

UNIVERSIDADE FEDERAL DE ITAJUBÁ

**PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA ELÉTRICA**



**PROJETO E IMPLEMENTAÇÃO DE UM REGULADOR DE
VELOCIDADE E CORRENTE BASEADO NA TÉCNICA DE
CONJUNTOS APROXIMADOS (ROUGH SETS)**

Ulisses Pessin Camatta

DISSERTAÇÃO DE MESTRADO

Itajubá, Março de 2009

2009	Ulisses Pessin Camatta	Dissertação de Mestrado
-------------	-------------------------------	--------------------------------

**PROJETO E IMPLEMENTAÇÃO DE UM REGULADOR DE VELOCIDADE E
CORRENTE BASEADO NA TÉCNICA DE CONJUNTOS APROXIMADOS
(ROUGH SETS)**

**Dissertação submetida ao
Programa de Pós-Graduação em
Engenharia Elétrica da UNIFEI
como requisito parcial para obtenção
do título de mestre em Engenharia Elétrica.**

Área de Concentração:
Automação e Sistemas Elétricos Industriais.

Orientador: Prof. Dr. Carlos Alberto Murari Pinheiro
Co-Orientador: Prof. Dr. Ângelo José Junqueira Rezek

Itajubá, Março de 2009

Dedicatória

*Em primeiro lugar dedico a Deus.
Em seguida, porém não muito distante, aos meus pais Aluisio e Maria Goreti e ao meu
irmão Hercules, pelo amor, apoio imensurável, carinho, amizade, incentivo, paciência e
por sempre acreditarem no meu trabalho.*

*“O Senhor é meu pastor e nada me faltará”
(Salmos 23,1)*

AGRADECIMENTOS

A Deus, engenheiro do universo, pela oportunidade de começar e terminar este trabalho;

A todos da minha enorme família: pais, irmão, tios, primos e avós, pela amizade, atenção, carinho e pelos momentos essenciais de descontração;

Aos amigos que venho cultivando desde o início da minha vida. Sejam eles amigos de infância, ensino médio (Larissa, Adson, Kleverson, Juliana, Amábia, Marlice, Luciano, Marília, Stéferson, Carolina, Caliandra e Renata), graduação (Ângelo, Eduardo de Oliveira, Eduardo Melo, Marcelo Barbalho, Fausto, Alan, Amaral, Kesley, Claudio, Maurício, Vinicius, Sávio e Wagner) ou os mais recentes feitos já residindo em Itajubá (Marcos Galvão, Alexandre, Daniel, Marco Túlio, Flávio, Luiz, Carlos Henrique, Eben, Robson, Solon, Eduardo, Gabriela e Brenda). Não esquecendo dos grandes amigos não citados mas que foram de fundamental importância para eu estar onde estou. Até mesmo aqueles que nem sabem o quanto me ajudaram. Todos estes contribuíram para que eu me tornasse o homem que hoje sou;

Aos professores e amigos Valberto, Cortez, Zambroni e Vitório pelo apoio e incentivo. Ao professor Luiz Eduardo um agradecimento particular pelo espaço cedido, laboratório, mesmo às vezes sem saber tê-lo feito;

Ao povo brasileiro e a CNPq pelo apoio financeiro;

Um agradecimento especial aos amigos e professores Pinheiro e Rezek pelo apoio, pela parceria, atenção, amizade, confiança e estímulo.

SUMÁRIO

SUMÁRIO	i
ÍNDICE DE FIGURAS	iii
ÍNDICE DE TABELAS	vi
LISTA DE ABREVIATURAS	vii
RESUMO	1
ABSTRACT	2
CAPÍTULO 1	3
Introdução.....	3
CAPÍTULO 2	5
Resenha Bibliográfica	5
CAPÍTULO 3	7
Desenvolvimento.....	7
3.1 Introdução	7
3.2 Bancada de Teste.....	7
3.2.1 Modelagem do Motor CC.....	8
3.2.2 Chopper	12
3.2.3 Transdutores e Sistema de Aquisição de Dados	14
3.3 Identificação de Sistemas	14
3.4 Controle Simétrico	19
3.5 Conjuntos Aproximados (Rough Set RS)	27
CAPÍTULO 4	44
Aplicações de Controladores Aproximados em Malhas de Controle de Máquinas Elétricas .	44
CAPÍTULO 5	52
Resultados	52

CAPÍTULO 6	60
Conclusão e Trabalhos Futuros	60
APÊNDICE A	62
APÊNDICE B	64
REFERÊNCIAS BIBLIOGRÁFICAS	84

ÍNDICE DE FIGURAS

CAPÍTULO 3	7
Figura 3.1 – Bancada de testes	8
Figura 3.2 – Diagrama de blocos da bancada de testes	8
Figura 3.3 – Características típicas de velocidade por torque (corrente)	9
Figura 3.4 – Modelo básico do motor CC	9
Figura 3.5 – Diagrama de blocos do modelo de um motor CC	11
Figura 3.6 – Chopper de acionamento do motor da bancada	13
Figura 3.7 – Operação PWM típica de um Chopper	13
Figura 3.8 – Operação do TCA 875	14
Figura 3.9 – Procedimento para identificação.....	15
Figura 3.10 – Comparação da corrente real e simulada	17
Figura 3.11 – Comparação da velocidade real e simulada	18
Figura 3.12 – Informação dos módulos de (3.24)	19
Figura 3.13 – Simetria no diagrama de módulo	20
Figura 3.14 – Diagrama de Bode de (3.29)	22
Figura 3.15 – Resposta ao degrau de (3.29)	23
Figura 3.16 – Diagrama em blocos das malhas de controle resultante.....	24
Figura 3.17 – Diagrama de Bode da malha de regulação de corrente	25
Figura 3.18 – Resposta sem e com a malha de regulação de corrente	25
Figura 3.19 – Diagrama de Bode da malha de regulação de velocidade.....	26
Figura 3.20 – Resposta típica do sistema	27
Figura 3.21 – Representação das regiões e aproximações em $S' = (U, A)$	31
Figura 3.22 – Representação das aproximações em $U = (R^+, S)$	34
Figura 3.23 – Arquivo texto do SI.....	36
Figura 3.24 – Software ROSETTA.....	37
Figura 3.25 – Selecionando o tipo de planilha de dados	37
Figura 3.26 – Abrindo um arquivo de dados.....	37
Figura 3.27 – Visualizando o SI.....	38
Figura 3.28 – Visualização dos dados	38
Figura 3.29 – Opções de discretização.....	39
Figura 3.30 – Tela de opção.....	39
Figura 3.31 – Opção de visualização dos dados discretizados	42

Figura 3.32 – Visualização dos dados discretizados	40
Figura 3.33 – Janela de obtenção de redutos.....	40
Figura 3.34 – Geração de regras.....	41
Figura 3.35 – Opção para visualização das regras	41
Figura 3.36 – Regras geradas pelo ROSETTA	42
Figura 3.37 – Valores da função original e do modelo baseado nas regras.....	42
CAPÍTULO 4.....	44
Figura 4.1 – Exemplo de malha de controle.....	45
Figura 4.2 – Comparação entre as saídas dos controladores (clássico e aproximado).....	46
Figura 4.3 – Dinâmica da malha sob ação dos dois tipos de controladores	46
Figura 4.4 – Dados do SI para o regulador de velocidade.....	47
Figura 4.5 – Comparação entre os dados do SI e os obtidos com as regras (velocidade).....	48
Figura 4.6 – Ampliação da Fig. 4.5.....	48
Figura 4.7 – Dados do SI para o regulador de corrente.....	49
Figura 4.8 – Comparação entre os dados do SI e os obtidos com as regras (corrente)	49
Figura 4.9 – Ampliação da Fig. 4.8.....	50
Figura 4.10 – Resposta dinâmica mediante variação de referência de velocidade.....	50
Figura 4.11 – Resposta dinâmica mediante variação de carga	51
CAPÍTULO 5.....	52
Figura 5.1 – Resposta na presença de distúrbios, meia carga e referência de 1 [pu]	53
Figura 5.2 – Resposta na presença de distúrbios, carga plena e referência de 1 [pu].....	53
Figura 5.3 – Resposta na presença de distúrbios, meia carga e referência de 0,5 [pu]	54
Figura 5.4 – Resposta na presença de distúrbios, carga plena e referência de 0,5 [pu].....	54
Figura 5.5 – Resposta na presença de distúrbios, carga plena e referência de 0,3 [pu].....	55
Figura 5.6 – Resposta na presença de distúrbios, meia carga e referência de 0,4 [pu]	55
Figura 5.7 – Resposta na presença de distúrbios, carga plena e referência de 0,4 [pu].....	56
Figura 5.8 – Resposta na presença de distúrbios, meia carga e referência de 0,7 [pu]	56
Figura 5.9 – Resposta na presença de distúrbios,carga plena e referência de 0,7 [pu].....	56
Figura 5.10 – Resposta na presença de distúrbios, meia carga e referência de 1 [pu]	57
Figura 5.11 – Resposta na presença de distúrbios, carga plena e referência de 1 [pu].....	57
Figura 5.12 – Respostas simuladas com partida a carga plena e referência de 1 [pu].....	58
Figura 5.13 – Respostas reais com partida a carga plena e referência de 1 [pu].....	58
Figura 5.14 – Respostas reais com partida a carga plena e referência de 0,8 [pu].....	59

APÊNDICE A	62
Figura A1 – Circuito PWM.....	62
APÊNDICE B	64
Figura B1 – Circuito para o controlador PI.....	64
Figura B2 – Circuito para o controlador aproximado	65

ÍNDICE DE TABELAS

CAPÍTULO 3	7
Tabela 3.1 – Dados de placa do motor CC utilizado na bancada.....	12
Tabela 3.2 – Parâmetros do PI variando a banda de simetria.....	22
Tabela 3.3 – Exemplo de um SI.....	28
Tabela 3.4 – Exemplo de um SI com atributos de condição e decisão.....	29
Tabela 3.5 – Conjuntos elementares obtido da Tabela 3.4 (B).....	32
Tabela 3.6 – Conjuntos elementares obtido da Tabela 3.4 (B').....	33

LISTA DE ABREVIATURAS

RS -	Rough Set
IGBT -	Insulated Gate Bipolar Transistor (Transistores Bipolares de porta isolada)
CC -	Corrente Contínua
PI -	Proporcional Integrativo
V_{cc} -	Tensão Contínua
V_a	Tensão na armadura
PWM -	Pulse Width Modulation (Modulação por largura de pulso)
I_a -	Corrente na armadura
ω -	Velocidade angular
V_{REF} -	Tensão de referência
R_a -	Resistência da armadura
L_a -	Indutância da armadura
V_{fem} -	Tensão contra-eletromotriz
T-	Torque
R_f -	Resistência do campo
L_f -	Indutância do campo
V_f -	Tensão no campo
I_f -	Corrente no campo
γ -	Diferença angular entre os campos de armadura e campo
ϕ -	Fluxo magnético de campo
K_t e K_t' -	Constante mecânica (proporcional ao torque)
K_a e K_a' -	Constante elétrica (proporcional à V_{fem})
M_B -	Conjugado de atrito
M_J -	Conjugado do momento de inércia
M_L -	Conjugado de carga
B -	Coefficiente de atrito
J -	Momento de inércia

V_o -	Tensão de saída do chopper
V_d -	Tensão de entrada do chopper
T_{on} -	Tempo de condução do chopper
T_{off} -	Tempo de bloqueio do chopper
T -	Período do PWM
TCA 875 -	Circuito integrado responsável pelo PWM
SKHI 22A -	Drive de disparo para o IGBT
PLC711B -	Placa de aquisição de dados da Advantech
ARX -	Auto Regressive with eXogeneous input (auto-regressivo com entradas externas)
u -	Entrada do sistema
y -	Saída do sistema
a_1, \dots, a_n e b_1, \dots, b_n -	Constante de saída e entrada respectivamente
n -	Ordem do sistema
φ -	Matriz com valores de entrada e saída
θ -	Matriz com valores das constantes de entrada e saída
\hat{y} -	Saída estimada
τ -	Transposta
N -	Número de amostras
E -	Matriz de erros
T_e -	Constante de tempo elétrica (função transferência)
T_m -	Constante de tempo mecânica (função transferência)
K_e -	Constante de ganho elétrica (função transferência)
K_m -	Constante de ganho mecânica (função transferência)
V_n -	Constante de ganho proporcional do PI
T_n -	Tempo de integração do PI
V_1 e V_2 -	Ganho das funções transferências F_1 e F_2 respectivamente
T_1 e T_2 -	Constante de tempo das funções transferências F_1 e F_2 respectivamente
W_1 e W_2 -	Frequência da função com maior e menor constante de tempo respectivamente
W_n -	Frequência relacionada a T_n

W_c -	Frequência de corte
σ -	Banda de simetria
F_m -	Função da malha interna
F_{ex} -	Função da malha externa
TCA -	Teoria dos Conjuntos Aproximados
SI -	Sistema de Informação
U, A, B, C, S, X, Q e V_a -	Conjuntos e subconjuntos
f_a -	Função de classificação
POS -	Região Positiva
NEG -	Região Negativa
DUV -	Região Duvidosa ou de Fronteira
E -	Conjunto Elementar
η -	Relação que indica a representação de um conjunto em outro
μ -	Função de precisão de aproximação
$Sup\{\}$ -	Indica o maior valor
$Inf\{\}$ -	Indica o menor valor

RESUMO

A necessidade de se controlar processos e sistemas em geral é um procedimento básico em diversas áreas de atividades, principalmente nas industriais. Vários campos do conhecimento humano estão envolvidos no desenvolvimento de métodos e técnicas que estão em constante evolução, sempre visando à obtenção de maior eficiência, robustez e eficácia.

Este trabalho de dissertação propõe a utilização de um método recente para malhas de controle aplicadas em acionamentos de máquinas elétricas. A idéia consiste na utilização de conceitos de conjuntos aproximados (Rough Sets) visando à obtenção de uma classe de controladores denominados aproximados.

Neste trabalho será abordado o controle de corrente e velocidade no acionamento de motores de corrente contínua como exemplificação da proposição em questão. Serão utilizadas topologias em cascata compostas por malhas reguladoras de corrente e velocidade.

Além de resultados de simulações numéricas, serão apresentados também resultados práticos obtidos por meio de uma bancada de ensaios montada especificamente para esta finalidade. Os resultados obtidos serão comparados com informações provenientes de malhas de controle que utilizam compensadores convencionais (proporcionais e integrais).

O cálculo dos ganhos dos controladores será realizado por técnicas conhecidas, tais como a otimização simétrica. Para os controladores aproximados serão utilizadas ferramentas computacionais específicas para processar conjuntos aproximados.

Técnicas de identificação de sistemas foram empregadas para a obtenção de modelos matemáticos do processo que constitui a bancada de ensaio utilizada nos ensaios práticos.

A bancada experimental foi montada utilizando-se tecnologias modernas, empregando transistores de potência IGBT, circuitos de disparo com circuitos integrados dedicados, sondas de efeito Hall para medições de corrente, sistemas de aquisição de dados computadorizados e softwares comerciais para simulações, coleta de dados e controle.

ABSTRACT

The need of controlling process and systems in general consists of a basic necessity in several areas of activity, usually industrial applications. A large number of human knowledge fields are involved on the development of methods and techniques which are in continuous evolution, always aiming the best effectiveness, robustness and efficiency.

This work proposes the utilization of a recent control method to application in the command of electrical machines. The idea consists in applying of concepts of Rough Sets to obtain a class of controllers called approximated.

In this work will be presented the current and speed control in command of direct current motors as exemplification of the method. Speed and current controlled loop cascade topology will be used.

Numerical simulated results will be shown, as well as practical results obtained using an experimental structure especially built for this purpose. The obtained results will be compared with information provided by control loops using conventional controllers (proportional and integral).

The definition of the controllers gain will be realized using well-known techniques, such as symmetrical optimum. For the approximated controllers will be used specific tools to compute rough sets.

Techniques of systems identification were used to get mathematical models of the process that composes the structure used in the tests.

The experimental test bench was built using modern technologies, employing power transistors IGBT, switching circuits with dedicated integrated circuits, Hall effect sensors for current measurements, computerized acquisition data systems and commercial software for simulation, data collection and control.

CAPÍTULO 1

INTRODUÇÃO

A teoria de sistemas de controle é uma área importante em aplicações de engenharia em geral. Esta área é responsável por projetos de controladores visando a melhoria do desempenho de um sistema real em conformidade a determinadas especificações. Os parâmetros dos controladores podem ser ajustados por meio de métodos empíricos e/ou analíticos. Nos métodos empíricos estes ajustes dependem de tentativas e erros ou da experiência do operador. Técnicas analíticas baseiam-se em valores quantitativos provenientes do modelo matemático do sistema real a ser controlado. Geralmente, se os dados dos modelos são adequados, os métodos quantitativos exigem um menor número de ajustes ou re-projetos dos parâmetros dos controladores, do que com técnicas empíricas.

Atualmente técnicas de simulação computacional são frequentemente utilizadas para auxiliar os projetos de controladores.

O controle de velocidade de máquinas elétricas rotativas é fundamental para vários processos industriais. Uma máquina elétrica tem sua velocidade afetada por variações de carga ou distúrbios externos. Para realizar compensações nestes contextos um sistema de controle usa um compensador, regulador, ou controlador de velocidade que tem como objetivo manter a velocidade do sistema regulada em determinados níveis de operação.

Em sistemas de acionamento, a velocidade de uma máquina está diretamente relacionada a determinadas grandezas. Para sistemas industriais como laminadores, bobinadoras, braços robóticos, e outros, a falta de regulação de velocidade pode resultar em produtos de má qualidade.

Muitos desses sistemas ou processos utilizam máquinas de corrente contínua, seja pela sua característica de partida ou pela menor complexidade de acionamento e controle quando comparado a motores de indução.

Esta dissertação utilizará um método novo para sistemas de controle que emprega conceitos provenientes da teoria de conjuntos aproximados (Rough Sets). Os resultados obtidos serão comparados com uma técnica conhecida como controle simétrico. O

desenvolvimento do trabalho constará de partes teóricas, simulações computacionais e implementações práticas.

Esta dissertação está dividida nos capítulos descritos a seguir. O **Capítulo 2** contém uma revisão bibliográfica, mostrando algumas referências que utilizam alguns conceitos empregados neste trabalho.

O **Capítulo 3** apresenta o desenvolvimento da proposta, contendo a construção de uma bancada experimental, a modelagem matemática do processo utilizado, a identificação de parâmetros do sistema, as técnicas de controle utilizadas e outros detalhes pertinentes ao trabalho. O **Capítulo 4** apresenta alguns conceitos básicos da teoria de conjuntos aproximados visando à aplicação em sistemas de controle. No **Capítulo 5** são mostrados os resultados obtidos. E no **Capítulo 6** estão as conclusões e sugestões para trabalhos futuros.

CAPÍTULO 2

RESENHA BIBLIOGRÁFICA

O controle de velocidade de máquinas consiste em uma área de estudos constantes devido aos campos de aplicações relacionados. O primeiro trabalho significativo de controle automático foi o regulador centrífugo construído por James Watt no século XVIII para o controle de velocidade de máquinas a vapor [12].

As máquinas elétricas rotativas constituem uma gama de dispositivos com muitas aplicações práticas. No ano de 1983, apenas nos EUA calculava-se em 24 milhões de motores elétricos usados em impressoras, toca-discos e toca-fitas que apresentavam algum tipo de controle de velocidade [6].

A máquina elétrica girante de menor complexidade, sob a ótica de sistemas de controle de velocidade é o motor de corrente contínua (CC). O motivo consiste em características inerentes às máquinas de corrente contínua, que apresentam altos torques de partida e ampla faixa de controle de velocidade. Em contrapartida este tipo de máquina apresenta uma construção mais complexa que outras máquinas elétricas, e por consequência custos maiores, manutenções mais constantes, peso mais elevado e suas velocidades máximas são limitadas pelas características de comutação de suas escovas.

Princípios de funcionamento, construção, topologias e técnicas de acionamento de máquinas de corrente contínua podem ser vistas em [11]. Uma topologia de acionamento bastante empregada em aplicações práticas é a de excitação independente [2], [3], [5], [8], [20]-[22]. Para esta configuração, existem duas variáveis capazes de alterar e controlar a velocidade dessas máquinas de maneira significativa, por meio da tensão de armadura e da corrente de excitação [21]. Em muitas aplicações essa corrente é mantida constante objetivando manter condições de torque, e o controle de velocidade é realizado variando-se a tensão de armadura [2], [3], [8], [20] e [22].

Um dos métodos eficientes para se alterar a tensão de armadura de uma máquina de corrente contínua é com a utilização de pontes retificadoras controladas, onde as tensões de saída das pontes estão diretamente relacionadas com o ângulo de disparo dos semicondutores das mesmas [21].

Outra opção é a utilização de choppers, conversores CC-CC com função redutora de tensão [3]. Várias configurações e detalhes destes conversores podem ser encontrados em [2]. Em geral os choppers podem apresentar melhores fatores de qualidade de energia [3]. Outra opção é a utilização de conversores conhecidos na literatura como PWM Rectifiers [5].

Um tipo de controlador bastante usado na indústria é o chamado Proporcional Integrativo (PI). Para aplicações em motores de corrente contínua normalmente usa-se um PI para a malha de velocidade e outro para a malha de corrente [21]. Técnicas clássicas para obtenção dos ganhos de um PI são encontradas em [12], onde os parâmetros são obtidos pelo método dos lugares das raízes ou por técnicas de resposta em frequência. Em [7] encontram-se os fundamentos da técnica de controle conhecida como método simétrico. Outras referências desta técnica para aplicações diversas podem ser encontradas em [4], [9] e [19].

Os métodos analíticos de projetos de controladores requerem o modelo do processo (ou planta) que se deseja controlar. Métodos de identificação de sistemas são utilizados para se obter modelos de sistemas reais. A referência [1] mostra os conceitos, técnicas e procedimentos fundamentais sobre identificação de sistemas.

Recentemente o trabalho [18] apresentou possibilidade de implementações de controladores baseados na teoria dos conjuntos aproximados (Rough Set) partindo-se de modelos baseados em regras.

Nesta dissertação serão usados conceitos similares visando à implementação de malhas de controles reais aplicadas no acionamento de máquinas elétricas.

Conceitos fundamentais e detalhes sobre conjuntos aproximados são encontrados em [10], [13]-[17] e [23]. Alguns conceitos básicos necessários para os desenvolvimentos realizados nesta dissertação serão apresentados oportunamente.

CAPÍTULO 3

DESENVOLVIMENTO

3.1 – Introdução

Neste capítulo serão descritos os desenvolvimentos realizados para testar e comparar as estratégias de controle utilizadas na proposta deste trabalho. Será detalhada a montagem de uma bancada de ensaio empregada nos testes e comparações, a modelagem e identificação do processo utilizado, a realização de malhas de controle projetadas por meio da técnica de controle simétrico, e a fundamentação necessária a elaboração de um sistema de controle que utiliza conceitos de conjuntos aproximados (Rough Sets).

3.2 – Bancada de Testes

A bancada de testes consta de um motor de corrente contínua (CC), um sistema de chaveamento (chopper) estático responsável pelo acionamento do motor, transdutores de velocidade e corrente, um sistema de aquisição de dados e um microcomputador. Programas aplicativos de aquisição de dados e controle foram desenvolvidos usando o ambiente do MATLAB/SIMULNIK. A Fig. 3.1 contém uma foto ilustrativa da bancada de testes e a Fig. 3.2 apresenta o diagrama de blocos da estrutura da bancada de testes.

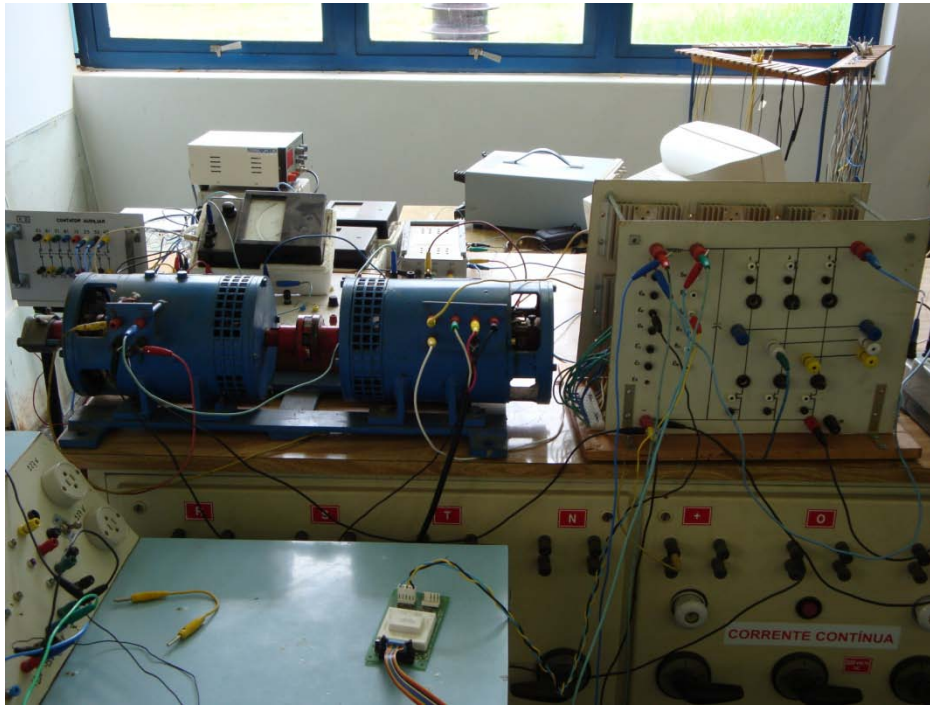


Figura 3.1 – Bancada de testes.

