
Universidade Estadual de Campinas
Faculdade de Engenharia Elétrica e de Computação
Departamento de Sistemas de Controle e Energia
DSCE-FEEC-UNICAMP

Algoritmo Preditivo Baseado em Modelo Aplicado ao Controle de Velocidade do Motor de Indução

Autor: Eudemario Souza de Santana
Orientador: Prof. Dr. Edson Bim

*Tese submetida à Faculdade de Engenharia Elétrica e de Computação da
Universidade Estadual de Campinas, como parte dos requisitos exigidos
para obtenção do título de
Doutor em Engenharia Elétrica*

Comissão Julgadora:
Prof. Dr. Edson Bim - DSCE/FEEC/UNICAMP
Prof. Dr. Antonio Augusto Rodrigues Coelho - UFSC
Dr. José Alberto Torrico Altuna - Commodity Systems
Prof. Dr. Wagner Caradori do Amaral - DCA/FEEC/UNICAMP
Prof. Dr. Marconi Kolm Madrid - DSCE/FEEC/UNICAMP
Prof. Dr. Akebo Yamakami - DT/FEEC/UNICAMP

Campinas, 10 de Abril de 2007

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

Sa59a Santana, Eudemario Souza de
Algoritmo preditivo baseado em modelo aplicado ao controle de velocidade do motor de indução / Eudemario Souza de Santana. --Campinas, SP: [s.n.], 2007.

Orientador: Edson Bim
Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Maquinas elétrica de indução. 2. Acionamento elétrico. 3. Controle preditivo. 4. Kalman, Filtragem de. I. Bim, Edson. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Speed control of induction motor using predictive algorithm

Palavras-chave em Inglês: Induction motor, Model based predictive control, MBPC, Kalman filter

Área de concentração: Energia Elétrica

Titulação: Doutor em Engenharia Elétrica

Banca examinadora: Antonio Augusto Rodrigues Coelho, José Alberto Torrico Altuna, Wagner Caradori do Amaral, Marconi Kolm Madrid, Akebo Yamakami

Data da defesa: 10/04/2007

Programa de Pós-Graduação: Engenharia Elétrica

Algoritmo Preditivo Baseado em Modelo Aplicado ao Controle de Velocidade do Motor de Indução

Autor: Eudemario Souza de Santana
Orientador: Prof. Dr. Edson Bim

*Tese submetida à Faculdade de Engenharia Elétrica e de Computação da
Universidade Estadual de Campinas, como parte dos requisitos exigidos
para obtenção do título de
Doutor em Engenharia Elétrica*

Campinas, 10 de Abril de 2007

As cabeças banais simplesmente não podem se decidir a escrever do modo como pensam, porque pressentem que, nesse caso, o resultado teria um aspecto muito simplório. Mas já seria alguma coisa. Se eles apenas se dedicassem com honestidade à sua obra e simplesmente quisessem comunicar o pouco e usual que de fato pensaram, da maneira como pensaram, seriam legíveis e até mesmo instrutivos dentro de sua esfera própria. Só que, em vez disso, esforçam-se para dar a impressão de ter pensado mais e com mais profundidade do que o fizeram realmente.

Estas pessoas apresentam o que têm a dizer em fórmulas forçadas, difíceis, com neologismos e frases prolixas que giram em torno dos pensamentos e os escondem.

Oscilam entre o esforço de comunicar e o de esconder o que pensaram.

Schopenhauer 1788-1860. A arte de escrever, L&PM POCKET

Resumo

Esta tese trata do emprego do controle preditivo baseado no modelo (MBPC-Model Based Predictive Control) no acionamento do motor de indução trifásico, para controle de fluxo de rotor e velocidade. A estratégia MBPC baseia-se na minimização do erro entre as referências futuras e a predição do modelo, para gerar os sinais de controle. Nesta tese, o motor de indução é descrito por espaço de estados e, diferentemente, do MBPC não linear, que emprega algoritmos de busca para determinar os sinais de controle, a estratégia escolhida faz linearizações sucessivas. Assim sendo, a cada ciclo gera-se a lei de controle, sendo que esta é dada por uma equação algébrica. São necessários ao controlador preditivo o conhecimento das tensões de terminal do estator e das seguintes variáveis de estado: corrente de estator, fluxo de rotor e velocidade de eixo. Para a estimação dos estados é empregado o filtro de Kalman estendido. O torque de carga é tratado como uma perturbação e sua magnitude é obtida por duas abordagens: pela equação eletromecânica e pelo filtro de Kalman estendido. Resultados de simulação computacional e experimentais validam a proposta.

Palavras-chave: Motor de indução, controle preditivo baseado no modelo, acionamento elétrico e MBPC.

Abstract

This thesis presents the results concerning the control of rotor flux and speed of the induction motor using MBPC strategy, which is based on the error minimization between the future set point and model prediction, resulting in control signals. In the case studied in this thesis the motor model is described in space-state. The non linear MBPC employs search algorithms to find the control signals, whereas the technique used in this thesis made successive linearizations on model; therefore in every control cycle a new algebraic control law is found. The predictive control needs to know the stator voltage and the following state variables: stator current, rotor flux and speed. In the order to estimate the states an extended Kalman filter is employed. The load torque is considered as a disturbance and its amplitude is obtained in two ways: by calculation via electromechanical equation and by estimation via Kalman filter. The proposal has been validated by simulations and experiments.

Keywords: Induction motor, Model based predictive control and MBPC.

Agradecimentos

Aos meus pais: Nilza e Edmundo.

Ao prof. Edson Bim por ter me orientado desde o mestrado e ter lançado o desafio do controle preditivo.

Ao prof. Wagner Amaral pela ajuda na teoria e nas simulações do MBPC.

À CAPES pela bolsa, à FAPESP pela estrutura computacional e à FEEC/UNICAMP por fornecer tudo que precisava para minha formação na pós-graduação.

Aos colegas Hélio, Leandro, Rogério e Alfeu, pelo agradável, briguento, engraçado, bagunçado, ... convívio no laboratório.

Aos vizinhos de lab. Capovilla, Dudu e Tatá.

A André, Milton Filho, Marcelo Villalva, Leonardo Araújo, Marcos Lazarini, Torrico Altuna pelo auxílio técnico em muitas áreas distintas.

Aos amigos Arismar, Téogenes, Henrique, Max, Titi, George, B4, Nagato, Japa, Protásio, Adelauro, Luis Negão, Esaó, Taxista, Gabriel, Baco, Ananda, Janaísa, Ampas, Marcos, Duvier, Iara, Julião do Pará, Manga, Igor (meu neto), Dedeco, Léo Vaca, Daniel Barra, Leandro Barra e Ilton, entre outros mais que minha memória não permitiu citar, por terem tornado a minha estadia em Campinas mais agradável possível.

Ao amigo Braço de Radiola (vulgo Irênio) por ter me tolerado durante cinco anos na pensão do seu Ruy e na Casa do lado.

Ao amigo e moradores da Estação de Esqui: Buzola, Fêra, Senhor, Tio Guga, Pol e Danilo.

Sumário

Resumo	ii
Abstract	iii
Agradecimentos	iv
Lista de Figuras	viii
Lista de Símbolos	x
Acrônimos	xi
Capítulo 1 Introdução	1
1.1 Tema estudado	2
1.2 Organização do texto	3
Capítulo 2 Controle Preditivo Baseado em Modelo	5
2.1 Introdução	5
2.2 Fundamentos do MBPC	5
2.2.1 Influência dos parâmetros do MBPC	8
2.2.1.1 Esforço de controle	8
2.2.1.2 Horizonte inicial de predição	8
2.2.1.3 Horizonte final de predição	8
2.2.1.4 Horizonte de controle	8
2.2.1.5 Período do ciclo de controle	8
2.3 Algumas técnicas MBPC	9
2.3.1 Controle Preditivo Funcional	9
2.3.2 Controlador de Matriz Dinâmica	10
2.3.3 Controlador Preditivo Generalizado	11
2.3.4 Outras estratégias MBPC	12
2.4 MBPC empregado	14
2.4.1 Modelo da planta	14
2.4.1.1 Discretização	15
2.4.1.2 Linearização	15

2.4.2	Predição	15
2.4.3	Lei de controle	16
2.4.4	MBPC variacional	17
2.5	Conclusões	18
Capítulo 3	Controle do Motor de Indução	21
3.1	Introdução	21
3.2	Controle preditivo na área de acionamentos elétricos	23
3.3	MBPC aplicado ao acionamento elétrico	25
3.4	MBPC do motor de indução	26
3.4.1	Modelo e controle para o motor de indução	26
3.4.1.1	Linearização e discretização do modelo do motor de indução em espaços de estado	26
3.4.1.2	Diagrama esquemático do acionamento do MI com o MBPC	29
3.5	Conclusões	31
Capítulo 4	Resultados de Simulações	33
4.1	Introdução	33
4.2	Resultados para o MBPC variacional	33
4.2.1	Testes a vazio com diversos perfis de velocidade	34
4.2.1.1	Perfil de velocidade retangular	34
4.2.1.2	Perfil de velocidade trapezoidal	36
4.2.1.3	Perfil de velocidade triangular	37
4.2.1.4	Teste em baixa velocidade	38
4.2.2	Testes com carga	38
4.2.2.1	Teste com $T_c \propto \omega_r^2$ e com T_c calculado	38
4.2.2.2	Teste com degrau de carga nominal ao eixo do motor e com T_c calculado	40
4.2.2.3	Teste com $T_c \propto \omega_r^2$ e T_c estimado com o FKE	41
4.2.2.4	Teste com degrau de carga nominal e T_c estimado com o FKE	41
4.2.3	Teste de variação de fluxo	42
4.2.3.1	Redução de fluxo de rotor	42
4.2.4	Testes com mudança do esforço de controle e com filtro na referência	45
4.3	MBPC não variacional	47
4.4	Conclusões do capítulo	49
Capítulo 5	Resultados experimentais	51
5.1	Introdução	51
5.2	Descrição funcional do sistema implementado	51
5.2.1	Algoritmo de controle	54
5.3	Resultados experimentais	57

5.3.1	Testes a vazio	57
5.3.2	Teste com carga	62
5.4	Conclusões do capítulo	64
Capítulo 6	Conclusões e sugestões de trabalhos futuros	65
6.1	Conclusão	65
6.2	Sugestões para trabalhos futuros	66
6.2.1	MBPC	66
6.2.2	Estimador de estado	66
6.2.3	Acionamento	67
6.2.4	Implementação	67
Apêndice A	Filtro de Kalman Estendido	69
A.1	Introdução	69
A.2	Modelos do MI para estimação	69
A.3	Algoritmo do filtro de Kalman estendido	71
A.4	Algumas considerações adicionais	72
A.5	Resultados experimentais do FKE	74
A.6	Breve revisão dos estimadores de estados para MI	76
Apêndice B	Dados empregados	79
B.1	Dados do motor	79
Apêndice C	Lei de controle	81
Apêndice D	Programa	83
Referências bibliográficas		109

Lista de Figuras

2.1	Diagrama de blocos do MBPC.	7
2.2	Influência do horizonte de controle n_u	9
3.1	Diagrama de blocos do controle de velocidade e fluxo de rotor da máquina de indução com controladores PI.	22
3.2	Controle de corrente preditivo por histerese. Fronteira circular centrada da referência e as sete possíveis trajetórias do vetor espacial corrente de estator.	24
3.3	Diagrama de bloco representativo do acionamento proposto.	32
4.1	Teste com perfil de velocidade retangular de 600 rpm à -600 rpm com motor a vazio.	35
4.2	Teste com perfil de velocidade trapezoidal de 600 rpm à -600 rpm com motor a vazio.	36
4.3	Teste com perfil de velocidade triangular de 600 rpm à -600 rpm, com motor a vazio.	37
4.4	Teste com perfil de velocidade retangular de 30 rpm à 0 rpm com motor a vazio.	38
4.5	Teste com perfil de velocidade trapezoidal de 600 rpm à -600 rpm, com carga proporcional ao quadrado da velocidade.	39
4.6	Teste com perfil de velocidade retangular de 600 rpm à 0 rpm, com carga nominal (12,3 N.m) aplicada ao eixo no instante 0,1 s.	40
4.7	Teste com perfil de velocidade trapezoidal de 600 rpm à -600 rpm, com carga proporcional ao quadrado da velocidade.	41
4.8	Teste com perfil de velocidade retangular de 600 rpm à 0 rpm, com carga nominal (12,3 N.m) aplicada ao eixo no instante 0,1 s.	42
4.9	Variação de fluxo; de 0.565 Wb a 0,34 Wb.	44
4.10	Teste com perfil de velocidade retangular de 450 rpm à -450 rpm, com motor a vazio.	46
4.11	Teste com perfil de velocidade retangular de 450 rpm à -450 rpm com motor a vazio e referência filtrada.	47
4.12	Teste com perfil de velocidade retangular de 900 rpm à -900 rpm com motor a vazio.	48

4.13	Teste com perfil de velocidade retangular de 900 rpm à -900 rpm com motor a vazio.	49
5.1	Ilustração dos dispositivos empregados na bancada para realização dos experimentos.	53
5.2	Foto da bancada.	54
5.3	Configurações do DSP realizado na função <i>main</i>	55
5.4	Descrição da interrupção das medições, do FKE e da SV-PWM.	55
5.5	Interrupção do MBPC.	56
5.6	Reversão de 450 rpm a -450 rpm.	58
5.7	Variação em degrau de 0 a 300 rpm	59
5.8	Reversão de velocidade; 600 rpm a -600 rpm em 600 ms.	60
5.9	Reversão de velocidade; 150 rpm a -150 rpm em 120 ms.	60
5.10	Variação de 0 rpm a 450 rpm, com tempos de subida e descida de 300 ms.	61
5.11	Variação triangular de 450 rpm a -450 rpm, com tempos de subida e de descida de 600 ms.	61
5.12	Variação de fluxo de rotor do valor nominal a 50% do valor nominal em 300 ms.	62
5.13	Teste com carga de 10,36 N.m e velocidade igual à 600 rpm. Canais 1, 2 e 4 mostram a velocidade estimada, i_{sd} e i_{sq} . No canal 1, o valor zero Volt indica a velocidade de 600 rpm.	63
A.1	Diagrama de bocos do filtro de Kalman estendido.	72
A.2	Correntes reais e estimadas.	75
A.3	Componentes $\alpha - \beta$ estimadas do fluxo de rotor para 518 rpm.	76

Lista de Símbolos

- $\hat{y}(k + j/k)$ - Predição da saída j passos a frente, calculada no instante discreto k
 $w(k + j)$ - Referência futura
 λ - Parâmetro da função custo que pondera o esforço de controle no caso de haver apenas uma entrada
 W_u - Matriz que pondera o esforço de controle no caso de haver mais de uma entrada
 W_y - Matriz que pondera cada predição das saídas
 n_1 - Horizonte inicial de predição
 n_y - Horizonte final de predição
 n_u - Horizonte de controle
 $\bar{V}_{s\alpha\beta}$ - Vetor espacial de tensão de estator na referência estacionária
 \bar{V}_{sdq} - Vetor espacial de tensão de estator na referência síncrona
 $\bar{I}_{s\alpha\beta}$ - Vetor espacial de corrente de estator na referência estacionária
 \bar{I}_{sdq} - Vetor espacial de corrente de estator na referência síncrona
 $\bar{\Phi}_{r\alpha\beta}$ - Vetor espacial de fluxo de rotor na referência estacionária
 $\bar{\Phi}_{rdq}$ - Vetor espacial de fluxo de rotor na referência síncrona
 T_a - Tempo de amostragem
 V_{sd} e V_{sq} - Tensões de estator de eixo direto e de eixo em quadratura
 i_{sd} e i_{sq} - Correntes de estator de eixo direto e de eixo em quadratura
 Φ_{rd} e Φ_{rq} - Fluxo de rotor de eixo direto e de eixo em quadratura
 ω_r - Velocidade elétrica do rotor
 T_c - Torque de carga
 θ_{Φ_r} - Ângulo da posição espacial do fluxo de rotor

Acrônimos

AG - Algoritmos genéticos
CAD - Conversor analógico digital
CARIMA - Controller auto-regressive integrated moving-average
CDA - Conversor digital analógico
DMC - Dinamic matrix control
DSP - Digital signal processor
FKE - Filtro de Kalman estendido
FOC - Field orientation control
GPC - Generalized predictive control
MBPC - Model based predictive control
MI - Motor de indução
PFC - Predictive functional control
QEP - Quadrature Encoder Pulse
RNA - Redes neurais artificiais
SISO - Single input single output
SV-PWM - Space Vector Pulse Width Modulation

CAPÍTULO 1

Introdução

Grande parte das estratégias de controle empregadas para acionamentos elétricos com o motor de indução (MI) utilizam controladores com uma entrada e uma saída (SISO - **S**ingle-**i**nterface **S**ingle-**o**utput) para controlar com malhas distintas o fluxo e a velocidade. Esta abordagem considera que uma específica entrada relaciona-se com uma específica saída e, conseqüentemente, ignora a influência desta e de outras entradas nas demais saídas, ou mesmo em outras variáveis. Esta suposição não é verdadeira para o MI, pois as tensões de eixos direto e em quadratura influenciam simultaneamente o fluxo (estator, entreferro ou rotor) e a velocidade. Uma alternativa para considerar as interações entre variáveis é realizar o controle multivariável; nesta abordagem, a dinâmica do processo está contemplada e, conseqüentemente, o desempenho do acionamento elétrico é melhorado.

Dentre as diversas estratégias de controle multivariável, tem-se destacado a do controle preditivo baseado em modelo (MBPC - **M**odel **B**ased **P**redictive **C**ontrol), o qual gera a lei de controle a partir da minimização de uma função custo, que tem como termos o erro entre as predições das saídas a serem controladas, as referências futuras e a penalidade do esforço de controle, cuja função é propiciar uma resposta mais rápida dos atuadores e dar estabilidade numérica ao cálculo da lei de controle (Coelho e do Amaral, 1991). Se o modelo empregado para realizar a predição fosse não linear, a função custo seria também não linear, tornando a minimização possível apenas com a utilização de algoritmos complexos, as vezes exigindo métodos numéricos; diferentemente do que ocorre no emprego de modelos lineares, cuja solução é encontrada analiticamente. Nesta tese, são realizadas linearizações sucessivas e, conseqüentemente, é encontrada uma solução analítica diferente a cada ciclo de atuação do controle.

Para que o desempenho do MBPC seja satisfatório, é necessário que a predição seja similar ao comportamento futuro da planta, e isto só é possível se o modelo empregado representa fielmente a dinâmica do processo, que na presente tese é o MI, escolhido por ser amplamente empregado na indústria devido ao seu baixo custo e robustez. O MI tem as

equações diferenciais que o representam conhecidas e parâmetros que podem ser obtidos pelos tradicionais testes de rotor bloqueado e a vazio.

Um dos motivos do destaque do MBPC nos meios acadêmico e industrial está na flexibilidade do uso de várias representações para o modelo da planta (Campello e do Amaral, 2002), entre as quais podem ser citadas: função de transferência, modelos formulados empregando a resposta ao impulso ou ao degrau, espaço de estados, modelos baseados em redes neurais ou lógica nebulosa, etc. Estas representações podem ser contínuas ou discretizadas no tempo, sendo esta última a mais propícia ao acionamento elétrico, pois o controle do MI é implementado digitalmente com o uso de microcontrolador, DSP ou FPGA.

O MBPC estudado neste trabalho emprega o modelo discreto de quarta ordem do MI, obtido a partir da transformação trifásica para o sistema de dois eixos, isto é, o direto e o em quadratura, sendo que o fluxo de rotor deve estar sempre alinhado com o eixo direto. As componentes de eixo direto e em quadratura da corrente do estator, do fluxo do rotor e da velocidade do eixo constituem-se nas variáveis de estado. Ressalte-se que nas equações do modelo está presente a frequência elétrica da fonte de alimentação, o que torna o sistema variante no tempo, pois esta frequência deve ser alterada à medida que se varia a velocidade e/ou torque exigido no eixo do motor.

Para a representação do MI no MBPC escolheu-se a modelagem em espaços de estados, que possibilita descrever sistemas multivariáveis, não lineares e variantes no tempo. A opção pela modelagem em espaço de estados traz um inconveniente ao MBPC: a necessidade de se conhecer os estados a cada instante da amostragem. Em outras modelagens, como no caso da função de transferência, apenas as saídas precisam ser conhecidas. As correntes de estator podem ser medidas sem maiores dificuldades, entretanto, a medição do fluxo de rotor é impraticável, pois exigiria sensores e técnicas especiais a serem empregados, o que encareceria o acionamento, bem como diminuiria a sua robustez; acrescente-se ainda o fato de que, em funcionamento, no MI gaiola de esquilo, não se tem como acessar o circuito do rotor, o que impossibilita o cálculo do fluxo do rotor, a partir de suas correntes e tensões. Por esta razão, optou-se por empregar o filtro de Kalman estendido (FKE) que permite determinar estados desconhecidos do MI a partir das medições de tensões e correntes de estator.

1.1 Tema estudado

Pesquisadores vêm propondo técnicas alternativas às existentes no acionamento do MI. Para aumentar a robustez e diminuir custos, as variáveis de estado e os parâmetros têm sido estimados (Ohnishi et al., 1994) e, para melhorar o desempenho, novas estratégias de controle são empregadas - inclusive as baseadas em inteligência artificial (Bose, 1994).

Algumas estratégias de controle, como as denominadas *Deadbeat*, têm como objetivo fazer o sistema atingir a referência em um período de amostragem. Outras técnicas empregadas no acionamento do MI são denominadas de preditivas (Lorentz et al., 1994), entretanto não têm a característica antecipativa da estratégia MBPC (Kennel e Linder, 2000).

Esta tese trata do emprego do controlador preditivo baseado no modelo para o controle da velocidade e do fluxo de rotor do MI, sendo estas variáveis estimadas via filtro de Kalman estendido. O MBPC empregado utiliza o modelo do MI linearizado sucessivamente, tal que a lei de controle obtida é uma equação algébrica. Análise teórica, simulações computacionais e testes experimentais compõem o conteúdo deste trabalho.

1.2 Organização do texto

Capítulo 2 *Controle Preditivo Baseado em Modelo*

Com a motivação de familiarizar os profissionais de acionamentos elétricos com o MBPC, um breve histórico das técnicas MBPC, bem como suas propriedades são apresentadas. As estratégias mais populares são descritas separadamente, enquanto que soluções especiais para casos específicos (com restrições, não linearidades difíceis de modelar, controle com múltiplas taxas de amostragem, etc) são relatadas na revisão bibliográfica. Particularmente a estratégia linearizada com representação do modelo em espaço de estados que é empregado nesta tese para controle direto de fluxo e de velocidade do MI é detalhada.

Capítulo 3 *Controle do Motor de Indução*

A revisão de alguns métodos de controle aplicado ao MI é a base desse capítulo, destacando o clássico FOC (**F**ield **O**riented **C**ontrol), que emprega geralmente controladores de corrente do tipo PI para manipular o fluxo do rotor e o torque desenvolvido pelo MI. Como o termo *controle preditivo* é empregado há bastante tempo na área de acionamentos elétricos para descrever outras técnicas de controle, faz-se uma breve discussão sobre suas diferenças do controlador preditivo baseado no modelo. Por fim, é mostrado como o MBPC linearizado é empregado ao controle do MI.

Capítulo 4 *Resultados de Simulações*

Os resultados de simulação do controlador MBPC aplicado ao acionamento do MI são apresentados, nas mais diferentes situações. São analisadas as variações dos parâmetros do MBPC, evidenciando como pode-se adequar o controle preditivo linearizado a aplicações específicas para acionamento do MI. Também é avaliada a adaptabilidade do controlador à variações de parâmetros da carga.

Capítulo 5 *Resultados Experimentais*

Descrevem-se a bancada empregada para realizar os experimentos e as técnicas usadas no programa feito em linguagem C para o DSP TMS320F2812 da Texas Instruments. A seguir, resultados experimentais são mostrados e discutidos.

Capítulo 6 *Conclusões e Sugestões de Trabalhos Futuros*

As conclusões dos resultados mais importantes, obtidos da análise teórica e dos resultados de simulação e experimentais, são sumarizadas. O capítulo é encerrado com as sugestões para que o desempenho do acionamento implementado seja melhorado.

CAPÍTULO 2

Controle Preditivo Baseado em Modelo

2.1 Introdução

Embora os controladores PID sejam largamente utilizados na indústria, o seu desempenho não é satisfatório para todas as aplicações e outras alternativas de controle vem sendo pesquisadas para superar suas desvantagens. O baixo desempenho do PID ocorre nas situações em que tem de lidar com plantas não lineares, especialmente se forem multi-variáveis; outro inconveniente é a dificuldade de controlar sistemas cujo valor de referência varia com o tempo, pois sendo fixos os ganhos do controlador PID, não há garantia de ótimo desempenho, se a faixa de operação da planta é ampla, como é o caso da maioria dos acionamentos elétricos. Dentre as propostas alternativas ao PID há o controle preditivo baseado em modelo (MBPC), cuja a teoria básica é abordada no presente capítulo.

O MBPC gera a cada instante de amostragem a lei de controle, a partir da minimização de uma função custo, que leva em consideração o erro entre a predição das saídas e das referências futuras; desta forma, a cada ciclo de controle o ganho é determinado. Geralmente nessa função custo é adicionado um termo que permite ajustar o esforço de controle e que melhora a estabilidade numérica. Outra característica que torna o MBPC atrativo é a possibilidade de lidar explicitamente com as restrições na formulação da lei de controle.

2.2 Fundamentos do MBPC

A expressão *controle preditivo baseado em modelo* identifica o conjunto de estratégias de controle fundamentado no modelo da planta para realizar a predição das saídas, compará-las com as referências futuras e através da minimização de uma função custo determinar-se a seqüência de valores futuros para a entrada do processo. Diferentemente dos controladores clássicos, como o PID, que não levam em consideração as implicações futuras dos sinais de controle gerados, o MBPC se utiliza das informações futuras tendo,

assim, comportamento antecipativo; esta característica é atrativa em projetos que se deseja rápida mudança de referência sem *overshoot*.

Uma outra característica do MBPC que o difere dos controladores PID é a possibilidade de lidar de forma explícita com as restrições, de maneira que a minimização da função objetivo é sujeita às limitações do sistema.

Deve-se salientar que o desempenho das técnicas de controle que empregam o modelo será tão bom quanto mais precisa for a representação da planta. Ainda que as equações que regem os fenômenos físicos sejam precisas, deve-se escolher a forma de modelar mais adequada levando-se em consideração: a quantidade de variáveis, a possibilidade de se medir ou estimar as variáveis de estado, o custo computacional para realizar a predição com o modelo adotado, etc.

O algoritmo básico de qualquer estratégia MBPC tem os seguintes estágios:

Predição :

A predição do comportamento futuro das saídas a serem controladas é realizada baseada no modelo da planta, e é geralmente nesta modelagem que se encontram as maiores diferenças entre os diversos controladores MBPC. As predições contêm entre seus termos as entradas futuras, que são os valores que se desejam encontrar para aplicar ao processo. Se a representação do modelo é implementada em espaço de estados faz-se necessário conhecer também os valores atuais dos estados, que podem ser obtidos via medições ou estimados;

Obtenção da lei de controle :

O sinal de controle a ser aplicado no processo é o resultado da minimização do erro entre a predição (etapa anterior) e a referência futura. Geralmente, nesta função custo a ser minimizada é incluído um termo adicional que pondera o esforço de controle e melhora a estabilidade numérica. A função custo empregada na presente tese é a que considera o erro quadrático, e é dada por

$$J = \sum_{j=n_1}^{j=n_y} (\hat{y}(k+j/k) - w(k+j))^2 + \sum_{j=1}^{j=n_u} \lambda \Delta u^2(k+j-1/k) \quad (2.1)$$

Sendo:

- n_1 o horizonte inicial de predição das saídas;
- n_y o horizonte final de predição das saídas;
- n_u o horizonte de controle;
- λ é a ponderação no sinal de controle;
- $\hat{y}(k+j/k)$ é a predição da saída j passos adiante, calculada no instante discreto k ;
- $w(k+j)$ é o sinal de referência futura e;
- $\Delta u(k+j-1/k)$ é a variação no sinal de controle, calculado no instante k .

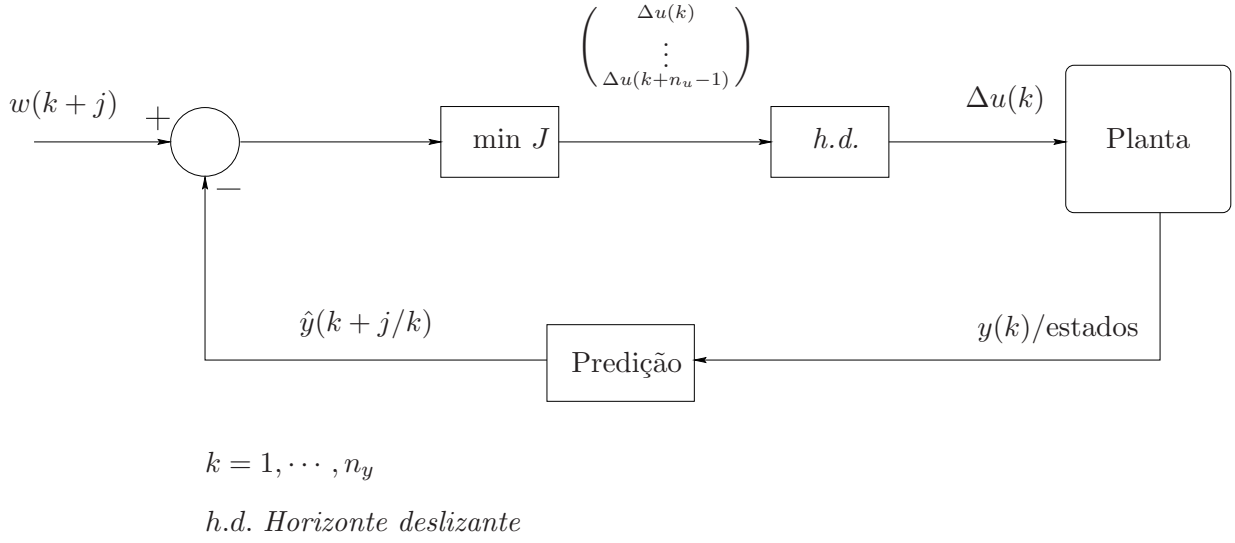


Figura 2.1: Diagrama de blocos do MBPC.

Deve-se notar que nesta função custo a variável livre é o incremento da entrada Δu e não o valor da entrada u . Geralmente faz-se esta escolha para melhorar a resposta em regime permanente. Claramente, o índice de desempenho mínimo ($J = 0$) indica que nenhum ajuste precisa ser feito no sistema de controle, pois o erro $\hat{y}(k+j/k) - w(k+j)$ é nulo, e assim, o sinal de controle não precisa ser modificado, isto é $\Delta u(k+j-1) = 0$. Entretanto, segundo Soeterboek (1992), o uso do incremento $\Delta u(k)$ em vez da entrada $u(k)$ tem o inconveniente de tornar a escolha de λ mais difícil.

Na função custo é assumido que o incremento de controle é nulo após o horizonte de controle, ou seja,

$$\Delta u(k+j-1) = 0, \quad j \geq n_u \quad (2.2)$$

Aplicação das entradas :

Após a obtenção dos sinais de controle, são escolhidos os sinais a serem aplicados ao processo; no caso de ser adotada a estratégia de horizonte deslizante, apenas o primeiro sinal de controle é aplicado ao sistema. Uma outra opção é utilizar mais de uma destas entradas pré-calculadas como feito por Ling e Lim (1996). Após esta etapa retorna-se à predição.

O diagrama de blocos mostrado na Figura 2.1 ilustra uma estratégia MBPC típica. A predição $\hat{y}(k+j/k)$ poderia ser realizada em malha aberta, entretanto, para garantir que possíveis erros na obtenção do modelo prejudiquem o desempenho do controle utilizam-se termos baseados nas medições das saídas ou da obtenção dos estados, via medições ou estimadores.

2.2.1 Influência dos parâmetros do MBPC

2.2.1.1 Esforço de controle

O parâmetro λ pondera o esforço de controle, de forma que, quanto menor for λ maior é o sinal de controle gerado; outra funcionalidade de λ é tornar a inversão matricial na geração da lei de controle sempre possível, como pode ser notado na equações (C.3) e (2.23). Ainda que a inversão matricial seja possível com $\lambda = 0$, esta escolha significa que o atuador pode fornecer energia infinita e isto não corresponde a dispositivos reais, que possuem limitações. Além do mais, valores de entrada muito grandes podem ser gerados com $\lambda = 0$, o que ocasionará *overshoots* elevados.

2.2.1.2 Horizonte inicial de predição

Na maioria das estratégias encontradas na literatura, o horizonte inicial de predição n_1 é escolhido como sendo a unidade. Entretanto, se a planta controlada tiver tempo morto, pode-se selecionar n_1 a fim de evitar cálculos supérfluos, pois a planta não responderá antes do tempo morto, ou seja, se em um sistema discreto o tempo morto é de K_{tm} amostragens, recomenda-se que $n_1 \geq K_{tm}$.

2.2.1.3 Horizonte final de predição

A escolha do horizonte de predição n_y é fundamental para o desempenho do controle, pois a seleção de um valor elevado melhora a estabilidade do sistema, porém pode aumentar demasiadamente o custo computacional. Se, por exemplo, fosse possível implementar o MBPC sem restrições na função custo, com n_y infinito, a escolha adequada de λ torna o sistema estável (Hu e Linnemann, 2002); isto é possível pelo fato de que para $n_y = \infty$ o controlador tem tempo ilimitado para garantir que a saída da planta alcance o valor de referência desejado. Outro fator que deve ser levado em consideração na escolha de n_y é a confiança que o projetista tem no modelo empregado na representação da planta, e se não houver garantia que o modelo descreva a planta com precisão, deve-se escolher n_y menor.

2.2.1.4 Horizonte de controle

O efeito da escolha do horizonte de controle n_u sobre o sistema de controle é similar ao de λ , pois grandes valores de n_u geram sinais de controle elevados. Caso escolha-se $n_u = 1$ a magnitude das entradas representa um valor médio que leva as saídas ao encontro das referências, como se a planta estivesse em malha aberta com entrada constante. Caso escolha-se um valor de $n_u > 1$ o controle gerará sinais de controle que fazem a saída seguir a referência de forma mais próxima. A Figura 2.2 ilustra o comportamento do sistema numa mudança de referência com $n_y = 4$, caso $n_u = 1$ e $n_u = n_y$; nota-se nesta figura que o controle atua mais fortemente se $n_u = n_y$.

