domínio da frequência. Esse trabalho compara técnicas de *redução de ordem*, para as quais é necessária a existência prévia de um modelo de ordem elevada a partir do qual se pode fazer alguma redução. A *dedução de ordem* proposta parte de uma idéia oposta: a partir do agrupamento de subsistemas monta-se o modelo completo. Esse modelo vai sendo refinado e tendo sua ordem aumentada a partir do aumento gradual das ordens dos subsistemas até que a variação da resposta em frequência, em função das modificações nos subsistemas, seja menor que uma tolerância determinada pelo projetista. Com isso obtém-se o modelo de menor ordem possível que representa o sistema dentro da faixa de frequências desejada.

---

A identificação de sistemas lineares por método de mínimos quadrados tem papel central nas técnicas de controle adaptativo de sistemas dinâmicos lineares. ZHANG e FENG (1998) colocam o problema de haver perturbação por ruídos com distribuição não uniforme, o que resulta na polarização dos estimadores. Somando-se a isso, vem o fato de não ser possível utilizar os métodos tradicionais de estimação por mínimos quadrados em sistemas de malha fechada, devido à correlação entre saída e entrada introduzida pela realimentação. É proposto um método de mínimos quadrados indireto modificado que permite obter estimativas tanto para os parâmetros da planta quanto da malha de realimentação, mesmo quando não há conhecimento prévio do sistema realimentado, ou de um modelo para o erro. Resultados de simulação mostram que o método determina precisamente os parâmetros, além de também ser robusto com relação a ruídos correlacionados, sendo indicado para a identificação de sistemas de malha fechada.

Para que um sistema linear invariante no tempo possa ser controlado via alocação de pólos, é necessário que ele satisfaça a condição de controlabilidade [ver p. ex. OGATA (1995)]. Em estratégias de controle adaptativo essa condição é influenciada pela estimação dos coeficientes da planta, que podem fazer com que a matriz de Sylvester associada aos coeficientes seja singular, tornando o cálculo do controlador impossível. CHEN e CAO (1997) apresentam um método de estimação de parâmetros modificado que garante a convergência da estimação e conseqüentemente da estabilização adaptativa do sistema em um número finito de passos. O algoritmo modificado requer um esforço computacional reduzido, se comparado a outros métodos de estimação modificados. A existência de uma seqüência de modificadores para os parâmetros é a condição necessária e suficiente para que se obtenha a estabilização adaptativa, e como resultado incidental fica demonstrado que a condição de controlabilidade não é necessária para o método de controle adaptativo indireto.

KARACAN, HAPOGLU et al (1997) aplicam uma estratégia de controle polinomial no projeto de um controlador PID adaptativo indireto para uma coluna de destilação sujeita a variação de parâmetros. O modelo discretizado da planta, quando em regime estacionário, é de segunda ordem, sendo que os tempos característicos são da ordem de dezenas de segundos. São apresentados resultados das simulações e dos testes experimentais para o controlador adaptativo e para o controlador PID submetidos a uma entrada degrau. Para esse teste específico o controlador adaptativo teve desempenho superior ao controlador PID, pois esse último permitiu a existência de um erro estacionário.

---

### 1.3. Conteúdo do Texto

No Capítulo 2 são abordados os aspectos gerais dos métodos de controle e identificação implementados. Inicialmente apresenta-se o controlador por realimentação proporcional de saída, ou tacométrico, o controlador polinomial, o qual permite seguir um modelo de referência através de filtros na malha direta e na malha de realimentação, a estimação recursiva que atualiza a cada iteração os valores dos parâmetros da planta, e o controle polinomial auto-sintonizável, a qual é uma estratégia de controle que utiliza parâmetros estimados recursivamente para determinar um controlador por modelo de referência.

Em seguida, no Capítulo 3 é mostrado como foi feita a identificação do sistema experimental utilizado, partindo-se dos sub-elementos até o sistema completo, passando pela construção de um modelo reduzido. Mostram-se também os resultados da utilização do estimador recursivo para a identificação do sistema completo.

O Capítulo 4 apresenta o sistema de controle e suas conexões com a bancada mecânica. A caracterização das fontes de corrente utilizadas para o acionamento e perturbação do sistema também é apresentada. Está descrita a implementação dos sistemas de controle, obedecendo a ordem do Capítulo 2: resultados de controle tacométrico, polinomial e polinomial auto-sintonizável, sendo que todos os controladores foram testados para a atuação no sistema reduzido e no sistema completo.

No Capítulo 5 encontram-se as análises dos resultados e as conclusões do trabalho desenvolvido. É feita a análise dos resultados de identificação obtidos a partir dos métodos apresentados no Capítulo 3. Esses resultados são utilizados no Capítulo 4 para o projeto dos controladores implementados, os quais têm seu desempenho avaliado no Capítulo 5, que por ser o capítulo de encerramento, também contém as observações finais e conclusões.

Após as conclusões e considerações finais encontram-se as referências bibliográficas para as citações do texto, seguidas pelos apêndices, com diagramas e descrições de funcionamento dos circuitos eletrônicos implementados para a instrumentação do sistema, uma breve descrição do método simplex não linear utilizado no Capítulo 3 para fim de identificação, além de listagens dos programas de simulação em ambiente MATLAB® e a listagem do programa em linguagem “C” utilizado na implementação experimental dos algoritmos de controle.

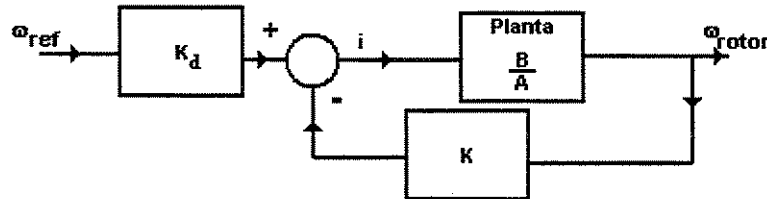
## Capítulo 2

### Controle de Sistemas Rotativos

Os sistemas rotativos controlados abordados neste trabalho têm como ênfase o controle de rotação, e por isso o primeiro método de controle a ser descrito é o método de controle tacométrico. Seguindo o texto, encontra-se o método polinomial, que se apresenta como um método mais versátil em relação ao tacométrico. Mais adiante é apresentada a estimação recursiva utilizando o método dos mínimos quadrados, e será esta estimação, somada ao método polinomial, que resultará no controlador auto-sintonizável, detalhado na última seção.

#### 2.1. Controle Tacométrico

Foi implementado o controle tacométrico descrito no diagrama da Figura 2.1-1, o qual descreve um sistema de controle com realimentação de saída, que compensa aumentos de velocidade na saída com decréscimos na corrente de entrada [OGATA (1995) e SHAIAM e HASSUL (1993)].



**Figura 2.1-1** – Controlador tacométrico implementado.

A função de transferência de malha fechada para o sistema controlado é:

$$G_c(z) = \frac{\Omega_{ref}}{\Omega_{rotor}} = \frac{K_d B}{A + KB} \quad (2.1-1)$$

O controlador é projetado escolhendo-se o ganho  $K$  em função da posição das raízes da equação característica do sistema controlado:

$$A + KB = 0 \quad (2.1-2)$$

A posição das raízes do controlador tacométrico é escolhida através de método clássico da teoria de controle, detalhado no Apêndice C, e pode ser encontrado em OGATA (1995), FRANKLIN, POWELL et al (1990) e ÅSTRÖM e WITTENMARK (1984).

Uma vez escolhido o ganho  $K$ , e conseqüentemente os pólos do sistema,  $K_d$  é escolhido de forma que o ganho estático do sistema controlado seja unitário:

$$G_c(z=1) = \frac{K_d B(z=1)}{A(z=1) + KB(z=1)} = 1$$

$$\Downarrow \quad (2.1-3)$$

$$K_d = \frac{A(z=1) + KB(z=1)}{B(z=1)}$$

Desse modo, o erro estacionário para a resposta ao degrau unitário é nulo.

## 2.2. Controle Polinomial

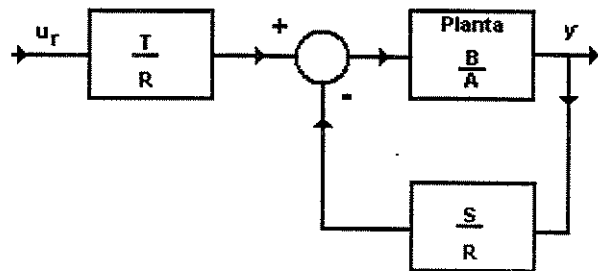
Ao contrário do controle tacométrico, que tem apenas o ganho  $K$  como parâmetro livre para projeto, o controle polinomial permite a obtenção de uma planta controlada que tem comportamento igual a um modelo desejado, podendo-se alocar tanto pólos quanto zeros, como descrito em ÅSTRÖM e WITTENMARK (1984) e FRANKLIN, POWELL et al (1990).

Dado um sistema descrito pela função de transferência:

$$G(z) = \frac{B(z)}{A(z)} \quad (2.2-1)$$

deseja-se projetar um controlador utilizando os sistemas polinomiais constituídos pelos polinômios  $R$ ,  $S$  e  $T$  (figura 2.2-1), que faça a planta ter comportamento igual a um modelo de referência:

$$G_m(z) = \frac{B_m(z)}{A_m(z)} \quad (2.2-2)$$



**figura 2.2-1** - Sistema com controle polinomial.

A função de transferência do sistema com o controlador é

$$G_c(z) = \frac{BT}{AR + BS} \quad (2.2-3)$$

onde foi omitida a variável complexa  $z$  por simplicidade. Mas o objetivo é fazer a planta controlada se comportar como o modelo, ou seja:

$$G_c = \frac{BT}{AR + BS} = \frac{B_m}{A_m} \quad (2.2-4)$$

### 2.2.1. Desenvolvimento

A equação característica de  $G_c$  é  $AR + BS = 0$ , através da qual se obtém os pólos do sistema controlado, e  $BT = 0$  é a equação que determina seus zeros.

Da Equação 2.2-4 pode-se ver que  $G_c$  tem como zeros os da planta sem controle ( $B$ ) e as raízes do polinômio  $T$ . Se os zeros da planta sem controle não fizerem parte dos zeros do modelo de referência, eles deverão ser cancelados para que a planta controlada  $G_c$  não tenha essa influência em sua dinâmica.

Para cancelar esses zeros é necessário que haja pólos coincidentes na planta controlada, ou seja, o denominador da função de transferência,  $AR + BS$ , deverá ter raízes nas mesmas posições que o numerador. Entretanto,  $AR + BS$  tem que deixar o sistema estável, e por isso os pólos têm que ser estáveis. Se  $B$  for fatorado da seguinte forma:

$$B = B^- B^+ \quad (2.2-5)$$

sendo  $B^+$  o polinômio com raízes dentro do círculo unitário e  $B^-$  o polinômio com raízes fora do círculo unitário, então  $AR + BS$  deve excluir  $B^-$  como fator, e com isso  $B_m$  deve ser da forma:

$$B_m = B^- B_m^+ \quad (2.2-6)$$

Isso implica em não se poder mudar zeros instáveis e, ainda, incorporá-los ao modelo. Explicitando as últimas imposições, faz-se de  $B^+$  um fator de  $AR + BS$ . Escrevendo

$$R = B^+ R^+ \quad (2.2-7)$$

pode-se rerepresentar a equação (2.2-4):

$$G_c = \frac{B^+ B^- T}{B^+ (AR + B^- S)} = \frac{B^- B_m^+}{A_m} \quad (2.2-8)$$

Assim, a equação básica do método polinomial excluindo-se os zeros indesejados torna-se:

$$\frac{T}{AR' + B'S} = \frac{B'_m}{A_m} \quad (2.2-9)$$

Deve-se notar na Equação 2.2-9 que pode haver uma incompatibilidade de ordens entre o numerador da planta controlada e o do modelo. Para que se contorne esse problema introduz-se nesse passo um polinômio observador  $A_o$  no sistema:

$$\begin{cases} AR' + B'S = A'_m A_o \\ T = B'_m A_o \end{cases} \quad (2.2-10)$$

A denominação *polinômio observador* advém da semelhança de comportamento da dinâmica do controlador. Assim como na teoria de controle por realimentação de estados, o polinômio observador se assemelha aos *observadores de estado*, por não ser identificável a partir da entrada e saída do sistema controlado, funcionando apenas como elemento auxiliar na obtenção da estratégia de controle desejada.

Com isso, a equação característica do sistema de malha fechada se torna:

$$AR + BS = B^+ A_o A_m \quad (2.2-11)$$

Essa equação é chamada *equação diofantina*, e uma maneira de se obter os polinômios  $R$  e  $S$  utilizando-a é montar um sistema de equações lineares para os coeficientes de termos de mesma ordem dos polinômios envolvidos. A matriz resultante desse procedimento é chamada matriz de Sylvester (ver implementação na Seção 4.3).

Uma vez obtida a equação característica, é necessário escolher a ordem dos polinômios de forma a garantir a causalidade do sistema controlado. Escolhendo  $O(R) = O(S) = O(T)$ , onde  $O(\cdot)$  representa a função que retorna a ordem do argumento, e supondo  $O(B) \leq O(A)$ , tem-se:

$$O(AR + BS) = \max(O(AR), O(BS)) = \max(O(A) + O(R), O(B) + O(S)) = O(A) + O(R) = O(AR)$$

Logo:

$$O(AR + BS) = O(AR) = O(B^+ A_o A_m)$$

Assim:

$$O(R) = O(B^+) + O(A_o) + O(A_m) - O(A)$$

Mas

$$O(R) = O(T) = O(A_o) + O(B'_m)$$

Então

$$O(R) = O(A_o) + O(B'_m) = O(B^+) + O(A_o) + O(A_m) - O(A)$$

E portanto:

$$O(A_m) - O(B_m) = O(A) - O(B^+) \quad (2.2-12)$$

Supondo  $O(S) = O(A) - 1$ :

$$O(R) = O(S) = O(B) - 1 = O(B^+) + O(A_o) + O(A_m) - O(A)$$

E assim:

$$O(A_o) = 2O(A) - O(A_m) - O(B^+) - 1 \quad (2.2-13)$$

Assim, as equações (2.2-12) e (2.2-13) constituem um conjunto de restrições necessárias para garantir a causalidade do sistema mecânico controlado pelo método polinomial.

## 2.3. Identificação Recursiva

Nesta seção, será apresentado o método de identificação recursiva que utiliza estimação de parâmetros por mínimos quadrados e que tem embutido no algoritmo um fator de esquecimento, o qual pondera os dados utilizados na estimação.

### 2.3.1. O Modelo ARMA

Considerando um modelo linear discreto de terceira ordem, sem perda de generalidade, no qual  $q$  representa o operador avanço unitário:

$$G(q) = \frac{b_1 q^2 + b_2 q + b_3}{q^3 + a_1 q^2 + a_2 q + a_3}$$

A resposta no tempo pode ser escrita como:

$$y_k = -a_1 x_1(k) - a_2 x_2(k) - a_3 x_3(k) + b_1 x_4(k) + b_2 x_5(k) + b_3 x_6(k)$$

O índice  $k$  representa as medidas lidas no tempo  $t=kT$ , sendo  $T$  o intervalo de amostragem. A formulação de estados para o sistema tem o seguinte vetor de estados:

$$x_k = \begin{Bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \\ x_6(k) \end{Bmatrix} = \begin{Bmatrix} y_{k-1} \\ y_{k-2} \\ y_{k-3} \\ u_{k-1} \\ u_{k-2} \\ u_{k-3} \end{Bmatrix} \quad (2.3-1)$$

O modelo escrito dessa forma é denominado modelo ARMA (AutoRegressive Moving Average - média móvel autorregressiva) e tem a propriedade de fornecer o estado atual exatamente a partir de valores passados de entrada e saída. O modelo de estado completo é:

$$\begin{cases} x_{k+1} = \Phi x_k + \Gamma u_k \\ y_k = C x_k \end{cases} \quad (2.3-2)$$

sendo que

$$\begin{cases} \Phi = \begin{bmatrix} -a_1 & -a_2 & -a_3 & b_1 & b_2 & b_3 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \\ \Gamma = [0 \ 0 \ 0 \ 1 \ 0 \ 0]^T \\ C = [-a_1 \ -a_2 \ -a_3 \ b_1 \ b_2 \ b_3] \end{cases}$$

Com o modelo ARMA, a única equação necessária para se calcular toda resposta do sistema é

$$y_k = C x_k = -a_1 x_1(k) - a_2 x_2(k) - a_3 x_3(k) + b_1 x_4(k) + b_2 x_5(k) + b_3 x_6(k) \quad (2.3-3)$$

pois essa equação já embute as equações de estado. Deve-se ressaltar que o modelo está definido nos coeficientes das matrizes  $\Phi$  e  $C$ . O vetor de parâmetros  $\theta$  será definido a seguir e sua construção será a base do algoritmo de estimação recursiva.

### 2.3.2. Colocação do Problema

De toda a gama de modelos, é desejável que se escolha um modelo com o menor número de parâmetros possível e que ele descreva o comportamento da planta. É interessante também que a correspondência entre observações e modelo estimado seja unívoca, mas não

raramente são escolhidos modelos que simplificam o projeto mesmo não sendo unívoca a relação entre modelo e planta real.

Uma vez escolhido o modelo paramétrico da planta, deve-se fazer a estimação dos parâmetros  $\hat{\theta}$  que melhor representam os dados existentes. Para isso é necessário ter alguma idéia da representatividade dos valores  $\hat{\theta}$  em relação aos parâmetros  $\theta^0$ , que são os parâmetros reais da planta. Como  $\theta^0$  é desconhecido, é necessário definir o erro de forma a poder calculá-lo a partir de  $u_k$  e  $y_k$ . O erro definido a seguir é chamado erro de equação.

Para obter o erro de equação é necessário conhecer as equações de movimento do sistema, como, por exemplo, através da descrição por variáveis de estado. Generalizando e escolhendo um sistema contínuo com vetor de parâmetros  $\theta$ :

$$\dot{x} = f(x, u, \theta) \quad (2.3-5)$$

A forma de  $f$  é supostamente conhecida, mas não o são os coeficientes  $\theta^0$  que descrevem a planta. É suposto possível medir não só  $u_k$ , mas também  $x$  e  $\dot{x}$ . Sendo assim, tudo é conhecido no sistema, menos  $\theta^0$ . Então, é possível, deste modo, montar uma equação de erro de forma a evidenciar a discrepância entre a resposta medida e aquela calculada a partir dos parâmetros  $\theta$ :

$$\dot{x}_r - f(x_r, u_r, \theta) = e(t, \theta) \quad (2.3-6)$$

sendo  $e(t, \theta^0) = 0$  ( $\theta^0$  é o vetor de parâmetros reais da planta). O índice  $r$  enfatiza que os valores utilizados são os valores medidos.

O vetor  $e$  contém os erros de equação, que podem formar uma equação não negativa:

$$J(\theta) = \int_0^{t_0} e^T(t, \theta) e(t, \theta) dt \quad (2.3-7)$$

Utilizando-se a equação 2.3-7, pode-se fazer uma busca em  $\theta$  para determinar  $\hat{\theta}$  tal que  $J(\hat{\theta}) = 0$  e dessa forma obter  $\hat{\theta}$  equivalente a  $\theta^0$ . Se uma parametrização unívoca é escolhida, somente um parâmetro fará  $e(t, \theta^0) = 0$  e assim  $\hat{\theta} = \theta^0$ . Se houver ruído presente, o erro não convergirá para zero quando  $\hat{\theta} = \theta^0$ , mas  $J$  será mínima naquele ponto.

Contudo, a hipótese de haver sensores para medir todos os estados e suas derivadas é muito restritiva e não é realista para sistemas contínuos. Ainda assim, em sistemas discretos lineares há um caso no qual é possível ter acesso a todos esses estados: o modelo ARMA, e isso acontece porque os estados nesse modelo são os valores recentes de entrada e saída. Explicitamente tem-se para um sistema discreto:

$$x_r(k+1) - \Phi x_r(k) - \Gamma u_r(k) = e(k, \theta) \quad (2.3-8)$$

Para o modelo ARMA:

$$\begin{Bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \\ x_5(k+1) \\ x_6(k+1) \end{Bmatrix} - \begin{bmatrix} -a_1 & -a_2 & -a_3 & b_1 & b_2 & b_3 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \\ x_6(k) \end{Bmatrix} - \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{Bmatrix} u(k) = \begin{Bmatrix} e_1(k) \\ e_2(k) \\ e_3(k) \\ e_4(k) \\ e_5(k) \\ e_6(k) \end{Bmatrix} \quad (2.3-9)$$

Substituindo a Equação 2.3-1 na Equação 2.3-9 conclui-se que para qualquer  $\theta$ , os erros de equação são todos zero, exceto  $e_1$ , que é dado por:

$$x_1(k+1) + a_1 x_1(k) + a_2 x_2(k) + a_3 x_3(k) - b_1 x_4(k) - b_2 x_5(k) - b_3 x_6(k) = e_1(k, \theta) \quad (2.3-10)$$

ou

$$y_r(k) + a_1 y_r(k-1) + a_2 y_r(k-2) + a_3 y_r(k-3) - b_1 u_r(k-1) - b_2 u_r(k-2) - b_3 u_r(k-3) = e_1(k, \theta) \quad (2.3-11)$$

A medida de desempenho da Equação 2.3-7 fica sendo:

$$J(\theta) = \sum_{k=0}^N e_1^2(k, \theta) \quad (2.3-12)$$

### 2.3.3. Mínimos Quadrados

Para fazer a estimação dos parâmetros via mínimos quadrados pode-se utilizar o modelo ARMA para o sistema, e definir o erro como sendo o erro de equação. Reescrevendo a Equação 2.3-11 para o caso de ordem  $n$ , obtém-se:

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) - b_1 u(k-1) - \dots - b_n u(k-n) = e(k, \theta) \quad (2.3-13)$$

(lembrando que os valores de entradas e saídas são os valores reais observados).

Supondo que os valores observados sejam

$$\{y(0), y(1), \dots, y(N), u(0), u(1), \dots, u(N)\}$$

e que se deseja calcular

$$\theta = \{a_1 \ a_2 \ a_3 \ b_1 \ b_2 \ b_3\}^T$$

tal que se tenha a melhor representação dos dados observados. Como  $y_k$  depende das observações até  $n$  períodos atrás, o primeiro erro que pode ser construído é  $e(n, \theta)$ . Supondo

que se defina o vetor de erros  $E$  a partir da aplicação repetida da equação (2.3-13) para  $k=n, n+1, \dots, N$ , o resultado obtido é:

$$\begin{aligned} y(n) &= \varphi^T(n)\theta + e(n, \theta) \\ y(n+1) &= \varphi^T(n+1)\theta + e(n+1, \theta) \\ &\vdots \\ y(N) &= \varphi^T(N)\theta + e(N, \theta) \end{aligned} \quad (2.3-14)$$

sendo aqui utilizado  $\varphi(k)$  como vetor de estados (ou *regressor*) do modelo ARMA:

$$\varphi(k) = [-y(k-1) \quad \dots \quad -y(k-n) \quad u(k-1) \quad \dots \quad -u(k-n)]^T$$

Para tornar as equações ainda mais compactas, pode-se definir:

$$\begin{aligned} Y(N) &= [y(n) \quad \dots \quad y(N)]^T \\ \Phi(N) &= [\varphi(n) \quad \dots \quad \varphi(N)]^T \\ e(N, \theta) &= [e(n) \quad \dots \quad e(N)]^T \\ \theta &= [a_1 \quad \dots \quad a_n \quad b_1 \quad \dots \quad b_n]^T \end{aligned} \quad (2.3-15)$$

sendo que  $\Phi$  possui  $2n$  colunas e  $(N-n+1)$  linhas. Com isso a equação de erro pode ser reescrita como:

$$Y = \Phi\theta + e(N, \theta) \quad (2.3-16)$$

O método dos mínimos quadrados consiste em seleccionar  $\theta$  tal que a soma dos quadrados dos erros  $e(k)$  é a menor possível. Em termos da Equação 2.3-14 define-se:

$$J(\theta) = \sum_{k=n}^N e^2(k, \theta) \quad (2.3-17)$$

E em termos da Equação 2.3-16,  $J$  é dado por:

$$J(\theta) = e^T(N, \theta)e(N, \theta) \quad (2.3-18)$$

Deseja-se determinar  $\hat{\theta}_{LS}$  (a estimativa por mínimos quadrados de  $\theta^0$ ), que é o vetor  $\theta$  com a seguinte propriedade:

$$J(\hat{\theta}_{LS}) \leq J(\hat{\theta}) \quad (2.3-19)$$

sendo  $J(\theta)$  uma função quadrática dos  $2n$  parâmetros de  $\theta$ . Do cálculo com várias variáveis tem-se o resultado que diz que uma condição necessária para  $J(\hat{\theta}_{LS})$  ser mínima é que as derivadas parciais de  $J$  com relação a  $\theta$  sejam zero para  $\theta = \hat{\theta}_{LS}$ .

$$\begin{aligned} J(\hat{\theta}) &= e^T e \\ &= (Y - \Phi\theta)^T (Y - \Phi\theta) \\ &= Y^T Y - \theta^T \Phi^T Y - Y^T \Phi\theta + \theta^T \Phi^T \Phi\theta \end{aligned} \quad (2.3-20)$$

Aplicando as regras de derivação para escalares com relação a vetores:

$$J_{\theta} = \left\{ \frac{\partial J}{\partial \theta_i} \right\} = -2Y^T \Phi + 2\theta^T \Phi^T \Phi \quad (2.3-21)$$

Transpondo a Equação 2.3-21, fazendo  $\theta = \hat{\theta}_{LS}$  e igualando-a a zero (condição de mínimo) obtém-se:

$$\Phi^T \Phi \hat{\theta}_{LS} = \Phi^T Y \quad (2.3-22)$$

A Equação 2.3-22 é chamada equação normal. Para que a Equação 2.3-22 determine  $\theta$  de forma efetiva é necessário que a entrada seja uma excitação que atue no sistema de forma a se poder observar ao máximo suas características dinâmicas. Esse tipo de excitação é denominado excitação persistente (PE ou *persistent excitation*). Submetendo o sistema a uma entrada PE, de forma que  $\Phi^T \Phi$  seja não singular e que  $\theta$  seja identificável, o conjunto de parâmetros  $\theta$  deve ser mínimo e  $J(\hat{\theta}_{LS})$  é única (FRANKLIN et al, 1990).

Supondo que  $u_k$  seja PE, e que  $\theta$  seja identificável, pode-se obter explicitamente a solução:

$$\hat{\theta}_{LS} = (\Phi^T \Phi)^{-1} \Phi^T Y \quad (2.3-23)$$

A estimação feita dessa forma chama-se estimação por *mínimos quadrados não ponderada*, e a medida de desempenho dada pela Equação 2.3-18 reflete uma suposição de paridade de importância dos erros. Nem sempre isso é verdadeiro, pois uma pequena modificação pode introduzir uma escala de importância entre os erros. Com a introdução da escala, o método passa a se chamar método dos *mínimos quadrados ponderado* e tem como medida de desempenho:

$$J(\theta) = \sum_{k=n}^N w(k) e^2(k, \theta) = e^T w e \quad (2.3-24)$$

Na Equação 2.3-24,  $w(k)$  é suposto sempre positivo, e o cálculo da equação normal (2.3-23) resulta em:

$$\Phi^T W \Phi \hat{\theta}_{WLS} = \Phi^T W Y \quad (2.3-25)$$

sendo sujeita à condição de se ter uma matriz de coeficientes  $\Phi$  não singular, para que então:

$$\hat{\theta}_{WLS} = (\Phi^T W \Phi)^{-1} \Phi^T W Y \quad (2.3-26)$$

Quando  $W=I$ , a Equação 2.3-26 se reduz à equação para mínimos quadrados não ponderado (Equação 2.3-22). Uma escolha comum é  $w(k) = (1 - \gamma)\gamma^{N-k}$ ;  $\gamma < 1$ . Essa escolha

pondera as observações recentes ( $k$  próximo a  $N$ ) como mais importantes que as passadas ( $k$  próximo a  $n$ ).

### 2.3.4. Algoritmo Recursivo

Na seção anterior foi mostrado o método dos mínimos quadrados ponderado para se fazer a estimação dos parâmetros de um modelo ARMA. O método apresentado é útil para se fazer a estimação de parâmetros quando se tem uma coleção de pontos de entrada e saída do sistema. Mas, às vezes, deseja-se manter  $\hat{\theta}_{WLS}$  atualizado: ou para ser possível melhorar a estimativa de  $\theta$ , ou porque se deseja monitorar mudanças nos parâmetros, ou então porque as informações são recebidas sequencialmente. Nesses casos é necessário modificar a metodologia de determinação de  $\hat{\theta}_{WLS}$ , e assim obter um algoritmo para fazer o cálculo recursivo da estimação via mínimos quadrados (ou *Recursive Least Squares*, em inglês).

Abordando o problema a partir da equação normal (2.3-24), na qual se tem a solução  $\hat{\theta}_{WLS}$  para  $N$  pontos experimentais, pode-se estudar as consequências de se incluir mais uma observação. É necessário considerar as implicações da inclusão de mais um ponto em  $\Phi^T W \Phi$  e em  $\Phi^T W Y$ . Especificamente, considera-se o caso  $w(k) = a\gamma^{N-k}$ . Se  $a=1$  e  $\gamma=1$ , obtém-se mínimos quadrados não ponderado; se  $a=1-\gamma$ , o resultado é o método dos mínimos quadrados ponderado exponencialmente.