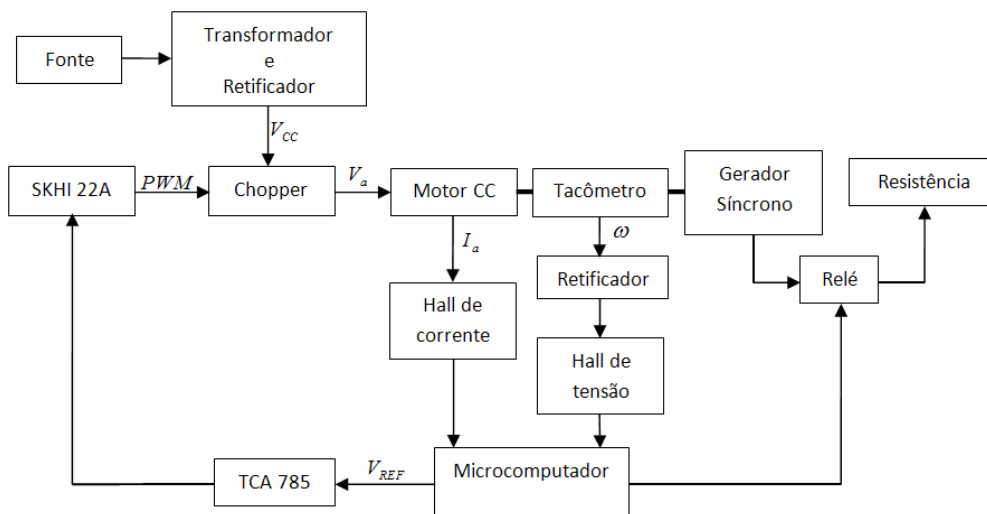


Figura 3.2 – Diagrama de blocos da bancada de testes.

3.2.1 – Modelagem do Motor CC

Máquinas de corrente contínua são empregadas freqüentemente em sistemas de tração de trens, em navios, em laminadores industriais e outros processos por apresentarem determinadas características de torques (Fig. 3.3), segundo suas estruturas

construtivas [11]. Para os motores de excitação independentes a curva de velocidade por torque é razoavelmente constante. Os motores CC podem utilizar sistemas de acionamentos e controles mais simples que motores de corrente alternada em função do contexto das suas aplicações. Entretanto, sistemas reais tendem a apresentar não-linearidades e variações de parâmetros que podem exigir estratégias de controle mais elaboradas.

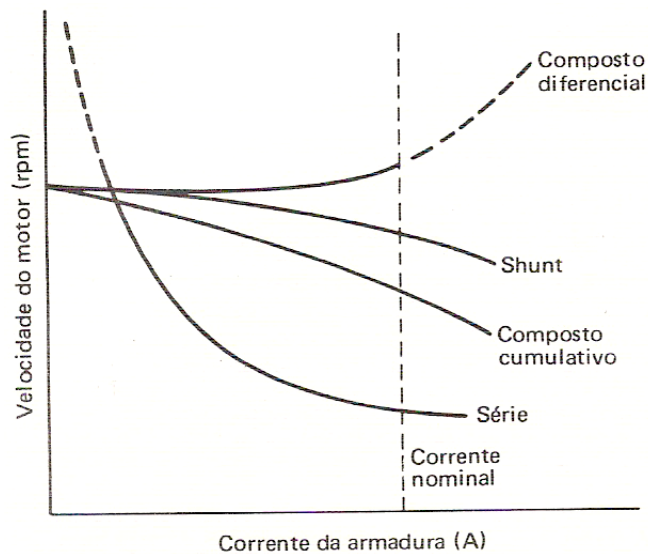


Figura 3.3 – Características típicas de velocidade por corrente (torque).

A Fig. 3.4 ilustra o modelo correspondente de um motor CC com excitação independente, com os circuitos do campo V_f e da armadura V_a .

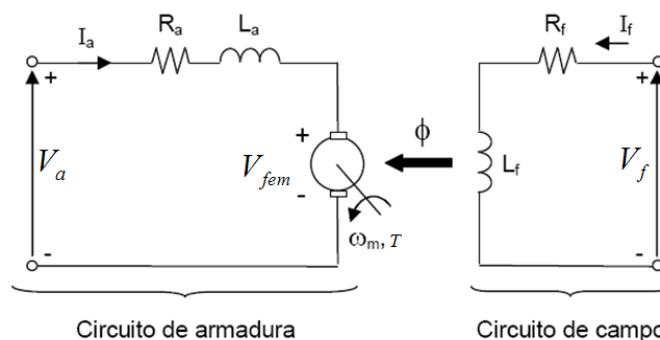


Figura 3.4 – Modelo básico de um motor CC.

A rotação de um motor CC deve-se à diferença angular γ entre o campo magnéticos da armadura e do campo gerados pelas correntes I_a e I_f , respectivamente. Um torque

T (3.1) resultante da força produzida pelo sistema, proporciona uma rotação com velocidade angular ω no sentido de zerar a diferença angular dos campos, alinhando os pólos das bobinas de armadura e de campo. A constante K_t é função das características do enrolamento da armadura.

$$T = K_t \phi I_a \quad (3.1)$$

A rotação resultante da bobina de armadura produz uma tensão induzida nas extremidades do enrolamento que se opõe a tensão aplicada, e é denominada força contra-eletromotriz induzida V_{fem} (3.2), onde K_a é uma constante que depende das propriedades magnéticas do núcleo do rotor e da geometria do enrolamento.

$$V_{fem} = K_a \phi \omega \quad (3.2)$$

Equacionando os circuitos de armadura e de campo têm-se (3.3) e (3.4).

$$V_a(t) = V_{fem}(t) + R_a I_a(t) + L_a \frac{dI_a(t)}{dt} \quad (3.3)$$

$$V_f(t) = R_f I_f(t) + L_f \frac{dI_f(t)}{dt} \quad (3.4)$$

A parte mecânica do modelo (3.5) é obtida com a somatória dos conjugados do sistema. Ou seja, o torque do motor T , o conjugado de atrito M_B que varia com a velocidade, o conjugado do momento de inércia M_J que é função da aceleração angular do sistema, e o conjugado de carga M_L aplicada no eixo do motor. B representa o coeficiente de atrito do sistema e J o momento de inércia do mesmo.

$$T(t) = M_L(t) + \underbrace{B\omega(t)}_{M_B} + \underbrace{J \frac{d\omega(t)}{dt}}_{M_J} \quad (3.5)$$

As equações de modelagem podem ser expressas no domínio da frequência. E considerando um sistema de excitação independente pode-se escrever (3.6), (3.7), (3.8) e (3.9).

$$T(s) = K_t \phi I_a(s) = K_t' I_a(s) \quad (3.6)$$

$$V_{fem} = K_a \phi \omega = K_a' \omega \quad (3.7)$$

$$T(s) = M_L(s) + B\omega(s) + Js\omega(s) \quad (3.8)$$

$$\frac{\omega(s)}{T(s) - M_L(s)} = \frac{1}{B + Js}$$

$$V_a(s) = V_{fem}(s) + R_a I_a(s) + L_a s I_a(s) \quad (3.9)$$

$$\frac{I_a(s)}{V_a(s) - V_{fem}(s)} = \frac{1}{R_a + L_a s}$$

Das expressões (3.6), (3.7), (3.8) e (3.9) obtém-se o diagrama de blocos do modelo resultante do motor, conforme ilustrado na Fig. 3.5.

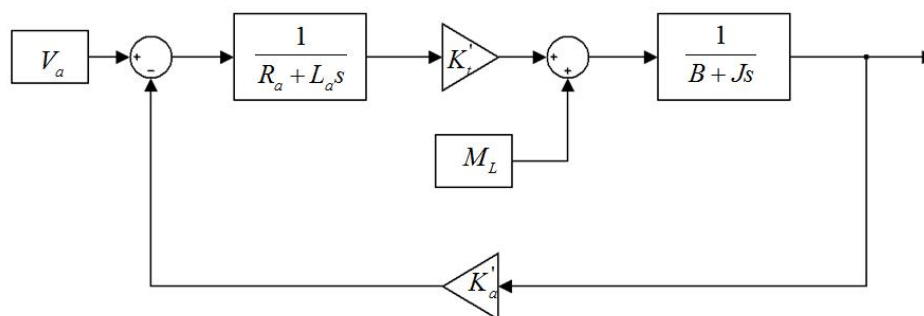


Figura 3.5 – Diagrama de blocos do modelo de um motor CC.

O motor usado na bancada de testes têm os dados de placa ilustrados na Tabela 3.1. Como carga mecânica empregou-se um gerador síncrono acoplado ao eixo do motor

CC. Resistências elétricas comutadas por relés, implementaram variações de carga no gerador e conseqüentemente no motor.

Tabela 3.1 – Dados de placa do motor CC utilizado na bancada.

Motor CC				
	Armadura		Campo	
Corrente	7,5	[A]	300	[mA]
Tensão	150	[Vdc]	150	[Vdc]
Velocidade	1800 [RPM]			
Potência	1 [CV]			

3.2.2 – Chopper

Os *choppers* ou conversores CC-CC são circuitos usados para conversão de uma tensão contínua constante em uma tensão contínua variável. Em geral é utilizado a variação do período de condução de um elemento semicondutor de potência, como transistores MOSFET, IGBTs, etc. O sinal de controle do semicondutor pode ser operado de duas maneiras, em frequência fixa ou variável. No método de frequência fixa o período é fixo e o tempo de condução varia. Essa técnica é conhecida como modulação em largura de pulso ou PWM (*Pulse Width Modulation*). Por meio da modulação da largura do pulso é possível controlar a tensão média de saída do conversor. A outra técnica tem o período do sinal de comando e do tempo de condução variáveis, e é conhecida como modulação de frequência.

Na bancada de teste o *chopper* (Fig. 3.6) é inserido em série como o motor, e sua finalidade é variar a tensão de armadura da máquina. É comum adicionar um indutor de alisamento em série com o circuito de armadura do motor objetivando filtrar a corrente de armadura resultante.

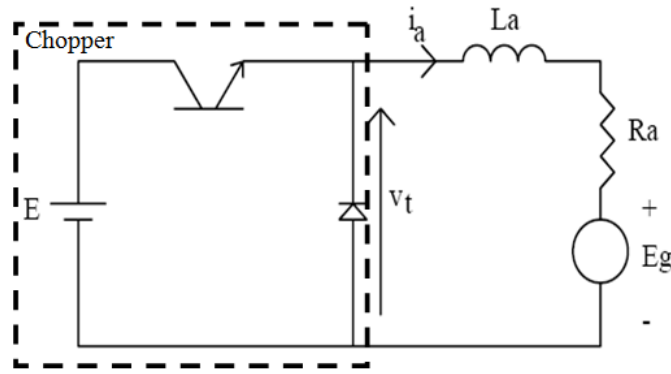


Figura 3.6 – Chopper de acionamento do motor da bancada.

O circuito do *chopper* apresenta dois estados básicos: semicondutor conduzindo ou bloqueado. Idealmente considera-se o semicondutor (transistor) uma chave ideal com dois estados básicos: ligada ou desligada. O diodo é usado para manter a condução de corrente no período que o semicondutor fica desligado. A tensão de saída do *chopper* (V_o) é resultado da média (3.10) da tensão de entrada ($E = V_d$) chaveada em um determinado período de tempo T_{on} . A Fig. 3.7 ilustra a forma de onda típica resultante.

$$V_o = \frac{1}{T} \int_0^{T_{on}} V_d(t) dt + \int_{T_{on}}^T 0 dt \quad (3.10)$$

$$V_o = \frac{T_{on}}{T} V_d$$

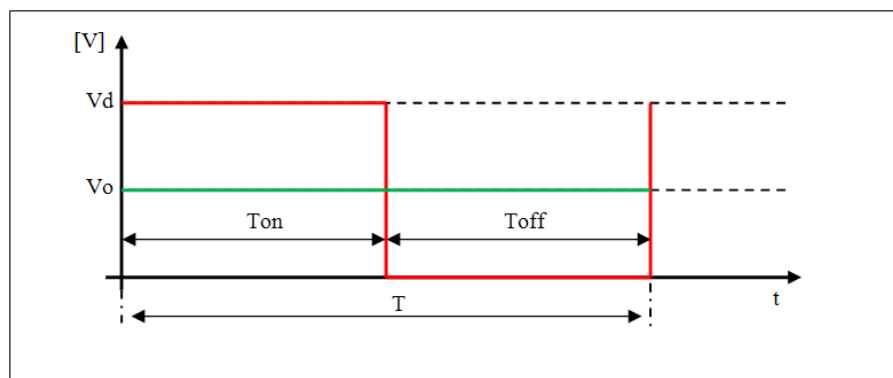


Figura 3.7 – Operação PWM típica de um Chopper.

Na montagem da bancada de teste foi utilizado um IGBT com um circuito *snubber*. Para a fonte CC foi utilizado um transformador, um retificador trifásico e um capacitor de filtro. Como circuito de disparo do IGBT utilizou-se circuitos específicos, um TCA 785 e um SKHI 22A (driver próprio para comandos de IGBTs). O sinal PWM é gerado

com a comparação de um sinal dente de serra com uma referência V_{REF} (Fig. 3.8). Essa lógica de comparação é realizada pelo circuito integrado TCA 785 (o período do PWM é o mesmo do sinal dente de serra, sendo definido por um sinal externo como um oscilador, por exemplo). O sinal de referência é fornecido por uma saída analógica do sistema de aquisição utilizado. No Apêndice A tem-se o circuito de disparo e o *driver* do *chopper* da bancada de ensaio.

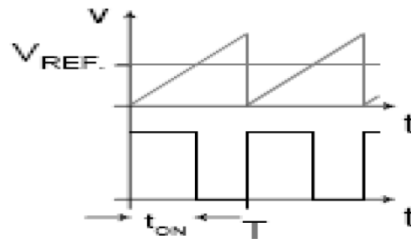


Figura 3.8 – Operação do TCA 785.

3.2.3 – Transdutores e Sistema de Aquisição de Dados

Para o sistema de aquisição de dados foi utilizada uma placa da Advantech modelo PLC711B. Um tacômetro foi usado como transdutor de velocidade e uma sonda HALL como transdutor de corrente da armadura do motor CC. A interface do computador com a placa de aquisição é realizada por meio do software SIMULINK/MATLAB.

3.3 – Identificação de sistema

Identificação de sistemas consiste na modelagem matemática a partir de dados experimentais de um processo ou sistema real. Os dados para modelagem devem ter as informações necessárias para caracterizar o processo. São usados valores de entrada específicos para excitar o sistema (pulso, degrau, etc.).

Para o processo de identificação é necessária a escolha da estrutura do modelo, por exemplo, um modelo auto-regressivo com entradas externas ARX (Auto Regressive with eXogeneous inputs). Este modelo é definido pela equação recursiva (3.11), onde y e u são informações da saída e da entrada do sistema respectivamente, $a_1 \dots a_n$ e $b_1 \dots b_n$ são os coeficientes do modelo e n define a ordem do mesmo.

$$y(t) + a_1y(t-1) + \dots + a_ny(t-n) = b_1u(t-1) + \dots + b_nu(t-n) \quad (3.11)$$

Os parâmetros do modelo podem ser ajustados para se adequar a exatidão da resposta real do sistema. A Fig. 3.9 mostra os procedimentos necessários para a identificação de um sistema em geral.

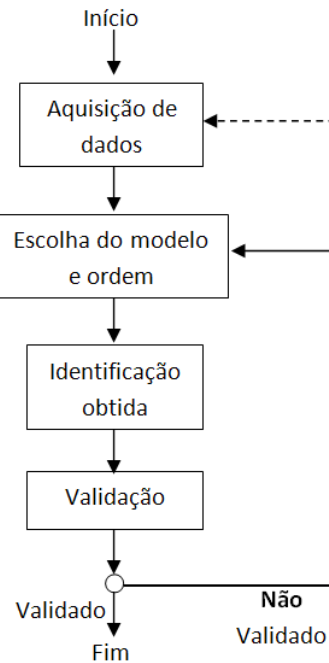


Figura 3.9 – Procedimento para identificação de sistemas.

Os coeficientes do modelo são estimados usando o método dos mínimos quadrados. A técnica de mínimos quadrados visa minimizar o erro entre o sinal real da saída com o sinal estimado do modelo. Considerando o modelo (3.11), tem-se (3.12) cuja representação na forma matricial é dada por (3.13).

$$y(t) = -a_1y(t-1) + \dots - a_ny(t-n) + b_1u(t-1) + \dots + b_nu(t-n) \quad (3.12)$$

$$y(t) = \varphi(t)\theta \quad (3.13)$$

Onde $\varphi(t)$ são informações de medidas da entrada e saída de um sistema θ são os coeficientes $(a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n)$ resultantes do modelo.

Assumindo (3.13) sendo a saída estimada \hat{y} e tendo a saída real y do sistema, tem-se o erro (3.14).

$$\varepsilon(t) = y(t) - \hat{y}(t) = y(t) - \varphi(t)\theta \quad (3.14)$$

Uma média quadrática dos erros é realizada (3.15), onde N é o número de amostras e o índice T indica matriz transposta.

$$J = \frac{1}{N} \sum_{t=1}^N (\varepsilon(t))^2 = \frac{1}{N} \sum_{t=1}^N (y(t) - \varphi(t)\theta)^2 = \frac{1}{N} E^T E, \quad (3.15)$$
$$E = \begin{bmatrix} \varepsilon(1) \\ \vdots \\ \varepsilon(N) \end{bmatrix}.$$

Desenvolvendo (3.15) tem-se (3.16).

$$J = \frac{1}{N} (y - \varphi\theta)^T (y - \varphi\theta) = [y^T y - \theta^T \varphi^T y - y^T \varphi\theta - \theta^T \varphi^T \varphi\theta] \quad (3.16)$$

Derivando (3.16) e igualando a zero para obter o mínimo da função de custo J , vem (3.17).

$$\frac{\partial J}{\partial \theta} = \frac{1}{N} [-2\varphi^T y + 2\varphi^T \varphi\theta] = 0 \quad (3.17)$$

Desenvolvendo (3.17) tem-se (3.18).

$$\varphi^T \varphi\theta = \varphi^T y \quad (3.18)$$

Isolando a variável de interesse θ de (3.18) tem-se (3.19).

$$\theta = [\varphi^T \varphi]^{-1} \varphi^T y \quad (3.19)$$

Conhecido a essência da identificação de sistemas e nomeando as expressões (3.8) e (3.9) como sendo a função transferência mecânica e elétrica do motor CC respectivamente, usou-se uma ferramenta (*toolbox*) do MATLAB para fazer a identificação do processo. O conjunto de dados para a identificação do motor CC foi obtido por meio de dois ensaios.

O ensaio com rotor travado foi realizado aplicando um degrau de tensão na armadura (sinal de entrada), tendo como sinal de saída a corrente de armadura. Na Fig. 3.10 tem-se a ilustração dos dados do ensaio e do modelo estimado. Processando os dados do ensaio por meio da ferramenta “ident”, obtém-se a função de transferência (3.20).

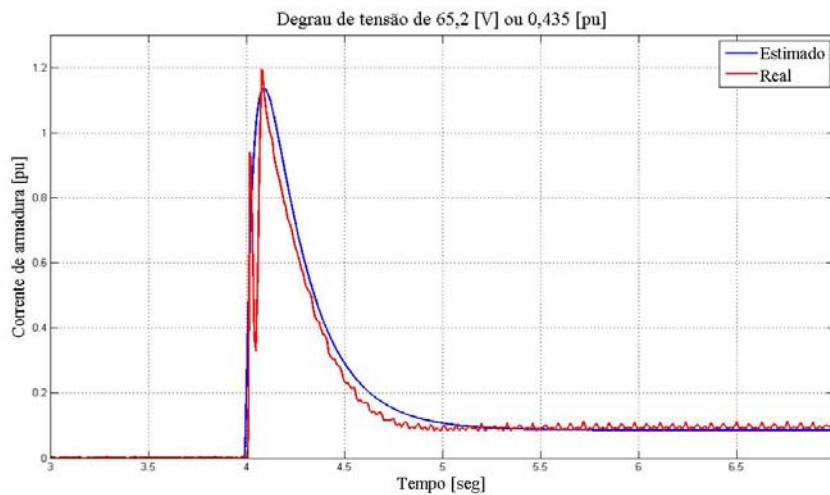


Figura 3.10 – Comparação da corrente real e da estimada.

$$\frac{I_a(s)}{V_a(s) - V_{fem}(s)} = \frac{K_e}{T_e s + 1} = \frac{3,62}{39,5 \cdot 10^{-3} s + 1} \quad (3.20)$$

A função transferência mecânica (3.21), por conseguinte, foi obtida aplicando o mesmo nível de tensão na armadura do ensaio anterior, agora, porém com o rotor livre. As informações são a corrente de armadura como entrada e a velocidade como saída. O modelo resultante é expresso por (3.21). A Fig. 3.11 mostra a comparação entre a velocidade estimada e a real.

$$\frac{\omega(s)}{T(s) - M_L(s)} = \frac{K_m}{T_m s + 1} = \frac{5,93}{4,3 s + 1} \quad (3.21)$$

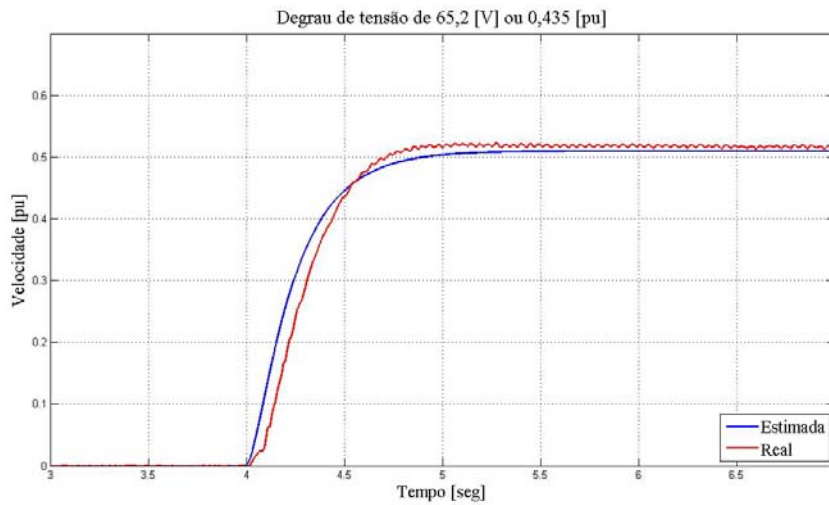


Figura 3.11 – Comparação da velocidade real e estimada.

Aproveitando os dados dos ensaios anteriores obteve-se o valor da resistência de armadura por meio da equação (3.3), considerando a tensão contra-eletromotriz nula (com o rotor travado) e corrente de armadura constante. Com o valor da resistência de armadura (3.22) calcula-se a constante K'_a por meio de (3.2) e (3.3), considerando o sistema em regime permanente e a corrente de armadura constante, tem-se (3.23).

$$R_a = 4,27[\Omega] \quad (3.22)$$

$$V_a(t) = K'_a \omega + R_a I_a(t) + L_a \frac{dI_a(t)}{dt} \quad (3.23)$$

$$K'_a = \frac{R_a I_a(t) - V_a(t)}{\omega}$$

$$K'_a = 0,8$$

Em relação ao modelo indicado na Fig. 3.5, as constantes de tempo T_e e T_m são expressas pelas relações L_a/R_a e J/B , respectivamente. Os ganhos K_e e K_m estão normalizados e o ganho K'_i é unitário, uma vez que 1 [pu] de corrente corresponde a 1 [pu] de torque.

Na identificação do modelo do sistema considerou-se a operação do mesmo em regiões lineares. Eventuais efeitos não-lineares devido à saturação do fluxo do motor ou do circuito de disparo do acionamento do sistema não foram considerados na modelagem resultante.

3.4 – Controladores Projetados pelo Método Simétrico

Um controlador visa modificar as características da resposta de um sistema controlado, podendo ajustar o valor do tempo de resposta desejado, o valor do sobre-sinal (*overshoot*) ou as duas características. Os controladores utilizados foram do tipo proporcional-integral (*PI*), um controlador para regulação da velocidade e outro para a corrente. Para sintonizar o valor do ganho proporcional (V_n) e do tempo de integração (T_n) de cada controlador, foi utilizado o método simétrico. Esta técnica emprega informações dos valores dos módulos das funções do controlador e da planta controlada. Para a definição dos parâmetros é utilizada uma função genérica (3.24) composta por duas funções F_1 , F_2 e a função do controlador *PI*. A Fig. 3.12 ilustra as respostas típicas das funções mencionadas.

$$F_G(s) = \underbrace{\frac{V_n(T_n s + 1)}{T_n s}}_{PI} + \underbrace{\frac{V_1}{T_1 s + 1}}_{F_1} + \underbrace{\frac{V_2}{T_2 s + 1}}_{F_2} \quad (3.24)$$

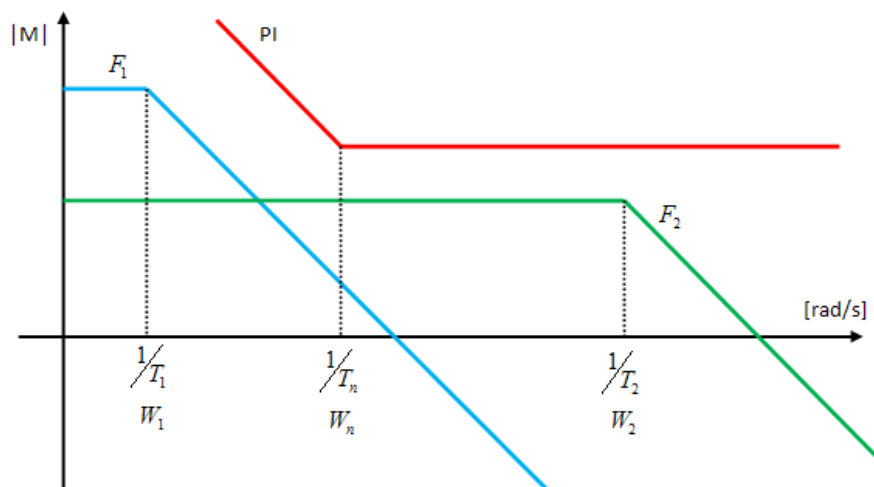


Figura 3.12 – Informações dos módulos de (3.24).