2.2.1.5 Período do ciclo de controle

Um parâmetro que não está presente explicitamente na função objetivo, embora seja determinante para o desempenho do sistema, é o tempo de atuação do controle, ou seja, o período em que será realizado um novo ciclo de controle T_a em que modelo do processo é discretizado e empregado para geração de uma nova lei de controle. Deve-se escolher T_a

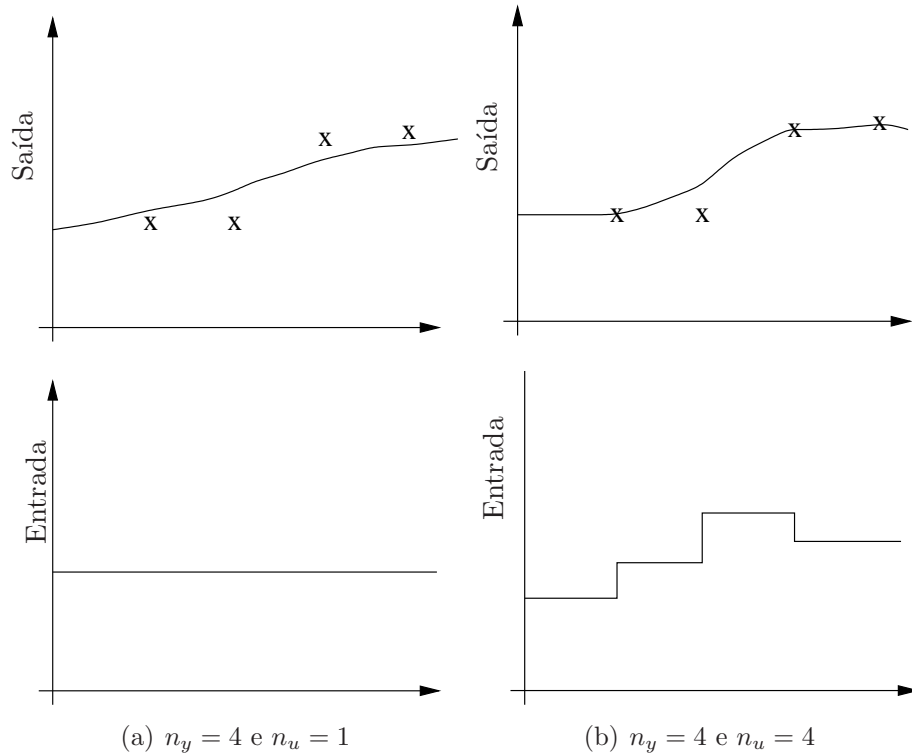


Figura 2.2: Influência do horizonte de controle n_u .

para que as respostas sejam antecipativas e que a predição baseada no modelo represente apropriadamente o comportamento futuro da planta. Para isto os valores adotados para T_a devem considerar a menor constante de tempo do sistema (τ_{menor}) e o valor de n_y , tais que $n_y T_a$ (que corresponde ao valor previsto em tempo) seja aproximadamente $15\% \tau_{menor}$; caso sejam adotados valores de $n_y T_a$ elevados a predição pode não corresponder ao comportamento futuro da planta, prejudicando o controle.

2.3 Algumas técnicas MBPC

2.3.1 Controle Preditivo Funcional

A estratégia de Controle Preditivo Funcional (PFC-**P**redictive **F**unctional **C**ontrol) foi uma das primeiras técnicas MBPC a serem desenvolvidas. Das características do PFC, a flexibilidade de uso de vários modelos e o baixo custo computacional têm constituído-se nas motivações principais de seu emprego. Outra característica notável é a sua simplicidade, associada a sua capacidade de lidar de forma explícita com as restrições, mesmo quando os processos são instáveis (Rossiter e Richalet, 2002).

O algoritmo básico do PFC emprega o conceito de pontos coincidentes, o que significa que os erros entre a predição e as referências futuras são considerados apenas para estes pontos selecionados; geralmente escolhe-se apenas um ponto coincidente. A minimização da função objetivo é substituída por uma igualdade algébrica, o que torna o algoritmo

menos dispendioso computacionalmente e mais simples, viabilizando sua aplicação em processos de dinâmica rápida.

No PFC a trajetória futura desejada para a saída da planta, em um ponto coincidente n_y passos adiante, é dada por

$$y_{traj} = w_{k+n_y} - \beta^{n_y}(w_{k+n_y} - y_{k+n_y}) \quad (2.3)$$

Da leitura desta última equação, conclui-se que a trajetória futura y_{traj} desejada é uma aproximação de primeira ordem, no qual o ajuste do parâmetro β é fundamental, uma vez que é responsável pela similaridade entre y_{traj} e o comportamento real futuro da planta. Portanto, a comparação entre o comportamento futuro da planta, descrita pela equação (2.3), e a predição y_{k+n_y} determina os sinais futuros de controle. Matematicamente tem-se

$$y_{k+n_y} = y_{traj} \quad (2.4)$$

Nota-se nesta última equação, que o modelo empregado para o PFC é irrelevante, pois seja qual for a modelagem empregada para realizar a predição, os sinais de controle futuros são necessários para representar y_{k+n_y} apropriadamente; estes sinais de controle futuros são as variáveis livres a serem determinadas para torna a igualdade dada pela equação (2.4) verdadeira.

A independência do modelo permite o emprego de modelagens desenvolvidas mais recentemente, como no artigo de Skrjanc e Matko (2000), no qual os autores descrevem o emprego do PFC com o modelo nebuloso do tipo Takagi-Sugeno, no controle de trocadores de calor. Este dispositivo industrial tem comportamento altamente não linear e de difícil modelagem, de maneira que foram necessárias seis regras nebulosas para descrever o seu comportamento. Mesmo com todas estas inovações propostas para representar o processo industrial, os conceitos da estratégia de controle utilizados são os mesmos do PFC clássico. ou seja, as futuras modelagens a serem desenvolvidas poderão ser empregadas no PFC. Os resultados exibidos no artigo mostram que o controlador preditivo funcional nebuloso teve bom desempenho.

Apesar de sua eficácia comprovada em várias implementações, a falta de rigor matemático na formulação do algoritmo do PFC não permite conhecer *a priori* o desempenho do sistema de controle, bem como seu emprego não é viável em processos com dinâmica complexa e nem em processos multivariáveis (Rossiter, 2003).

2.3.2 Controlador de Matriz Dinâmica

O algoritmo do Controlador de Matriz Dinâmica (DMC- **D**inamic **M**atrix **C**ontrol) é baseado na otimização de uma função custo quadrático do tipo da equação (2.1). A inovação proposta neste método (Cutler e Ramaker, 1980) é a utilização da modelagem

via resposta ao degrau, dada por

$$y(t) = \sum_{i=1}^{i=\infty} g_i \Delta u(t-i) \quad (2.5)$$

Sendo:

- $y(t)$ o sinal de saída;
- $\Delta u(t)$ a variação do sinal de controle e;
- g_i (com $i = 1, 2, \dots$) representa o i -ésimo coeficiente da resposta ao degrau.

A descrição matricial da predição do DMC é

$$\hat{\mathbf{Y}} = \mathbf{G}\mathbf{u} + \mathbf{f} \quad (2.6)$$

sendo que

- \mathbf{f} representa os termos da resposta livre do sistema e, portanto, não dependem dos sinais de controle futuros;
- \mathbf{u} são os sinais de controle futuros;
- \mathbf{G} é a chamada matriz dinâmica - tem como elementos os coeficientes da resposta ao degrau - e é dada por

$$\mathbf{G} = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{n_u} & g_{n_u-1} & \cdots & g_1 \\ \vdots & \vdots & \ddots & \vdots \\ g_{n_y} & g_{n_y-1} & \cdots & g_{n_y-n_u+1} \end{bmatrix} \quad (2.7)$$

Como a resposta dinâmica é dada por $\mathbf{G}\mathbf{u}$, conclui-se que o comportamento dinâmico do controle é determinado pelos elementos de \mathbf{G} , daí a denominação deste algoritmo de Controlador de Matriz Dinâmica.

Segundo Camacho e Bordons (1999), o DMC tem sido aplicado a sistemas industriais petroquímicos, principalmente em plantas multivariáveis que possuem restrições na operação. A descrição de uma aplicação recente é feita por Zhang et al. (2004); os autores descrevem a aplicação do algoritmo DMC no controle da largura de lâminas fabricadas para separar eletrodos de baterias. Os resultados mostrados no artigo demonstram o bom desempenho do DMC.

2.3.3 Controlador Preditivo Generalizado

Esta é a estratégia MBPC mais encontrada na literatura, principalmente pelo fato de ser uma das propostas MBPC mais bem fundamentadas do ponto de vista matemático,

permitindo ao implementador determinar *a priori* o desempenho do sistema. O Controlador Preditivo Generalizado (GPC- **G**eneralized **P**redictive **C**ontrol) descreve a planta com o modelo CARIMA (**C**ontroller **A**uto-**R**egressive **I**ntegrated **M**oving-**A**verage), dado por

$$A(z^{-1})\Delta y(k) = z^{-d}B(z^{-1})\Delta u(k-1) + C(z^{-1})e(k) \quad (2.8)$$

Sendo:

- $u(k-1)$, o sinal de controle;
- $y(k)$, a saída do processo;
- $e(k)$, sinal de ruído (geralmente considera-se ruído branco de média zero);
- os polinômios que descrevem o sistema dados por

$$A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_{na}z^{-na} \quad (2.9)$$

$$B(z^{-1}) = b_0 + b_1z^{-1} + \dots + b_{nb}z^{-nb} \quad (2.10)$$

$$C(z^{-1}) = 1 + c_1z^{-1} + \dots + c_{nc}z^{-nc} \quad (2.11)$$

Como pode ser notado na equação (2.8) o modelo utilizado no GPC é similar a uma função de transferência discreta, permitindo a modelagem apenas de sistemas lineares. Esta modelagem tem como principais vantagens:

- Representar os processos instáveis;
- Representar os processos com integradores;
- Possuir poucos parâmetros, quando comparada aos modelos obtidos com a resposta ao impulso e a resposta ao degrau.

Como desvantagens podem ser citadas:

- A necessidade de se conhecer, ou pelo menos supor, a ordem do sistema;
- O modelo CARIMA dificulta a aplicação desta estratégia aos processos multivariáveis;
- O fato da predição ser complexa.

Uma formulação em espaço de estados para o GPC é descrita em (Ordys e Clarke, 1993), no qual são comparados os enfoques polinomial e em espaço de estados.

Como todas as técnicas MBPC, o GPC possibilita a inclusão explícita das restrições, como descrito por Tsang e Clarke (1988), Geng e Geary (1998) e Camacho (1993).

2.3.4 Outras estratégias MBPC

Para contornar a falta de desempenho das populares estratégias MBPC citadas anteriormente em casos específicos, como restrições nos atuadores ou comportamento não linear da planta, outras técnicas têm sido estudadas. Este tópico dedica-se a destacar estas outras soluções desenvolvidas para solucionar as limitações dos algoritmos mais usuais.

Para plantas não lineares, cujo modelo não está disponível ou mesmo de difícil obtenção, uma alternativa é a utilização das redes neurais artificiais RNA's, como foi proposto por Shin e Park (1998). Os autores utilizaram uma rede neural de três camadas para

realizar a predição e a minização da função objetivo foi feita com a técnica de algoritmos genéticos (AG); os resultados de simulação mostraram que o controle é mais eficaz com a utilização do AG do que com a técnica de minimização quase-Newton, embora com AG o custo computacional seja maior.

Draeger et al. (1995) desenvolveram um algoritmo MBPC para controle do PH de um reator de neutralização, baseado no DMC e cuja modelagem do processo é realizada via RNA's; o treinamento da RNA usa como dados a entrada e a saída do processo industrial, medidas com o sistema sendo controlado por um PI, portanto, sem o emprego de informação sobre o modelo da planta nem de limitações das condições de operação. Para a seleção dos pesos das ligações sinápticas dos neurônios, foi utilizado o algoritmo genético como método de otimização. Diferentemente do artigo do parágrafo anterior, este é validado por resultados experimentais, que foram satisfatórios apesar das não linearidades da planta. Salienta-se que os autores sugerem que o algoritmo proposto pode ser empregado em sistemas multivariáveis.

Poulsen et al. (2001) desenvolveram um algoritmo MBPC linearizado para controle do volume de água em dois tanques acoplados. Para garantir que os erros na modelagem e a presença de perturbações não deteriorassem o desempenho do sistema, principalmente em regime permanente, foi acrescentado uma ação integral; isto aumentou a ordem do sistema, pois uma variável de estado auxiliar foi criada. Como as restrições do sistema foram consideradas, a lei de controle é obtida via algoritmo de busca. A comparação dos resultados de simulação e experimentais com o tradicional controle PID evidenciou o melhor desempenho do MBPC proposto.

Quando a planta é linear e tem parâmetros variantes no tempo, é adequado que se empregue estratégias adaptativas. Nestas os parâmetros são estimados a cada amostragem e uma nova lei de controle é gerada tendo o modelo sempre atualizado. Se as mudanças nas magnitudes dos parâmetros são lentas e determinísticas, a estimação de parâmetros é eficiente, garantindo bom desempenho ao algoritmo de controle; entretanto, quando as mudanças ocorrem abruptamente e randomicamente, o desempenho do controle fica prejudicado. Lindoff e Holst (1997) propõem uma estratégia que leva em consideração o comportamento estatístico da mudança dos parâmetros, permitindo uma modelagem precisa; também é desenvolvida a formulação algébrica para a lei de controle se o horizonte de predição for igual a um e a dois. Segundo os autores, para horizontes de predição maiores, o cálculo da lei de controle torna-se demasiadamente trabalhoso e deve ser implementado computacionalmente. Os resultados de simulação mostrados evidenciam o bom desempenho da proposta.

Apesar da maioria das estratégias MBPC ter sido formulada no domínio discreto no tempo, alguns autores ainda desenvolveram estratégias na forma contínua, como descrito nas referências (Gawthrop et al., 1998) e (Fontes e Magni, 2003). Gawthrop et al. (1998) realizaram o estudo do GPC para sistemas não lineares e multivariáveis, representando a planta em espaço de estados. Os resultados das simulações realizadas com o uso do programa Simulink/Matlab, mostraram o bom desempenho do GPC não linear em relação

ao GPC linearizado. Fontes e Magni (2003) propuseram um algoritmo preditivo robusto, analisando a estabilidade do sistema de controle ainda que sujeito a certos tipos de perturbações. A proposta foi aplicada ao controle de posição e velocidade de um robô.

2.4 MBPC empregado

A maioria das técnicas de controle preditivo encontradas na literatura tratam de plantas cujos modelos podem ser representados por equações diferenciais lineares. Entretanto, nos sistemas físicos reais, as relações entre as variáveis são intrinsecamente não lineares e o emprego das técnicas de controle que levem em conta estas não linearidades geralmente são complexas e computacionalmente dispendiosas. Isto torna as aplicações nos processos industriais cujas constantes de tempo envolvidas são pequenas inviável, pois, exige que o sistema de controle atue rapidamente, solicitando um rápido de processamento computacional para gerar a lei de controle a cada amostragem.

A alternativa às técnicas não lineares empregada neste trabalho é a linearização sucessiva do modelo a cada ciclo de controle; como consequência, um sistema linear diferente para cada instante representa o processo não linear na região no ponto de operação atual.

2.4.1 Modelo da planta

O modelo de uma planta em espaço de estados é dado por

$$\begin{cases} \dot{x} = f(x) + Bu + \delta = A(x)x + Bu + \delta \\ y = Cx \end{cases} \quad (2.12)$$

na qual

- $x \in \mathbb{R}^{(n \times 1)}$ denota o vetor de estados;
- $u \in \mathbb{R}^{(p \times 1)}$ denota o vetor de entradas ;
- $\delta \in \mathbb{R}^{(n \times 1)}$ é o vetor de perturbação;
- $f(x)$ uma função não linear que pode ser desmembrada em $A(x)x$, sendo que a matriz característica do sistema é $A(x) \in \mathbb{R}^{(n \times n)}$;
- A matriz de entrada $B \in \mathbb{R}^{(n \times p)}$;
- O vetor de saída $y \in \mathbb{R}^{(q \times 1)}$;
- A matriz de saída $C \in \mathbb{R}^{(q \times n)}$ depende das saídas escolhidas para serem controladas.

A representação do modelo em espaço de estados, dada pela equação (2.12), é contínua no tempo; no entanto, a maioria das estratégias MBPC encontradas na literatura são desenvolvidas empregando modelos discretos, pois as implementações são realizadas digitalmente.

2.4.1.1 Discretização

A representação discreta, do modelo em espaços de estado contínuo dado pela equação (2.12), é determinada encontrando-se as matrizes discretas: característica do sistema A_d , de entrada B_d e de saída C_d . Estas são dadas por

$$\begin{cases} A_d = e^{AT_a} \approx I + AT_a + \frac{(AT_a)^2}{2} \\ B_d = \int_0^{T_a} e^{AT_a - \tau A} B d\tau \approx BT_a + \frac{ABT_a^2}{2} \\ C_d = C \end{cases} \quad (2.13)$$

com T_a representando o período de amostragem. O modelo discreto, portanto, é representado por

$$\begin{cases} x_{k+1} = A_d(x_k)x_k + B_d u_k + \delta_k \\ y_k = C_d x_k \end{cases} \quad (2.14)$$

2.4.1.2 Linearização

Na equação (2.14), o termo $A_d(x_k)x_k$ é não linear. A linearização deste termo gerará uma nova matriz do sistema A_{dl} - que não depende das variáveis de estado - e um termo adicional.

Se no instante de amostragem k os estados são conhecidos e representados por x'_k , tem-se que a matriz do sistema para o caso discreto e linearizado A_{dl} é encontrada através de

$$A_{dl} = \frac{d[A_d(x_k)x_k]}{dx_k} \Big|_{x'_k}.$$

Este procedimento gera um termo extra $[f(x'_k) - A(x'_k)x'_k]$, que somado à perturbação δ_k resulta em $D(x_k, \delta_k)$, ou seja, ainda que $\delta_k = 0$, há um termo adicional na equação (2.15) devido à linearização do modelo. Pode-se então escrever o seguinte sistema

$$\begin{cases} x_{k+1} = A_{dl}(x)x_k + B_d u_k + D(x_k, \delta_k) \\ y_{k+1} = C_d x_{k+1} \end{cases} \quad (2.15)$$

2.4.2 Predição

Para a técnica de controle proposta nesta tese a predição é realizada em espaço de estados (Rossiter, 2003)

$$\hat{\mathbf{y}} = H' \mathbf{x} + H'' \mathbf{u} + H''' \mathbf{d} \quad (2.16)$$

Sendo $\mathbf{d} \in \Re^{[(n_y n) \times 1]}$ o vetor predição da perturbação e

$$H' = [CA \quad CA^2 \quad \dots \quad CA^{n_y}]^T \quad (2.17)$$

$$H'' = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ CA^2B & CAB & CB & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{n_y-1}B & CA^{n_y-2}B & \cdots & & CB \end{bmatrix} \quad (2.18)$$

$$H''' = \begin{bmatrix} C & 0 & 0 & \cdots & 0 \\ CA & C & 0 & \cdots & 0 \\ CA^2 & CA & C & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{n_y-1} & CA^{n_y-2} & CA^{n_y-3} & \cdots & C \end{bmatrix} \quad (2.19)$$

Nas quais

- $H' \in \Re^{[(n_y q) \times n]}$
- $H'' \in \Re^{[(n_y q) \times (n_u p)]}$
- $H''' \in \Re^{[(n_y q) \times (n_y n)]}$

Há dois tipos de perturbações: as determinísticas, que são aquelas que podem ser preditas exatamente e as estocásticas, que não podem ser preditas exatamente (Soeterboek, 1992). No presente artigo assume-se que o δ_k é determinístico e tem valores constante nas predições; com isto consegue-se diminuir o esforço computacional.

2.4.3 Lei de controle

A função objetivo escolhida para obtenção da lei de controle é a mesma da equação (2.1), que reescrita matricialmente torna-se:

$$J = (\hat{\mathbf{y}} - \mathbf{w})^T W_y (\hat{\mathbf{y}} - \mathbf{w}) + \mathbf{u}^T W_u \mathbf{u} \quad (2.20)$$

na qual

- $\mathbf{w} \in \Re^{(n_y q) \times 1}$ é o vetor de referências futuras para as saídas a serem controladas;
- $W_y \in \Re^{(n_y q) \times (n_y q)}$, matriz definida-positiva (geralmente é diagonal) que permite enfatizar cada uma das saídas controladas e suas predições;
- $W_u \in \Re^{n_u \times n_u}$, matriz definida-positiva e usualmente diagonal que pondera o esforço de controle das entradas;
- $\mathbf{u} \in \Re^{n_u \times n_u}$ é o vetor de entradas.

No caso de plantas não lineares a minimização da função objetivo deve de ser resolvida numericamente. No entanto, se o modelo é linearizado obtém-se algebricamente o seu valor mínimo; neste caso, a lei de controle tem a mesma forma dos sistemas lineares, porém as matrizes que representam os parâmetros da planta (H' , H'' e H''') não se mantêm constantes de um ciclo de controle para outro, pois contêm entre seus elementos termos

que dependem dos estados e termos do modelo que são variantes no tempo e, portanto, mudam a cada instante.

Minimizando a equação (2.20), conforme descrito no Apêndice C, tem-se a lei de controle dada por

$$\mathbf{u} = (H''^T W_y H'' + W_u)^{-1} H''^T W_y (\mathbf{w} - H' \mathbf{x} - H''' \mathbf{d}) \quad (2.21)$$

que só não tem conhecido o termo relativo a predição da perturbação, isto é, o termo \mathbf{d} . Soeterboek (1992) discute como encontrar a perturbação e sua predição a partir de informações básicas, como, por exemplo, se seu comportamento é senoidal, constante, etc. Isto torna possível obter não somente o valor exato da perturbação bem como realizar sua predição. Entretanto, no caso tratado nesta tese trabalho, a perturbação é o torque de carga T_c , cujo valor e comportamento no tempo não são conhecidos *a priori*; portanto, as técnicas descritas não se aplicam. No entanto, T_c relaciona-se com os estados da máquinas através da equação eletromecânica, de tal maneira que se os demais estados são conhecidos pode-se estimá-lo. No decorrer deste texto é apresentada duas maneiras de se estimar T_c :

1. Calculando-o diretamente a partir da equação eletromecânica;
2. Acrescentando-o como estado no filtro de Kalman estendido e considerando sua dinâmica lenta, ou seja, $T_c \approx 0$, em relação ao tempo de amostragem empregado para discretizar o modelo usado no FKE.

Os resultados mostrados nas simulações do capítulo 4 mostram o desempenho de ambas as abordagens.

2.4.4 MBPC variacional

Com o intuito de obter melhores respostas em regime, emprega-se o modelo variacional - também chamado modelo aumentado; nesta modelagem é utilizado o incremento da entrada Δu em vez da sua magnitude u . Desta forma, o modelo a ser empregado no MBPC proposto é dado por

$$\begin{cases} \begin{bmatrix} x_{k+1} \\ u_k \end{bmatrix} = \tilde{A} \begin{bmatrix} x_k \\ u(k-1) \end{bmatrix} + \tilde{B} u_k^\Delta + \tilde{D} \\ y_{k+1} = \tilde{C} \begin{bmatrix} x_{k+1} \\ u(k-1) \end{bmatrix} \end{cases} \quad (2.22)$$

Sendo

$$\begin{aligned} \tilde{A} &= \begin{bmatrix} A_{al}(x) & B_d \\ 0_{p \times n} & I_{p \times p} \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B_d \\ I_{p \times p} \end{bmatrix} \\ \tilde{D} &= \begin{bmatrix} D(x_k, \delta_k) \\ 0_{p \times 1} \end{bmatrix}, \quad \tilde{C} = [C_d \quad 0_{q \times p}] \end{aligned}$$

de dimensões

- $\tilde{A} \in \mathfrak{R}^{[(n+p) \times (n+p)]}$
- $\tilde{B} \in \mathfrak{R}^{[(n+p) \times p]}$;
- $\tilde{D} \in \mathfrak{R}^{[(n+p) \times 1]}$;
- $\tilde{C} \in \mathfrak{R}^{[q \times (n+p)]}$.

O vetor de estados na modelagem aumentada é

$$\tilde{x} = \begin{bmatrix} x_{k+1} \\ u_k \end{bmatrix} \in \mathfrak{R}^{(n+p) \times 1}.$$

O vetor de entradas é dado por

$$u_k^\Delta = \begin{bmatrix} \Delta u_k^1 \\ \Delta u_k^2 \\ \vdots \\ \Delta u_k^p \end{bmatrix} \in \mathfrak{R}^{p \times 1}.$$

O modelo varacional tem como desvantagem o aumento do custo computacional, pois aumenta a ordem do sistema representado em espaço de estados. Porém, as respostas em regime são melhores, com erros entre a referência e os valores reais aproximadamente nulos.

A predição é similar àquela dada pela equação (2.16), trocando-se o vetor \mathbf{u} de entradas pelo \mathbf{u}^Δ e substituindo nas matrizes, dadas nas equações (2.17), (2.18) e (2.19), A por \tilde{A} , B por \tilde{B} , C por \tilde{C} e D por \tilde{D} de maneira que

- $H' \in \mathfrak{R}^{[(n_y q) \times (n+p)]}$
- $H'' \in \mathfrak{R}^{[(n_y q) \times (n_u p)]}$
- $H''' \in \mathfrak{R}^{[(n_y q) \times n_y(n+p)]}$

Para uma função objetivo quadrática, similar à (2.20), substituindo \mathbf{u} por \mathbf{u}^Δ e considerando que $\mathbf{u}^\Delta = 0$, para $k > n_u$ a minimização da função objetivo têm a forma idêntica a do modelo não varacional (C.3), ou seja,

$$\mathbf{u}^\Delta = (H''^T W_y H'' + W_u)^{-1} H''^T W_y (\mathbf{w} - H' \mathbf{x} - H''' \tilde{\mathbf{d}}) \quad (2.23)$$

Sendo o novo vetor predição da perturbação $\tilde{\mathbf{d}} \in \mathfrak{R}^{q \times n_y(n+p)}$.

2.5 Conclusões

A análise sobre o controle preditivo mostrou que seu algoritmo demanda mais custo computacional do que técnicas mais tradicionais, como o controlador PID, isto significa que para implementações em tempo real deve-se selecionar adequadamente os parâmetros do MBPC para que se obtenha a melhor relação entre a demanda numérica e o desempenho do sistema. O elevado custo computacional ocorre pela geração de um ganho a cada ciclo de

controle, através da minimização de uma função custo. O MBPC tem custo computacional elevado pelo fato da lei de controle conter multiplicações e inversão matriciais. Entretanto, como o ganho é ajustado de acordo com as condições de operação da planta a cada ciclo, o desempenho do MBPC é superior do que os controladores com ganho fixo.

As descrições das estratégias preditivas e uma breve revisão bibliográfica evidenciaram que a característica principal que diferencia as várias técnicas MBPC é a possibilidade de lidar com vários modelos. A aplicação do MBPC permite o emprego de modelos multivariáveis, contemplando toda a dinâmica do processo.

O MBPC a ser empregado para controle de fluxo e velocidade do MI foi desenvolvido em duas vertentes: uma com cálculo dos incrementos de controle e outra com cálculo do valor da entrada. A primeira aumenta o custo computacional, já que a ordem do modelo da planta é aumentado, mas tem melhores respostas em regime.

CAPÍTULO 3

Controle do Motor de Indução

3.1 Introdução

O PI é o controlador mais usual nas aplicações de acionamentos elétricos, embora o seu desempenho não seja satisfatório em todas as condições de operação dos motores. Alguns de seus inconvenientes são listados a seguir

- Embora sejam encontrados na literatura acadêmica vários métodos matematicamente fundamentados, geralmente, os ajustes dos ganhos proporcional e integral do PI na indústria são escolhidos empiricamente (Palm III, 1986).
- Os ganhos são escolhidos para um ponto de operação e, portanto, se houver mudança do ponto de operação não há garantia de bom desempenho;
- Não lida de forma explícita com restrições das saídas, entradas ou variáveis de estado da planta.

Um esquema básico do controle, por orientação de fluxo de motores de indução, empregando PI, está ilustrado no diagrama de blocos da Figura 3.1. Caso se deseje controlar diretamente o fluxo, um outro controlador PI deve ser inserido para que gere a referência i_{sd}^* . Como se observa, na malha de controle de velocidade, o valor de referência i_{sq}^* é gerado a partir do erro de velocidade.

Observe ainda na Figura 3.1, a existência de controladores em cascata, o que dificulta o ajuste de seus ganhos.

Uma alternativa aos controladores PI é o controle preditivo, sendo que esta expressão é empregada, na área de acionamentos elétricos, para definir uma gama de diversas soluções. Para melhor compreender estas soluções é apresentada na seção 3.2 uma revisão de aplicações encontradas na literatura.

Na seção 3.3 é apresentado a discretização e linearização do modelo do motor de indução em variáveis de estado, bem como o algoritmo preditivo aplicado ao controle de fluxo de rotor e velocidade.

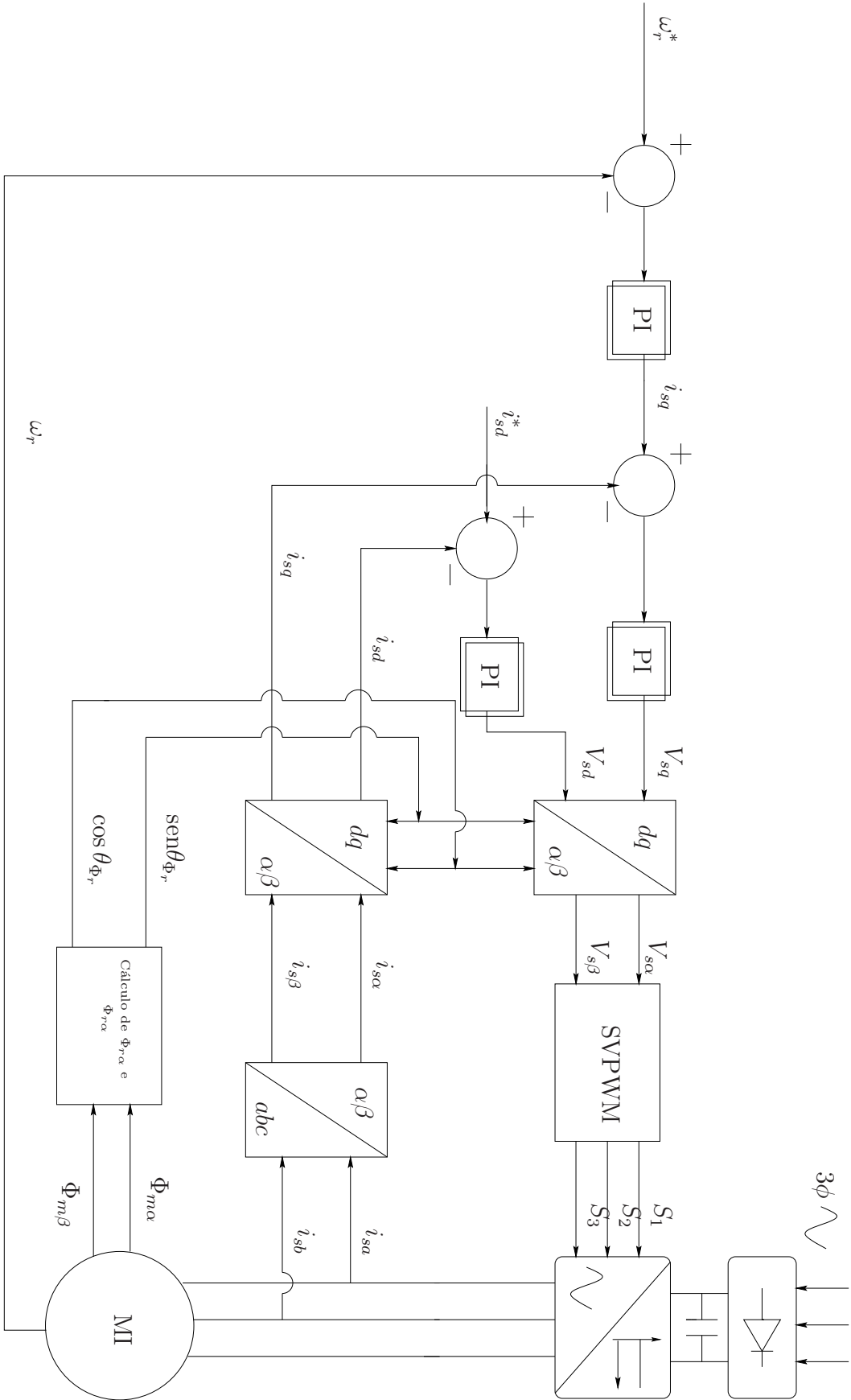


Figura 3.1: Diagrama de blocos do controle de velocidade e fluxo de rotor da máquina de indução com controladores PI.

O capítulo é encerrado na seção 3.4 com as conclusões sobre as potencialidades do emprego do MBPC em MI.

3.2 Controle preditivo na área de acionamentos elétricos

Nas linhas de pesquisa de acionamentos elétricos o termo *controle preditivo* é abrangente, pois várias estratégias são assim denominadas.

Tradicionalmente, controladores preditivos empregados no acionamento do motor de indução geram os sinais de controle a partir das equações do modelo, com um horizonte de apenas um ciclo futuro; além disto, estes controladores são do tipo SISO, o que significa que as dinâmicas elétricas e mecânicas são manipuladas independentemente. O MBPC, por sua vez, utiliza uma função custo na qual a predição do comportamento das saídas da planta, obtida a partir do modelo, é empregada para determinar os sinais futuros de controle que definem o processo no horizonte futuro que se deseja.

São descritas, na sequência, algumas estratégias preditivas com o intuito de diferenciá-las do controle preditivo baseado em modelo; em termos gerais, as estratégias preditivas estudadas em acionamentos elétricos não têm a característica antecipativa deste último.

Altuna (2002) implementa o controlador preditivo de corrente para um MI a partir da equação da tensão de estator em coordenadas estacionárias, dada por

$$\bar{V}_{s,\alpha\beta} = R_s \bar{I}_{s,\alpha\beta} + L'_s \frac{d\bar{I}_{s,\alpha\beta}}{dt} + \frac{L_m}{L_r} \frac{\bar{\Phi}_{r\alpha\beta}}{dt} \quad (3.1)$$

na qual

- $\bar{V}_{s,\alpha\beta}$, $\bar{I}_{s,\alpha\beta}$ e $\bar{\Phi}_{r\alpha\beta}$ são respectivamente, os vetores espaciais da tensão, corrente e fluxo de estator;
- L_r e L_m são respectivamente própria de rotor e de magnetização;
- L'_s é a denominada indutância transitória de estator.

O objetivo desta estratégia preditiva é conseguir que na próxima amostragem a corrente $\bar{I}_{s,\alpha\beta}(k+1)$ seja igual a referência $\bar{I}_{s,\alpha\beta}^*(k+1)$. A referência de corrente futura $\bar{I}_{s,\alpha\beta}^*(k+1)$ é dada por

$$\bar{I}_{s,\alpha\beta}^*(k+1) = 2\bar{I}_{s,\alpha\beta}^*(k) - \bar{I}_{s,\alpha\beta}^*(k-1).$$

Para gerar o valor de tensão que deve ser aplicado à máquina utiliza-se a equação (3.1) na forma discreta, isto é

$$\bar{V}_{s,\alpha\beta}(k+1) = R_s \bar{I}_{s,\alpha\beta}(k) + \sigma L_s \frac{\bar{I}_{s,\alpha\beta}^*(k+1) - \bar{I}_{s,\alpha\beta}(k)}{T_a} + \frac{L_m}{L_r} \frac{\bar{\Phi}_{r\alpha\beta}(k) - \bar{\Phi}_{r\alpha\beta}(k-1)}{T_a} \quad (3.2)$$

sendo T_a o período do ciclo de controle.

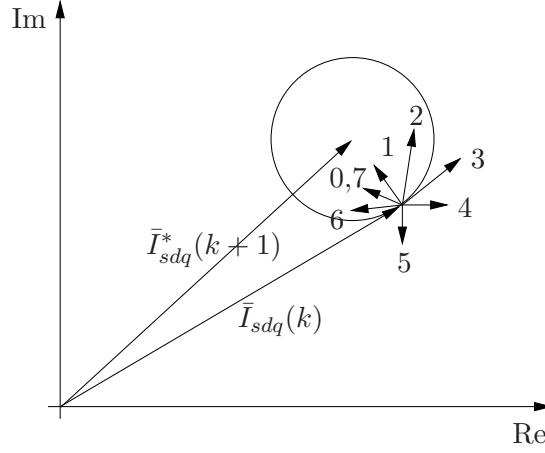


Figura 3.2: Controle de corrente preditivo por histerese. Fronteira circular centrada da referência e as sete possíveis trajetórias do vetor espacial corrente de estator.