Para o tempo  $N+1$ , obtém-se a partir das equações (2.3-15):

$$\Phi(N+1) = [\varphi(n) \quad \dots \quad \varphi(N) \quad \varphi(N+1)] \quad (2.3-27)$$

e

$$\Phi^T W \Phi = \sum_{k=n}^{N+1} \varphi(k) w(k) \varphi^T(k) = \sum_{k=n}^{N+1} \varphi(k) a \gamma^{N+1-k} \varphi^T(k) \quad (2.3-28)$$

A equação (2.3-28) pode ser escrita como a soma de dois termos:

$$\begin{aligned} \Phi^T W \Phi &= \sum_{k=n}^{N+1} \varphi(k) a \gamma^{N-k} \varphi^T(k) + \varphi(N+1) a \varphi^T(N+1) \\ &= \Phi^T(N) W(N) \Phi(N) + \varphi(N+1) a \varphi^T(N+1) \end{aligned} \quad (2.3-29)$$

Da solução analítica dada pela equação (2.3-26) para se determinar  $\hat{\theta}_{WLS}$ , vê-se que é necessário que exista a inversa da matriz  $\Phi^T W \Phi$ . Por convenção, define-se a matriz  $P$ , a inversa da matriz de covariância, como:

$$P(N+1) = [\Phi^T(N+1)W\Phi(N+1)]^{-1} \quad (2.3-30)$$

sendo que  $P$  é quadrada de ordem  $2n$ . Reescrevendo a equação (2.3-29) em termos de  $P$ :

$$P(N+1) = [\gamma P^{-1}(N) + \varphi(N+1)a\varphi^T(N+1)]^{-1} \quad (2.3-31)$$

É necessário, então, obter a inversa da soma de duas matrizes. Da álgebra matricial vem o resultado (FRANKLIN, POWELL et al, 1990, p.379):

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \quad (2.3-32)$$

Para aplicar (2.3-6) a (2.3-5), faz-se:

$$\begin{aligned} A &= \gamma P^{-1}(N) \\ B &= \varphi(N+1) \equiv \varphi \\ C &= w(N+1) \equiv a \\ D &= \varphi^T(N+1) \equiv \varphi^T \end{aligned}$$

E assim:

$$P(N+1) = \frac{P(N)}{\gamma} - \frac{P(N)}{\gamma} \varphi \left( \frac{1}{a} + \varphi^T \frac{P(N)}{\gamma} \varphi \right)^{-1} \varphi^T \frac{P(N)}{\gamma} \quad (2.3-33)$$

Para a determinação de  $\hat{\theta}_{WLS}$  também é necessário o termo  $\Phi^T W \Phi$ , que pode ser escrito como:

$$\Phi^T W \Phi = \begin{bmatrix} \varphi(n) & \dots & \varphi(N) & \varphi(N+1) \end{bmatrix} \begin{bmatrix} a\gamma^{N+1} & 0 & \dots & 0 \\ 0 & \ddots & & \\ \vdots & & a\gamma & 0 \\ 0 & \dots & 0 & a \end{bmatrix} \begin{bmatrix} y(n) \\ \vdots \\ y(N) \\ y(N+1) \end{bmatrix} \quad (2.3-34)$$

A equação (2.3-34) pode ser reescrita como:

$$\Phi^T W Y(N+1) = \gamma \Phi^T W Y(N) + \varphi a y(N+1) \quad (2.3-35)$$

Substituindo agora a expressão para  $P(N+1)$  dada pela equação (2.3-33):

$$\hat{\theta}_{WLS}(N+1) = \left[ \frac{P}{\gamma} - \frac{P}{\gamma} \varphi \left( \frac{1}{a} + \varphi^T \frac{P}{\gamma} \varphi \right)^{-1} \varphi^T \frac{P}{\gamma} \right] [\gamma \Phi^T W Y(N) + \varphi a y] \quad (2.3-36)$$

Ao se multiplicar os fatores da equação (2.3-36), chega-se à conclusão que  $P\Phi^T W Y(N) = \hat{\theta}_{WLS}(N)$ , e com isso a equação (2.3-36) reduz-se a:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + \frac{P}{\gamma} \varphi a y - \frac{P}{\gamma} \varphi \left( \frac{1}{a} + \varphi^T \frac{P}{\gamma} \varphi \right)^{-1} \varphi^T \frac{P}{\gamma} \hat{\theta}_{WLS}(N) + \frac{P}{\gamma} \varphi \left( \frac{1}{a} + \varphi^T \frac{P}{\gamma} \varphi \right)^{-1} \varphi^T \frac{P}{\gamma} \varphi a y \quad (2.3-37)$$

Se agora for colocada a matriz identidade

$$I = \left( \frac{1}{a} + \varphi^T \frac{P}{\gamma} \varphi \right)^{-1} \left( \frac{1}{a} + \varphi^T \frac{P}{\gamma} \varphi \right) \quad (2.3-38)$$

entre  $\varphi$  e  $a$  no segundo termo do lado direito da equação (2.3-37), pode-se combinar os dois termos que multiplicam  $y$  para reduzir a equação (2.3-37) a:

$$\hat{\theta}_{WLS}(N+1) = \hat{\theta}_{WLS}(N) + L(N+1) \{y(N+1) - \varphi^T \hat{\theta}_{WLS}(N)\} \quad (2.3-39)$$

sendo

$$L(N+1) = \frac{P}{\gamma} \varphi \left( \frac{1}{a} + \frac{\varphi^T P \varphi}{\gamma} \right)^{-1} \quad (2.3-40)$$

As equações (2.3-30), (2.3-39) e (2.3-40) juntas formam um algoritmo para calcular  $\hat{\theta}$  recursivamente, que é descrito abaixo:

- Calcular o erro de equação:

$$\varepsilon(k) = \omega(k) - \varphi^T(k-1) \hat{\theta}(k-1) \quad (2.3-41)$$

- Atualizar o vetor de parâmetros:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k) \varepsilon(k) \quad (2.3-42)$$

- Calcular o novo passo de atualização do vetor de parâmetros:

$$K(k) = P(k-1) \varphi(k-1) [\lambda + \varphi^T(k-1) P(k-1) \varphi(k-1)]^{-1} \quad (2.3-43)$$

- Calcular a matriz  $P$ :

$$P(k) = [I - K(k) \varphi^T(k-1)] P(k-1) / \lambda \quad (2.3-44)$$

Notar na Equação 2.3-43 que o termo

$$\lambda + \varphi^T P \varphi$$

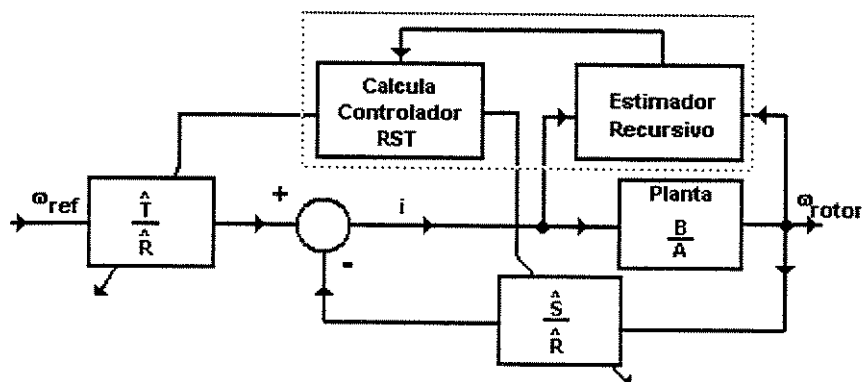
é um termo escalar no caso em que  $\varphi$  é um vetor coluna, e portanto a inversão de matriz reduz-se a uma divisão escalar.

## 2.4. Controle Polinomial Auto-sintonizável

Nas seções 2.2 e 2.3 foram apresentados os métodos de controle polinomial e de estimação por mínimos quadrados recursivo. Utilizando-os como ponto de partida, pode-se

projetar um controlador polinomial adaptativo que utiliza o método de estimação recursivo apresentado para obter os coeficientes da planta controlada e com isso recalculando os polinômios do controlador em tempo real, de forma que o sistema controlado tenha resposta igual ao modelo de referência escolhido.

Para o controle polinomial implementado foi escolhido um modelo de referência com os mesmos zeros que a planta identificada. O diagrama do sistema adaptativo pode ser visto abaixo:



**Figura 2.4-1** – Diagrama do Sistema com Controlador Adaptativo

O controlador adaptativo tem seu funcionamento baseado na utilização conjunta de um estimador recursivo e um controlador polinomial. O controlador propriamente dito envia um sinal de controle a partir dos polinômios  $\hat{R}$ ,  $\hat{S}$  e  $\hat{T}$ . Estes polinômios são inicializados com os valores de um controlador polinomial projetado para atuar em um sistema linear com parâmetros invariantes no tempo. Ao mesmo tempo, um estimador recursivo atualiza os valores dos coeficientes da planta a partir dos dados de entrada e saída medidos, como no algoritmo apresentado na Seção 2.3.

A partir desses novos parâmetros estimados, um novo conjunto de polinômios controladores é calculado para que na próxima iteração a ação de controle aplicada esteja adequada aos novos parâmetros obtidos para a planta:

$$\hat{\theta} \Rightarrow \hat{A}, \hat{B} \Rightarrow \hat{R}, \hat{S}, \hat{T} \quad (2.4-1)$$

Essa característica qualifica esse controlador como um controlador adaptativo indireto ou explícito, pois os elementos controladores são calculados a partir de parâmetros da planta explicitamente caracterizados. O algoritmo para controle auto-sintonizável está apresentado a seguir, e nele estão descritos todos os passos necessários à implementação do controlador, desde a etapa de identificação até a etapa de cálculo dos polinômios  $\hat{R}$ ,  $\hat{S}$  e  $\hat{T}$ :

$$\begin{aligned}
\varepsilon(k) &= \omega(k) - \varphi^T(k-1)\hat{\theta}(k-1) \\
\hat{\theta}(k) &= \hat{\theta}(k-1) + K(k)\varepsilon(k) \\
K(k) &= P(k-1)\varphi(k-1)\left[\lambda + \varphi^T(k-1)P(k-1)\varphi(k-1)\right]^{-1} \\
P(k) &= \left[I - K(k)\varphi^T(k-1)\right]P(k-1)/\lambda \\
&\Downarrow \\
&\hat{A}, \hat{B} \\
&\Downarrow \\
\hat{B}\hat{T} &= B_m A_o \\
\hat{A}\hat{R} + \hat{B}\hat{S} &= A_m A_o
\end{aligned}
\tag{2.4-2}$$

$B_m$  e  $A_m$  são respectivamente o numerador e o denominador do modelo de referência que dá o desempenho desejado do sistema controlado. As Equações (2.4-2) revelam que a cada iteração exige-se o cálculo de uma equação *diofantina* análoga à Equação (2.2-11) para se obter os novos valores para os coeficientes dos polinômios do controlador, o que resulta na necessidade de resolução de um sistema linear de mesma ordem que o controlador. Este ponto deve ser levado em conta pelo projetista, pois o número de operações cresce com o quadrado da ordem do sistema, e a taxa de aquisição do sistema impõe um limite no tempo de execução disponível para se processar os dados. Para o controlador polinomial adaptativo ficam mantidas todas as restrições aplicáveis ao método de estimação recursiva e ao projeto de controladores polinomiais não adaptativos, acrescentando-se o fato de ser uma boa medida iniciar a execução do algoritmo com um conjunto de parâmetros próximo ao conjunto de parâmetros real do sistema objetivando reduzir o transiente de estimação para que não influencie na estabilidade do controlador.

## Capítulo 3

### Identificação do Sistema

O sistema experimental completo em estudo tem quatro graus de liberdade, sendo formado por dois motores e dois rotores conectados por duas correias e um eixo, mas para o fim de estimação de parâmetros físicos será feita a modelagem por partes, que resulta na construção do modelo final desejado, e que tem seus parâmetros físicos identificáveis experimentalmente.

Será mostrado o modelamento dos componentes de um grau de liberdade isoladamente, ou seja, um motor CC e um rotor. Nessa seção, também será mostrado o modelo de primeira ordem utilizado para o controle de um sistema reduzido, sem eixo.

Em seguida está a seção na qual serão mostrados os componentes de dois graus de liberdade, que são os conjuntos formados por: i) um motor e um rotor conectados por uma correia; ii) dois rotores conectados pelo eixo. Esse procedimento possibilita a estimação isolada dos parâmetros dinâmicos (coeficientes de atrito de rolamento dos motores e dos rotores, frequência natural de acoplamento entre motor e rotor através de uma correia, e frequência natural de acoplamento entre os dois rotores conectados pelo eixo).

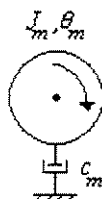
A próxima seção descreve a modelagem do sistema completo, com quatro graus de liberdade, e que tem seus parâmetros expressos por valores mensuráveis através dos sistemas simplificados de um e dois graus de liberdade, além do modelo obtido por estimação recursiva a partir de dados coletados experimentalmente.

#### 3.1. Modelos de Primeira Ordem

Partindo para a identificação do sistema a partir de blocos geradores menores, serão apresentadas na sequência as identificações dos modelos de primeira ordem, que são o ponto de partida para a estimação dos parâmetros do conjunto a ser modelado. O diagrama geral da bancada, no qual estão incluídos os componentes básicos aqui descritos está mostrado no Capítulo 1, Figura 1.1-1.

### 3.1.1. Motor

A seguir está o diagrama que descreve o motor CC com ímã permanente utilizado, sendo suposto que a dissipação de rolamento ocorre por atrito viscoso:



**Figura 3.1-1** – Motor e parâmetros.

No sistema internacional de unidades (SI), a constante  $K_e$ , que relaciona a diferença de potencial entre os terminais do motor CC em rotação quando não há passagem de corrente, é numericamente igual à constante de torque  $K_t$  do motor, que relaciona o torque que movimenta o motor com a corrente que circula por ele.

$$\begin{cases} V_m = K_e \omega_m \\ T_m = K_t i \end{cases} \quad (3.1-1)$$

Acionando-se o motor através de um segundo motor para manter sua rotação constante, embora sem circulação de corrente, e medindo-se a ddp entre seus terminais, obtém-se:

| $\omega_m$ (rad/s) | $V_m$ (V) |
|--------------------|-----------|
| 11                 | 0,55      |
| 19                 | 0,95      |
| 30                 | 1,51      |
| 40                 | 2,01      |

**Tabela 3.1-1** – Tensão nos terminais do motor.

Dos dados da tabela acima calcula-se a constante elétrica do motor como sendo:

$$K_e = 0,0503 \frac{V_s}{\text{rad}}$$

Ou seja,

$$K_t = 0,0503 \frac{\text{Nm}}{\text{A}}$$

Fazendo-se o motor girar com uma rotação constante devido ao acionamento através da fonte de corrente pode-se determinar seu amortecimento, pois o módulo do torque da corrente de acionamento é igual ao módulo do torque dissipativo de rolamento:

$$T_i = T_r \Rightarrow K_t i = c_m \omega_m \Rightarrow c_m = \frac{K_t i}{\omega_m} \quad (3.1-2)$$

Para uma velocidade de regime de 42rad/s com corrente 1,22A, obtém-se:

$$c_m = 1,954 \times 10^{-4} \text{ Nms}$$

Partindo dos valores estimados para a constante de torque e para o amortecimento, e a partir de testes de parada, pode-se determinar o fator de amortecimento do motor que possibilita a obtenção do momento de inércia do motor:

$$J_m \dot{\omega}_m = -c_m \omega_m$$

$$\Downarrow$$

$$\omega_m(t) = \kappa \exp(-\alpha_m t), \quad \alpha_m = \frac{c_m}{J_m} \quad (3.1-3)$$

$$\Downarrow$$

$$J_m = \frac{c_m}{\alpha_m}$$

O fator de amortecimento foi obtido a partir de teste de parada com o motor, cujo resultado é apresentado abaixo:

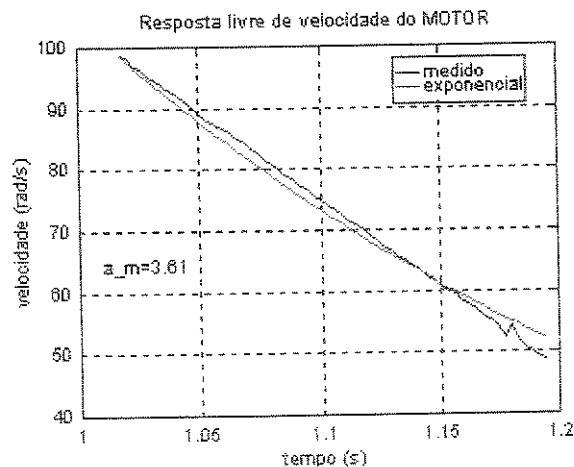


Figura 3.1-2 – Teste de parada do motor em torno da rotação de regime desejada.

A figura acima mostra o decaimento da velocidade do motor quando submetido a um teste de parada. O fator de amortecimento obtido para a resposta livre do motor, com valor  $\alpha_m = 3,61 \text{ s}^{-1}$ , ajusta a curva experimental em toda a região de interesse localizada em torno da rotação normal desejada.

E assim:

$$J_m = \frac{c_m}{\alpha_m} = 5,41 \times 10^{-5} \text{ kg.m}^2$$

Com isso ficam determinados todos os elementos que caracterizam o motor e é possível, então, prosseguir com a estimação dos parâmetros do sistema.

### 3.1.2. Rotor

De forma análoga ao motor, supõe-se que o rotor também tem dissipação de rolamento por atrito viscoso, como pode ser visto abaixo:

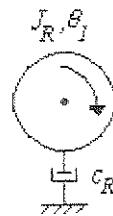


Figura 3.1-3 – Rotor e seus parâmetros.

Ao contrário do caso do motor, o rotor tem o momento de inércia conhecido, restando determinar o fator de amortecimento de rotação, o que pode ser feito com um teste de parada, medindo-se a resposta livre do rotor:

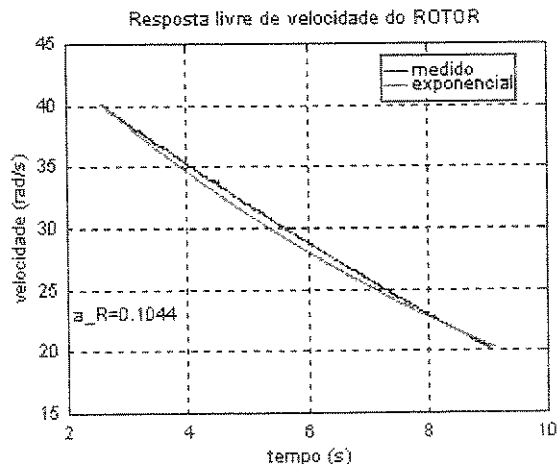


Figura 3.1-4 – Teste de parada do rotor em torno da rotação de regime desejada.

A figura acima mostra o decaimento da velocidade do rotor quando submetido a um teste de parada. O fator de amortecimento obtido para a resposta livre do rotor, com valor  $c_R = 0,1044 \text{ s}^{-1}$ , ajusta a curva experimental em toda a região de interesse localizada em torno da rotação normal desejada.

Partindo dos valores estimados para o fator de amortecimento e para o momento de inércia, obtém-se o amortecimento do rotor:

$$\begin{aligned}
 J_R \dot{\omega}_1 &= -c_R \omega_1 \\
 \Downarrow \\
 \omega_1(t) &= \kappa \exp(-a_R t), \quad a_R = \frac{c_R}{J_R} \quad (3.1-4) \\
 \Downarrow \\
 c_R &= J_R \cdot a_R
 \end{aligned}$$

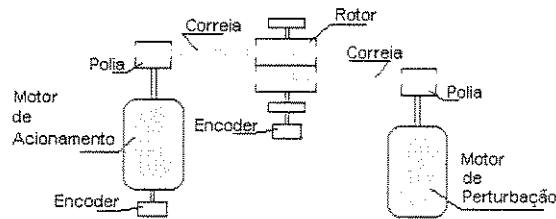
E assim, sabendo que  $J_R = 3,2 \times 10^{-3} \text{ kg.m}^2$  e que  $a_R = 1,044 \times 10^{-1}$ , calcula-se:

$$c_R = 3,34 \times 10^{-4} \text{ Nms}$$

Estes valores obtidos representam o conhecimento básico da dinâmica do sistema rotativo a ter seus parâmetros estimados.

### 3.1.3. Modelo Simplificado com Dois Motores e um Rotor

Conectando dois motores a um rotor através de correias, pode-se construir um modelo de três graus de liberdade, mas se for suposto que as correias são rígidas chega-se a um modelo com um grau de liberdade e com duas entradas de torque. Este modelo simplificado foi utilizado para testar e implementar os algoritmos de controle e estimação. O diagrama deste sistema pode ser visto abaixo:



**Figura 3.1-5** – Sistema com três graus de liberdade.

O sistema da figura 3.1-5 pode ser modelado como:

$$J_{eq} \dot{\omega}_1 = -c_{eq} \omega_1 + K_{teq} i + K_{teq} i_p \quad (3.1-5)$$

sendo:

$J_{eq}$  - momento de inércia equivalente no rotor

$c_{eq}$  - amortecimento equivalente no rotor

$K_{teq}$  - constante de torque equivalente no rotor

$\omega_1$  - velocidade angular do rotor

$i$  - corrente no motor de acionamento

$i_p$  - corrente no motor de perturbação

Esse sistema acima é identificado como um sistema de primeira ordem com duas entradas de corrente e uma saída de rotação pelo rotor, e pode ser reescrito como:

$$\begin{aligned}\dot{\omega}_1 &= -\frac{c_{eq}}{J_{eq}}\omega_1 + \frac{K_{teq}}{J_{eq}}i + \frac{K_{teq}}{J_{eq}}i_p \\ &\Downarrow \\ \dot{\omega}_1 &= -a\omega_1 + bi + bi_p\end{aligned}\quad (3.1-6)$$

Partindo da situação de regime, com o rotor a uma velocidade de rotação de 70 rad/s, o sistema teve seus parâmetros *estimados* através do método Simplex não linear, chegando-se à seguinte função de transferência entre entrada de corrente do motor de acionamento e saída em velocidade no rotor:

$$G(s) = \frac{b}{s + a} = \frac{9,4}{s + 0,14} \quad (3.1-7)$$

Esta função de transferência tem um pólo real e nenhum zero, ou seja, o sistema é modelado como um filtro passa-baixa de primeira ordem. Abaixo encontram-se resultados desta estimação comparados à resposta real da planta:

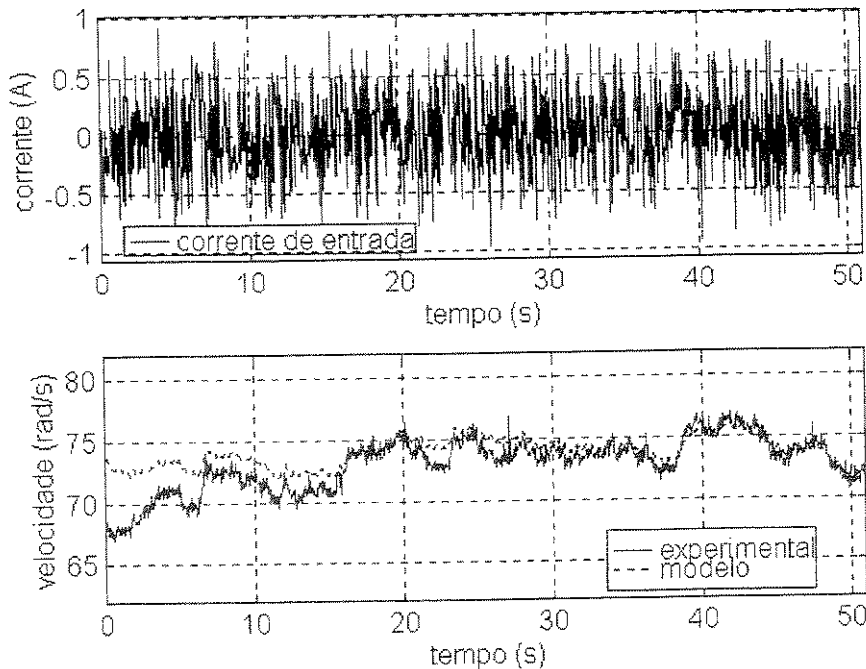


Figura 3.1-6 – Resposta do sistema simplificado a perturbação aleatória até 20Hz.

A Figura 3.1-6 mostra a resposta da planta a corrente de entrada aleatória entre 0Hz e 20Hz e a resposta em velocidade do rotor tanto para o modelo simulado (em vermelho)

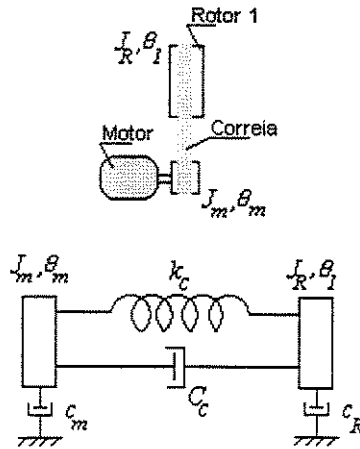
quanto para o sistema real (em azul). O modelo escolhido representa o sistema, mas os efeitos da simplificação podem ser vistos através do fato de o sinal de velocidade medido apresentar oscilações de baixa amplitude e alta frequência modulando a resposta, oscilações essas que não são observadas na resposta do modelo. Para efeito de controle, essas oscilações são consideradas perturbações na medida da saída.

### **3.2. Modelos de Segunda Ordem**

Uma vez elaborados os modelos para os elementos básicos da bancada, é possível começar a fazer associações entre eles de forma a ser possível chegar a modelos de conjunto. O primeiro modelo de conjunto a ser montado é o modelo para um motor e um rotor, conectados por uma correia. Para esta nova situação, os valores dos coeficientes de atrito de rolamento do motor e do rotor usados são aqueles obtidos na seção 3.1. A correia acrescenta fatores de atrito, e é modelada em termos de uma constante elástica linear. O efeito da correia é distribuído pelo sistema rotativo na forma de constantes elásticas torcionais equivalentes e amortecimentos torcionais equivalentes. O acoplamento entre motor e rotor, através da correia tem caráter oscilatório, fator este que será explorado na obtenção de valores experimentais para o modelo.

#### **3.2.1. O Conjunto Motor, Rotor e Correia**

O primeiro sistema de segunda ordem é descrito esquematicamente na Figura 3.2-1, que contém tanto informações da disposição geométrica quanto das propriedades mecânicas associadas aos componentes.



**Figura 3.2-1** – Sistema formado por um motor, uma correia e um rotor.

Essa figura mostra como estão ligados o motor, a correia e o rotor, e como esses componentes serão denominados no decorrer do texto. É suposto um modelo de amortecimento viscoso para a dissipação de energia introduzida pela correia no sistema, correia esta que está modelada por uma constante elástica linear, mas que se reflete no movimento do motor e do rotor como uma constante elástica torcional equivalente.

O sistema de equações que descrevem o sistema modelado é apresentado a seguir, e nele estão contabilizadas as interações entre os dois componentes através da conexão elástica da correia e das perdas por atrito envolvidas nessa conexão e no rolamento:

$$\begin{cases} J_m \ddot{\theta}_m = -c_m \dot{\theta}_m - k_c \theta_m R_m R_m + k_c \theta_1 R_R R_m - C_c \dot{\theta}_m R_m R_m + C_c \dot{\theta}_1 R_R R_m + K_i i \\ J_R \ddot{\theta}_1 = -c_R \dot{\theta}_1 - k_c \theta_1 R_R R_R + k_c \theta_m R_m R_R - C_c \dot{\theta}_1 R_R R_R + C_c \dot{\theta}_m R_m R_R \end{cases} \quad (3.2-1)$$

sendo  $k_c \theta_1 R_R$  a força elástica na correia associada ao deslocamento  $\theta_1$ , e  $k_c \theta_m R_m R_R$  o torque no rotor devido à força elástica na correia associada ao deslocamento  $\theta_m$ .

Define-se agora um conjunto de parâmetros que serão identificados sempre que possível neste e em outros modelamentos utilizados no decorrer deste trabalho, e que possuem sentido físico relevante para o entendimento das hipóteses feitas durante a determinação de valores experimentais:

$$\begin{aligned} n &\equiv \frac{R_R}{R_m} \\ K_{cR} &\equiv k_c R_R^2 \\ K_{cm} &\equiv k_c R_m^2 \\ c_{cR} &\equiv C_c R_R^2 \\ c_{cm} &\equiv C_c R_m^2 \end{aligned} \quad (3.2-2)$$

sendo  $K_{cR}$  e  $K_{cm}$  as constantes elásticas torcionais equivalentes da correia atuando respectivamente no motor e no rotor. De forma análoga,  $c_{cR}$  e  $c_{cm}$  são as constantes para o amortecimento viscoso atuando no rotor e no motor. A relação de transmissão,  $n$ , é o fator de ponderação entre o deslocamento angular do motor e o do rotor.

Reorganizando o sistema (3.2-1), chega-se ao seguinte sistema de equações para aceleração angular:

$$\begin{cases} \ddot{\theta}_m = -\frac{c_m}{J_m}\dot{\theta}_m + n\frac{K_{cm}}{J_m}\left(\theta_1 - \frac{1}{n}\theta_m\right) - \frac{c_{cm}}{J_m}\dot{\theta}_m + n\frac{c_{cm}}{J_m}\dot{\theta}_1 + \frac{K_t}{J_m}i \\ \ddot{\theta}_1 = -\frac{c_R}{J_R}\dot{\theta}_1 - \frac{K_{cR}}{J_R}\left(\theta_1 - \frac{1}{n}\theta_m\right) - \frac{c_{cR}}{J_R}\dot{\theta}_1 + \frac{c_{cR}}{J_R}\frac{1}{n}\dot{\theta}_m \end{cases} \quad (3.2-3)$$

É interessante reescrever o sistema de equações (3.2-3) na forma de modelo de estados, e isso pode ser feito da seguinte maneira:

$$\dot{\bar{x}} = A\bar{x} + Bi \quad (3.2-4)$$

sendo os estados dados por:

$$\bar{x} = \begin{bmatrix} \theta_{tm} & \frac{1}{n}\dot{\theta}_m & \dot{\theta}_1 \end{bmatrix}^T \quad (3.2-5)$$

$$\theta_{tm} \equiv \left(\theta_1 - \frac{1}{n}\theta_m\right) \quad (3.2-6)$$

As matrizes desse modelo são escritas em termos dos parâmetros físicos do sistema:

$$A = \begin{bmatrix} 0 & -1 & 1 \\ \frac{K_{cm}}{J_m} & -\frac{(c_m + c_{cm})}{J_m} & \frac{c_{cm}}{J_m} \\ -\frac{K_{cR}}{J_R} & \frac{c_{cR}}{J_R} & -\frac{(c_R + c_{cR})}{J_R} \end{bmatrix} \quad (3.2-7)$$

$$B = \begin{bmatrix} 0 & \frac{K_t}{nJ_m} & 0 \end{bmatrix}^T \quad (3.2-8)$$

Neste ponto deve-se observar que embora o conjunto de equações 3.2-3 seja um conjunto com duas equações diferenciais ordinárias de segunda ordem, o que implicaria num sistema de quarta ordem para o modelamento no espaço de estados, a escolha de um grau de liberdade relativo  $\theta_{tm}$  reduz a ordem do sistema ao excluir o modo de corpo rígido. No presente caso restaram um modo com frequência natural nula, restando um modo com raiz real pura e um modo com raiz complexa.

Partindo das equações (3.2-3) obtém-se para  $\ddot{\theta}_{tm} = \left( \ddot{\theta}_1 - \frac{1}{n} \ddot{\theta}_m \right)$ , que é a aceleração angular relativa entre motor e rotor, a seguinte equação:

$$\ddot{\theta}_{tm} + \left( \frac{c_{cR}}{J_R} + \frac{c_{cm}}{J_m} \right) \dot{\theta}_{tm} + \left( \frac{K_{cR}}{J_R} + \frac{K_{cm}}{J_m} \right) \theta_{tm} + \frac{c_R}{J_R} \dot{\theta}_1 - \frac{c_m}{J_m} \frac{1}{n} \dot{\theta}_m = 0 \quad (3.2-9)$$

Definindo  $\sigma_{tm} = \frac{c_{cR}}{J_R} + \frac{c_{cm}}{J_m}$  e  $\omega_{tm}^2 = \frac{K_{cR}}{J_R} + \frac{K_{cm}}{J_m}$ , pode-se reescrever (3.2-9) como

$$\ddot{\theta}_{tm} + \sigma_{tm} \dot{\theta}_{tm} + \omega_{tm}^2 \theta_{tm} + a_R \dot{\theta}_1 - a_m \frac{1}{n} \dot{\theta}_m = 0 \quad (3.2-10)$$

e observando os valores experimentais medidos para os coeficientes  $c_{cR}$  e  $c_{cm}$ , tem-se para os coeficientes da Equação 3.2-10 os valores:

$$\begin{aligned} \sigma_{tm} &= 313,11 \\ a_R &= 1,044 \times 10^{-1} \\ a_m &= 3,61 \end{aligned} \quad (3.2-11)$$

e assim chega-se à equação de movimento simplificada para o par motor-rotor, dependente unicamente da variável de deslocamento relativo entre motor e rotor:

$$\ddot{\theta}_{tm} + \sigma_{tm} \dot{\theta}_{tm} + \omega_{tm}^2 \theta_{tm} = 0 \quad (3.2-12)$$

que tem solução geral da forma:

$$\theta_{tm}(t) = e^{-\sigma_{tm} t} [A \sin(\omega_{dm} t) + B \cos(\omega_{dm} t)] \quad (3.2-13)$$

Como medida auxiliar, é necessário fazer a determinação dos parâmetros equivalentes de momento de inércia e amortecimento entre motor e rotor, respectivamente  $J_{mR}$ , o momento de inércia equivalente do conjunto, e  $c_{mR}$ , o fator de amortecimento do conjunto.