Os parâmetros do *PI* são definidos de maneira a impor uma simetria entre a maior frequência W_2 (função com menor constante de tempo T_2) e W_n , com referência a frequência de corte W_c . A Fig. 3.13 apresenta a simetria imposta pelo controlador *PI*, onde σ é a banda de simetria e a curva em vermelho é o gráfico do valor do módulo das funções do sistema controlado mais a do controlador.

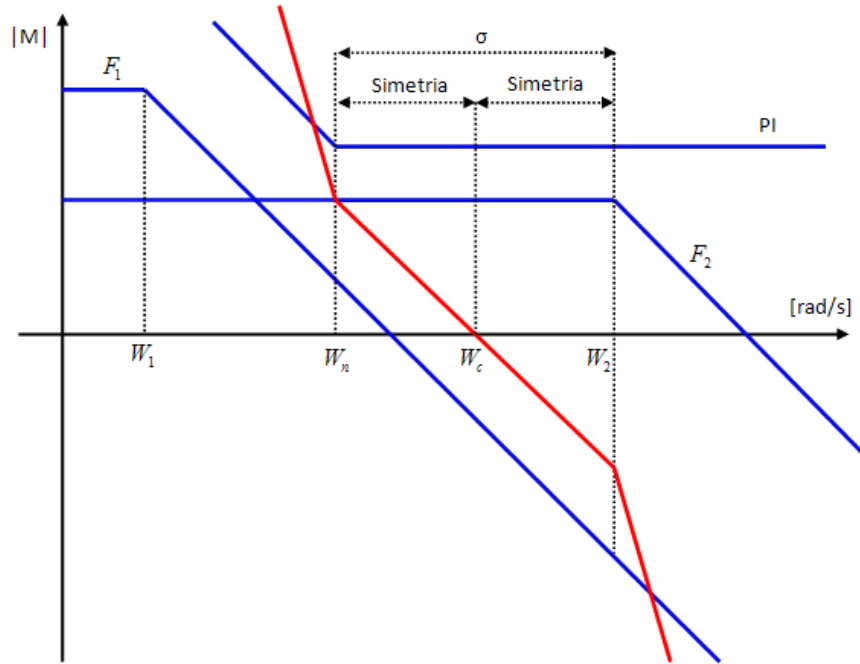


Figura 3.13 – Simetria no diagrama de módulo.

Com base na Fig. 3.13 obtém-se (3.25) que representa um ganho unitário para a frequência de corte, e (3.26) que mostra as relações entre as frequências envolvidas.

$$20\log(|PI|) + 20\log(|F_1|) + 20\log(|F_2|) = 20\log(1) \quad (3.25)$$

$$\log(V_n) + \log\left(\sqrt{\frac{V_1^2}{T_1^2 W_c^2 + 1}}\right) + \log(V_2) = \log(1)$$

$$\log(V_n) = \log(1) - [\log\left(\sqrt{\frac{V_1^2}{T_1^2 W_c^2 + 1}}\right) + \log(V_2)]$$

$$\log(V_n) = \log\left(\frac{1}{V_2 \sqrt{\frac{V_1^2}{T_1^2 W_c^2} + 1}}\right) \Rightarrow V_n = \frac{1}{V_2 \sqrt{\frac{V_1^2}{T_1^2 W_c^2} + 1}}$$

$$\log(\sigma) = \log(W_2) - \log(W_n) \Rightarrow \sigma = \frac{W_2}{W_n} \quad (3.26)$$

$$\log(W_c) = \log(W_2) - \frac{\log(\sigma)}{2} \Rightarrow W_c = \frac{W_2}{\sqrt{\sigma}}$$

$$\log(W_n) = \log(W_c) - \frac{\log(\sigma)}{2} \Rightarrow \log(W_n) = \log\left(\frac{W_2}{\sqrt{\sigma}}\right) - \frac{\log(\sigma)}{2} \Rightarrow W_n = \frac{W_2}{\sigma}$$

Substituindo W_c de (3.26) em (3.25) tem-se o ganho proporcional (3.27) do *PI*.

$$V_n = \frac{1}{V_2 \sqrt{\frac{V_1^2}{T_1^2 \frac{W_2^2}{\sigma}} + 1}} \quad (3.27)$$

O tempo de integração T_n é obtido com o valor de W_n de (3.26), resultando (3.28).

$$T_n = T_2 \sigma \quad (3.28)$$

Assumindo valores para (3.24) tem-se (3.29), e por meio de (3.27) e (3.28) são calculados os parâmetros do *PI* variando a banda de simetria ilustrada na Tabela 3.2.

$$F_G(s) = \underbrace{\frac{V_n(T_n s + 1)}{T_n s}}_{PI} + \underbrace{\frac{2}{10s + 1}}_{F_1} + \underbrace{\frac{3}{1s + 1}}_{F_2} \quad (3.29)$$

Tabela 3.2 – Parâmetros do PI variando a banda de simetria.

σ	V_n	T_n
1	1,68	1
3	0,98	3
4	0,85	4
5	0,76	5
7	0,65	7

Com o auxílio do MATALB pode ser obtido o diagrama de Bode de (3.29) e a resposta ao degrau em malha fechada para cada *PI* sintonizado. A Fig. 3.14 e a Fig. 3.15 ilustram as respostas obtidas.

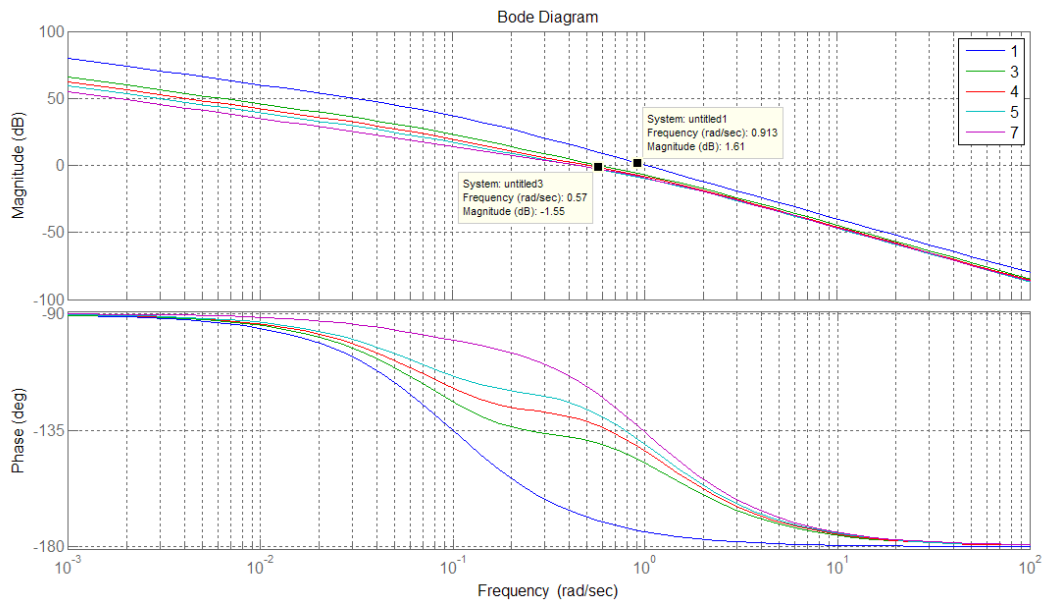


Figura 3.14 – Diagrama de Bode de (3.29).

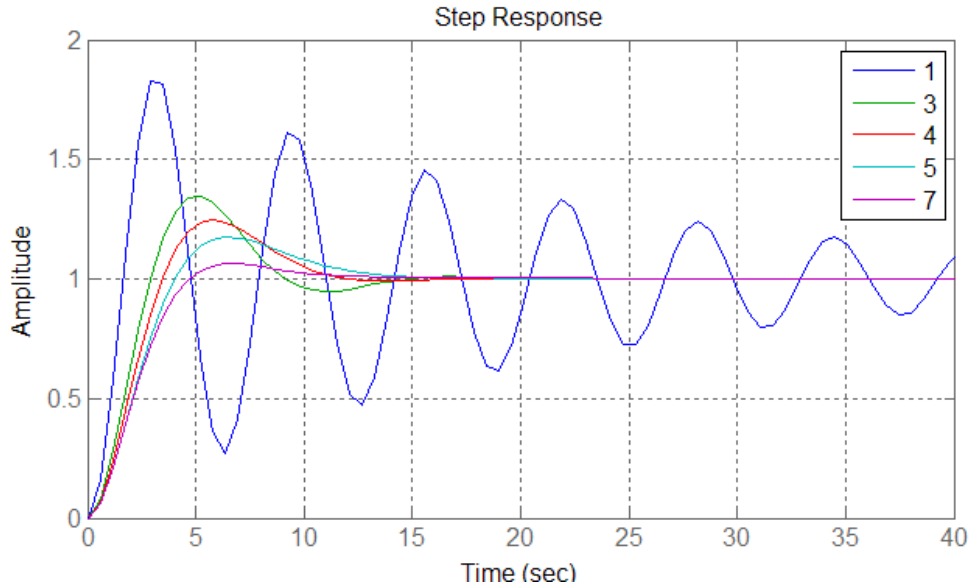


Figura 3.15 – Resposta ao degrau de (3.29).

A Fig. 3.15 ilustra as mudanças das características dinâmicas determinadas pelo método simétrico variando-se a banda de simetria. Esta técnica é aplicável em sistemas compostos por mais de uma função de primeira ordem, tendo uma função principal (a de maior constante de tempo), e outra também de primeiro grau representando todas as funções restantes da planta dada por uma função equivalente. Considerando um conjunto qualquer de funções de primeiro grau (3.30), considera-se que o tempo equivalente (3.31) aproxima-se da somatória de todas as constantes de tempo presentes.

$$F_{eq} = \frac{1}{T_I s + 1} + \frac{1}{T_{II} s + 1} = \frac{1}{T_I T_{II} s^2 + (T_I + T_{II})s + 1} \quad (3.30)$$

$$F_{eq} = \frac{1}{T_{eq} s + 1}; \quad T_{eq} = T_I + T_{II} \quad (3.31)$$

Estas funções com pequenas constantes de tempo podem representar os tempos de atrasos decorrentes de filtros de transdutores ou de atuadores, e da própria estrutura que compõe o sistema de aquisição de dados do sistema de controle. No processo utilizado, as constantes de tempo de maiores valores são as das funções de transferência referentes à parte elétrica e mecânica do sistema. A Fig. 3.16 ilustra a estrutura do sistema resultante, com uma malha de regulação de velocidade (em cor azul) e uma malha de

Para (3.34) e (3.35) obtém-se o diagrama de Bode ilustrado na Fig. 3.17. Respostas sem compensação e com a malha de regulação de corrente estão ilustradas na Fig. 3.18.

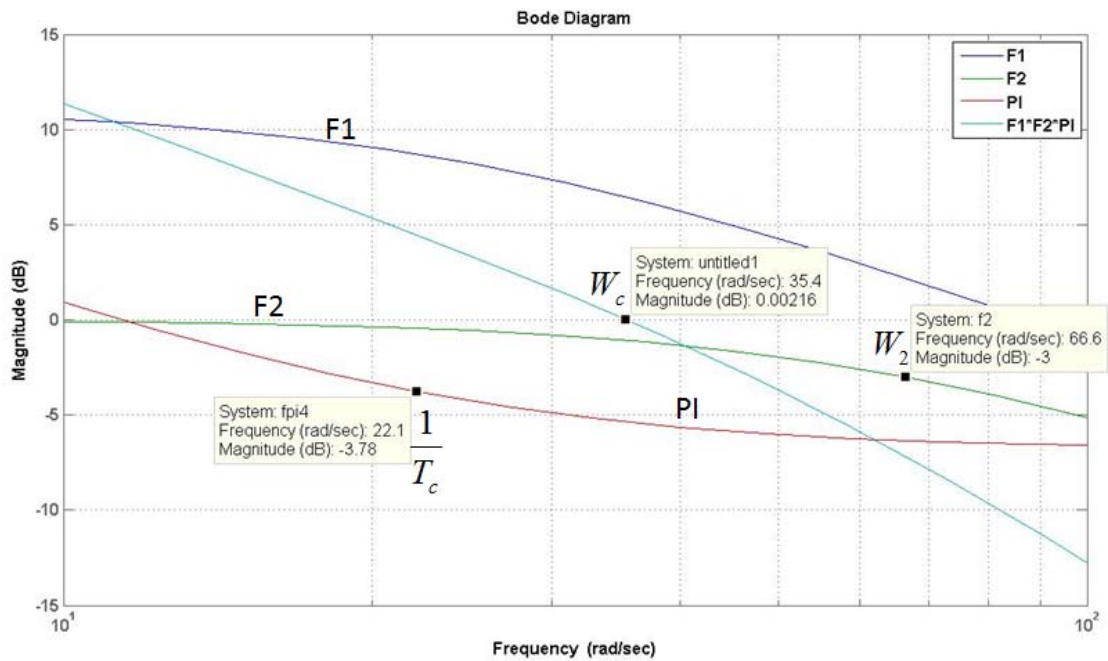


Figura 3.17 – Diagrama de Bode da malha de regulação de corrente.

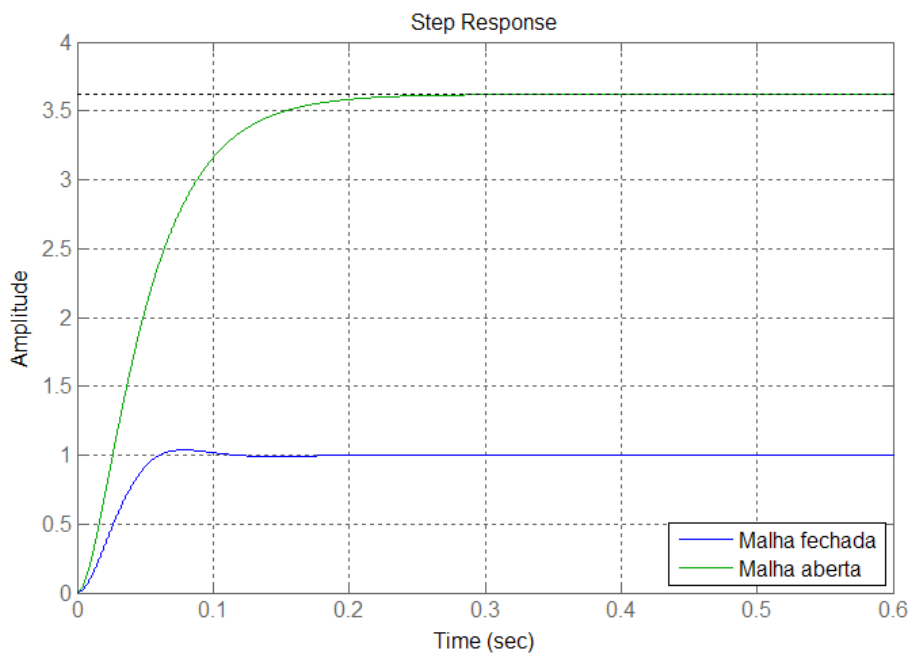


Figura 3.18 – Resposta sem e com a malha de regulação de corrente.

O procedimento para obter os parâmetros do *PI* da malha de regulação de velocidade é idêntico ao procedimento anterior. O tempo equivalente para esta malha é

a somatório de todas as constantes de tempo da malha interna, inclusive do PI de corrente, mais a constante de tempo de $FR(s)$ que é de $2/100$, resultando em (3.36). Os valores (3.37) dos parâmetros do PI de velocidade são calculados mediante (3.36), e o diagrama de Bode resultante está indicado na Fig. 3.19.

Resultados de simulações do sistema sem e com as compensações estão presentes na Fig. 3.20. Como característica da resposta de velocidade para uma entrada em degrau unitário, nota-se um sobre-sinal de aproximadamente 20%. Este valor pode ser atenuado usando-se um filtro de primeira ordem no sinal da referência de entrada.

$$F_{ex}(s) = \frac{V_v(T_v s + 1)}{T_v s} \cdot \frac{5,93}{(4,34s + 1)} \cdot \frac{1}{(0,1195s + 1)} \quad (3.36)$$

$$V_c = 3,067; \quad T_c = 0,478 \quad (3.37)$$

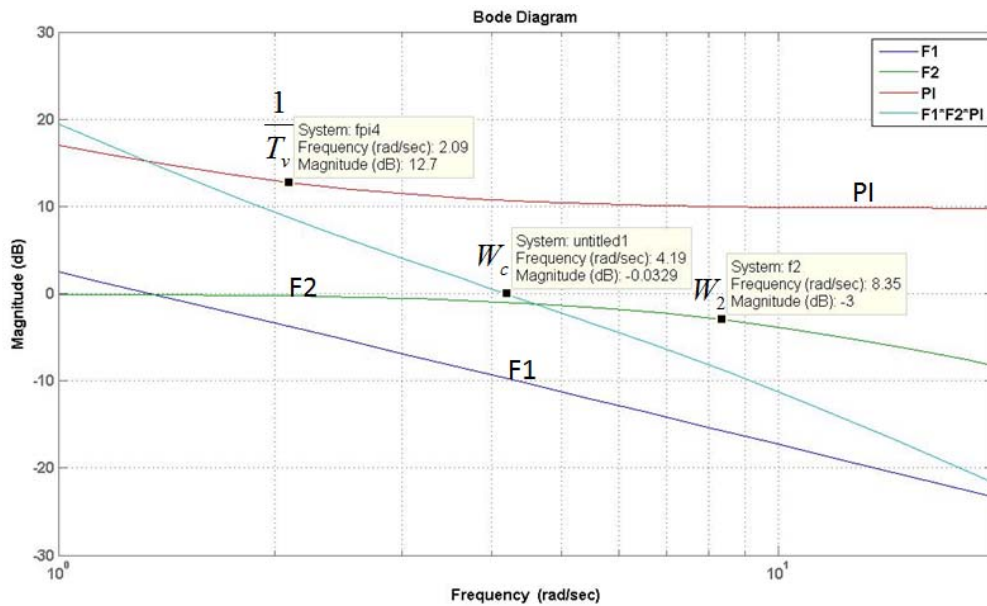


Figura 3.19 – Diagrama de Bode da malha de regulação de velocidade.

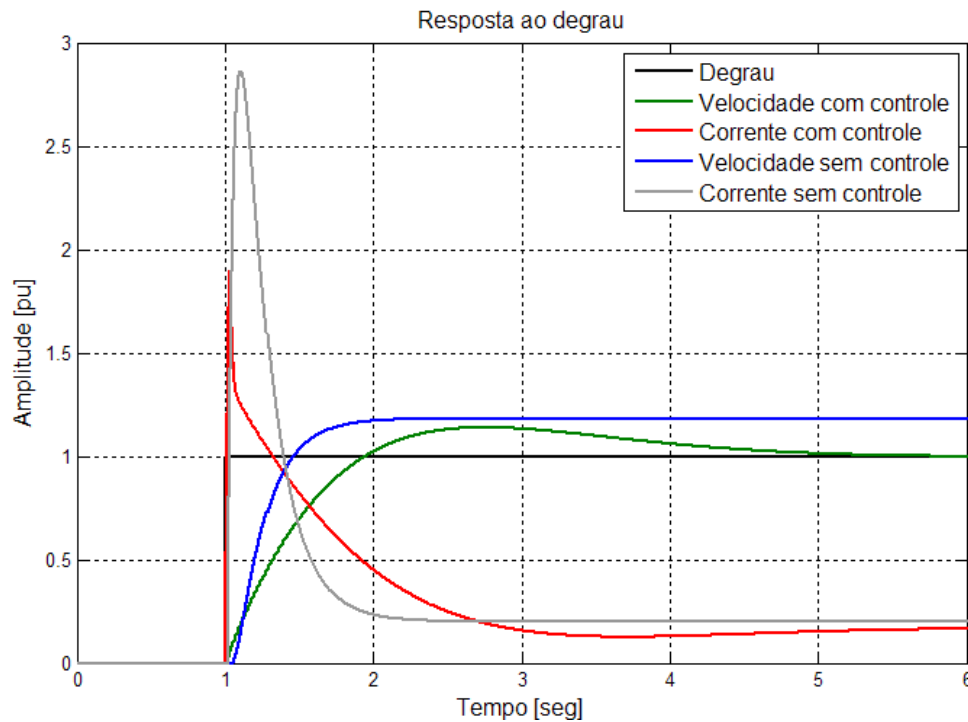


Figura 3.20 – Respostas típicas do sistema.

3.5 – Conjuntos Aproximados (Rough Set RS)

A Teoria dos Conjuntos Aproximados (TCA) é uma abordagem matemática que visa à manipulação de incertezas e imprecisões em conjuntos de dados. Sua principal vantagem é não necessitar de informações adicionais ou preliminares, como distribuição de probabilidade, intervalos de confiança, valores de possibilidades, etc.

Conjuntos de dados na forma de tabelas podem ser oriundos de diversas áreas, por exemplo: informações de dados de venda de um determinado produto; medidas de um sistema físico; conhecimentos de um especialista; etc. Uma tabela de dados genérica é denominada de Sistema de Informação (SI). Um SI é composto por atributos de condição e decisão.

A partir de um SI a teoria em questão propõe extrair informações ou conhecimentos específicos a aplicação almejada. Para tanto é utilizado o conceito de indiscernibilidade dos elementos contido em um SI. Por meio desta relação são realizadas reduções de redundâncias ou inconsistências nos dados de um SI.

O formalismo matemático desta teoria baseia-se na teoria dos conjuntos. Alguns exemplos serão dados ao decorrer da conceituação. A seguir será descrito brevemente a base dos conceitos da TCA [10], [13] e [17].

Um SI é um par ordenado $S = (U, A)$ onde U é um conjunto finito e não vazio de objetos ou exemplos chamado universo e A é um conjunto finito e não vazio de informações disponíveis chamados atributos. Para cada atributo $a \in A$ existe um valor correspondente V_a determinado por uma função de classificação $f_a : U \rightarrow V_a$.

Tabela 3.3 – Exemplo de um SI.

	a_1	a_2
x_1	16-30	50
x_2	16-30	0
x_3	31-45	1-25
x_4	31-45	1-25
x_5	46-60	26-49
x_6	16-30	26-49
x_7	46-60	26-49

Para a Tabela 3.3 temos os seguintes conjuntos definidos

$$U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\},$$

$$A = \{a_1, a_2\},$$

$$V_{a_1} = \{16, \dots, 60\},$$

$$V_{a_2} = \{0, \dots, 50\},$$

$$f_{a_1}(x_1) = [16 - 30], f_{a_1}(x_4) = [31 - 45], f_{a_2}(x_2) = 0, \text{ ou}$$

$$f(x_1, a_1) = [16 - 30], f(x_4, a_1) = [31 - 45], f(x_2, a_2) = 0.$$

Nesta mesma tabela nota-se uma semelhança entre alguns valores, por exemplo, entre $f_{a_1}(x_1), f_{a_1}(x_2)$ e $f_{a_1}(x_6)$, entre $f_{a_2}(x_3)$ e $f_{a_2}(x_4)$ e entre outros valores, que são definidos como indiscerníveis. Considerando o conjunto A , definem-se como indiscerníveis os objetos x_3 e x_4 , e também x_5 e x_7 .

Considerando a_1 e a_2 como atributos de condição na Tabela 3.3 e adicionando um atributo de decisão a_3 , tem-se a Tabela 3.4. Neste exemplo, os valores do atributo são informações simbólicas: “Sim” e “Não”. Qualquer símbolo alfa-numérico pode ser usado nos atributos de condição.

Tabela 3.4 – Exemplo de um SI com atributos de condição e decisão.

	a_1	a_2	a_3
x_1	16-30	50	Sim
x_2	16-30	0	Não
x_3	31-45	1-25	Não
x_4	31-45	1-25	Sim
x_5	46-60	26-49	Não
x_6	16-30	26-49	Sim
x_7	46-60	26-49	Não

Nota-se que os objetos x_3 e x_4 ditos como indiscerníveis para $A = \{a_1, a_2\}$ agora são discernível devido à adição da informação de decisão, mas os objetos x_5 e x_7 permanecem indiscerníveis. Um SI como apresentado na Tabela 3.4 pode ter redundância de informação, mesmas condições definindo a mesma decisão, como mostrado para os objetos x_3 e x_7 , esses são representados por apenas uma regra (3.38).

$$SE (a_1 = [46 - 60]) \ E (a_2 = [26 - 49]) \ ENTÃO (a_3 = Não) \quad (3.38)$$

Para um SI que apresenta apenas objetos concisos, um conjunto de regras semelhante a (3.38) seria suficiente para representá-lo. Caso contrário, usa-se a teoria dos conjuntos aproximados, onde a decisão será dada pela relação de aproximação entre duas decisões. Esta condição é representada por outra regra adicional (3.39).

$$SE (a_1 = [31 - 45]) \ E (a_2 = [1 - 25]) \ ENTÃO (a_3 = Não \ OU \ Sim) \quad (3.39)$$

Uma relação $R \subseteq X \times X$ que é reflexiva (xRx), simétrica (se xRy então yRx) e transitiva (se xRy e yRq então xRq) é chamada de relação equivalente. A classe equivalente de um elemento $x \in X$ contendo todos os objetos $y \in X$ tal que xRy , denomina-se conjunto elementar. De fato, se x relaciona-se com y (xRy), eles são indiscerníveis. O formalismo matemático resultante pode ser representado por (3.40), assumindo $S = (U, A)$ e $B \subset A$.

$$I_A(B) = \{(x, x') \in U^2 \mid \forall a \in B \ a(x) = a(x')\} \quad (3.40)$$

Para a Tabela 3.4 definem-se algumas condições de indiscernibilidade (3.41).

$$\begin{aligned}
 I(\{a_1\}) &= \{\{x_1, x_2, x_6\}, \{x_3, x_4\}, \{x_5, x_7\}\} \\
 I(\{a_2\}) &= \{\{x_1\}, \{x_2\}, \{x_3, x_4\}, \{x_5, x_6, x_7\}\} \\
 I(\{a_1, a_2\}) &= \{\{x_1\}, \{x_2\}, \{x_3, x_4\}, \{x_5, x_7\}, \{x_6\}\}
 \end{aligned} \tag{3.41}$$

Considerando um subconjunto formado por um valor indiscernível no atributo de decisão, pode-se determinar o quanto esse subconjunto é representável usando uma relação entre os números de elementos deste subconjunto e um novo SI é formado pelos conjuntos elementares resultantes.