Para melhorar a resposta dinâmica do sistema, emprega-se a orientação de fluxo do rotor. Nesta situação, a equação da tensão de estator é dada por

$$\begin{aligned} \bar{V}_{sdq}(k+1) = & R_s \bar{I}_{sdq}(k) + \sigma L_s \frac{\bar{I}_{sdq}^*(k+1) - \bar{I}_{sdq}(k)}{t_a} + \frac{L_m}{L_r} \frac{\bar{\Phi}_{rdq}(k) - \bar{\Phi}_{rdq}(k-1)}{t_a} + \\ & + j\omega_s [\sigma L_s \bar{I}_{sdq}(k) + \frac{L_m}{L_r} \bar{\Phi}_{rdq}(k)] \end{aligned} \quad (3.3)$$

No caso da orientação de fluxo ser perfeita ($\Phi_r = \Phi_{rd}$) e do controle do fluxo ser eficiente ($\Phi_r \approx cte$), a equação anterior é simplificada para

$$\bar{V}_{sdq}(k+1) = R_s \bar{I}_{sdq}(k) + \sigma L_s \frac{\bar{I}_{sdq}^*(k+1) - \bar{I}_{sdq}(k)}{T_a} + j\omega_s [\sigma L_s \bar{I}_{sdq}(k) + \frac{L_m}{L_r} \bar{\Phi}_{rd}(k)] \quad (3.4)$$

sendo a referência de corrente dada por

$$\bar{I}_{sdq}^*(k+1) = 2\bar{I}_{sdq}^*(k) - \bar{I}_{sdq}^*(k-1).$$

Nota-se nesta estratégia preditiva, que a referência futura é calculada a cada instante com base na referência atual, o que não permite resposta antecipativa, pois não há a garantia de que a referência futura calculada seja idêntica à referência futura real.

Outra estratégia preditiva foi proposta por Holtz e Stadtfeld (1983), na qual, vetor espacial $\bar{I}_{sdq}^*(k+1)$ é gerado a partir da referência atual $\bar{I}_{s,\alpha\beta}^*(k)$, isto é,

$$\bar{I}_{s,\alpha\beta}^*(k+1) = \bar{I}_{s,\alpha\beta}^*(k) + \frac{d\bar{I}_{s,\alpha\beta}^*(t)}{dt} \Big|_{t=k} [t(k+1) - t(k)] \quad (3.5)$$

Como pode ser visto na Figura 3.2, uma fronteira circular centrada no vetor de referência de corrente predito $\bar{I}_{s,\alpha\beta}^*(k+1)$ é criada, na qual o raio do círculo corresponde à distância entre $\bar{I}_{s,\alpha\beta}^*(k+1)$ e o valor atual do vetor espacial corrente de estator $\bar{I}_{s,\alpha\beta}(k)$. Esta fronteira é chamada por Kennel e Linder (2000) de área ou espaço de histerese.

São calculadas as sete possíveis posições do vetor corrente para os oito estados de chaveamento do inversor, e, dentre os estados que fazem a corrente ficar dentro da área circular, escolhe-se um. Se, por exemplo, o critério de escolha for a minimização da frequência de chaveamento, será selecionado o estado que permite que as chaves fiquem ligadas o maior tempo possível. Este controlador foi chamado de preditivo ótimo por Lorentz et al. (1994), pois emprega o modelo do MI para prever a posição espacial do vetor corrente de estator e seleciona o estado e o período de chaveamento do inversor segundo algum critério de otimização.

Diferentemente destes métodos, o MBPC pode atuar no horizonte de futuro desejado pelo projetista e, por esta razão, alguns autores referem-se a ele como *Long-range predictive control* (Zhang et al., 1997) para diferenciar das estratégias preditivas usuais na área de acionamentos elétricos.

3.3 MBPC aplicado ao acionamento elétrico

Conforme comentado na seção 3.2, a expressão controle preditivo é empregado há algum tempo para descrever várias propostas de acionamento elétrico. Kennel e Linder (2000) descrevem algumas destas estratégias e apresentam uma variedade de técnicas de controle preditivo que podem ser aplicadas para acionar motores elétricos, incluindo o MBPC como uma alternativa viável ao controle do MI.

A aplicação do MBPC para controle de posição de um motor *brushless DC*, cujo modelo é linear, foi estudada experimentalmente por Low et al. (1997) com sucesso, evidenciando o bom desempenho do MBPC, ainda que o torque da carga seja variável ou que ocorram mudanças abruptas no valor da inércia.

O uso do algoritmo MBPC no acionamento de um motor de indução foi estudado por Zhang et al. (1997). Os autores projetaram controladores para as correntes de estator de eixo direto e de eixo em quadratura e compararam os resultados de duas implementações PI: uma no modelo estacionário e outra no síncrono. Os resultados de simulação apontaram o desempenho superior do MBPC, embora, as comparações tenham sido baseadas em um único teste, no qual o sistema de controle não contemplou a parte mecânica do sistema.

Kennel et al. (1997) utilizam o GPC sem considerar a tensão induzida no estator pelo fluxo de rotor, tornando linear as equações de tensão do estator, utilizadas para

implementar o controle de corrente. Os autores utilizam um controlador GPC para cada variável que desejam controlar. O GPC do controle do componente de eixo direto da corrente de estator tem como saída a tensão de eixo direto de estator. Há ainda dois controladores GPC em cascata: um que gera o componente de eixo em quadratura da tensão de estator, a partir da referência da corrente de eixo em quadratura que, por sua vez, é gerada pelo outro GPC, a partir das referências de velocidade do rotor. Os resultados foram superiores aos tradicionais PI.

Hedjar et al. (2004) apresentaram uma proposta na qual uma estrutura MBPC em cascata é empregada para controlar a velocidade e o torque do MI. Nesta configuração, o torque de carga não é considerado na modelagem. Entretanto, para que o acoplamento de cargas ao eixo do motor não cause erro da velocidade, uma ação integral é adicionada na função objetivo. As simulações mostraram o bom desempenho da proposta de controle, ainda que os parâmetros resistência de estator e de rotor variem. Deve-se salientar que a proposta é desenvolvida com o MBPC contínuo, o que dificulta a implementação em tempo real se forem empregados microcontrolador ou DSP.

3.4 MBPC do motor de indução

Conforme descrito no capítulo dois, uma das etapas do controle preditivo é a predição das variáveis de saída feita com base no modelo da planta, que deve ser escolhido adequadamente para cada implementação. Modelos para plantas lineares com parâmetros invariantes no tempo são facilmente encontrados na literatura (Camacho e Bordons, 1999), entretanto, o modelo do MI é não linear e com parâmetros variantes no tempo.

A modelagem utilizada nesta tese emprega a orientação de campo do rotor e as equações diferenciais do MI são escritas na forma de espaço de estados.

3.4.1 Modelo e controle para o motor de indução

3.4.1.1 Linearização e discretização do modelo do motor de indução em espaços de estado

Deseja-se controlar o fluxo de rotor Φ_r e a velocidade ω_r , empregando a orientação de fluxo, o que significa que a sua componente em quadratura torna-se nula e, portanto, sua magnitude é dada pelo componente direto, isto é, $\Phi_r = \Phi_{rd}$ e, conseqüentemente, o estado Φ_{rq} deve ser retirado da representação do motor de indução, reduzindo a ordem do modelo.

O modelo do MI em espaço de estados é dado por

$$\begin{cases} \dot{x} = f(x) + Bu + \delta = A(x)x + Bu + \delta \\ y = Cx \end{cases} \quad (3.6)$$

na qual

- o vetor de estados $x \in \mathfrak{R}^{(4 \times 1)}$ e é dado por

$$x = [i_{sd} \quad i_{sq} \quad \Phi_{rd} \quad \omega_r]^T;$$

- o vetor de entrada $u \in \mathfrak{R}^{(2 \times 1)}$ e é dado por

$$u = [V_{sd} \quad V_{sq}]^T,$$

sendo V_{sd} e V_{sq} os valores das tensões de estator de eixo direto e eixo em quadratura, respectivamente;

- o vetor de perturbação $\delta \in \mathfrak{R}^{(4 \times 1)}$ e é dado por

$$\delta = [0 \quad 0 \quad 0 \quad -\frac{p}{J}T_c]^T$$

sendo T_c o torque de carga, p o número de pares de pólos e J a inércia do rotor;

- a função $f(x)$ é não linear, que pode ser desmembrada em $A(x)x$, sendo que a matriz característica do sistema $A(x) \in \mathfrak{R}^{(4 \times 4)}$ e é dada por

$$A = \begin{bmatrix} \frac{-1}{\tau'} & \omega_s & \frac{L_m}{L'_s L_r \tau_r} & 0 \\ -\omega_s & \frac{-1}{\tau'} & -\frac{\omega_r L_m}{L'_s L_r} & 0 \\ \frac{L_m}{\tau_r} & 0 & \frac{-1}{\tau_r} & 0 \\ 0 & \kappa \Phi_{rd} & 0 & 0 \end{bmatrix};$$

- A matriz de entrada $B \in \mathfrak{R}^{(p \times 1)}$ e é dada por

$$B = \begin{bmatrix} 1/L'_s & 0 \\ 0 & 1/L'_s \\ 0 & 0 \\ 0 & 0 \end{bmatrix};$$

- O vetor de saída $y \in \mathfrak{R}^{(2 \times 1)}$ e é dada por

$$y = [\Phi_{rd} \quad \omega_r]^T;$$

- A matriz de saída $C \in \Re^{(2 \times 4)}$ e depende das saídas escolhidas a serem controladas, portanto,

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.7)$$

Nas matrizes anteriores têm-se que

- L'_s é a indutância transitória do estator;
- $\tau_r = L_r/R_r$ é a constante de tempo do rotor;
- $\tau'_s = L'_s/R_s$ e $\tau'_r = L'_r/R_r$ são as constantes de tempo transitórias do estator e do rotor, respectivamente. Com $L'_r = \sigma L_r$;
- ω_s é a velocidade síncrona;
- $\kappa = (3/2)(p/J)(L_m/L_r)$.

Deve-se salientar que apesar de existirem formulações para o MBPC na forma contínua, ele é tradicionalmente implementado na forma discreta, na qual as matrizes do sistema A_d , de entrada B_d e de saída C_d são dadas pela equação (2.13). Nesta tese é adotada uma aproximação de primeira ordem, que é obtida ignorando-se os últimos termos de A_d e B_d desta mesma equação; então, para que a representação discreta tenha um comportamento similar ao modelo contínuo, devem ser adotados tempos de amostragem significativamente menores que as constantes de tempo do modelo. O modelo da planta discretizado e linearizado é dado por

$$\begin{cases} x_{k+1} = A_{dl}(x'_k)x_k + B_d u_k + D(x_k, \delta_k) \\ y_{k+1} = C_d x_{k+1} \end{cases} \quad (3.8)$$

sendo que no caso do MI

$$A_{dl} = \begin{bmatrix} 1 - \frac{T_a}{\tau'} & T_a \omega_s & \frac{T_a L_m}{L'_s L_r \tau_r} & 0 \\ -T_a \omega_s & 1 - \frac{T_a}{\tau'} & -\frac{T_a \omega'_r L_m}{L'_s L_r} & -\frac{T_a L_m \Phi'_{rd}}{L'_s L_r} \\ \frac{T_a L_m}{\tau_r} & 0 & 1 - \frac{-T_a}{\tau_r} & 0 \\ 0 & T_a \kappa \Phi'_{rd} & T_a \kappa i'_{sq} & 1 \end{bmatrix} \quad (3.9)$$

e o vetor $D(x_k, \delta_k)$ é

$$D(x_k, \delta_k) = \begin{bmatrix} 0 \\ \frac{T_a L_m \Phi'_{rd} \omega'_r}{L'_s L_r} \\ 0 \\ -T_a \kappa i'_{sq} \Phi'_{rd} - T_a \frac{p}{J} T_c \end{bmatrix} \quad (3.10)$$

Na representação por espaço de estados dada por (2.22), o torque de carga T_c é tratado como uma perturbação, e, como ele se relaciona com os estados da máquina o seu valor pode ser obtido, a partir dos estados conhecidos: i_{sd} , i_{sq} , Φ_{rd} e ω_r . Ao se empregar a equação eletromecânica do motor na sua forma discreta, o torque de carga é dado por

$$T_{c(k)} = \left[\frac{L_m}{L_r} \frac{3}{2} i_{sq(k)} \Phi_{rd(k)} \right] - \left[\frac{J}{p} \frac{(\omega_{r(k+1)} - \omega_{r(k)})}{T_a} \right] \quad (3.11)$$

Nota-se que o segundo termo depende da inércia total do sistema motor-carga e da taxa de variação da velocidade. Supondo que a inércia da carga é desconhecida e, para simplificar, considerando-se que a inércia total é própria inércia do motor, tem-se que, durante o transitório, este segundo termo é estimado erroneamente; durante o regime permanente o segundo termo tem magnitude nula ($\omega_{r(k+1)} - \omega_{r(k)} = 0$), não influenciando no cálculo do torque da carga. Ressalte-se que a estimação de T_c realizada com a equação (3.11) tem melhor precisão durante os transitórios se o perfil de velocidade do motor tem variações mais lentas, como por exemplo, trapezoidal; mudanças mais abruptas na velocidade, como no perfil retangular, causarão durante o período transitório erros maiores na estimação.

A presença da velocidade síncrona ω_s nas equações (3.9) torna o modelo, que já é não linear, variante no tempo. Como ω_e é a velocidade de escorregamento e que $\bar{\Phi}_{r,dq} = \Phi_{rd}$, a frequência síncrona ω_s pode ser obtida através de

$$\omega_s = \omega_r + \omega_e = \omega_r + \frac{L_m}{\tau_r} \frac{i_{sq}}{\Phi_{rd}} \quad (3.12)$$

3.4.1.2 Diagrama esquemático do acionamento do MI com o MBPC

Um esquema para controle da máquina de indução é exibido na Figura 3.3 (de Santana et al., 2006). Neste trabalho as correntes e tensões de estator são medidas, enquanto o vetor fluxo de rotor e a velocidade são determinados pelo FKE.

O controle pode ser sintetizado nas seguintes etapas:

Etapas 1 - Medição e transformação abc para $\alpha - \beta$

A partir das tensões de linha $V_{s,ab}$ e $V_{s,bc}$ e das correntes de fase i_{sa} e i_{sb} medidas são obtidas as variáveis bifásicas no sistema estacionário, empregando as seguintes equações:

$$\begin{cases} V_{s\alpha} = \frac{2}{3}V_{s,ab} + \frac{1}{3}V_{s,bc} \\ V_{s\beta} = \frac{1}{\sqrt{3}}V_{s,bc} \end{cases} \quad (3.13)$$

$$\begin{cases} i_{s\alpha} = i_{sa} \\ i_{s\beta} = \frac{1}{\sqrt{3}}(i_{sa} + 2i_{sb}) \end{cases} \quad (3.14)$$

Etapa 2 - *Estimação de fluxos e transformação $\alpha - \beta$ para $d - q$*

As tensões e correntes transformadas obtidas na etapa anterior são as entradas do filtro de Kalman estendido, que tem a função de estimar os fluxos de rotor $\Phi_{r\alpha}$ e $\Phi_{r\beta}$ e da velocidade.

Conhecidos os componentes $\Phi_{r\alpha}$ e $\Phi_{r\beta}$, determinam-se as funções trigonométricas

$$\begin{cases} \sin \theta_{\Phi_r} = \frac{\Phi_{r\beta}}{\sqrt{\Phi_{r\alpha}^2 + \Phi_{r\beta}^2}} \\ \cos \theta_{\Phi_r} = \frac{\Phi_{r\alpha}}{\sqrt{\Phi_{r\alpha}^2 + \Phi_{r\beta}^2}} \end{cases} \quad (3.15)$$

que são empregadas nas transformações $\alpha\beta$ para dq , dadas por

$$\begin{cases} V_{sd} = V_{s\alpha} \cos \theta_{\Phi_r} + V_{s\beta} \sin \theta_{\Phi_r} \\ V_{sq} = -V_{s\alpha} \sin \theta_{\Phi_r} + V_{s\beta} \cos \theta_{\Phi_r} \end{cases} \quad (3.16)$$

$$\begin{cases} i_{sd} = i_{s\alpha} \cos \theta_{\Phi_r} + i_{s\beta} \sin \theta_{\Phi_r} \\ i_{sq} = -i_{s\alpha} \sin \theta_{\Phi_r} + i_{s\beta} \cos \theta_{\Phi_r} \end{cases} \quad (3.17)$$

$$\begin{cases} \Phi_{rd} = \Phi_{s\alpha} \cos \theta_{\Phi_r} + \Phi_{s\beta} \sin \theta_{\Phi_r} \\ \Phi_{rq} = -\Phi_{s\alpha} \sin \theta_{\Phi_r} + \Phi_{s\beta} \cos \theta_{\Phi_r} \end{cases} \quad (3.18)$$

Etapa 3 - *Linearização das equações de estado*

Nesta etapa o modelo discreto é linearizado, como descrito na página 15.

Etapa 4 - *Predição do sistema linearizado variante no tempo*

A predição é realizada com o emprego modelo não variacional é dada pela equação (2.16).

Etapa 5 - *Determinação dos sinais de controle*

Encontram-se as os incrementos futuras $\Delta V_{sd}(k+j-1/k)$ e $\Delta V_{sq}(k+j-1/k)$, para $j = 1, \dots, n_u$, que minimizem a função objetivo J ; nesta tese emprega-se $n_u = 1$,

$n_y = 2$ e $T_a = 6$ ms. A função objetivo empregada é a quadrática, como a da equação (2.20).

Os incrementos de controle são somados aos valores anteriores das tensões, obtendo-se assim os valores de tensão de eixo direto e em quadratura a serem aplicados ao MI.

As etapas 3, 4 e 5 são realizadas de uma única vez, pois como o sistema é linearizado tem-se uma lei de controle (equações (2.23)), na qual todos estes procedimentos estão incluídos, pois para horizontes de predição pequenos pode-se realizar cálculos, como multiplicações ou inversão de matrizes, antecipadamente.

Etapa 6 - Aplicação dos sinais de controle V_{sd} e V_{sq}

Escolheu-se a estratégia de horizonte deslizante e aplicou-se apenas o primeiro valor calculado na etapa anterior. Na realidade, estes valores ainda são transformados para as coordenadas estacionárias, obtendo-se $V_{s\alpha}$ e $V_{s\beta}$, que são as entradas do algoritmo SV-PWM.

$$\begin{cases} V_{s\alpha} = V_{sd} \cos \theta_{\Phi_r} - V_{sq} \sin \theta_{\Phi_r} \\ V_{s\beta} = V_{sd} \sin \theta_{\Phi_r} + V_{sq} \cos \theta_{\Phi_r} \end{cases} \quad (3.19)$$

Uma vez aplicadas as tensões retorna-se à etapa 1.

3.5 Conclusões

Com o objetivo de melhorar as respostas dinâmicas do motor tem-se procurado alternativas aos tradicionais PI, sendo uma delas, os controladores preditivos.

Geralmente, na área de acionamentos elétricos, os controladores preditivos referem-se a propostas que não permitem levar em consideração as implicações futuras das escolhas dos sinais de controle atuais, já o MBPC é formulado levando em consideração um horizonte futuro que é escolhido pelo projetista do sistema de controle, permitindo incluir até as restrições que a planta possua.

A aplicação do MBPC ao acionamento do motor de indução pode ser realizada de maneira mais eficaz se o modelo tiver a orientação de fluxo; esta orientação permite também a diminuição da ordem do modelo, diminuindo o custo computacional do algoritmo de controle. Para a presente tese escolheu-se a orientação do fluxo do rotor, cuja magnitude e posição espacial é obtida com o emprego do filtro de Kalman estendido.

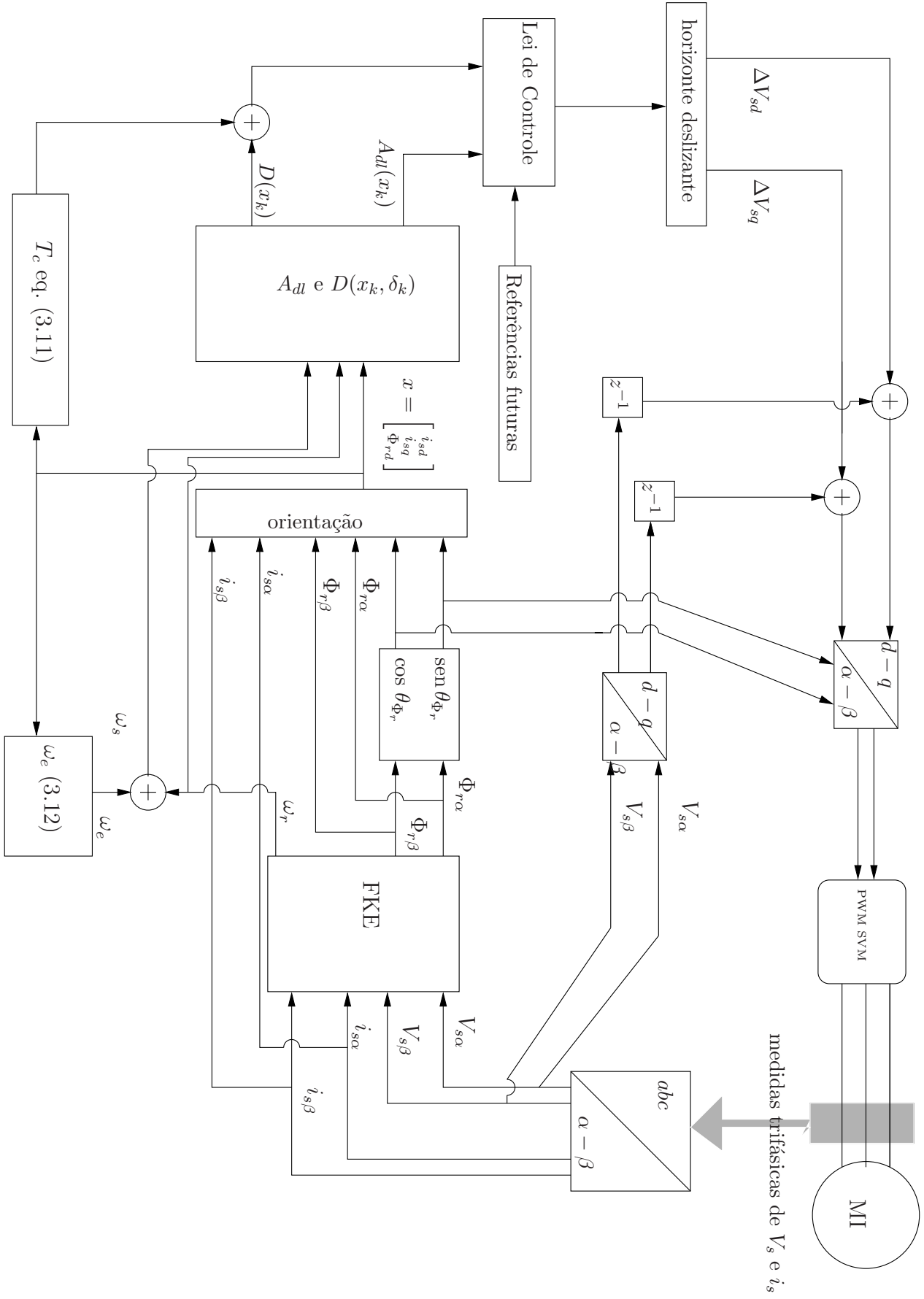


Figura 3.3: Diagrama de bloco representativo do acionamento proposto.

CAPÍTULO 4

Resultados de Simulações

4.1 Introdução

Neste capítulo são apresentados e analisados os resultados de simulações numéricas. As seções que compõem este capítulo são as seguintes:

- *Seção 4.2:*

Apresentação dos resultados do controlador preditivo com cálculo dos acréscimos de tensão, ou seja, usando o modelo variacional. São mostrados os resultados do modelo com FKE estimando o fluxo de rotor e a velocidade (com a aproximação $\dot{\omega}_r \approx 0$). Foram realizadas simulações para as condições a vazio e com torque mecânico aplicado ao eixo. Quando emprega-se a aproximação $\dot{\omega}_r \approx 0$, o torque de carga é calculado a partir da equação eletromecânica do MI. Saliente-se que os modelos empregados no FKE e no MBPC em todas as simulações estão em p.u.

São apresentados também os resultados para a estimação do torque de carga pelo filtro de Kalman estendido, considerando T_c como um estado adicional e com dinâmica lenta, isto é $\dot{T}_c \approx 0$;

- *Seção 4.3:*

Apresentação de alguns resultados do MBPC não variacional.

- *Seção 4.4:*

As conclusões sobre as simulações são sumarizadas na conclusão.

4.2 Resultados para o MBPC variacional

As simulações foram realizadas com os parâmetros descritos no Apêndice B e emprega o programa Simulink/Matlab. O modelo discretizado e linearizado, conforme descrito na seção 3.4 é o empregado no MBPC para acionamento do MI; calcula-se, como sinal de controle, o acréscimo de tensão ΔV_{sd} e ΔV_{sq} a serem adicionados aos atuais valores de tensão de eixos direto e em quadratura.

Para todas as simulações foi utilizado um horizonte de predição n_y igual a 2 e um horizonte de controle n_u igual a 1; esta escolha tem como objetivo diminuir o custo computacional do algoritmo preditivo, pois, a seleção de valores maiores para estes horizontes, aumenta as ordens das matrizes H' , H'' e H''' . O controle atua em ciclos de tempo de 6 ms, valor que é significativamente menor que as constantes de tempo do motor.

A matriz que pondera as saídas controladas é a matriz identidade $W_y = I_{4 \times 4}$ e aquela que pondera o esforço de controle é dada por

$$W_u = \begin{bmatrix} 0,15 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} W_{u/V_{sd}} & 0 \\ 0 & W_{u/V_{sq}} \end{bmatrix}$$

sendo que

- $W_{u/V_{sd}}$ a matriz que pondera o esforço de controle da entrada ΔV_{sd} e;
- $W_{u/V_{sq}}$ a matriz que pondera o esforço de controle da entrada ΔV_{sq} .

As figuras que mostram os resultados são divididas em quatro gráficos (a, b, c e d) cuja descrição é dada a seguir:

- (a) A curva na cor verde representa a corrente de estator de eixo direto; as curvas na cores vermelha, azul e preta, representam, a referência, o valor real e o valor estimado pelo FKE do fluxo de rotor, respectivamente,;
- (b) As curvas nas cores vermelha, azul e preto, representam, respectivamente, a referência, o valor real e o valor estimado da velocidade;
- (c) A curva na cor preta representa a corrente de estator em quadratura; nos testes de carga é mostrada simultaneamente o torque de carga real em roxo e o estimado em verde;
- (d) A curva na cor azul é a tensão de estator em quadratura e aquela na preta a tensão de eixo direto.

4.2.1 Testes a vazio com diversos perfis de velocidade

4.2.1.1 Perfil de velocidade retangular

Na Figura 4.1 é mostrado comportamento de variáveis elétricas e mecânicas durante a reversão de velocidade (de 600 rpm a -600 rpm) com o MI a vazio. Na Figura 4.1(b) é destacado (região circulada) a característica antecipativa do controlador preditivo; o motor começa a mudar de velocidade com uma antecedência de $n_y T_a$, que no presente caso é de 12 ms, antes da modificação da referência.

Nota-se na Figura 4.1(a) que, apesar da mudança abrupta na velocidade, o fluxo de rotor mantém-se próximo à referência, exceto nos transitórios de velocidade.

Na orientação pelo fluxo do rotor tem-se uma relação direta entre Φ_{rd} e i_{sd} ; o comportamento de ambas as variáveis diferencia-se apenas em velocidades próximas a zero, devido à imprecisão da estimação de estado nesta situação. Resultados de simulações exibidos posteriormente mostram detalhadamente o comportamento das variáveis com velocidade nula.

Apesar do bom desempenho do controle do fluxo de rotor (mostrado na Figura 4.1(a)) é necessário salientar que a corrente do motor é muito maior que a nominal; as correntes de estator i_{sq} e i_{sd} alcançam valores máximos em torno 12A e 4,8 A, respectivamente, no momento na mudança de referência. Salienta-se que o valor nominal para o motor em questão é de 4,86 A.

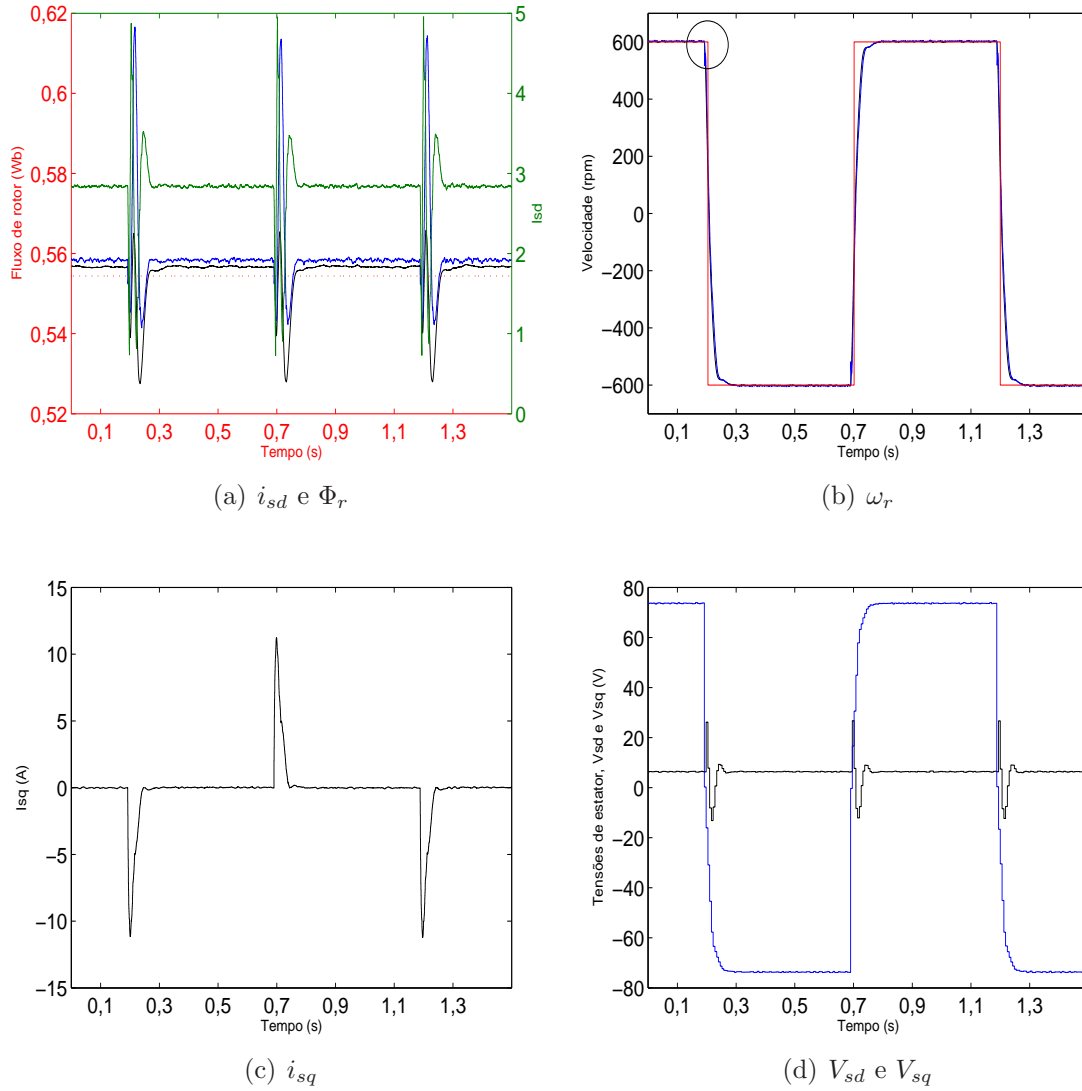


Figura 4.1: Teste com perfil de velocidade retangular de 600 rpm à -600 rpm com motor a vazio.

4.2.1.2 Perfil de velocidade trapezoidal

Quando se deseja alterar a velocidade do MI, ou mesmo revertê-la, usualmente emprega-se um perfil de velocidade menos rigoroso que o retangular (mostrado na seção anterior). Uma alternativa é o perfil trapezoidal, com reversão de velocidade, como está mostrado na Figura 4.2(b); a referência de velocidade muda de 600 rpm a -600 em 0,5 s.

Nota-se na Figura 4.2(a) que a corrente de eixo direto, que se relaciona de forma proporcional ao fluxo quando o motor está orientado, praticamente mantém seu valor em 2,8 A, enquanto que a componente i_{sq} (Figura 4.2(c)) atinge valores máximos, em módulo, em torno de 0,75 A, significativamente menores que os 12 A do teste de reversão retangular, mostrado na seção anterior.

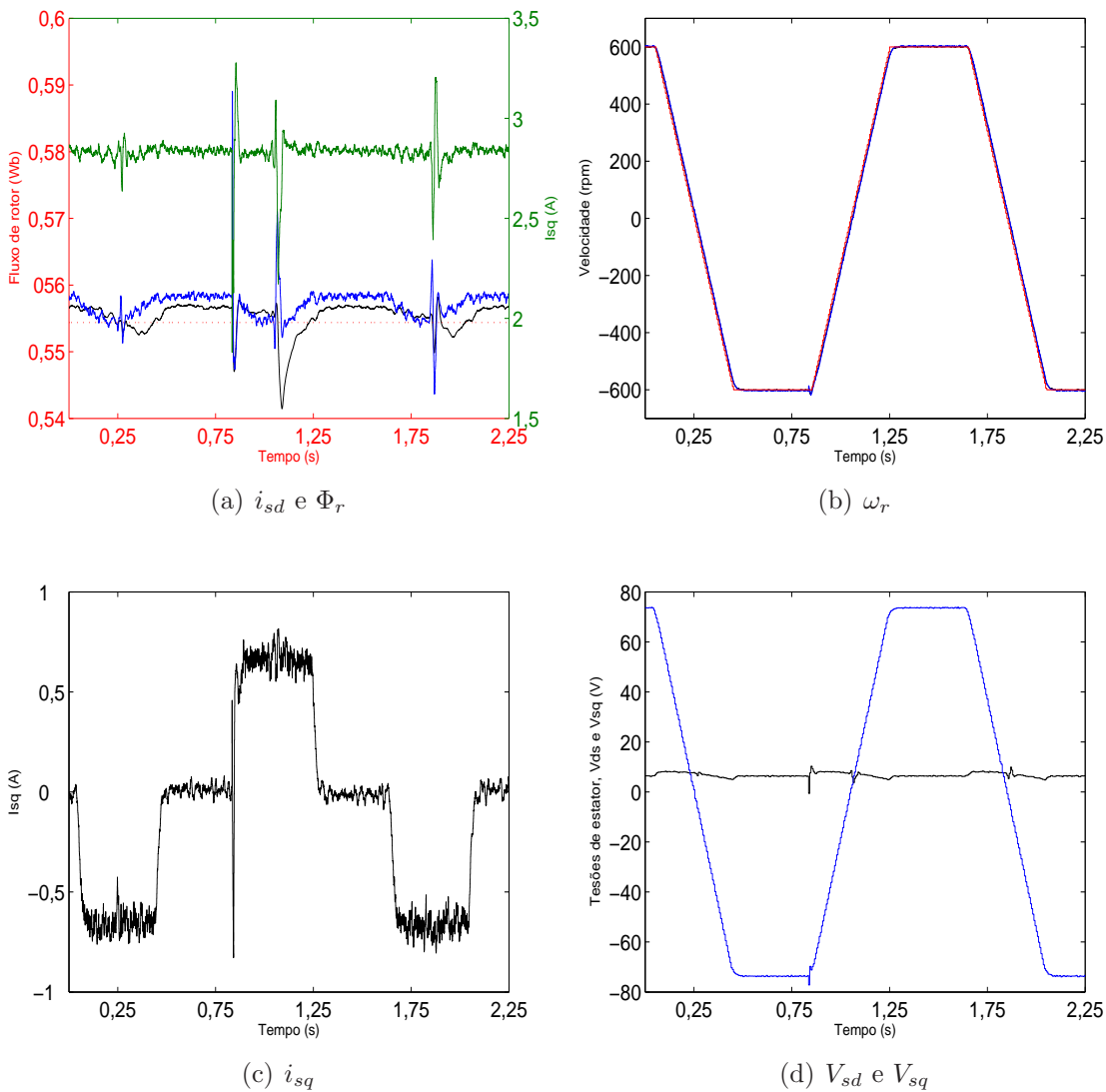


Figura 4.2: Teste com perfil de velocidade trapezoidal de 600 rpm à -600 rpm com motor a vazio.