Supondo uma transmissão não elástica através da correia, pode-se escrever a seguinte equação para torque no rotor durante um experimento de parada sem excitação externa:

$$J_{mR} \ddot{\theta}_1 = c_{mR} \dot{\theta}_1 \quad (3.2-14)$$

sendo cada termo descrito por:

$$\begin{cases} J_{mR} \ddot{\theta}_1 = J_R \ddot{\theta}_1 + J_m \frac{\ddot{\theta}_1}{R_m} R_m \\ c_{mR} \dot{\theta}_1 = c_R \dot{\theta}_1 + c_m \frac{\dot{\theta}_m}{R_m} R_m + C_c R_m \dot{\theta}_m R_R + C_c R_R \dot{\theta}_1 R_R \end{cases} \quad (3.2-15)$$

Para a condição de correia não elástica, tem-se a relação entre posição angular no eixo e posição angular no motor descrita abaixo:

$$\theta_m = n\theta_1 = \frac{R_R}{R_m}\theta_1 \quad (3.2-16)$$

e assim:

$$\begin{cases} J_{mR}\ddot{\theta}_1 = J_R\ddot{\theta}_1 + J_m \frac{R_R^2}{R_m^2}\ddot{\theta}_1 = J_R\ddot{\theta}_1 + n^2 J_m \ddot{\theta}_1 \\ c_{mR}\dot{\theta}_1 = c_R\dot{\theta}_1 + c_m \frac{R_R^2}{R_m^2}\dot{\theta}_1 + 2C_c R_R^2 \dot{\theta}_1 = (c_R + n^2 c_m + 2c_{cR})\dot{\theta}_1 \end{cases} \quad (3.2-17)$$

ou, igualando coeficientes das derivadas da posição angular do motor:

$$\begin{cases} J_{mR} = J_R + n^2 J_m \\ c_{mR} = (c_R + n^2 c_m + 2c_{cR}) \end{cases} \quad (3.2-18)$$

O momento de inércia do conjunto pode ser calculado a partir dos valores conhecidos das inércias envolvidas, sendo que, a partir do fator de amortecimento calculado, pode-se determinar o fator de amortecimento da correia.

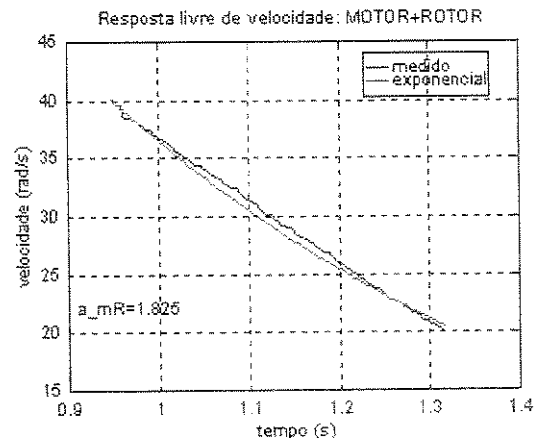
**Simbologia:**

- $J_R$  - momento de inércia do rotor.
- $J_{mR}$  - momento de inércia equivalente do conjunto motor e rotor.
- $k_c$  - constante elástica linear da correia.
- $K_{cR}$  - constante elástica torcional equivalente da correia no rotor.
- $K_{cm}$  - constante elástica torcional equivalente da correia no motor.
- $c_{mR}$  - amortecimento equivalente para motor, correia e rotor.
- $c_R$  - amortecimento de rotação do rotor.
- $c_m$  - amortecimento de rotação do motor.
- $\omega_{nim}$  - frequência de oscilação natural do sistema
- $\omega_{dtm}$  - frequência de oscilação amortecida do sistema
- $\zeta_{tm}$  - fator de amortecimento do sistema

Dando continuidade, passa-se à utilização do modelamento feito, fazendo-se a determinação experimental do amortecimento e da frequência natural do conjunto. Como

parte do amortecimento advém isoladamente do motor e do rotor, e esses valores já estão determinados, a medição do amortecimento do conjunto resulta na obtenção do fator de amortecimento atribuído à correia. Uma vez determinado o amortecimento do conjunto, pode-se partir para a determinação da frequência natural do sistema acoplado, e isso permite que se determine o valor da constante elástica da correia.

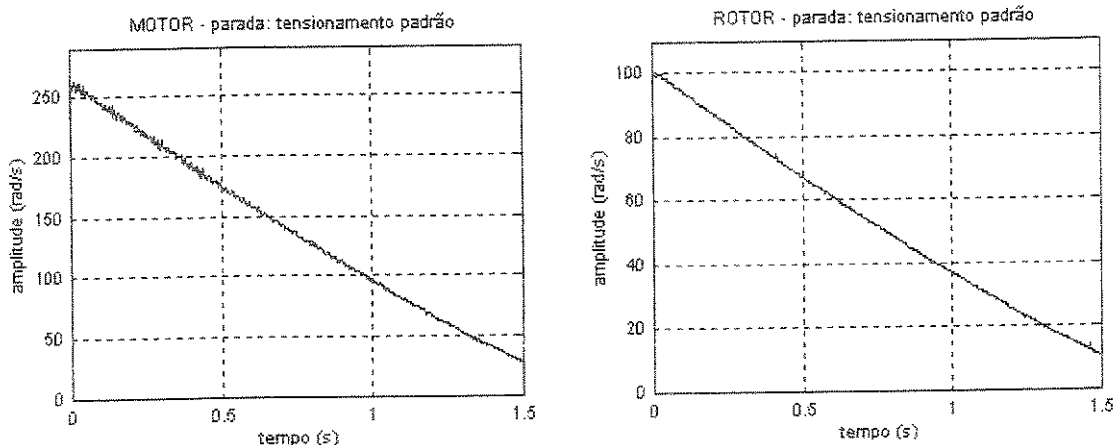
Os gráficos abaixo mostram os resultados de experimentos para determinar  $\alpha_{mR}$  e  $\omega_{dtn}$  a partir da medida da resposta livre do conjunto, tomando-se valores entre 20 rad/s e 40 rad/s, que são velocidades em torno da velocidade de regime desejada de 30 rad/s.



**Figura 3.2-2** – Resposta livre do conjunto motor, correia e rotor, *medida no rotor*.

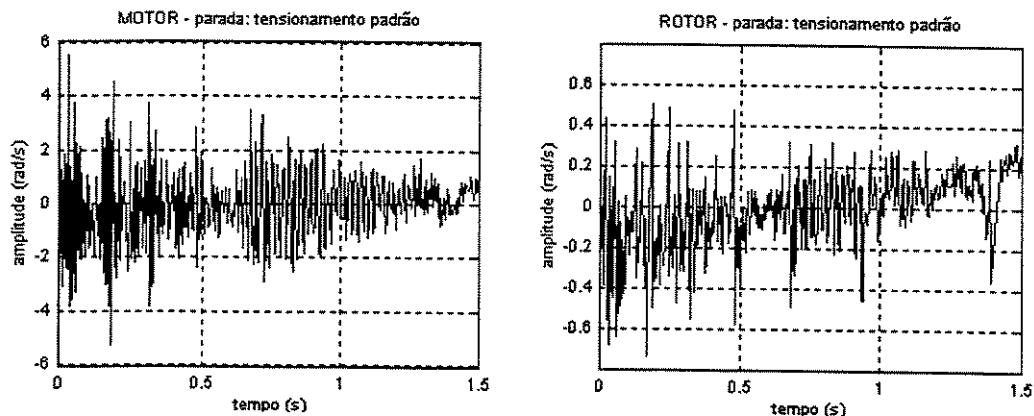
O ajuste da curva para essa medida experimental resultou num valor de amortecimento para o conjunto de  $\alpha_{mR}=1,825s^{-1}$ . Percebe-se a concordância entre os valores medidos e ajustados.

A frequência natural amortecida ( $\omega_{dtn}$ ) do conjunto foi obtida a partir de medidas de parada, com o sistema tensionado de forma padrão. Foram utilizados os mesmos conjuntos de dados que resultaram no valor para o amortecimento. Nesses conjuntos pode-se notar que, sobreposto ao sinal majoritário de decaimento de velocidade, há um sinal oscilatório.



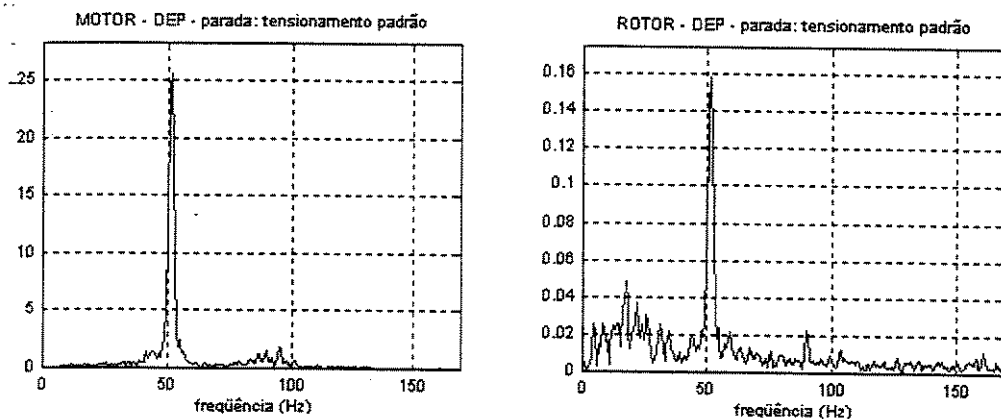
**Figura 3.2-3** – Respostas livres do conjunto, medidas no motor e no rotor.

Dos sinais de resposta livre retira-se a média móvel e obtém-se os seguintes sinais, os quais contêm a informação oscilatória numa forma evidenciada:



**Figura 3.2-4** – Respostas livres do conjunto retirando-se médias móveis.

A amplitude da oscilação no rotor é 10 vezes menor que a amplitude no motor, mas a informação temporal não é a melhor técnica de análise para esses resultados. Fazendo-se os auto-espectros dos sinais da Figura 3.2-4, obtém-se:



**Figura 3.2-5** – Auto-espectros das respostas livres sem média móvel.

Os resultados da Figura 3.2-5 mostram de forma clara a existência de uma frequência natural característica, como era esperado pelo modelo, e que parecia clara na informação temporal. Os gráficos mostram que tanto no motor quanto no rotor existe um pico pronunciado na frequência 51Hz.

Os valores obtidos com o procedimento estão listados a seguir:

$$k_c = 3950 \text{ N/m}$$

$$K_{cm} = 6,88 \text{ Nm}$$

$$K_{cR} = 43 \text{ Nm}$$

$$c_{cR} = 2,45 \times 10^{-3}$$

$$\omega_{dtm} = 323,58 \text{ rad/s}$$

$$\omega_{ntm} = 359,07 \text{ rad/s}$$

$$f_{ntm} = 57,15 \text{ Hz}$$

$$\zeta_m = 0,436$$

O modelo de estados numérico é escrito como:

$$A = \begin{bmatrix} 0 & -1.0000e+000 & 1.0000e+000 \\ 1.2707e+005 & -1.2371e+001 & 8.7614e+000 \\ -1.3434e+004 & 9.2624e-001 & -1.0306e+000 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 3.7179e+002 & 0 \end{bmatrix}$$

### 3.2.2. O Conjunto Rotores e Eixo

Este segundo conjunto reduzido do problema experimental vai permitir a obtenção dos valores característicos que compõem o eixo. Assim como para o motor e o rotor, os dois rotores conectados pelo eixo possuem uma resposta dinâmica oscilatória que será utilizada para a obtenção de parâmetros do modelo. O eixo é suposto como tendo momento de inércia desprezível em relação aos rotores, entrando no sistema como uma mola torcional amortecida.

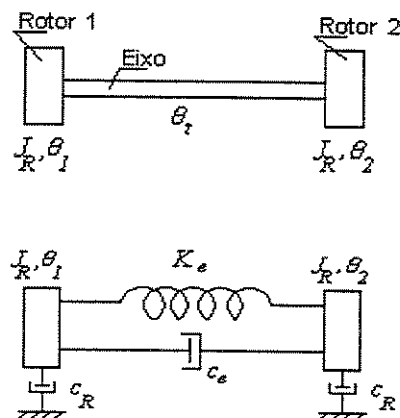


Figura 3.2-6 – Sistema formado por dois rotores e um eixo.

O sistema formado por dois rotores e um eixo é descrito esquematicamente nas figuras acima, que contêm tanto informações geométricas quanto propriedades mecânicas associadas aos componentes.

O sistema modelado pode ser descrito pelo sistema de equações a seguir, e contém as interações entre os componentes envolvidos:

$$\begin{cases} J_R \ddot{\theta}_1 = -c_R \dot{\theta}_1 - K_e \theta_1 + K_e \theta_2 - c_e \dot{\theta}_1 + c_e \dot{\theta}_2 & (i) \\ J_R \ddot{\theta}_2 = -c_R \dot{\theta}_2 - K_e \theta_2 + K_e \theta_1 - c_e \dot{\theta}_2 + c_e \dot{\theta}_1 & (ii) \end{cases} \quad (3.2-19)$$

Subtraindo-se uma equação da outra no sistema de equações (3.2-19), obtém-se:

$$J_R (\ddot{\theta}_1 - \ddot{\theta}_2) = -c_R (\dot{\theta}_1 - \dot{\theta}_2) - 2K_e (\theta_1 - \theta_2) - 2c_e (\dot{\theta}_1 - \dot{\theta}_2) \quad (3.2-20)$$

e definindo a posição angular relativa entre os rotores (deformação do eixo) como

$$\theta_i = (\theta_1 - \theta_2) \quad (3.2-21)$$

obtém-se a seguinte equação para oscilações livres de torque, partindo de condição inicial de deformação do eixo:

$$J_R \ddot{\theta}_i + (c_R + 2c_e) \dot{\theta}_i + 2K_e \theta_i = 0 \quad (3.2-22)$$

A Equação 3.2-22 é uma equação linear homogênea com parâmetros concentrados e invariantes no tempo, sendo sua solução geral conhecida, podendo ser escrita como:

$$\theta_i(t) = e^{-\zeta \omega_n t} [A \sin(\omega_d t) + B \cos(\omega_d t)] \quad (3.2-23)$$

sendo

$$\begin{cases} \omega_n = \frac{2K_e}{J_R} \\ \zeta \omega_n = \frac{c_R + 2c_e}{2J_R} \\ \omega_d = \omega_n \sqrt{1 - \zeta^2} \end{cases} \quad (3.2-24)$$

O sistema de equações para aceleração angular é:

$$\begin{cases} \ddot{\theta}_1 = -\frac{c_R}{J_R} \dot{\theta}_1 - \frac{K_e}{J_R} (\theta_1 - \theta_2) - \frac{c_e}{J_R} \dot{\theta}_1 + \frac{c_e}{J_R} \dot{\theta}_2 \\ \ddot{\theta}_2 = -\frac{c_R}{J_R} \dot{\theta}_2 + \frac{K_e}{J_R} (\theta_1 - \theta_2) - \frac{c_e}{J_R} \dot{\theta}_2 + \frac{c_e}{J_R} \dot{\theta}_1 \end{cases} \quad (3.2-25)$$

O modelo de estados contínuo para o sistema é definido a seguir:

$$\dot{\vec{x}} = A\vec{x} + B\vec{i} \quad (3.2-26)$$

sendo

$$\bar{x} = [\theta_t \quad \dot{\theta}_1 \quad \dot{\theta}_2]^T \quad (3.2-28)$$

$$A = \begin{bmatrix} 0 & 1 & -1 \\ -\frac{K_e}{J_R} & -\frac{(c_R + c_e)}{J_R} & \frac{c_e}{J_R} \\ \frac{K_e}{J_R} & \frac{c_e}{J_R} & -\frac{(c_R + c_e)}{J_R} \end{bmatrix} \quad (3.2-29)$$

$$B = \begin{bmatrix} 0 & \frac{1}{J_R} & 0 \end{bmatrix}^T \quad (3.2-30)$$

Simbologia:

$J_R$  - momento de inércia de um rotor

$K_e$  - constante elástica torcional do eixo

$c_R$  - amortecimento de rotação de um rotor

$c_e$  - amortecimento de torção do eixo

$\theta_1$  - rotação absoluta do rotor 1

$\theta_2$  - rotação absoluta do rotor 2

$\theta_t$  - rotação relativa entre rotor 1 e rotor 2

$\omega_n$  - frequência de oscilação natural do sistema

$\omega_d$  - frequência de oscilação amortecida do sistema

$\zeta$  - fator de amortecimento do sistema

A determinação da constante elástica do eixo foi feita a partir de medidas da frequência amortecida  $\omega_d$  e do amortecimento  $\zeta\omega_n$ . O experimento teve como procedimento a excitação impulsiva no rotor 1 feita com um martelo, medindo-se a deformação através do extensômetro, e processando a *resposta livre* obtida após o término da excitação. Foi ajustado um modelo que partindo das mesmas condições iniciais dos dados medidos reproduzisse o comportamento oscilatório com a forma dada pela Equação 3.2-24.

Os resultados obtidos podem ser observados nas figuras abaixo. Nos gráficos com escala de tempo, o traço vermelho mostra o sinal medido, enquanto o traço azul mostra o sinal ajustado. A linha verde é a linha do decaimento ajustado. Nos gráfico com escala de frequência está exibida a densidade espectral de energia do sinais utilizados para ajuste.

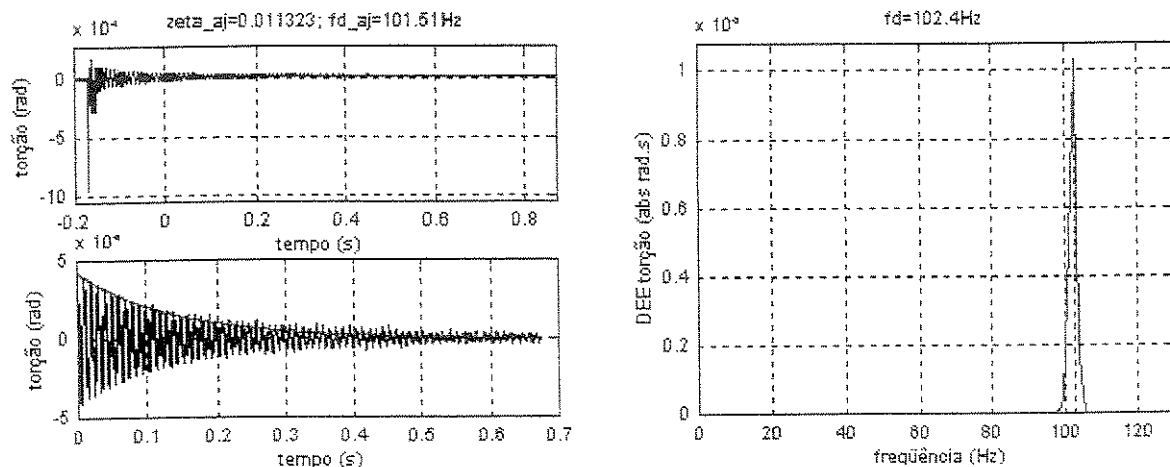


Figura 3.2-7 – Resposta livre de deformação no eixo – teste 1.

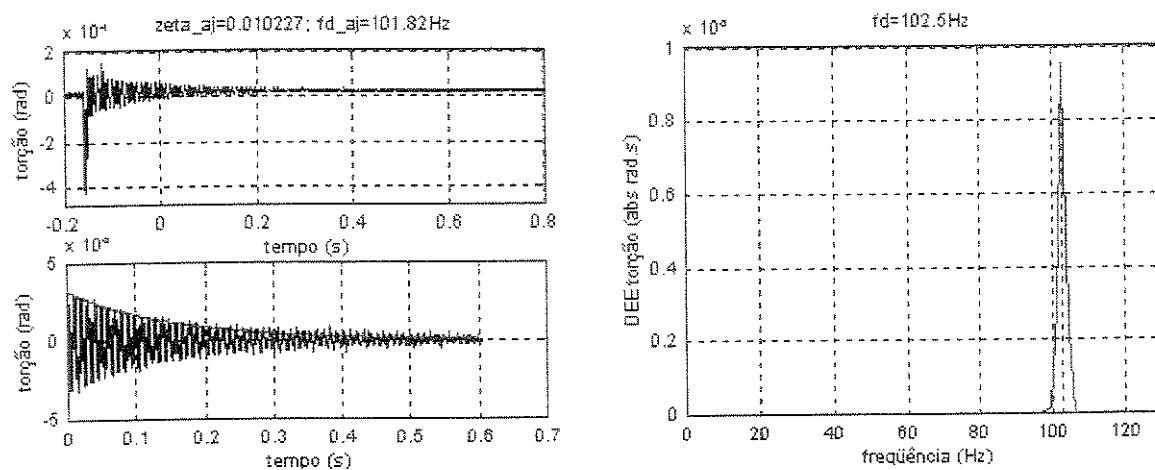


Figura 3.2-8 – Resposta livre de deformação no eixo – teste 2.

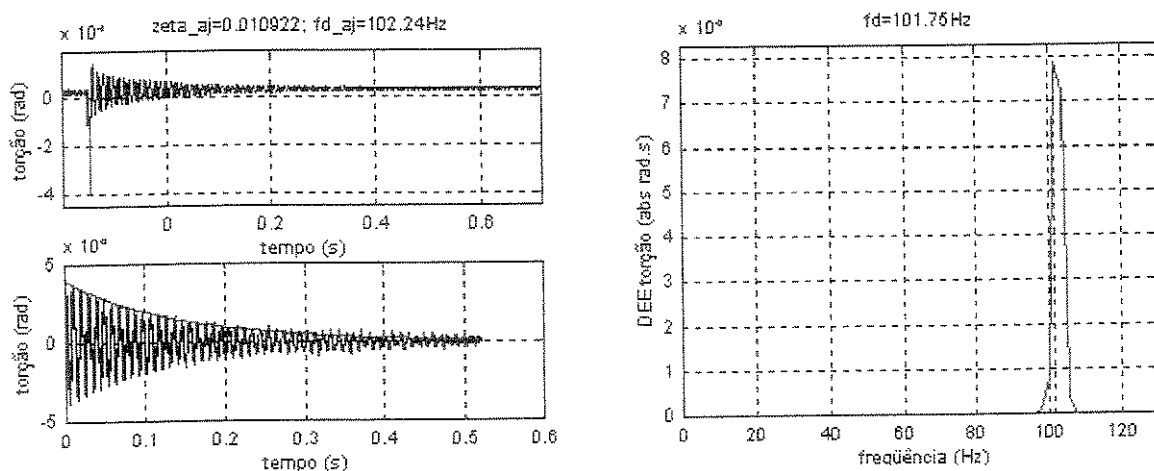


Figura 3.2-9 – Resposta livre de deformação no eixo – teste 3.

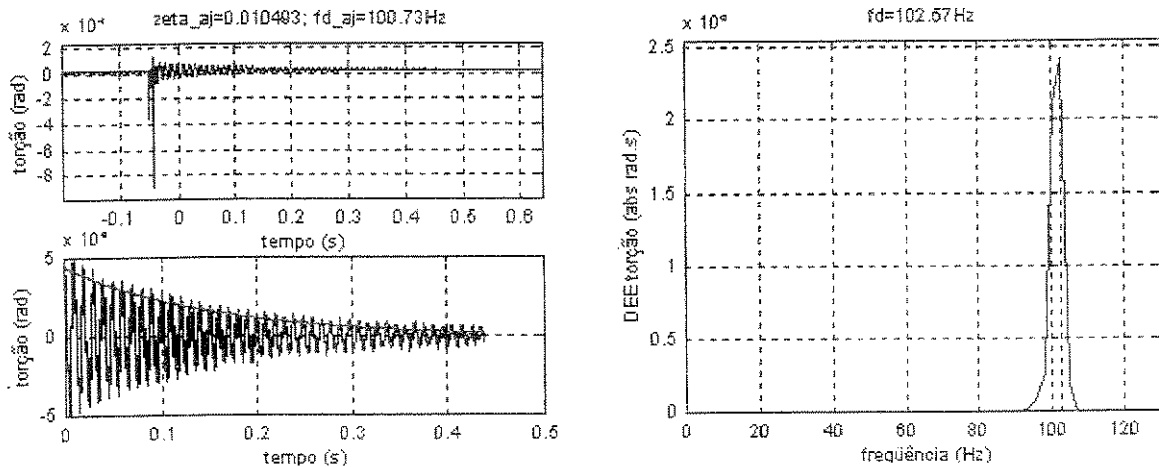


Figura 3.2-10 – Resposta livre de deformação no eixo – teste 4.

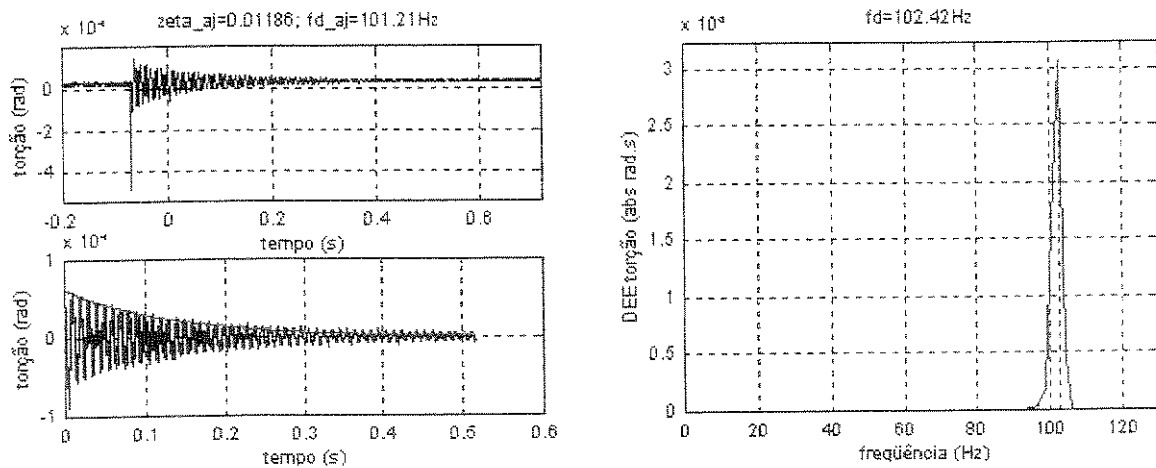


Figura 3.2-11 – Resposta livre de deformação no eixo – teste 5.

Visualmente, pode-se observar a concordância entre resultados experimentais e modelo estimado, sendo que a frequência média obtida para o conjunto é 101,5Hz e fator de amortecimento médio de  $1,1 \times 10^{-2}$ .

De acordo com a Equação 3.2-23, os dados acima correspondem à resposta de sistema com os seguintes coeficientes:

$$\zeta \omega_n = 6.993$$

$$\omega_d = 637.74 \text{ rad/s}$$

$$f_d = 101.50 \text{ Hz}$$

e com isso pode-se estimar os parâmetros físicos do modelo:

$$K_e = 650 \text{ Nm/rad}$$

$$c_e = 2.22 \times 10^{-2} \text{ Nms}$$

$$\omega_n = 637.78 \text{ rad/s}$$

$$f_n = 101.51 \text{ Hz}$$

$$\zeta = 1.097 \times 10^{-2}$$

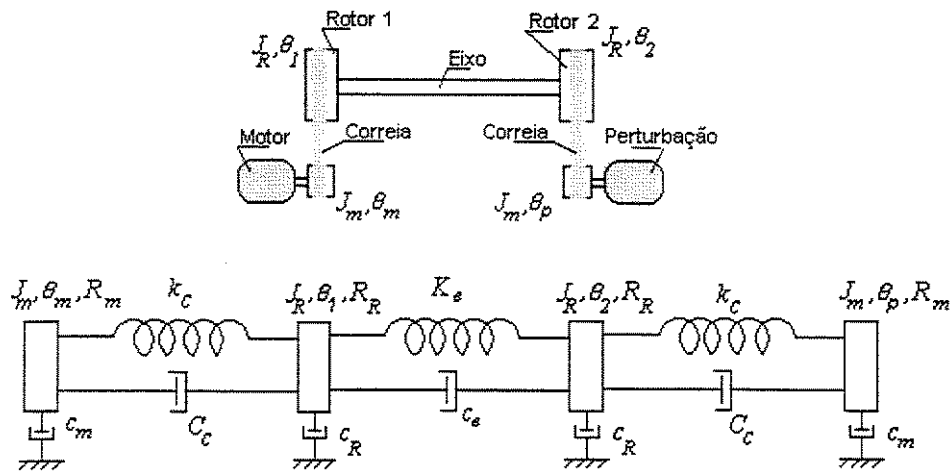
Substituindo-se os valores estimados no modelo de estados, chega-se às seguintes matrizes de estado numéricas, que podem ser utilizadas para representar o sistema:

$$A = \begin{bmatrix} 0 & -1.0000e+000 & 1.0000e+000 \\ -2.0310e+005 & -7.0442e+000 & 6.9398e+000 \\ 2.0310e+005 & 6.9398e+000 & -7.0442e+000 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 3.1250e+002 & 0 \end{bmatrix}^T$$

### 3.3. Modelo de Sétima Ordem

Tendo-se obtido os modelos de estágios reduzidos do sistema experimental, passa-se agora para a obtenção do modelo completo, adotando-se a mesma sistemática de montagem e agrupando-se os termos de forma a se obter um modelo com mesmos parâmetros físicos que os modelos reduzidos. Este modelo supõe duas entradas: uma de acionamento e uma de perturbação, sendo que esta última será utilizada para simular variações na demanda de torque. Especificamente para o caso implementado, supõe-se motores, correias e rotores idênticos.



**Figura 3.3-1** – Sistema formado por dois motores, dois rotores, duas correias e um eixo.

De maneira análoga às seções 3.2.1 e 3.2.2, constrói-se um sistema de equações diferenciais ordinárias de segunda ordem, com parâmetros concentrados e invariantes no tempo, equações essas que vão descrever o comportamento e a conexão entre os elementos do sistema experimental:

$$\begin{cases} \ddot{\theta}_m = -\frac{c_m}{J_m}\dot{\theta}_m + n\frac{K_{cm}}{J_m}\left(\theta_1 - \frac{1}{n}\theta_m\right) - \frac{c_{cm}}{J_m}\dot{\theta}_m + n\frac{c_{cm}}{J_m}\dot{\theta}_1 + \frac{K_t}{J_m}i \\ \ddot{\theta}_p = -\frac{c_m}{J_m}\dot{\theta}_p + n\frac{K_{cm}}{J_m}\left(\theta_2 - \frac{1}{n}\theta_p\right) - \frac{c_{cm}}{J_m}\dot{\theta}_p + n\frac{c_{cm}}{J_m}\dot{\theta}_2 + \frac{K_t}{J_m}i_p \\ \ddot{\theta}_1 = -\frac{c_R}{J_R}\dot{\theta}_1 - \frac{K_e}{J_R}(\theta_1 - \theta_2) - \frac{c_e}{J_R}\dot{\theta}_1 + \frac{c_e}{J_R}\dot{\theta}_2 - \frac{K_{cR}}{J_R}\left(\theta_1 - \frac{1}{n}\theta_m\right) - \frac{c_{cR}}{J_R}\dot{\theta}_1 + \frac{c_{cR}}{J_R}\frac{1}{n}\dot{\theta}_m \\ \ddot{\theta}_2 = -\frac{c_R}{J_R}\dot{\theta}_2 + \frac{K_e}{J_R}(\theta_1 - \theta_2) - \frac{c_e}{J_R}\dot{\theta}_2 + \frac{c_e}{J_R}\dot{\theta}_1 - \frac{K_{cR}}{J_R}\left(\theta_2 - \frac{1}{n}\theta_p\right) - \frac{c_{cR}}{J_R}\dot{\theta}_2 + \frac{c_{cR}}{J_R}\frac{1}{n}\dot{\theta}_p \end{cases} \quad (3.3-1)$$

sendo também nesse sistema isolados parâmetros, de forma que é possível utilizar o conhecimento sobre as partes isoladas já identificadas:

$$\begin{aligned} n &\equiv \frac{R_R}{R_m} \\ K_{cR} &\equiv k_c R_R^2 \\ K_{cm} &\equiv k_c R_m^2 \\ c_{cR} &\equiv C_c R_R^2 \\ c_{cm} &\equiv C_c R_m^2 \end{aligned} \quad (3.2-2)$$

sendo  $K_{cR}$  e  $K_{cm}$  as constantes elásticas torcionais equivalentes da correia atuando respectivamente no motor e no rotor. De forma análoga,  $c_{cR}$  e  $c_{cm}$  são os coeficientes de atrito viscoso atuando no rotor e no motor. A relação de transmissão,  $n$ , é o fator de ponderação entre o deslocamento angular do motor e o do rotor.