Assumindo $S' = (U, A)$ sendo um conjunto elementar de S , onde $X \subseteq U$ e $B \subseteq A$, define-se o conjunto aproximado inferior de X em B , $B_*(X)$ que é o conjunto dos elementos de B contidos totalmente em X . O conjunto aproximado superior de X em B , $B^*(X)$ é o conjunto de elementos de B que possuem algum elemento de X , também conhecido como região positiva.

$$B_*(X) = \{x \in X : B(x) \subseteq X\} = POS_B(X) \tag{3.42}$$

$$B^*(X) = \{x \in X : B(x) \cap X \neq \emptyset\} \tag{3.43}$$

Uma região negativa é a união de todos os elementos do universo que não tem conexão com os elementos de X . Uma região de fronteira ou duvidosa é o conjunto dos elementos do universo que não pertencem nem a área positiva e nem negativa (3.45).

$$NEG_B(X) = U - B^*(X) \tag{3.44}$$

$$DUV_B(X) = B^*(X) - B_*(X) \tag{3.45}$$

As representações dessas regiões e do conjunto aproximado superior e inferior são representadas na Fig. 3.21.

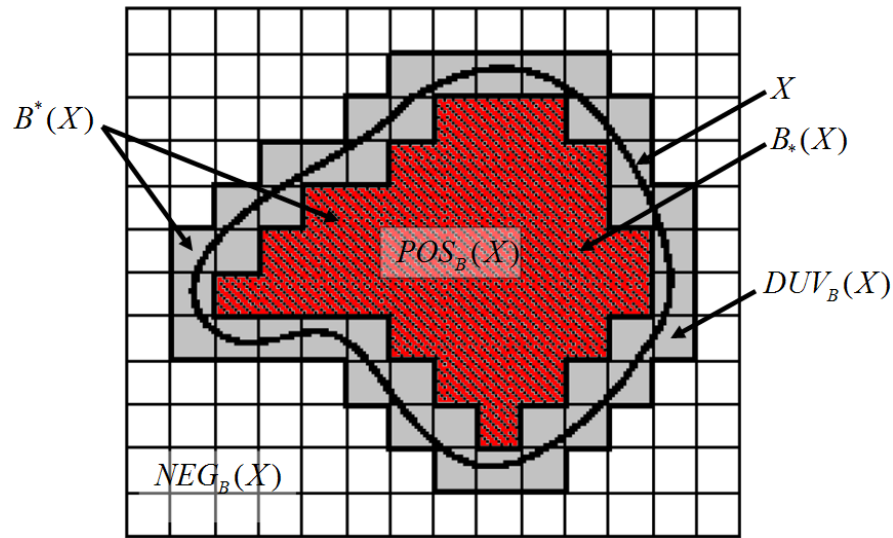


Figura 3.21 – Representação das regiões e aproximações em $S' = (U, A)$.

A relação (3.46) é usada para dizer o quanto o conjunto X pode estar representado em um conjunto $B \subseteq A$, onde o entre barras “ $| \cdot |$ ” indica o número de elementos (átomos) que participam do conjunto. Esta relação é chamada de precisão de aproximação. Se esta relação for unitária indica que o conjunto B define fielmente X de modo conciso. Se for nula B não define X , e para valores entre “0” e “1” indica o grau de representação de X em B .

$$\eta_B(X) = \frac{|B_*(X)|}{|B^*(X)|} \quad (3.46)$$

Uma relação (3.47) importante para TCA é a que indica o grau da confiabilidade de X representado por um conjunto elementar E . Esta relação é denominada função de pertinência aproximada (*rough membership function*).

$$\mu_E(X) = \frac{|E \cap X|}{|E|} \quad (3.47)$$

Para o SI exemplificado anteriormente, o conceito de conjunto elementar é aplicado aos atributos de condições $B = \{a_1, a_2\}$ da Tabela 3.4 resultando na Tabela 3.5.

Tabela 3.5 – Conjuntos elementares obtido da Tabela 3.4 (B).

	a_1	a_2	a_3
$E_1 = \{x_1\}$	16-30	50	Sim
$E_2 = \{x_2\}$	16-30	0	Não
$E_3 = \{x_3, x_4\}$	31-45	1-25	Não/Sim
$E_4 = \{x_5, x_7\}$	46-60	26-49	Não
$E_5 = \{x_6\}$	16-30	26-49	Sim

Admitindo que X é o conjunto formado pelos objetos com valor “Sim”, $X = \{\{x_1\}, \{x_4\}, \{x_6\}\}$, tem-se:

$$POS_B(X) = B_* = \{\{x_1\}, \{x_6\}\}$$

$$B^* = \{\{x_1\}, \{x_3, x_4\}, \{x_6\}\}$$

$$NEG_B(X) = \{\{x_2\}, \{x_5, x_7\}\}$$

$$DUV_B(X) = \{\{x_3, x_4\}\}$$

$$\eta_B(X) = \frac{|B_*(X)|}{|B^*(X)|} = \frac{2}{4} = 0,5$$

$$\mu_{E_1}(X) = \frac{|E_1 \cap X|}{|E_1|} = \frac{|\{x_1\} \cap \{\{x_1\}, \{x_4\}, \{x_6\}\}|}{|\{x_1\}|} = \frac{1}{1} = 1$$

$$\mu_{E_3}(X) = \frac{|E_3 \cap X|}{|E_3|} = \frac{|\{x_3, x_4\} \cap \{\{x_1\}, \{x_4\}, \{x_6\}\}|}{|\{x_3, x_4\}|} = \frac{1}{2} = 0,5$$

$$\mu_{E_5}(X) = \frac{|E_5 \cap X|}{|E_5|} = \frac{|\{x_6\} \cap \{\{x_1\}, \{x_4\}, \{x_6\}\}|}{|\{x_6\}|} = \frac{1}{1} = 1$$

A relação $\eta_B(X)$ indica que B não é suficiente para representar X concisamente. As relações $\mu_{E_1}(X)$ e $\mu_{E_5}(X)$ indicam as condições dos conjuntos elementares E_1 e

E_5 , respectivamente, representam o conjunto X , ou seja, para estas condições tem-se a certeza que a decisão é “sim”. A razão de valor 1/2 representa uma indecisão entre “Sim” e “Não”.

Para efeitos comparativos, adota-se que $B' = \{a_2\}$ (exclui-se o atributo a_1), e seguindo o mesmo procedimento tem-se a Tabela 3.6.

Tabela 3.6 – Conjuntos elementares obtido da Tabela 3.4 (B').

	a_2	a_3
$E_1 = \{x_1\}$	50	Sim
$E_2 = \{x_2\}$	0	Não
$E_3 = \{x_3, x_4\}$	1-25	Não/Sim
$E_4 = \{x_5, x_7, x_6\}$	26-49	Não/Não/Sim

$$POS_{B'}(X) = B'_* = \{\{x_1\}\}$$

$$B'^* = \{\{x_1\}, \{x_3, x_4\}, \{x_5, x_7, x_6\}\}$$

$$NEG_{B'}(X) = \{\{x_2\}\}$$

$$DUV_{B'}(X) = \{\{x_3, x_4\}, \{x_5, x_7, x_6\}\}$$

$$\eta_{B'}(X) = \frac{|B'_*(X)|}{|B'^*(X)|} = \frac{1}{6} = 0,167$$

$$\mu_{E_1}(X) = \frac{|E_1 \cap X|}{|E_1|} = \frac{|\{x_1\} \cap \{\{x_1\}, \{x_4\}, \{x_6\}\}|}{|\{x_1\}|} = \frac{1}{1} = 1$$

$$\mu_{E_3}(X) = \frac{|E_3 \cap X|}{|E_3|} = \frac{|\{x_3, x_4\} \cap \{\{x_1\}, \{x_4\}, \{x_6\}\}|}{|\{x_3, x_4\}|} = \frac{1}{2} = 0,5$$

$$\mu_{E_4}(X) = \frac{|E_4 \cap X|}{|E_4|} = \frac{|\{x_5, x_7, x_6\} \cap \{\{x_1\}, \{x_4\}, \{x_6\}\}|}{|\{x_5, x_7, x_6\}|} = \frac{1}{3} = 0,333$$

Comparando os resultados da TCA para duas tabelas baseadas em B e B' , observa-se a contribuição do atributo a_1 para a decisão. A relação $\eta_{B'}(X)$ diminuiu drasticamente, e ainda, o elemento x_6 passa a participar da classificação. Antes este elemento apresentava uma certeza na representação de X . O baixo valor da relação $\mu_{E_4}(X)$ indica que o conjunto elementar E_4 representa fracamente o conjunto X , se esse valor se aproxima de zero é possível considerar que tenha ocorrido um erro na construção do SI. Para o caso de uma mesma representação de conjuntos e valores das relações mostrados, assumindo B e B' , afirmar-se-ia que o atributo a_1 é dispensável.

Usando a TCA, como visto nas definições até o momento, podemos reduzir ou desconsiderar algum(s) atributo(s) dependendo da sua importância na tomada de decisão ou na precisão desejada. Inconsistências em um SI podem ser identificadas por meio das relações mostradas.

Descrito os conceitos da TCA para valores simbólicos ou inteiros, será mostrada uma extensão para valores reais [14]-[16] e [23].

Assumindo o espaço real aproximado $U = (R^+, S)$, onde R^+ é um conjunto de valores reais positivos e $S \subseteq R^+$ a seqüência x_1, x_2, \dots, x_i tal que $x_1 < x_2 < \dots < x_i$, interpreta-se este espaço como pontos de uma semi-reta. Todos os valores das amostras induzem uma partição $\pi(S) = \{\{0\}, (0, x_1), \{x_1\}, (x_1, x_2), \{x_2\}, \dots, \{x_n\}, (x_n, x_{n+1})\}$, onde (x_n, x_{n+1}) é um intervalo aberto. Seja $S(x)$ uma partição que contém x , para $x \in S$ vem $S(x) = \{x\}$. Por outro lado para x entre dois valores de S , ou seja, $x \in (x_n, x_{n+1})$, usam-se os conceitos de aproximação superior e inferior, onde x_n e x_{n+1} são, respectivamente, aproximação inferior $S_*(x)$ e superior $S^*(x)$, cujo conceito está ilustrado na Fig. 3.22, assumindo um intervalo fechado $Q(x) = [0, x]$ e um intervalo também fechado $[x_n, x_{n+1}]$ de $S(x)$ dado por $S'(x)$.

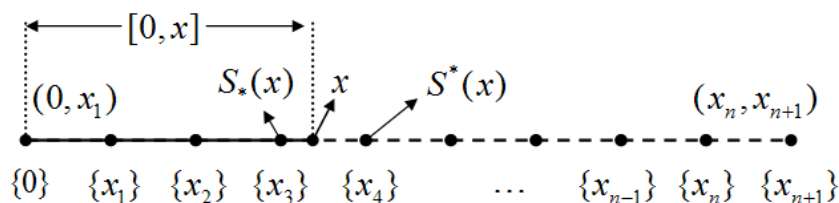


Figura 3.22 – Representação das aproximações em $U = (R^+, S)$

$$S_*(Q(x)) = \{y \in R^+ : S(y) \subseteq Q(x)\} \quad (3.48)$$

$$S^*(Q(x)) = \{y \in R^+ : S(y) \cap Q(x) \neq \emptyset\} \quad (3.49)$$

As expressões para aproximação inferior (3.48) e superior (3.49) são formadas por um conjunto de valores, um segmento de reta. Uma simplificação pode ser feita mantendo os conceitos, e a representação das aproximações passa a ser apenas um número real, ou seja, (3.50) e (3.51).

$$S_*(x) = \text{Sup}\{y \in S : y \leq x\} \quad (3.50)$$

$$S^*(x) = \text{Inf}\{y \in S : y \geq x\} \quad (3.51)$$

Uma analogia com (3.47) é realizada e apresentada em (3.52), onde $\Delta(x, y) = y - x$.

$$\mu_{Q(x)} = \frac{\Delta(Q(x) \cap S(y))}{\Delta(S(y))} \quad (3.52)$$

Substituindo (3.52) tem-se (3.53).

$$\mu_{Q(x)} = \frac{\Delta([0, x] \cap [S_*(x), S^*(x)])}{\Delta([S_*(x), S^*(x)])} = \frac{\Delta([x, S^*])}{\Delta([S_*, S^*])} = \frac{(S^* - x)}{(S^* - S_*)} \quad (3.53)$$

A relação (3.53) indica o erro de aproximação, ou seja, o quanto falta em x para alcançar $S^*(x)$.

Adotando os espaços aproximados $A = (X, B)$ e $B = (Y, C)$, com X e Y sendo conjuntos de valores reais, assume $f : X \rightarrow Y$ a função que define y a partir de x . Assim, as funções f_* e f^* constituem a função inferior (3.54) e superior (3.55), respectivamente.

$$f_*(x) = C_*(f(x)) \quad (3.54)$$

$$f^*(x) = C^*(f(x)) \quad (3.55)$$

Valores intermediários em um SI podem ser estimados por meio de interpolações (3.56) utilizando a definição de função de pertinência aproximada [14].

$$y = f(x) = f(S_*(x)) + \mu_{Q(x)} \cdot \Delta f(S(x)) \quad (3.56)$$

Neste trabalho não será utilizada função de pertinência aproximada, mas uma interpolação linear conforme indicado em [18]. Emprega-se uma expressão (3.57) de interpolação de valores reais usando informações de aproximações inferiores e superiores dos dados utilizados. A precisão da aproximação está relacionada diretamente com a discretização dos dados utilizados.

$$y = f(x_1, x_2, \dots, x_N) = y^* + \frac{(y^* - y_*)}{N} \sum_{N=1}^N \frac{(x - x_*)}{(x^* - x_*)} \quad (3.57)$$

Neste trabalho serão utilizados pacotes computacionais (como o Rosetta, por exemplo) desenvolvidos para trabalhar com conjuntos aproximados, para extrair as informações necessárias para as aplicações desenvolvidas na dissertação. Os próximos itens mostrarão os procedimentos utilizados. Com o objetivo de comprovar a capacidade da TCA de aproximar uma função contínua em t usando as definições citadas, define-se $y(t) = 1 - e^{-0.1t}$, onde t varia de zero a cem, com amostras unitárias. O SI gerado pela função exponencial citada é mostrado na Fig. 3.23, onde o arquivo de dados está no formato texto (txt). A figura em questão ilustra apenas os valores iniciais e finais do SI representado.

t	y
0	0
1	0.0952
2	0.1813
3	0.2592
4	0.3297
5	0.3935
...	...
97	0.9999
98	0.9999
99	0.9999
100	1

Figura 3.23 – Arquivo texto do SI.

Criado o arquivo no formato de texto do SI, aciona-se o software ROSETTA, (<http://www.idi.ntnu.no/~aleks/rosetta/>), e carrega-se o arquivo de dados (Fig. 3.24, 3.25 e 3.26).

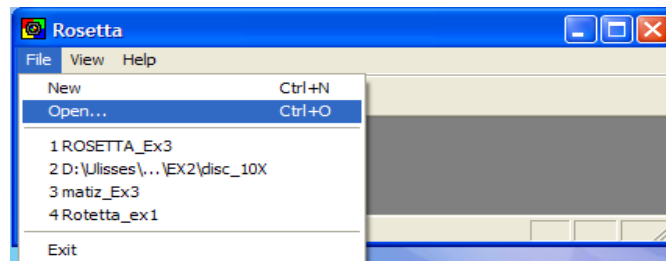


Figura 3.24 – Software ROSETTA.

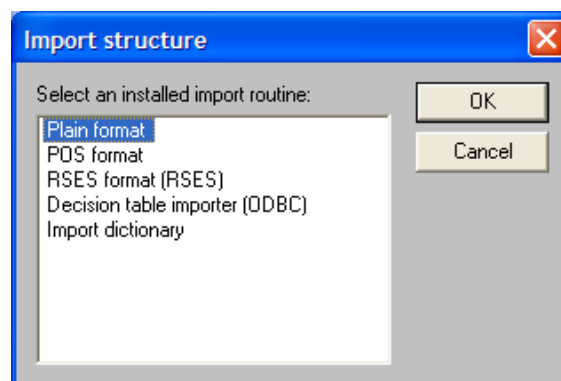


Figura 3.25 – Selecionando o tipo de planilha de dados.

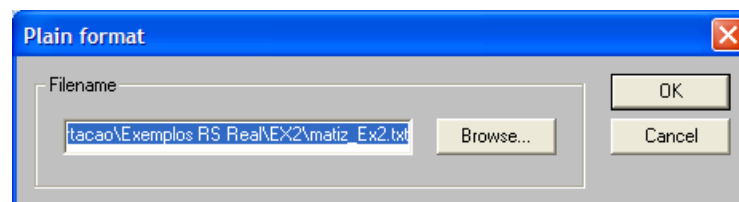


Figura 3.26 – Abrindo um arquivo de dados.

Com o SI carregado devidamente, o mesmo pode ser visto usando a opção *View* (Fig. 3.27).

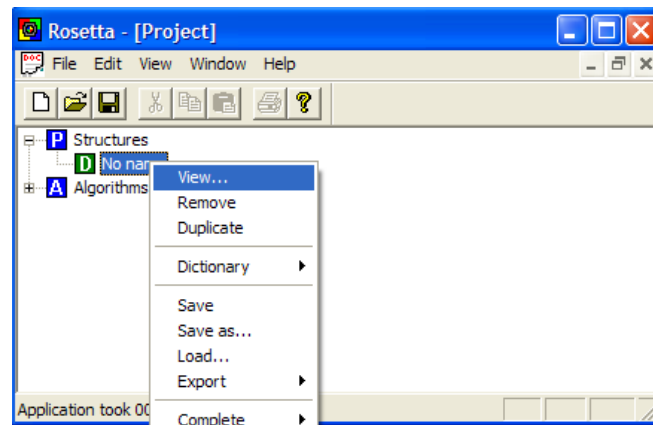


Figura 3.27 – Visualizando o SI.

O ROSETTA então indicará o SI que será trabalhado (Fig. 3.28).

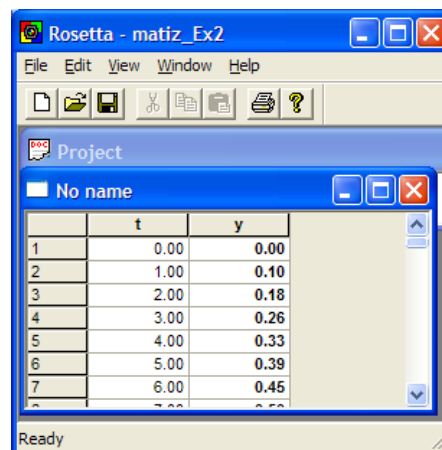


Figura 3.28 – Visualização dos dados.

A primeira etapa para trabalhar com informações contínuas é a discretização das mesmas. Uma das opções do software é opção *Equal frequency binning* (Fig. 3.29). A opção dividirá o conjunto de exemplos em partes iguais onde o número de divisões é solicitado pelo usuário usando a opção *Advanced parameters* (Fig. 3.30).

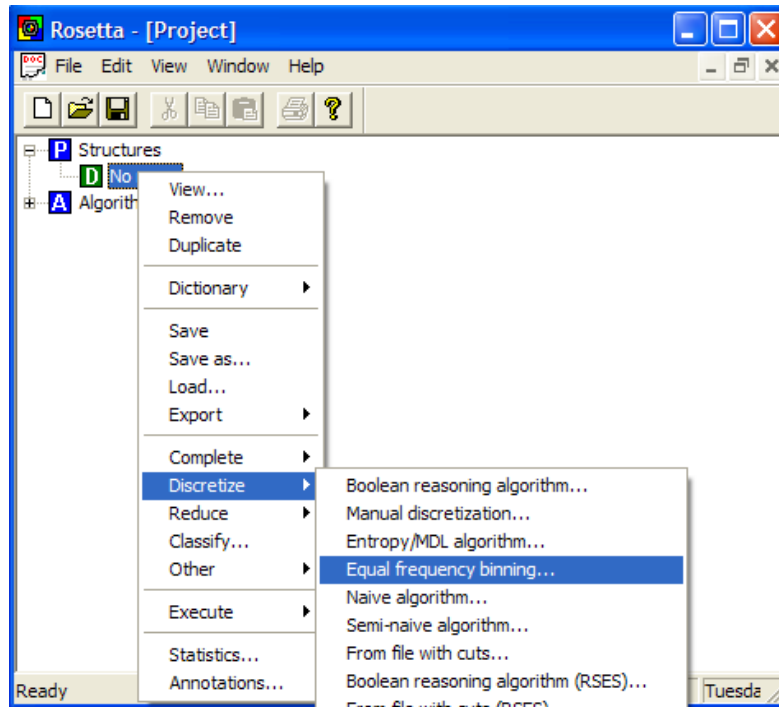


Figura 3.29 – Opções de discretização.

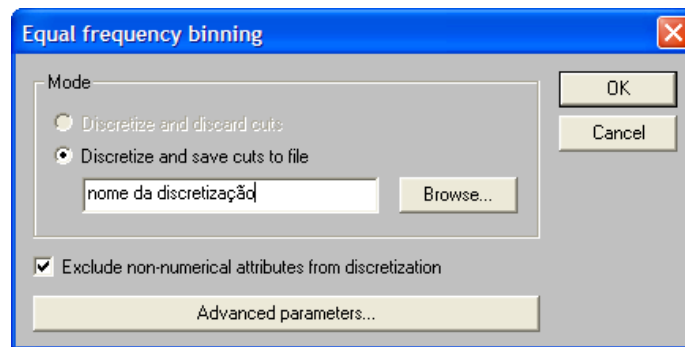


Figura 3.30 – Tela de opções.

Definiu-se que a discretização será de dez divisões, as Fig. 3.31 e Fig. 3.32 ilustram a visualização dos dados discretizados por faixas.

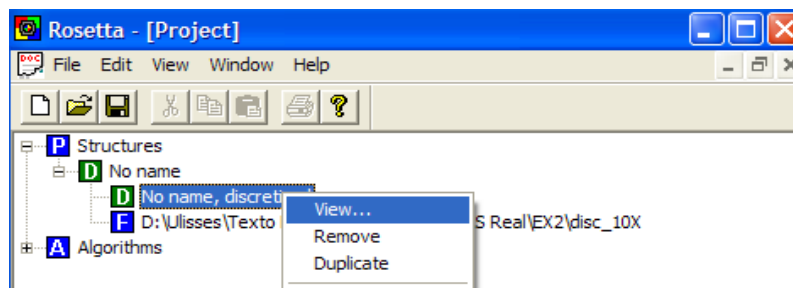
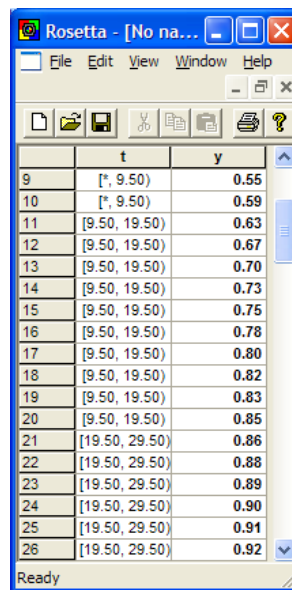


Figura 3.31 – Opção de visualização dos dados discretizados.



	t	y
9	[*, 9.50)	0.55
10	[*, 9.50)	0.59
11	[9.50, 19.50)	0.63
12	[9.50, 19.50)	0.67
13	[9.50, 19.50)	0.70
14	[9.50, 19.50)	0.73
15	[9.50, 19.50)	0.75
16	[9.50, 19.50)	0.78
17	[9.50, 19.50)	0.80
18	[9.50, 19.50)	0.82
19	[9.50, 19.50)	0.83
20	[9.50, 19.50)	0.85
21	[19.50, 29.50)	0.86
22	[19.50, 29.50)	0.88
23	[19.50, 29.50)	0.89
24	[19.50, 29.50)	0.90
25	[19.50, 29.50)	0.91
26	[19.50, 29.50)	0.92

Figura 3.32 – Visualização dos dados discretizados.

A etapa seguinte é mostrada na Fig. 3.33. A opção *Exhaustive Calculation* é uma opção que trabalha com a TCA de forma exaustiva.

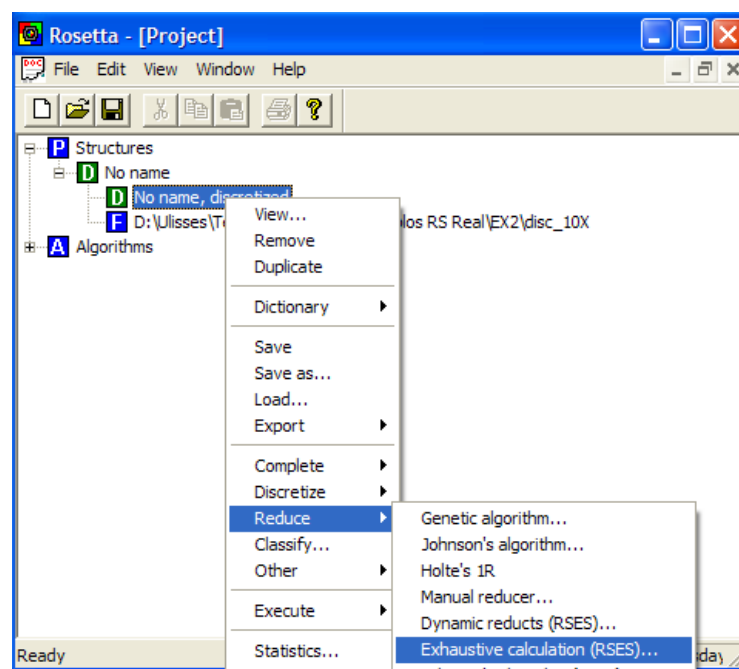


Figura 3.33 – Janela de obtenção de redutos.