4.2.1.3 Perfil de velocidade triangular

Os resultados da reversão de velocidade de 600 rpm à -600 rpm em 0,5 s são mostrados na Figura 4.3. A magnitude de i_{sq} é de 0,75 A.

Nota-se na área circunscrita da Figura 4.3(b) que o motor não alcança a velocidade de 600 rpm; isto ocorre pela característica antecipativa do MBPC, que com 12 ms já é informado da mudança de referência. Caso deseje-se que o motor chegue mais próximo dos 600 rpm deve-se diminuir o esforço de controle da velocidade (aumentando o elemento $W_{u/V_{sq}}$) ou diminuir a taxa de atuação de controle, de tal forma, que o algoritmo não antecipe demasiadamente a mudança de referência.

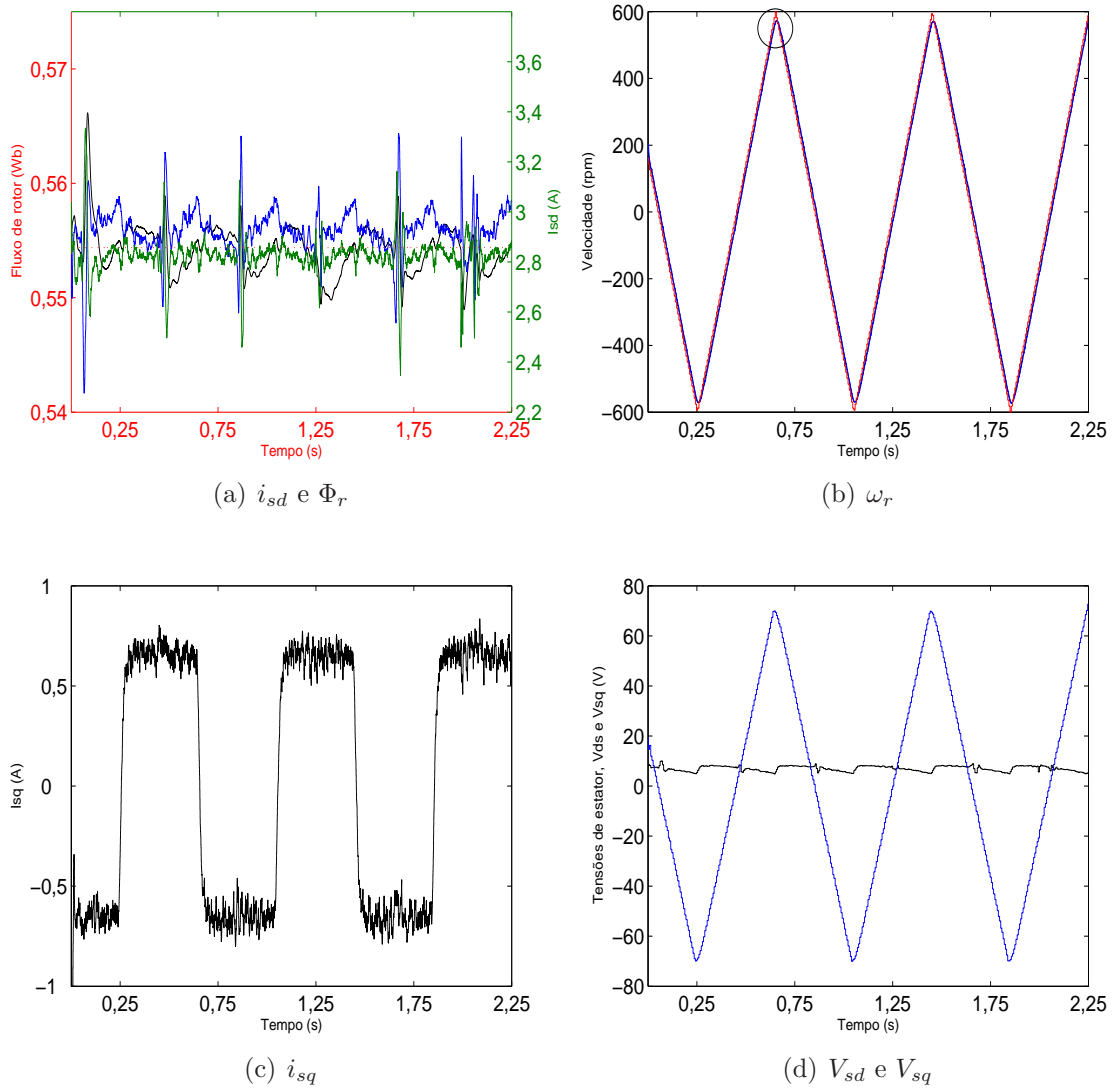


Figura 4.3: Teste com perfil de velocidade triangular de 600 rpm à -600 rpm, com motor a vazio.

4.2.1.4 Teste em baixa velocidade

O desempenho dos acionamentos realizados com MI geralmente são deteriorados quando opera-se com velocidade reduzida. A Figura 4.4 mostra a operação do MI em 30 rpm e 0 rpm. Vê-se que a velocidade é mantida em torno do valor desejado porém com erro de mais ou menos 7 rpm. Caso se deseje maior precisão deve-se realizar algum refinamento na estimação de estado, de maneira que a velocidade seja melhor estimada e empregar uma discretização mais precisa do modelo MBPC.

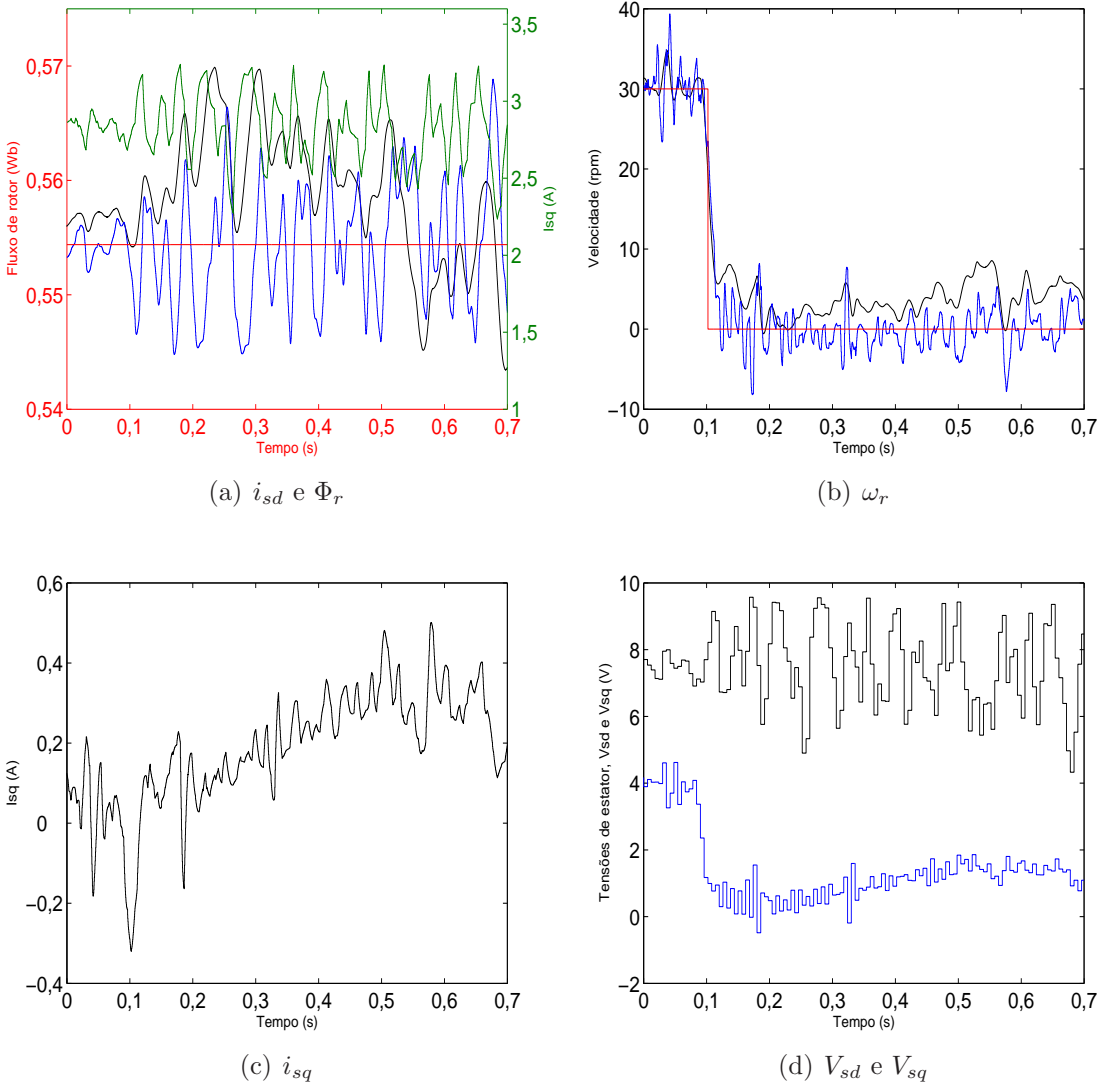


Figura 4.4: Teste com perfil de velocidade retangular de 30 rpm à 0 rpm com motor a vazio.

4.2.2 Testes com carga

4.2.2.1 Teste com $T_c \propto \omega_r^2$ e com T_c calculado

Neste teste é simulado o acoplamento ao eixo do MI de uma carga, cuja inércia é $2J_m$ e cuja característica do seu torque é proporcional ao quadrado da velocidade; T_c é sempre

frenante, pois inverte seu sinal juntamente com a velocidade, atingindo o valor nominal de 12,3 N.m aos 600 rpm, como mostrado na Figura 4.5(c). Saliente-se que os modelos empregados no controlador e no FKE consideram apenas a inércia do motor.

A estimação de T_c é precisa apenas em regime permanente, pelos motivos já discutidos na seção 3.5.

Vê-se na Figura 4.5(b) que apesar de T_c nominal ser aplicado ao motor, a regulação da velocidade em regime permanente é satisfatória.

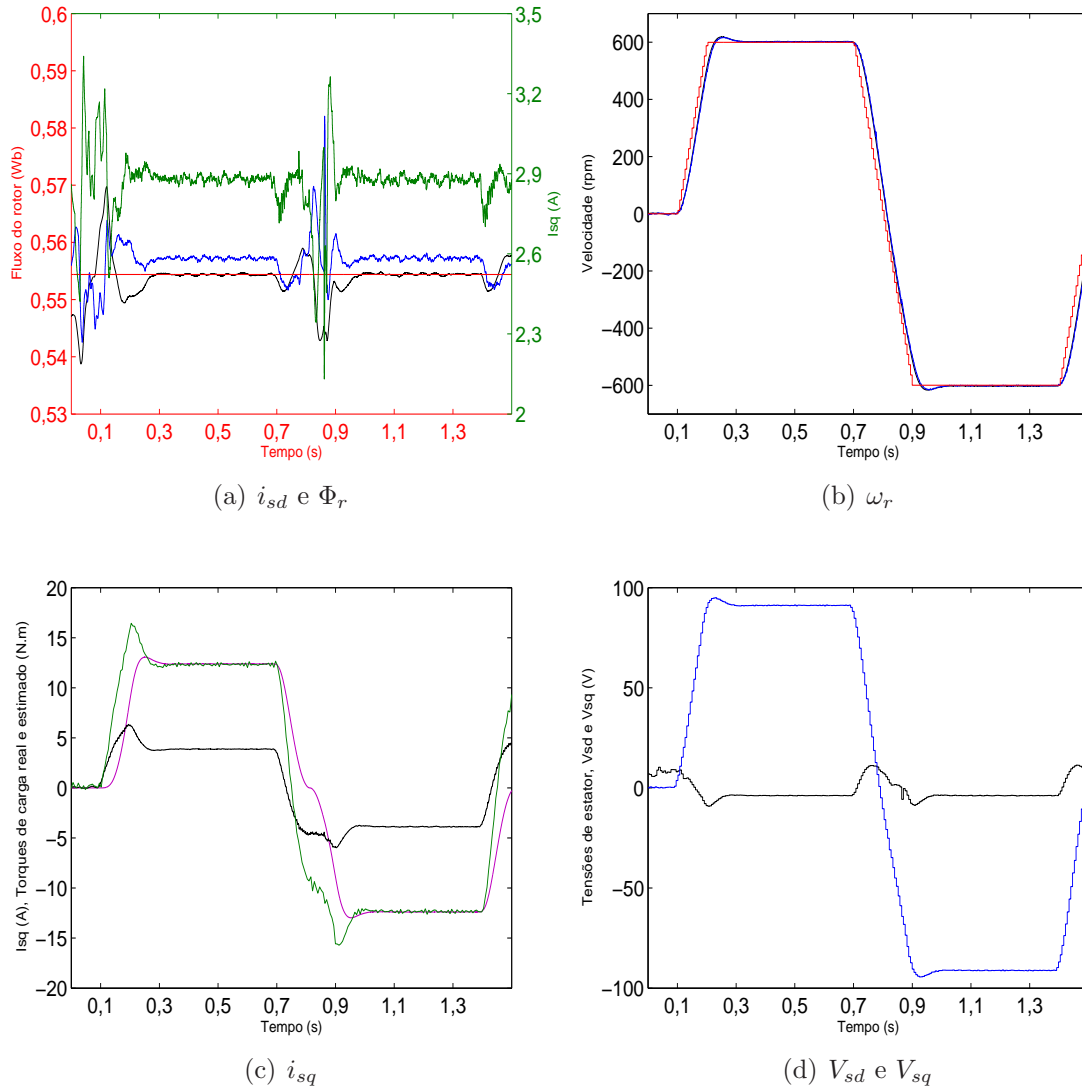


Figura 4.5: Teste com perfil de velocidade trapezoidal de 600 rpm à -600 rpm, com carga proporcional ao quadrado da velocidade.

4.2.2.2 Teste com degrau de carga nominal ao eixo do motor e com T_c calculado

Neste teste, é acoplada abruptamente uma carga com inércia $2J_m$ e $T_c = 12,3$ N.m (valor nominal) em $t = 0,1$ s.

Nota-se na Figura 4.6(b) a redução da velocidade do momento de aplicação da carga mecânica. Caso fosse possível prever a entrada da carga, o desempenho do controlador seria ainda melhor, pois seriam gerados sinais de controle que aumentassem a velocidade antes da entrada da carga, minimizando a variação de ω_r no instante do acoplamento da carga.

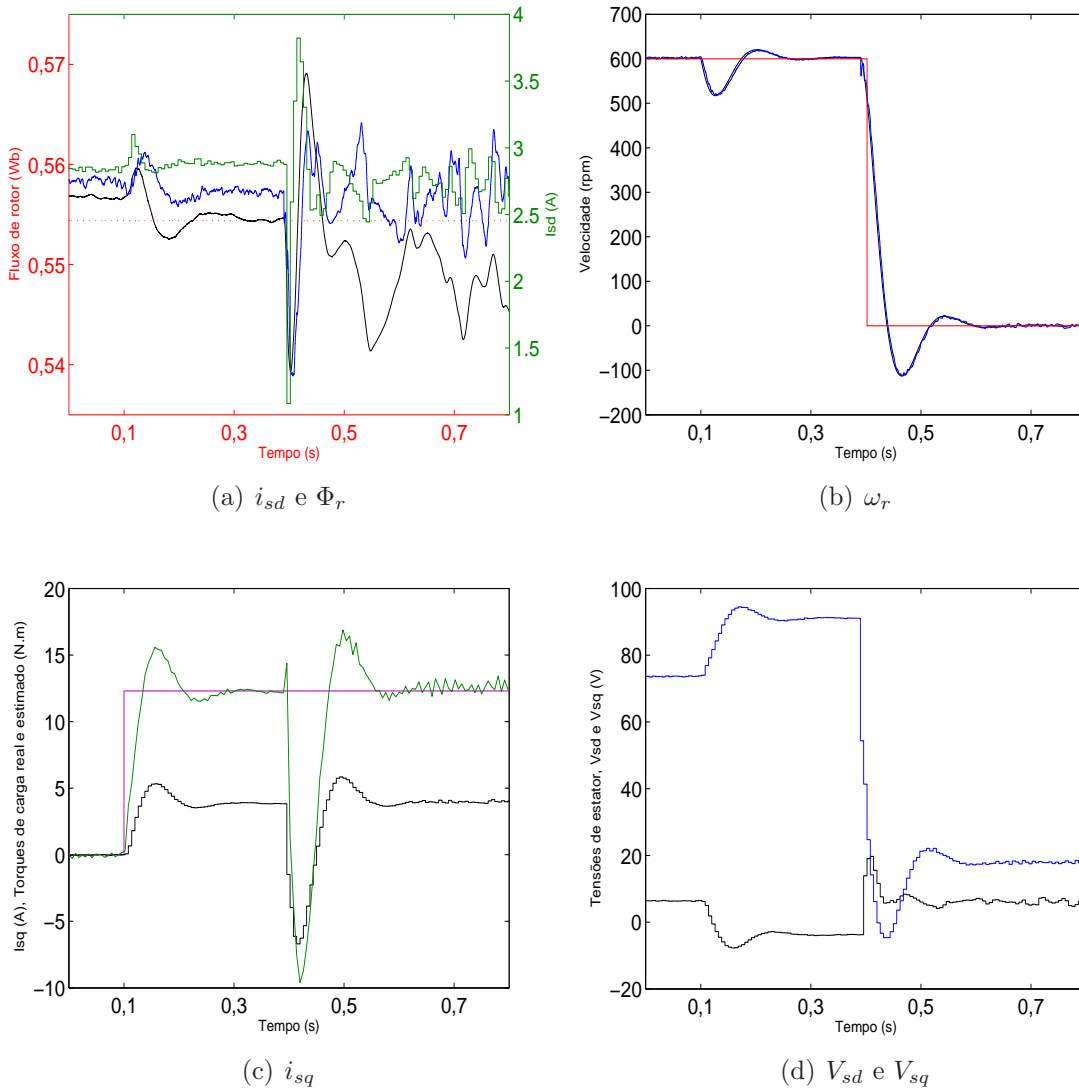


Figura 4.6: Teste com perfil de velocidade retangular de 600 rpm à 0 rpm, com carga nominal (12,3 N.m) aplicada ao eixo no instante 0,1 s.

4.2.2.3 Teste com $T_c \propto \omega_r^2$ e T_c estimado com o FKE

Uma proposta alternativa à anterior, na qual o torque de carga é estimado a partir do conhecimento das variáveis de estado, é estimá-lo diretamente pelo filtro de Kalman estendido. Como considera-se que o comportamento de T_c é desconhecido, fez-se a aproximação que sua dinâmica é lenta, de maneira que sua magnitude não varia significativamente entre duas amostragens consecutivas. Os resultados mostrados na Figura 4.7 evidenciam que o desempenho desta abordagem é similar ao de T_c calculado mostrado na subseção 4.2.2.2.

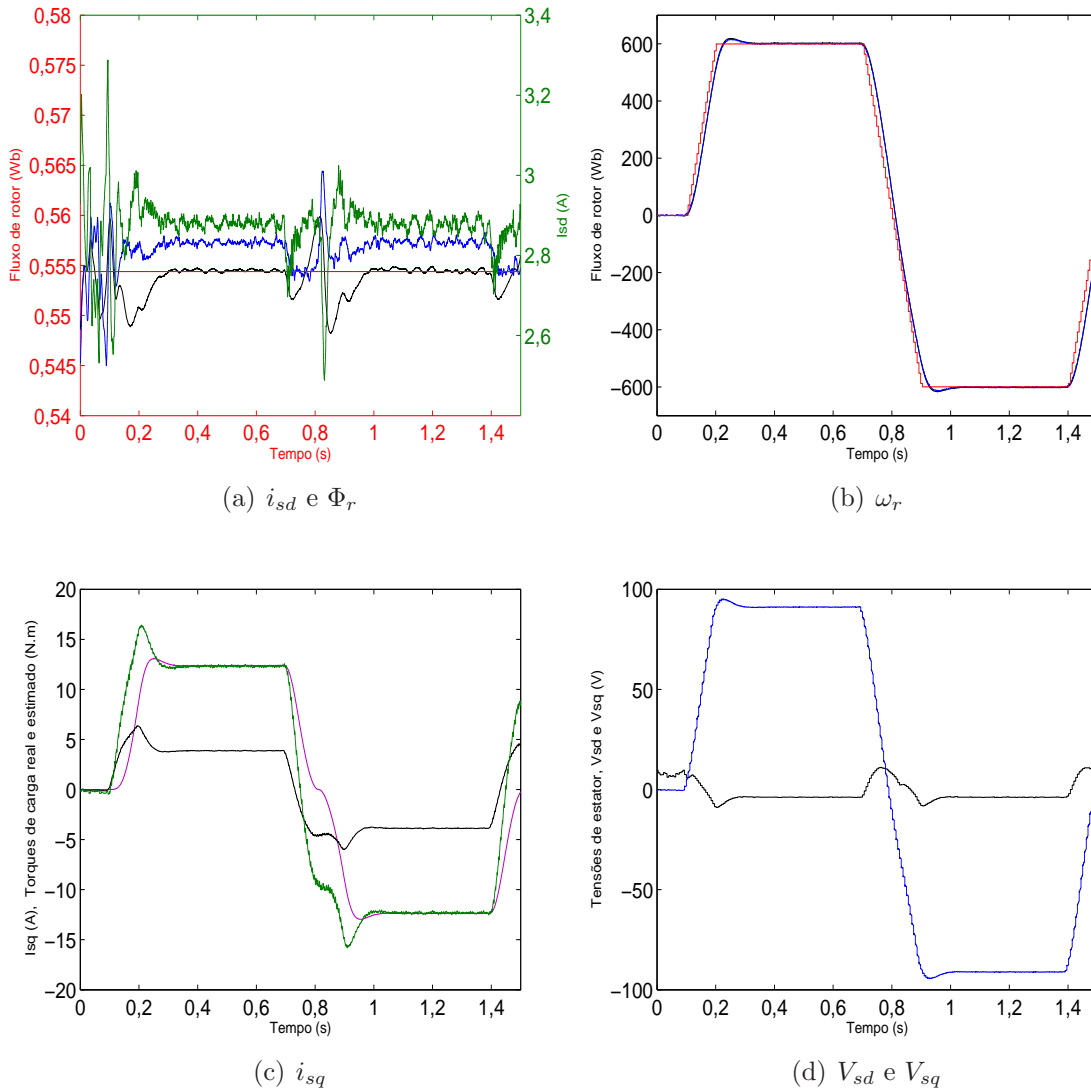


Figura 4.7: Teste com perfil de velocidade trapezoidal de 600 rpm à -600 rpm, com carga proporcional ao quadrado da velocidade.

4.2.2.4 Teste com degrau de carga nominal e T_c estimado com o FKE

A Figura 4.8 mostra o comportamento das variáveis para a inclusão abrupta de inércia e variação em degrau do torque de carga. Adiciona-se $2J_m$ a inércia do motor e $T_c = 12,3$

N.m. Os resultados evidenciam a boa resposta do sistema, ainda que haja discordância entre o valor total da inércia (que é de $3J_m$) e o valor que é empregado (J_m) tanto pelo FKE quanto pelo MBPC.

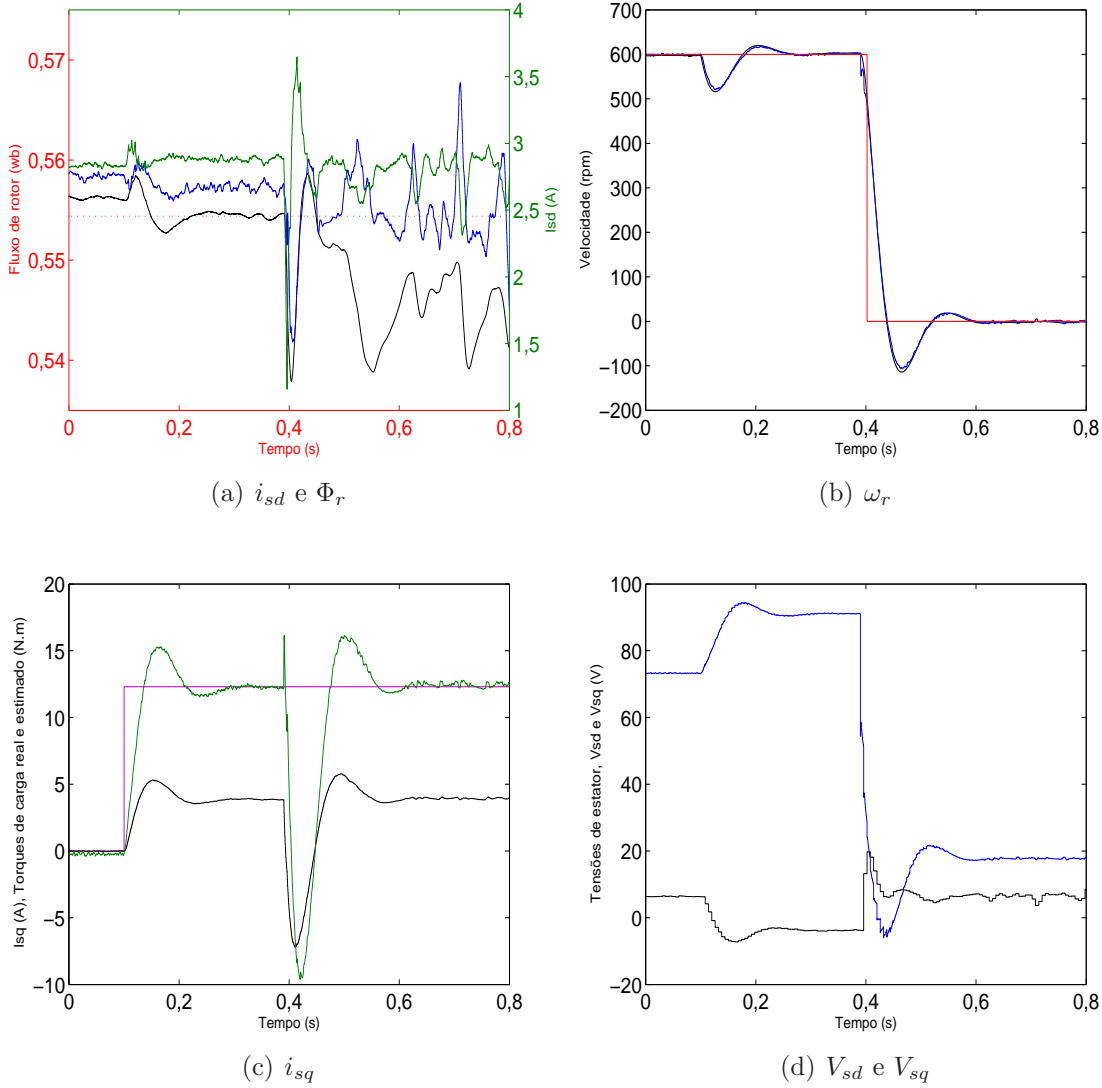


Figura 4.8: Teste com perfil de velocidade retangular de 600 rpm à 0 rpm, com carga nominal (12,3 N.m) aplicada ao eixo no instante 0,1 s.

4.2.3 Teste de variação de fluxo

4.2.3.1 Redução de fluxo de rotor

Apesar da maioria das aplicações de acionamentos do MI serem implementadas com fluxo constante para garantir o melhor aproveitamento da sua capacidade de torque, algumas estratégias utilizam o controle do fluxo para melhorar o rendimento do motor ou mesmo para aumentar a velocidade (Xu e Novotny, 1992). O MBPC empregado mostrou-se eficaz no controle do fluxo de rotor, como é mostrado na Figura 4.9(a), já que a referência

foi modificada do seu valor nominal (0,565 Wb) a 60% deste valor, 0,34 Wb, (em 0,1 s) e os fluxos medido e estimado acompanharam esta mudança da forma satisfatória.

Durante o transitório do fluxo, há a variação máxima de 4,3 % no valor da velocidade do rotor, cujo valor de referência é de 602 rpm.

Para melhorar o desempenho das estratégias de controle por orientação com o MI alimentado por uma fonte de tensão, é necessário empregar artifícios matemáticos para desacoplar as tensões e correntes; sendo assim,

- V_{sd} influencia apenas i_{sd} e, por conseqüência, controla diretamente Φ_r ;
- V_{sq} influencia apenas i_{sq} e, por conseqüência, se o fluxo for mantido constante, controle diretamente o torque eletromagnético.

Diferentemente destes acionamentos, o controle multivariável realiza o controle de fluxo e velocidade diretamente, pela manipulação de ambas tensões, como está mostrado na Figura 4.9(d).

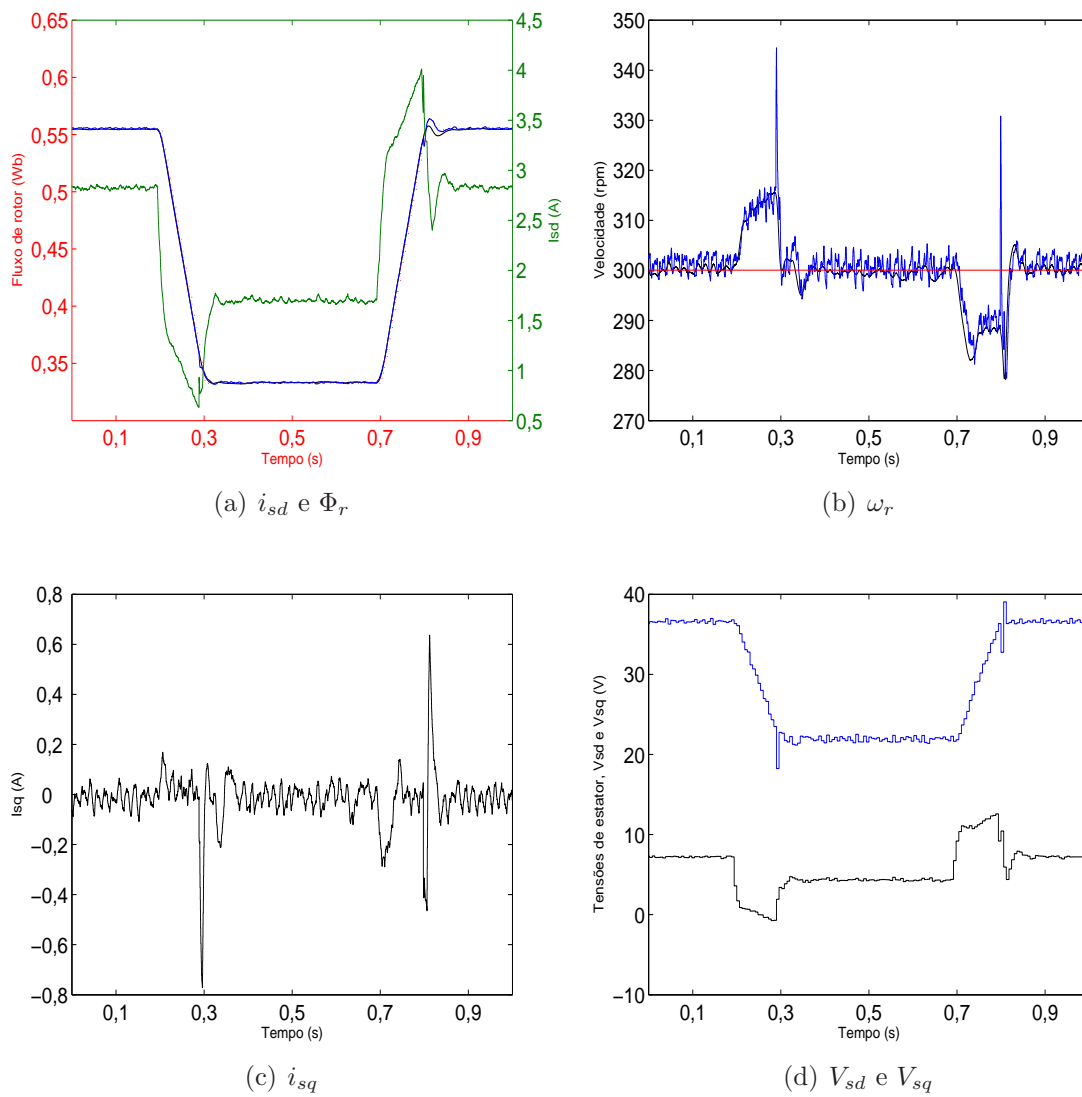


Figura 4.9: Variação de fluxo; de 0,565 Wb a 0,34 Wb.

4.2.4 Testes com mudança do esforço de controle e com filtro na referência

Na Figura 4.10 é mostrado um teste de reversão para o MI a vazio e com a matriz do esforço de controle dada por

$$W_u = \begin{bmatrix} 0,15 & 0 \\ 0 & 0,2 \end{bmatrix}$$

Nota-se que, o termo que pondera a tensão em quadratura, é cinco vezes menor que dos testes anteriores, o que significa maior esforço de controle. Vê-se na Figura 4.10(b) que ocorre um pequeno *overshoot* na resposta, e, este fato, leva a corrente em quadratura a atingir 14 A, muito acima do valor nominal, como mostrado na Figura 4.10(c). Estes altos valores devem-se a variação abrupta das tensões de eixo direto e em quadratura, que por sua vez, são elevadas devido ao tempo do ciclo de controle ser alto ($T_a = 6$ ms), significando que altos ganhos têm de ser gerados pelo controlador, para que a referência seja atingida em um tempo razoável.

Uma solução para estes altos valores é a inclusão de um filtro digital passa-baixa, que suaviza a referência. Na Figura 4.11(b) esta mostrado o desempenho do sistema de controle que emprega o filtro

$$w^{sua}(k+j) = \gamma w^{sua}(k+j-1) + (1-\gamma)w(k+j) \quad (4.1)$$

Na qual

- w e w^{sua} , denotam, respectivamente, as referências real e suavizada;
- γ é o parâmetro que ajusta o filtro e seu intervalo de variação é de $0 < \gamma < 1$, sendo que quanto mais próximo de 1 mais suave torna-se a curva. Para a simulação em questão empregou-se $\gamma = 0,75$.

No resultado da Figura 4.11(c), nota-se que o valor máximo de i_{sq} ficou em torno de 5 A, significativamente menor que da simulação sem referência filtrada.