O modelo de estados para o sistema é definido a seguir:

$$\dot{\vec{x}} = A\vec{x} + B_1 i + B_2 i_p \quad (3.3-3)$$

sendo

$$\vec{x} = \begin{bmatrix} \theta_{im} & \theta_{ip} & \theta_t & \frac{1}{n}\dot{\theta}_m & \frac{1}{n}\dot{\theta}_p & \dot{\theta}_1 & \dot{\theta}_2 \end{bmatrix}^T \quad (3.3-4)$$

$$\theta_{im} \equiv \left(\theta_1 - \frac{1}{n}\theta_m\right), \quad \theta_{ip} \equiv \left(\theta_2 - \frac{1}{n}\theta_p\right), \quad \theta_t \equiv (\theta_1 - \theta_2) \quad (3.3-5)$$

$$A = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ \frac{K_{cm}}{J_m} & 0 & 0 & \frac{-(c_m + c_{cm})}{J_m} & 0 & \frac{c_{cm}}{J_m} & 0 \\ 0 & \frac{K_{cm}}{J_m} & 0 & 0 & \frac{-(c_m + c_{cm})}{J_m} & 0 & \frac{c_{cm}}{J_m} \\ -\frac{K_{cR}}{J_R} & 0 & -\frac{K_e}{J_R} & \frac{c_{cR}}{J_R} & 0 & \frac{-(c_R + c_e + c_{cR})}{J_R} & \frac{c_e}{J_R} \\ 0 & -\frac{K_{cR}}{J_R} & \frac{K_e}{J_R} & 0 & \frac{c_{cR}}{J_R} & \frac{c_e}{J_R} & \frac{-(c_R + c_e + c_{cR})}{J_R} \end{bmatrix} \quad (3.3-6)$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 & \frac{K_t}{nJ_m} & 0 & 0 & 0 \end{bmatrix}^T \quad (3.3-7)$$

$$B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{K_t}{nJ_m} & 0 & 0 \end{bmatrix}^T \quad (3.3-8)$$

De forma análoga ao caso de dois graus de liberdade, a introdução de um grau de liberdade relativo reduz a ordem do modelo de estados ao desconsiderar o modo de corpo rígido com frequência nula.

Encontra-se abaixo a compilação de todos os valores de parâmetros do modelo obtidos durante as identificações, segundo sua ordem de obtenção, pois foi utilizado um procedimento sequencial para se chegar aos valores apresentados:

|                                                        |                                                            |
|--------------------------------------------------------|------------------------------------------------------------|
| $J_R = 3,2 \times 10^{-3} \text{ kg.m}^2$              | - momento de inércia do rotor.                             |
| $c_m = 1,954 \times 10^{-4} \text{ Nms}$               | - amortecimento do motor.                                  |
| $K_e = 5,031 \times 10^{-2} \text{ Vs/rad}$            | - constante elétrica do motor.                             |
| $K_t = 5,031 \times 10^{-2} \text{ Nm/A}$              | - constante de torque do motor.                            |
| $a_m = 3,61 \text{ s}^{-1}$                            | - constante de amortecimento                               |
| $J_m = 5,41 \times 10^{-5}$                            | - momento de inércia do motor.                             |
| $a_R = 1,044 \times 10^{-1} \text{ s}^{-1}$            | - constante de amortecimento do rotor.                     |
| $c_R = 3,34 \times 10^{-4} \text{ Nms}$                | - amortecimento do rotor.                                  |
| $a_{mR} = 1,825 \text{ s}^{-1}$                        | - constante de amortecimento do conjunto: motor e rotor.   |
| $c_{mR} = 6,46 \times 10^{-3} \text{ Nms}$             | - amortecimento torcional do conjunto motor e rotor.       |
| $c_{cR} = 2,45 \times 10^{-3} \text{ Nms}$             | - amortecimento torcional equivalente da correia no rotor. |
| $c_{cm} = 4,47 \times 10^{-4} \text{ Nms}$             | - amortecimento torcional equivalente da correia no motor. |
| $\omega_{dim} = 51,5 \text{ Hz} = 323,6 \text{ rad/s}$ | - frequência amortecida: motor e rotor.                    |

|                                                     |                                                                 |
|-----------------------------------------------------|-----------------------------------------------------------------|
| $\sigma_{tm}=313.11 \text{ rad/s}$                  | - parte real do pólo: motor e rotor.                            |
| $\zeta_{tm}= 0.436$                                 | - fator de amortecimento do pólo: motor e rotor.                |
| $\omega_{nm}=57,15 \text{ Hz}=359,07 \text{ rad/s}$ | - frequência natural do pólo: motor e rotor.                    |
| $k_c=3950 \text{ N/m}$                              | - constante elástica linear da correia.                         |
| $K_{cR}=43 \text{ Nm/rad}$                          | - constante elástica torcional equivalente da correia no rotor. |
| $K_{cm}=6.88 \text{ Nm/rad}$                        | - constante elástica torcional equivalente da correia no motor. |
| $\omega_d=101,5 \text{ Hz}=637.34 \text{ rad/s}$    | - frequência amortecida: rotores e eixo.                        |
| $\zeta=1.097 \times 10^{-2}$                        | - fator de amortecimento: rotores e eixo.                       |
| $\omega_n=101,51 \text{ Hz}=637.8 \text{ rad/s}$    | - frequência natural: rotores e eixo.                           |
| $a_e=6,993 \text{ s}^{-1}$                          | - constante de amortecimento do eixo.                           |
| $c_e=2.22 \times 10^{-2} \text{ Nms}$               | - amortecimento torcional do eixo.                              |
| $K_e=650 \text{ Nm/rad}$                            | - constante elástica torcional do eixo.                         |

Com os parâmetros pode-se construir as matrizes de estado numéricas para o caso específico do sistema modelado (sistema contínuo):

$$A = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1.2707E+5 & 0 & 0 & -1.2371E+1 & 0 & 8.7614E+0 & 0 \\ 0 & 1.2707E+5 & 0 & 0 & -1.2371E+01 & 0 & 8.7614E+0 \\ -1.3434E+4 & 0 & -2.0310E+5 & 9.2624E-1 & 0 & -7.9705E+0 & 6.9398E+0 \\ 0 & -1.3434E+4 & 2.0310E+5 & 0 & 9.2624E-1 & 6.9398E+0 & -7.9705E+0 \end{bmatrix}$$

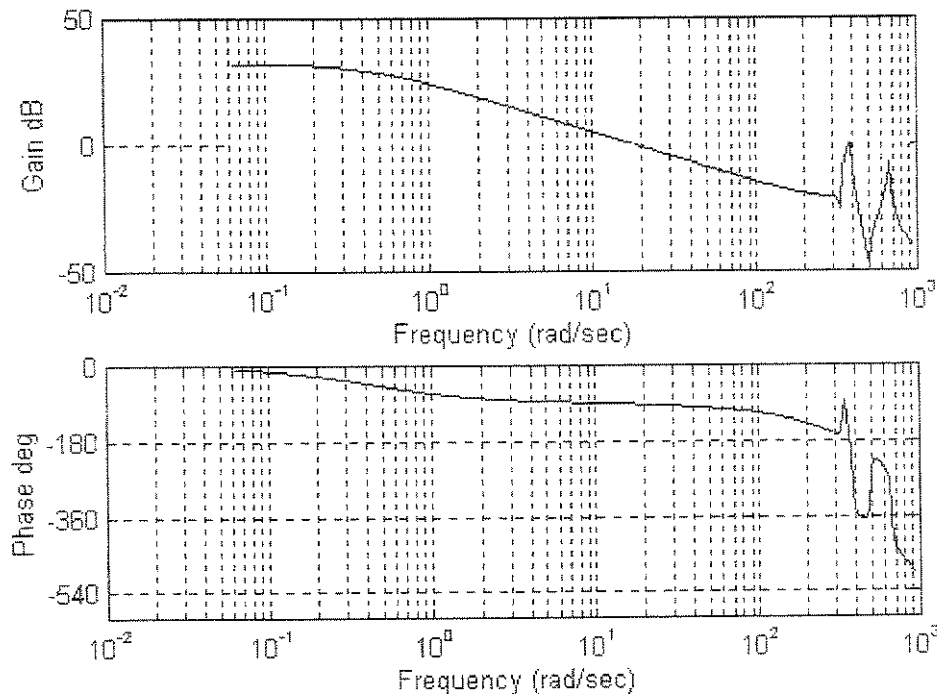
$$B_1 = \begin{bmatrix} 0 & 0 & 0 & 3,7179E+2 & 0 & 0 & 0 \end{bmatrix}^T$$

$$B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 3,7179E+2 & 0 & 0 \end{bmatrix}^T$$

O centro de interesse nesse trabalho foi observar a resposta em velocidade no rotor 1, tendo como entrada a corrente em torno da corrente de regime no motor de acionamento. O sistema implementado tem a característica de calcular a velocidade de rotação do rotor a partir dos pulso enviados pelo *encoder* ao computador, sendo o valor da velocidade entregue pelo computador no ciclo seguinte ao ciclo da medida, ou seja, é introduzido um atraso unitário na saída de velocidade. Para o caso discretizado com período de amostragem  $T_a=2,9\text{ms}$ , é mostrada a seguir a função de transferência para entrada de corrente no motor de acionamento e saída de velocidade no rotor 1, supondo-se um segurador de ordem zero na entrada e o atraso de um ciclo na saída:

$$G_{ve}(z) = \frac{(1.897z^6 + 3.255z^5 - 2.411z^4 + 8.628z^3 - 2.63z^2 + 3.375z + 1.525) \times 10^{-2}}{z^8 - 2.266z^7 + 3.814z^6 - 4.373z^5 + 4.266z^4 - 3.599z^3 + 2.049z^2 - 0.8876z}$$

A seguir encontram-se os gráficos de Bode para essa função de transferência, mostrando toda a FRF do sistema para a faixa de frequências alcançável com a taxa de aquisição utilizada.



**Figura 3.3-2** – FRF modelada para  $G_{ve}=V_e/I$ .

Nos gráficos acima, mostrando a resposta em frequência modelada para o sistema de 8ª ordem, pode-se ver um platô desde  $10^{-2}$  rad/s até 1 rad/s, seguido de um decaimento até 300 rad/s, onde há um pequeno vale e dois picos delimitando outro vale, sendo que o primeiro pico, em 380 rad/s, tem amplitude maior que o outro pico, localizado em 680 rad/s.

Embora o centro de interesse fosse a velocidade do rotor 1, também estava disponível o sinal de deformação do eixo que conecta os dois rotores. Para a mesma situação de discretização que a saída pelo rotor, obteve-se a seguinte função de transferência para a entrada de corrente em torno do regime e saída em termos de deformação no eixo:

$$G_{sg}(z) = \frac{(1.402z^6 + 8.255z^5 - 7.769z^4 + 0.5905z^3 + 6.418z^2 - 7.784z - 1.08) \times 10^{-5}}{z^7 - 2.266z^6 + 3.814z^5 - 4.373z^4 + 4.266z^3 - 3.599z^2 + 2.049z - 0.8876}$$

Dada a faixa de frequências que se encaixa no intervalo de discretização utilizado, segue abaixo a FRF para a saída em deformação no eixo:

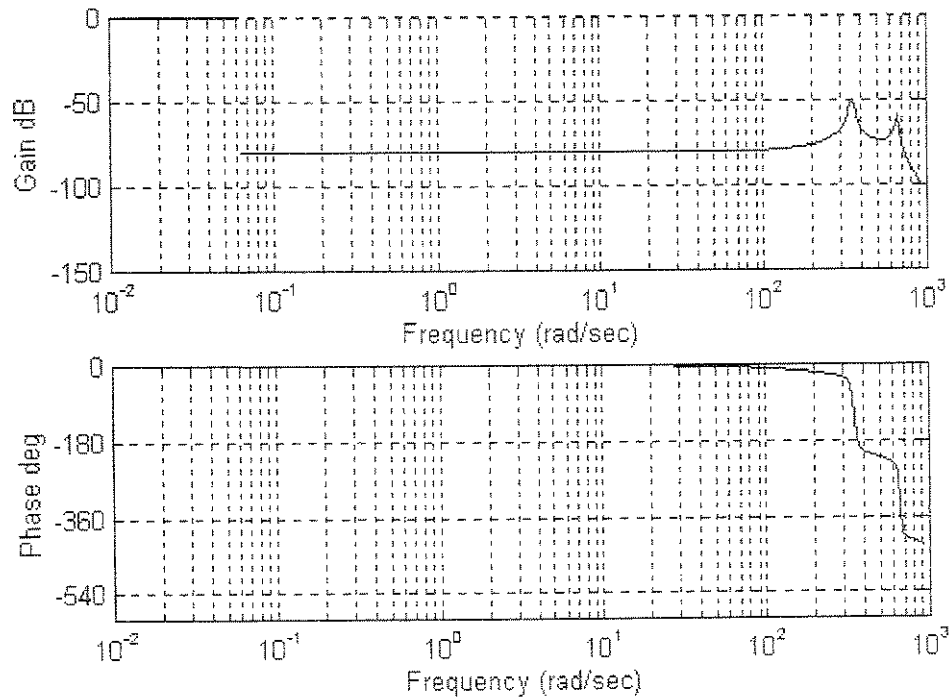


Figura 3.3-3 – FRF modelada para  $G_{sg}/I$ .

Esses gráficos mostram que há um platô desde  $10^{-2}$  rad/s até 200 rad/s. Distintamente, pode-se reconhecer dois picos: um em 340 rad/s com amplitude  $-50$  dB e um em 680 rad/s, com amplitude em torno de  $-60$  dB.

Os modelos para as duas saídas indicam que não há resposta do sistema para frequências superiores a 120 Hz, e que a resposta em frequência no rotor 1 é muito mais acentuada para frequências até 0,15 Hz ( $\sim 1$  rad/s) do que para frequências maiores que essa. Por outro lado, a resposta em frequência da deformação do eixo foi prevista como sendo mais acentuada para frequências entre 50 Hz e 120 Hz (300 rad/s e 500 rad/s).

Uma vez estimado o modelo, e estando de posse das frequências naturais esperadas para o sistema de 8ª ordem, realizou-se medidas nesse sistema modelado com 4 graus de liberdade fazendo-se excitação em corrente. Aplicou-se um sinal aleatório com amplitude 2 A a partir de uma corrente de regime de 4 A, sendo que as frequências foram distribuídas uniformemente em toda a faixa de 0 a 170 Hz. Foram tomadas medidas das saídas de velocidade no rotor 1 e de deformação no eixo.

Para a validação do modelo construído, foi feita a estimação via método recursivo, adotando um modelo com mesmo número de parâmetros que o modelo construído. Foi utilizado o algoritmo da Seção 2.3, que tem uma descrição detalhada a seguir:

1. Escolher  $\alpha$ ,  $\gamma$  e  $N$ ;
2. Se  $\alpha = 1$  e  $\gamma = 1$ , obtém-se mínimos quadrados não ponderado;

Se  $a = 1 - \gamma$  e  $0 < \gamma < 1$ , o resultado é o método dos mínimos quadrados ponderado exponencialmente;

Escolher valores iniciais para  $P(N)$  e  $\hat{\theta}(N)$

4. Formar  $\varphi^T(N+1)$  a partir de  $\{y(0), y(1), \dots, y(N), y(0), u(1), \dots, u(N)\}$

5. Fazer  $k = N$ ;

$$6. L(k+1) = \frac{P(k+1)}{\gamma} \varphi(k+1) \left( \frac{1}{a} + \frac{\varphi(k+1)^T P(k) \varphi(k+1)}{\gamma} \right)^{-1}$$

7. Lê  $y(k+1)$  e  $u(k+1)$ ;

$$8. \hat{\theta}(k+1) = \hat{\theta}(k) + L(k+1) \{y(k+1) - \varphi^T(k+1) \hat{\theta}(k)\}$$

$$9. P(k+1) = \frac{1}{\gamma} [I - L(k+1) \varphi^T(k+1)] P(k)$$

10. Formar  $\varphi(k+2)$

11. Faz  $k = k+1$

12. Volta para o passo 6

... A seguir apresentam-se os resultados gráficos da estimação comparados com os do modelo, e neles se encontram pólos e zeros da planta modelada e da planta estimada:

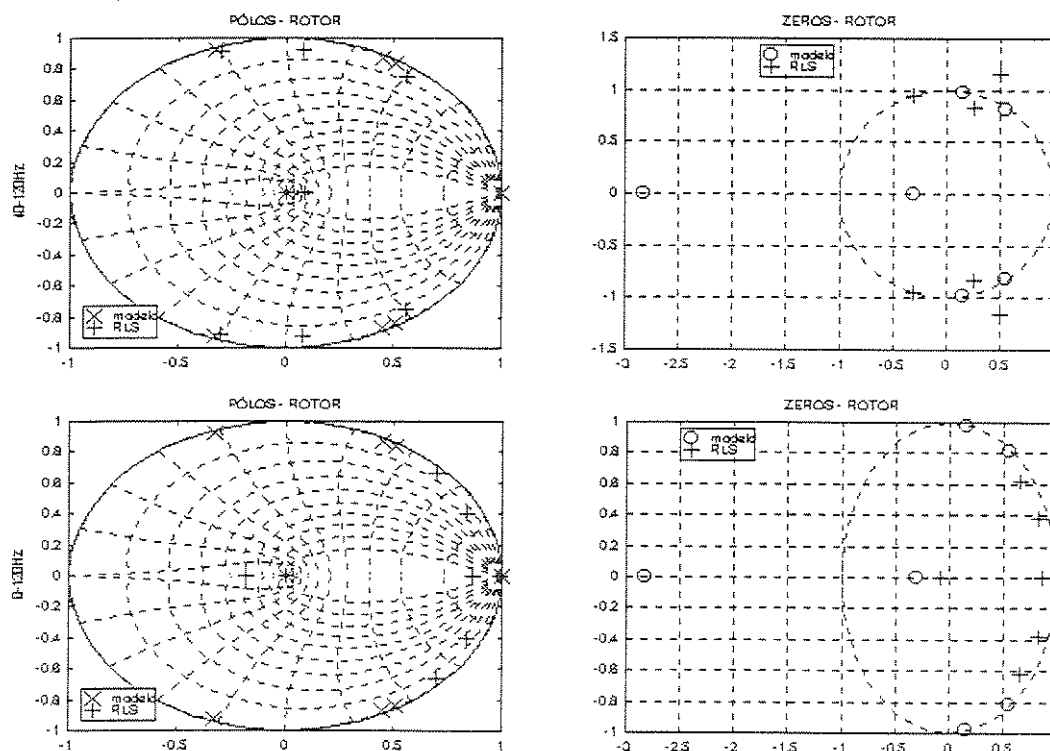


Figura 3.3-4 – Localização de pólos e zeros modelados e estimados.

Nos gráficos para os sinais filtrados entre 40Hz e 120Hz, pode-se notar que há 1 pólo real e 2 pólos complexos estimados a partir do modelo que se aproximam dos pólos modelados, enquanto há um pólo complexo estimado que se encontra a meio caminho entre os pólos modelados. Os zeros modelados encontram-se todos, a menos de um, dentro do círculo unitário, enquanto dos zeros estimados, existem dois fora do círculo unitário.

Partindo do modelo estimado entre 40Hz e 120Hz, utilizou-se os sinais filtrados entre 0Hz e 120Hz e obteve-se uma planta estimada que tem um pólo coincidente com o pólo modelado, e todos os zeros dentro do círculo unitário. Abaixo se encontra a planta estimada pelo algoritmo recursivo para essa última região:

$$\hat{G}_{ve}(z) = \frac{(2.599z^6 - 9.708z^5 + 15.65z^4 - 13.4z^3 + 5.91z^2 - 0.8898z - 0.13) \times 10^{-2}}{z^8 - 4.732z^7 + 9.731z^6 - 11z^5 + 6.98z^4 - 2.077z^3 - 0.02462z^2 + 0.1246z}$$

A figura abaixo apresenta o diagrama de Bode para o sistema estimado:

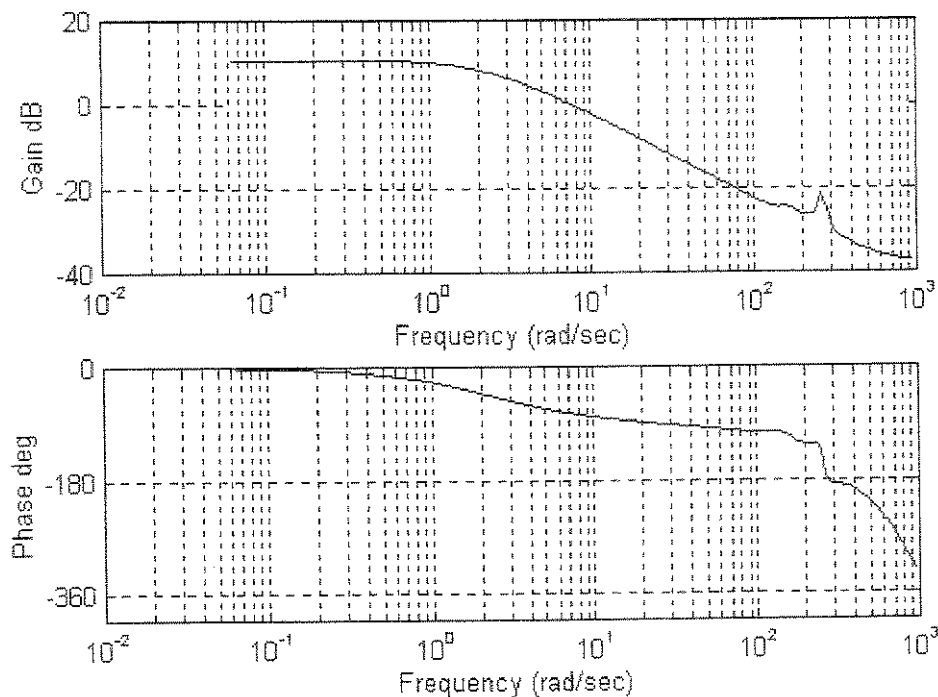
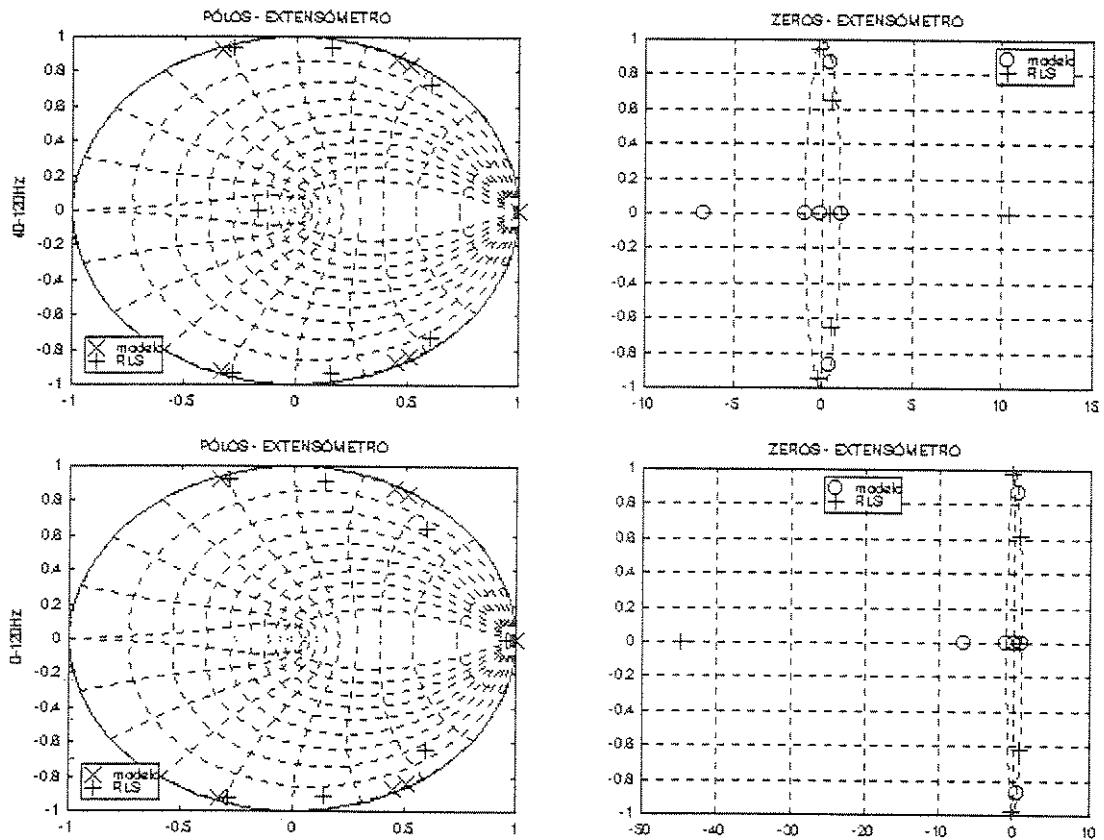


Figura 3.3-5 – FRF estimada para  $\hat{G}_{ve} = \frac{V_e}{I}$ .

O sistema estimado recursivamente para a região entre 0Hz e 120Hz tem uma resposta em frequência que apresenta um platô entre 0,07rad/s e 2rad/s, cai até 100rad/s, tem um pico pouco pronunciado em 150rad/s e um pico pronunciado em 200rad/s.

Assim como para o caso da saída em velocidade, a saída em deformação teve seu modelo validado por uma estimação recursiva para se obter um modelo estimado com mesmo

número de parâmetros que o modelo construído. Abaixo estão os pólos e zeros das plantas modelada e estimada:



**Figura 3.3-6** – Localização de pólos e zeros de  $\hat{G}_{sg} = \frac{SG}{I}$ .

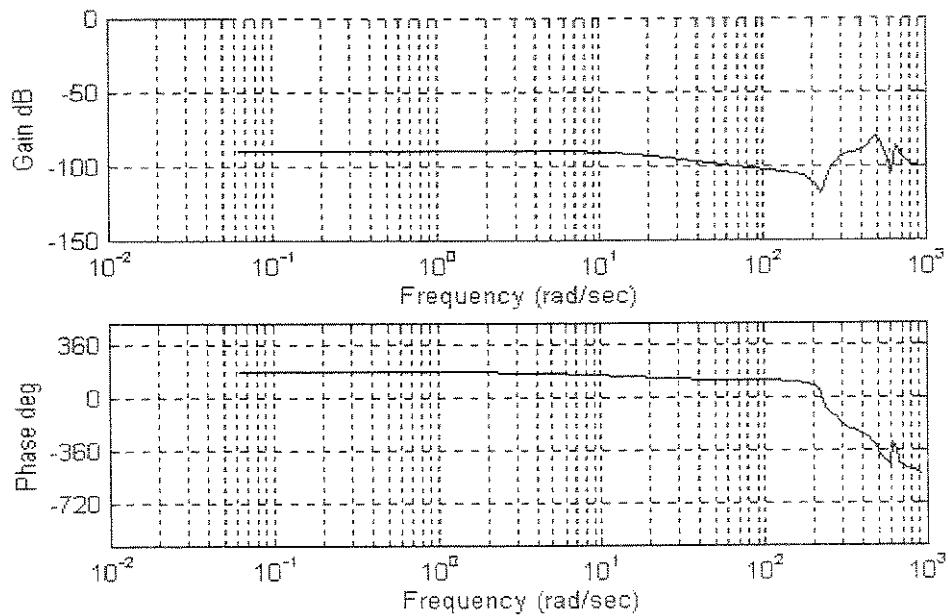
Para os sinais filtrados entre 40Hz e 120Hz, o modelo estimado teve dois pólos complexos que se aproximaram dos pólos modelados. Os zeros, tanto do modelo construído quanto do modelo estimado, estão todos (exceto um) dentro do círculo unitário.

Na região entre 0Hz e 120Hz a situação não ficou diferente da região anterior, tendo os pólos se deslocado um pouco no modelo estimado, e também o zero externo ao círculo.

Abaixo se encontra a função de transferência estimada para a região entre 0Hz e 120Hz, lembrando que a taxa de amostragem é  $T_a=2,9\text{ms}$ :

$$\hat{G}_{sg}(z) = \frac{(-0.03287z^6 - 1.414z^5 + 2.826z^4 - 3.303z^3 + 3.216z^2 - 2.832z + 1.159) \times 10^{-5}}{z^7 - 1.844z^6 + 2.901z^5 - 3.448z^4 + 3.207z^3 - 2.484z^2 + 1.358z - 0.5858}$$

O gráfico de Bode para esse modelo é apresentado a seguir:



**Figura 3.3-7** – FRF estimada para  $\hat{G}_{sg} = \frac{SG}{I}$ .

No gráfico da amplitude da função de resposta em frequência pode-se notar o platô entre 0,07rad/s e 20 rad/s, uma queda até 200rad/s, seguida de um vale e logo após se tem um pico mais intenso em 500rad/s e outro pico, menos intenso, em 650rad/s.

O modelo obtido a partir dos parâmetros do sistema medido por partes possui três frequências características, sendo duas em torno de 50Hz e uma de 100Hz. As duas frequências mais baixas advêm dos acoplamentos entre motores e rotores, enquanto a frequência de 100Hz deve-se ao acoplamento entre rotores.

No entanto, ao se analisar os dados do sistema completo, percebe-se na saída pelo rotor apenas uma frequência em torno de 80Hz, e na saída pelo extensômetro percebe-se também as frequências 80Hz e 100Hz. Ou seja, o efeito do acoplamento entre rotores só é observável pela saída de deformação, enquanto o efeito de acoplamento entre motores e rotores aparece como uma única frequência em torno de 80Hz.

## Capítulo 4

### Implementação dos Métodos de Controle

Nesta seção será apresentado o corpo experimental das técnicas de controle mencionadas no Capítulo 2. Inicialmente será apresentada a configuração geral da bancada experimental, com todos os seus componentes, seguida de uma apresentação da instrumentação eletrônica utilizada para a obtenção dos resultados experimentais. A primeira implementação de controlador apresentada é a do controlador tacométrico. Em seguida apresenta-se os resultados para o controlador polinomial e para o controlador polinomial auto-sintonizável. Todas as implementações foram feitas para dois sistemas: um sistema experimental reduzido de primeira ordem e o sistema completo, de sétima ordem, para o qual há resultados experimentais para controle tacométrico e simulados para controle polinomial e polinomial auto-sintonizável.

#### 4.1. Instrumentação e Conexão com o Computador

A bancada experimental tem vários componentes que estão distribuídos em um sistema integrado pelo computador, o qual organiza as atividades de coleta e envio de informações durante os experimentos, fazendo a leitura de dados e enviando sinais de controle. Pode-se ver abaixo como foram ligados os componentes do sistema:

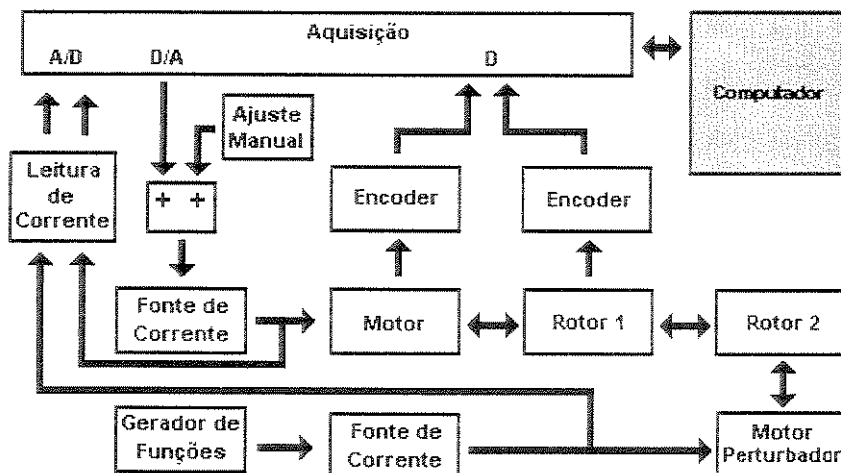


Figura 4.1-1 – Fluxo de sinais na bancada experimental.