Finalmente a etapa de geração das regras: *Generate rules* (Fig. 3.34).

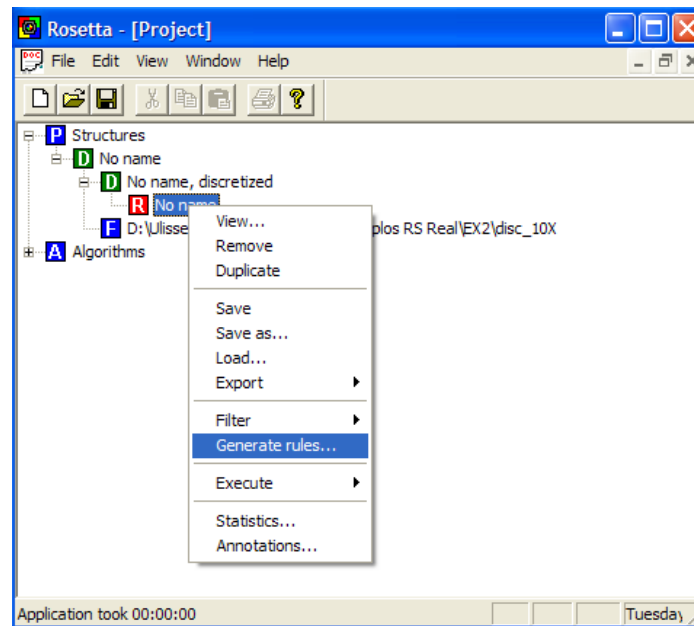


Figura 4.34 – Geração de regras.

As regras podem ser vistas usando a opção *View* mostrada na Fig. 3.35.

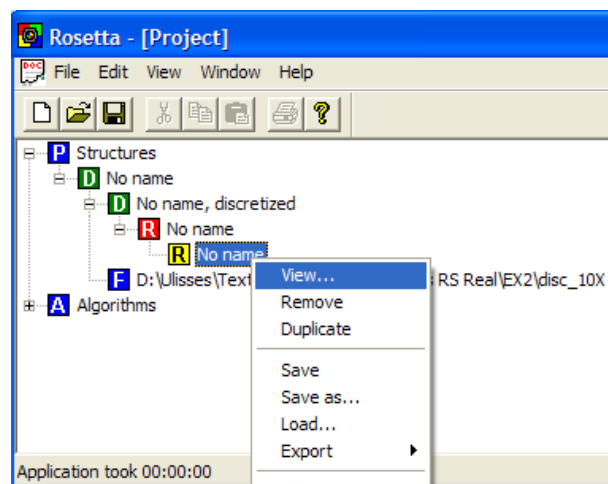


Figura 3.35 – Opção para visualização das regras.

As regras geradas pelo ROSETTA são fornecidas como apresentado na Fig. 3.36.

	Rule	LHS Support
1	$t([*, 9.50)) \Rightarrow y(0.00) \text{ OR } y(0.10) \text{ OR } y(0.18) \text{ OR } y(0.26) \text{ OR } y(0.33) \text{ OR } y(0.39) \text{ OR } y(0.45) \text{ OR } y(0.50) \text{ OR } y(0.55) \text{ OR } y(0.59)$	10
2	$t([9.50, 19.50)) \Rightarrow y(0.63) \text{ OR } y(0.67) \text{ OR } y(0.70) \text{ OR } y(0.73) \text{ OR } y(0.75) \text{ OR } y(0.78) \text{ OR } y(0.80) \text{ OR } y(0.82) \text{ OR } y(0.83) \text{ OR } y(0.85)$	10
3	$t([19.50, 29.50)) \Rightarrow y(0.86) \text{ OR } y(0.88) \text{ OR } y(0.89) \text{ OR } y(0.90) \text{ OR } y(0.91) \text{ OR } y(0.92) \text{ OR } y(0.93) \text{ OR } y(0.94)$	10
4	$t([29.50, 39.50)) \Rightarrow y(0.95) \text{ OR } y(0.96) \text{ OR } y(0.97) \text{ OR } y(0.98)$	10
5	$t([39.50, 49.50)) \Rightarrow y(0.98) \text{ OR } y(0.99)$	10
6	$t([49.50, 59.50)) \Rightarrow y(0.99) \text{ OR } y(1.00)$	10
7	$t([59.50, 69.50)) \Rightarrow y(1.00)$	10
8	$t([69.50, 79.50)) \Rightarrow y(1.00)$	10
9	$t([79.50, 90.50)) \Rightarrow y(1.00)$	11
10	$t([90.50, *)) \Rightarrow y(1.00)$	10

Figura 3.36 – Regras geradas pelo ROSETTA.

Na Fig. 3.36 cada linha representa uma regra e o caractere * indica os valores extremos da variável em questão. Para a primeira e terceira regra da Fig. 3.36 define-se então (3.58) e (3.59) respectivamente.

$$R1: \text{SE } (0 \leq t \leq 9,50) \text{ ENTÃO } y = 0,00 + (0,59 - 0,00) \cdot \left(\frac{t - 0,00}{0,95 - 0,00} \right) \quad (3.58)$$

$$R3: \text{SE } (19,5 \leq t \leq 29,5) \text{ ENTÃO } y = 0,86 + (0,94 - 0,86) \cdot \left(\frac{t - 19,50}{29,50 - 19,50} \right) \quad (3.59)$$

Representando todas as regras geradas pelo ROSETTA de forma equivalente a (3.58) e (3.59) em um algoritmo no MATLAB, compara-se o quão próximo o modelo representado pelas regras aproxima da função original como ilustrado na Fig. 3.37.

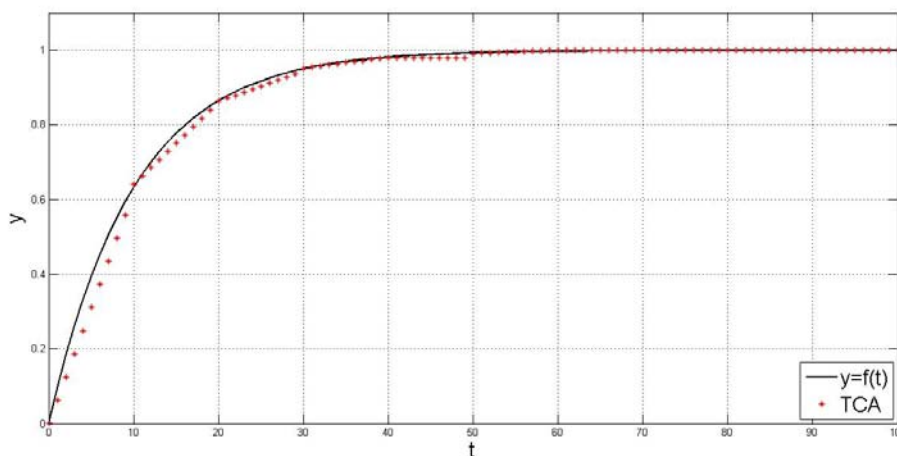


Figura 3.37 – Valores da função original e do modelo baseado nas regras.

É fácil notar que um conjunto de regras adequado pode fornecer um modelo baseado em regras que representou adequadamente a função de exemplo.

Este conceito será utilizado nos próximos capítulos com a finalidade de gerar representações de controladores baseados em regras visando a obtenção de malhas de controle do processo apresentado anteriormente.

CAPÍTULO 4

APLICAÇÕES DE CONTROLADORES APROXIMADOS EM MALHAS DE CONTROLE DE MÁQUINAS ELÉTRICAS

Nos últimos itens do capítulo anterior foram apresentados os conceitos básicos dos conjuntos aproximados. Mostrou-se também a aplicação destes conceitos na modelagem de funções genéricas, para maiores detalhes ver [18]. Na referência citada são sugeridas aplicações em sistemas de controle reais.

Neste capítulo serão mostradas implementações reais de controladores definidos como aproximados por usarem técnicas da TCA. Os controladores resultantes terão ações proporcionais e integrais objetivando a realização prática de malhas de controle de corrente e velocidade para o acionamento de máquinas de corrente contínua.

Neste contexto serão utilizados os modelos dos reguladores de corrente e velocidade convencionais projetados anteriormente objetivando a implementação de controladores aproximados. Neste sentido, seja o diagrama de blocos ilustrado na Fig. 4.1 que exemplifica uma determinada malha de controle. A variável x_1 define o erro da malha, x_2 expressa uma ação proporcional, x_3 indica a ação integral, a variável y simboliza a soma das duas ações de controle, e a variável x_4 representa a grandeza controlada. Esta malha de controle pode ser simulada sob diversas condições operacionais visando à obtenção de dados com a finalidade de gerar as regras do controlador aproximado correspondente. As informações das variáveis x_1 , x_3 , x_4 e y resultantes de simulações do diagrama da Fig. 4.1, podem definir um sistema de informação objetivando gerar as regras para um controlador aproximado.

As variáveis para um controlador aproximado com ação proporcional e integral poderiam ser o erro da malha de controle x_1 , a informação da integral do erro x_3 e a resposta do sistema x_4 .

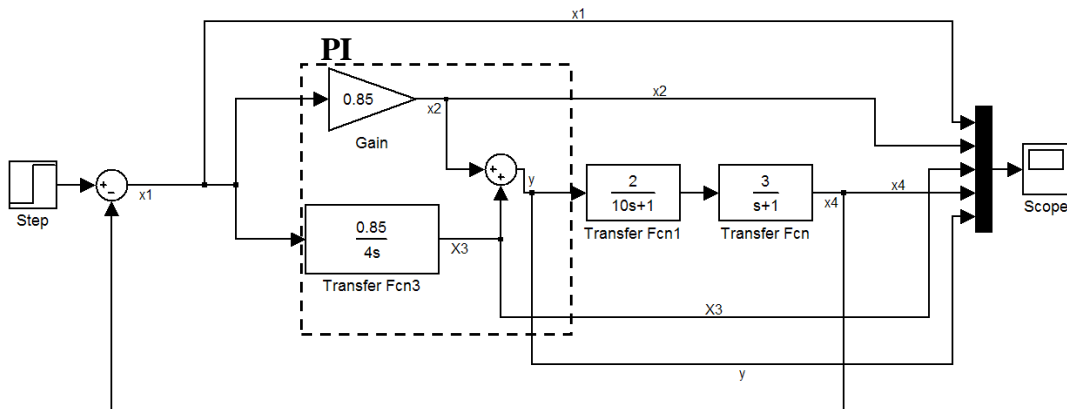


Figura 4.1 – Exemplo de malha de controle.

Conforme ilustrado no capítulo anterior, para este contexto também foi utilizado o software ROSETTA para gerar as regras que representam o controlador aproximado em questão. O conjunto de regras obtido está indicado a seguir.

- R1: $x1([*, 0.05]) \text{ AND } x3([*, 0.02]) \text{ AND } x4([*, 0.85]) \Rightarrow y(0.0000)$
- R2: $x1([0.9501, *]) \text{ AND } x3([*, 0.02]) \text{ AND } x4([*, 0.85]) \Rightarrow y(0.8500)$
- R3: $x1([0.9501, *]) \text{ AND } x3([0.02001, 0.34]) \text{ AND } x4([*, 0.85]) \Rightarrow y(0.8691) \text{ OR } y(0.8842) \text{ OR } y(0.8954) \text{ OR } y(0.9030)$
- R4: $x1([0.7501, 0.95]) \text{ AND } x3([0.02001, 0.34]) \text{ AND } x4([*, 0.85]) \Rightarrow y(0.9073) \text{ OR } y(0.9085) \dots \text{ OR } y(0.8867) \text{ OR } y(0.8757)$
- R5: $x1([0.05001, 0.75]) \text{ AND } x3([0.02001, 0.34]) \text{ AND } x4([*, 0.85]) \Rightarrow y(0.8627) \text{ OR } y(0.8482) \text{ OR } \dots \text{ OR } y(0.7349) \text{ OR } y(0.7131)$
- R6: $x1([0.05001, 0.75]) \text{ AND } x3([0.34001, *]) \text{ AND } x4([*, 0.85]) \Rightarrow y(0.6908) \text{ OR } y(0.6681) \dots y(0.5529) \text{ OR } y(0.5300)$
- R7: $x1([0.05001, 0.75]) \text{ AND } x3([0.34001, *]) \text{ AND } x4([0.85001, 1]) \Rightarrow y(0.5074) \text{ OR } y(0.4851) \text{ OR } y(0.4632) \text{ OR } y(0.4416)$
- R8: $x1([*, 0.05]) \text{ AND } x3([0.34001, *]) \text{ AND } x4([0.85001, 1]) \Rightarrow y(0.4206) \text{ OR } y(0.4000)$
- R9: $x1([*, 0.05]) \text{ AND } x3([0.34001, *]) \text{ AND } x4([1.00001, *]) \Rightarrow y(0.3800) \text{ OR } y(0.3606) \text{ OR } y(0.3417) \dots \text{ OR } y(0.1576) \text{ OR } y(0.1482)$
- R10: $x1([*, 0.05]) \text{ AND } x3([0.02001, 0.34]) \text{ AND } x4([1.00001, *]) \Rightarrow y(0.1394) \text{ OR } y(0.1313) \text{ OR } y(0.1238) \dots \text{ OR } y(0.1666) \text{ OR } y(0.1665)$
- R11: $x1([*, 0.05]) \text{ AND } x3([0.02001, 0.34]) \text{ AND } x4([0.85001, 1]) \Rightarrow y(0.1627) \text{ OR } y(0.1636) \text{ OR } y(0.1645) \text{ OR } \dots \text{ OR } y(0.1682) \text{ OR } y(0.1681)$

Os valores das regras podem ser computados por meio da expressão de interpolação (3.57). Por exemplo, para a regra R3 a expressão correspondente é dada por (4.1).

$$y = 0,8691 + \left(\frac{0,903 - 0,8691}{3}\right) \cdot \left(\frac{x1 - 0,9501}{1 - 0,9501}\right) + \left(\frac{x3 - 0,02}{0,34 - 0,02}\right) + \left(\frac{x4 - 0}{0,85 - 0}\right) \quad (4.1)$$

A Fig. 4.2 mostra os resultados de uma simulação com os dados de saída de um controlador PI convencional (gráfico em cor azul) e do controlador aproximado (em cor vermelha). A Fig. 4.3 ilustra o comportamento da variável controlada da malha de controle sob a ação dos dois controladores (clássico e aproximado), usando como base o modelo do processo representado na Fig. 4.1.

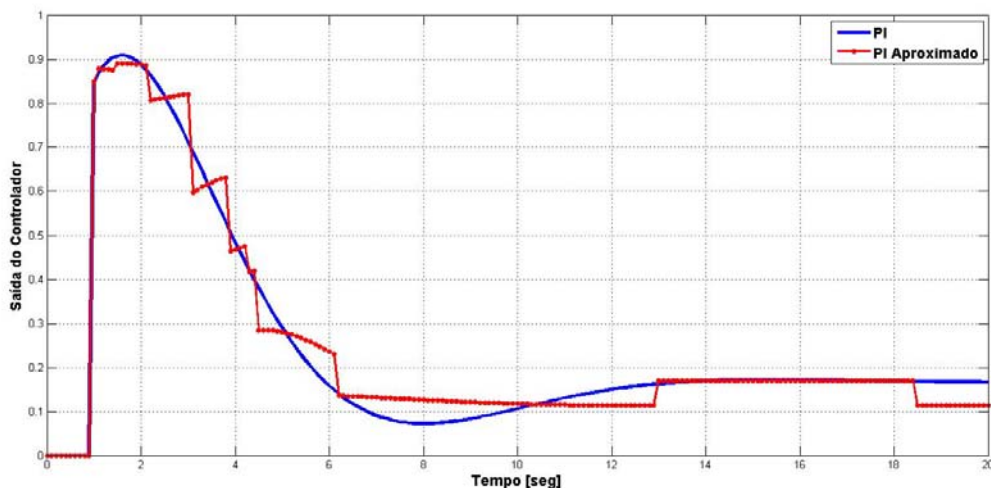


Figura 4.2 – Comparação entre as saídas dos controladores (clássico e aproximado).

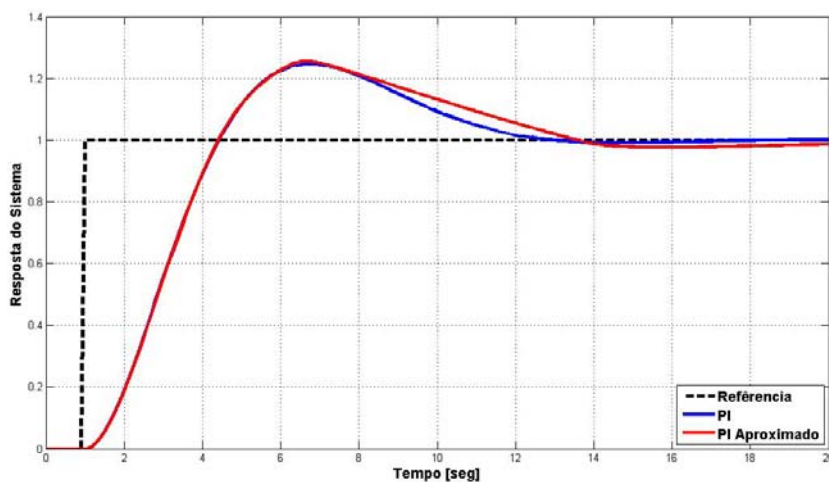


Figura 4.3 – Dinâmica da malha sob ação dos dois tipos de controladores.

O procedimento adotado na obtenção dos controladores aproximados para a implementação das malhas de regulação de corrente e velocidade está indicado a seguir. A função do controlador de velocidade convencional utilizado está indicada em (4.2). Os atributos são as informações do erro da malha, da sua integral e da saída do controlador. Explicitando (4.2) com intuito de deixar a saída do controlador em função dos atributos de condições tem-se (4.3), onde $V_v = 3,067$ e $T_v = 0,478$.

$$PI_v(s) = \frac{Y_v(s)}{E_v(s)} = \frac{V_v(T_v s + 1)}{T_v s} \quad (4.2)$$

$$\underbrace{Y_v(s)}_y = 3,067 \cdot \underbrace{E_v(s)}_{x1} + 4,4163 \cdot \underbrace{\frac{E_v(s)}{s}}_{x2} \quad (4.3)$$

Simulou-se a função em questão para valores aleatórios nas variáveis da mesma na faixa entre -1 e 1. A Fig. 4.4 ilustra os resultados correspondentes. As regras resultantes (obtidas por meio do software Rosetta) estão apresentadas no algoritmo de implementação do regulador de velocidade aproximado, cuja listagem encontra-se no Apêndice B. A Fig. 4.5 apresenta a comparação dos resultados obtidos pelas regras que representam o controlador aproximado (pontos vermelhos), com os dados (pontos azuis) da função do controlador convencional (4.3). A Fig. 4.6 é uma ampliação de uma região da Fig. 4.5.

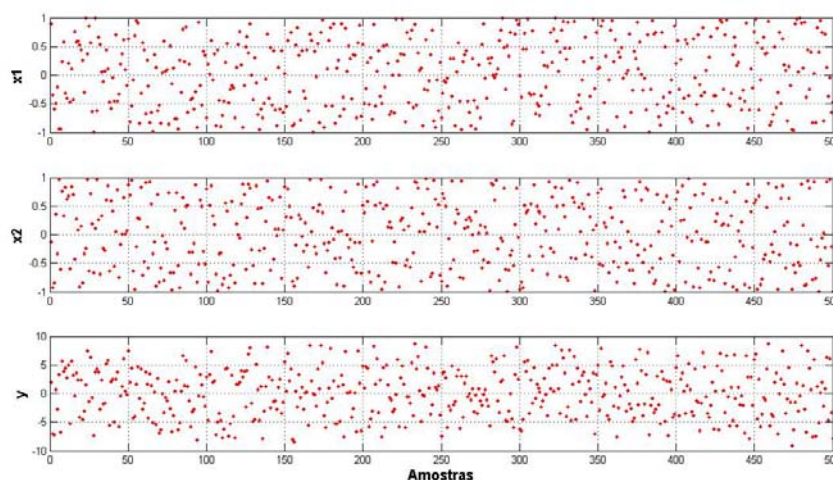


Figura 4.4 – Dados do SI para o regulador de velocidade.

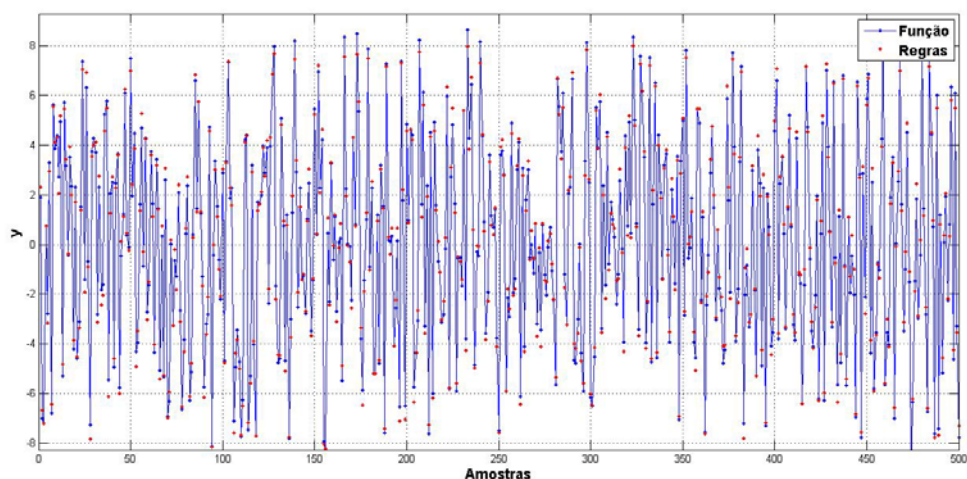


Figura 4.5 – Comparação entre os dados do PI clássico e do controlador aproximado.

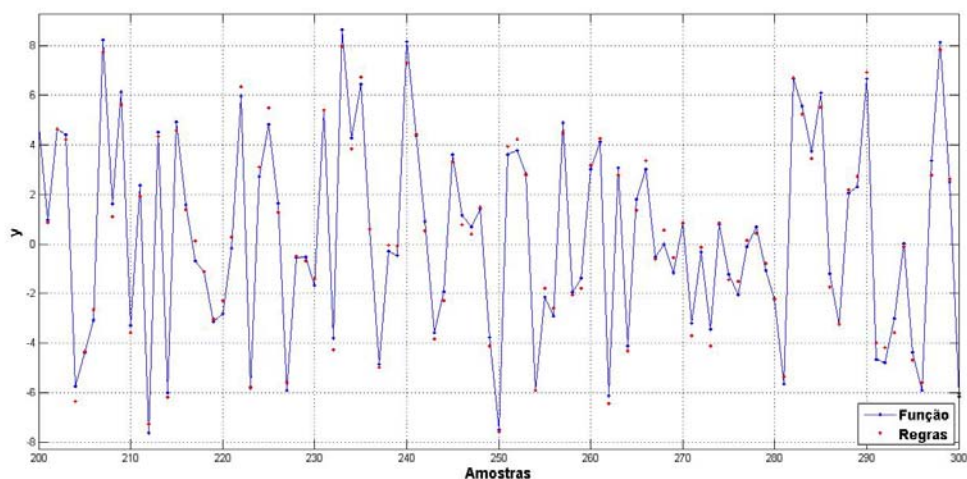


Figura 4.6 – Ampliação da Fig. 4.5.

Procedeu-se de forma similar para a obtenção do controlador de corrente definido por (4.4) e (4.5), onde $V_c = 0,4567$ e $T_c = 45 \cdot 10^{-3}$. A Fig. 4.7 ilustra os dados obtidos de simulações da expressão (4.5) resultante

$$PI_c(s) = \frac{Y_c(s)}{E_c(s)} = \frac{V_c(T_c s + 1)}{T_c s} \quad (4.4)$$

$$\underbrace{Y_c(n)}_y = 0,5074 \cdot \underbrace{E_c(n)}_{x1} + -0,4059 \cdot \underbrace{E_c(n-1)}_{x2} + \underbrace{Y_c(n-1)}_{x3} \quad (4.5)$$

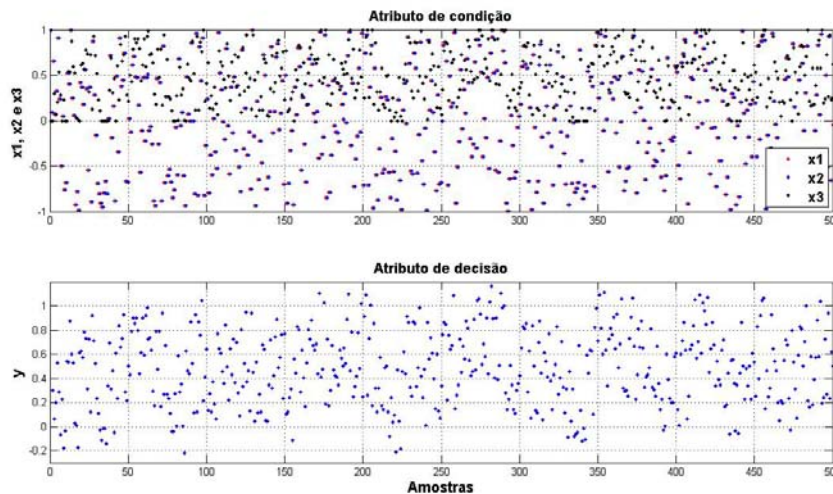


Figura 4.7 – Dados do regulador de corrente.