A comparação dos gráficos 4.10(b) e 4.11(b) de velocidade, mostra que com filtro a inversão acontece mais lentamente, o que permitiu também uma melhor estimação por parte do FKE, pois, como já explicado, adotou-se a aproximação $\dot{\omega}_r \approx 0$.

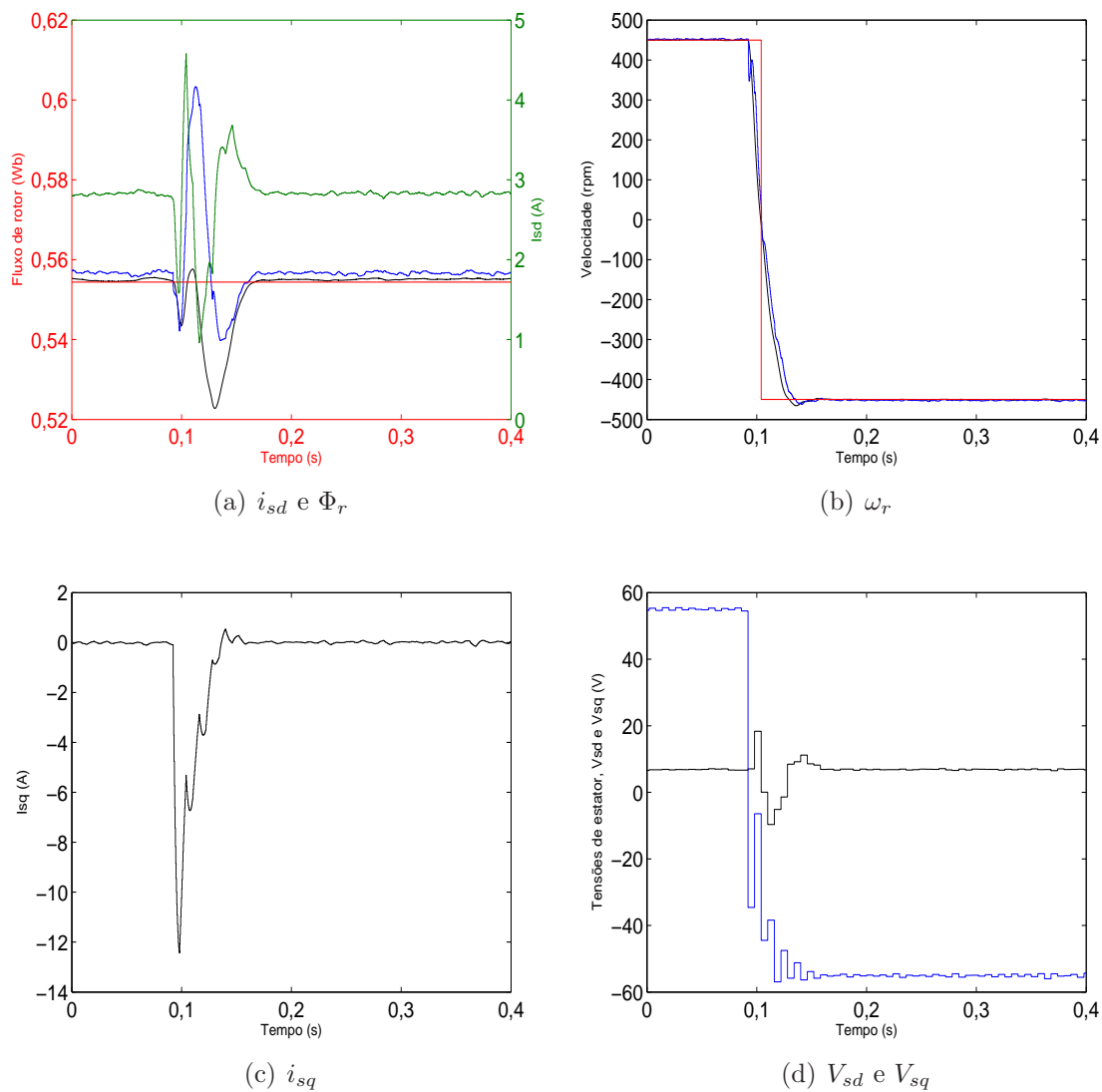


Figura 4.10: Teste com perfil de velocidade retangular de 450 rpm à -450 rpm, com motor a vazio.

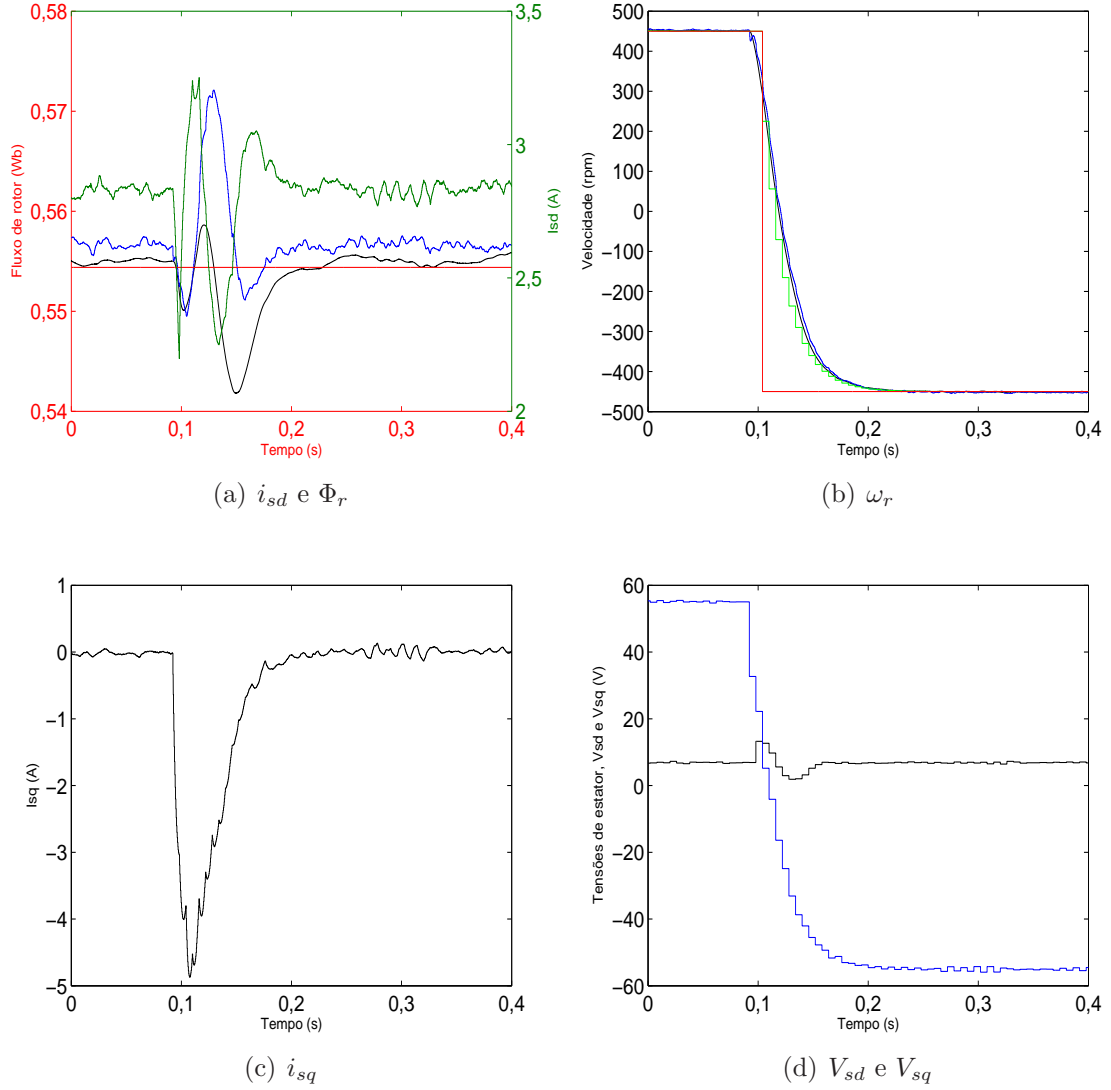


Figura 4.11: Teste com perfil de velocidade retangular de 450 rpm à -450 rpm com motor a vazio e referência filtrada.

4.3 MBPC não variacional

Pode-se também empregar o MBPC, em que os sinais de controle são as tensões que devem ser aplicadas e não o acréscimo ao valor atual. Na Figura 4.12 são mostrados os resultados da inversão de velocidade do MBPC não variacional, com ciclo de controle de $T_a=10$ ms, $n_y = 2$, a matriz de ponderação do esforço de controle dada por $W_u = I_{2 \times 2}$ e a de ponderação das predições dada por

$$W_y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 40 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 40 \end{bmatrix}$$

A característica antecipativa agora fica mais evidente, pois nesta configuração o sistema antecipa-se em 20 ms, como ilustrado na área circular da Figura 4.12(b).

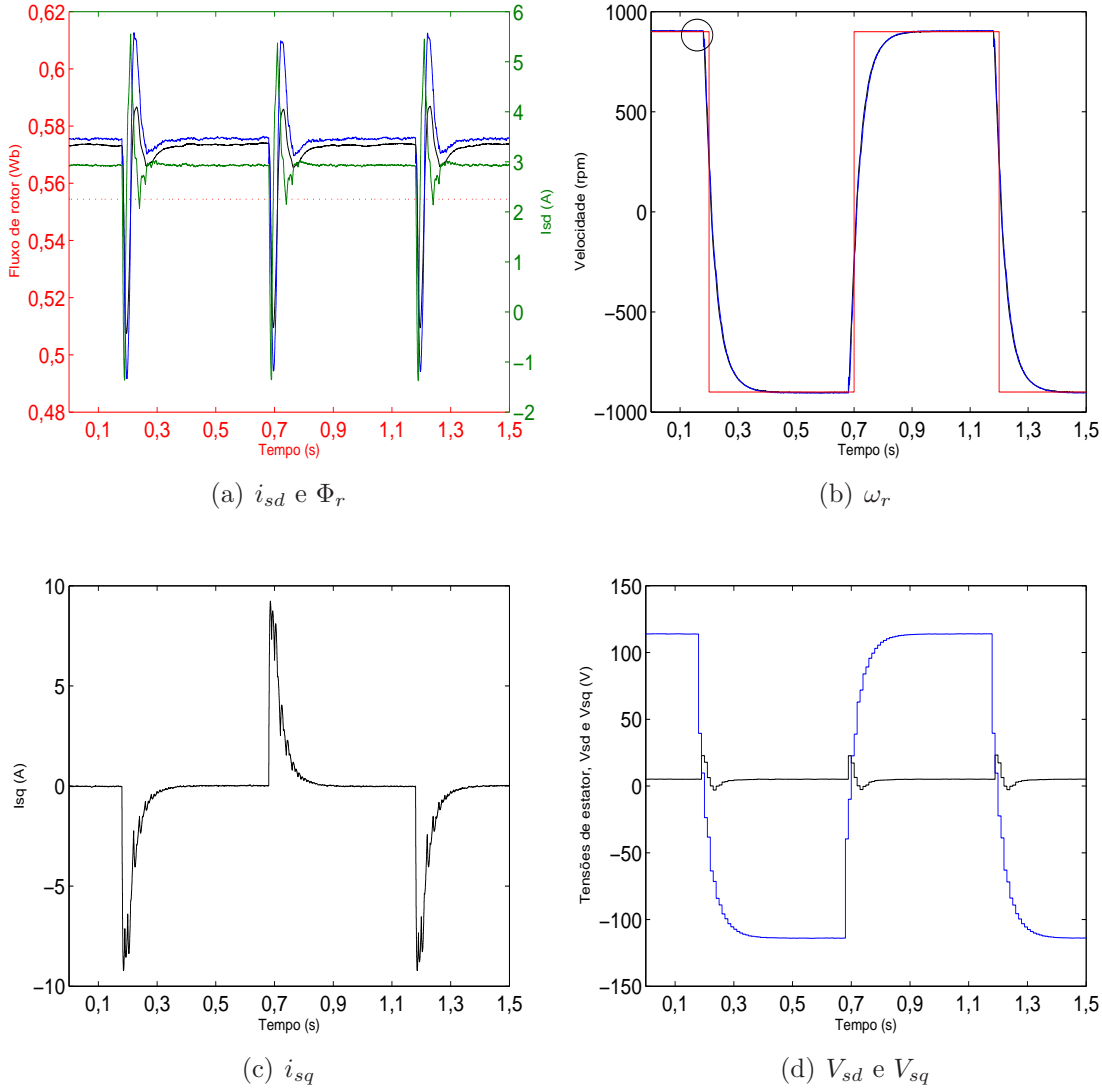


Figura 4.12: Teste com perfil de velocidade retangular de 900 rpm à -900 rpm com motor a vazio.

Na Figura 4.13 tem-se o teste de reversão de velocidade de 900 rpm à -900 rpm, com a variação do período do ciclo de controle, sendo agora $T_a = 6$ ms. Vê-se que há erro em regime permanente. Salienta-se que só foi possível realizar testes com bom desempenho do MBPC não variacional, para T_a aproximadamente maior que 8 ms e menor que 14 ms. Diferentemente do MBPC variacional que tinha bom desempenho de 2 ms até 14 ms.

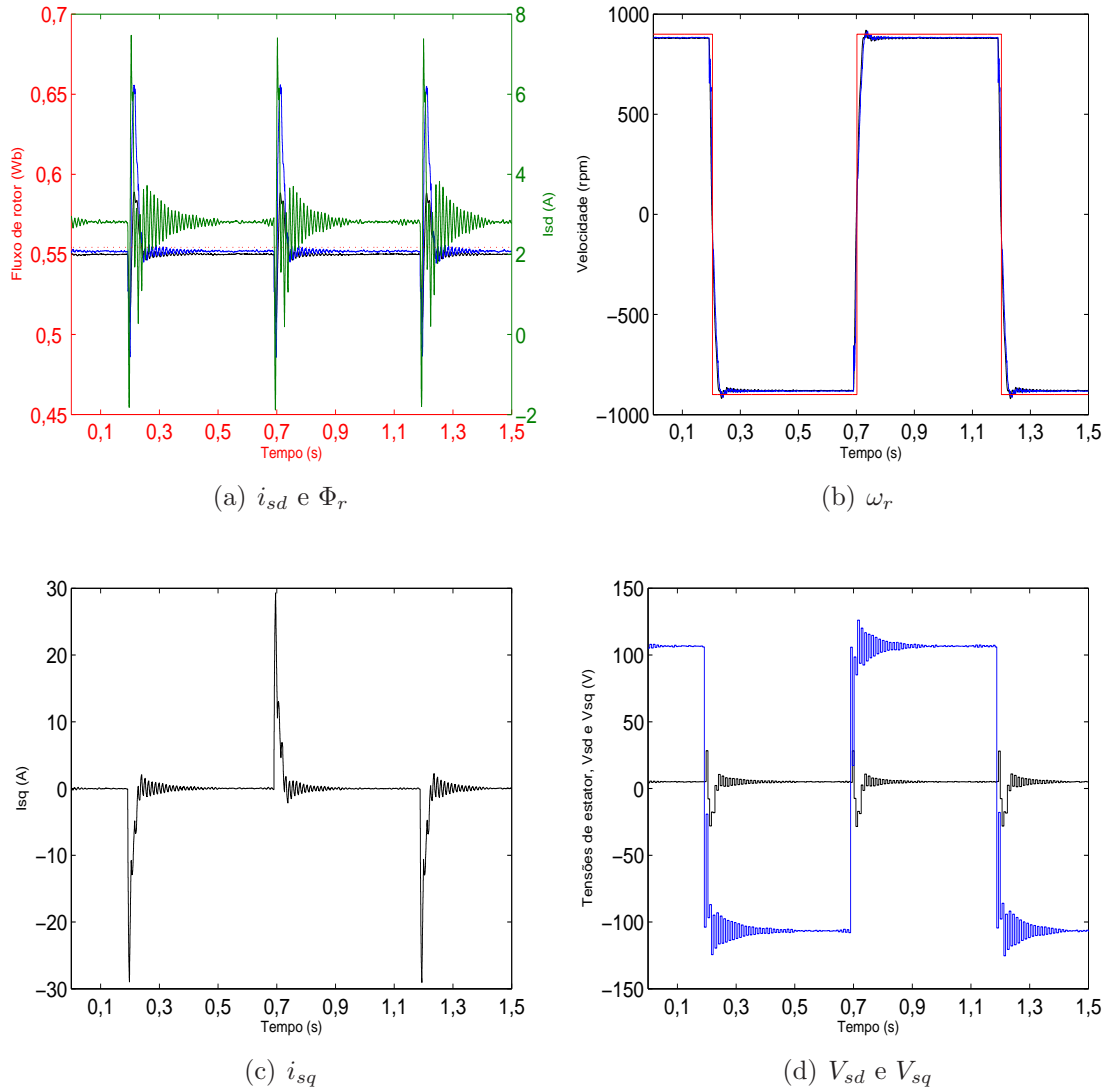


Figura 4.13: Teste com perfil de velocidade retangular de 900 rpm à -900 rpm com motor a vazio.

4.4 Conclusões do capítulo

Os resultados exibidos mostraram o bom desempenho do MBPC linearizado, tanto nos testes com o motor a vazio quanto sob carga. Quando as referências de velocidade mudam abruptamente verificou-se que os valores de corrente gerados são elevados e, isto se deve ao fato dos ciclos de controle T_a do MBPC serem grandes, de maneira que, as variações exigidas nas tensões aplicadas são muito elevadas. A solução adotada foi o emprego de um filtro digital passa-baixa, que suaviza a referência de velocidade.

O controlador não variacional também foi implementado em simulação e feita a escolha adequada dos parâmetros do MBPC o sistema apresentou bom desempenho. Ressalte-se

que as respostas em regime só foram satisfatórias para elevados tempos de ciclo de controle, $8 \text{ ms} < T_a < 14 \text{ ms}$.

CAPÍTULO 5

Resultados experimentais

5.1 Introdução

O algoritmo preditivo linearizado foi testado experimentalmente. O sistema composto pela estimação, conversão de variáveis $\alpha\beta \Leftrightarrow d - q$, medição de velocidade, modulação *Space Vector* e o controle MBPC linearizado foram implementados em um processador digital de sinais.

Descrevem-se, inicialmente, neste capítulo, o programa realizado em linguagem C e os equipamentos empregados na bancada; como a literatura sobre implementações para acionamentos elétricos é vasta e facilmente acessada, preferiu-se referenciar trabalhos anteriores e manuais que expõem as técnicas empregadas do que descrevê-las aqui. Na seção 5.3 são apresentados e analisados os resultados experimentais obtidos. O capítulo termina com a apresentação das conclusões obtidas dos experimentos.

5.2 Descrição funcional do sistema implementado

Processadores digitais de sinais têm sido projetados para aplicações específicas; para controle de motores há, por exemplo, o DSP TMS320F2812 da Texas Instruments, empregado no presente trabalho. Para que os dados sejam fornecidos e adquiridos pelo processador, os fabricantes de dispositivos eletrônicos criaram placas que contêm, processador, periféricos e interfaces tais como: conversores analógico/digital e digital/analógico; comparadores; memória adicional; unidades de entrada e saída, etc. Como à época da aquisição do DSP TMS320F2812 ainda não havia tais placas de avaliação completas, adquiriu-se o chamado *starter kit* eZdspTM da Spectrum Digital (Spe, 2002), que não contém tantos dispositivos adicionais.

O DSP empregado tem dois gerenciadores de eventos independentes (Tex, 2003b) que controlam diversas funcionalidades, podendo, se desejar, controlar dois sistemas independentemente com um único DSP. Dentre estas funcionalidades destacam-se os *timers* de propósito geral, comparadores, unidade de contagem de pulsos (QEP - **Q**uadrature

Encoder Pulse), registradores (para controlar interrupções, início da conversão analógica/digital, período dos *timers*, ...), entre outras.

Além das características específicas para acionamentos elétricos, o DSP empregado tem elevado poder de processamento (150 MHz), representa os números em 32 bits - o que significa uma boa precisão -, e permite que todo o algoritmo seja desenvolvido em ponto flutuante. Saliente-se que o DSP empregado processa os números em ponto fixo e a biblioteca IQmath (Tex, 2002) desenvolvida pela Texas Instruments trata de converter os números em ponto flutuantes usados no programa para o formato adequado para o processador. A possibilidade de programar em ponto flutuante diminui significativamente o tempo gasto na confecção do programa. O código do algoritmo implementado está disponível no Apêndice D.

A Figura 5.1 mostra o diagrama dos elementos de *hardware* da implementação experimental e a Figura 5.2 exhibe uma foto da bancada utilizada.

As correntes e tensões medidas com sensores de efeito Hall são adequadas ao conversor analógico/digital (CAD) do DSP (Tex, 2003a) que funciona de 0 a 3 V. Além do condicionamento, há uma proteção com diodos que garantem que o valor nunca ultrapasse 3 V, evitando danos ao CAD. Como os sinais de tensão são pulsos com largura variável e necessita-se da componente fundamental das tensões de linha, empregou-se um filtro passa-baixa com frequência de corte igual a metade da frequência de chaveamento.

A medição da velocidade é feita por um *encoder* que gera dois pulsos em quadratura, sendo a velocidade proporcional à quantidade de pulsos medidos num intervalo de tempo, e o sentido da velocidade é dado pelo pulso que está à frente. Detalhes sobre o método de medição da velocidade ver (Bertonha, 2006; de Araujo Silva, 2000). Como as saídas dos dois pulsos em quadratura do *encoder* têm amplitude máxima de 5 V, foi necessário condicioná-los para a valor máximo de 3 V.

Depois que os tempos de chaveamento das três das pernas do inversor são calculados pelo algoritmo da modulação por vetores espaciais, eles são carregados nos registradores CMPR1, CMPR2 e CMPR3 da unidade PWM do DSP. Toda vez que o valor do *timer* coincide com o valor do tempo carregado no registrado de comparação, significa que uma das chaves de uma perna do inversor deve fechar e a outra abrir. Para evitar que durante o chaveamento duas pernas do inversor fiquem fechadas simultaneamente - o que configura um curto-circuito, o TMS320F2812 permite que seja fornecido o valor do tempo morto entre as chaves.

Como *starter kit* usado não possui conversor ditital analógico (CDA), para que as variáveis processadas no DSP pudessem ser analisadas, construiu-se um CDA de 12 bits de quatro canais .

O *kit* se comunica com o computador pela porta paralela; os programas feitos em linguagem C são posteriormente carregados na memória do DSP mediante o uso do *software* Code Composer Studio, que tem a função de gerenciar a comunicação bidirecional entre o PC e o DSP, o que permite o monitoramento em tempo real das variáveis processadas no DSP.

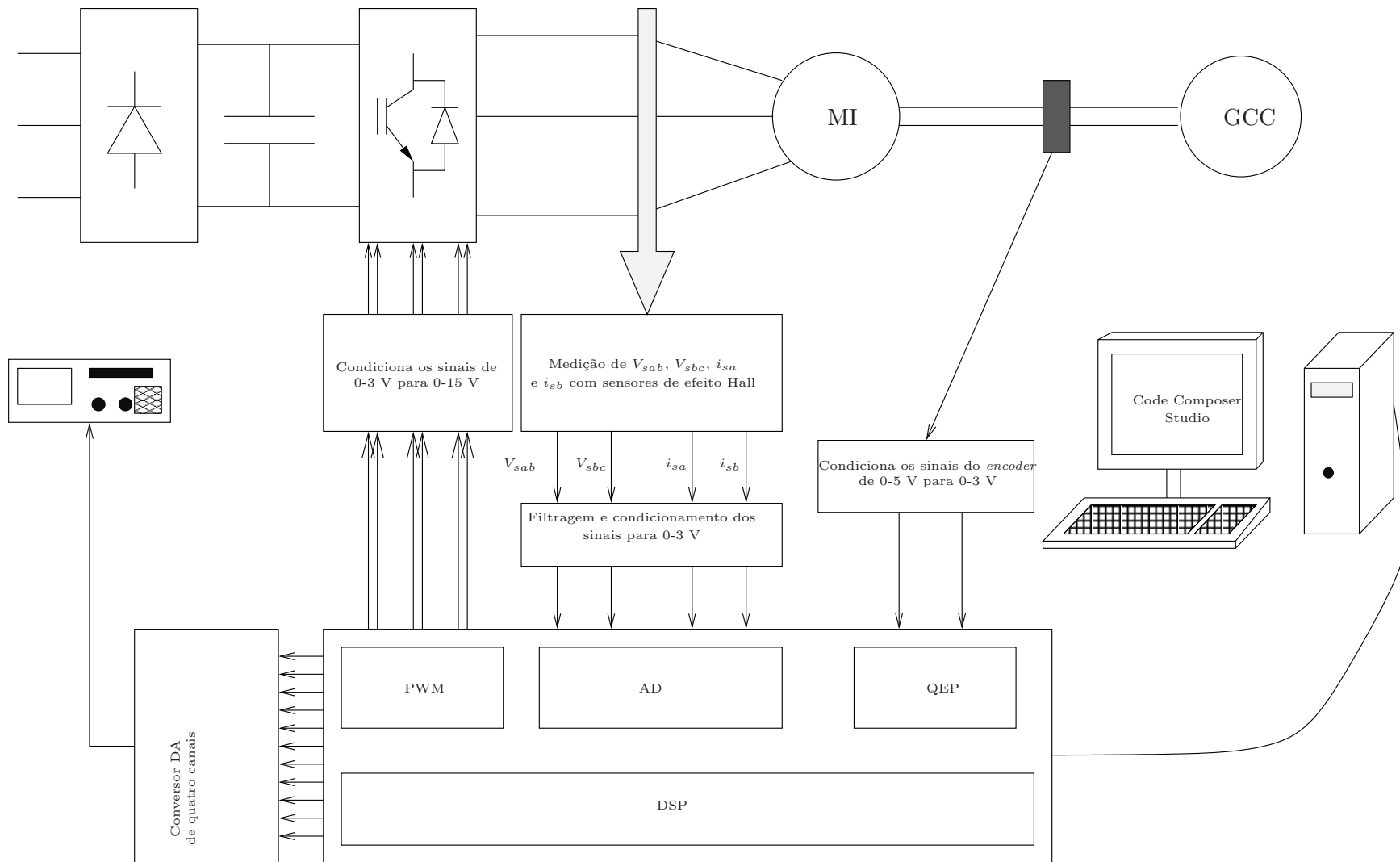


Figura 5.1: Ilustração dos dispositivos empregados na bancada para realização dos experimentos.

5.2.1 Algoritmo de controle

O algoritmo descrito na Figura 5.3 mostra a seqüência presente na função *main* do código. Nesta função, são inicializados os conversores AD, configuradas as entradas e saídas digitais, habilitadas e configuradas as interrupções via *timers*. A última instrução da função *main* é um *for* infinito, que é executado indefinidamente com a finalidade de manter o DSP sempre operando à espera das duas interrupções empregadas, nas quais os algoritmos são realizados, como descrito a seguir:

1. A interrupção de $300\ \mu\text{s}$, descritas na Figura 5.4, inicia-se com a medição dos valores de tensão de linha e correntes de estator, que são transformados para ponto flutuante, posteriormente, para coordenadas estacionárias $\alpha\beta$. Este valores de tensão e correntes são entradas para o FKE (Apêndice A). As saídas do FKE são as variáveis estimadas: $\Phi_{r\alpha}$, $\Phi_{r\beta}$ e ω_r . As variáveis são transformadas do eixo estacionário para $d-q$ ficando disponíveis ao algoritmo MBPC, e as variáveis $d-q$ da tensão, geradas pelo algoritmo MBPC, são transformadas para $\alpha\beta$ e empregadas na SVM, que é executada logo em seguida.
2. A interrupção do algoritmo MBPC é mostrada na Figura 5.5. Primeiramente, foi medida a velocidade do motor, para fins de comparação com a velocidade estimada e na seqüência são geradas as referências futuras de fluxo e velocidade; o algoritmo recebe como entrada as variáveis de tensão (V_{sd} e V_{sq}) e corrente (i_{sd} e i_{sq}) de estator, o fluxo de rotor Φ_{rd} e a velocidade ω_r . O algoritmo MBPC é executado com suas saídas dadas pelos acréscimos de tensão de eixos direto e em quadratura, que devem ser somados ao valor anterior da tensão. Estes valores são disponibilizados na memória para serem empregados pela interrupção anterior, que é responsável por transformar as tensões para o eixo bifásico estacionário e realizar a SVM.



Figura 5.2: Foto da bancada.

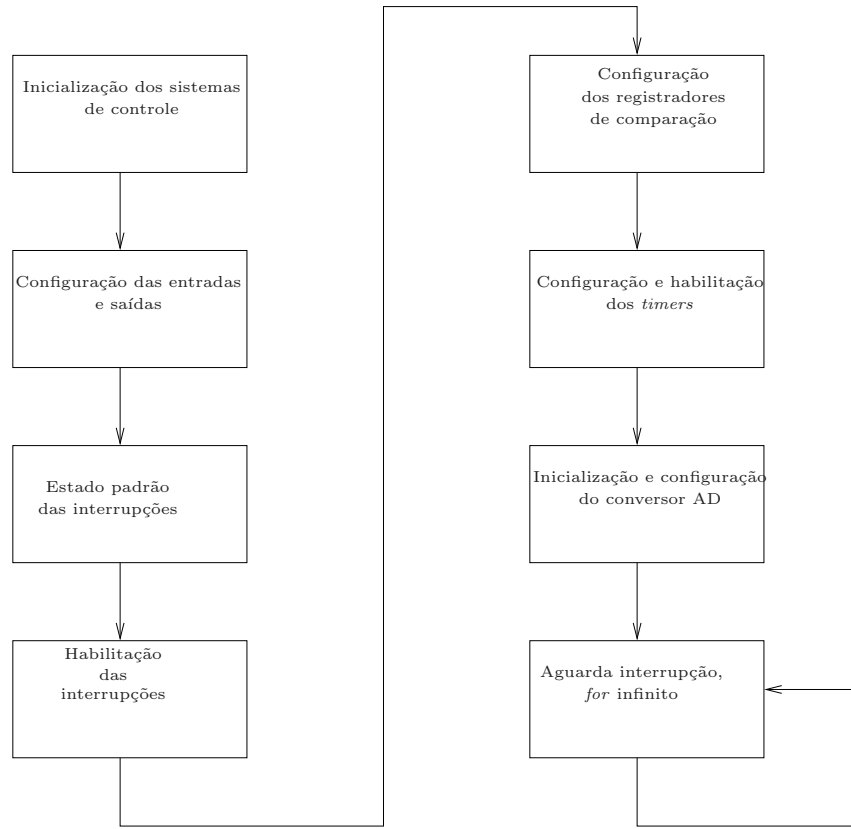


Figura 5.3: Configurações do DSP realizado na função *main*.

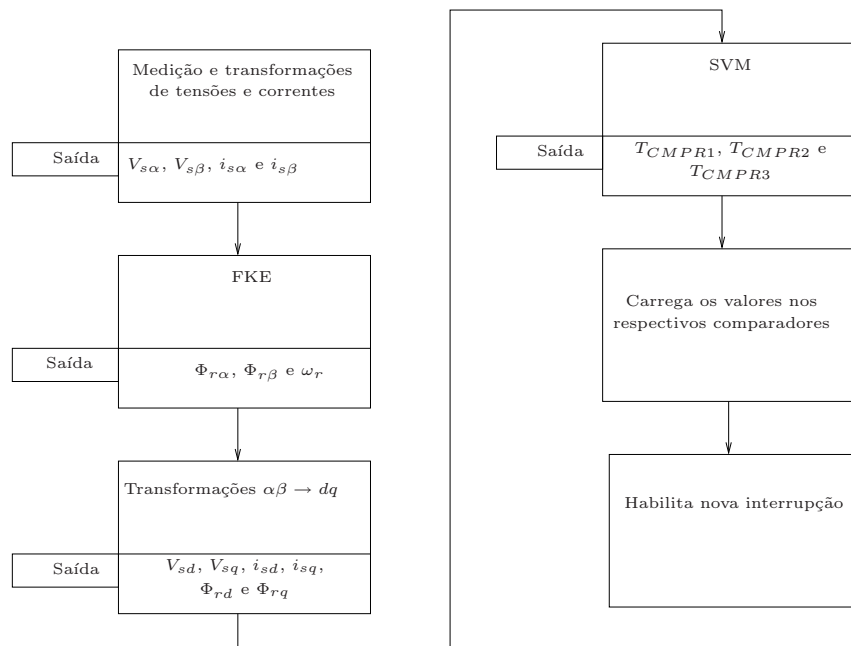


Figura 5.4: Descrição da interrupção das medições, do FKE e da SV-PWM.

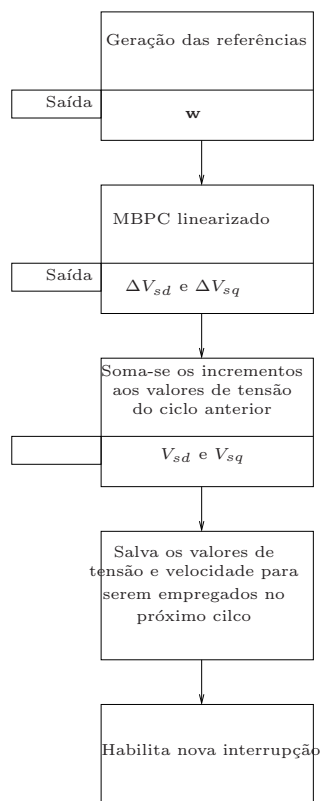


Figura 5.5: Interrupção do MBPC.

5.3 Resultados experimentais

Para avaliar a proposta, testes experimentais foram realizados com o motor a vazio e com carga acoplado ao eixo. Os parâmetros do MBPC são os mesmos da simulação, como descritos na pág. 32.

5.3.1 Testes a vazio

A Figura 5.6 mostra o teste de reversão de velocidade em degrau. O gráfico da Figura 5.6(a) tem as três formas de onda superpostas, enquanto o da Figura 5.6(b) ilustra as três formas de onda separadas - para que sejam identificados individualmente. Na Figura 5.6(c) vê-se o comportamento da velocidade estimada no instante da mudança da referência. Nota-se que as medidas estão ruidosas, provavelmente por causa do campo eletromagnético gerado quando o motor é alimentado pelo inversor de frequência, o que prejudicou significativamente a medição da velocidade, pois os pulsos do *encoder* eram deformados.

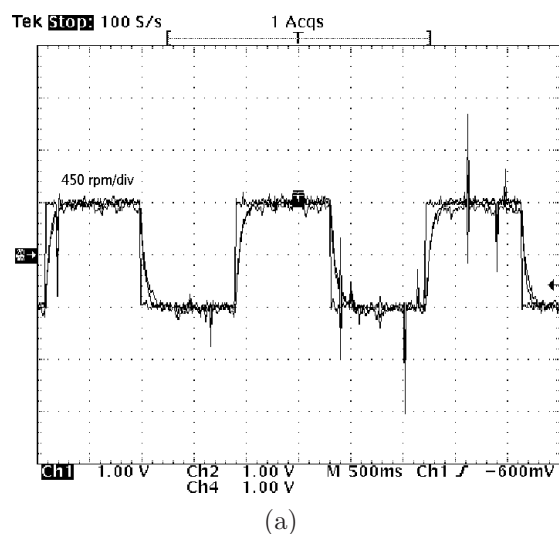
Os resultados do teste da variação quadrangular da velocidade de 0 a 300 rpm é mostrado na Figura 5.7. Na Figura 5.7(a) têm-se a velocidade de referência, estimada e medida. Na Figura 5.7(b), os testes da Figura 5.7(a) são repetidos, mas com a função de eliminação de ruídos de alta frequência através do cálculo da média de todas as amostras em um intervalo de aquisição ativada no osciloscópio. Nesta última figura pode-se ver que na média o acionamento segue as referências impostas.