O computador centraliza o comando do sistema, recebendo e enviando dados para a placa de aquisição, que possui entradas e saídas analógicas e digitais. O circuito de leitura de corrente e a fonte controlada de corrente recebem e enviam sinais para a placa de aquisição, e os encoders enviam pulsos de rotação que são processados pelo computador para se obter as medidas de velocidade. A cada ciclo de amostragem, os sinais dos sensores são adquiridos e processados pelo algoritmo global, que estima os parâmetros da planta, recalcula os coeficientes do controlador e a respectiva lei de controle, a qual é aplicada ao motor de acionamento através de uma saída analógica em tensão da placa de aquisição e do circuito da fonte controlada de corrente. A perturbação é provocada por um circuito equivalente, porém com sua entrada aplicada através de um gerador de sinais.

As fontes de corrente utilizadas para alimentar o sistema foram projetadas para fornecer a corrente elétrica necessária aos motores, segundo uma tensão de referência selecionável (amplificador de transcondutância). No caso do motor de acionamento, a tensão de referência foi gerada pela saída da placa A/D. Foram montados os circuitos mostrados no Apêndice, tanto para o motor de acionamento, quanto para o motor de perturbação. Abaixo se observa o diagrama de Bode experimental que caracteriza os circuitos:

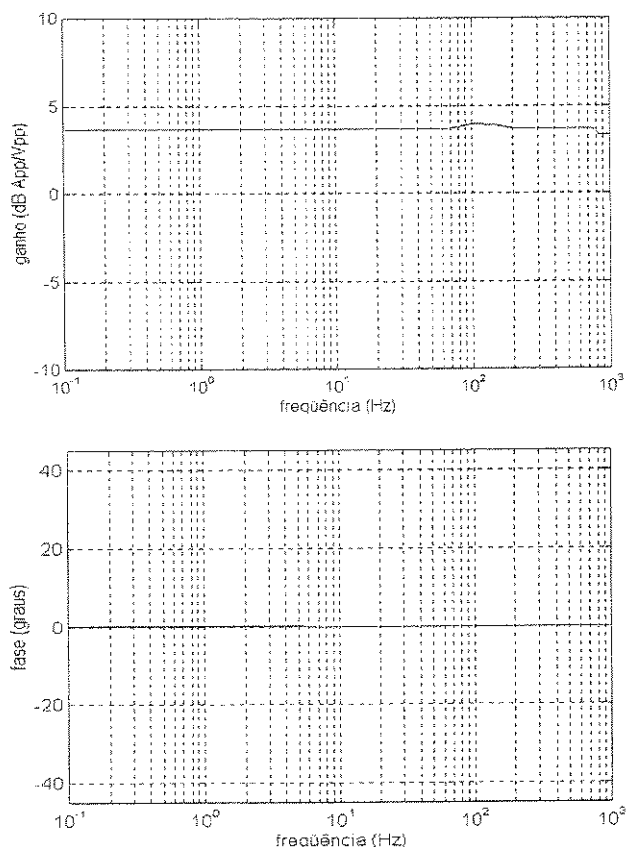


Figura 4.1-2 – Diagramas de Bode para as fontes de corrente.

Pode-se observar pelos diagramas que o módulo se mantém constante e igual a 1 em toda a faixa de  $10^{-1}\text{Hz}$  a  $10^3\text{Hz}$  e a fase apresenta um ligeiro decréscimo ao final da faixa. Para fins de controle com taxa de amostragem  $T_a=2.9\text{ms}$ , ou  $f_a=330\text{Hz}$ , isso significa que os circuitos são adequados para fornecer a corrente desejada para o sistema controlado.

## 4.2. Controle Tacométrico

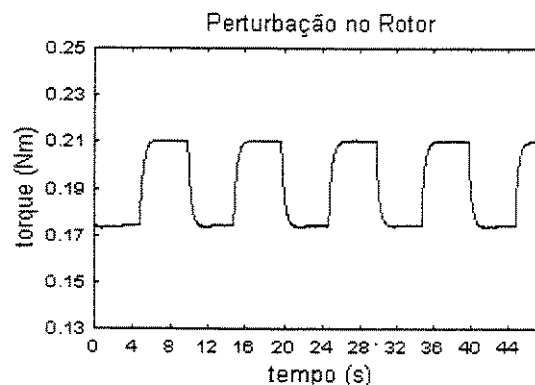
Nesta seção são apresentados os resultados da aplicação de um controlador tacométrico aos dois sistemas experimentais utilizados. Inicialmente é apresentado o resultado para o sistema reduzido, e depois para o sistema completo.

### 4.2.1. Sistema Reduzido

Todos os controladores, o tacométrico, o polinomial e o polinomial adaptativo, foram testados experimentalmente no sistema reduzido apresentado na Seção 3.1.3, tendo como objetivo a regulação da velocidade de rotação do rotor em um patamar de  $70\text{ rad/s}$ . Esse sistema foi controlado utilizando-se uma taxa de amostragem  $T_a=49.3\text{ms}$  ( $f_a=20\text{Hz}$ ), partindo-se do modelo discretizado mostrado a seguir:

$$G(z) = \frac{B(z)}{A(z)} = \frac{b}{z^2 + az} = \frac{0.4618}{z^2 - 0.9919z}$$

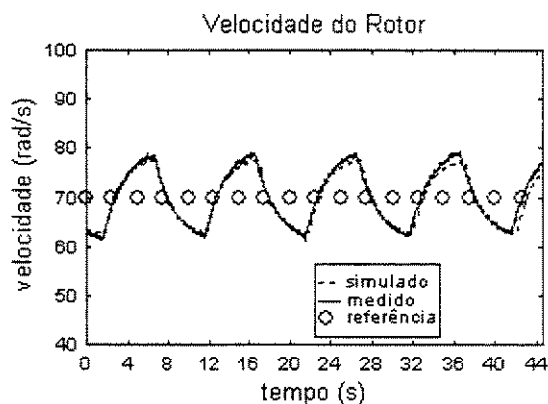
Através do motor de perturbação foi aplicado um torque de perturbação ao rotor:



**Figura 4.2-1** – Exemplo de torque perturbador aplicado ao rotor.

O torque perturbador teve sua amplitude determinada em  $0.05\text{Nm}$  e frequência  $0.1\text{Hz}$ , ou seja, na região em que as correias podem ser consideradas rígidas.

O controlador tacométrico apresentado na Seção 2.1 foi implementado com ganho  $K=0,0125$  e a resposta pode ser vista abaixo:

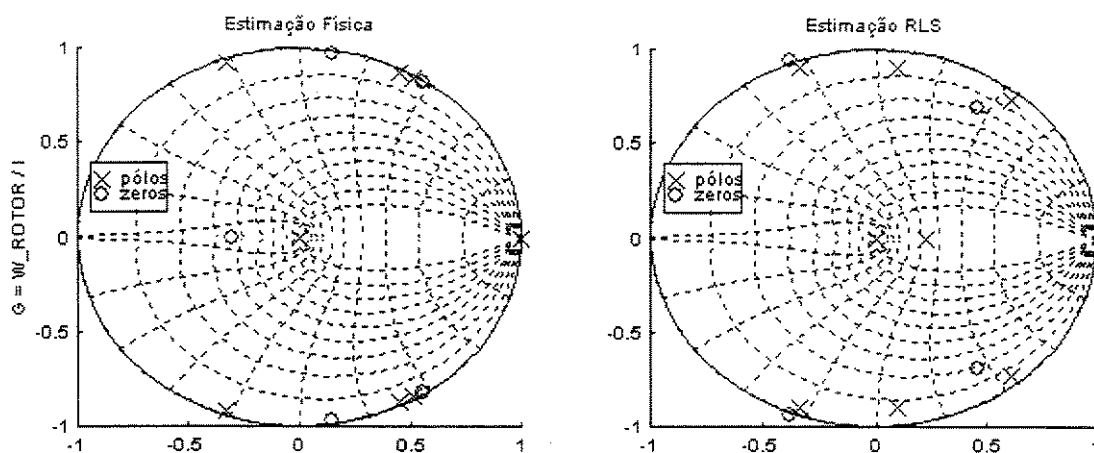


**Figura 4.2-2** – Controlador tacométrico sujeito a perturbação.

A Resposta da Figura 4.2-2 mostra que dada a velocidade de rotação desejada de 70 rad/s, o controlador tacométrico manteve a resposta do sistema entre 60 rad/s e 80 rad/s quando submetido à perturbação na carga.

#### 4.2.2. Sistema Completo

O primeiro passo antes de fazer o controle tacométrico do sistema é, partindo de um modelo, escolher o ganho de realimentação tacométrica de acordo com o posicionamento das raízes. Foi escolhido o modelo estimado como modelo inicial para o sistema:

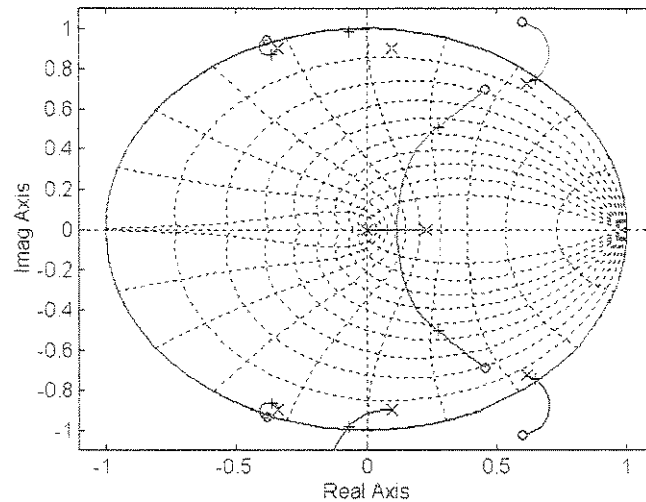


**Figura 4.2-3** – Pólos e zeros do modelo paramétrico e do modelo estimado.

O modelo obtido através dos parâmetros físicos tem um de seus pólos complexos atenuado por um zero quase coincidente, sendo que dois dos pólos complexos têm valores

parecidos. O modelo estimado tem três pólos complexos, sendo que dois são atenuados por zeros próximos a eles.

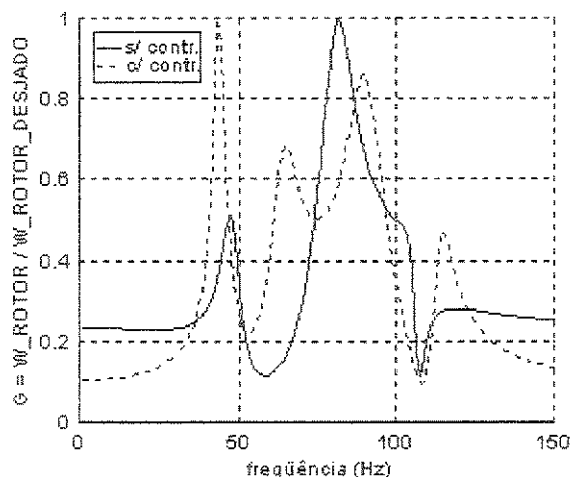
Procedeu-se então a elaboração do gráfico de lugar das raízes, com a finalidade de projetar o controlador. O gráfico de lugar das raízes (“root locus”) pode ser visto a seguir (ver Seção 7.2):



**Figura 4.2-4** - Root locus para controle tacométrico.

Dadas as raízes possíveis de se obter com realimentação tacométrica, foi escolhido  $K=10$ , pois implicaria em deslocar um pólo de maneira perceptível experimentalmente e ainda manter todas as raízes dentro do círculo unitário.

Feito o teste com controle tacométrico utilizando ganho de realimentação  $K=10$ , processou-se os dados, fazendo-se estimação recursiva da planta controlada, e obtendo-se o seguinte resultado:



**Figura 4.2-5** – Respostas em frequência: sistema controlado e não controlado.

Pode-se notar que o sistema não controlado possui um pico de grande amplitude em 80Hz e um pico de menor amplitude em 45Hz. Já o sistema controlado possui um pico muito pronunciado em 40 Hz, um menor em 90Hz, um menor em 65Hz, e um menor ainda em 120Hz. Ambos os sistemas exibem vales em 60Hz e 120Hz.

### 4.3. Controle Polinomial

Continuando na mesma linha da seção anterior, esta seção contém o projeto e simulação do controlador polinomial tanto para planta reduzida como para a planta completa.

#### 4.3.1. Sistema Reduzido

O controlador polinomial e o polinomial adaptativo tiveram como modelo de referência uma planta com frequência natural  $2\pi$  rad/s e fator de amortecimento 0.7, cuja função de transferência está descrita abaixo ( $T_a=49,3\text{ms}$ ):

$$G_m(z) = \frac{Bm(z)}{Am(z)} = \frac{0.087918}{z^2 - 1.5709z + 0.64813}$$

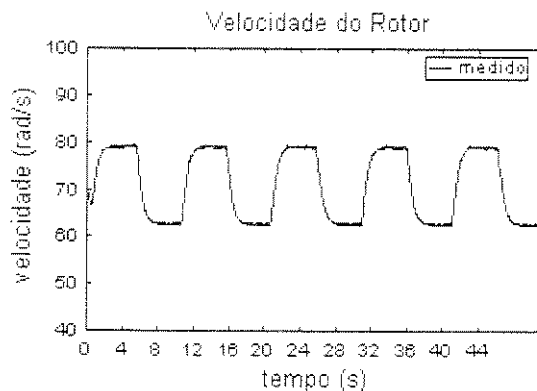
Os polinômios do controlador são:

$$T(z) = 1.953z^{-8}$$

$$S(z) = (1.891z^7 - 6.79z^6 + 10.33z^5 - 8.131z^4 + 2.904z^3 - 0.0386z^2 - 0.17z) \times 10^4$$

$$R(z) = z^8 + 1.948z^7 + 2.089z^6 - 490.3z^5 + 1274z^4 - 1425z^3 + 760.7z^2 - 129z - 17.74$$

A resposta do sistema pode ser vista abaixo:



**Figura 4.3-1** – Resposta a controlador com modelo de referência.

Assim como o controlador tacométrico, o controlador polinomial sujeito à perturbação na demanda de torque manteve a rotação do rotor entre 60rad/s e 80 rad/s, verificando-se uma oscilação de 14% em torno do regime desejado de 70 rad/s, ou seja, a perturbação não foi compensada pelo controlador.

#### 4.3.2. Sistema Completo

O modelo utilizado foi o modelo estimado recursivamente. Como modelo de referência foi escolhida uma planta com frequência natural 60rad/s, fator de amortecimento  $0.01s^{-1}$  e com ganho estático unitário. O observador escolhido foi o polinômio  $A_0=z^8$ , porque é o que tem dinâmica mais rápida possível ("dead beat"). Os polinômios utilizados para projeto são ( $T_d=2.9ms$ ):

$$\begin{aligned} B &= b_1 z^6 + b_2 z^5 + b_3 z^4 + b_4 z^3 + b_5 z^2 + b_6 z^1 + b_7 \\ &= (2.60z^6 - 9.71z^5 + 15.7z^4 - 1.34z^3 + 5.91z^2 - 0.89z^1 - 0.13) \times 10^{-2} \\ A &= a_1 z^8 + a_2 z^7 + a_3 z^6 + a_4 z^5 + a_5 z^4 + a_6 z^3 + a_7 z^2 + a_8 z + a_9 \\ &= z^8 - 4.73z^7 + 9.731z^6 - 11.0z^5 + 6.98z^4 - 2.08z^3 - 0.0246z - 0.125z \end{aligned}$$

$$B_m = (62.5z^6 - 2.33z^5 + 3.76z^4 - 3.22z^3 + 1.42z^2 - 0.214z^1 - 0.0313) \times 10^{-1}$$

$$A_m = z^8 - 2.938z^7 + 2.907z^6 - 0.968z^5$$

$$A_o = z^8$$

A matriz de Sylvester para o sistema é:

$$E = \begin{bmatrix} a_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_2 & a_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_3 & a_2 & a_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_4 & a_3 & a_2 & a_1 & 0 & 0 & 0 & 0 & 0 & b_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_5 & a_4 & a_3 & a_2 & a_1 & 0 & 0 & 0 & 0 & b_2 & b_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & 0 & 0 & 0 & b_3 & b_2 & b_1 & 0 & 0 & 0 & 0 & 0 \\ a_7 & a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & 0 & 0 & b_4 & b_3 & b_2 & b_1 & 0 & 0 & 0 & 0 \\ a_8 & a_7 & a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & 0 & b_5 & b_4 & b_3 & b_2 & b_1 & 0 & 0 & 0 \\ a_9 & a_8 & a_7 & a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & 0 & 0 \\ 0 & a_9 & a_8 & a_7 & a_6 & a_5 & a_4 & a_3 & a_2 & b_7 & b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & 0 \\ 0 & 0 & a_9 & a_8 & a_7 & a_6 & a_5 & a_4 & a_3 & 0 & b_7 & b_6 & b_5 & b_4 & b_3 & b_2 & b_1 \\ 0 & 0 & 0 & a_9 & a_8 & a_7 & a_6 & a_5 & a_4 & 0 & 0 & b_7 & b_6 & b_5 & b_4 & b_3 & b_2 \\ 0 & 0 & 0 & 0 & a_9 & a_8 & a_7 & a_6 & a_5 & 0 & 0 & 0 & b_7 & b_6 & b_5 & b_4 & b_3 \\ 0 & 0 & 0 & 0 & 0 & a_9 & a_8 & a_7 & a_6 & 0 & 0 & 0 & 0 & b_7 & b_6 & b_5 & b_4 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_9 & a_8 & a_7 & 0 & 0 & 0 & 0 & 0 & b_7 & b_6 & b_5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_9 & a_8 & 0 & 0 & 0 & 0 & 0 & 0 & b_7 & b_6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_7 \end{bmatrix}$$

Calculando-se os polinômios R, S e T como descrito no capítulo 2 obtêm-se os resultados para a simulação de controle polinomial para uma entrada tipo degrau unitário, como mostrado a seguir (programa rst1030.m, no apêndice):

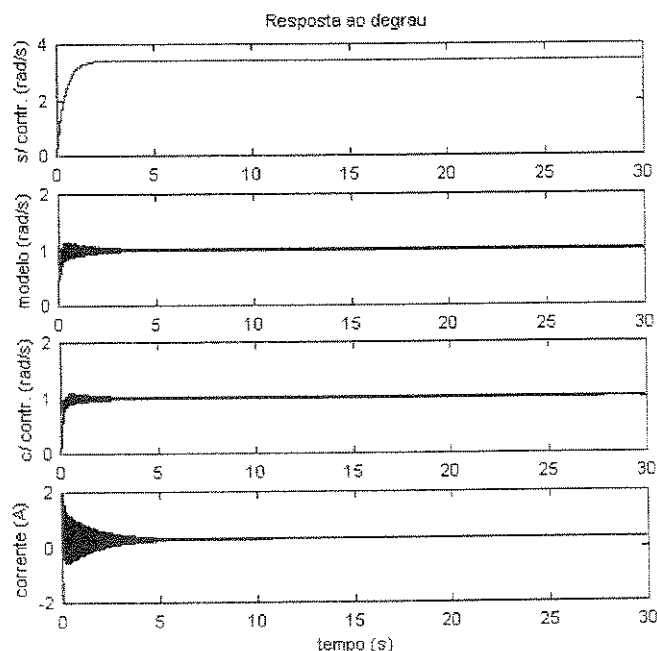


Figura 4.3-2 – Resposta ao degrau unitário.

A resposta ao degrau do sistema não controlado mostra que ele leva 2,5s para chegar ao regime de 3,5rad/s. O sistema controlado respondeu ao degrau seguindo o modelo, e a corrente necessária para fazer o acionamento esteve em todo o intervalo de tempo dentro da faixa compreendida por  $-2A$  e  $+2A$ .

#### 4.4. Controle Auto-sintonizável

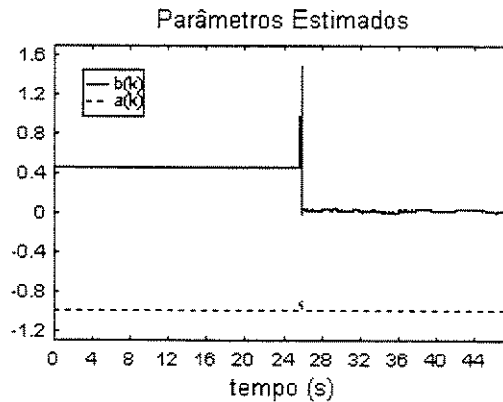
São apresentados nesta seção os resultados da aplicação de controle auto-sintonizável para o sistema reduzido (experimental) e para o sistema completo (simulação).

##### 4.4.1. Sistema Reduzido

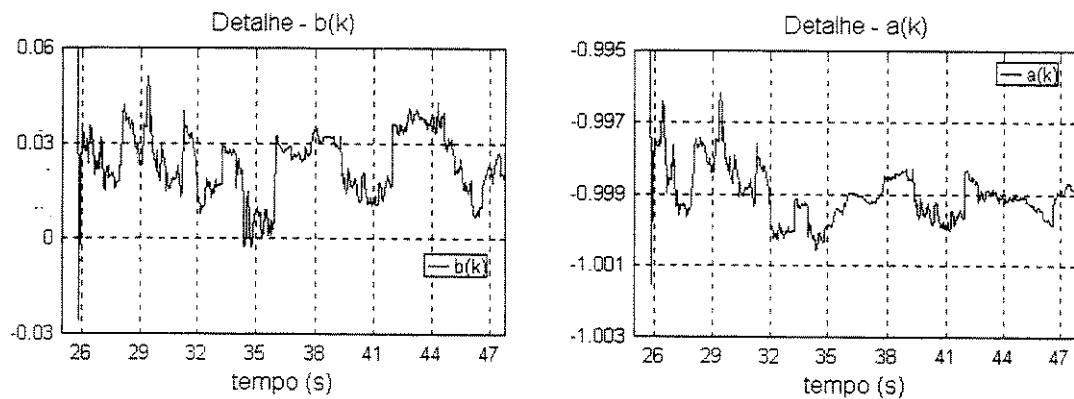
O controlador polinomial e o polinomial adaptativo tiveram como modelo de referência uma planta com frequência natural  $2\pi$  rad/s e fator de amortecimento 0,7, cuja função de transferência está descrita abaixo:

$$G_m(z) = \frac{Bm(z)}{Am(z)} = \frac{0,087918}{z^2 - 1,5709z + 0,64813}$$

Abaixo se encontram gráficos da evolução da estimação dos parâmetros:



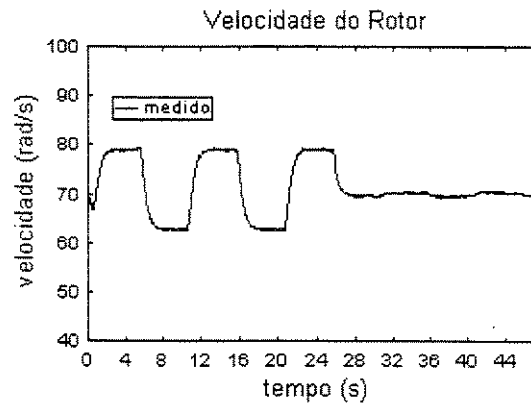
**Figura 4.4-1** – Evolução temporal dos parâmetros estimados.



**Figura 4.4-2** – Detalhes da evolução temporal dos parâmetros estimados.

As Figuras 4.4-1 e 4.4-2 apresentam as modificações nos parâmetros da planta à medida que foi aplicada a estratégia adaptativa: até o instante 25s, os parâmetros foram mantidos fixos para efeito de controle, mas quando a estimação entra em funcionamento, percebe-se um transiente em ambos os parâmetros, sendo que  $a(k)$  estabiliza-se em torno de  $-0,999$ , enquanto  $b(k)$  oscila entre 0,00 e 0,03, acompanhando as variações do torque de perturbação.

A resposta do sistema durante todo o experimento pode ser vista a seguir:



**Figura 4.4-3** – Resposta a controlador com modelo de referência.

Até o instante 25s o controlador utilizado era controlador polinomial, e assim como o controlador tacométrico, o controlador polinomial sujeito à perturbação na demanda de torque manteve a rotação do rotor entre 60rad/s e 80 rad/s, verificando-se uma oscilação de 14% em torno do regime desejado. No instante 25s foi ligado o algoritmo adaptativo, e a velocidade de saída passou a oscilar 0,7% em torno do regime desejado.

#### 4.4.2. Sistema Completo

Dando continuidade à seção anterior, nesta seção são mostrados os resultados da implementação do controlador polinomial auto-sintonizável. O algoritmo, apresentado na Seção 2.4, é descrito detalhadamente abaixo, e nele estão contidas todas as informações utilizadas para montar o programa que realizou o controle no sistema experimental.

1. Definir o modelo de partida para a planta:  $G(z) = \frac{B}{A}$ ;
2. Fazer  $B^- = B$ ;
3. Fazer  $B^+ = 1$ ;
4. Definir o modelo de referência:  $G_m(z) = \frac{B_m}{A_m}$ ;
5. Sendo  $B_m = B^- B_m^+$ , e  $B_m^+$  tal que  $G_m(z=1) = 1$ ;
6. Definir a ordem do polinômio observador  $A_o$ ;
7. Escolher  $a$  e  $N$ , respectivamente o fator de esquecimento e o ponto inicial de estimação;
8. Construir  $\hat{\theta}$  a partir de  $B$  e  $A$ ;
9. Inicializar  $P$ ;
10. Formar  $\varphi^T(N+1)$  a partir de  $\{y(0), y(1), \dots, y(N), y(0), u(1), \dots, u(N)\}$ ;
11. Fazer  $k = N$ ;
12.  $L(k+1) = \frac{P(k+1)}{\gamma} \varphi(k+1) \left( \frac{1}{a} + \frac{\varphi(k+1)^T P(k) \varphi(k+1)}{\gamma} \right)^{-1}$ ;
13. Lê  $y(k+1)$  e  $u(k+1)$ ;
14.  $\hat{\theta}(k+1) = \hat{\theta}(k) + L(k+1) \{y(k+1) - \varphi^T(k+1) \hat{\theta}(k)\}$ ;
15.  $P(k+1) = \frac{1}{\gamma} [I - L(k+1) \varphi^T(k+1)] P(k)$ ;
16. Formar  $\varphi(k+2)$ ;
17. Desmembrar  $\hat{\theta}$  em  $B$  e  $A$ ;
18. Fazer  $B^- = B$ ;
19. Determinar  $B_m = B^- B_m^+$ , e  $B_m^+$  tal que  $G_m(z=1) = 1$ ;
20. Faz  $\hat{T} = B_m^+ A_o$ ;
21. Determinar  $\hat{R}$  e  $\hat{S}$  a partir da equação diofantina:  $\hat{A}\hat{R} + B^-\hat{S} = A_m A_o$

22. Calcular lei de controle  $\hat{R}u(k) = \hat{T}u_r(k) - \hat{S}y(k)$

23. Faz  $k = k + 1$

24. Volta para o passo 12

A partir de um modelo linear discretizado de 8<sup>a</sup> ordem, estimado para o sistema formado por 2 motores, 2 correias, 2 rotores e um eixo, e num regime de 30 rad/s no rotor 1, fez-se com que os parâmetros desse modelo variassem com o tempo. Para esse sistema variante no tempo foi simulada a ação de um controlador polinomial adaptativo baseado em estimação recursiva por mínimos quadrados.

As funções de transferência ( $T_d=2,9$ ms) do controlador inicial são:

$$\frac{T}{R}(z) = \frac{1,953z^8}{z^8 + 1,948z^7 + 2,089z^6 - 490,3z^5 + 1274z^4 - 1425z^3 + 760,7z^2 - 129z - 17,74}$$

$$\frac{S}{R}(z) = \frac{(1,891z^7 - 6,79z^6 + 10,33z^5 - 8,131z^4 + 2,904z^3 - 0,0386z^2 - 0,17z) \times 10^4}{z^8 + 1,948z^7 + 2,089z^6 - 490,3z^5 + 1274z^4 - 1425z^3 + 760,7z^2 - 129z - 17,74}$$

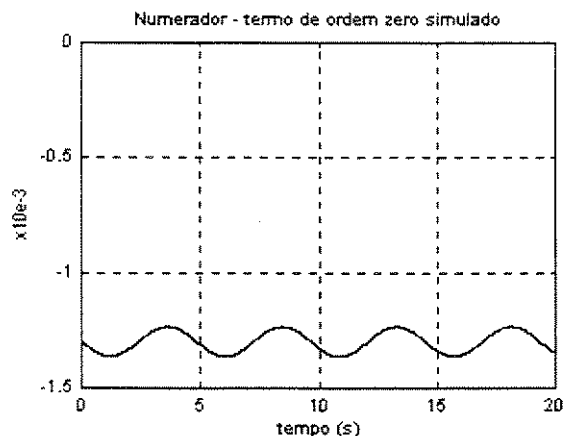
A planta identificada da qual se parte é:

$$\hat{G}_{ve}(z) = \frac{(2,599z^6 - 9,708z^5 + 15,65z^4 - 13,4z^3 + 5,91z^2 - 0,8898z - 0,13) \times 10^{-2}}{z^8 - 4,732z^7 + 9,731z^6 - 11z^5 + 6,98z^4 - 2,077z^3 - 0,02462z^2 + 0,1246z}$$

O modelo de referência é:

$$\frac{B_m}{A_m}(z) = \frac{B'_m B^-}{A_m}(z) = \frac{(0,614z^6 - 2,292z^5 + 3,695z^4 - 3,163z^3 + 1,395z^2 + 0,21z - 0,031) \times 10^{-1}}{z^8 - 2,784z^7 + 2,601z^6 - 0,8163z^5}$$

A simulação foi realizada fazendo-se apenas o termo de ordem zero do numerador da planta identificada variar 5% em torno de seu valor inicial, de forma senoidal:

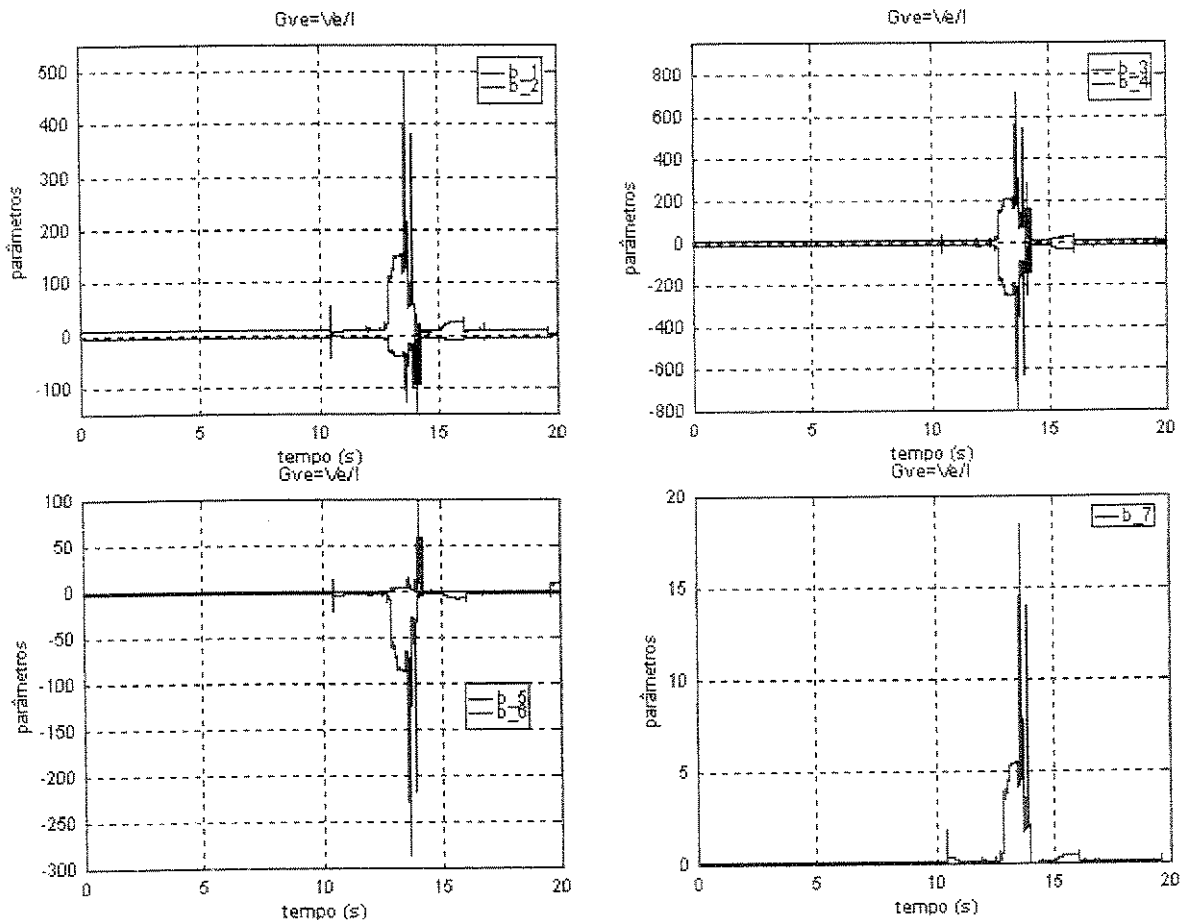


**Figura 4.4-4** – Variação simulada do termo de ordem zero do numerador.

A Figura 4.4-4 mostra a variação real do coeficiente  $b_7$ , que é o termo de ordem zero do numerador da planta simulada, cujos parâmetros são estimados conforme o modelo abaixo:

$$\hat{G} = \frac{b_1 z^6 + b_2 z^5 + b_3 z^4 + b_4 z^3 + b_5 z^2 + b_6 z^1 + b_7}{a_1 z^8 + a_2 z^7 + a_3 z^6 + a_4 z^5 + a_5 z^4 + a_6 z^3 + a_7 z + a_8 z + a_9}$$

sendo  $a_1=1$  e  $a_9=0$ . Esta planta foi estimada pelo algoritmo recursivo, que retornou os parâmetros mostrados nos gráficos a seguir, nos quais se percebe como a simulação de variação real de um parâmetro reflete-se na estimação de todos os outros parâmetros:



**Figura 4.4-5** – Evolução temporal dos parâmetros do numerador da planta.

No numerador, todos os parâmetros tiveram suas estimações com um transiente de alta amplitude entre os instantes 10s e 15s. A estimação do coeficiente  $b_7$ , que foi o coeficiente variante no tempo, também apresentou um transiente nesse intervalo de tempo.