As regras resultantes estão apresentadas no algoritmo de implementação do controlador de corrente aproximado, cuja listagem encontra-se no Apêndice B. A Fig. 4.8 apresenta a comparação dos resultados obtidos pelas regras que representam o controlador aproximado (pontos vermelhos), com os dados (pontos azuis) da expressão do controlador convencional (4.5). A Fig. 4.9 é uma ampliação de uma região da Fig. 4.8.

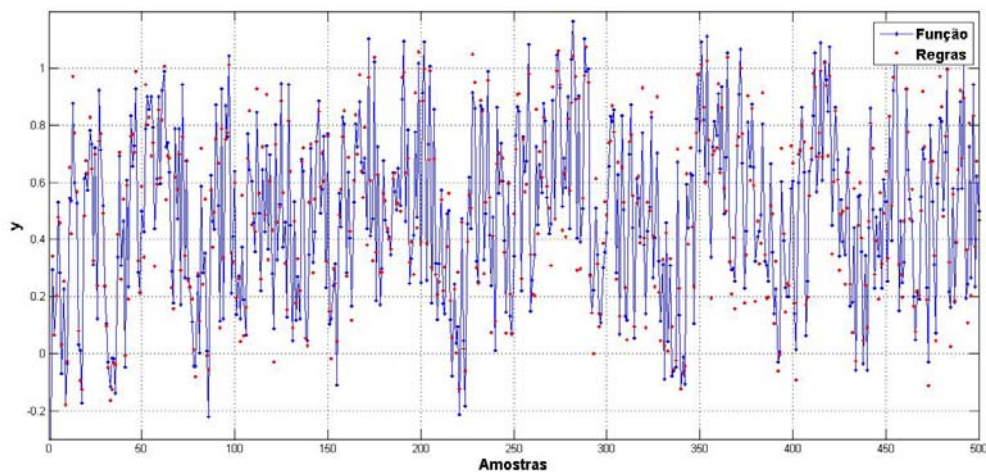


Figura 4.8 – Comparação entre os dados do PI clássico e do controlador aproximado.

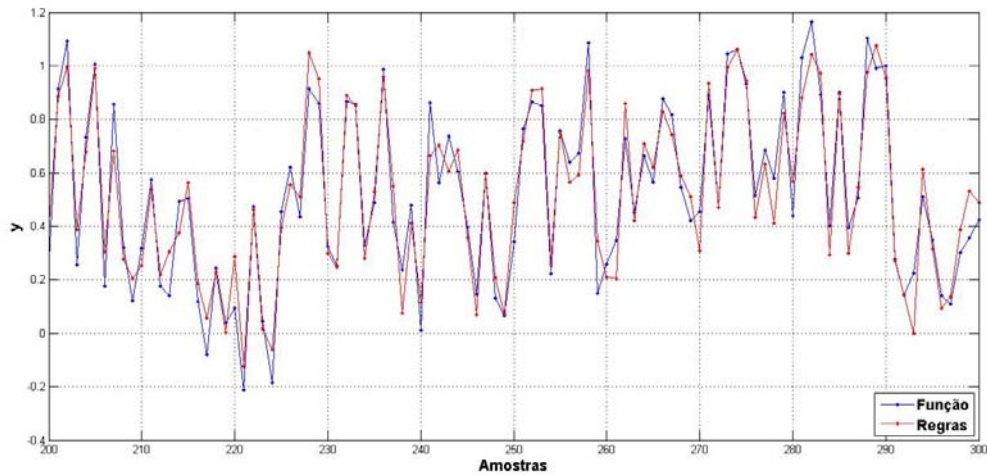


Figura 4.9 – Ampliação da Fig. 4.8.

A Fig. 4.10 ilustra resultados de simulações do sistema citado sob efeito das malhas de controle de corrente e velocidade convencionais e aproximadas mediante variações da referência de velocidade. A Fig. 4.11 mostra a regulação de corrente e velocidade mediante variações de carga no sistema.

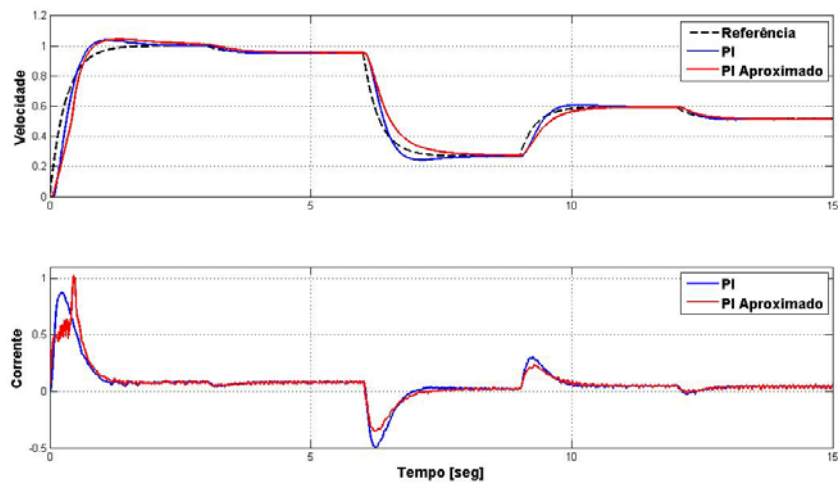


Figura 4.10 – Respostas dinâmicas mediante variação da referência de velocidade.

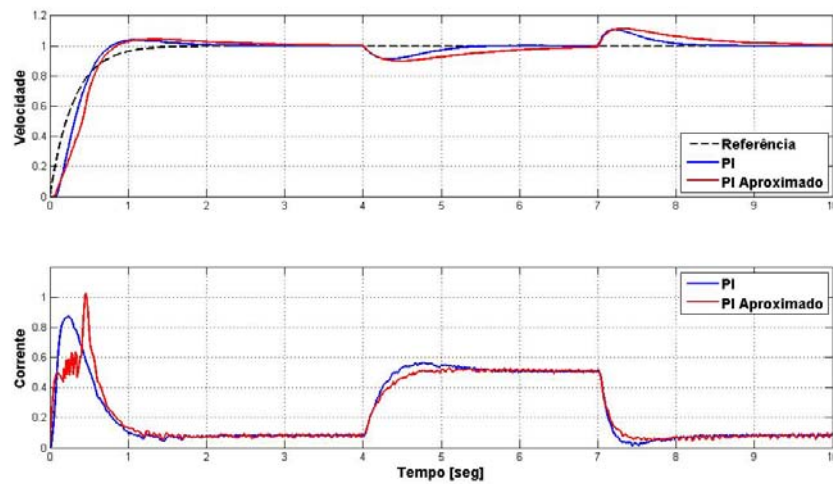


Figura 4.11 – Respostas dinâmicas mediante variação de carga no sistema.

Uma vez que os resultados das simulações do modelo do sistema em estudo se mostraram adequados, a próxima etapa mostrará os resultados práticos obtidos na bancada de ensaio. Os resultados experimentais ensaiados serão apresentados no próximo capítulo.

CAPÍTULO 5

RESULTADOS

Os principais resultados experimentais obtidos neste trabalho serão apresentados neste capítulo. Serão mostradas as respostas dos ensaios realizados comparando-se os resultados obtidos com as técnicas de controle utilizadas, tanto com malhas convencionais utilizando compensadores PI, como com malhas empregando controladores aproximados.

As informações obtidas originaram-se de medidas reais realizadas na bancada de ensaio construída. Os dados foram coletados por meio da placa de aquisição de dados utilizada no sistema. O tempo de amostragem foi ajustado em 10 [ms] e a janela de coleta de dados deve duração típica de 10 segundos.

As perturbações de torque no sistema foram realizadas por intermédio de comutação de cargas elétricas ligadas ao gerador síncrono acoplado mecanicamente ao motor de corrente contínua da bancada. As variações das cargas foram definidas em relação à potência nominal do sistema e foram definidas nas condições de “carga plena”, “meia carga” e “a vazio” (motor sem carga). As cargas foram definidas a partir de dois níveis de operação: com excitação do campo do gerador síncrono em corrente nominal; com metade da corrente nominal de excitação.

Nas mesmas condições operacionais dos ensaios práticos realizados, foram incluídos também resultados de simulações numéricas que utilizaram as informações da modelagem matemática do sistema. A finalidade é verificar a precisão da modelagem e validar os resultados práticos obtidos.

Serão apresentadas primeiramente as respostas das malhas utilizando compensadores PI convencionais. Depois serão apresentadas as respostas das malhas empregando-se controladores aproximados.

As Fig. 5.1 a 5.5 são relativas à partida do motor sem carga. A curva tracejada em cor preta indica a informação de referência (*set-point*) de velocidade que consiste de uma entrada em degrau unitário filtrada por um filtro de primeira ordem, cuja finalidade é evitar variações muito bruscas de comando comuns em sistemas práticos. As curvas em verde e azul indicam a velocidade e a corrente do sistema real, respectivamente. As

curvas em cor vermelha e na cor roxa são valores da velocidade e da corrente do motor simuladas a partir do modelo do sistema. Foram utilizados controladores proporcionais e integrais obtidos pelo método de otimização simétrico. Após quatro segundos da partida, o sistema é perturbado com a entrada de carga elétrica no sistema e aos sete segundos esta carga é retirada. Na Fig. 5.1 a variação é de meia carga e na Fig. 5.2 é de carga plena. Nas Fig. 5.3 e 5.4 têm-se as mesmas condições de variações de carga, mas agora com referências de entrada de velocidade em 0,5 [pu].

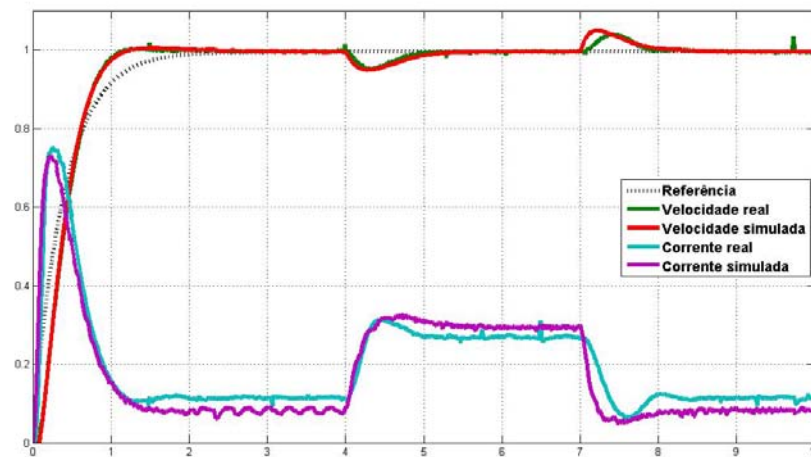


Figura 5.1 – Respostas na presença de distúrbios, meia carga e referência de 1 [pu].

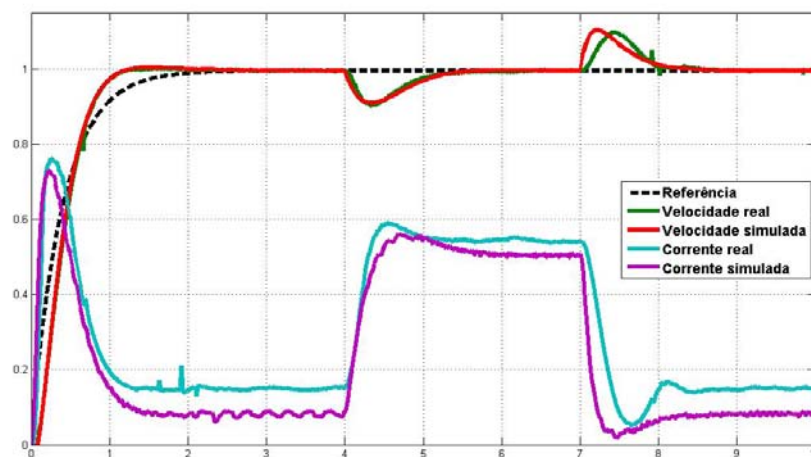


Figura 5.2 – Respostas na presença de distúrbios, carga plena e referência de 1 [pu].

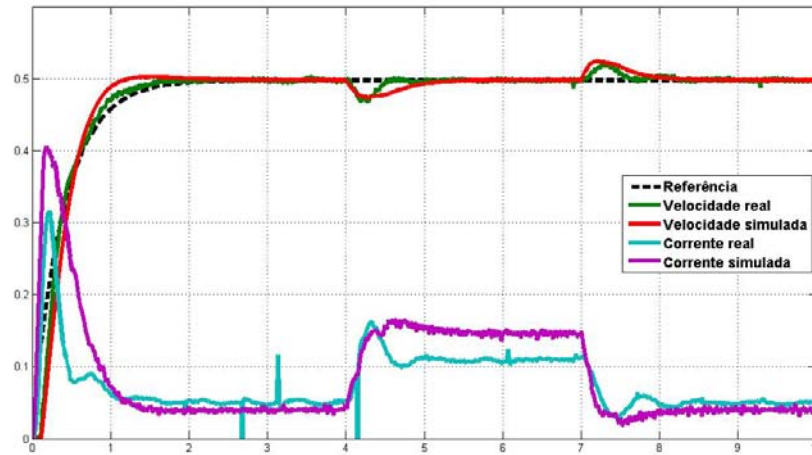


Figura 5.3 – Respostas na presença de distúrbios, meia carga e referência de 0,5 [pu].

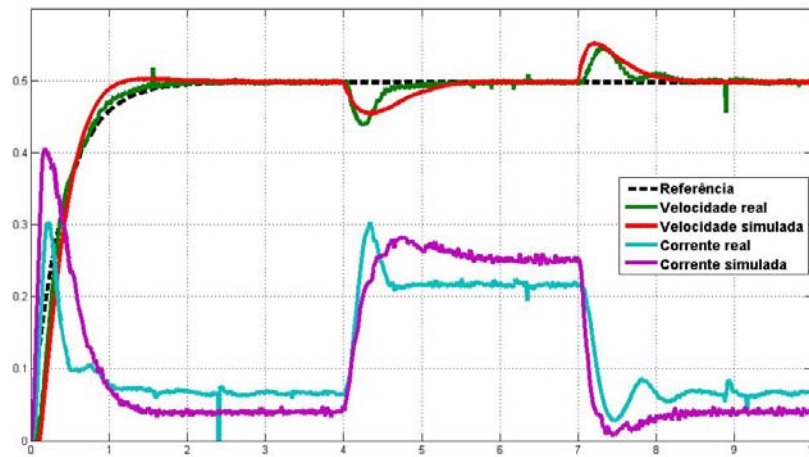


Figura 5.4 – Respostas na presença de distúrbios, carga plena e referência de 0,5 [pu].

A Fig. 5.5 mostra os resultados obtidos para um *set-point* de 0,3 [pu] e perturbação a plena carga. Os resultados obtidos no sistema real e no modelo simulado apresentaram semelhanças razoáveis (principalmente para velocidades nominais) validando a modelagem realizada na bancada de testes. As diferenças obtidas com valores de referência distintos de velocidade, podem ser atribuídas a não-linearidades não modeladas como saturações decorrentes do conversor D/A e do circuito do *driver* de potência empregados no sistema.

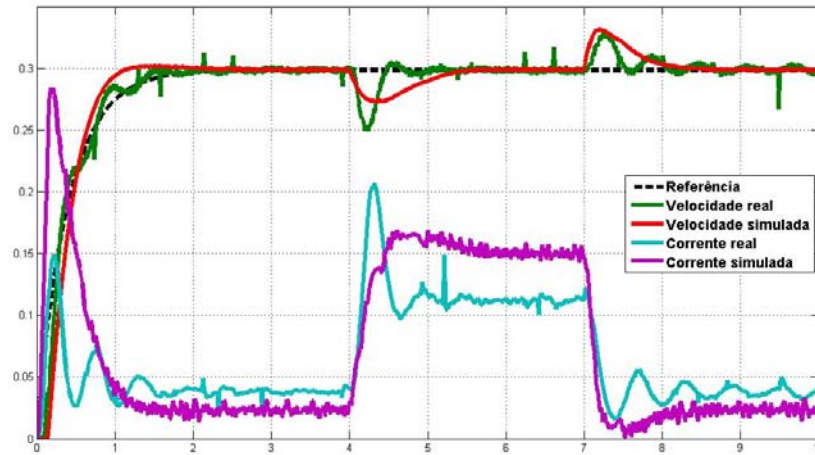


Figura 5.5 – Respostas na presença de distúrbios, carga plena e referência de 0,3 [pu].

As Fig. 5.6 a 5.11 referem-se a comparações de resultados reais entre malhas de controle com compensadores convencionais e controladores aproximados. Foram utilizadas condições similares de referência de velocidade, situações de partida e variações de carga semelhantes às empregadas anteriormente. Em cor azul estão representados os resultados reais obtidos com compensadores convencionais e em cor vermelha os resultados com controladores aproximados.

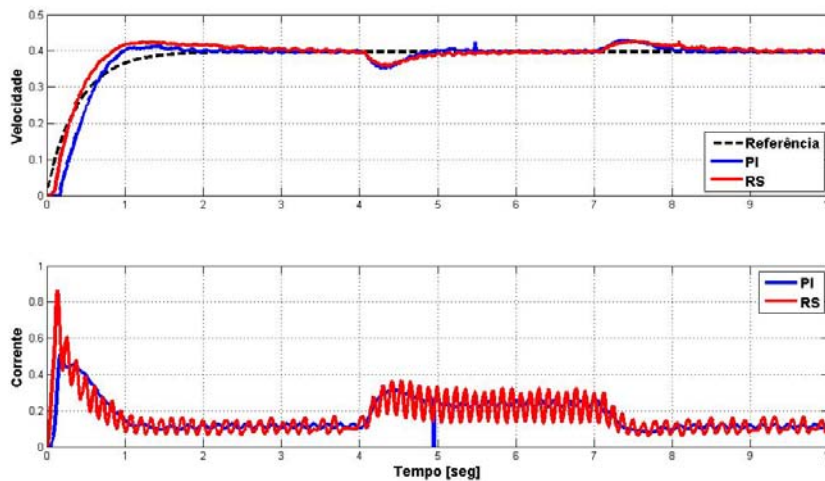


Figura 5.6 – Respostas na presença de distúrbios, meia carga e referência de 0,4 [pu].

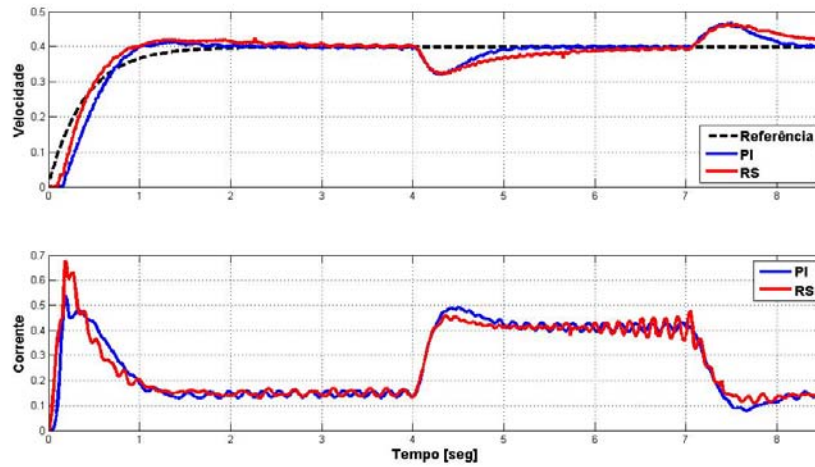


Figura 5.7 – Respostas na presença de distúrbios, carga plena e referência de 0,4 [pu].

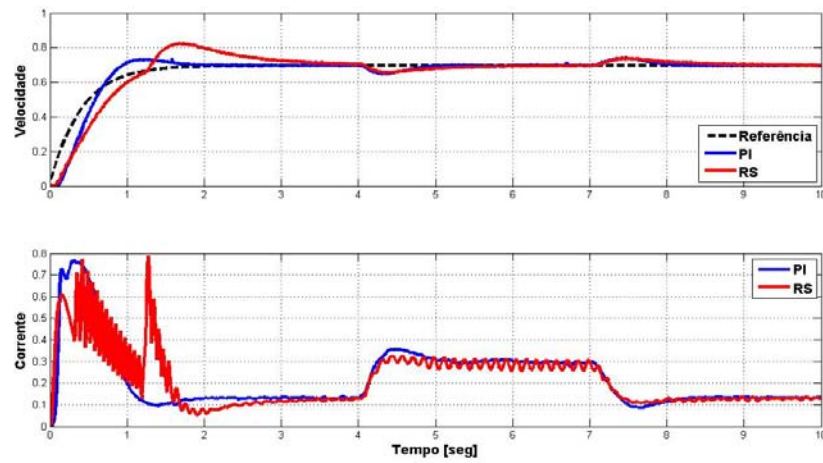


Figura 5.8 – Respostas na presença de distúrbios, meia carga e referência de 0,7 [pu].

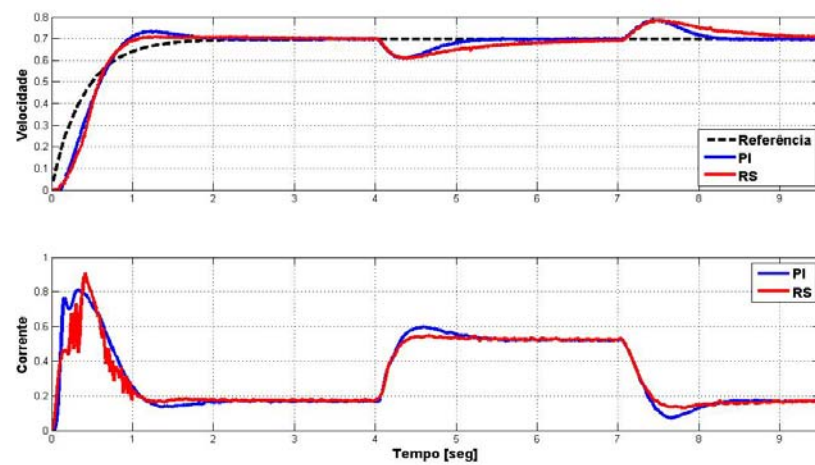


Figura 5.9 – Respostas na presença de distúrbios, carga plena e referência de 0,7 [pu].

As oscilações limitadas a certas faixas que aparecem na regulação de corrente relativa ao controlador aproximado, ocorrem com maior frequência em situações de rotações mais baixas e com cargas menores. Isto pode ser atribuído ao mapeamento das regras geradas na obtenção do controlador de corrente aproximado. Com um maior número de dados ou regras representativas, este mapeamento pode se tornar mais eficiente e deste modo as oscilações tendem a serem menores. A finalidade principal deste trabalho consiste em mostrar a potencialidade do uso de controladores aproximados em aplicações práticas.

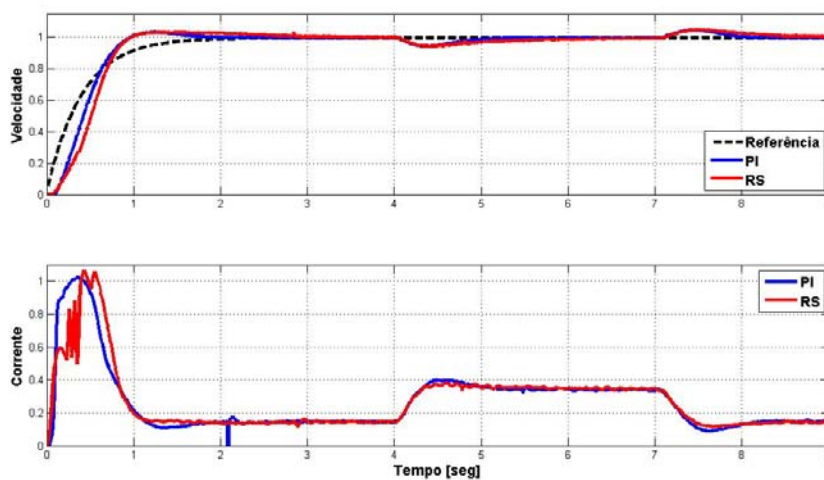


Figura 5.10 – Respostas na presença de distúrbios, meia carga e referência de 1 [pu].

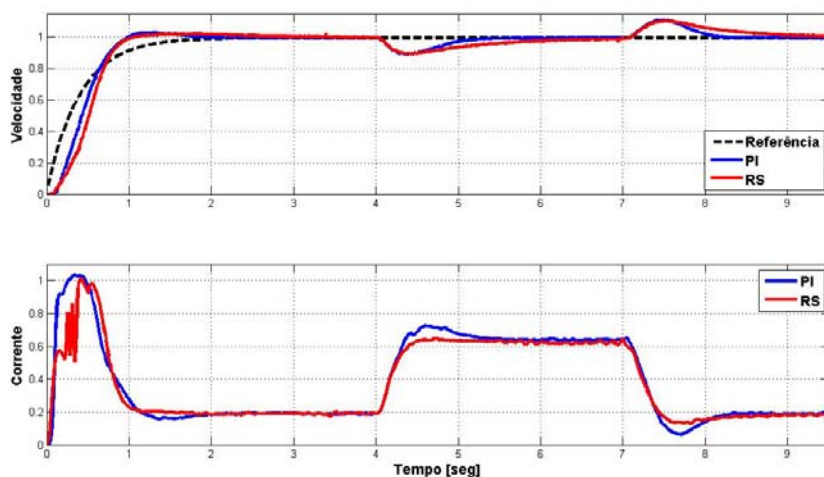


Figura 5.11 – Respostas na presença de distúrbios, carga plena e referência de 1 [pu].