Os resultados mostrados na Figura 5.8 são relativos ao teste da reversão de velocidade de perfil trapezoidal, em que a velocidade de referência atinge os valores máximo e mínimo de 600 rpm e -600 rpm, respectivamente, em 600 ms. Outro teste com perfil de velocidade trapazoidal, com valores variando de 150 rpm a -150 rpm em 120 ms, é mostrado na Figura 5.9.

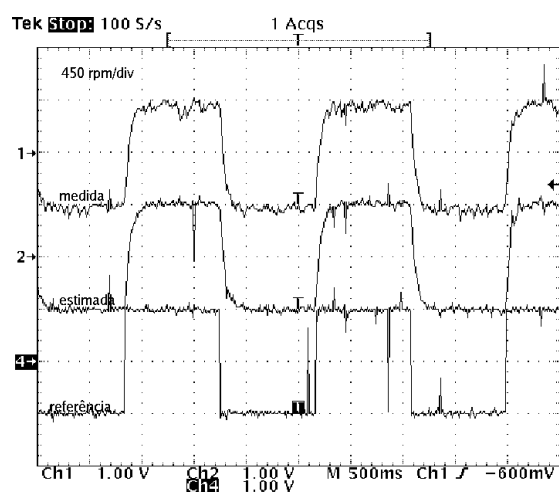
O teste de variação retangular de carga de 0 a 450 rpm é exibido na Figura 5.10; são mostrados os valores de referência, estimado e medido da velocidade.

Na Figura 5.11(a) é mostrado o resultado do teste da variação triangular da velocidade de 450 rpm a -450 rpm, e na Figura 5.11(b) são mostrados os valores de referência e estimados da velocidade e da corrente i_{sq} , que quando o sistema está orientado e Φ_r está constante, relaciona-se diretamente com o torque do MI. Nota-se que i_{sq} positivo no ciclo de aumento da velocidade e i_{sq} negativo para o ciclo negativo.

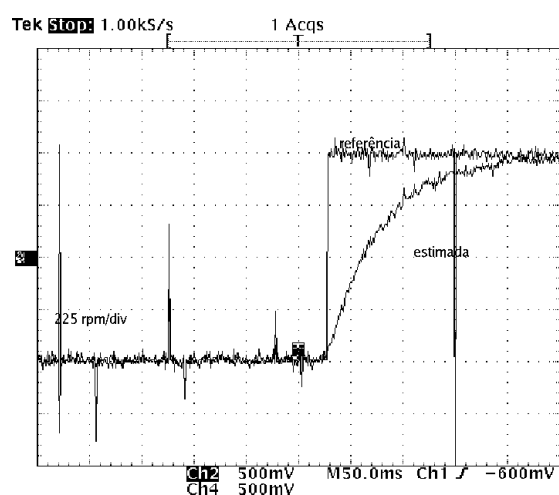
Para avaliar o comportamento do acionamento, quando há variação de fluxo fez-se o teste no qual há diminuição do fluxo de rotor do valor nominal até 50% deste em 300 ms, com a velocidade de referência do MI de 300 rpm. Os resultados, mostrados na Figura 5.12(a), mostram que o controlador atuou de maneira eficiente, tanto na modificação da magnitude do fluxo, quanto na regulação de velocidade. Na Figura 5.12(b) tem-se o comportamento dos fluxos de rotor de referência e estimado e a corrente i_{sd} ; quando há orientação de fluxo de rotor, o valor de i_{sd} relaciona-se diretamente com Φ_r , como pode ser notado.



(a)

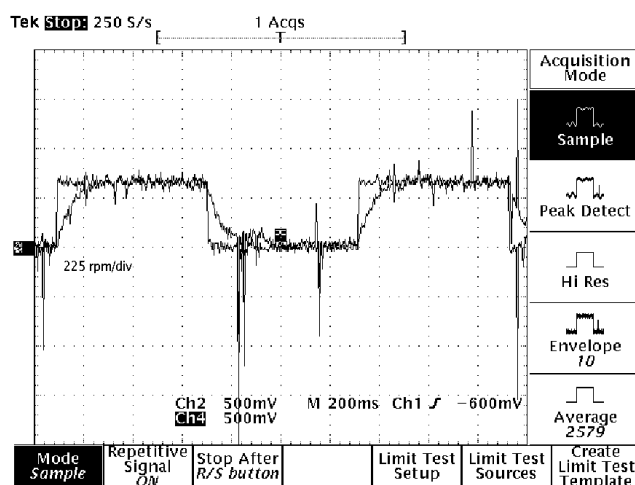


(b)

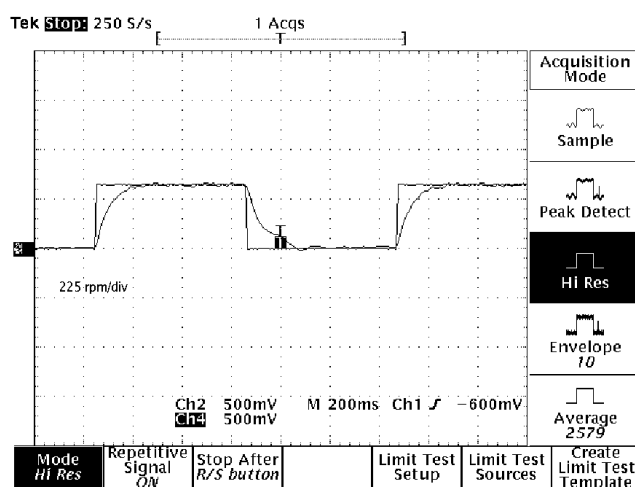


(c)

Figura 5.6: Reversão de 450 rpm a -450 rpm.



(a)



(b)

Figura 5.7: Variação em degrau de 0 a 300 rpm .

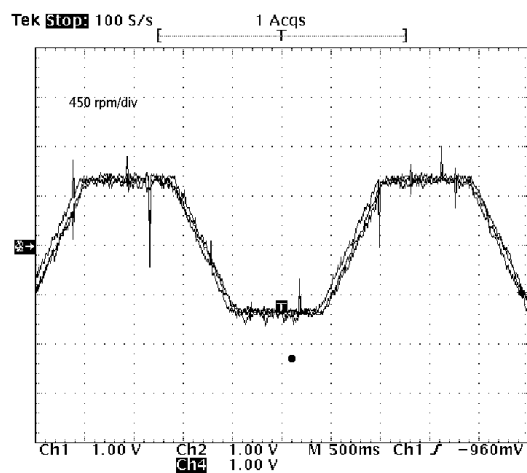


Figura 5.8: Reversão de velocidade; 600 rpm a -600 rpm em 600 ms.

A reversão de velocidade de 150 rpm a -150 rpm é mostrado no gráfico da Figura ??.

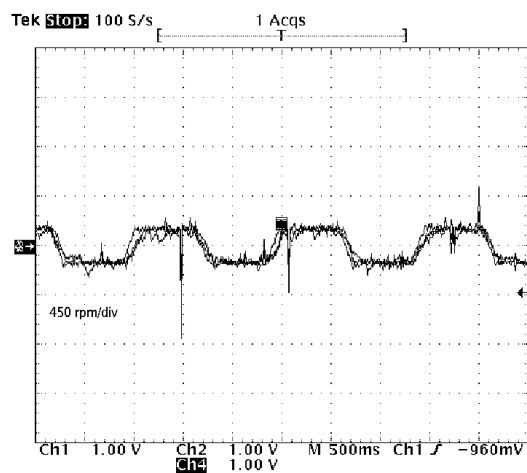


Figura 5.9: Reversão de velocidade; 150 rpm a -150 rpm em 120 ms.

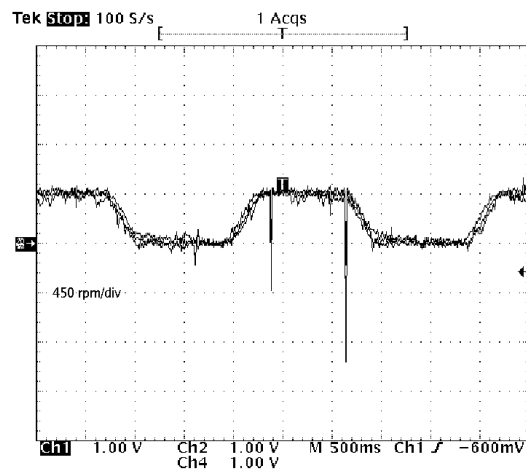


Figura 5.10: Variação de 0 rpm a 450 rpm, com tempos de subida e descida de 300 ms.

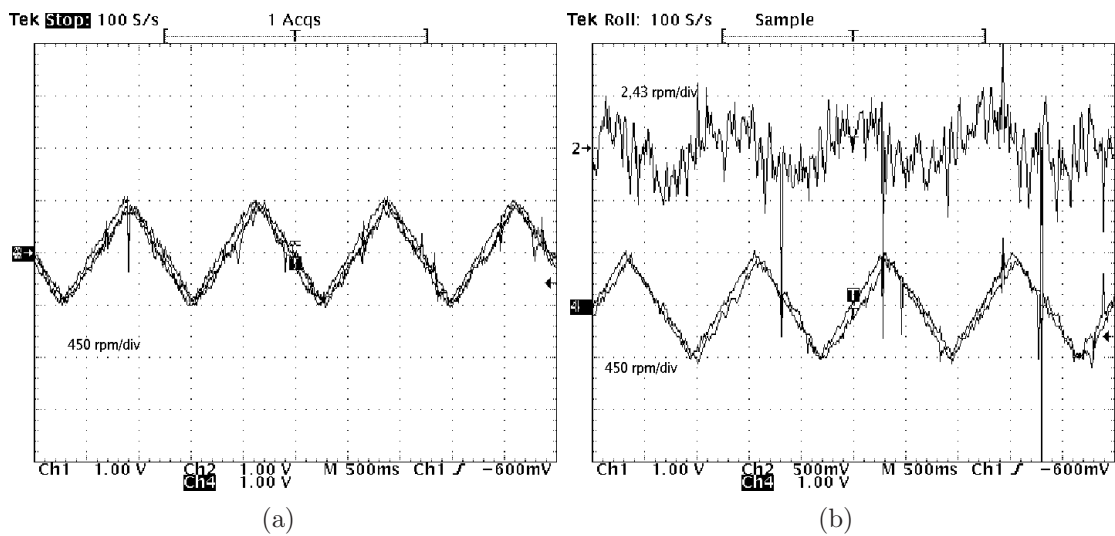
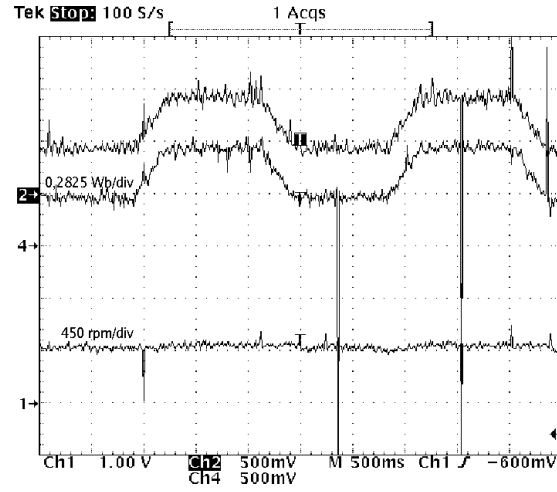
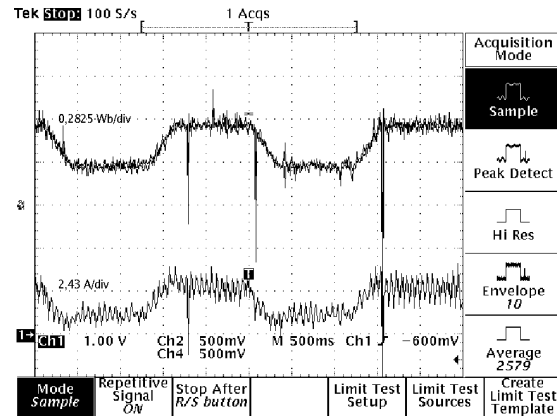


Figura 5.11: Variação triangular de 450 rpm a -450 rpm, com tempos de subida e de descida de 600 ms.



(a)



(b)

Figura 5.12: Variação de fluxo de rotor do valor nominal a 50% do valor nominal em 300 ms.

5.3.2 Teste com carga

As simulações realizadas com carga, cujos resultados são exibidos no capítulo anterior, supunham que a inércia da carga tinha magnitude duas vezes maior que a do MI. O gerador de corrente contínua empregado para os testes experimentais tem inércia de aproximadamente $0,185 \text{ Kg.m}^2$, ou seja, seu valor é 27,61 maior que a do MI. O MBPC proposto não se mostrou robusto a esta variação quando há torque de carga em degrau aplicado ao eixo, de forma que, foi necessário modificar o valor da inércia utilizado pelo algoritmo preditivo, embora este valor não deva ser necessariamente igual ao valor real; empregou-se no MBPC o valor $0,0804 \text{ Kg.m}^2$, que é doze vezes maior que a do MI. Como houve grande alteração no modelo do MI, tornou-se necessário encontrar novos valores

para as matrizes de ponderação do esforço de controle e da predição; após vários testes escolheu-se as novas matrizes como sendo dadas por

$$W_u = I_{2 \times 2} \text{ e } W_y = \begin{bmatrix} 0,5 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 0,5 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}$$

As entradas são ponderadas igualmente pela matriz identidade, enquanto as predições da velocidade tem ponderação 20 vezes maior que a do fluxo. Isto se deve ao aumento significativo da inércia que só terá desempenho satisfatório se gerar valores de tensão em quadratura maiores. A Figura 5.13 mostra o desempenho do acionamento nesta situação. O torque da carga é de 10,36 N.m, valor abaixo do nominal, que é de 12,3 N.m. Vê-se que a corrente i_{sd} , mostrada pelo canal 2, mantém seu valor constante, o que evidencia que o fluxo do rotor também tem sua magnitude constante, já que a orientação de fluxo é empregada. Nota-se também que a corrente i_{sd} , que é proporcional ao torque eletromecânico desenvolvido pelo MI, se o fluxo de rotor é constante, tem sua magnitude elevada durante a aplicação de carga. A velocidade (curva mostrada pelo canal 1, cujo valor de zero Volt corresponde à 600 rpm) sofre alteração durante a colocação e retirada do torque de carga, mas a regulação de ω_r foi satisfatória durante o regime permanente.

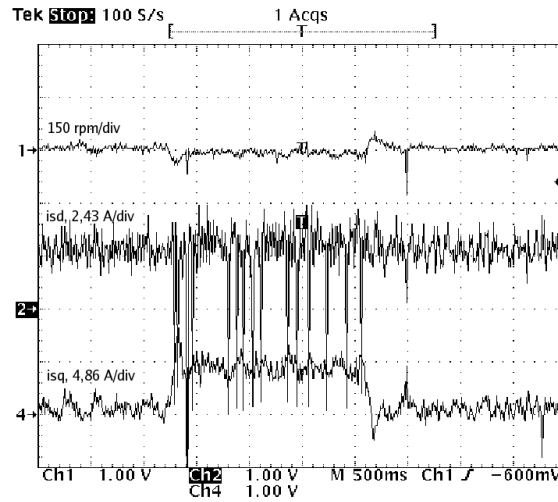


Figura 5.13: Teste com carga de 10,36 N.m e velocidade igual à 600 rpm. Canais 1, 2 e 4 mostram a velocidade estimada, i_{sd} e i_{sq} . No canal 1, o valor zero Volt indica a velocidade de 600 rpm.

5.4 Conclusões do capítulo

Os resultados experimentais exibidos mostraram o potencial de aplicação da estratégia MBPC para acionar o MI. Nota-se que o desempenho foi significativamente melhor em altas velocidades, na qual o modelo do MI o representa bem; já em baixa velocidade, situação esta na qual há forte dependência do modelo em relação às resistências de estator e rotor, o desempenho foi deficiente.

Não foi possível avaliar a proposta sob condições de carga similares a das simulações devido ao gerador CC empregado ter uma inércia significativamente maior que a do MI ($J_{gcc} = 27,61J_{MI}$). Os testes experimentais mostraram que o algoritmo não tem bom desempenho quando a inércia real é significativamente diferente do empregado no MBPC. Para melhorar a resposta do sistema de controle deve-se empregar valores mais próximos do valor real; no presente caso o algoritmo de controle empregou um valor de inércia da carga doze vezes maior do que o do MI, enquanto o valor real é 27,61 vezes maior.

CAPÍTULO 6

Conclusões e sugestões de trabalhos futuros

6.1 Conclusão

Neste trabalho fez-se a aplicação de um algoritmo MBPC para o controle digital do fluxo de rotor e da velocidade mecânica de um motor de indução. O modelo do motor de indução empregado no MBPC é o orientado segundo o fluxo do rotor; este fluxo e a velocidade de rotação do eixo foram estimados mediante o filtro de Kalman estendido.

Tradicionalmente, os acionamentos propostos empregam fontes de tensão, cujos sinais de controle são gerados a partir de controladores de corrente; isto exige uma malha externa para controlar o fluxo e a velocidade/torque. Conforme discutido anteriormente, controladores em cascata são difíceis de serem sintonizados. Uma das contribuições dadas nesta tese, é fato de os sinais de tensão serem gerados diretamente dos erros entre as previsões das variáveis controladas - fluxo de rotor e velocidade, e as suas referências futuras.

Dentre as características positivas do algoritmo MBPC aplicado ao acionamento do motor de indução destaca-se o fato de não serem necessários ciclos de controle pequenos, diferentemente do que ocorre na maioria das estratégias de controle. Recentemente, são encontrados na literatura controladores PI, que atuam com ciclo de controle em torno de $100\ \mu\text{s}$, enquanto que neste trabalho, operou-se, nos testes experimentais, com um ciclo bem maior, isto é, cerca de 6 ms. Isto significa que, apesar do elevado custo computacional, as implementações em tempo real são possíveis, devido ao valor elevado do ciclo de controle em relação às estratégias convencionais. Entretanto, como já discutido anteriormente, ciclos de controle elevados só são possíveis se o modelo representar bem a planta.

Os resultados experimentais do motor a vazio mostraram que o MBPC teve bom desempenho em velocidades acima de 120 rpm, sendo possível verificar a sua principal característica, que é a resposta antecipativa do controlador.

Sob carga, pelo fato de ter sido acoplada uma máquina de corrente contínua ao eixo do motor de indução, o valor empregado da inércia no algoritmo MBPC teve que ser alterado

para obter-se um bom desempenho. maior que o valor empregado no algoritmo de controle, o que permite concluir que a proposta é robusta em relação a mudança de inércia.

Deve-se destacar o fraco desempenho do sistema de controle em velocidades abaixo de 120 rpm. Um dos motivos para a deterioração do desempenho em baixas velocidades foi ruído presente na montagem experimental, bem como a forte dependência do algoritmo MBPC e do FKE dos parâmetros do modelo do MI. Uma alternativa para melhorar o desempenho do acionamento do MI seria empregar estratégias de controle preditivo adaptativo e de multimodelo.

Apesar do DSP empregado ter tido desempenho satisfatório na implementação, saliente-se que a primeira interrupção - que contém o FKE e a SVPWM - ocupou aproximadamente $125 \mu s$ dos $300 \mu s$ destinados a ela. Caso o MBPC empregado tivesse um horizonte de controle n_y maior, o período do ciclo de controle T_a diminuiria e o custo computacional aumentaria. Desta forma, sugere-se que a estimação de estado seja realizada por uma alternativa que demande menos cálculos do que o FKE.

6.2 Sugestões para trabalhos futuros

6.2.1 MBPC

Função custo :

Avaliar o desempenho do MBPC com outras funções custos além da quadrática; empregar, por exemplo, a norma ∞ - que oferece maior robustez a variação paramétrica - ou a norma unitária.

Restrições :

Incluir explicitamente na função custo as restrições de tensão de estator ($V_s < V_{s/nominal}$) e de corrente de estator (i_s).

MBPC não linear :

Resolver a minimização da função custo sem linearização do modelo.

Melhora na predição da perturbação :

Quando a característica do torque de carga é modelada - em algumas cargas T_c varia de forma conhecida com ω_r - pode-se realizar uma predição mais eficaz do que a simples consideração que T_c é constante. Isto deve melhorar o comportamento dinâmico do acionamento quando houver carga acoplada ao eixo.

6.2.2 Estimador de estado

Empregar outro estimador :

O FKE é uma ferramenta eficiente para resolver a estimação de sistemas que possuem características estocásticas nos estados ou medições, entretanto, o seu custo computacional é elevado. Como o MI é um sistema que pode ser considerado determinístico, pode-se empregar observadores, como por exemplo, o de Luenberger ou em modos deslizantes;

Estimar parâmetros :

Pode-se estimar a constante de tempo do rotor e a inércia da carga, pois a atualização destes parâmetros no estimador e no modelo empregado no MBPC melhorariam o desempenho do acionamento, especialmente em baixas velocidades.

Estimação do torque de carga :

Implementar o MBPC linearizado com o torque de carga sendo calculado por um estimador.

6.2.3 Acionamento

Seleção da referência de fluxo :

O desempenho do algoritmo MBPC linearizado é satisfatório quando deseja-se fluxo variável; isto pode ser empregado na melhoria da eficiência, bastando escolher o valor adequado de fluxo que permite otimizar a eficiência do MI;

Mudar o fluxo de referência para a orientação :

Avaliar o desempenho do MBPC linearizado, caso seja empregado o fluxo do estator ou do entreferro no modelo.

6.2.4 Implementação

Bancada :

Construção de uma bancada menos susceptível ao ruído eletromagnético.

APÊNDICE A

Filtro de Kalman Estendido

A.1 Introdução

Apesar do controle proposto empregar as variáveis elétricas orientadas segundo o fluxo de rotor, ou seja, com os eixos de referência girando na velocidade síncrona, a estimação é realizada empregando o modelo do motor em coordenadas estacionárias.

A escolha do modelo também deve levar em consideração as variáveis que deseja estimar. Nesta tese são abordadas duas alternativas:

1. Estimação das componentes estacionárias do fluxo de rotor e da velocidade, que é representado por um modelo de quinta ordem;
2. Estimação das componentes estacionárias do fluxo de rotor, da velocidade e do torque de carga, que é representado por um modelo de sexta ordem.

Ambas abordagens são descritas a seguir.

A.2 Modelos do MI para estimação

A estimação das variáveis de estado foram feitas considerando os modelos de quinta e sexta ordem do MI, como descritos na seqüência:

Modelo de sexta ordem Este tem como variáveis de estado as correntes de estator, os fluxos de rotor, a velocidade e o torque de carga. Salienta-se que as variáveis eletromagnéticas estão na referência estacionária. As correntes de estator $i_{s\alpha}$ e $i_{s\beta}$ são obtidas via transformações das correntes medidas das fases a e b (como descrito na equação (3.14)) e os demais estados são estimados.

As variáveis de entrada do FKE são, além das correntes de estator $\alpha - \beta$, as tensões $V_{s\alpha}$ e $V_{s\beta}$, que são obtidas através das equações (3.13).

Como a característica da carga é desconhecida, considerou-se sua dinâmica lenta de tal forma que pode-se escrever $\dot{T}_c \approx 0$, o que implica que a taxa de amostragem do FKE deve ser pequena o suficiente para não registrar uma variação elevada da magnitude do torque de carga em um período de amostragem.

Verifica-se nas simulações do capítulo 4 que com a aproximação $\dot{T}_c \approx 0$ foi possível estimar o torque de carga, com erros nos transitórios devido a suposição que a carga não possui inércia, o que não é verdadeiro.

O modelo completo descrito em espaço de estado tem as seguintes matrizes

$$x = [i_{s\alpha} \quad i_{s\beta} \quad \Phi_{r\alpha} \quad \Phi_{r\beta} \quad \omega_r \quad T_c]^T$$

$$u = [V_{s\alpha} \quad V_{s\beta}]^T$$

$$A = \begin{bmatrix} \frac{-1}{\tau'} & 0 & \frac{L_m}{L'_s L_r \tau_r} & \frac{\omega_r L_m}{L'_s L_r} & 0 & 0 \\ 0 & \frac{-1}{\tau'} & -\frac{\omega_r L_m}{L'_s L_r} & \frac{L_m}{L'_s L_r \tau_r} & 0 & 0 \\ \frac{L_m}{\tau_r} & 0 & \frac{-1}{\tau_r} & -\omega_r & 0 & 0 \\ 0 & \frac{L_m}{\tau_r} & \omega_r & \frac{-1}{\tau_r} & 0 & 0 \\ 0 & \kappa \Phi_{r\alpha} & \kappa \Phi_{r\beta} & 0 & 0 & -\frac{1}{J} T_c \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$B = \begin{bmatrix} 1/L'_s & 0 \\ 0 & 1/L'_s \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix};$$

Modelo de quinta ordem Este possui as mesmas entradas do de sexta, entretanto, como assume-se que $\dot{\omega}_r \approx 0$, torna-se desnecessária a representação do torque de carga como estado. Esta aproximação é válida desde que o tempo de amostragem escolhido não registre variação significativa na velocidade em um período de amostragem. Sendo assim, as matrizes do modelo de quinta ordem são:

$$x = [i_{s\alpha} \quad i_{s\beta} \quad \Phi_{r\alpha} \quad \Phi_{r\beta} \quad \omega_r]^T$$

$$u = [V_{s\alpha} \quad V_{s\beta}]^T$$

$$A = \begin{bmatrix} \frac{-1}{\tau'} & 0 & \frac{L_m}{L'_s L_r \tau_r} & \frac{\omega_r L_m}{L'_s L_r} & 0 \\ 0 & \frac{-1}{\tau'} & -\frac{\omega_r L_m}{L'_s L_r} & \frac{L_m}{L'_s L_r \tau_r} & 0 \\ \frac{L_m}{\tau_r} & 0 & \frac{-1}{\tau_r} & -\omega_r & 0 \\ 0 & \frac{L_m}{\tau_r} & \omega_r & \frac{-1}{\tau_r} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$B = \begin{bmatrix} 1/L'_s & 0 \\ 0 & 1/L'_s \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix};$$

A.3 Algoritmo do filtro de Kalman estendido

As seguintes etapas descrevem o FKE:

Etapa 1 *Inicialização do vetor de estados e matrizes de covariância:*

Inicializa-se o vetor de estados $x = x(t_0)$. As matrizes Q (diagonal $\in \mathbb{R}^{5 \times 5}$), R (diagonal $\in \mathbb{R}^{2 \times 2}$) e a matriz de covariância P (diagonal $\in \mathbb{R}^{5 \times 5}$).

Etapa 2 *Predição do vetor de estados:*

No instante de amostragem k faz-se a predição dos estados para $k+1$, com o emprego do modelo e da entrada atual

$$\hat{x}_{k+1/k} = A_d \hat{x}_k + B_d u_k \quad (\text{A.1})$$

Etapa 3 *Estimação da matriz de covariância da predição P^* :*

Esta matriz é dada por

$$P_{k+1}^* = f_{k+1} \hat{P}_k f_{k+1}^T + Q \quad (\text{A.2})$$

sendo \hat{f}_{k+1} a matriz gradiente dada por

$$\hat{f}_{k+1} = \frac{\partial}{\partial x} (A_d x + B_d u) |_{x=\hat{x}_{k+1}} \quad (\text{A.3})$$

Etapa 4 *Cálculo do ganho do filtro de Kalman estendido:*

A matriz ganho do filtro de Kalman estendido é calculada por

$$K_{k+1} = P_{k+1} h_{k+1}^T [h_{k+1} P_{k+1} h_{k+1}^T + R]^{-1} \quad (\text{A.4})$$

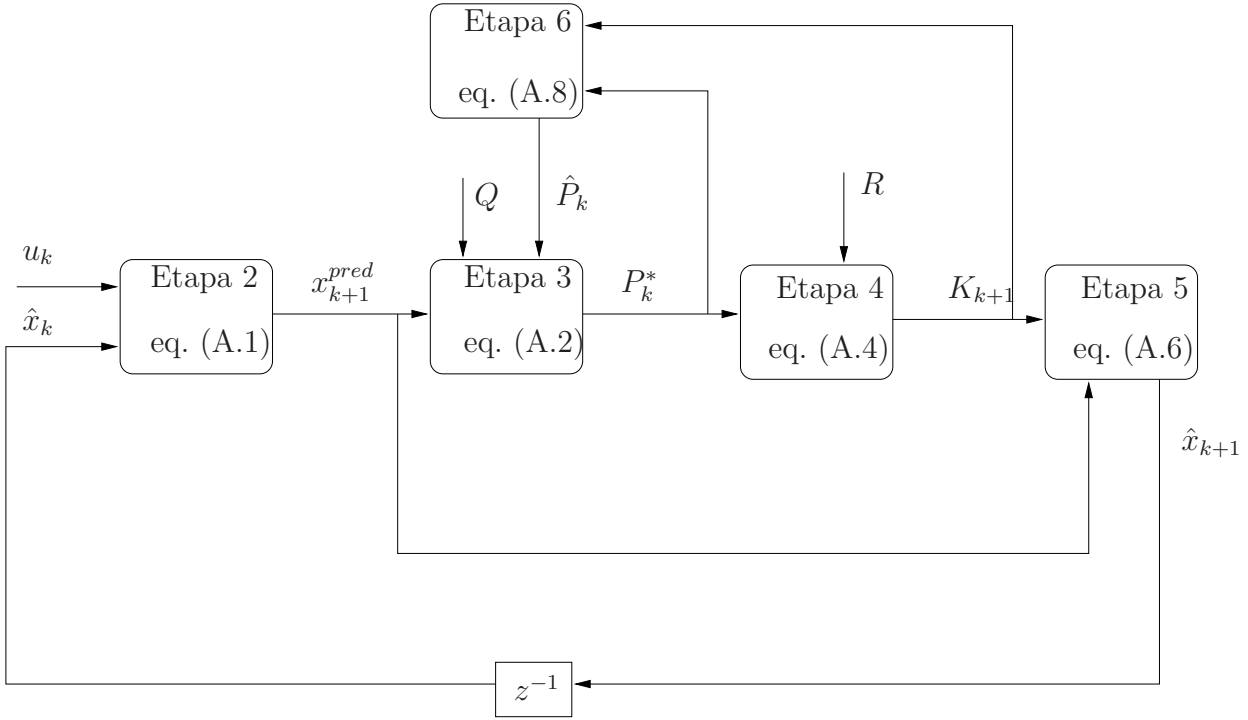


Figura A.1: Diagrama de blocos do filtro de Kalman estendido.

sendo

$$h_{k+1} = \frac{\partial}{\partial x} |C_d x| \big|_{\hat{x}=x_{k+1}} \quad (\text{A.5})$$

Etapa 5 *Estimação do vetor de estado:*

Os valores estimados são encontrados por

$$\hat{x}_{k+1} = \hat{x}_{k+1/k} + K_{k+1}[y_{k+1} - \hat{y}_{k+1}] \quad (\text{A.6})$$

sendo

$$\hat{y} = C_d x_{k+1} = [i_{sd(k+1)} i_{sq(k+1)}]^T \quad (\text{A.7})$$

Etapa 6 *Cálculo da matriz de covariância do erro de estimação:*

Esta é obtida de

$$\hat{P}_{k+1} = P_{k+1}^* - K_{k+1} h_{k+1} P_{k+1}^* \quad (\text{A.8})$$

A Figura A.1 mostra um diagrama de blocos do algoritmo do FKE.

A.4 Algumas considerações adicionais

As equações do modelo do motor utilizadas no FKE estão na referência estacionária $\alpha - \beta$; para adequá-las ao uso no MBPC - que tem sua modelagem descrita na referência $d - q$ - é necessário obter o seno e o cosseno de θ_{Φ_r} , como descrito na equação (3.15).

O motor sob teste é o mesmo de Valdenegro (2001) e, dessa maneira, as matrizes-parâmetros do FKE, empregadas pelo referido autor são aproveitadas neste trabalho. Entretanto, o trabalho citado implementa apenas a abordagem com $\dot{\omega}_r \approx 0$ com o modelo de quinta ordem, de maneira que a matriz Q é acrescida de uma sexta linha e uma sexta coluna correspondente ao torque da carga (caso do modelo de sexta ordem). As matrizes são dadas por:

$$Q = \begin{bmatrix} 0,0152 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,0152 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,00457 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,00457 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,00763 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix} \quad (\text{A.9})$$

$$R = 0,30518I_{2 \times 2} \quad \text{e} \quad P_0 = 0,004882I_{5 \times 5} \quad (\text{A.10})$$

Sendo

- Q a matriz ruído do sistema que modela o ruído nos estados;
- R a matriz ruído de medição que modela o ruído nos estados medidos no caso as correntes de estator, e;
- P_0 a matriz inicial de covariância predita. Esta indica o grau de conhecimento que se tem das condições iniciais do sistema; quanto mais se conhece os valores iniciais dos estados menor devem ser os valores de P_0 .

Os métodos analíticos para a obtenção da matriz de covariância Q são complexos; na prática os valores de Q são escolhidos por experimentação. Para simplificar o método de escolha de Q fez-se duas suposições (Pereira, n.d.):

1. A matriz Q é constante;
2. Não há correlação entre os erros das variáveis do processo.

A segunda suposição reduz Q a uma matriz diagonal.

Outro fator que auxilia na seleção dos valores da matriz de covariância Q é o conhecimento de como os estados estão se comportando em relação aos seus valores reais, de maneira que, aos estados que são modelados mais precisamente atribui-se menores valores de covariância. Se o modelo empregado representa de forma idêntica o comportamento da planta, então, tem-se Q dada pela matriz nula. No presente caso nota-se na matriz Q da equação (A.9) que o valor da ponderação do torque de carga é significativamente maior que os demais, já que a modelagem do torque de carga é a mais imprecisa dentre os estados, pois considerou-se que sua variação era insignificante de uma amostragem a outra. Apesar da mesma aproximação ser realizada para a velocidade (no emprego do modelo de quinta ordem), deve-se salientar que a informação da velocidade está presente em todas as equações elétricas (duas de estator e duas de rotor), de maneira que sua relação

com os outros estados é significativamente mais forte que a do torque de carga. Isto fica mais evidente no teste mostrado na página 41, no qual o torque de carga é aplicado ao eixo abruptamente, pois apesar de influenciar no valor dos demais estados, a sua própria magnitude independe de qualquer outra variável do MI.

A matriz R pondera o ruído nas medições; se sensores distintos são empregados para medir as variáveis, deve-se escolher a matriz R como sendo diagonal, o que representa que o ruído de uma variável medida não interfere na medição de outra variável. Os valores de R podem ser determinados, pelo menos para os primeiros testes, de acordo com os dados dos fabricantes dos sensores. Pode-se considerar também que R é constante. Como no presente trabalho é empregado o mesmo tipo de sensor para medir ambas as correntes, a matriz R é escolhida com seus valores idênticos e diagonal. Salienta-se que no caso particular das medições serem perfeitas tem-se R dado pela matriz nula.

A aproximação utilizada para discretizar o modelo do motor para uso no FKE foi de segunda ordem (como descrito na equação (2.13)). Salienta-se que a aproximação de segunda ordem não aumenta significativamente o custo computacional do FKE em relação a de primeira ordem, pois o maior peso do FKE se dá pelas várias multiplicações matriciais.