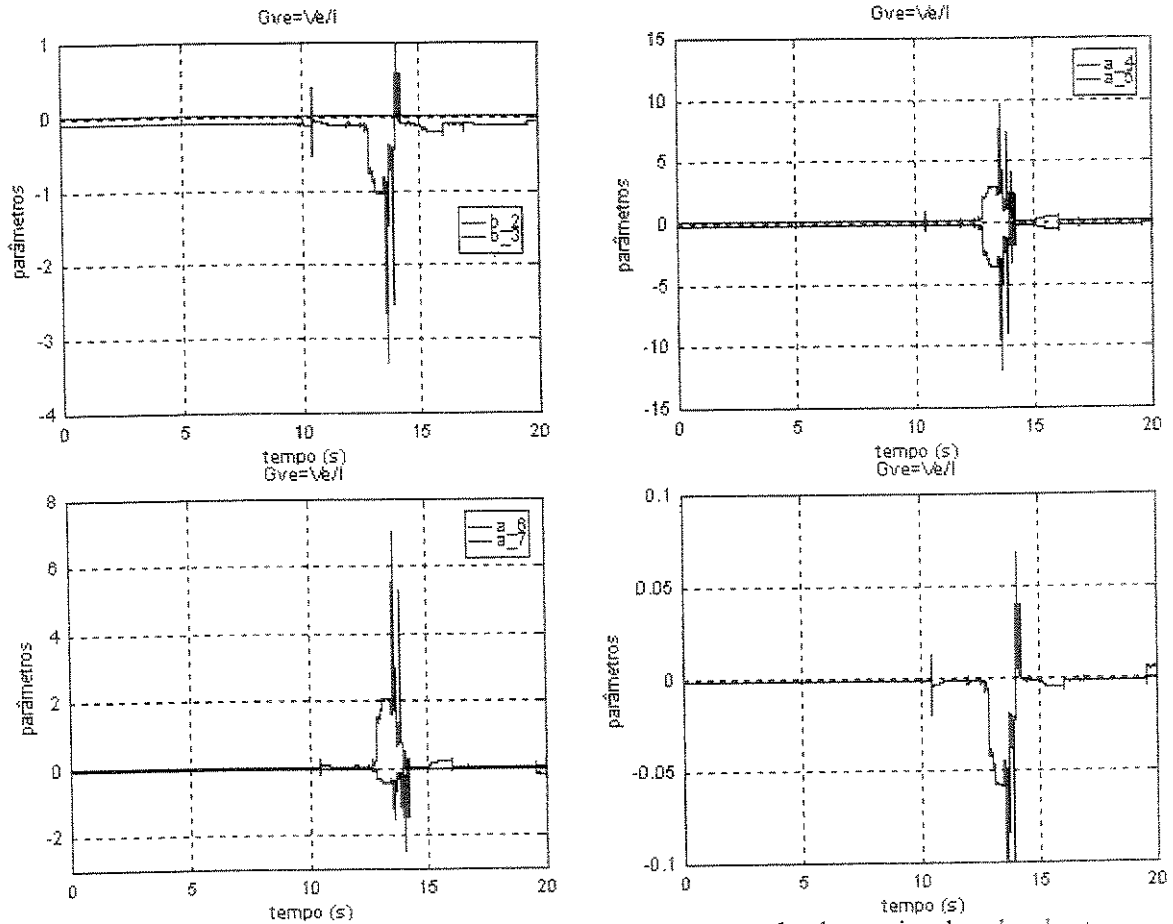


Figura 4.4-6 – Evolução temporal dos parâmetros do denominador da planta.

Os parâmetros estimados tiveram uma estimação com grandes variações durante o tempo de simulação entre os instantes 10s e 15s, e esse comportamento pode ser ligado ao comportamento da inversa da matriz de covariância,  $P$ . O módulo do traço de  $P$  pode ser visto na figura a seguir:

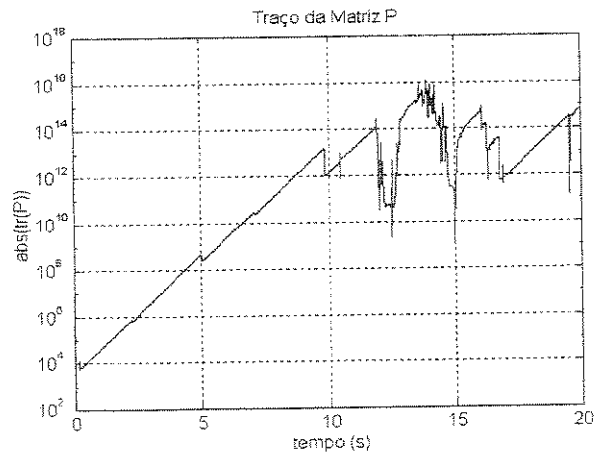
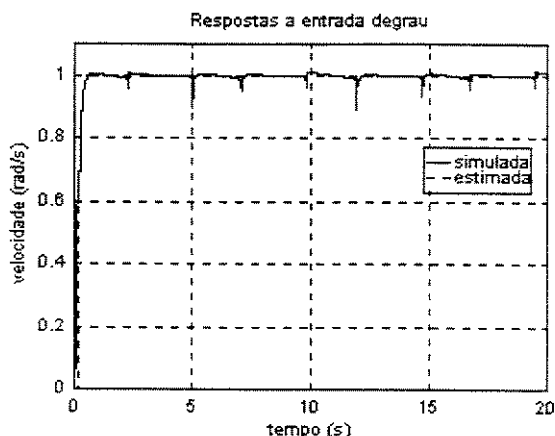


Figura 4.4-7 – Módulo do traço de  $P$ .

A Figura 4.4-7 permite observar que a matriz  $P$  cresce muito rapidamente, e isso influencia a estimação dos parâmetros da planta. Exatamente na região na qual o traço de  $P$

apresenta um transiente, os valores estimados de todos os parâmetros também sofrem um transiente. Embora esse comportamento indique o surgimento de problemas de ordem numérica na estimação, os valores de saída têm sua estimação feita de forma precisa, mostrando que para esse caso simulado, o algoritmo recursivo prevê de forma satisfatória a saída do sistema variante no tempo.



**Figura 4.4-8** – Simulação de resposta controlada para o sistema.

O controlador polinomial projetado tem como finalidade a regulação do sistema, ou seja, para esse caso simulado aplicou-se um degrau unitário de velocidade na entrada do sistema controlado e desejava-se obter como comportamento de saída a velocidade constante. Pode-se ver pela Figura 4.4-8 que a saída controlada manteve-se estável a menos de instantes nos quais aconteceram picos de velocidade. Nota-se a coincidência desses picos com saltos nos valores do traço da matriz  $P$ , observáveis na Figura 4.4-7, sugerindo a correlação entre esses eventos.

## Capítulo 5

### Análise dos Resultados e Conclusões

Este capítulo trata da análise de todos os resultados obtidos, o que inclui os resultados dos Capítulos 3, no qual foram apresentadas as estimações para os parâmetros físicos do sistema, além da estimação dos parâmetros numéricos do modelo de forma recursiva, e do Capítulo 4, no qual estão apresentados resultados de simulações e medidas de experimentos de controle tacométrico, polinomial e auto-sintonizável.

#### 5.1. Identificação do sistema

Inicialmente foram feitas as medidas experimentais para modelar os elementos de um grau de liberdade, cujos parâmetros físicos são as peças fundamentais para mais tarde permitir a construção do modelo com quatro graus de liberdade. Esses elementos são o motor e o rotor. Ambos os elementos foram modelados como tendo amortecimento de rolagem viscoso, e a obtenção desses parâmetros se deu a partir de *testes de parada* com esses elementos. Depois, foi feita a determinação da constante de torque do motor CC com ímã permanente que aciona o sistema. O momento de inércia do rotor era conhecido previamente, e o momento de inércia do motor foi determinado experimentalmente. Os valores obtidos para os momentos de inércia são coerentes pois o valor do momento de inércia obtido para o motor é duas ordens de grandezas menor que o do rotor, e isso é compatível com as características desses elementos.

Com esse conjunto de parâmetros já determinado, pôde-se prosseguir com o modelamento e fazer as identificações dos sistemas de dois graus de liberdade envolvidos. São eles: o conjunto formado por um motor e um rotor acoplados por uma correia, e o conjunto formado por dois rotores acoplados por um eixo.

O primeiro desses conjuntos tem a conexão elástica feita pela correia, que introduz uma frequência natural de oscilação. Fazendo-se um teste de parada com o conjunto foi possível obter tanto o amortecimento introduzido pela correia, quanto a constante elástica linear da correia. A

freqüência natural desse conjunto foi identificada como sendo 51Hz, com fator de amortecimento  $0,436s^{-1}$ . No sistema completo espera-se a existência de duas freqüências naturais com valores próximos a esse, pois nele existem dois conjuntos de motor e rotor conectados por correias. As constantes elásticas torcionais equivalentes da correia são: 6,88Nm e 43Nm, respectivamente para o motor e para o rotor.

O outro conjunto, formado por dois rotores e um eixo, teve seus parâmetros estimados por testes de excitação impulsiva, obtendo-se, para ele, uma freqüência natural de 102Hz, com fator de amortecimento  $0,02s^{-1}$ . A constante elástica torcional obtida para o eixo foi 650Nm/rad, e este valor é compatível com um eixo oco de aço, com uma parede de ~1mm. Este valor obtido para a constante elástica também pode ser comparado com os valores das constantes elásticas torcionais equivalentes das correias, obtidas para o motor e para o rotor, e é possível constatar que o eixo é pelo menos uma ordem de grandeza mais rígido que a correia, e o fator de amortecimento do eixo é uma ordem de grandeza menor que o amortecimento do outro conjunto.

O modelo construído a partir dos modelos reduzidos identificados tem resposta em freqüência para saída de velocidade no rotor, que apresenta alta sensibilidade para baixas freqüências e baixa sensibilidade para as freqüências naturais dos acoplamentos entre rotor e motor, e entre rotores. A saída de deformação no eixo, ao contrário da saída em velocidade, apresenta baixa sensibilidade a excitações de baixa freqüência, mas tem alta sensibilidade a freqüências em torno das ressonâncias dos acoplamentos envolvidos.

Como era desejado utilizar o estimador recursivo para controle adaptativo, foi feita uma identificação recursiva para o modelo completo, e com isso, a verificação dos resultados do modelo estimado. Também fez-se a preparação para a implementação dos controladores que utilizariam esses modelos como modelos de partida mais próximos daquilo que se obteria na estimação 'on-line'. Esses resultados mostraram a presença de um pico em 70Hz e um em 50Hz, ao invés de dois picos em 51Hz, ou seja, houve uma separação das freqüências esperadas devido a uma discrepância entre os valores medidos e o modelamento.

## 5.2. Métodos de Controle

A partir dos modelos estimados recursivamente foram calculados os três tipos propostos de controladores para o sistema: tacométrico, polinomial e auto-sintonizável, começando pelo tacométrico, passando para o polinomial e depois para o auto-sintonizável.

O controle tacométrico teve como objetivo o deslocamento do pólo identificado em 70Hz, de forma que esse pólo fosse apresentado em 90Hz. O resultado experimental mostrou que o pólo alvo não mais estava presente, e esse comportamento foi na direção desejada.

Com o sucesso do controle tacométrico a partir do modelo estimado recursivamente, passou-se para a simulação do controlador polinomial, pois isso resultaria na obtenção de valores realistas para as correntes necessárias ao controle polinomial ou auto-sintonizável que se desejava implementar no sistema. O controle polinomial simulado mostrou que o sistema controlado acompanha o modelo de referência, como desejado, e que a corrente simulada nunca ultrapassa os valores possíveis de se obter experimentalmente, o que também era desejado.

Uma vez simulada a atuação do controlador polinomial, passou-se para a simulação do controlador adaptativo em sistema com parâmetros variantes no tempo. A simulação consistiu em fazer variar senoidalmente com o tempo *um* parâmetro do numerador da planta previamente estimada recursivamente, e com isso estimar a planta variante recursivamente para se recalculer o controlador polinomial a cada iteração, obtendo a lei de controle que faria o sistema se comportar como o modelo de referência.

O estimador para o sistema completo apresenta problema de crescimento da inversa da matriz de covariância,  $P$ , crescimento esse que influencia diretamente a estimação dos parâmetros. Apesar disso, os resultados da simulação indicam que o estimador conseguiu prever o comportamento da planta variante e aplicar a lei de controle, e com isso o sistema controlado respondeu como o modelo de referência.

Na implementação para o sistema reduzido sujeito a perturbação na demanda de torque, os controladores não adaptativos falharam em manter a saída de regime no patamar desejado, visto que foram diretamente influenciados e resultando em oscilação de 14% em torno do valor desejado. O desempenho adaptativo obtido foi satisfatório, pois reduziu a oscilação na saída da planta para 0,7% em torno do regime desejado.

### 5.3. Conclusões

De acordo com os objetivos deste trabalho, foram cumpridas todas as etapas para a obtenção experimental de um sistema de controle adaptativo em uma bancada rotativa sujeita a uma demanda de torque variável.

O cálculo do controlador em tempo real é o ponto chave para a obtenção de um controlador confiável, devendo-se levar em conta não só a existência de uma solução para o sistema de equações utilizado para determinar os coeficientes dos polinômios, mas também as limitações impostas pela taxa de aquisição do sistema, a qual determina o tempo disponível para a realização dos cálculos.

O método de identificação recursiva implementado funciona de forma satisfatória e permite prever a resposta do sistema mesmo quando não se tem um modelo preciso do sistema utilizado. A restrição aplicável à estimação é a possível existência de múltiplos conjuntos de coeficientes, o que pode levar a inconsistências no projeto dos polinômios do controlador. Por outro lado, esse fator depende do sistema com o qual se está trabalhando e para o caso escolhido o método de obtenção do controlador não sofreu interferências desse tipo durante a atuação.

Os controladores não adaptativos implementados tiveram seus desempenhos compatíveis com suas características de projetos. O controlador tacométrico teve resultado satisfatório para a situação de regime estacionário, mas quando se varia a demanda de torque, já não tem bom desempenho. Para o controlador polinomial aplicam-se as mesmas restrições, pois o mesmo não se mostrou robusto no sentido de acomodar variações na demanda de torque.

A estratégia adaptativa implementada resultou na atuação dentro dos limites desejados, ou seja, incorporou as alterações da planta de forma a manter a saída dentro dos níveis desejados enquanto o sistema era submetido a uma demanda de torque variável.

Os resultados experimentais obtidos com o sistema reduzido, somados às simulações com o sistema completo, permitem especular que a implementação experimental com o sistema completo é viável, e isso indica a possibilidade de aplicação inclusive para problemas industriais. O algoritmo para controle auto-sintonizável tratado aqui pode ter seu desempenho melhorado se for incluído no algoritmo o reajuste da inversa da matriz de covariância,  $P$ , assunto tratado por ÅSTRÖM e WITTENMARK (1989) e GOODWIN, ELLIOT et al (1983).

---

Extensões para a mesma linha de tratamento são a implementação de controladores adaptativos diretos, ou seja, aqueles controladores para os quais se obtém diretamente os parâmetros dos polinômios do controlador a partir do método de estimação, sem necessidade de uma atualização explícita dos parâmetros do modelo do sistema.

Para ampliar a classe de problemas que podem ser submetidos a métodos de controle como os descritos nesse trabalho, é interessante destacar a abordagem para controle adaptativo direto ou indireto utilizando redes neurais e lógica nebulosa, pois embora não estejam diretamente relacionadas às técnicas consideradas nessa monografia, permitem tratar a questão das não linearidades intrínsecas ao problema de controle de sistemas rotativos como o estudado.

...  
...  
...

## Bibliografia

- ÅSTRÖM, K. J., WITTENMARK, B., *On Self Tuning Regulators*, Automatica, v.9, p.185-199, 1973.
- ÅSTRÖM, K. J., WITTENMARK, B., *Computer Controlled Systems – Theory and Design*, Prentice Hall, New Jersey, USA, 1984.
- ÅSTRÖM, K.J., WITTENMARK, B. *Adaptive Control*, 2<sup>a</sup> ed., Addison-Wesley Publ. Co., 1989.
- BANDYOPADHYAY, B., UNBENHAUEN, H., PATRE, B.M., *A New Algorithm for Compensator Design for Higher-order Systems via Reduced Model*, Automatica, v.34, n.7, p.917-920, 1998.
- CHEN, H.F., CAO, X.R., *Controllability is Not Necessary for Adaptive Pole Placement Control*, IEEE Trans. Autom. Control, v.49, n.9, Sept. 1997.
- FRANKLIN, G.F., POWELL, J.D., WORKMAN M.L. *Digital Control of Dynamic Systems*, Addison-Wesley Publ. Co., 1990.
- GOODWIN, G.C., SIN, K.S. *Adaptive Control of Nonminimum Phase Systems*, IEEE Trans. Aut. Control, v.AC-26, n.2, April 1981.
- GOODWIN, G.C., ELLIOT, H., TEOH, E.K. *Deterministic Convergence of a Self-tuning Regulator with Covariance Resetting*, IEEE Proc. D, v.130, n.1, January 1983.
- GUO, L., CEHN, H.F. *The Åström-Wittenmark Self-tuning Regulator Revisited and ELS-based Adaptive Trackers*, IEEE Trans. Aut. Control, v.36, n.7, July 1991.
- KARACAN, S., HAPOGLU, H., CABBAR, Y., ALPBAZ, M., *Pole Placement Self Tuning Control for Packed Distillation Column*, Chem. Eng. Process., v.36, p.309-315, 1997.
- KERNIGHAM, B. W., RITCHIE, D. M., *C – A Linguagem de Programação*, 1<sup>a</sup> ed, 9<sup>a</sup> reimpressão, Editora Campus, Rio de Janeiro, 1994.
- KHATER, E. *Acelerômetro Angular*, Rio de Janeiro: COPPE/Universidade Federal do Rio de Janeiro, 1992 (Tese de Mestrado).
- KIDD, P.T., *Comparison of the Performance of a Model-reference Adaptive System and a Classical Linear Control System Under Non-ideal Conditions*, Int. J. Control, v.42, n.3, p.671-694, 1985.

OGATA, K., *Discrete-Time Control Systems*, 2<sup>nd</sup> ed., Prentice Hall, New Jersey, USA, 1995.

SHAFIQ, M., LU, J., TAHAGI, T., *A New Method for Self Tuning Control of Nonminimum Phase Continuous Systems Based on Pole-zero Placement*, IEICE Trans. Fund., v.E79-A, n.4, April 1996.

SHAIAM, B., HASSUL, M., *Control System Design Using MATLAB*, Prentice Hall, New Jersey, USA, 1993.

UNBENHAUEN, H., KEUCHEL, U., *Model Reference Adaptive Control Applied to Electrical Machines* Int. J. Adapt. Contr. And Sign. Process., v.6, p.95-109, 1992.

WILSON, B.H., TAYLOR, J.H., *A Frequency Domain Model-Order-Deduction Algorithm for Linear Systems*, J. Dyn. Systems Meas. Control, v.120, p.185-192, June 1998.

ZHANG, Y., FENG, C.B., *Identification of Closed-loop Systems via Least Squares Method*, Int. J. Adapt. Control Signal Process., v.12, p.28-38, 1998.

---

## Apêndice A

### Instrumentação

A seguir estão listados os diagramas dos circuitos da instrumentação eletrônica utilizada para medir a deformação no eixo através de um conjunto de extensômetros (“strain gauges”) montado em configuração de ponte de Wheatstone, e os circuitos para fornecer e medir as correntes de acionamento e perturbação para os motores CC utilizados. As leituras de corrente e deformação eram feitas por três canais analógicos da placa de aquisição Advantech PCL-818HG, cuja taxa de aquisição máxima é 100kHz para todos canais, e sendo o sinal de corrente para acionamento fornecido pela saída analógica da mesma placa. A leitura de velocidades foi feita a partir de dois encoders de 2500 pulsos por rotação em quadratura, pulsos esses que eram analisados pelo programa do Apêndice B e convertidos em leituras de velocidade do motor e do rotor.

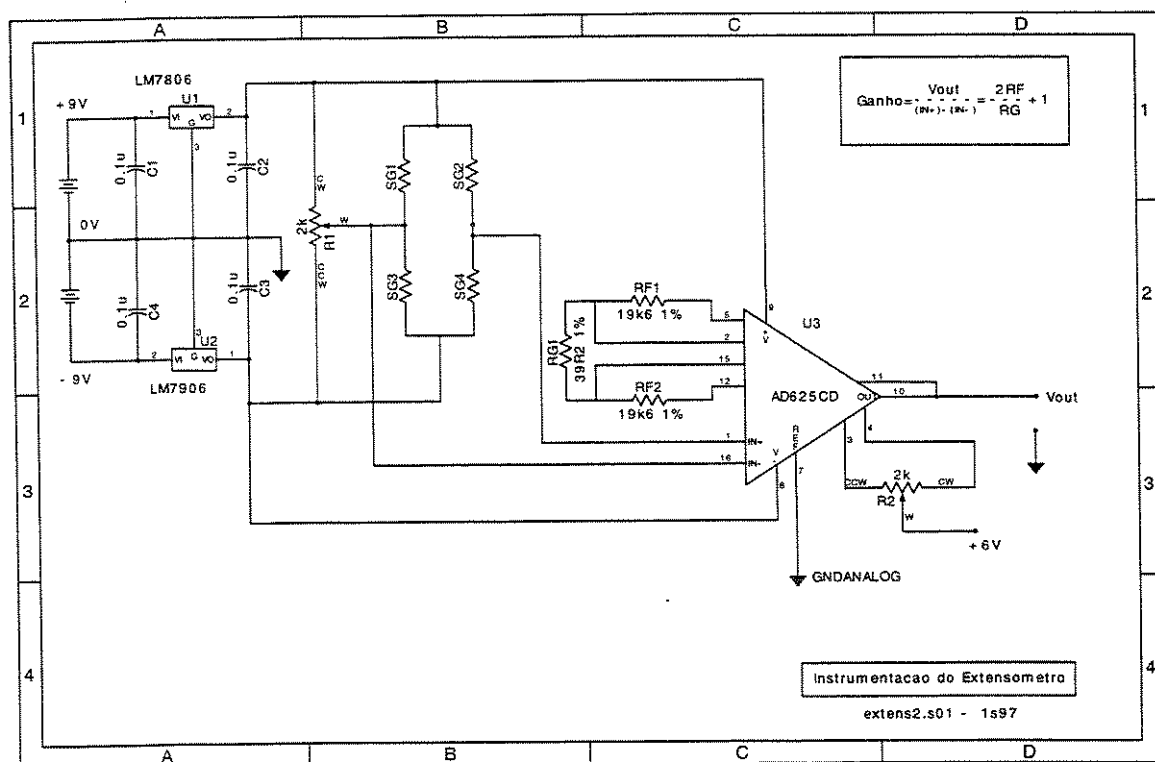
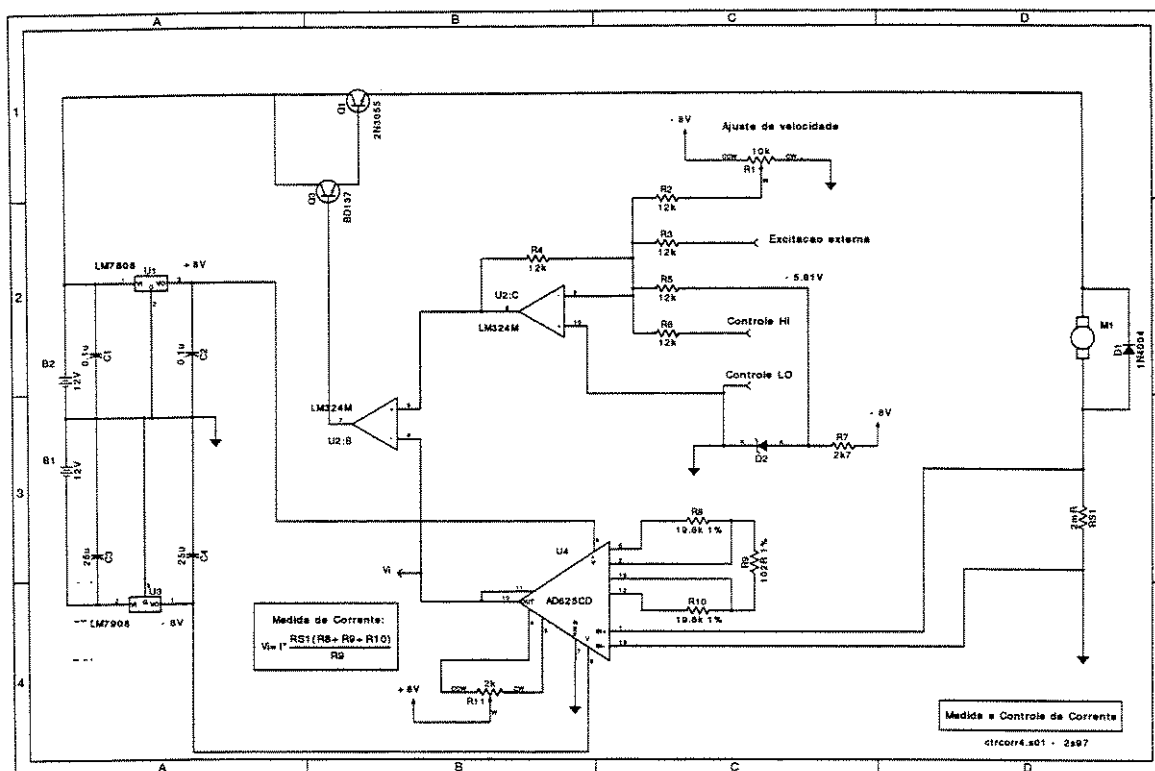


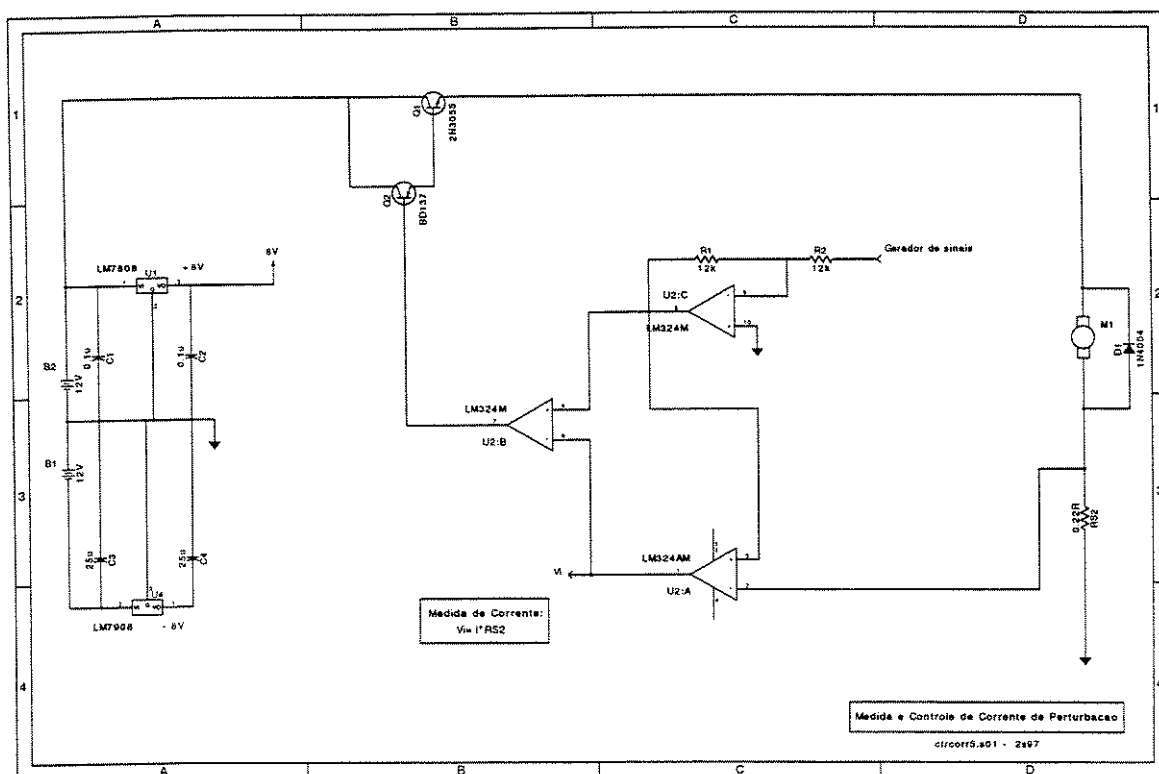
Figura A-1 – Circuito de leitura de deformação no eixo através de extensômetro.

O sinal de deformação gerado pelos extensômetros montados em ponte é lido pelo amplificador de instrumentação AD625CD, que amplifica com ganho 327 o sinal de tensão gerado, e envia esse sinal para uma das entradas analógicas da placa de aquisição PCL818HG.



**Figura A-2** – Circuito para fornecimento e leitura de corrente de acionamento.

A Figura A-2 mostra que a leitura de corrente é feita através da queda de tensão no resistor “shunt” RS1, lida de forma diferencial pelo amplificador AD625CD (ganho 327). Essa leitura é enviada para uma das entradas analógicas da placa PCL818HG, e também é utilizada na malha de controle de corrente. O sinal de tensão proveniente da leitura de corrente é comparado com um sinal de referência: o regime (“set-point”) é ajustado manualmente, e a lei de controle proveniente da saída analógica é somada ao regime.