A Fig. 5.12 mostra comparações de simulações do processo em questão com o motor partindo com carga plena, onde após sete segundos do início do ensaio a mesma é retirada. Os gráficos são relativos a malhas de controle com compensadores convencionais e com controladores aproximados. A Fig. 5.13 mostra resultados reais para as mesmas condições operacionais. A Figura 5.14 indica os resultados para uma referência de 0,8 [pu]. Verifica-se que o desempenho dos controladores aproximados ficou melhor nestas condições operacionais do que com compensadores convencionais. A explicação é que as regras resultantes dos controladores aproximados apresentam uma determinada capacidade de interpolação dos dados provenientes das ações de comandos oriundas das informações de sistemas reais (mesmo na presença de eventuais comportamentos não-linearidades que processos reais possam apresentar). Isto indica que esta abordagem pode ser interessante em aplicações de sistemas de controle de processos não-lineares, ou até mesmo com parâmetros variantes no tempo.

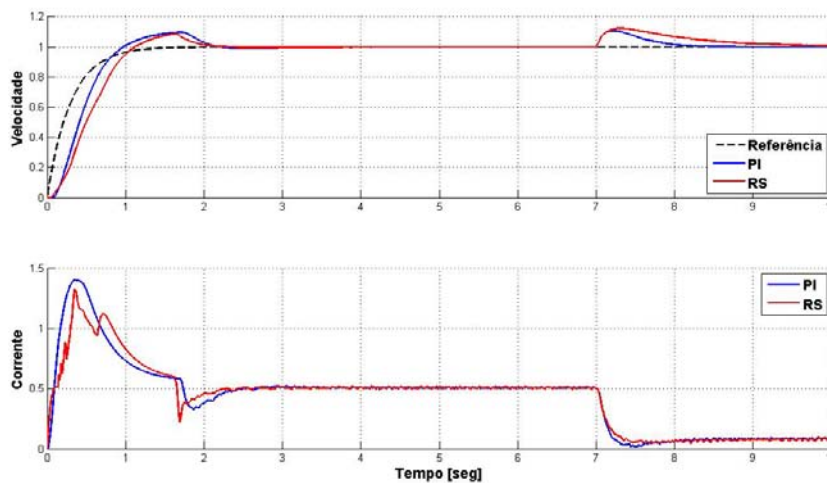


Figura 5.12 – Respostas simuladas com partida a carga plena e referência de 1 [pu].

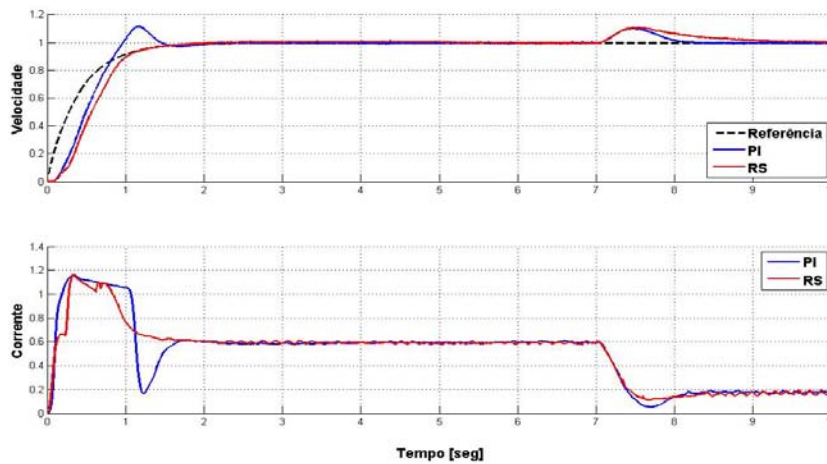


Figura 5.13 – Respostas reais com partida a carga plena e referência 1 [pu].

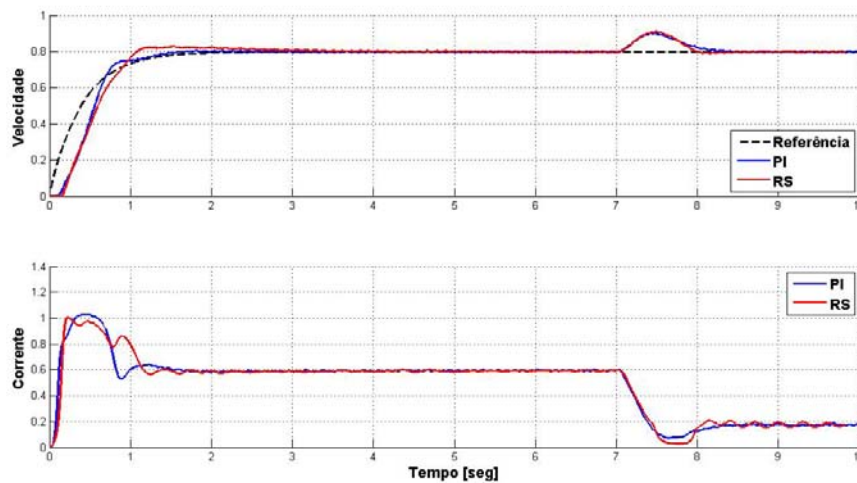


Figura 5.14 – Respostas reais com partida a carga plena e referência 0,8 [pu].

Os resultados obtidos das malhas de controle utilizando compensadores convencionais e das malhas com controladores aproximados são semelhantes, uma vez que as regras dos últimos foram derivadas das funções originais dos primeiros.

A finalidade principal do trabalho foi mostrar a potencialidade de aplicações de controladores aproximados em processos reais, principalmente naqueles que envolvem mais de uma malha de controle, como em malhas de regulação de velocidade e corrente de máquinas elétricas.

Os resultados obtidos indicaram que na regulação de corrente, principalmente na presença de cargas maiores, as malhas com controladores aproximados apresentaram respostas menos abruptas do que as com compensadores convencionais. Isto é interessante em aplicações de acionamentos elétricos, pois implica em características de torque mais homogêneas.

Na prática, sistemas de controle reais tendem a apresentar não-linearidades e os controladores aproximados por serem aproximadores de funções, tendem a mapear características de compensações mais adequadas que os controladores convencionais.

Em algumas aplicações práticas, os valores dos ganhos dos compensadores das malhas de controle devem ser modificados em relação aos dados de operação dos *set-point* utilizados. Neste contexto, controladores aproximados podem ser empregados para o mapeamento efetivo dos ganhos dentro do conjunto das regras utilizadas, possibilitando deste modo sistemas de controle mais eficientes.

CAPÍTULO 6

CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho teve como objetivo principal realizar implementações práticas de uma técnica nova para sistemas de controle que utiliza controladores aproximados. Como exemplificação prática utilizou-se uma aplicação clássica relacionada a acionamentos de máquinas de corrente contínua envolvendo malhas típicas de regulação de corrente e velocidade.

Os resultados obtidos foram comparados com técnicas de controle conhecidas como o método de compensação simétrico.

Foram empregados recursos computacionais para simulações numéricas dos modelos matemáticos desenvolvidos. Uma bancada de testes foi montada para validar os resultados de forma experimental.

Os resultados alcançados por meio de simulações computacionais e por ensaios práticos foram bons e validaram a proposição do trabalho.

As semelhanças ou diferenças nas características das respostas obtidas entre as técnicas aplicadas se devem as diversidades estruturais envolvidas nas mesmas.

Como esperado, os resultados provenientes de simulações e ensaios práticos foram bem parecidos, confirmando a qualidade dos modelos matemáticos desenvolvidos.

Algumas diferenças nas dinâmicas obtidas, como na forma de pequenas oscilações na regulação de corrente para o controlador aproximado, podem ser facilmente minimizadas aumentando-se o número de regras do controlador aproximado, especificamente nas regiões de operações com menores valores de referência de rotação.

O melhor desempenho das malhas com controladores aproximados em condições de cargas maiores pode ser atribuído a capacidade de interpolação dos mesmos. Na prática, processos reais apresentam não-linearidades como saturações, zonas mortas e outras que não são plenamente compensadas com controladores convencionais.

Os resultados obtidos indicam que as técnicas de controle com controladores aproximados podem ser promissoras em aplicações práticas, apresentando a potencialidade de serem aplicados em sistemas de controle gerais, principalmente em processos não-lineares.

Como trabalho futuro pode-se elencar a aplicação dos conceitos utilizados nesta dissertação no controle de processos reais mais complexos, por exemplo em sistemas com múltiplas entradas e saídas que apresentem não-linearidades.

Outra área de pesquisa a ser investigada é a aplicação das idéias utilizadas nesta dissertação na geração de regras difusas para controladores nebulosos.

APÊNDICE A

A.1 – Circuito PWM

Esboça-se aqui o circuito responsável pela geração do PWM, Fig. A1. O principal componente é o circuito integrado TCA 780, este componente gera um sinal PWM a partir de uma comparação entre uma sinal dente de serra e uma referência de tensão contínua. O sinal dente de serra tem o seu período definido por um gerador de frequência externo e sua inclinação ou máximo pico pode ser regulado através da resistência variável R4.

A referência de tensão contínua para a comparação com a triangular é conectada no ponto TP1 do circuito, ou pino de número 11, o PWM proporcional a esta tensão contínua é obtida no ponto TP2.

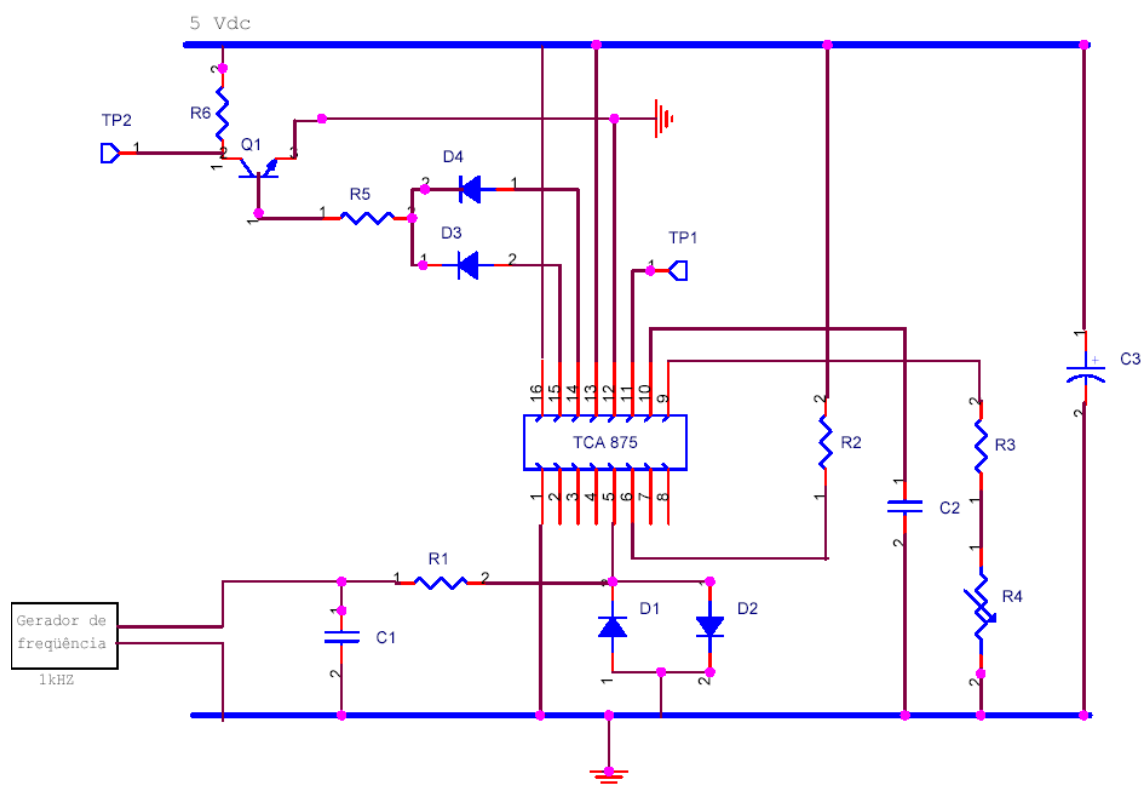


Figura A1 – Circuito PWM

Valores dos componentes:

R1= 1K Ω ;

R2= 15K Ω ;

R3, R5= 47K Ω ;

R4= 100K Ω ;

R6= 100 Ω ;

D1, D2, D3, D4= 1N4148;

C1= 47nF;

C2= 56nF;

C3= 47 μ F;

Q1= BC548;

O PWM gerado por este circuito tem amplitude de cinco volts [Vcc]. Para o driver de disparo do IGBT (SKHI 22A) é necessário um sinal PWM com amplitude de quinze volts [Vcc]. Para esta finalidade pode-se usar o buffer 7417.

APÊNDICE B

B.1 – Implementação do Controlador PI Simétrico

O projeto no ambiente do SIMULINK para os controladores PI convencional é apresentado na Fig. B1, onde o mesmo se apresenta subdividido em:

- 1- Bloco responsável pela entrada e saída da carga.
- 2- Bloco para leitura e calibração da variável de velocidade.
- 3- Bloco para leitura e calibração da variável de corrente.
- 4- PI de Velocidade representado por um subsistema.
- 5- PI de Corrente representado por um subsistema.

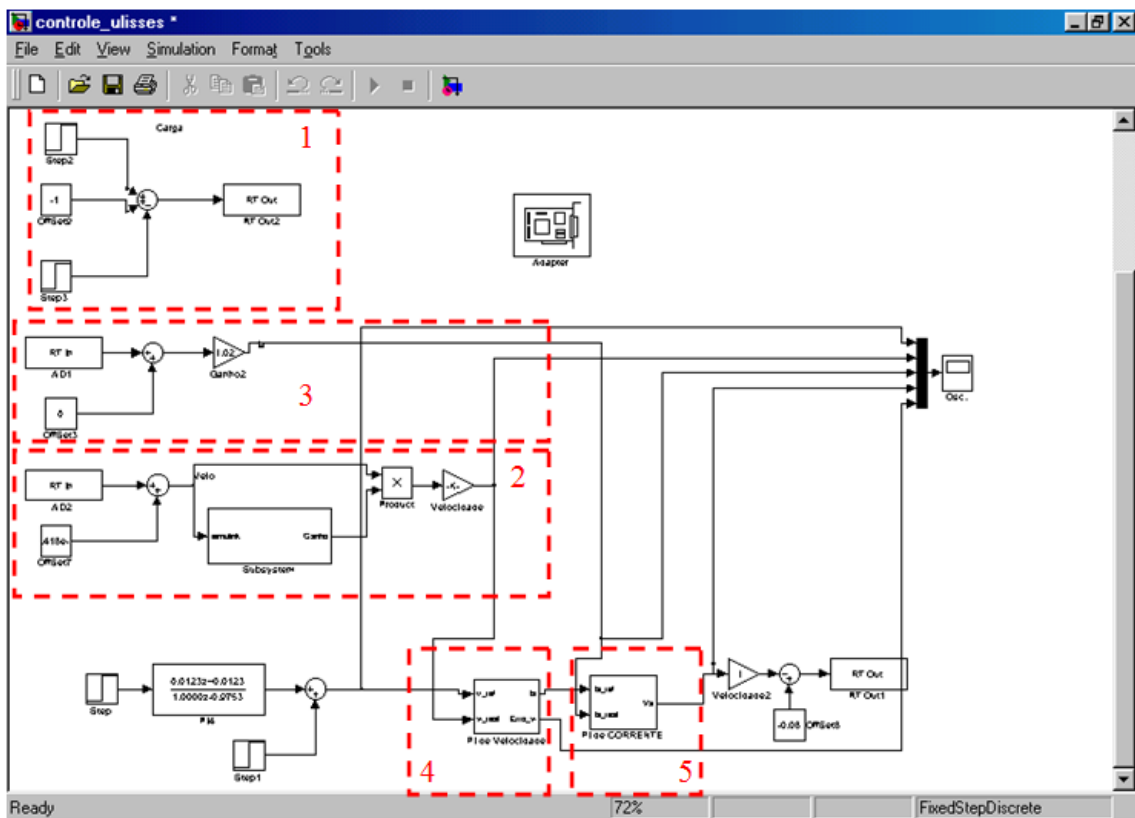


Figura B1- Diagrama para o controlador PI

B.2 – Implementação do Controlador Aproximado

Para o projeto, também no SIMULINK, do controlador aproximado usou-se o bloco S-Function, esse bloco substitui os controladores PI convencionais pelo o conjunto de regras, a linguagem de programação deste bloco é a linguagem C. A Fig. B2 representa o diagrama de controle usado para os controladores aproximados, onde a base do projeto é mantida a mesma, tendo as seguintes alterações:

- 1- Controlador Aproximado de velocidade.
- 2- Controlador Aproximado de corrente.

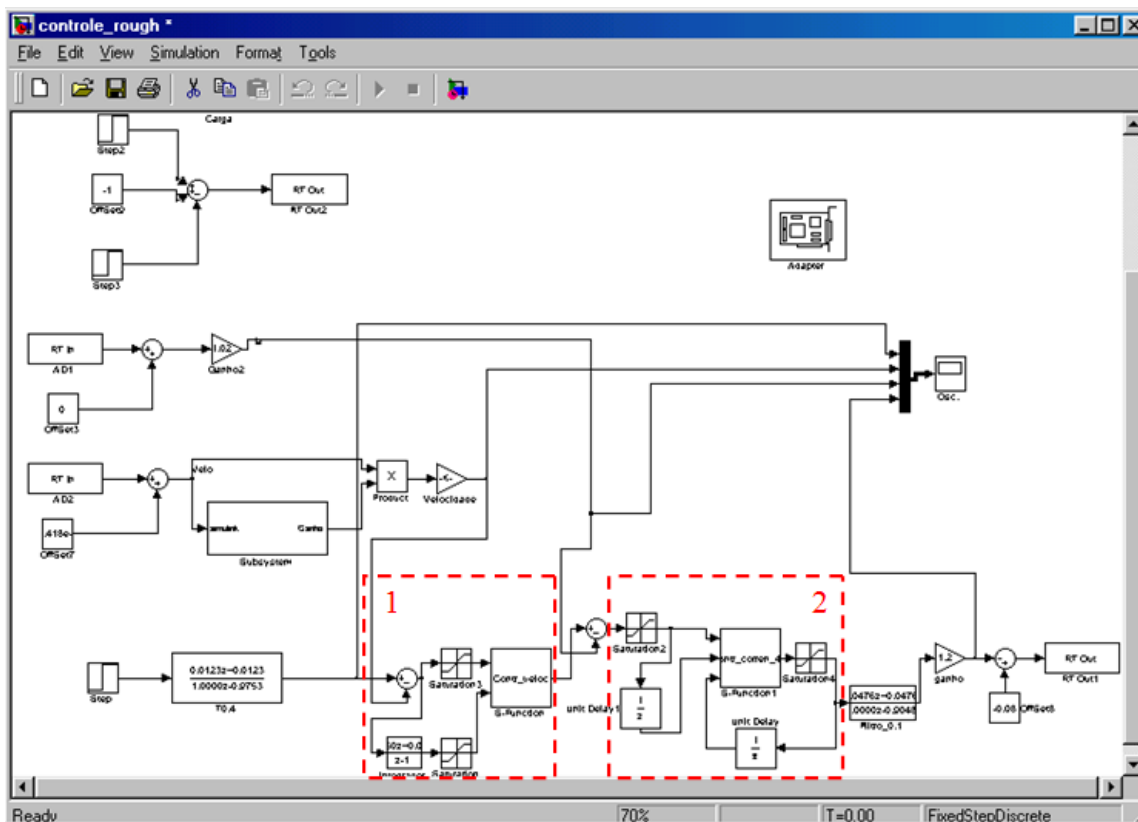


Figura B1- Diagrama para o controlador aproximado

B.3 – S-Function do Regulador Aproximado de Velocidade

Para o controlador aproximado de velocidade, observou-se por meio de simulações que algumas regras geradas pelo software ROSETTA não seriam usadas, então foi reduzido de vinte e cinco para apenas cinco regras. A seguir é apresentado o algoritmo em C para o controlador aproximado de velocidade.

```

/*****
File : Contr_veloc.c

                CONTROLADOR DE VELOCIDADE
                Projeto de um Controlador Aproximado
                usando a técnica de ROUGHS SETS

                Ulisses Pessin Camatta
Obs.: usar a função " mex -g nome da função.c " para gerar o
      arquivo necessário para ligar o simulink e S-function
      *****/

#define S_FUNCTION_NAME  Contr_veloc // Declara-se o nome da função
com o mesmo do arquivo
#define S_FUNCTION_LEVEL 2

#include "simstruc.h"

/*  Function: mdlInitializeSizes_____
 *  Abstract:
 *  Setup sizes of the various vectors.
 */

static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumSFcnParams(S, 0);
    if (ssGetNumSFcnParams(S) != ssGetSFcnParamsCount(S)) {
        return; /* Parameter mismatch will be reported by Simulink */
    }

    if (!ssSetNumInputPorts(S, 2)) return; // numero de portas de
entrada
    ssSetInputPortWidth(S, 0, DYNAMICALLY_SIZED);
    ssSetInputPortWidth(S, 1, DYNAMICALLY_SIZED);
    //ssSetInputPortWidth(S, 2, DYNAMICALLY_SIZED);

    ssSetInputPortDirectFeedThrough(S, 0, 1); // habilita a entrada da
porta 1
    ssSetInputPortDirectFeedThrough(S, 1, 1); // habilita a entrada da
porta 2
    //ssSetInputPortDirectFeedThrough(S, 2, 1); // habilita a entrada
da porta 3

    if (!ssSetNumOutputPorts(S,1)) return; //// numero de portas de
saída
    ssSetOutputPortWidth(S, 0, DYNAMICALLY_SIZED);

    ssSetNumSampleTimes(S, 1); // numero de tempos de amostragem
existente no bloco de função

    //ssSetOptions(S,
                //SS_OPTION_WORKS_WITH_CODE_REUSE |
                //SS_OPTION_EXCEPTION_FREE_CODE |
                //SS_OPTION_USE_TLC_WITH_ACCELERATOR);

```

```

}

/* Function: mdlInitializeSampleTimes
-----
* Abstract:
*   Specify that we inherit our sample time from the driving block.
*/
static void mdlInitializeSampleTimes(SimStruct *S)
{
    ssSetSampleTime(S, 0, 0.01); // define o tipo e o periodo de
    amostargem
    ssSetOffsetTime(S, 0, 0.0);

    //ssSetModelReferenceSampleTimeDefaultInheritance(S);
}

/* Function: mdlOutputs
*/

static void mdlOutputs(SimStruct *S, int_T tid)
{
    int_T i;
    InputRealPtrsType u = ssGetInputPortRealSignalPtrs(S,0);
    real_T *y = ssGetOutputPortRealSignal(S,0);
    int_T width = ssGetOutputPortWidth(S,0);

    for (i=0; i<width; i++)
    {
/*
*****
**
*           CONTROLADOR DE VELOCIDADE
*
*           ***** AS REGAS SAO POSTAS AQUI *****
*
*****
**
*/
        float x1,x2,x3; // Declarando as variaveis de entrada x1->erro
x2-> integral do erro
        float x1_min1,x1_max1, x1_min2,x1_max2,x1_min3,x1_max3,
x1_min4,x1_max4,x1_min5,x1_max5; // declaracao dos limites superior e
inferior de x1
        float x2_min3,x2_max3; //x2_min1,x2_max1,
x2_min2,x2_max2,x2_min3,x2_max3, x2_min4,x2_max4,x2_min5,x2_max5; //
declaracao dos limites superior e inferior de x2
        float y1; // saida auxiliar usada para os calculos

x1=*u[i,0]; // erro de velocidade
x2=*u[i,1]; // integral do erro

//REGRAS

y1=0;

```

```

// Variaveis Discretizadas, 5 semi retas de x1 e x2
// X1-----
x1_min1= -1.0100;   x1_max1= -0.5901; //x1_min1= -0.9924;
x1_min2= -0.5901;   x1_max2= -0.2382;
x1_min3= -0.2382;   x1_max3=  0.2349;
x1_min4=  0.2349;   x1_max4=  0.6083;
x1_min5=  0.6083;   x1_max5=  1.0100;   //x1_max5=  0.9995;
// X2-----
// x2_min1= -1.0100;   x2_max1= -0.6230; //x2_min1= -0.9880;
// x2_min2= -0.6230;   x2_max2= -0.2052;
x2_min3= -0.2052;   x2_max3=  0.2037;
// x2_min4=  0.2037;   x2_max4=  0.6152;
// x2_min5=  0.6152;   x2_max5=  1.0100;   //x2_max5=  0.9937;

// REGRA 1
if ((x1>=x1_min5 && x1<=x1_max5 ) && (x2>=x2_min3 &&
x2<=x2_max3 ))

//          y_vet=[1.0162    3.9332]; // Vetor "y" para
interpolação
y1 = 1.0162+((3.9332-1.0162)/2)*((x1-x1_min5)/(x1_max5 - x1_min5
)+(x2-x2_min3)/(x2_max3 - x2_min3)); //FIM DO IF
//_____x1_____
_____x2_____

// REGRA 13
if ((x1>=x1_min4 && x1<=x1_max4 ) && (x2>=x2_min3 &&
x2<=x2_max3 ))

//          y_vet=[0.1558    2.3557]; // Vetor "y" para
interpolação
y1 = 0.1558+((2.3557-0.1558)/2)*((x1-x1_min4)/(x1_max4 - x1_min4
)+(x2-x2_min3)/(x2_max3 - x2_min3)); //FIM DO IF
//_____x1_____
_____x2_____

// REGRA 17
if ((x1>=x1_min1 && x1<=x1_max1 ) && (x2>=x2_min3 &&
x2<=x2_max3 ))

//          y_vet=[-3.9378    -0.6848]; // Vetor "y" para
interpolação
y1 = -3.9378+((-0.6848-(-3.9378))/2)*((x1-x1_min1)/(x1_max1 - x1_min1
)+(x2-x2_min3)/(x2_max3 - x2_min3)); //FIM DO IF
//_____x1_____
_____x2_____

// REGRA 20
if ((x1>=x1_min3 && x1<=x1_max3 ) && (x2>=x2_min3 &&
x2<=x2_max3 ))

//          y_vet=[-1.5188    1.7013]; // Vetor "y" para
interpolação