A.5 Resultados experimentais do FKE

Para o bom desempenho do acionamento proposto é necessário boa precisão na obtenção das magnitudes das variáveis estimadas e da orientação de fluxo. A Figura A.2 mostra as correntes α e β reais e estimadas. Vê-se que os valores se assemelham bastante. Os componentes $\alpha - \beta$ do fluxo de rotor estimado são mostrados na Figura A.3(a). A Figura A.3(b) mostra que as componentes estão defasadas de aproximadamente 90° .

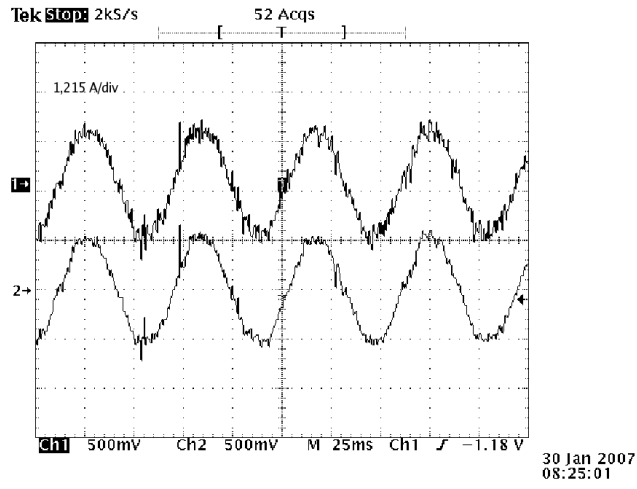
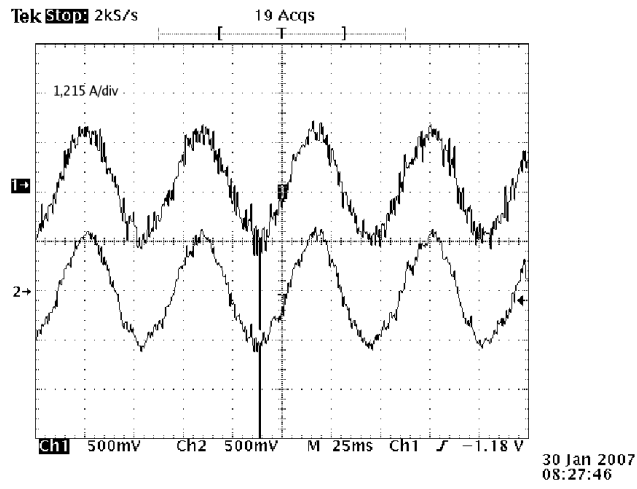
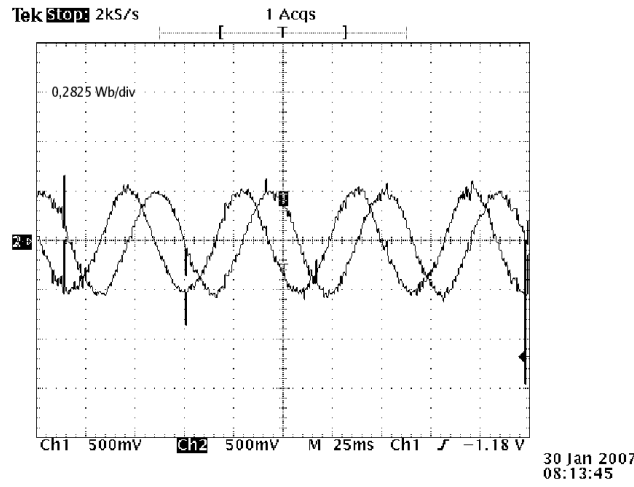
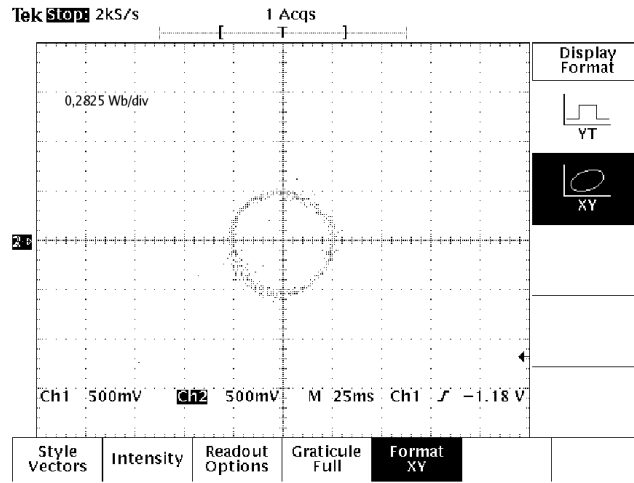
(a) $i_{s\alpha}$ medido (canal 1) e estimado (canal 2)(b) $i_{s\beta}$ medido (canal 1) e estimado (canal 2)

Figura A.2: Correntes reais e estimadas.

(a) $\Phi_{r\alpha}$ e $\Phi_{r\beta}$ (b) $\Phi_{r\alpha} \times \Phi_{r\beta}$ Figura A.3: Componentes $\alpha - \beta$ estimadas do fluxo de rotor para 518 rpm.

A.6 Breve revisão dos estimadores de estados para MI

O algoritmo do FKE é complexo e tem custo computacional elevado. Para sistemas em que os estados a serem estimados têm comportamento estocástico o FKE é uma das alternativa mais eficientes, no entanto, para plantas que possuam comportamento determinístico, como é o caso do MI, em que as medições não são contaminadas por ruído

acentuado, há uma gama de soluções com bom desempenho. Este tópico oferece uma breve revisão bibliográfica dos textos dedicados à estimação de estados do MI.

Vas (1998) apresentou a descrição de técnicas de estimação clássicas, bem como mais modernas que empregam técnicas de inteligência artificial. O livro aborda como realizar o controle por orientação de fluxo e direto de torque sem sensores de velocidade.

Valdenebro (2001) comparou experimentalmente três métodos estimação de fluxo de rotor e velocidade para controle empregado orientação de fluxo. São estudados o FKE e os observadores de Luenberger e em modos deslizantes. Outro autor que realizou comparação entre estimadores de estado foi Cuibus et al. (2000); o FKE e o observador de Luenberger foram implementados experimentalmente, enquanto que a estimação de fluxo de rotor e velocidade via RNA foi avaliada por simulação.

Foi estudado em (Du e Brdys, 1993; Du e Brdys, 1991) a estimação de estado, medindo correntes e tensões de estator, via observador de Luenberger com três abordagens, estimando:

1. Φ_r e τ_r ;
2. Φ_r e ω_r ,
3. Φ_r , ω_r e T_c

Na primeira alternativa considera-se que $\dot{\tau}_r \approx 0$ e na segunda $\dot{\omega}_r \approx 0$. Segundo o autor não é possível estimar τ_r e ω_r simultaneamente. A terceira alternativa emprega o modelo do motor de indução com as equações de tensão -de estator e rotor- e a equação eletromecânica. Considera-se $\dot{T}_c \approx 0$. Salienta-se que os modelos empregados nas segunda e terceira alternativa são os mesmos empregados na presente tese.

Ohnishi et al. (1994) apresentaram técnicas para obtenção de parâmetros e estados de motores *brushless* e de indução, para controle *sensorless*. São analisadas propostas para determinação da velocidade mecânica, da inércia total do acoplamento motor-carga e do torque de carga.

APÊNDICE B

Dados empregados

B.1 Dados do motor

- Tensão nominal 220/380 V,
- Corrente nominal 8,4/4,86 A,
- Frequência nominal 60 Hz,
- Potência nominal 3 HP,
- Velocidade nominal 1730 rpm,
- Número de pólos 4,
- $R_s = 2,65 \, \Omega$, $R_r = 1,8755 \, \Omega$, $L_m = 0,19634 \, \text{H}$, $L_{ls} = L_{lr} = 9,95862 \, \text{mH}$.

APÊNDICE C

Lei de controle

O algoritmo preditivo consiste em minimizar a função custo na qual esta presente o erro entre a predição baseada no modelo e as referências futuras, bem como um termo adicional que melhora a estabilidade numérica.

A predição de um processo em espaço de estados é dada por

$$\hat{\mathbf{y}} = H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} \quad (\text{C.1})$$

Com as matrizes H' , H'' e H''' sendo definidas na página 15. \mathbf{x} representa o vetor de estados, \mathbf{u} vetor dos sinais de controle e vetor distúrbio \mathbf{d} . A função custo quadrática, na sua forma matricial, é dada por

$$J = (\hat{\mathbf{y}} - \mathbf{w})^T W_y (\hat{\mathbf{y}} - \mathbf{w}) + \mathbf{u}^T W_u \mathbf{u} \quad (\text{C.2})$$

- $\mathbf{w} \in \Re^{(n_y q) \times 1}$ é o vetor de referências futuras;
- $W_y \in \Re^{(n_y q) \times (n_y q)}$, matriz definida-positiva (geralmente é diagonal) que permite enfatizar individualmente cada predição das saídas;
- $W_u \in \Re^{n_u \times n_u}$, matriz definida-positiva e usualmente diagonal que pondera o esforço de controle das entradas.

Substituindo-se a predição (equação (C.1)) na função custo encontra-se

$$J = (H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} - \mathbf{w})^T W_y (H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} - \mathbf{w}) + \mathbf{u}^T W_u \mathbf{u}$$

Encontra-se as entradas de controle que minimizam a função custo derivando-se J em relação a \mathbf{u} e igualando a zero. A derivada de J é dada por

$$\frac{\partial J}{\partial \mathbf{u}} = 2 \left[\frac{\partial}{\partial \mathbf{u}} (H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} - \mathbf{w}) \right]^T W_y (H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} - \mathbf{w}) + 2\mathbf{u}^T W_u \mathbf{u}$$

Do termo entre colchetes apenas $H''\mathbf{u}$ depende da entrada, resultando em

$$\frac{\partial J}{\partial \mathbf{u}} = 2H''^T W_y (H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} - \mathbf{w}) + 2\mathbf{u}^T W_u \mathbf{u}$$

O mínimo é determinado fazendo-se a derivada igual a zero

$$\frac{\partial J}{\partial \mathbf{u}} = 0$$

ou seja,

$$2H''^T W_y (H'\mathbf{x} + H''\mathbf{u} + H'''\mathbf{d} - \mathbf{w}) + 2\mathbf{u}^T W_u \mathbf{u} = 0$$

manipulando a equação anterior

$$H''^T W_y H'\mathbf{x} + H''^T W_y H''\mathbf{u} + H''^T W_y H'''\mathbf{d} - H''^T W_y \mathbf{w} + \mathbf{u}^T W_u \mathbf{u} = 0$$

$$(H''^T W_y H'' + W_u)\mathbf{u} = -H''^T W_y (H'\mathbf{x} + H'''\mathbf{d} - \mathbf{w})$$

Encontra-se

$$\mathbf{u} = (H''^T W_y H'' + W_u)^{-1} H''^T W_y (\mathbf{w} - H'\mathbf{x} - H'''\mathbf{d}) \quad (\text{C.3})$$

APÊNDICE D

Programa

```
#include "DSP281x_Device.h"      // DSP281x Headerfile Include File
#include "DSP281x_Examples.h"    // DSP281x Examples Include File
#include "Control_Output.h"
#include "IQmathLib.h"
#include "DSP281x_GlobalPrototypes.h"
#define GLOBAL_Q 17
long GlobalQ = GLOBAL_Q; // Colocado para a visualização no IQGel
// Prototype statements for functions found within this file.
void t1_t2(void); // Rotina SVM
void scope(_iq16 input, _iq16 max, int canal);
void configura_adc(void);
void inicializacao(void);
void teste1(void);
interrupt void eva_timer1_isr(void); // Interrupção timer1 EVA
interrupt void evb_timer3_isr(void); // Interrupção timer3

// Declaração de variáveis para o timer
Uint32  EvaTimer1InterruptCount;
Uint32  EvbTimer3InterruptCount;
//////////Variáveis para testes e para o SVM//////////
_iq ok, bode2;
int A, B, C, setor, bode1;
_iq t1, t2, t11_sat, t22_sat;
_iq vsalpha, vsbeta, X, Y, Z;
_iq vref1, vref2, vref3, ta, tb, tc;
_iq angulo, delta_angulo;
_iq t1_sat, t2_sat, t1_aux, t2_aux;
_iq periodopwm;
```

```

_iq pi_2,magmag;
_iq vsd_ref,vsq_ref,vsalpha_ast,vsbeta_ast;
//////////Variáveis para medição da velocidade//////////
int delta_thetar,thetar1=0,thetar2=0;
_iq omegar_pu,omegar_pu_s,delta_t,delta_rad,omegar_rads;
_iq cte_rad;

//////////Variáveis para medição da velocidade//////////
int delta_thetar2,thetar12=0,thetar22=0;
_iq omegar_pu2,omegar_pu_s,delta_t2,delta_rad2,omegar_rads2;
_iq cte_rad2;

/// Variáveis para conversão AD//////////
int tensaovab,tensaovbc,corrente_a,corrente_b;
_iq multi,vab,vbc,ia,ib,is,vvs;
_iq ia_amperes,ib_amperes,vab_volts,vbc_volts;
_iq us_alpha_pu_s,us_beta_pu_s,al_va,al_vb;
/// Fim de variáveis para conversão AD//////////

//Variáveis para transformação em pu
_iq vsd,vsq,us_alpha_pu,us_beta_pu,is_alpha_med_pu,is_beta_med_pu;

//Variáveis do FKE
int ii,jj,kk,nn,mm,pp;
_iq TEMP,TEMP2, x_pred[5],al,al1,x_est_s; //Vetor de estados preditos
_iq x_est[5],f_kmais1[5][5],aa[5][5];
_iq f_kmais1_vezes_P[5][5],erro_is_alpha,erro_is_beta;
_iq Q[5];
_iq R;
_iq P_ast_vezes_hT[5][2],inv_do_fke[2][2];
_iq K_fke[5][2],dif_y_med_menos_y_est[2];
_iq K_fke_vezes_dif_y_med_menos_y_est[5];
_iq h_vezes_P[2][5],hP[2][2],inv_hP[2][2];
_iq determinante,inv_do_determinante;
_iq K_fke_vezes_h_vezes_P[5][5];
_iq refs_Hlinhaxx_H3linhasxGdd[4];
_iq mag_phir,sen_theta=_IQ(0),cos_theta=_IQ(1);
//Para testar. Inicializarei algumas variáveis
_iq omegag;
_iq is_alpha_med,is_beta_med;
_iq us_alpha,us_beta;

```

```

//Passo 1: Inicializar P_ast
_iq P_ast[5][5];
_iq isd,isq,phird,phirq; //Estas variáveis foram usadas no MBPC também

//Referências do fluxo e da velocidade

_iq reftrapwr=_IQ(0),reftrapwr1=_IQ(0),deltatrapwr=_IQ(0.0),refmaxwr=_IQ(0.2);
_iq angulotrapwr=_IQ(0.02),refminwr=_IQ(0);
int rwr=0,rphir=0,teste;
_iq reftrapphir=_IQ(0),reftrapphir1=_IQ(0),deltatrapphir=_IQ(0.009);
_iq refmaxphir=_IQ(0.5),angulotrapphir=_IQ(0.009),refminphir=_IQ(0);

_iq alisq=_IQ(0.9),isq_s=_IQ(0),x_est_s_fora=_IQ(0);

//Parametros do MBPC
_iq us_d_antt=_IQ(0),us_q_antt=_IQ(0),omegar_antt=_IQ(0);
int ii1,jj1,kk1,nn1,mm1,pp1;
int cf=0,mudar_ref=0;
_iq alpha_ref;
_iq Wu[2]={_IQ(1),_IQ(0.4)},TEMP1=_IQ(0);
_iq Wy[4]={_IQ(0.5),_IQ(10),_IQ(0.5),_IQ(10)};
_iq invff[2][2]={_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq H2linhasTWy[2][4]={_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq H3linhasxGdd[4]={_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq Hlinhaxx[4]={_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq omegar_ant=_IQ(0),omegag_mbpc=_IQ(0);
_iq vs_alpha_asterisco=_IQ(0),vs_beta_asterisco=_IQ(0);
_iq us_d_ant=_IQ(0),us_q_ant=_IQ(0);
_iq invffH2linhasTWy[2][4]={_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq refs[4]={_IQ(0.95),_IQ(0.33333),_IQ(0.95),_IQ(0.33333)};
_iq ref_auxv[3]={_IQ(0),_IQ(0),_IQ(0)},ref_auxf[3]={_IQ(0),_IQ(0),_IQ(0)};
_iq refs_Hlinhaxx[4]={_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq deltas_u_dq[2]={_IQ(0),_IQ(0)},us_d_aplicado=_IQ(0),us_q_aplicado=_IQ(0);
_iq Hlinha[4][6]={_IQ(0.089192),_IQ(0),_IQ(0.94545),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(1),_IQ(0),_IQ(0),_IQ(0.053788),_IQ(0),_IQ(0.92249),_IQ(0),_IQ(1.2464),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0)};
_iq x_d[6]={_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0),_IQ(0)};

//////////Início do programa fonte//////////
void main(void)
{
    inicializacao();

```

```

// Step 1. Initialize System Control:
// PLL, WatchDog, enable Peripheral Clocks
// This example function is found in the DSP281x_SysCtrl.c file.
    InitSysCtrl();

// Step 2. Initialize GPIO:
// This example function is found in the DSP281x_Gpio.c file and
// illustrates how to set the GPIO to it's default state.
// InitGpio(); // Skipped for this example

// Initialize only GPAMUX and GPBMUX for this test
    EALLOW;
    // Enable PWM pins
    GpioMuxRegs.GPAMUX.all = 0x03FF; //EVA PWM 1-6 pins
    GpioMuxRegs.GPBMUX.all = 0x0000;
    GpioMuxRegs.GPBDIR.all = 0xFFFF;
    GpioMuxRegs.GPFMUX.all = 0x0000;
    GpioMuxRegs.GPFDIR.all = 0xFFFF;
    EDIS;

// Step 3. Clear all interrupts and initialize PIE vector table:
// Disable CPU interrupts
    DINT;

// Initialize PIE control registers to their default state.
// The default state is all PIE interrupts disabled and flags
// are cleared.
// This function is found in the DSP281x_PieCtrl.c file.
    InitPieCtrl();

// Disable CPU interrupts and clear all CPU interrupt flags:
    IER = 0x0000;
    IFR = 0x0000;

// Habilitando interrupções

// Initialize the PIE vector table with pointers to the shell Interrupt
// Service Routines (ISR).
// This will populate the entire table, even if the interrupt
// is not used in this example. This is useful for debug purposes.
// The shell ISR routines are found in DSP281x_DefaultIsr.c.
// This function is found in DSP281x_PieVect.c.

```

```

InitPieVectTable();
InitAdc();//inicializa o ADC para o estado default
configura_adc();

//EvaRegs.EVAIMRA.bit.T1PINT = 1; //Habilita interrupções pelo GP1
//EvaRegs.EVAIFRA.bit.T1PINT = 1; //quanto período coincide com contador
EALLOW; // This is needed to write to EALLOW protected registers
PieVectTable.T1PINT = &eva_timer1_isr;
PieVectTable.T3PINT = &evb_timer3_isr;
//PieVectTable.T4PINT = &evb_timer4_isr;
EDIS; // This is needed to disable write to EALLOW protected registers
// Interrupts that are used in this example are re-mapped to
// ISR functions found within this file.

// Step 4. Initialize all the Device Peripherals:
// This function is found in DSP281x_InitPeripherals.c
//InitPeripherals(); // Not required for this example

// Step 5. User specific code, enable interrupts:
EvaTimer1InterruptCount = 0;
EvbTimer3InterruptCount = 0;

// Enable CPU INT1 which is connected to CPU-Timer 0:
//Habilita interrupção 4 para o T1PINT
PieCtrlRegs.PIEIER2.bit.INTx4 = 1;
//Habilita interrupção 4 para o T3PINT
PieCtrlRegs.PIEIER4.bit.INTx4 = 1;

// Enable TINT0 in the PIE: Group 1 interrupt 7
// PieCtrlRegs.PIEIER1.bit.INTx7 = 1;
IER |= (M_INT2 | M_INT4); // | M_INT3 | M_INT4 | M_INT5);

// Enable global Interrupts and higher priority real-time debug events:
EINT; // Enable Global interrupt INTM
ERTM; // Enable Global realtime interrupt DBGM

// inicialização dos comparadores
EvaRegs.CMPR1=2;
EvaRegs.CMPR2=0;
EvaRegs.CMPR3=0;

```

```

// EVA Configure T1PWM, T2PWM, PWM1-PWM6
// Initalize the timers
// Configuração do Timer1
EvaRegs.T1PR =1406;// Período
EvaRegs.T1CNT =0x0000;// Valor inicial do período
EvaRegs.T1CON.all = 0xB42;// Seta forma da onda e comparações

EvaRegs.EVAIMRA.bit.T1PINT = 1; //Habilita interrupções pelo GP1
EvaRegs.EVAIFRA.bit.T1PINT = 1; //quanto período coincide com contador
// Drive T1/T2 PWM by compare logic
EvaRegs.GPTCONA.bit.TCMPOE = 1;
// Polarity of GP Timer 1 Compare = Active low
EvaRegs.GPTCONA.bit.T1PIN = 1;
// Polarity of GP Timer 2 Compare = Active high
EvaRegs.GPTCONA.bit.T2PIN = 2;

EvaRegs.GPTCONA.bit.T1TOADC = 2; //Habilitando a chamada de medição
do AD pelo Timer 1
// Configuração do Timer 2
EvaRegs.T2PR=50000;
EvaRegs.T2CNT=0x0000;
EvaRegs.T2CON.all=0x187C;

// Configuração do Timer3
EvaRegs.T3PR =28120;// Período
EvaRegs.T3CNT =0x0000;// Valor inicial do período
EvaRegs.T3CON.all = 0xB42;// Seta forma da onda e comparações

EvaRegs.EVBIMRA.bit.T3PINT = 1; //Habilita interrupções pelo GP1
EvaRegs.EVBIFRA.bit.T3PINT = 1; //quanto período coincide com contador

EvaRegs.ACTRA.all = 0x0666;
EvaRegs.DBTCNA.all = 0x0DFC; // Desabilita deadband
EvaRegs.COMCONA.all = 0xA600; // Controla comparadores

// Step 5. User specific code, enable interrupts:

// Just sit and loop forever:
// PWM pins can be observed with a scope.
for(;;);
}
//////////Fim do programa fonte//////////
//////////Início da interrupção pelo timer1 EVA//////////

```

```

interrupt void eva_timer1_isr(void)
{
    GpioDataRegs.GPBSET.bit.GPIOB15=1;
    EvaTimer1InterruptCount++;

    thetar2=EvaRegs.T2CNT;
    delta_thetar=thetar2-thetar1;
    delta_rad=_IQmpyI32(_IQ(0.001308996939),delta_thetar);
    omegar_rads=_IQdiv(delta_rad,delta_t);
    omegar_pu=_IQdiv(omegar_rads,_IQ(188.5));
    thetar1=thetar2;

    /////////// Início da rotina para leitura de tensões e correntes////////////////////

    tensaovab=AdcRegs.ADCRESULT0>>4;
    tensaovbc=AdcRegs.ADCRESULT2>>4;
    correntea=AdcRegs.ADCRESULT5>>4;
    correnteb=AdcRegs.ADCRESULT7>>4;

    vab=_IQmpyI32(multi,tensaovab)-_IQ(1.5);
    vbc=_IQmpyI32(multi,tensaovbc)-_IQ(1.5);
    ia=_IQmpyI32(multi,correntea)-_IQ(1.5);
    ib=_IQmpyI32(multi,correnteb)-_IQ(1.5);

    //Medição da corrente. Se 2,7 A rms no motor equivale a 0,5 V de pico
    //na placa, então como a placa pode medir até 3 V valor máximo de
    corrente mensurável
    //é 3 (1,5/0,5) vezes maior, ou seja, 8,1 A rms ou 11,455 A pico. Para
    converter o
    //valor lido para Âmpere rms devo multiplicar o valor de tensão obtido do AD por
    //8,1/1,5=5,4 (para o valor de pico 11,455/1,5=7,6366); e para
    transformar em pu multiplico o
    //valor de tensão lido por 11,455/(6,873*1,5)=1,111111111 (sendo que
    //o 6,873 é o valor base da corrente)
    ia_amperes=_IQmpy(_IQ(5.4),ia);
    //scope(_IQ16(0.25),_IQ16(1),0);
    //scope(ia_amperes>>1,_IQ16(4.86),0);
    ib_amperes=_IQmpy(_IQ(5.4),ib);
    //scope(_IQ16(-0.25),_IQ16(1),1);
    //scope(ib_amperes>>1,_IQ16(4.86),1);
    //Para medição da tensão tem-se que 220 V rms no motor equivale a 1,4
    V de pico na

```

```

//na placa, então para obter o valor em V rms no dsp devo multiplicar o
//valor em volts por (220/1,4=157,1428). Para tranformar em pu devo
//multiplicar por (1/1,4=0,71428)
    vab_volts=_IQmpy(_IQ(149.28566),vab);
    //scope(_IQ16(0.5),_IQ16(1),2);
    //scope(vab_volts>>1,_IQ16(220),0);
    vbc_volts=_IQmpy(_IQ(149.28566),vbc);//,_IQ(0.9615384615));
    //scope(_IQ16(-0.5),_IQ16(1),3);
    //scope(vbc_volts>>1,_IQ16(220),1);
    //Calculo de grandezas em alpha e beta em V rms e A rms
us_alpha=_IQmpy(_IQ(0.6666666),vab_volts)+_IQmpy(_IQ(0.3333333),vbc_volts);
us_beta=_IQmpy(_IQ(0.5773502691896),vbc_volts);//(1/sqrt(3))*Vbc
is_alpha_med=ia_amperes;//ia
is_beta_med=_IQmpy(_IQ(0.57735026),ia_amperes)+_IQmpy(_IQ(1.154700),ib_amperes);
    //Encontrando as tensões alpha e beta em pu
    us_alpha_pu=_IQdiv(us_alpha,_IQ(220));
    us_beta_pu=_IQdiv(us_beta,_IQ(220));

//scope(us_alpha_pu>>1,_IQ16(0.5),0);
//scope(us_beta_pu>>1,_IQ16(0.5),1);

    //-----
vvs=_IQmpy(us_alpha_pu,us_alpha_pu)+_IQmpy(us_beta_pu,us_beta_pu);
vvs=_IQsqrt(vvs);
//scope(vvs>>1,_IQ16(0.576),0);
    //Encontrando as correntes alpha e beta em pu
    is_alpha_med_pu=_IQmpy(is_alpha_med,_IQ(0.2057636494));
    is_beta_med_pu=_IQmpy(is_beta_med,_IQ(0.2057636494));
//scope(is_alpha_med_pu>>1,_IQ16(1),2);
//scope(is_beta_med_pu>>1,_IQ16(1),3);
is=_IQmpy(is_alpha_med,is_alpha_med)+_IQmpy(is_beta_med,is_beta_med);
is=_IQsqrt(is);
//scope(is>>1,_IQ16(5.4),1);
//Passo 2: Predição do vetor de estados (x_pred)
aa[0][0]=_IQ(0.93517);
aa[0][1]=_IQmpy(_IQ(0.0014827),x_est[4]);
aa[0][2]=_IQ(0.015475)+_IQmpy(_IQ(0.037604),_IQmpy(x_est[4],x_est[4]));
aa[0][3]=_IQmpy(_IQ(0.64085),x_est[4]);
aa[0][4]=_IQ(0);
aa[1][0]=_IQmpy(_IQ(-0.0014827),x_est[4]);
aa[1][1]=_IQ(0.93517);
aa[1][2]=_IQmpy(_IQ(-0.64085),x_est[4]);
aa[1][3]=_IQ(0.015475)+_IQmpy(_IQ(0.037604),_IQmpy(x_est[4],x_est[4]));

```



```

aa[1][4]=_IQ(0);
aa[2][0]=_IQ(0.0043038);
aa[2][1]=_IQmpy(_IQ(-0.00025219),x_est[4]);
aa[2][2]=_IQ(0.99731)-_IQmpy(_IQ(0.0063956),_IQmpy(x_est[4],x_est[4]));
aa[2][3]=_IQmpy(_IQ(-0.11131),x_est[4]);
aa[2][4]=_IQ(0);
aa[3][0]=_IQmpy(_IQ(0.00025219),x_est[4]);
aa[3][1]=_IQ(0.0043038);
aa[3][2]=_IQmpy(_IQ(0.11131),x_est[4]);
aa[3][3]=_IQ(0.99731)-_IQmpy(_IQ(0.0063956),_IQmpy(x_est[4],x_est[4]));
aa[3][4]=_IQ(0);
aa[4][0]=_IQ(0);
aa[4][1]=_IQ(0);
aa[4][2]=_IQ(0);
aa[4][3]=_IQ(0);
aa[4][4]=_IQ(1);

nn=5;
mm=5;
for(ii=0;ii<nn;ii++)
{
    for(jj=0;jj<mm;jj++)
    {
        TEMP=_IQmpy(aa[ii][jj],x_est[jj])+TEMP;
    }
    x_pred[ii]=TEMP;//Esta predição está sem as entradas
    TEMP=_IQ(0);
}

//Acrescentando as entradas us_alpha_pu e us_beta_pu
x_pred[0]=x_pred[0]+_IQmpy(_IQ(0.67525),us_alpha_pu);
x_pred[1]=x_pred[1]+_IQmpy(_IQ(0.67525),us_beta_pu);
x_pred[0]=x_pred[0]+_IQmpy(_IQ(0.001558),us_alpha_pu);
x_pred[1]=x_pred[1]+_IQmpy(_IQ(0.001558),us_beta_pu);
//Passo 3: Estimaco da matriz de covarincia da estimaco (P_ast)
f_kmais1[0][0]=_IQ(0.93517);
f_kmais1[0][1]=_IQmpy(_IQ(0.0014827),x_pred[4]);
f_kmais1[0][2]=_IQ(0.015475)+_IQmpy(_IQ(0.037604),_IQmpy(x_pred[4],x_pred[4]));
f_kmais1[0][3]=_IQmpy(_IQ(0.64085),x_pred[4]);
f_kmais1[0][4]=_IQmpy(_IQ(0.0014827),x_pred[1])+_IQmpy(_IQ(0.075208,
_IQmpy(x_pred[4],x_pred[2]))+_IQmpy(_IQ(0.64085),x_pred[3]));
f_kmais1[1][0]=_IQmpy(_IQ(-0.0014827),x_pred[4]);
f_kmais1[1][1]=_IQ(0.93517);

```

```

f_kmais1[1][2]=_IQmpy(_IQ(-0.64085),x_pred[4]);
f_kmais1[1][3]=_IQ(0.015475)+_IQmpy(_IQ(0.037604),_IQmpy(x_pred[4],x_pred[4]));
f_kmais1[1][4]=_IQmpy(_IQ(-0.0014827),x_pred[0])+_IQmpy(_IQ(0.075208)
,_IQmpy(x_pred[4],x_pred[3]))-_IQmpy(_IQ(0.64085),x_pred[2]);
f_kmais1[2][0]=_IQ(0.0043038);
f_kmais1[2][1]=_IQmpy(_IQ(-0.00025219),x_pred[4]);
f_kmais1[2][2]=_IQ(0.99731)-_IQmpy(_IQ(0.0063956),_IQmpy(x_pred[4],x_pred[4]));
f_kmais1[2][3]=_IQmpy(_IQ(-0.11131),x_pred[4]);
f_kmais1[2][4]=_IQmpy(_IQ(-0.00025219),x_pred[1])-_IQmpy(_IQ(0.0127912)
,_IQmpy(x_pred[4],x_pred[2]))-_IQmpy(_IQ(0.11131),x_pred[3]);
f_kmais1[3][0]=_IQmpy(_IQ(0.00025219),x_pred[4]);
f_kmais1[3][1]=_IQ(0.0043038);
f_kmais1[3][2]=_IQmpy(_IQ(0.11131),x_pred[4]);
f_kmais1[3][3]=_IQ(0.99731)-_IQmpy(_IQ(0.0063956),_IQmpy(x_pred[4],x_pred[4]));
f_kmais1[3][4]=_IQmpy(_IQ(0.00025219),x_pred[0])-_IQmpy(_IQ(0.0127912)
,_IQmpy(x_pred[4],x_pred[3]))+_IQmpy(_IQ(0.11131),x_pred[2]);
f_kmais1[4][0]=_IQ(0);
f_kmais1[4][1]=_IQ(0);
f_kmais1[4][2]=_IQ(0);
f_kmais1[4][3]=_IQ(0);
f_kmais1[4][4]=_IQ(1);

// Multiplicando f_kmais1 por P. A matriz P será calculada depois.
// Na primeira iteração utiliza-se o valor inicial
nn=5;
mm=5;
pp=5;

for(ii=0;ii<nn;ii++)
{
    for(kk=0;kk<pp;kk++)
    {
        for(jj=0;jj<mm;jj++)
        {
            TEMP=_IQmpy(f_kmais1[ii][jj],P_ast[jj][kk])+TEMP;
        }
        f_kmais1_vezes_P[ii][kk]=TEMP;
        TEMP=_IQ(0);
    }
}

//Multiplicando f_kmais1_vezes_P pela transposta de f_kmais1
nn=5;
mm=5;

```

```

pp=5;
for(ii=0;ii<nn;ii++)
{
    for(kk=0;kk<pp;kk++)
    {
        for(jj=0;jj<mm;jj++)
        {
            TEMP=_IQmpy(f_kmais1_vezes_P[ii][jj],f_kmais1[kk][jj])+TEMP;
            //Em f_kmais1 (aa) troquei o j e o k de posição
        }
        P_ast[ii][kk]=TEMP;//Este P_ast está sem somar Q. Somo em seguida
        TEMP=_IQ(0);
    }
}

//Somando os valores de Q à P_ast. Como
//Q é diagonal, então basta somar os valores da diagonal de Q aos respectivos
//elementos de P_ast, resultando no P_ast desejado
P_ast[0][0]=P_ast[0][0]+Q[0];
P_ast[1][1]=P_ast[1][1]+Q[1];
P_ast[2][2]=P_ast[2][2]+Q[2];
P_ast[3][3]=P_ast[3][3]+Q[3];
P_ast[4][4]=P_ast[4][4]+Q[4];
//Passo 4: Cálculo do ganho do FKE (K_fke)
//Multiplicando P_ast vezes hT off-line dá
//P_ast_vezes_hT[5][2]
P_ast_vezes_hT[0][0]=P_ast[0][0];
P_ast_vezes_hT[0][1]=P_ast[0][1];
P_ast_vezes_hT[1][0]=P_ast[1][0];
P_ast_vezes_hT[1][1]=P_ast[1][1];
P_ast_vezes_hT[2][0]=P_ast[2][0];
P_ast_vezes_hT[2][1]=P_ast[2][1];
P_ast_vezes_hT[3][0]=P_ast[3][0];
P_ast_vezes_hT[3][1]=P_ast[3][1];
P_ast_vezes_hT[4][0]=P_ast[4][0];
P_ast_vezes_hT[4][1]=P_ast[4][1];
//Abaixo o valor de h*P_ast*hT+R (hP[2][2])
hP[0][0]=P_ast[0][0]+R;
hP[0][1]=P_ast[0][1];
hP[1][0]=P_ast[1][0];
hP[1][1]=P_ast[1][1]+R;
determinante=_IQmpy(hP[0][0],hP[1][1])-_IQmpy(hP[1][0],hP[0][1]);
inv_do_determinante=_IQdiv(_IQ(1),determinante);
inv_hP[0][0]=_IQmpy(hP[1][1],inv_do_determinante);