**Figura A-3 – Circuito para leitura e fornecimento de corrente de perturbação.**

A leitura e fornecimento da corrente de perturbação é feita com um circuito similar ao da corrente de acionamento da Figura A-2, com a diferença de o circuito da Figura A-3 não utilizar um amplificador de instrumentação AD625CD, mas sim um amplificador operacional LM324 em conjunto com um resistor “shunt” com maior resistência elétrica para efetuar a medida de corrente de perturbação.



```

extern "C" {
#endif
_cdecl _pcl818HG
(int, unsigned int *);
#ifdef __cplusplus
}
#endif

extern pcl818HG(int, unsigned int *);

#define Nv 500          // # aquis. da porta digital para o proc. da vel.
#define N 4             // # aquis. de correntes (motor1 e 4) sg e somador.
#define Max 7030        // tamanho maximo de aquisicoes de sg e cm
#define dcorrente 4.1   //4.96 // nivel dc da corrente (amperes)
#define G 385.3137      // ganho de tensao no shunt
#define Pi 3.141596

float Rm =2e-3; //10.8    // (ohm) resistencia associada 'a corrente do motor
int ganho=8.;      // ganho da placa: 0: +/- 5V; 5: 0-1V; 8:+/-10V
//int ganho=9; // +/-1V; ref. p. 118 do manual
//float g=1;
float g=10.;

float polsaida=6.11; // offset da saida D/A (6.01V do zener + 0.1V de perda no cabo)

float Ka=0.0043;    // constante da armadura do motor
float Kc=12;        // const. de rigid. torc. da correia

unsigned int param[60]; // usado na inicializacao da placa
unsigned char DATA[Nv]; // buffer para dados vindo do encoder
unsigned int DATA_CSG[N]; // Conversion DATA buffer, for A/D
unsigned int far *dat; // dat=DATA_CSG
unsigned int DATA_SAIDA[1]; // Conversion DATA buffer, for D/A
unsigned int far *saida; // saida=DATA_SAIDA

float velocl[Max]; // grava velocidade 1 - velocidade no eixo
//float veloc2[Max]; // grava velocidade 2 - velocidade no motor
float veloc2[1];
float vref[Max]; // grava velocidade de referencia
//float vref[1]; // grava velocidade de referencia

float w0; // velocidade inicial do motor (rpm).

// float Kvi=40;
//float Kvi=55.045; //RPM/A // Constante de proporcionalidade entre
// corrente do motor 1 e velocidade do eixo.

// motor com encoder:
float Kvi=350; // rad/s/A

// variaveis de media do sinal do extensometro:
float ultimos_sg[30];
float media_sg;
int isg;

// abaixo esta a lista de variaveis de estado

float x1[3]={0,0,0}; // corrente do motor {atual,t-1,t-2}
float x2=0;
float x3=0;
float x3ant=0;
float x3ant2=0;
float x4=0;
float x5ant=0;
float x5=0;
float x6=0;
float vell=0; // velocidade no eixo
float vel2=0; // velocidade no motor
float tetam=0;
float tetall=0;
//float x7=0;

// vetor de ganhos:

```

```

float k1=0;
float k2=0;
float k3=0;
float k4=0;
float k5=0;
float k6=0;
//float k7=0;

// #####
// # #
// # EIXO 2 # time
// # #
// #####
float TETA=2.7345e-4; // fator de conversao da deformacao para radianos (para eixo 2)

//jrbn tempo entre duas medidas sucessivas dos estados
float T=0.0029; //jrbn// com velocidade: 0.0029333;

float amostr=17.; // TA=amostr*T - tempo final de amostragem

clock_t t_i,t_f; // para medir tempo total de amostragem.

// relacao de diametros
float n=100/40;

float dcvel=30.0; //rad/s
float dcvel0; //rad/s
float i_reg; // A - corrente de regime
float dcvelradn=78.5398; //dcvel*n*pi/30
int refDA=10; // Volts -> referencia para D/A

char nome[80]; // nome do arquivo .m
//char dir[80]="u:\\matlab\\tese\\tese_97\\"; // diretorio de trabalho
//char dir[80]="u:\\brandao\\tese\\matlab\\bancada\\uso\\"; // diretorio de trabalho
char dir[80]="w:\\ftp\\brandao\\uso\\"; // diretorio de trabalho
char dir_nome[160]; // diretorio + nome
FILE -- *arq;

unsigned int huge vetorCM[Max]; // vetor para cor. motor
//unsigned int huge vetorCM2[Max]; // vetor para a corrente do motor de perturbacao
unsigned int huge vetorCM2[1]; // vetor para a corrente do motor de perturbacao
unsigned int huge vetorSG[Max]; // vetor para sg
// unsigned int huge vetorSOM[Max]; // vetor para somador
// unsigned int huge vetorU[Max]; // contem informacoes sobre u
// int ref_vel[Max]; // vetor de velocidades de entrada (controle tacometrico)

long int index; // index do loop principal
int conta; // controla incremento de index;

int max=Max; // cria variavel com valor MAX

//float u_aleat[4096]={};

float uflag=1; // utilizado para poder variar a saida
float vflag=10; // rpm

// inicializacao do gerador aleatorio:
double u_aux_i[5]={0,0,0,0,0};
double u_aux_o[5]={0,0,0,0,0};

float aux_u[2]={0,0}; // vetor auxiliar={atual,t-1}

float kt,kd; // ganhos do controle tacometrico

// variaveis da estimacao de parametros:
double K[2],
    P[2][2],
    fi[2],
    erro,
    tetach[2][1], // matriz com parametros a cada amostragem.
// tetach[2][Max], // matriz com parametros a cada amostragem.
    v_est[1], // vetor com velocidades estimadas.
// v_est[Max], // vetor com velocidades estimadas.
    Ktbd; // Ktbd=Kt*bd[index]=constante do controle adaptativo.

```

```

// variaveis do controle RST adaptativo:
int na,nb,nc; // ordens dos polinomios A,B e Ac
MATRIX Ac,B_aux,A_aux;
MATRIX E,D,M;

//
double v1=0,v2=0, // guardam a media das velocidades e das correntes
       i1=0,i2=0;

/*****/

// Prototipos:
void processa_e_controlo(int flag);

/*****/

void inic_estim(double Kt, double Kd, float vel)
// Inicializa parametros da identificacao recursiva por minimos
// quadrados.
{
    double //b=45., // motor + 1 polia (no motor)
           //b=18., // motor + 1 polia (na polia)
           //b=200., // motor
           b=9.4, // motores+rotor (no rotor)
           //a=0.41; // parametros de G(s)=b/(s+a)=45/(s+.41).
           a=0.14; //motores+rotor

    K[0]=K[1]=0;

    P[0][0]=P[1][1]=1000; // P0=1000*I
    P[1][0]=P[0][1]=0;

    fi[0]=-vel;
    fi[1]=x1[0]; // inicializa fi com velocidade e corrente atuais
    erro=0;

    // parametros iniciais=parametros discretos da planta linear:
    tetach[0][0]=-exp(-a*T*amostr); // ad
    tetach[1][0]=b/a*(1.-exp(-a*T*amostr)); // bd
    // tetach[0][0]=-0.9800; // T=17*2.9ms
    // tetach[1][0]=2.196;

    v_est[index]=vel; // inicializa com ultima velocidade medida.

    kt=Kt;
    kd=1/tetach[1][0]*(1+tetach[0][0])+kt; // condicao para ganho estatico unitario
    Ktbd=Kt*tetach[1][0]; // inicializa Ktbd.

} // inic_param

/*****/

void estima_param(int aux, float vel_atual, float vel_ant)
// Estima parametros utilizando quadrados minimos recursivo.
{
    static float aux_K[2],
               aux2;
    static float la=.978; // fator de esquecimento (ate' 10-03: .993)
    int i,j;

    // inicia estimacao colocando os primeiros valores
    if (index<aux) { // no minimo 4 !!
        tetach[0][index]=tetach[0][0];
        tetach[1][index]=tetach[1][0];
        v_est[index]=vel_atual;
        fi[0]=-vel_atual;
    // fi[0]=-veloc2[index];
        fi[1]=x1[0];
    }; // if index < aux

    // Estima:
    if (index>=aux) { // no minimo 4 !!
        fi[0]=-vel_ant;
    // fi[0]=-veloc2[index-1];
    }
}

```

```

    fi[1]=x1[2];

    // estima valor atual da velocidade a partir da estimacao de parametros:
    v_est[index]=tetach[0][index-1]*fi[0]+tetach[1][index-1]*fi[1];

    //erro entre valor estimado e medido:
    erro=vel_atual-v_est[index];
    // erro=veloc2[index]-v_est[index];

    // calcula as matrizes da estimacao P e K, e o novo vetor de parametro tetach:
    aux_K[0]=P[0][0]*fi[0]+P[0][1]*fi[1];
    aux_K[1]=P[1][0]*fi[0]+P[1][1]*fi[1];
    aux2=1a+(fi[0]*aux_K[0]+fi[1]*aux_K[1]);

    for (i=0;i<2;i++) {
        K[i]=aux_K[i]/aux2;
        tetach[i][index]=tetach[i][index-1]+K[i]*erro;
        for (j=0;j<2;j++)
            P[i][j]=(P[i][j]-aux_K[i]*aux_K[j]/aux2)/1a;
    }; // for i

}; // if index >=aux
} // estima_param

/*****/

void taco_adapt(int aux)
//===
// Adaptacao do controle tacometrico
//===
{
    if (index>=aux+2*aux & (index % 1==0)) { // para desabilitar, usar Max
        // Ktbd=Kt*bd=constante:
        kt=Ktbd/tetach[1][index];

        // Aloca polos para zero (kt=ad^2/4/bd):
        // kt=tetach[0][index]*tetach[0][index]/4/tetach[1][index];

        // condicao de ganho estatico unitario
        kd=(1+tetach[0][index])/tetach[1][index]+kt;

    }; // if index >=aux (atualiza kt e kd)
} // FIM adaptacao taco_ad

/*****/

void diofan(float *R, float *S, MATRIX B, int nb, MATRIX A, int n, MATRIX Ac, int nc, MATRIX
E, MATRIX M, MATRIX D)
//
// Calcula coeficientes dos polinomios R e S, dada a equacao diofantina:
//
//
// AR+BS=Ac
//
// B,A,Ac - polinomios conhecidos.
// nb,n,nc - ordem desses polinomios (ex: 1a ordem: n=1 -> float A[n+1]=A[2])
//
// Ogata, K., Discrete Time Control Systems, 2 ed., p.519
//
{
    int i,j;

    // n =2; //MatCol(A)-1; // ordem de A
    // nb=1; //MatCol(B)-1; // ordem de B menor= ordem de A
    // nc=3; //MatCol(Ac)-1; // ordem do observador

    mat_fill(E,ZERO_MATRIX);
    for (j=0;j<n;j++) {
        for (i=j;i<=(j+n);i++)
            E[i][j]=A[n-i+j][0];
    }
}

```

---

```

    for (i=j; i<=(j+nb); i++)
        E[i][j+n]=B[nb-i+j][0];
    }

    mat_fill(D, ZERO_MATRIX);
    for (i=0; i<=nc; i++)
        D[i][0]=Ac[nc-i][0];

    mat_fill(M, ZERO_MATRIX);
    M=mat_lsolve(E,D);

// Separa M em R e S:
for (i=0; i<n; i++) {
    *(R+i)=M[n-1-i][0];
    *(S+i)=M[2*n-1-i][0];
}

// mat_free(E);
// mat_free(D);
// mat_free(M);

} // fim - diofan

/*****/

void le_velocidade(void)
{
    int i;

    for (i=0; i<Nv; i++) DATA[i]=inportb(0x303);
}

/*****/
/*
void processa_v(void)
{
    int i,Np2,ini2,fim2;
    unsigned char dado;

    // processamento do encoder 2

    i=-1;
    do
    {
        i++;
        dado=(DATA[i] & 0x10);
    } while (dado != 0x00);          // ate achar a borda de descida
    do
    {
        i++;
        dado=(DATA[i] & 0x10);
    } while (dado != 0x10);          // ate' achar a borda de subida

    Np2=0; ini2=i;
    do
    {
        do
        {
            i++;
            if (i==Nv) break;
            dado=(DATA[i] & 0x10);
        } while (dado != 0x00);
        do
        {
            i++;
            if (i==Nv) break;
            dado=(DATA[i] & 0x10);
        } while (dado != 0x10);
        if (i<Nv)
        {
            fim2=i;
            Np2++;
        }
    } while (i<Nv-1);
}

```

---

```

        if (Np2==0) vel2=0;
    else

        // encoder c/ 2500 pulsos/s:
        vel2=(1583.3613)*((float)(Np2))/((float)(fim2-ini2)); // rad/s

        // encoder c/ 5000 pulsos/s:
        vel2=(1583.3613)*((float)(Np2))/((float)(fim2-ini2))/2; // rad/s
    }
    */

/*****/

/*****/
void processa_v(void)
{
    int i,Np2,ini2,fim2;
    unsigned char dado0,dado4; // bits 0 e 4 do sinal dos encoders
    unsigned char masc0=0x01, masc4=0x10; // mascaras para ler bits 0 e 4.
    unsigned char byte_zero=0x00;

    // processamento do encoder 1

    i=-1;
    do
    {
        i++;
        dado0=(DATA[i] & masc0);
    } while (dado0 != byte_zero); // ate achar a borda de descida
    do
    {
        i++;
        dado0=(DATA[i] & masc0);
    } while (dado0 != masc0); // ate achar a borda de subida

    Np2=0; ini2=i;
    do
    {
        do
        {
            i++;
            if (i==Nv) break;
            dado0=(DATA[i] & masc0);
        } while (dado0 != byte_zero);
        do
        {
            i++;
            if (i==Nv) break;
            dado0=(DATA[i] & masc0);
        } while (dado0 != masc0);
        if (i<Nv)
        {
            fim2=i;
            Np2++;
        }
    } while (i<Nv-1);

    if (Np2==0) vel1=0;
else

    // encoder c/ 2500 pulsos/s:
    vel1=(1583.3613)*((float)(Np2))/((float)(fim2-ini2)); // rad/s

    // processamento do encoder 2

    i=-1;
    do
    {
        i++;
        dado4=(DATA[i] & masc4);
    } while (dado4 != byte_zero); // ate achar a borda de descida
    do
    {
        i++;

```

```

        dado4=(DATA[i] & masc4);
    } while (dado4 != masc4);        // ate' achar a borda de subida

    Np2=0; ini2=i;
    do
    {
        do
        {
            i++;
            if (i==Nv) break;
            dado4=(DATA[i] & masc4);
        } while (dado4 != byte_zero);
        do
        {
            i++;
            if (i==Nv) break;
            dado4=(DATA[i] & masc4);
        } while (dado4 != masc4);
        if (i<Nv)
        {
            fim2=i;
            Np2++;
        }
    } while (i<Nv-1);

    if (Np2==0) vel2=0;
else
    // encoder c/ 2500 pulsos/s:
    vel2=(1583.3613)*((float)(Np2))/((float)(fim2-ini2)); // rad/s

    // encoder c/ 5000 pulsos/s:
    vel2=(1583.3613)*((float)(Np2))/((float)(fim2-ini2))/2; // rad/s
}

/*****/

void inicializa(void)
{
    int i;

    dat = DATA_CSG;
    saida= DATA_SAIDA;
    param[0] = 0;          /* Board number */
    param[1] = 0x300;      /* Base I/O address */
    param[4] = 2;          /* IRQ level : IRQ2 */
    param[5] = 2;          /* Pacer rate = 1M / (2 * 5) = 100KHz */
    param[6] = 5;
    param[7] = 0;          /* Trigger mode, 0 : pacer trigger */
    param[8] = 0;          /* Non-cyclic */
    param[10] = FP_OFF(dat); /* Offset of A/D DATA buffer A */
    param[11] = FP_SEG(dat); /* Segment of A/D DATA buffer A */
    param[12] = 0;          /* Data buffer B address, 0 if not used */
    param[13] = 0;          /* must set to 0. */
    param[14] = N;          /* A/D conversion number */
    param[15] = 0;          /* A/D conversion start channel */
    param[16] = N-1;        /* A/D conversion stop channel */
    param[17] = ganho;      /* Gain code */
    param[20] = FP_OFF(saida); /* Offset of D/A DATA buffer A */
    param[21] = FP_SEG(saida); /* Segment of D/A DATA buffer A */
    param[22] = 0;          /* Data buffer B address, 0 if not used */
    param[23] = 0;          /* must set to 0 */
    param[24] = 1;          /* number of conversions */
    param[25] = 0;          /* D/A start channel */
    param[26] = 0;          /* D/A stop channel */

    clrscr();

    pcl818HG(3,param);
    if (param[45]!=0)
    {
        printf("\nDriver initialization failed!");
        exit(1);
    }
}

```

```

    }

    pci818HG(4,param);
    if (param[45]!=0)
    {
        printf("\nA/D Initialization failed!");
        exit(1);
    }

    pci818HG(12,param);
    if (param[45]!=0)
    {
        printf("\nD/A Initialization failed!");
        exit(1);
    }

    pci818HG(20,param); // inicializacao do D/I
    if (param[45]!=0)
    {
        printf("\nD/I Initialization failed!");
        exit(1);
    }
}

/*****/
void beep(void)
{
    sound(300); delay(50); nosound();
}

/*****/
void toque(int tempo)
{
    // avisa:
    beep();
    delay(tempo);

    // primeiro
    beep();
    delay(900);

    // segundo
    beep();
    delay(900);
}

/*****/
void mostra_v_c(void)
{
    char ch;

    gotoxy(25,17); printf("-- VELOCIDADE --");
    gotoxy(25,18); printf("EIXO : ");
    gotoxy(35,18); printf(" rad/s");
    gotoxy(25,19); printf("MOTOR :");
    gotoxy(35,19); printf(" rad/s");
    gotoxy(25,20); printf("=====");
    gotoxy(22,21); printf("CORRENTE :      A");
    gotoxy(22,22); printf("CORR_PERT :      A");

    while (kbhit()) ch=getch();

    do
    {
        le_velocidade();
        processa_v();
        processa_e_controlo(0); // le canais analogicos

        gotoxy(32,18);
        vl=(.95*vl+.05*vell);
        printf("%3.f",vl); // rad/s *9.5493); // rpm
    }

```

```

        gotoxy(32,19);
        v2=(.95*v2+.05*vel2);
        printf("%3.f",v2); // rad/s *9.5493); // rpm
        gotoxy(32,21);
        i1=(.95*i1+.05*((float)(DATA_CSG[0] & 0xFFB) - 2047.0)*g*0.4882812/Rm/1000);
//      printf("%2.4f",((float)(DATA_CSG[0] & 0xFFB) - 2047.0)*g*0.4882812/Rm/1000);
//      amperes
        printf("%2.2f",i1); // amperes
        gotoxy(32,22);
        i2=(.95*i2+.05*((float)(DATA_CSG[3] & 0xFFF) - 2047.0)*g*0.4882812/.23/1000);
//      printf("%2.4f",((float)(DATA_CSG[3] & 0xFFF) - 2047.0)*g*0.4882812/.23/1000);
//      amperes - .23R -resistencia de leitura
        printf("%2.2f",i2); // amperes - .23R -resistencia de leitura
    }
    while (!kbhit());
    while (kbhit()) ch=getch();
}

//*****

double filt1_2wn(double ent[5], double sai[5])
//
//      Filtro discreto passa baixa de quarta
//      ordem que filtra sinal sai com entrada ent,
//      com frequencia de corte=1/2*Fnyquist
//
{
    static float b[5]={ 0.09940,0.2923,0.4031,0.2923,0.09940},
        a[4]={ -0.1518,0.8402,-0.1864,0.1168};
    return(b[0]*ent[0]+b[1]*ent[1]+b[2]*ent[2]+b[3]*ent[3]+b[4]*ent[4]
        -a[0]*sai[1]-a[1]*sai[2]-a[2]*sai[3]-a[3]*sai[4]);
} // fim filt1_2wn

//*****

double filt3_4wn(double ent[5], double sai[5])
{
    static float b[5]={ 0.2924, 1.1069, 1.6308, 1.1069, 0.2924},
        a[4]={ 2.0292, 1.9590, 0.8665, 0.1889};
    return(b[0]*ent[0]+b[1]*ent[1]+b[2]*ent[2]+b[3]*ent[3]+b[4]*ent[4]
        -a[0]*sai[1]-a[1]*sai[2]-a[2]*sai[3]-a[3]*sai[4]);
} // fim filt3_4wn

//*****

double filt5_6wn(double ent[5], double sai[5])
{
    static float b[5]={ 0.3482, 1.3606, 2.0253, 1.3606, 0.3482},
        a[4]={ 2.7210, 2.9982, 1.5320, 0.3174};
    return(b[0]*ent[0]+b[1]*ent[1]+b[2]*ent[2]+b[3]*ent[3]+b[4]*ent[4]
        -a[0]*sai[1]-a[1]*sai[2]-a[2]*sai[3]-a[3]*sai[4]);
} // fim filt5_6wn

double filt_95wn(double ent[5], double sai[5])
{
    static float b[5]={ 0.5613, 2.2202, 3.3178, 2.2202, 0.5613},
        a[4]={ 3.5957, 4.9170, 3.0277, 0.7090};
    return(b[0]*ent[0]+b[1]*ent[1]+b[2]*ent[2]+b[3]*ent[3]+b[4]*ent[4]
        -a[0]*sai[1]-a[1]*sai[2]-a[2]*sai[3]-a[3]*sai[4]);
} // fim filt_95wn

//*****

void pergunta_constantes(void)
{
    char ch;

```

```

while (kbhit()) ch=getch();
clrscr();
gotoxy(10,8); printf("0 - Sair");
gotoxy(10,10); printf("1 - Define o diretorio de trabalho");
gotoxy(10,12); printf("2 - Continua com o diretorio corrente:\n %s \n",dir);
ch=getch();
if (ch=='0') { clrscr(); normvideo(); exit(0); }
else
if (ch=='1')
{
    gotoxy(1,16); printf("Diretorio de trabalho: "); clrscr();
    scanf("%s",&dir);
}

while (kbhit()) ch=getch();
clrscr();

gotoxy(1,1); printf("Const. Kt (realimentacao de velocidade) = %f, deseja mudar?
[S/N]",k1);
while (kbhit()) ch=getch();
ch=getch();
if ( (ch=='S') || (ch=='s') )
{
    gotoxy(10,2); printf("Defina o valor de Kt: ");
    scanf("%f",&k1);
}

gotoxy(1,4); printf("Const. K (ganho direto de velocidade) = %f, deseja mudar?
[S/N]",k2);
while (kbhit()) ch=getch();
ch=getch();
if ( (ch=='S') || (ch=='s') )
{
    gotoxy(10,5); printf("Defina o valor de K: ");
    scanf("%f",&k2);
}

gotoxy(1,7); printf("Amostragem = %2.f * %f segundos, deseja mudar? [S/N]",amostr,T);
while (kbhit()) ch=getch();
ch=getch();
if ( (ch=='S') || (ch=='s') )
{
    gotoxy(10,8); printf("Novo valor de amostragem (num. ineiro): ");
    scanf("%f",&amostr);
}

gotoxy(1,19); printf("Velocidade desejada (rad/s) = %f, deseja mudar? [S/N]",dcvel);
while (kbhit()) ch=getch();
ch=getch();
if ( (ch=='S') || (ch=='s') )
{
    gotoxy(10,20); printf("Defina o valor dc de velocidade: (rad/s) ");
    scanf("%f",&dcvel);
    dcvelradn=dcvel*n*0.1047196;
}

gotoxy(1,21); printf("Range da placa = %f V pico, deseja mudar? [S/N]",g);
while (kbhit()) ch=getch();
ch=getch();
if ( (ch=='S') || (ch=='s') )
{
    gotoxy(10,22); printf("Digite 1 para ganho=0.5");
    gotoxy(10,23); printf("Digite 2 para ganho=1.0");
    gotoxy(10,24); printf("Digite 3 para ganho=5.0\n");
    gotoxy(10,25); printf("Digite 4 para ganho=10.0");
    while (kbhit()) ch=getch();
    ch=getch();

    if (ch=='2')
    {
        g=1.0; ganho=9; }
    else
    if (ch=='1')
    {
        g=0.5; ganho=1; }
    else

```

```

        if (ch=='3')
        { g=5.0; ganho=0; }
        else
        { g=10.0; ganho=8; }

        inicializa();
    }
    clrscr();
    gotoxy(1,2); printf("Referencia para saida D/A = %d, deseja mudar?",refDA);
    while (kbhit()) ch=getch();
    ch=getch();
    if ( (ch=='S') || (ch=='s') )
    {
        if (refDA==5) refDA=10;
        else refDA=5;
    }
}

/*****/

float sat(in) // funcao saturacao
float in;

{

    if (in>1.) return(1.);
    if (in<-1.) return(-1.);
    else return(in);

} // sat

/*****/

void processa_e_controlo(int flag)
{
    static double u; // saida para a placa
    static double wi=20.; // Hz - freq. inicial do sweep
    static double wf=80.; // Hz - freq final do sweep
    static double A0=-6e-4; // -1*media do sweep
    static double A=.5*.77; // amplitude
    static float n1=.01; // -inclinacao da rampa inicial
    static float n2=.2; // -inclinacao da rampa final
    static float freq=1.5; // frequencia (Hz)
    static float r[2]={1.,-5.909e-1}, // motor e motor+polia
        //s[2]={7.0740e-3,0.}, // motor
        //s[2]={3.144e-2,0.}, // motor
        s[2]={7.86e-2,0.}, // eixo
        //s[2]={0.11502,0.}, // polial (motor+polias+eixo)
        // t[2]={7.9127e-3,0.}, // motor
        //t[2]={3.5167e-2,0.}, // motor
        t[2]={8.7918e-2,0.}, // eixo
        //t[2]={0.12866,0}, // polial (motor+polias+eixo)

    float dither=0; // sinal de dithering para melhorar estimacao

    int aux=512;

    float tetamp;

    pcl1818HG(5,param); // A/D conv. with S/W data transfer */

/*
    if (param[45] != 0)
    {
        printf("\n\nA/D convrsion failed! param[45] = %d ",param[45]);
        clrscr(); normvideo(); exit(1);
    }
*/

// () vetorSOM[index]=DATA_CSG[2]; // guarda tensao do somador em vetorSOM

// x1 = ( ( (float) (DATA_CSG[0] & 0x0FFF) ) - 2047.0)*g*0.4882812/Rm - dccorrente;
// corrente do motor

```

```

vetorCM[index]=DATA_CSG[0];
//vetorCM2[index]=DATA_CSG[3];
//vetorCM2[1]=DATA_CSG[3];
x1[2]=x1[1];
x1[1]=x1[0];
x1[0] = ( (float) (vetorCM[index] & 0x0FFF) ) - 2047.0)*g*0.4882812/Rm/1000;
//amperes

if (flag==0 & index==0) {i_reg=x1[0]; return;}; // guarda corrente de regime

// TETA = constante para conversao da tensao do extensometro em radianos
x5 = ( (float) (DATA_CSG[1] & 0x0FFF) ) - 2047.0)*g*TETA/2048.0;

x5=x5-media_sg; // media_sg contem o offset estimado do sinal do extensometro.
vetorSG[index]=DATA_CSG[1];

// sg

// inicio do controle tacometrico

/** // controle tacometrico proporcional:
if (index<aux) {
// aux_u[1]=dcvel-kt*vel2;
aux_u[0]=0;
}
else if (conta==0) {
//taco_adapt(aux); // adapta ganhos do controle
aux_u[0]=(kd*(dcvel-dcvel0)-kt*(vel1-dcvel0)); // u=i*Rm => i_aplicada=u/Rm=i
}

u=aux_u[0]*Rm; // u=i_desejada*Rm => i_aplicada=u/Rm=i_desejada
**/

// degrau (controle proporcional e realimentacao unitaria):
if (index==512.) vflag*=-1;
dcvel=300-vflag;
u=(dcvel-vel2*9.5493)*k1*Rm/1000; // /Kvi*Rm/1000; // Kvi nao e' necessario.

// degrau (realimentacao unitaria -> k1=1):
if (index==512.) vflag*=-1;
dcvel=(300-vflag)*2*3.1415/60.; // velocidade em rad/s
u=(k2*dcvel-k1*vel2)*4/20*Rm/1000;

// sweep de velocidade w0=300rpm:
// dcvel=A0+A*sat(index/n1/Max)*sat((Max-index)/n2/Max)*sin((wi+(double)index/Max*(wf-
wi))*2.*3.1415*index*T);
// dcvel=A*sat(index/n1/Max)*sat((Max-index)/n2/Max)*sin((wi+(double)index/Max*(wf-
wi))*2.*3.1415*index*T);
// ref_vel[index]=dcvel; // guarda velocidade de referencia
// u=(dcvel*k2-vel2*k1)*Rm;
// u=((dcvel-dcvel0)*k2-(vel1-dcvel0)*k1)*Rm; // VELOCIDADE DO ROTOR (em torno do regime)
// if (u>1./Rm) u=1./Rm; // limita corrente a 1A
// vetorU[index]=u;
/*
dcvel---| Kd=K=k2 |--->0-----| G |-----> vel2
          ^
          |___---| Kt=k1 |---
*/

// fim do controle tacometrico

// Sweep de corrente:
// u=A0+A*sat(index/n1/Max)*sat((Max-index)/n2/Max)*sin((wi+(double)index/Max*(wf-
wi))*2.*3.1415*index*T);
// u=A*sat(index/n1/Max)*sat((Max-index)/n2/Max)*sin((wi+(double)index/Max*(wf-
wi))*2.*3.1415*index*T);

// onda quadrada :

// if (index>512 & (floor((index-512)/512.)==(index-512)/512.)) uflag*=-1;
// else if (index==512) uflag*=-1;
// u=fabs(uflag)-uflag;

// ===
// senoidal:

```

```

// ====

//      u=A*(index*amostr+conta)/Max/amostr*sin((index*amostr+conta)*T*2*3.1416*freq); //
seno((tempo atual)*2*pi*freq))
//      u=A*sin((index*amostr+conta)*T*2*3.1416*freq); // seno((tempo atual)*2*pi*freq))

//===
// senoidal por partes:
//===
/**
//      if (index<=Max/3)
//      u=1.26+.15*sin((index*amostr+conta)*T*2*3.1416*freq); // seno((tempo
atual)*2*pi*freq))
//      if (index<Max/3)
//      u=1.47+.4*sin((index*amostr+conta)*T*2*3.1416*freq); // seno((tempo
atual)*2*pi*freq))
//      if (index>Max/3)
//      u=1.9+.4*sin((index*amostr+conta)*T*2*3.1416*freq); // seno((tempo atual)*2*pi*freq))
// fim if index
//      u=u*Rm; // u=i*Rm => i_aplicada=u/Rm=i !!

**/
//===
// fim senoidal por partes
//===
//      degrau de corrente:
//      u=fabs(ufag)-uflag; // A
//      if (index==64) { uflag*=-1; u=1; } // B -sobe degrau vencendo atrito estatico
//      if (index<=66 && index>64) u=1; // B - vence atrito estatico
//      if (index==1088) uflag*=-1; // B - desce degrau
//      if (index==64||index==1088) uflag*=-1; // A - sobe e desce degrau

// Gerador aleatorio filtrado
**/ if (conta==0) {
//      for (aux=4;aux>0;aux--) {
//      u_aux_i[aux]=u_aux_i[aux-1];
//      u_aux_o[aux]=u_aux_o[aux-1];
//      } // for

// Gera numero aleatorio entre -0.99995 e 0.99995 (distribuicao uniforme):
u_aux_i[0]=random(20000)/10000.-0.99995;

//      u_aux_o[0]=filt1_2wn(u_aux_i,u_aux_o); // f_corte=1/2*wn
//      u_aux_o[0]=filt3_4wn(u_aux_i,u_aux_o); // f_corte=3/4*wn
//      u_aux_o[0]=filt5_6wn(u_aux_i,u_aux_o); // f_corte=5/6*wn
//      u_aux_o[0]=filt_95wn(u_aux_i,u_aux_o); // f_corte=.95*wn
//      u_aux_o[0]=u_aux_i[0]; // sem filtragem
} // if /**/

/** // aleatorio por partes:
//      if (index<Max/2)
//      u=1.47+.4*u_aux_o[0]; //
//      if (index>Max/2)
//      u=1.57+.4*u_aux_o[0]; //
//      u=u*Rm;
//      // fim if index /**/

/**
//      Pacotes aleatorios com intervalos sem excitacao:

//      if (uflag==1) { // uflag=1 -> u=aleatorio; uflag=-1 -> u=0;
//      // Gera numero aleatorio entre -0.99995 e 0.99995 (distribuicao uniforme):
//      u_aux_o[0]=random(20000)/10000.-0.99995;
//      }
//      else if (uflag==-1) {
//      u_aux_o[0]=0.;
//      }

//      if ((index+1) % 512==0) uflag*=-1; // liga/desliga aleatorio a cada 512 pontos.

**/ // fim - pacotes aleatorios

//      u=1.95*Rm*u_aux_o[0]; // sinal de saida A/D
//      dcvel=dcvel0+10*u_aux_o[0]; // gera vel. aleatoria de ref.

//      u=A*u_ale_al[index];
//      u=A*u_ale_c2[index];