```

```

y1 = -1.5188+((1.7013-(-1.5188))/2)*((x1-x1_min3)/(x1_max3 - x1_min3
)+(x2-x2_min3)/(x2_max3 - x2_min3)); //FIM DO IF
//_____x1_____
_____x2_____

// REGRA 25
    if ((x1>=x1_min2 && x1<=x1_max2 ) && (x2>=x2_min3 &&
x2<=x2_max3 ))

        // y_vet=[-2.4055    0.5309]; // Vetor "y" para
interpolação
y1 = -2.4055 +(( 0.5309-(-2.4055 )/2)*((x1-x1_min2)/(x1_max2 -
x1_min2)+(x2-x2_min3)/(x2_max3 - x2_min3)); //FIM DO IF
//_____x1_____
_____x2_____

// FIM DAS REGRAS-----

*y++=y1;

    }
}

/* Function: mdlTerminate
=====
* Abstract:
*   No termination needed, but we are required to have this routine.
*/
static void mdlTerminate(SimStruct *S)
{
}

#ifdef MATLAB_MEX_FILE    /* Is this file being compiled as a MEX-
file? */
#include "simulink.c"    /* MEX-file interface mechanism */
#else
#include "cg_sfund.h"    /* Code generation registration function */
#endif

```

B.4 – S-Function do Regulador de Corrente Aproximado

A seguir é apresentado o algoritmo em C para o controlador aproximado de corrente usado no bloco S-Function.

```
/*
*****
** File : Contr_corren_46.c
*****

                CONTROLADOR DE CORRNETE
    Projeto de um controlador aproximado usando a técnica de ROUGH S
SETS

                Ulisses Pessin Camatta

    Obs.: usar a função " mex -g nome da função.c " para gerar o
          arquivo necessário para ligar o simulink e S-function
*****
*/

#define S_FUNCTION_NAME  Contr_corren_46 // Declara-se o nome da
funcao com o mesmo do arquivo
#define S_FUNCTION_LEVEL 2

#include "simstruc.h"

/* Function: mdlInitializeSizes_____
 * Abstract:
 * Setup sizes of the various vectors.
 */

static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumSFcnParams(S, 0);
    if (ssGetNumSFcnParams(S) != ssGetSFcnParamsCount(S)) {
        return; /* Parameter mismatch will be reported by Simulink */
    }

    if (!ssSetNumInputPorts(S, 3)) return; // numero de portas de
entrada
    ssSetInputPortWidth(S, 0, DYNAMICALLY_SIZED);
    ssSetInputPortWidth(S, 1, DYNAMICALLY_SIZED);
    ssSetInputPortWidth(S, 2, DYNAMICALLY_SIZED);

    ssSetInputPortDirectFeedThrough(S, 0, 1); // habilita a entrada da
porta 1
    ssSetInputPortDirectFeedThrough(S, 1, 1); // habilita a entrada da
porta 2
    ssSetInputPortDirectFeedThrough(S, 2, 1); // habilita a entrada da
porta 3

    if (!ssSetNumOutputPorts(S,1)) return; //// numero de portas de
saida
    ssSetOutputPortWidth(S, 0, DYNAMICALLY_SIZED);
}
```

```

    ssSetNumSampleTimes(S, 1); // numero de tempos de amostagem
    existente no bloco de funcao

    //ssSetOptions(S,
        //SS_OPTION_WORKS_WITH_CODE_REUSE |
        //SS_OPTION_EXCEPTION_FREE_CODE |
        //SS_OPTION_USE_TLC_WITH_ACCELERATOR);
}

/* Function: mdlInitializeSampleTimes
-----
* Abstract:
*   Specifiy that we inherit our sample time from the driving block.
*/
static void mdlInitializeSampleTimes(SimStruct *S)
{

    ssSetSampleTime(S, 0, 0.01); // define o tipo e o periodo de
    amostagem
    ssSetOffsetTime(S, 0, 0.0);

    //ssSetModelReferenceSampleTimeDefaultInheritance(S);
}

/* Function: mdlOutputs
-----
* Abstract:
*   Funcao onde eh carrega as regas do ROUGHS SETS
*/

static void mdlOutputs(SimStruct *S, int_T tid)
{
    int_T          i;
    InputRealPtrsType u = ssGetInputPortRealSignalPtrs(S,0);
    real_T         *y    = ssGetOutputPortRealSignal(S,0);
    int_T          width = ssGetOutputPortWidth(S,0);

    for (i=0; i<width; i++)
    {
        /*
        *****
        **
        *                               CONTROLADOR DE CORRENTE
        *
        *          ***** AS REGAS SAO POSTAS AQUI *****
        *
        *****
        **
        */
        float x1,x2,x3; // Declarando as variaveis de entrada x1-
        >erro(n) de corrente x2-> erro(n-1) de corrente x3->Saida(n-1)
        float x1_min1,x1_max1, x1_min2,x1_max2,x1_min3,x1_max3,
        x1_min4,x1_max4; // declaracao dos limites superior e inferior de x1

```

```

float x2_min1,x2_max1, x2_min2,x2_max2,x2_min3,x2_max3,
x2_min4,x2_max4; // declaracao dos limites superior e inferior de x2
float x3_min1,x3_max1, x3_min2,x3_max2,x3_min3,x3_max3,
x3_min4,x3_max4; // declaracao dos limites superior e inferior de x3
float y1; // saida auxiliar usada para os calculos
int cot_if;

x1=*u[i,0]; // erro(n) de corrente
x2=*u[i,1]; // erro(n-1) de corrente
x3=*u[i,2]; // Saída(n-1)

//REGRAS
y1=0;cot_if=0;
// Variaveis Discretizadas, 3 semi retas de x1 , x2 , x3
// X1-----ERRO(n)-----
x1_min1= -1.0100; x1_max1= -0.5340;
x1_min2= -0.5340; x1_max2= -0.0108;
x1_min3= -0.0108; x1_max3= 0.4789;
x1_min4= 0.4789; x1_max4= 1.0100;

// X2-----ERRO(n-1)-----
x2_min1= -1.0100; x2_max1= -0.5340;
x2_min2= -0.5340; x2_max2= -0.0090;
x2_min3= -0.0090; x2_max3= 0.4811;
x2_min4= 0.4811; x2_max4= 1.0100;

// X3-----y(n-1)-----
x3_min1= -0.0100; x3_max1= 0.2456;
x3_min2= 0.2456; x3_max2= 0.4831;
x3_min3= 0.4831; x3_max3= 0.7259;
x3_min4= 0.7259; x3_max4= 1.0100;

// Interpolacao de X1, X2 e X3 em Y

// INICIO REGRA
*****
*****

// REGRA 1
if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min1 & x3<=x3_max1 ))
{
y1=-0.3593;
cot_if=cot_if + 1;
} //FIM DO IF

// REGRA 2
if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min1 & x3<=x3_max1 ))
{
//y_vet=[0.2965 0.5317]; // Vetor "y" para
interpolacao
y1 = 0.2965 + ((0.5317 - 0.2965)/3)*((x1-x1_min4)/(x1_max4 - x1_min4
) + (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min1)/(x3_max1 -
x3_min1));

```

```

//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 3
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min2 & x3<=x3_max2 ))
    {
y1=0.0649;
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 4
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min1 & x3<=x3_max1 ))

    {
y1=0.2040;
cot_if=cot_if + 1;
    } //FIM DO IF

// REGRA 5
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min3 & x3<=x3_max3 ))

    {
        //y_vet=[0.2845 0.6259]; // Vetor "y" para
interpolacao
y1 = 0.2845 + ((0.6259- 0.2845)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min4 )/(x2_max4 - x2_min4) + (x3-x3_min3 )/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 6
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min2 & x3<=x3_max2 ))

    {
        //y_vet=[-0.0684 0.3406]; // Vetor "y" para
interpolacao
y1 = -0.0684 + ((0.3406-(-0.0684))/3)*((x1-x1_min2)/(x1_max2 -
x1_min2 ) + (x2-x2_min3 )/(x2_max3 - x2_min3) + (x3-x3_min2
)/(x3_max2 - x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 7
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min1 & x3<=x3_max1 ))

    {
        //y_vet=[0.2274 0.6094]; // Vetor "y" para
interpolacao
y1 = 0.2274 + ((0.6094-0.2274)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min2 )/(x2_max2 - x2_min2) + (x3-x3_min1 )/(x3_max1 -
x3_min1));

```

```

//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 8
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min1 & x3<=x3_max1 ))

        {
y1=-0.1786;
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 9
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min1 & x3<=x3_max1 ))

        {
//y_vet=[-0.2203  0.2434]; // Vetor "y" para
interpolacao
y1 = -0.2203 + ((0.2434-(-0.2203))/3)*((x1-x1_min1)/(x1_max1 -
x1_min1 ) + (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min1
)/(x3_max1 - x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 10
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min1 & x3<=x3_max1 ))

        {
//y_vet=[0.5444  0.9443]; // Vetor "y" para
interpolacao
y1 = 0.5444 + ((0.9443-0.5444)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min1)/(x3_max1 -
x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 11
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {
// y_vet=[0.7171  1.0928]; // Vetor "y" para
interpolacao
y1 = 0.7171 + ((1.0928-0.7171)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 12
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min4 & x3<=x3_max4 ))

```

```

                {          //y_vet=[0.5044    0.8999]; // Vetor "y" para
interpolacao
y1 = 0.5044 + ((0.8999-0.5044)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min4)/(x2_max4 - x2_min4) + (x3-x3_min4)/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 13
        if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min3 & x3<=x3_max3 ))

                {          //y_vet=[0.4457    0.7635]; // Vetor "y" para
interpolacao
y1 = 0.4457 + ((0.7635-0.4457)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 14
        if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min3 & x3<=x3_max3 ))

                {          // y_vet=[0.0319    0.3394]; // Vetor "y" para
interpolacao
y1 = 0.0319 + ((0.3394-0.0319)/3)*((x1-x1_min1)/(x1_max1 - x1_min1 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 15
        if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min1 & x3<=x3_max1 ))

                {          // y_vet=[0.3387    0.7895]; // Vetor "y" para
interpolacao
y1 = 0.3387 + ((0.7895-0.3387)/3)*((x1-x1_min3)/(x1_max3 - x1_min3
) + (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min1)/(x3_max1 -
x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 16
        if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min4 & x3<=x3_max4 ))

                {          //y_vet=[0.1710 0.4391]; // Vetor "y" para
interpolacao
y1 = 0.1710 + ((0.4391-0.1710)/3)*((x1-x1_min1)/(x1_max1 - x1_min1
) + (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min4)/(x3_max4 -
x3_min4));

```

```

//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 17
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {          //y_vet=[0.6271 0.7334]; // Vetor "y" para
interpolacao
y1 = 0.6271 + ((0.7334 -0.6271)/3)*((x1-x1_min3)/(x1_max3 - x1_min3
) + (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 18
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {          //y_vet=[0.2302 0.5811]; // Vetor "y" para
interpolacao
y1 = 0.2302 + ((0.5811-0.2302)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 19
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {          //y_vet=[0.1325 0.3402]; // Vetor "y" para
interpolacao
y1 = 0.1325 + ((0.3402-0.1325)/3)*((x1-x1_min1)/(x1_max1 - x1_min1 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 20
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {          // y_vet=[0.6314 0.8725]; // Vetor "y" para
interpolacao
y1 = 0.6314 + ((0.8725-0.6314)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF

```

```

// REGRA 21
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {           // y_vet=[0.1340    0.3246]; // Vetor "y" para
interpolacao
y1 = 0.1340 + ((0.3246 -0.1340)/3)*((x1-x1_min2)/(x1_max2 - x1_min2
) + (x2-x2_min4)/(x2_max4 - x2_min4) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 22
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {           //y_vet=[0.3535    0.7895]; // Vetor "y" para
interpolacao
y1 = 0.3535 + ((0.7895-0.3535)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 23
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {           //y_vet=[-0.0591    0.0455]; // Vetor "y" para
interpolacao
y1 = -0.0591 + ((0.0455-(-0.0591))/3)*((x1-x1_min1)/(x1_max1 -
x1_min1 ) + (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min2
)/(x3_max2 - x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 24
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min1 & x3<=x3_max1 ))

        {           //y_vet=[0.5792 0.8355]; // Vetor "y" para
interpolacao
y1 = 0.5792 + ((0.8355 -0.5792)/3)*((x1-x1_min4)/(x1_max4 - x1_min4
) + (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min1)/(x3_max1 -
x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 25
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min4 & x3<=x3_max4 ))

```

```

                {           //y_vet=[0.9107 1.0755]; // Vetor "y" para
interpolacao
y1 = 0.9107 + ((1.0755-0.9107)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min4)/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 26
        if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min4 & x3<=x3_max4 ))

                {           //y_vet=[0.1030 0.4495]; // Vetor "y" para
interpolacao
y1 = 0.1030 + ((0.4495 -0.1030)/3)*((x1-x1_min1)/(x1_max1 - x1_min1
) + (x2-x2_min4)/(x2_max4 - x2_min4) + (x3-x3_min4)/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 27
        if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min2 & x3<=x3_max2 ))

                {           //y_vet=[0.1210 0.2943]; // Vetor "y" para
interpolacao
y1 = 0.1210 + ((0.2943-0.1210)/3)*((x1-x1_min1)/(x1_max1 - x1_min1 )
+ (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 28
        if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min2 & x3<=x3_max2 ))

                {           //y_vet=[0.7037 1.1030]; // Vetor "y" para
interpolacao
y1 = 0.7037 + ((1.1030-0.7037)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
        } //FIM DO IF
// REGRA 29
        if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min4 & x3<=x3_max4 ))

                {           //y_vet=[0.6453 1.1649]; // Vetor "y" para
interpolacao
y1 = 0.6453 + ((1.1649-0.6453)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min4)/(x2_max4 - x2_min4) + (x3-x3_min4)/(x3_max4 -
x3_min4));

```

```

//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 30
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min4 & x3<=x3_max4 ))

        {          //y_vet=[0.6050 0.8138]; // Vetor "y" para
interpolacao
y1 = 0.6050 + ((0.8138-0.6050)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min3 )/(x2_max3 - x2_min3) + (x3-x3_min4 )/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 31
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {          //y_vet=[0.2979 0.6460]; // Vetor "y" para
interpolacao
y1 = 0.2979 + ((0.6460-0.2979)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min1 )/(x2_max1 - x2_min1) + (x3-x3_min2 )/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 32
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {          //y_vet=[0.5760 0.9347]; // Vetor "y" para
interpolacao
y1 = 0.5760 + ((0.9347 -0.5760 )/3)*((x1-x1_min3)/(x1_max3 -
x1_min3 ) + (x2-x2_min2 )/(x2_max2 - x2_min2) + (x3-x3_min3
)/(x3_max3 - x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 33
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min4 & x3<=x3_max4 ))

        {          // y_vet=[0.4399 0.6795]; // Vetor "y" para
interpolacao
y1 = 0.4399 + ((0.6795-0.4399)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min3 )/(x2_max3 - x2_min3) + (x3-x3_min4 )/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF

```

```

// REGRA 34
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min4 & x3<=x3_max4 ))

        {           // y_vet=[0.2675    0.7263]; // Vetor "y" para
interpolacao
y1 = 0.2675 + ((0.7263-0.2675 )/3)*((x1-x1_min2)/(x1_max2 - x1_min2
) + (x2-x2_min4)/(x2_max4 - x2_min4) + (x3-x3_min4)/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 35
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min4 & x3<=x3_max4 ))

        {           //y_vet=[ 0.6362 0.7384]; // Vetor "y" para
interpolacao
y1 = 0.6362 + ((0.7384-0.6362)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min4)/(x3_max4 -
x3_min4));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 36
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min1 & x3<=x3_max1 ))

        {           //y_vet=[0.0953    0.4792]; // Vetor "y" para
interpolacao
y1 = 0.0953 + ((0.4792-0.0953)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min1)/(x3_max1 -
x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 37
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {           //y_vet=[-0.1106    0.2478]; // Vetor "y" para
interpolacao
y1 = -0.1106 + (( 0.2478-(-0.1106))/3)*((x1-x1_min1)/(x1_max1 -
x1_min1 ) + (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min2
)/(x3_max2 - x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 38
    if ((x1>=x1_min3 & x1<=x1_max3 ) & (x2>=x2_min3 & x2<=x2_max3
)& (x3>=x3_min2 & x3<=x3_max2 ))

```

```

        {           //y_vet=[0.4250 0.6188]; // Vetor "y" para
interpolacao
y1 = 0.4250 + ((0.6188-0.4250)/3)*((x1-x1_min3)/(x1_max3 - x1_min3 )
+ (x2-x2_min3)/(x2_max3 - x2_min3) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 39
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min1 & x3<=x3_max1 ))

        {           //y_vet=[0.0656 0.3148]; // Vetor "y" para
interpolacao
y1 = 0.0656 + ((0.3148-0.0656)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min1)/(x3_max1 -
x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 40
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min2 & x3<=x3_max2 ))

        {           //y_vet=[0.1217 0.4861]; // Vetor "y" para
interpolacao
y1 = 0.1217 + ((0.4861-0.1217)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 41
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {           //y_vet=[0.6041 0.8727]; // Vetor "y" para
interpolacao
y1 = 0.6041 + ((0.8727-0.6041)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min4)/(x2_max4 - x2_min4) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 42
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min4 & x2<=x2_max4
)& (x3>=x3_min3 & x3<=x3_max3 ))

        {
y1=0.0530;
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 43

```

```

        if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min3 & x3<=x3_max3 ))

            {          // y_vet=[0.8071  1.1137]; // Vetor "y" para
interpolacao
y1 = 0.8071 + ((1.1137-0.8071)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 44
    if ((x1>=x1_min4 & x1<=x1_max4 ) & (x2>=x2_min1 & x2<=x2_max1
)& (x3>=x3_min2 & x3<=x3_max2 ))

            {          //y_vet=[0.8678 0.9148]; // Vetor "y" para
interpolacao
y1 = 0.8678 + ((0.9148-0.8678)/3)*((x1-x1_min4)/(x1_max4 - x1_min4 )
+ (x2-x2_min1)/(x2_max1 - x2_min1) + (x3-x3_min2)/(x3_max2 -
x3_min2));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 45
    if ((x1>=x1_min1 & x1<=x1_max1 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min1 & x3<=x3_max1 ))

            {          //y_vet=[-0.2124  0.0375]; // Vetor "y" para
interpolacao
y1 = -0.2124 + (( 0.0375-(-0.2124))/3)*((x1-x1_min1)/(x1_max1 -
x1_min1 ) + (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min1
)/(x3_max1 - x3_min1));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF
// REGRA 46
    if ((x1>=x1_min2 & x1<=x1_max2 ) & (x2>=x2_min2 & x2<=x2_max2
)& (x3>=x3_min3 & x3<=x3_max3 ))

            {          // y_vet=[0.3940 0.6884]; // Vetor "y" para
interpolacao
y1 = 0.3940 + ((0.6884-0.3940)/3)*((x1-x1_min2)/(x1_max2 - x1_min2 )
+ (x2-x2_min2)/(x2_max2 - x2_min2) + (x3-x3_min3)/(x3_max3 -
x3_min3));
//_____x1_____
//_____x2_____x3_____
cot_if=cot_if + 1;
    } //FIM DO IF

// FIM DAS REGRAS
*****
*****

```



```
//REGRA EXTRA condições de partida

    if ((cot_if ==0) | (cot_if >=2))
    {
        if (x1>0) {y1=x3+0.02;}
        else {y1=x3-0.02;}
    }

*y++=y1;
    }
}

/* Function: mdlTerminate
=====
* Abstract:
*   No termination needed, but we are required to have this routine.
*/
static void mdlTerminate(SimStruct *S)
{
}

#ifdef MATLAB_MEX_FILE    /* Is this file being compiled as a MEX-
file? */
#include "simulink.c"      /* MEX-file interface mechanism */
#else
#include "cg_sfun.h"      /* Code generation registration function */
#endif
```

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Aguirre, L. A.; “Introdução à Identificação de Sistemas: Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais.” 2ª edição, 2000, UFMG, Belo Horizonte, MG.
- [2] Aranda, G. E. I.; “Projeto e implementação de um acionamento controlado analógico e digital para máquina de corrente contínua utilizando chopper de quatro quadrantes.” 2000. 0 f. Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal de Itajubá, Conselho Nacional de Desenvolvimento Científico e Tecnológico. Orientador: Rezek, A. J. J.
- [3] Assis, W. O.; “Análise comparativa de estratégia de controle para acionamento industriais utilizando CHOPPER.” 1997. 0 f. Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal de Itajubá, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior. Orientador: Rezek, A. J. J.
- [4] Aydin, O.; Akdag, A.; Stefanutti, P.; Hugo, N.; “Optimum controller design for a multilevel AC-DC converter system.” Applied Power Electronics Conference and Exposition, 2005. APEC 2005. Twentieth Annual IEEE Volume 3, Issue, 6-10 March 2005 Page(s):1660 - 1666 Vol. 3 Digital Object Identifier 10.1109/APEC.2005.1453262
- [5] Bilgin, H.F.; Kose, K.N.; Zenginobuz, G.; Ermis, M.; Cadirci, E.N.I.; Kose, H.; “A unity-power-factor buck-type PWM rectifier for medium/high-power DC motor drive applications.” Industry Applications, IEEE Transactions on Volume 38, Issue 5, Sept.-Oct. 2002 Page(s):1412 – 1425 Digital Object Identifier 10.1109/TIA.2002.803005
- [6] Bose, B. K.; “Technology trends in microcomputer control of electrical machines.” Industrial Electronics, IEEE Transactions on Volume 35, Issue 1, Feb. 1988 Page(s):160 - 177 Digital Object Identifier 10.1109/41.3080.
- [7] Fröhr, F.; Ortttenburger, F.; “Introduccion al control electronic”, Marcombo S.A., Siemens Aktiengesellschaft, Barcelona, 1986.
- [8] Gurbuz, F.; Akpınar, E.; “Simulation of pulse width modulated (PWM) chopper drives using z-transformation.” Electrotechnical Conference, 1998. MELECON 98., 9th Mediterranean Volume 2, 18-20 May 1998 Page(s):1255 - 1259 vol.2 Digital Object Identifier 10.1109/MELCON.1998.699436

-
- [9] Kaura, V.; Blasko, V.; “Operation of a phase locked loop system under distorted utility conditions.” Industry Applications, IEEE Transactions on Volume 33, Issue 1, Jan.-Feb. 1997 Page(s):58 – 63 Digital Object Identifier 10.1109/28.567077
- [10] Komorowski J.; Polkowski L.; Skowron A.; “Rough Sets: A Tutorial.” <http://folli.loria.fr/cds/1999/library/pdf/skowron.pdf>
<http://chc60.fgcu.edu/images/articles/RoughSetsTutorial.pdf>
- [11] KOSOW, Irving I.; “Máquinas Eléctricas e Transformadores.” 15. ed. Globo, 1996
- [12] OGATA, K.; “Engenharia de controle moderno.” São Paulo (SP), Brasil: Pearson do Brasil, 2003.
- [13] Pawlak, Z.; “Rough Sets.” International Journal of Information and Computer Sciences, Vol. 11, No. 5, pp. 341-356, 1982.
- [14] Pawlak, Z.; “Rough Real Function.” Bull. PAS, Tech. Ser. Vol. 35, No. 5-6, pp. 249-251
<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.50.6501>
- [15] Pawlak, Z. “Rough real functions and rough controllers.” Rough sets and data mining: Analysis of imprecise data, pp. 139-147. Boston : Kluwer Academic Publishers, 1997.
<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.6929>
- [16] Pawlak, Z., “Rough Sets: Theoretical Aspects of Reasoning about Data”, Kluwer Academic Publishers, Dordrecht, 1991.
- [17] Pila, A. D.; “Seleção de Atributos Relevantes para Aprendizagem de Máquina Utilizando a Abordagem de Rough Sets.” 2001. 142 f. Dissertação (Mestrado em Ciências da Computação e Matemática Computacional) - Universidade de São Paulo, Fundação de Amparo à Pesquisa do Estado de São Paulo. Orientador: Monard, M. C.
- [18] Pinheiro, C. A. M., Gomide, F., Carpinteiro, O., Lopez, B. I. L., “Granular Synthesis of Rule-Based Models and Function Approximation using Rough Sets”, chapter in the book “Novel Developments in Granular Computing Applications for Advanced Human Reasoning and Soft Computation”, ed. JingTao Yao, IGI Global, Hershey, 2009.

-
- [19]Preitl, St.; Precup, R.E.; “An Extension of Tuning Relations after Symmetrical Optimum Method for PI and PID Controllers.” *Automatica* (Elsevier Science), 1999 35, 1731-1736
- [20]Salah El Din Zein El Din, A.; “PLC-Based Speed Control of DC Motor.” *Power Electronics and Motion Control Conference, 2006. IPEMC '06. CES/IEEE 5th International Volume 2*, 14-16 Aug. 2006 Page(s):1 - 6 Digital Object Identifier 10.1109/IPEMC.2006.297190
- [21]Stephan, R.M.; “A simple model for a thyristor-driven DC motor considering continuous and discontinuous current modes.” *Education, IEEE Transactions on Volume 34, Issue 4, Nov. 1991* Page(s):330 – 335 Digital Object Identifier 10.1109/13.102850
- [22]Vhasure, S.; Prakasha; Krishnan, T.; “Fast response digital speed and current regulators for thyristor fed reversible regenerative DC motor drives.” *Power Electronics, Drives and Energy Systems for Industrial Growth, 1996., Proceedings of the 1996 International Conference on Volume 2*, 8-11 Jan. 1996 Page(s):764 - 769 vol.2 Digital Object Identifier 10.1109/PEDES.1996.535875
- [23]Yun, G.; Yuanbin, H.; “Application of rough set theory on system modeling.” *Intelligent Control and Automation. WCICA 2004. Fifth World Congress on Volume 3*, 15-19 June 2004 Page(s):2352 - 2354 Vol.3 Digital Object Identifier 10.1109/WCICA.2004.1342014