```

```

inv_hP[1][1]=_IQmpy(hP[0][0],inv_do_determinante);
inv_hP[0][1]=_IQmpy(-hP[0][1],inv_do_determinante);
inv_hP[1][0]=_IQmpy(-hP[1][0],inv_do_determinante);
//Multiplicando P_ast_vezes_hT*inv_hP
nn=5;
mm=2;
pp=2;
for(ii=0;ii<nn;ii++)
{
    for(kk=0;kk<pp;kk++)
    {
        for(jj=0;jj<mm;jj++)
        {
TEMP=_IQmpy(P_ast_vezes_hT[ii][jj],inv_hP[jj][kk])+TEMP;
        }
        K_fke[ii][kk]=TEMP;
        TEMP=_IQ(0);
    }
}
//Passo 5:Estimação do vetor de estado x_est
//Considerando que isalpha_med e isbeta_med são os valores medidos da correntes
//estacionárias e x[1] e x[2] são os valores estimados destas mesmas
corrente, tem-se
dif_y_med_menos_y_est[0]=is_alpha_med_pu-x_pred[0];
dif_y_med_menos_y_est[1]=is_beta_med_pu-x_pred[1];
//Multiplicando o ganho do FKE K_fke pela diferença entre os valores
reais e medidos
//da corrente de estator tem-se
nn=5;
mm=2;
for(ii=0;ii<nn;ii++)
{
    for(jj=0;jj<mm;jj++)
    {
TEMP=_IQmpy(K_fke[ii][jj],dif_y_med_menos_y_est[jj])+TEMP;
    }
    K_fke_vezes_dif_y_med_menos_y_est[ii]=TEMP;
    TEMP=_IQ(0);
}
//x_est é dado por
x_est[0]=x_pred[0]+K_fke_vezes_dif_y_med_menos_y_est[0];
x_est[1]=x_pred[1]+K_fke_vezes_dif_y_med_menos_y_est[1];
x_est[2]=x_pred[2]+K_fke_vezes_dif_y_med_menos_y_est[2];

```

```

x_est[3]=x_pred[3]+K_fke_vezes_dif_y_med_menos_y_est[3];
x_est[4]=x_pred[4]+K_fke_vezes_dif_y_med_menos_y_est[4];

x_est_s=_IQmpy(x_est_s,al)+_IQmpy(_IQ(1)-al,x_est[4]);
x_est_s_fora=x_est_s-_IQ(0.333333);
scope(x_est_s_fora>>1,_IQ16(0.333333),1);
//scope(x_est_s>>1,_IQ16(0.5),3);
//scope(x_est_s>>1,_IQ16(0.5),1);

erro_is_alpha=is_alpha_med_pu-x_est[0];
erro_is_beta=is_beta_med_pu-x_est[1];
//Orientando
mag_phr=_IQmag(x_est[2],x_est[3]);
sen_theta=_IQdiv(x_est[3],mag_phr);
cos_theta=_IQdiv(x_est[2],mag_phr);

    t1_t2();//Rotina do SVM

isd=_IQmpy(is_alpha_med_pu,cos_theta)+_IQmpy(is_beta_med_pu,sen_theta);
//scope(isd>>1,_IQ16(1),0);
isq=_IQmpy(-is_alpha_med_pu,sen_theta)+_IQmpy(is_beta_med_pu,cos_theta);

scope(isd>>1,_IQ16(2),2);
scope(isq>>1,_IQ16(2),3);

phird=_IQmpy(x_est[2],cos_theta)+_IQmpy(x_est[3],sen_theta);
phirq=_IQmpy(-x_est[2],sen_theta)+_IQmpy(x_est[3],cos_theta);
vsd=_IQmpy(us_alpha_pu,cos_theta)+_IQmpy(us_beta_pu,sen_theta);
vsq=_IQmpy(-us_alpha_pu,sen_theta)+_IQmpy(us_beta_pu,cos_theta);
omegag=x_est_s+_IQmpy(_IQ(0.0321),_IQdiv(isq,phird));

//Passo 6: Achando a matriz de covariancia do erro de estimacao
//Multiplicando em off-line h vezes P_ast tenho a matriz h_vezesP[2][5] dada por
h_vezes_P[0][0]=P_ast[0][0];
h_vezes_P[0][1]=P_ast[0][1];
h_vezes_P[0][2]=P_ast[0][2];
h_vezes_P[0][3]=P_ast[0][3];
h_vezes_P[0][4]=P_ast[0][4];
h_vezes_P[1][0]=P_ast[1][0];
h_vezes_P[1][1]=P_ast[1][1];
h_vezes_P[1][2]=P_ast[1][2];
h_vezes_P[1][3]=P_ast[1][3];
h_vezes_P[1][4]=P_ast[1][4];

```

```

//Multiplicando K_fke vezes h_vezes_P encontra-se K_fke_vezes_h_vezes_P[5][5]
nn=5;
mm=2;
pp=5;
for(ii=0;ii<nn;ii++)
{
    for(kk=0;kk<pp;kk++)
    {
        for(jj=0;jj<mm;jj++)
        {
            TEMP=_IQmpy(K_fke[ii][jj],h_vezes_P[jj][kk])+TEMP;
        }
        K_fke_vezes_h_vezes_P[ii][kk]=TEMP;
        TEMP=_IQ(0);
    }
}

//Subtraindo P_ast de K_fke_vezes_h_vezes_P encontra-se
nn=5;
for(ii=0;ii<nn;ii++)
{
    for(jj=0;jj<nn;jj++)
    {
        P_ast[ii][jj]=P_ast[ii][jj]-K_fke_vezes_h_vezes_P[ii][jj]; //Este
P_ast é o P_chapeu
    }
}

//Reabilitando o conversor A/D
AdcRegs.ADCTRL2.bit.RST_SEQ1=1;
AdcRegs.ADCTRL2.bit.RST_SEQ2=1;
//Escreve 1 no bit do flag da interrupção de SEQ2
AdcRegs.ADCST.bit.INT_SEQ2_CLR=1;
AdcRegs.ADCST.bit.INT_SEQ1_CLR=1;

// Enable more interrupts from this timer
EvaRegs.EVAIMRA.bit.T1PINT = 1;

// Note: To be safe, use a mask value to write to the entire
// EVAIFRA register. Writing to one bit will cause a read-modify-write
// operation that may have the result of writing 1's to clear
// bits other than those intended.
EvaRegs.EVAIFRA.bit.T1PINT = 1;

```

```

// Acknowledge interrupt to receive more interrupts from PIE group 2
PieCtrlRegs.PIEACK.all = PIEACK_GROUP2;

GpioDataRegs.GPBCLEAR.bit.GPIOB15=1;

}
interrupt void evb_timer3_isr(void)
{
    GpioDataRegs.GPASET.bit.GPIOB14=1;
    EvbTimer3InterruptCount++;

// Enable more interrupts from this timer
    EvbRegs.EVBIMRA.bit.T3PINT = 1;

// Note: To be safe, use a mask value to write to the entire
// EVAIFRA register. Writing to one bit will cause a read-modify-write
// operation that may have the result of writing 1's to clear
// bits other than those intended.
    EvbRegs.EVBIFRA.bit.T3PINT = 1;
    thetar22=EvaRegs.T2CNT;
    delta_thetar2=thetar22-thetar12;
    delta_rad2=_IQmpyI32(_IQ(0.001308996939),delta_thetar2);
    omegar_rads2=_IQdiv(delta_rad2,delta_t2);
    omegar_pu2=_IQdiv(omegar_rads2,_IQ(188.5));
    thetar12=thetar22;
    omegar_pu_s=_IQmpy(omegar_pu_s,a11)+_IQmpy(_IQ(1)-a11,omegar_pu);
    scope(omegar_pu_s>>1,_IQ16(0.5),0);

testel();

us_d_antt=vsd;
us_q_antt=vsq;
omegar_antt=x_est_s;
//x_d
x_d[0]=isd;
x_d[1]=isq;
x_d[2]=phird;
x_d[3]=x_est_s;//
x_d[4]=us_d_ant;
x_d[5]=us_q_ant;
omegag_mbpc=omegag;

```

```

invff[0][0]=_IQdiv(_IQ(1),_IQmpy(_IQ(1.5535),Wy[2])+Wu[0]);
invff[1][1]=_IQdiv(_IQ(1),Wu[1]+_IQmpy(_IQmpy(_IQmpy(_IQ(0.008),x_d[2])
,x_d[2],Wy[3])));

H2linhasTWy[0][2]=_IQmpy(_IQ(1.2464),Wy[2]);
H2linhasTWy[1][3]=_IQmpy(_IQmpy(_IQ(0.08959),x_d[2]),Wy[3]);

Hlinha[1][1]=_IQmpy(_IQ(0.0064119),x_d[2]);
Hlinha[1][2]=_IQmpy(_IQ(0.0064119),x_d[1]);
Hlinha[2][1]=_IQmpy(_IQ(0.20175),omegag_mbps);
Hlinha[3][0]=_IQmpy(_IQmpy(_IQ(-0.014504)
,x_d[2],omegag_mbps+_IQmpy(_IQ(0.00057189,x_d[1]));
Hlinha[3][1]=_IQmpy(_IQ(0.0042161),x_d[2]);
Hlinha[3][2]=_IQmpy(_IQmpy(_IQ(-0.085276),x_d[2]),x_d[3])+_IQmpy(_IQ(0.01247),x_d[1];
Hlinha[3][3]=_IQ(1)-_IQmpy(_IQmpy(_IQ(0.085274),
x_d[2]),x_d[2]);
Hlinha[3][5]=_IQmpy(_IQ(0.089599),x_d[2]);

H3linhasxGdd[1]=_IQmpy(_IQmpy(_IQ(-0.012824),x_d[1]),x_d[2])+x_d[3]-omegar_ant;
H3linhasxGdd[3]=_IQmpy(_IQ(0.085274),_IQmpy(_IQmpy(x_d[2],x_d[2]),x_d[3]))
-_IQmpy(_IQmpy(x_d[1],x_d[2]),_IQ(0.025648))+
_IQmpy(_IQ(2),x_d[3]-omegar_ant);

//multiplicando Hlinha e x_d
nn1=4;
mm1=6;
TEMP1=_IQ(0);
for(ii1=0;ii1<nn1;ii1++)
{
    for(jj1=0;jj1<mm1;jj1++)
    {
        TEMP1=_IQmpy(Hlinha[ii1][jj1],x_d[jj1])+TEMP1;
    }
    Hlinhaxx[ii1]=TEMP1;
    TEMP1=_IQ(0);
}

//invffH2linhaT=invff*H2linhaT
nn1=2;
mm1=2;
pp1=4;
TEMP1=_IQ(0);
for(ii1=0;ii1<nn1;ii1++)

```



```

{
    for(kk1=0;kk1<pp1;kk1++)
    {
        for(jj1=0;jj1<mm1;jj1++)
        {
            TEMP1=_IQmpy(invff[ii1][jj1],H2linhasTWy[jj1][kk1])+TEMP1;
        }
        invffH2linhasTWy[ii1][kk1]=TEMP1;
        TEMP1=_IQ(0);
    }
}

//Subtraindo refs de Hlinhaxx encontra-se
nn1=4;
for(ii1=0;ii1<nn1;ii1++)
{
    refs_Hlinhaxx[ii1]=refs[ii1]-Hlinhaxx[ii1];
}

//Subtraindo refs_Hlinhaxx de H3linhasxGdd
nn1=4;
for(ii1=0;ii1<nn1;ii1++)
{
    refs_Hlinhaxx_H3linhasxGdd[ii1]=refs_Hlinhaxx[ii1]-H3linhasxGdd[ii1];
}

//invffH2linhaT*(refs_Hlinhaxx_H3linhasxGdd)
nn1=2;
mm1=4;
TEMP1=_IQ(0);
for(ii1=0;ii1<nn1;ii1++)
{
    for(jj1=0;jj1<mm1;jj1++)
    {
        TEMP1=_IQmpy(invffH2linhasTWy[ii1][jj1],refs_Hlinhaxx_H3linhasxGdd[jj1])
    }
    deltas_u_dq[ii1]=TEMP1;
    TEMP1=_IQ(0);
}
us_d_aplicado=us_d_ant+deltas_u_dq[0];
//scope(us_d_aplicado>>1,_IQ16(1),0);
us_q_aplicado=us_q_ant+deltas_u_dq[1];
//scope(us_q_aplicado>>1,_IQ16(1),1);

```

```

us_d_ant=us_d_antt;
us_q_ant=us_q_antt;
omegar_ant=omegar_antt;

// Acknowledge interrupt to receive more interrupts from PIE group 2
PieCtrlRegs.PIEACK.all = PIEACK_GROUP4;

GpioDataRegs.GPBCLEAR.bit.GPIOB15=1;

}

/////Início da Rotina para calculos dos tempos de chaveamento do SVM/////
void t1_t2()
{
    if(ok==_IQ(0))
    {
        vsalpha=_IQmpy(_IQcos(angulo),magmag);
        vsbeta=_IQmpy(_IQsin(angulo),magmag);
    }
    else
    {
        vsalpha= _IQmpy(us_d_aplicado,cos_theta)-_IQmpy(us_q_aplicado,sen_theta);
        vsbeta= _IQmpy(us_d_aplicado,sen_theta)+_IQmpy(us_q_aplicado,cos_theta);
    }
    vsalpha_ast= _IQmpy(us_d_aplicado,cos_theta)-_IQmpy(us_q_aplicado,sen_theta);
    vsbeta_ast= _IQmpy(us_d_aplicado,sen_theta)+_IQmpy(us_q_aplicado,cos_theta);
    //scope(vsalp>>1,_IQ16(0.5),2);
    //scope(vsbet>>1,_IQ16(0.5),3);
    //scope(vsalp_ast>>1,_IQ16(0.5),2);
    //scope(vsbet_ast>>1,_IQ16(0.5),3);

    //scope(bode1>>1,_IQ16(1),0);
    angulo=angulo+delta_angulo;

    if(angulo>=_IQ(6.28))
    {
        angulo = _IQ(0);
    }
    //scope(bode2>>1,_IQ16(1),1); //_IQdiv(angulo,_IQ(6.28))
    vref1=vsbeta;
    vref2=_IQmpy(_IQ(0.5),_IQmpy(vsalp,_IQ(1.7302))-vsbet);
    vref3=_IQmpy(_IQ(0.5),_IQmpy(vsalp,_IQ(-1.7302))-vsbet);

    if(vref1>_IQ(0))
    {

```

```

A=1;
}
else
{
A=0;
}
if(vref2>_IQ(0))
{
B=1;
}
else
{
B=0;
}
if(vref3>_IQ(0))
{
C=1;
}
else
{
C=0;
}
setor=A+2*B+4*C; // Define o setor do hexágono
X=_IQmpy(_IQ(1.73205),_IQmpy(periodopwm,vsbeta));
Y=_IQmpy(_IQ(0.86603),_IQmpy(periodopwm,vsbeta))+_IQmpy(_IQ(1.5)
,_IQmpy(periodopwm,vsalpha));
Z=_IQmpy(_IQ(0.86603),_IQmpy(periodopwm,vsbeta))-_IQmpy(_IQ(1.5)
,_IQmpy(periodopwm,vsalpha));

if(setor==1)
{
t1=Z;
t2=Y;
}
if(setor==2)
{
t1=Y;
t2=-X;
}
if(setor==3)
{
t1=-Z;
t2=X;
}

```

```

}
if(setor==4)
{
t1=-X;
t2=Z;
}
if(setor==5)
{
t1=X;
t2=-Y;
}
if(setor==6)
{
t1=-Y;
t2=-Z;
}
//Calculo dos tempos de saturação caso t1+t2>tpwm
t1_aux=t1+t2;
if(t1_aux>periodopwm)
{
t11_sat = _IQdiv(periodopwm,t1_aux);
t1_sat=_IQmpy(t1,t11_sat);
t2_sat=_IQmpy(t2,t11_sat);
t1=t1_sat;
t2=t2_sat;
}
t2_aux=periodopwm-t1-t2;
ta=_IQmpy(t2_aux,_IQ(0.5));
tb=ta+t1;
tc=tb+t2;
//define os chaveamentos conforme o setor
if(setor==1)
{
EvaRegs.CMPR1=_IQint(tb);
EvaRegs.CMPR2=_IQint(ta);
EvaRegs.CMPR3=_IQint(tc);
}

else if(setor==2)
{
EvaRegs.CMPR1=_IQint(ta);
EvaRegs.CMPR2=_IQint(tc);
EvaRegs.CMPR3=_IQint(tb);

```

```

}

else if(setor==3)
{
    EvaRegs.CMPR1=_IQint(ta);
    EvaRegs.CMPR2=_IQint(tb);
    EvaRegs.CMPR3=_IQint(tc);
}

else if(setor==4)
{
    EvaRegs.CMPR1=_IQint(tc);
    EvaRegs.CMPR2=_IQint(tb);
    EvaRegs.CMPR3=_IQint(ta);
}

else if(setor==5)
{
    EvaRegs.CMPR1=_IQint(tc);
    EvaRegs.CMPR2=_IQint(ta);
    EvaRegs.CMPR3=_IQint(tb);
}

else if(setor==6)
{
    EvaRegs.CMPR1=_IQint(tb);
    EvaRegs.CMPR2=_IQint(tc);
    EvaRegs.CMPR3=_IQint(ta);
}

}

////////////////////Final da rotina do SVM////////////////////////////////////

//----- Rotina de visualização de dados com o DAC7625P
-----//

/*
Para usar esta rotina faça a seguinte modificação no arquivo DSP281x_GPio.h:

struct GPFDAT_BITS {          // bits   description
    Uint16 DADOS:12;          // 0-11
    Uint16 CANAL:2;            // 12-13
    Uint16 GPIOF14:1;          // 14
    Uint16 GPIOF15:1;          // 15

```

```

};
*/
void scope(_iq16 input, _iq16 max, int canal )
{
  GpioDataRegs.GPFDAT.bit.GPIOF14 = 1;
  GpioDataRegs.GPFDAT.bit.CANAL = canal;
  input = input + max;           //Soma offset.
  input = _IQ16div(input,(max<<1)); //Converte em pu.
  input = _IQ16mpy(input,_IQ16(4095)); //Multiplica pelo range de 12 bits.
  input = (input>>8)>>8;         //Coloca número nos 12 MSB.
  GpioDataRegs.GPFDAT.bit.DADOS = input;
  GpioDataRegs.GPFDAT.bit.GPIOF14 = 0;
}
//----- Rotina de visualização de dados com o DAC7625P
-----//

void configura_adc(void)
{
  //AdcRegs.ADCTRL1.bit.RESET = 1; // Reseta todo o sistema de conversão A/D
  AdcRegs.ADCTRL1.bit.CPS = 1;    // Ajusta a frequência de clock
para o conversor AD
  AdcRegs.ADCTRL1.bit.SEQ_CASC=1;
  AdcRegs.ADCTRL3.bit.ADCCLKPS = 100;

  AdcRegs.ADCMAXCONV.all = 0x00FF; // Setup 2 conv's on SEQ1
  AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0x00;
  AdcRegs.ADCCHSELSEQ1.bit.CONV02 = 0x02;
  AdcRegs.ADCCHSELSEQ2.bit.CONV05 = 0x05;
  AdcRegs.ADCCHSELSEQ2.bit.CONV07 = 0x07;
  AdcRegs.ADCTRL2.bit.EVA_SOC_SEQ1 = 1; // Impede início da
conversão pelo EVA
  AdcRegs.ADCTRL2.bit.INT_ENA_SEQ1= 0; // Desabilita pedido de interrupção
conversão pelo EVA
  AdcRegs.ADCTRL2.bit.INT_ENA_SEQ2 = 0; // Desabilita pedido de interrupção
  //EvaRegs.GPTCONB.bit.T3TOADC=2; // Habilita o início do
adc em cada periodo de timer
}
void teste1(void)
{
  scope(refs[1]>>1,_IQ16(0.5),3);
  refs[1]=refs[3];
  angulotrapwr=_IQ(0.006666);
  //inicio de referência trapezoidal para a velocidade

```

```

reftrapwr1=reftrapwr1+deltatrapwr;
reftrapwr=reftrapwr1-_IQ(0.33333);
if (reftrapwr1 >= _IQ(0.66666) )
{
if (rwr <= 150)
{
deltatrapwr = 0;
rwr++;
}
else
{
deltatrapwr = -angulotrapwr;
rwr=0;
}
}
if (reftrapwr1 <= _IQ(0) )
{
if (rwr <= 150)
{
deltatrapwr = 0;
rwr++;
}
else {deltatrapwr = angulotrapwr;
rwr=0;
}
}
//Mudando a referencia ny=2 de velocidade
refs[3]=reftrapwr;
//Fim de referência trapezoidal da velocidade
}

void inicializacao(void)
{
//////////Variáveis para medição da velocidade//////////
thetar1=0,thetar2=0;
omegar_pu=_IQ(0),delta_t=_IQ(0.0003),delta_rad=_IQ(0),omegar_rads=_IQ(0);
cte_rad=_IQ(0.0002083333333333);

//////////Variáveis para medição da velocidade//////////
thetar12=0,thetar22=0;
omegar_pu2=_IQ(0),delta_t2=_IQ(0.006),delta_rad2=_IQ(0),omegar_rads2=_IQ(0);
cte_rad2=_IQ(0.0002083333333333);

```

```

/// Variáveis para conversão AD//////////
multi=_IQ(0.00073260073260073260073260073260073);
ia_amperes=_IQ(0),ib_amperes=_IQ(0),vab_volts=_IQ(0),vbc_volts=_IQ(0);
/// Fim de variáveis para conversão AD//////////

//Variáveis para transformação em pu
us_alpha_pu=_IQ(0),us_beta_pu=_IQ(0),is_alpha_med_pu=_IQ(0),is_beta_med_pu=_IQ(0);
us_alpha_pu_s=_IQ(0),us_beta_pu_s=_IQ(0),al_va=_IQ(0.1),al_vb=_IQ(0.1);
//Variaveis do FKE
TEMP=_IQ(0),TEMP2=_IQ(0);
//Vetor de estados preditos
x_pred[0]=_IQ(0);
x_pred[1]=_IQ(0);
x_pred[2]=_IQ(0);
x_pred[3]=_IQ(0);
x_pred[4]=_IQ(0);
//Vetor de estados estimados
x_est[0]=_IQ(0);
x_est[1]=_IQ(0);
x_est[2]=_IQ(0);
x_est[3]=_IQ(0);
x_est[4]=_IQ(0);
//Vetor de covariância (na verdade é uma matriz, mas não precisa
representar tudo)
Q[0]=_IQ(0.015259);
Q[1]=_IQ(0.015259);
Q[2]=_IQ(0.004577);//0.000228818
Q[3]=_IQ(0.004577);
Q[4]=_IQ(0.007629);//0.06866455
R=_IQ(0.30518);//0.06103516/
//Para testar. Inicializarei algumas variáveis
omegag=_IQ(0);
is_alpha_med=_IQ(0),is_beta_med=_IQ(0);
us_alpha=_IQ(0),us_beta=_IQ(0);

//Passo 1: Inicializar P_ast
for(jj=0;jj<5;jj++)
{
    P_ast[jj][jj]=_IQ(0);
}
P_ast[0][0]=_IQ(0.004882);
P_ast[1][1]=_IQ(0.004882);
P_ast[2][2]=_IQ(0.004882);

```



```

P_ast[3][3]=_IQ(0.004882);
P_ast[4][4]=_IQ(0.004882);
al=_IQ(0.99);
al1=_IQ(0.8019);
alpha_ref=_IQ(0.75);
x_est_s=_IQ(0);
//////////Variáveis para testes e para o SVM//////////
ok=_IQ(0);
vsalpha=_IQ(0),vsbeta=_IQ(0);
angulo=_IQ(0),delta_angulo=_IQ(0.032648388559480760166641484654223);
magmag=_IQ(0.2886751345948128822545743902505);//
periodopwm=_IQ(1406);
pi_2=_IQ(1.570796);
teste=0;
vsd_ref=_IQ(0.01),vsq_ref=_IQ(0.27);
}

```

Referências Bibliográficas

- Albertos, P. e Ortega, R. (1989). Properties of generalized predictive control, *Automatica* **25**(5): 753–755.
- Altuna, J. A. T. (2002). *Análise dinâmica de controladores de corrente para máquinas de indução trifásicas alimentadas por inversor PWM*, Tese de doutorado, Universidade Estadual de Campinas.
- Bertonha, L. C. (2006). *Controle escalar v/f do motor de indução monitorando o fluxo de entreferro via terceira harmônica de tensão*, Dissertação de mestrado, Universidade Estadual de Campinas.
- Bose, B. K. (1994). Expert system, fuzzy logic, and neural network applications in power electronics and motion control, *Proceeding of the IEEE- Special issue on power electronics and motion control* **82**(8): 1303–1323.
- Camacho, E. (1993). Constrained generalized predictive control, *IEEE Transactions on Automatic Control* **38**(2): 327–332.
- Camacho, E. F. e Bordons, C. (1999). *Model Predictive Control*, Springer-Verlag.
- Campello, R. J. G. B. e do Amaral, W. C. (2002). *Arquiteturas e Metodologias para Modelagem e Controle de Sistemas Complexos Utilizando Ferramentas Clássicas e Modernas*, Tese de doutorado, Universidade Estadual de Campinas.
- Chen, W.-H., Ballance, D. J. e O'Reilly, J. (2000). Model predictive control of nonlinear systems: computational burden and stability, *IEE Proceeding on Control Theory and Applications* **147**(4): 387–394.
- Clarke, D. W. (1988). Application of generalized predictive control to industrial processes, *IEEE Control Systems Magazine* **8**(2): 49–55.
- Clarke, D. W. e Mohtadi, C. (1989). Properties of generalized predictive control, *Automatica* **25**(6): 859–875.
- Clarke, D. W., Mohtadi, C. e Tuffs, P. S. (1987a). Generalized predictive control - part i. the basic algorithm, *Automatica* **23**(2): 137–148.
- Clarke, D. W., Mohtadi, C. e Tuffs, P. S. (1987b). Generalized predictive control - part ii. extension and interpretations, *Automatica* **23**(2): 149–160.
- Coelho, A. A. R. e do Amaral, W. C. (1991). *Arquiteturas e Metodologias para Modelagem e Controle de Sistemas Complexos Utilizando Ferramentas Clássicas e Modernas*, Tese de doutorado, Universidade Estadual de Campinas.

- Cuibus, M., Bostan, V., Ambrosii, S., Ilas, C. e Magureanu, R. (2000). Luenberger, kalman and neural network observers for sensorless induction motor control, *The Third International Power Electronics and Motion Control Conference*, Vol. 3, pp. 1256–1261.
- Cutler, C. R. e Ramaker, B. C. (1980). Dynamic matrix control - a computer control algorithm, *Proceeding of Automatic Control Conference*.
- de Araujo Silva, L. (2000). *Avaliação de uma nova proposta de controle v/f em malha aberta*, Dissertação de mestrado, Universidade Estadual de Campinas.
- de Santana, E. S., Bim, E. e Amaral, W. C. (2006). Induction motor control using predictive algorithm, *17th International Conference on Electrical Machines*.
- der Klauw, A. C. V. e el Ata-Doss, S. A. (1991). Predictive functional control of flexible structures: a composite control approach, *International Conference on Control*, Vol. 2, pp. 1177–1182.
- Doncker, R. W. D. e Novotny, D. W. (1988). the universal field orientes controller, *Conference Recoder IEEE-IAS Annual Meeting*, pp. 450–456.
- Draeger, A., Engell, S. e Ranke, H. (1995). Model predictive control using neural networks, *IEEE Control Systems Magazine* **15**(5): 61–66.
- Du, T. e Brdys, M. A. (1991). Algorithms for joint state and parameter estimation in induction motor drive systems, *International Conference on Control*, Vol. 2, pp. 915–920.
- Du, T. e Brdys, M. A. (1993). Implementation of extended luenberger observers for joint and parameter estimation of pwm induction motor drive, *EPE*, pp. 439–444.
- el Ata-Doss, S. A., Fiani, P. e Richalet, J. (1991). Handling input and state constraints in predictive functional control, *Proceedings of the 30th Conference on Decision and Control*, Vol. 1, pp. 985–990.
- Fontes, F. A. e Magni, L. (2003). Min-max model predictive control of nonlinear systems using discontinuous feedbacks, *IEEE Transactions on Automatic Control* **48**(10): 1750–1755.
- Garcia, C. E., Prett, D. M. e Morari, M. (1989). Model predictive control: theory and practice - a survey, *Automatica* **25**(3): 335–348.
- Gawthrop, P. J., Demircioglu, H. e Siller-Alcala, I. I. (1998). Multivariable continuous-time generalised predictive control: a state-space approach to linear and nonlinear systems, *IEE Proceeding on Control Theory and Applications* **145**(3): 241–250.
- Geng, G. e Geary, G. M. (1998). The control of input-constrained nonlinear processes using numerical generalized predictive control methods, *IEEE Transactions on Industrial Electronics* **45**(3): 496–501.
- Guo-yu, W., Pu, H., Dong-feng, W. e Wan-ye, Y. (2002). Predictive functional control of integrating plant, *TENCON '02. Proceedings. 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*, Vol. 3, pp. 1401–1403.
- Hedjar, R., Toume, R., Boucher, P. e Dumur, D. (2004). End poit constraints nonlinear predictive control with integral action for induction motor, *Proceedings of the 2004 IEEE International Conference on Control Applications*, Vol. 1, pp. 1691–1696.

- Holtz, J. (1994). Pulsewidth modulation for electronic power conversion, *Proceeding of the IEEE- Special issue on power electronics and motion control* **82**(8): 1164–1194.
- Holtz, J. e Stadtfeld, S. (1983). A predictive controller for the stator current vector of ac machine fed from a switched voltage source, *Proceedings of the International Power Electronics Conference*, pp. 1665–1675.
- Hu, B. e Linnemann, A. (2002). Toward infinite-horizon optimality in nonlinear model predictive control, *IEEE Transactions on Automatic Control* **47**(4): 679–682.
- Kennel, R. e Linder, A. (2000). Predictive controle of inverter supplied electricla drives, pp. 761–766.
- Kennel, R., Linder, A. e Linke, M. (1997). Generalized predictive control (gpc)-ready for use in drive applications?, *Power Electronics Specialists Conference, 2001. PESC. 2001 IEEE 32nd Annual*, Vol. 4, pp. 1839–1844.
- Lindoff, B. e Holst, J. (1997). Adaptive predictive control for time-varying stochastic systems, *Proceedings of the 36th IEEE Conference on Decision and Control*, Vol. 3, pp. 2477–2482.
- Ling, K. V. e Lim, K. W. (1996). A state space gpc with extensions to multirate control, *Automatica* **32**(7): 1067–1071.
- Lorentz, R. D., Lipo, T. A. e Novotny, D. W. (1994). Motion controle with induction motors, *Proceeding of the IEEE- Special issue on power electronics and motion control* pp. 1215–1240.
- Low, K. S., Chiun, K. Y. e Ling, K. V. (1997). A dsp-based servo system using generalized predictive control, *Proceedings of the Power Conversion Conference*, Vol. 1, pp. 507–512.
- Low, K.-S., Chiun, K.-Y. e Ling, K.-V. (1998). Evaluating generalized predictive control for a brushless dc drive, *IEEE Transactions on Power Electronics* **13**(6): 1191–1198.
- Palm III, W. J. (1986). *Control Systems Engineering*, John Wiley & Sons.
- McDonnell, M. e el Ata-Doss, S. A. (1989). Predictive functional control of multivariable systems with more outputs than inputs, *Proceedings of the 28th IEEE Conference on Decision and Control*, Vol. 2, pp. 1591–1596.
- Mendalek, N., Fnaiech, F., Al-Haddad, K. e Dessaint, L.-A. (2002). A non-linear optimal predictive control of a shunt active power filter, *Industry Applications Conference, 2002. 37th IAS Annual Meeting*, Vol. 1, pp. 70–77.
- Ohnishi, K., Matsui, N. e Hori, Y. (1994). Estimation, identification , and sensorless controle in motio control system, *Proceeding of the IEEE- Special issue on power electronics and motion control* **82**(8): 1253–1265.
- Ordys, A. W. e Clarke, D. W. (1993). A state-space description for gpc controllers, *Int. J. Systems Sci.* **24**(9): 1727–1744.
- Pereira, G. A. S. (n.d.). Filtro de kalman: teoria e aplicações, *Technical report*.
- Poulsen, N. K., Kouvaritakis, B. e Cannon, M. (2001). Nonlinear constrained predictive control applied to a coupled-tanks apparatus, *IEE Proceedings of Control Theory and Applications* **148**(1): 17–24.

- Rani, K. Y. e Unbehauen, H. (1997). Study of predictive controller tuning methods, *Automatica* **33**(12): 2243–22482.
- Richalet, J., Estival, J. L. e Fiani, P. (1995). Industrial applications of predictive functional control to metallurgical industries, *Proceedings of the 4th IEEE Conference on Control Applications*, pp. 934–942.
- Rossiter, J. A. (2003). *Model Based Predictive Control - A Practical Approach*, CRC Press.
- Rossiter, J. A. e Richalet, J. (2002). Handling constraints with predictive functional control of unstable processes, *Proceedings of the American Control Conference*, Vol. 6, pp. 4746–4751.
- Shin, S. e Park, S. (1998). Ga-based predictive control for nonlinear processes, *Electronics Letters* **34**(20): 1980–1981.
- Skrjanc, I. e Matko, D. (2000). Predictive functional control based on fuzzy model for heat-exchanger pilot plant, *IEEE Transactions on Fuzzy Systems* **8**(6): 705–712.
- Soeterboek, R. (1992). *Predictive control - A unified approach*, Prentice Hall.
- Spe (2002). *eZdsp F2812TM Technical Reference*.
- Tex (2002). *IQmath Library - A Virtual Floating Point Engine*.
- Tex (2003a). *TMS320F28x DSP Analog-to-Digital Converter (ADC) Reference Guide*.
- Tex (2003b). *TMS320F28x DSP Event Manager (EV) Reference Guide*.
- Tsang, T. T. C. e Clarke, D. W. (1988). Generalised predictive control with input constraints, *IEE Proceedings D Control Theory and Applications (see also IEE Proceedings-Control Theory and Applications)* **135**(6): 451–460.
- Valdenebro, L. R. (2001). *Observadores adaptativos de fluxo e velocidade para motores de indução - estudo e implementação*, Tese de doutorado, Universidade Estadual de Campinas.
- Vas, P. (1998). *Sensorless vector and direct torque control*, Oxford science publications.
- Wams, B. e van den Boom, T. J. J. (1997). Youla-like parametrizations for predictive control of a class of linear time-varying systems, *Proceedings of the American Control Conference*, Vol. 3, pp. 2051–2052.
- Wan, Z. e Kothare, M. V. (2003). Efficient robust constrained model predictive control with a time varying terminal constraint set, *Systems & Control Letters* **48**(5): 375–383.
- Wang, D.-F., Han, P., Wang, G.-Y. e Lu, H.-M. (2002). Multi-model adaptive predictive functional control and its application, *Proceedings of First International Conference on Machine Learning and Cybernetics*, Vol. 4, pp. 2126–2130.
- Xu, X. e Novotny, D. W. (1992). Selection of the flux reference for induction machine drives in the field weakening region, *IEEE Transactions on Industry Applications* **28**(6): 1353–1358.
- Zhang, L., Norman, R. e Shepherd, W. (1997). Long-range predictive control of current regulated pwm for induction motor drives using the synchronous reference frame, *IEEE Transactions on control systems technology* **5**(1): 119–125.

- Zhang, Z. L., Shen, Y. e Wang, Y. (2004). Measure and dmc control of battery separator thickness uniformity, *8th International Conference on Control Automatic, Robotics and Vision*, pp. 851–854.