```

```

// fim gerador aleatorio */

//===
// Controle RST
//===
/**
// gera onda quadrada de referencia:
// if (index % 128==0) {
//     vflag*=-1;
//     dcvel=dcvel0+vflag;
// }

aux_u[1]=aux_u[0]; // guarda ultima saida

// adapta r,s e t:

/--
if ((index>aux) & (conta==0)) { // reduz a frequencia de atualizacao
// atualiza polinomios da planta estimada:
B_aux[0][0]=tetach[1][index];
A_aux[0][0]=1.;
A_aux[1][0]=tetach[0][index];
A_aux[2][0]=0.;
t[0]=7.7236e-2/tetach[1][index];
diofan(&r,&s,B_aux,nb,A_aux,na,Ac,nc,E,M,D);
}

// fim da adaptacao de r,s e t.

aux_u[0]=-r[1]*aux_u[1]+t[0]*(dcvel-dcvel0)+t[1]*(vref[index-1]-dcvel0)-s[0]*(vel1-dcvel0)-s[1]*(veloc1[index-1]-dcvel0);

u=aux_u[0]*Rm; // i_aplicada=u/Rm=aux_u[0]*Rm/Rm=aux_u[0] !!!
**/
//===
// FIM. controle RST
//===
...

// u=0.;

// Parada:
// if (index>7) u=-i_reg*Rm; // i_apl=u/Rm=-i_reg/Rm*Rm=-i_reg;

// Pulso de Corrente:
// if ((index>62) && (index<66)) u=10.*Rm; // i_apl=u/Rm=i_desej*Rm/Rm=i_desej

// if((index-63) % 128 ==0) u=-i_reg*Rm; // i_out=0
// if((index-64) % 128 ==0) u=-i_reg*Rm; //10.*Rm; // i_out=i_max
// if((index-65) % 128 ==0) u=-i_reg*Rm; //10.*Rm; // i_out=i_max
// if((index-66) % 128 ==0) u=-i_reg*Rm; //10.*Rm; // i_out=i_max
// if((index-67) % 128 ==0) u=-i_reg*Rm; //10.*Rm; // i_out=i_max

// if ((uflag==1) && (veloc1[index] > 40.)) uflag=0; // marca que ja passou da velocidade maxima
// if ((index>62) && (uflag==1)) u=(4.95-i_reg)*Rm; // i_apl=u/Rm=delta_i*Rm/Rm=delta_i

// FIM - PULSO DE CORRENTE

// dither=.1*Rm*(random(20000)/10000-.9995); // X*(randomico de -.9995 a .9995)

// if (index==8) aux_u[0]=-i_reg*Rm;
// u=aux_u[0];

u=-u+polsaida+dither;

// limita saida:
if (u<0) u=0;
if (u>10) u=10;

// Escreve no arquivo se u devia ser maior, menor ou igual
// ao que e' enviado para a saida.
/* if (u > refDA)
{

```

```

        u=refDA;
        vetorU[index]='*';
    }
    else
    if (u < 0)
    {
        u=0;
        vetorU[index]='-';
    }
    else vetorU[index]='+';*/

    DATA_SAIDA[0] = (int) (4095*u/refDA);
//    DATA_SAIDA[0]=(int)4095; // saida fixa em 10V
//    DATA_SAIDA[0] = (int) (2048*u/g+2047);
//    DATA_SAIDA[0]=DATA_CSG[1];
//    if ( (index % 2) == 0) DATA_SAIDA[0]=4095;
//    else DATA_SAIDA[0]=0;
//    DATA_SAIDA[0]=4095;

    if (flag)
    {
        pcl818HG(13,param);    /* D/A conv. with S/W data transfer */

        if (param[45] != 0)
        {
            printf("\n\nD/A convrsion failed! param[45] = %d ",param[45]);
            clrscr(); normvideo();    exit(1);
        }
    }
}

void grava(void)
{
    long int i;
    float temp;

/**/
    -// inicializa variaveis da media do extensometro:
    for (isg=1;isg<31;isg++) {
        ultimos_sg[isg]=0.0;
    } // for (isg...
    media_sg=0.0;
/**/

    if ((arq = fopen(dir_nome, "w")) == NULL)
    { clrscr(); normvideo(); printf("\nNao foi possivel gravar arquivo"); exit(0); }

    fprintf(arq,"%% Arquivo %s  \n\n",nome);

// jrbn  fprintf(arq," Frequencia de aquisicao da placa = 100 KHz \n");
    fprintf(arq,"NPT=%d ; %% Numero de pontos \n",index);

    fprintf(arq,"TA=%1.4e ;                %% segundos - periodo de
amostragem.\n",amostr*T);
    fprintf(arq,"TA_med=%1.4e ;                %% segundos - periodo de amostragem
medido.\n", (float) ((t_f-t_i)/CLK_TCK/(float)Max));
    fprintf(arq,"T=0:TA_med:NPT*TA_med-TA_med; %% vetor de tempo\n\n");
    fprintf(arq,"df=1/T(NPT);                %% intervalo de frequencia\n\n");
    fprintf(arq,"f=0:df:df*NPT-df;            %% vetor de frequencias\n\n");

//    fprintf(arq,"T_med=0:TA_med:NPT*TA_med-TA_med; %% outro vetor de tempo\n\n");

    fprintf(arq,"w_reg=%f; %% rad/s - velocidade de regime\n",dcvel0);
    fprintf(arq,"i_reg=%f; %% A - corrente de regime (motor de controle)\n\n",i_reg);

/**/
    fprintf(arq,"vel_eixo={\n");
    for (i=0; i<index; i++)
    {
        temp = veloc1[i]; // 9.5493 rpm;
        fprintf(arq,"%f;  ",temp);
        if ((i+1) % 5) == 0) fprintf(arq,"\n");
    }
    fprintf(arq,"]; % rad/s \n\n "); // % RPM\n\n");
/**/

```

---

```

/**
    fprintf(arq,"vel_motor={\n");
    for (i=0; i<index; i++)
    {
        temp = veloc2[i]; // rad/s
        fprintf(arq,"%1.5e; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; % rad/s \n\n "); // velocidade do motor
**/

/**
    fprintf(arq,"vel_estim={\n");
    for (i=0; i<index; i++)
    {
        temp = v_est[i]; // rad/s
        fprintf(arq,"%1.5e; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; % rad/s \n\n "); // velocidade estimada do motor
**/
**/
    fprintf(arq,"vel_ref={\n");
    for (i=0; i<index; i++)
    {
        temp = vref[i]; // rad/s
        fprintf(arq,"%f; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; % rad/s \n\n "); // */

/**
    fprintf(arq,"somador={\n");
    for (i=0; i<index; i++)
    {
        temp = ( ( (float) (vetorSOM[i] & 0xFFFF) ) - 2047.0)*g/2048.0;
        fprintf(arq,"%f; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; % Volts \n\n");

*/

    fprintf(arq,"cm={\n");
    for (i=0; i<index; i++)
    {
        temp = ( ( (float) (vetorCM[i] & 0xFFFF) ) - 2047.0)*g*0.4882812/Rm/1000;
//amperes
        fprintf(arq,"%f; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; % Amperes - corrente do motor de acionamento\n\n");
**/

/**
    fprintf(arq,"cm_pert={\n");
    for (i=0; i<index; i++)
    {
        temp = ( ( (float) (vetorCM2[i] & 0xFFFF) ) - 2047.0)*g*0.4882812/.23/1000;
//amperes - 1.25R -resistencia de leitura
        fprintf(arq,"%f; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; % Amperes - corrente do motor de perturbacao\n\n");

/**
    fprintf(arq,"bd={\n");
    for (i=0; i<index; i++)
    {
        temp = tetach[1][i];
        fprintf(arq,"%1.5e; ",temp);
        if ((i+1) % 5) == 0    fprintf(arq,"\n");
    }
    fprintf(arq,"]; %% numerador da planta \n\n ");

```

---

---

```

    fprintf(arq,"ad={\n");
    for (i=0; i<index; i++)
    {
        temp = tetach[0][i];
        fprintf(arq,"%1.5e; ",temp);
        if (( (i+1) % 5) == 0) fprintf(arq,"\n");
    }
    fprintf(arq,"]; %% polo da planta \n\n ");
/**/

/**/
    fprintf(arq,"sg={\n");
    for (i=0; i<index; i++)
    {
        temp=( ( (float) (vetorSG[i] & 0xFFFF) ) - 2047.0)*g*TETA/2048.0;

//        media_sg=10*media_sg; // da' peso maior para os ultimos 30 pontos.
        media_sg=0.0;
        for (isg=2;isg<31;isg++) {
            media_sg=media_sg+ultimos_sg[isg]; // faz media
            ultimos_sg[isg-1]=ultimos_sg[isg]; // atualiza vetor
        } // for (isg...
        media_sg=(media_sg+temp)/30; //40; // media atual=(3*media dos ultimos 30 +
media anterior)/4.
        ultimos_sg[30]=temp; // coloca ultimo valor de temp
        temp-=media_sg;

        fprintf(arq,"%1.5e; ",temp);
        if (( (i+1) % 5) == 0) fprintf(arq,"\n");
    }
    fprintf(arq,"]; % radianos\n\n");
/**/

/*    fprintf(arq,"satu={\n");
    for (i=0;i<index; i++)
    {
        if (vetorU[index]=='*') fprintf(arq,"1; ");
        else
            if (vetorU[index]=='-') fprintf(arq,"-1; ");
            else fprintf(arq,"0; ");
        if (( (i+1) % 10) == 0) fprintf(arq,"\n");
    }
    fprintf(arq,"];\n");
*/
    fclose(arq);
}

/*-----*/

void main()
{
    unsigned char ch;
    int i;

    Rm=G*Rm; // coloca em Rm a amplificacao do sinal de corrente

/**/
    // inicializa variaveis da media do extensometro:
    for (isg=1;isg<31;isg++) {
        ultimos_sg[isg]=0.0;
    } // for (isg...
    media_sg=0.0;
/**/

/**/    // Inicializa variaveis do controle RST:

    nc=3;
    Ac=mat_creat(nc+1,1,UNDEFINED);

    nb=0;
    B_aux=mat_creat(nb+1,1,UNDEFINED);

```

---

---

```

na=2;
A_aux=mat_creat(na+1,1,UNDEFINED);

    // polinomio do controlador (modelo de referencia)
Ac[0][0]=1.;
Ac[1][0]=-1.5709;
Ac[2][0]=6.4813e-1;
Ac[3][0]=0.; /**/

E=mat_creat(2*na,2*na,ZERO_MATRIX);
D=mat_creat(2*na,1,ZERO_MATRIX);
M=mat_creat(2*na,1,ZERO_MATRIX);

// fim da inicializacao RST

inicializa();

// mantem saida com offset (jrbn):
DATA_SAIDA[0] = (int) (4095*polsaida/refDA); // coloca offset na saida D/A
pcl818HG(13,param);                        /* D/A conv. with S/W data transfer */

do
{
    randomize(); // inicializa gerador aleatorio

    pergunta_constantes();

    dcvel0=dcvel; // guarda em dcvel0 o valor inicial de velocidade

    outportb(0x303,0x00);
    clrscr();
    gotoxy(22,16); printf("Tecle algo");

    mostra_v_c(); // mostra velocidade e corrente
    inic_estim(k1,k2,vel2); // inicializa estimacao de parametros

    clrscr();
    gotoxy(15,10); printf(" >> EIXO 2 <<");

    gotoxy(15,18); printf("Nome do arquivo: "); clrscr();
    scanf("%s",&nome);
    strcpy(dir_nome,dir);
    strcat(dir_nome,nome);

    index=0;
    conta=0;

    // toque(2000); // bipa espera (tempo) bipa espera .9 bipa e espera .9

    gotoxy(5,20); printf("Adquirindo e gravando..... Espere ou interrompa com enter para
controlar\n");

    //jrbn le^ uma vez a velocidade antes da aquisicao:
    le_velocidade();
    processa_v();

    // Guarda em w0 o valor da velocidade de regime do sistema nao controlado:
    w0=vel2*9.5493; // RPM.

/*
do
{
    le_velocidade();
    processa_v();

    veloc[index]=vel2; // guarda vel2 em veloc

    processa_e_controlo(0);

    //
    conta++;
    //
    if (conta/4.==floor(conta/4.))
    //
    { index++; conta=1; } // incrementa index de 4 em 4
    index++;

    outportb(0x303,ch);

```

---

```

        ch=!ch;
    } while ( (!kbhit()) && (index<Max) ) ;
*/

    while (kbhit()) ch=getch();

    gotoxy(5,22); printf("Controlando e gravando..... Espere ou interrompa com
enter\n");
    beep();

    //jrnb seta saida digital para medicao do tempo total:
    ch=0xff;
//jrnb outportb(0x303,ch); //jrnb

/** // Para inicialiazr com sinal randomico ja' em regime:
    for (i=0;i<=1000;i++) processa_e_controlo(1);
    beep();
**/

    t_i=clock(); // comeca a marcar o tempo total de medicao
    do
    {
        le_velocidade();
        processa_v();

        velocl[index]=vel1;          // guarda vel1 em velocl (eixo)
        veloc2[index]=vel2;          // guarda vel2 em veloc2 (motor)

        vref[index]=dcvel;           // guarda a velocidade de referencia em vref

        processa_e_controlo(1);

    /**/
    //      media_sg=10*media_sg; // da' peso maior para os ultimos 30 pontos.
    media_sg=0.0;
    for (isg=2;isg<31;isg++) {
        media_sg=media_sg+ultimos_sg[isg]; // faz media
        ultimos_sg[isg-1]=ultimos_sg[isg]; // atualiza vetor
    } // for (isg...
    media_sg=(media_sg+x5)/30; //40; // media atual=(3*media dos ultimos 30 +
ultima media)/4.
    ultimos_sg[30]=x5; // coloca ultimo valor de x5
    /**/

        // reduz taxa de amostragem:
        conta++;
        if ((conta/amostr)==floor(conta/amostr))
            { index++; conta=0; } // incrementa index

        // ERRADO (19-12-97):
    //      if (conta/amostr==floor(conta/amostr))
    //          { index++; conta=1; } // incrementa index

        outportb(0x303,ch);
        ch=!ch;
    } while ( (!kbhit()) && (index < Max) ) ;
    t_f=clock(); // guarda tempo final de medida.

    //jrnb reseta said digital para medicao do tempo total:
    ch=!ch; // jrnb
//jrnb outportb(0x303,ch); // jrnb

    // mantem saida com offset (jrnb):
    DATA_SAIDA[0] = (int) (4095*polsaida/refDA); // coloca offset na saida D/A
    pcl818HG(13,param);                          /* D/A conv. with S/W data
transfer */

    dcvel=dcvel0; // recoloca em dcvel valor inicial de velocidade.
    uflag=fabs(uflag); // reinicializa uflag
    vflag=fabs(vflag); // reinicializa vflag

    grava();
    beep();
} while (1);
clrscr(); normvideo();
}

```

## Apêndice C

### Lugar das raízes para controle tacométrico

Controladores tacométricos são utilizados para se obter a estabilização da velocidade de rotação de sistemas rotativos. Será mostrado abaixo um exemplo de projeto de controle tacométrico para uma planta linear de primeira ordem, indicando-se o procedimento de determinação da raízes do controlador. Abaixo está a planta que servirá como exemplo para o projeto de um controlador tacométrico através do método de posição da raiz (root locus):

$$G(z) = \frac{B(z)}{A(z)} = \frac{1}{z - 1,1} \quad (\text{C-1})$$

Esta planta é uma planta instável, pois o pólo se encontra fora do círculo unitário. Aplicando o controlador tacométrico descrito na Figura 2.1-1 à planta da Equação C-1, obtém-se para a planta controlada a função de transferência

$$G_c(z) = \frac{1}{z - 0,9 + K}$$

ou seja, é possível fazer com que a planta controlada tenha o seu pólo deslocado de acordo com o valor do ganho de realimentação  $K$ , o qual deve ser escolhido de forma a se obter uma planta estável. O procedimento de escolher o ganho  $K$  de forma a se obter pólos estáveis para a planta controlada chama-se lugar das raízes (“root locus”).

Abaixo encontra-se o programa utilizado para determinar o ganho tacométrico do controlador tacométrico para o sistema completo:

#### AN0705A1.M

```
%
% an0705a1.m
%
% Programa para projetar controlador tacométrico a partir
% de planta identificada por an0624d3.m.
%
% Utiliza dados \modelos\an0624d3.mat.
%
% programa criado a partir de \t0223\rootl.m
%
%
% datainicio : 05-07-98
% data      : 05-07-98
%
```

---

```

% DMC - FEM - UNICAMP
% José Brandão - brandao@webcom.com
%

load an0624d3
nd_ve_est=nd_est;
dd_ve_est=dd_est;

TA_med=2.93e-3; % segundos - taxa de amostragem

rlocus(nd_est,dd_est);
zgrid;

[ganhos,polos]=rlocfind(nd_est,dd_est);

kt=10; % ganho escolhido com rlocfind para fazer a
      % freq. intermediaria (70 Hz) aumentar

      % monta planta controlada:
[nc,dc]=feedback(nd_est,dd_est,kt,1,-1);
nc=nc(3:9); % tira zeros

save an0705a1 nc dc

break

load an0705b1

w=linspace(0,1/2/TA_med*2*pi,500);
[mag1,fas1,w1]=dbode(nd_est,dd_est,TA_med);
[mag2,fas2,w2]=dbode(nc,dc,TA_med);
[mag3,fas3,w3]=dbode(nc_est,dc_est,TA_med);

fig1=figure('Position',[120 120 560*0.54 420*0.59]);
set(gca,'DrawMode','fast');
hold on
plot(w1/2/pi,mag1/mag2(114),'k');
plot(w2/2/pi,mag2/mag2(114),'k');
plot(w3/2/pi,mag3/mag2,'k');
plot([88 88],[0 1],'k');
hold off
set(gca,'FontName','Arial','FontSize',8);
% zgrid
grid
whitebg(fig1);
% axis([0 w1(length(w1))/2/pi 0 .3])
ylabel('FRF (rad/s) / (rad/s) ');
xlabel('frequência');
title('Controle Tacométrico');
h=legend('não controlado','controlado-sim','controlado-RLS');
axes(h);

break

fig2=figure('Position',[120 120 560*0.54 420*0.59]);
set(gca,'DrawMode','fast');
hold on
plot(roots(dd_sg_est),'kx');
plot(roots(nd_sg_est),'ko');
hold off
set(gca,'FontName','Arial','FontSize',8);

```

---

---

```

zgrid
whitebg(fig2);
axis([-1 1 -1 1])
title('Estimação RLS');
h=legend('pólos','zeros');
axes(h);

break

k=17;
T=k*.0029; % taxa de amostragem do sistema
ad=-.9799;
bd=2.1962;

pol=[1 ad 1];

npt=50;
kt=linspace(0,.5,npt);
for q=1:npt
    raiz(1:2,q)=roots(pol.*[1 1 bd*kt(q)]);
end

% tabela de distância da raiz à origem em relação ao ganho kt
%[kt' abs(raiz')]

hold off
plot(kt,abs(raiz(1,:))','ob',kt,abs(raiz(2,:))','xb');
hold on
plot(kt,abs(raiz(1,:))','ob',kt,abs(raiz(2,:))','xb');
hold off
grid;
legend('raiz 1','raiz 2');
xlabel('Kt');
title('z^2+ad*z+bd*Kt; ad=-.9799; bd=2.1962.')

break

%[raizes_Kt,Kt]=rlocus(bd,[1 ad 0]);
%rlocus(bd,[1 ad 0]);
%zgrid;
%axis([-1 1 -1 1]);

% mostra resultado
%[Kt abs(raizes_Kt)]

pol=[1 ad 1];

npt=100;
kt=linspace(.4,.9,npt);
for q=1:npt
    raiz(1:2,q)=roots(pol.*[1 1 bd*kt(q)]);
end

% tabela de distância da raiz à origem em relação ao ganho kt
%[kt' abs(raiz')]

% fim - an0705a1.m

```

---

## Apêndice D

### Controle polinomial com modelo estimado

Para o caso com sistema completo, foi realizada a simulação do comportamento do controlador auto-sintonizável, e com isso permitir a avaliação do desempenho do sistema controlado antes de sua implementação. Abaixo estão listados o programa com a simulação da atuação do controle auto-sintonizável no sistema completo. Acompanham também duas funções complementares ao funcionamento do programa de simulação.

#### RST1030.M

```
%
%
% rst1030.m (a partir de bancrst.m)
%
%
% Faz controle por modelo de referência utilizando
% o método polinomial (R S T) descrito por Astrom
% (controle discreto, ed. de 1984, p.221).
%
% Utiliza os arquivos: - diofan2.m
%                   - tzesq.m
%
%
% DMC - FEM - UNICAMP - BRASIL
%
% José Brandão - brandao@webcom.com
%
% datainicio : 30-10-1998
% data       : 30-10-1998
%
clear all
format short e

% =====
% Descreve a planta
% =====

% carrega modelo identificado (RoTOR x Ip):
load an0928a3
nd_est=[2.5989e-002 -9.7080e-002 1.5653e-001 -1.3399e-001 5.9098e-002
-8.8980e-003 -1.2998e-003];
```

---

```
dd_est=[1.0000e+000 -4.7322e+000 9.7310e+000 -1.1001e+001 6.9799e+000
-2.0771e+000 -2.4621e-002 1.2458e-001 0];
```

```
TA=.0029;
```

```
i_reg=3.5; % A - corr. regime
```

```
w_reg=20; % rad/s - rot. rotor de regime
```

```
B=nd_est;
```

```
A=dd_est;
```

```
disp('sistema discreto:');
```

```
printsys(B,A,'z');
```

```
% =====
%
%   Determina Bmais e Bmenos
%
% =====
```

```
% determina pólos e zeros da planta
[z,p,k]=tf2zp(B,A);
```

```
% inicializa polinômios auxiliares
Bmenos=B;
Bmais=1;
```

```
% =====
%
%   Define o modelo
%
% =====
```

```
%---
%
% Definição do modelo Gm=Bm/Am:
%  $G_m(s) = \frac{w_m^2}{(s^2 + 2 \cdot \zeta \cdot w_m \cdot s + w_m^2)}$ 
%
%---
wm=60; % rad/s - freq. nat. do modelo
%zeta=0.41; % fator de amortecimento (igual ao da planta)
zeta=0.01; % fator de amortecimento
```

```
% função de transferência da planta contínua:
```

```
num_m=wm^2;
den_m=[1 2*zeta*wm wm^2];
```

```
%num_m=[1 2*zeta*wm .5*wm^2]
den_m=conv([1 10],den_m);
```

```
num_m=sum(den_m)/sum(num_m)*num_m;
```

```
% passa para espaço de estados (contínuo):
[am,bm,cm,dm]=tf2ss(num_m,den_m);
```

---

---

```

% discretiza no espaço de estados:
[fim,gamam]=c2d(am,bm,TA);

% retorna função de transferência discreta:
[Bmlinha,Am]=ss2tf(fim,gamam,cm,dm);

Bmlinha=sum(Am)/sum(Bmenos); % ganho estático unitário - rad/s/A
Am=[Am 0 0 0 0 0];

%break

% =====
%
%   Entra a dinâmica
%   do observador
%
% =====

OAc=2*length(A)-length(Am)-length(Bmais)-1;

%polos_Ao=input(['O observador deve ter ' num2str(OAc) ' pólo(s). Coloque-
o(s) em um vetor: ']);

%Ao=poly(polos_Ao);

Ao=[1 0 0 0 0 0 0 0 0];

% =====
%
%   Calcula R, S, e T
%
% =====

T=conv(Bmlinha,Ao);
T=tzesq(T);
Ac=conv(conv(Bmais,Ao),Am);
[R,S]=diofan2(B,A,Ac);
R=tzesq(R);
S=tzesq(S);

% numerador da planta controlada:
nc=conv(B,T);

AR=conv(A,R);

BS=conv(B,S);

% compatibiliza ordem das parcelas do denominador para que
% possam ser somadas:
BS=[zeros(1,length(AR)-length(BS)) BS];

% denominador da planta controlada
dc=AR+BS;

% =====
%
%   SIMULA
%
% =====
% Desenha resposta a degrau do sistema:

```

---

```

fig=figure;
whitebg(fig);
clf;
drawnow;

tempo=0:TA:5;
lt=length(tempo);
%u=sin(10*tempo);
u=ones(1,lt);
%u=tempo.*tempo/tempo(lt)^2;

% planta sem controle
subplot(411);
ysc=dlsim(B,A,u);
stairs(tempo,ysc);
set(gca,'FontName','Arial','FontSize',7)
grid
ylabel('s/ contr. (rad/s)');
xlabel('');
title('Resposta ao degrau para o Rotor');

% modelo incluindo zeros que não podem ser removidos:
subplot(412);
ym=dlsim(conv(Bmenos,Bmlinha),Am,u);
stairs(tempo,ym);
grid
set(gca,'FontName','Arial','FontSize',7)
ylabel('modelo (rad/s)');
xlabel('');

% planta controlada:
i_lim=2; % A - limite de corrente
yc=zeros(8,1);
i=zeros(8,1);
for k=9:length(tempo),
    i(k)=-R(2:9)*i(k-1:-1:k-8)+T*u(k:-1:k-8)-S*yc(k-1:-1:k-8);
    if abs(i(k))>i_lim,
        i(k)=i(k)/abs(i(k))*i_lim;
    end % if
    yc(k)=-A(2:9)*yc(k-1:-1:k-8)+B*i(k-2:-1:k-8);
end % for k

subplot(413);
stairs(tempo,yc);
grid
set(gca,'FontName','Arial','FontSize',7)
ylabel('c/ contr. (rad/s)');

subplot(414);
stairs(tempo,i);
grid
set(gca,'FontName','Arial','FontSize',7)
ylabel('corrente (A)');
xlabel('tempo (s)');

% fim rstl030.m

```

---

```

function [R,S]=diofan2(B,A,Ac)

%
% function [R,S]=diofan2(B,A,Ac)
%
%
%
% Calcula os coeficientes dos polinômios
% R e S de acordo com a equação diofantina:
%
%
%          AR + BS = Ac
%
%
% segundo as seguintes hipóteses:
% O(A) = 8
% O(B) = 6
% O(Ac) = 8+8 = 16 (Ac=Am.Ao)
%
% Ver o metodo de Ogata (controle discreto 2ed. p.519).
%
% DMC - FEM - UNICAMP - BRASIL
%
% José Brandão - brandao@webcom.com
%
% a partir de diofan.m
%
% datainicio : 22-agosto-1997
% data       : 30-julho-1998
%
%
% inicializa E:
E=zeros(17); % 17=(O(Ac)+1) elementos

% preenche E:
for col=1:9,
    E(col:col+8,col)=A';
end % for col

for col=10:17,
    E(col-6:col,col)=B';
end % for col

% Resolve sistema linear EM=Ac:
M=E\Ac';

% Retira S e R de M:
R=M(1:9)';
S=M(10:17)';

% fim diofan2.m

```

**TZESQ.M**

---

---

```
function poliout=tzesq(poliin)

%  retira zeros à esquerda em vetores de parâmetros de polinômios

% datainicio : 26-agosto-1997
% data       : 26-agosto-1997

% José Brandão - brandao@webcom.com

l=length(poliin);

poliout=[];

i=1;

% procura primeiro elemento diferente de zero
while (poliin(i)==0 & i<l-1),
    i=i+1;
end % while poli

% atualiza poliin
poliin=poliin(i:l);

% atualiza l
l=length(poliin);

% inicializa i
i=1;

% procura primeiro elemento não nulo (relativo)
while ((abs(poliin(i)/poliin(i+1))<=1e-9) & (i<l-1)),
    i=i+1;
end % while poli

% retorna polinômio sem zeros nas ordens superiores
poliout=poliin(i:l);

% fim tzesq
```

---

## Apêndice E

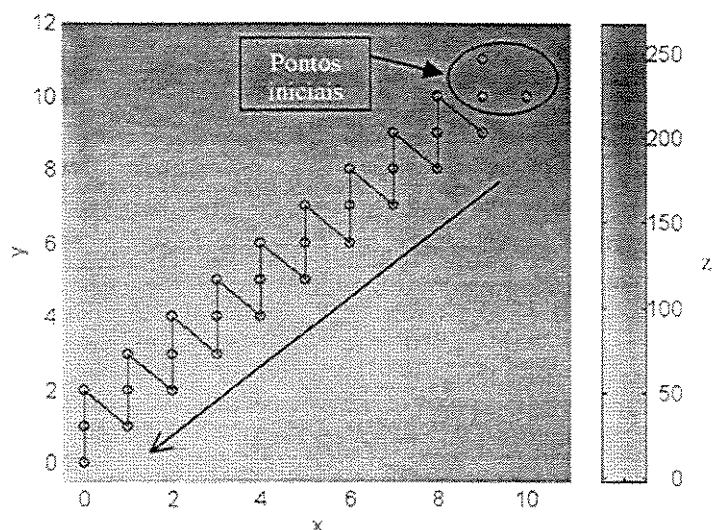
### Simplex Não Linear

O método Simplex é um algoritmo de minimização não-linear multidimensional. Ele permite determinar o mínimo de funções de múltiplas variáveis (PRESS, TEUKOLSKY et al, 1992, p.408). No pacote MATLAB® o método Simplex não linear está implementado através da função *fmins*.

Um *simplex* é uma figura geométrica que (em um espaço N-dimensional) possui N+1 vértices, sendo que os vértices são interconectados por segmentos de reta. Para duas dimensões, ele é um triângulo, para três, um tetraedro, e assim por diante. O funcionamento do método se dá através da construção de um novo simplex a partir do anterior, de forma que a cada iteração os vértices estejam mais próximos do mínimo desejado. O método Simplex foi escolhido por permitir minimizar uma função não linear sem que seja necessário conhecer todas as propriedades analíticas da função a ser minimizada.

Um algoritmo de minimização bi-dimensional (N=2 e o simplex é um triângulo) que utiliza somente uma reflexão para determinar o novo ponto da iteração é mostrado, assim como o resultado numérico obtido (Figura E-1) para a minimização da função  $z=x^2+y^2$ , no intervalo  $x \in [0,11]$  e  $y \in [0,12]$ .

---



**Figura E-1** – Resultado da minimização por simplex refletivo na função  $z=x^2+y^2$ .

O algoritmo é dado a seguir:

#### INICIALIZAÇÃO

- $N=2$ ; simplex tem três pontos;
- $x \in [0,1]$ ,  $y \in [0,12]$ ;
- $\delta=1$ ;
- Escolhe ponto inicial:  $p_1 = (x_1, y_1)$ ;
- $p_2 = p_1 + \delta(1,0) = (x_2, y_2)$ ;
- $p_3 = p_1 + \delta(0,1) = (x_3, y_3)$ .

#### LAÇO DE MINIMIZAÇÃO

- Determina  $i$  tal que  $z(p_i) > z(p_j)$ ,  $j \neq i$ ;
- Se  $i=1 \Rightarrow p_1 = (\min(|x_2|, |x_3|), \min(|y_2|, |y_3|))$ ;
- Se  $i=2 \Rightarrow p_2 = (2x_1 - x_2, 2y_1 - y_2)$ ;
- Se  $i=3 \Rightarrow p_3 = (2x_1 - x_3, 2y_1 - y_3)$ ;
- Repete laço de minimização até  $\frac{1}{3} \{\|p_1\| + \|p_2\| + \|p_3\|\} \leq 1$ .

**FIM**