

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA**

Lucas Cielo Borin

**PROCEDIMENTO DE PROJETO DE CONTROLADORES
ROBUSTOS PARA CONVERSORES DE POTÊNCIA POR
MEIO DE OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS**

**Santa Maria, RS, Brasil
2020**

Lucas Cielo Borin

**PROCEDIMENTO DE PROJETO DE CONTROLADORES ROBUSTOS
PARA CONVERSORES DE POTÊNCIA POR MEIO DE OTIMIZAÇÃO
POR ENXAME DE PARTÍCULAS**

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Engenharia Elétrica, Área de Concentração em Processamento de Energia Elétrica, da Universidade Federal de Santa Maria (UFSM-RS), como requisito parcial para obtenção do grau de **Mestre em Engenharia Elétrica**.

Orientador: Prof. Dr. Vinícius Foletto Montagner

Santa Maria, RS, Brasil
2020

Ficha catalográfica elaborada através do Programa de Geração Automática da Biblioteca Central da UFSM, com os dados fornecidos pelo(a) autor(a).

Borin, Lucas Cielo

Procedimento de Projeto de Controladores Robustos para Conversores de Potência por meio de Otimização por Enxame de Partículas / Lucas Cielo Borin - 2020

79 p.; 30 cm

Orientador: Vinícius Foletto Montagner

Dissertação (mestrado) – Universidade Federal de Santa Maria, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, RS, 2020

1. Engenharia Elétrica 2. Controle Robusto 3. Conversores de Potência 4. Incertezas Paramétricas 5. Otimização por Enxame de Partículas I. Montagner, Vinícius Foletto.

© 2020

Todos os direitos autorais reservados a Lucas Cielo Borin. A reprodução de partes ou do todo deste trabalho só poderá ser feita com autorização por escrito do autor.

Endereço: Av. Roraima, Nº 1000, Bairro Camobi, Santa Maria, RS, Brasil, CEP: 97105-900;

Fone: (55) 3220-9497;

Endereço Eletrônico: lukascielo@gmail.com.

Lucas Cielo Borin

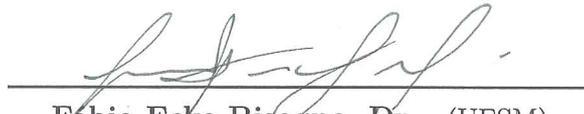
**PROCEDIMENTO DE PROJETO DE CONTROLADORES ROBUSTOS
PARA CONVERSORES DE POTÊNCIA POR MEIO DE OTIMIZAÇÃO
POR ENXAME DE PARTÍCULAS**

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Engenharia Elétrica, Área de Concentração em Processamento de Energia Elétrica, da Universidade Federal de Santa Maria (UFSM-RS), como requisito parcial para obtenção do grau de Mestre em Engenharia Elétrica.

Aprovado em 18 de Fevereiro de 2020:



Vinícius Foletto Montagner, Dr. (UFSM)
(Presidente/ Orientador)



Fábio Ecke Bisogno, Dr. (UFSM)



Fabrício Hoff Dupont, Dr. (UnoChapecó - Videoconferência)

Santa Maria, RS, Brasil
2020

DEDICATÓRIA

A todas as pessoas, que de alguma forma, tornaram possível a realização deste trabalho.

AGRADECIMENTOS

Primeiramente, agradeço ao Professor Vinícius Foletto Montagner pela orientação durante os últimos dois anos, que tornou possível a realização desta dissertação, por meio da troca de conhecimentos, dos valiosos ensinamentos e pela confiança depositada em mim. Também, estendo este agradecimento aos demais professores do Grupo de Eletrônica de Potência e Controle, pelos conhecimentos compartilhados.

A minha namorada, Isadora, por me amar, manter motivado e estar comigo em todos os momentos, sejam eles bons ou ruins, e aos meus pais, Rosmaria e Vanderlei, pelo amor e apoio, que sempre me manteve motivado.

Aos meus familiares, pelo amor, e a todos os meus amigos, pelas memórias de conversas e risadas compartilhadas.

A todos os colegas de laboratório e amigos, pelo ótimo ambiente de trabalho e pelo companheirismo, e um agradecimento especial para Gustavo, Thieli e Caio, pela ajuda, conversas, dicas e amizade que contribuíram diretamente na construção desta dissertação.

A Universidade Federal de Santa Maria, ao Programa de Pós-Graduação em Engenharia Elétrica, ao Centro de Tecnologia, ao Grupo de Eletrônica de Potência e Controle, aos demais professores e técnicos administrativos, por me proporcionarem um ensino de excelente qualidade, e a CAPES pelo apoio financeiro.

Por fim, a todos que de alguma forma me ajudaram e fazem parte da minha vida, e que por ventura posso ter esquecido de agradecer, muito obrigado por tudo.

“Pass on what you have learnt:
strength, mastery ... but
weakness, falling and failure also.
Yes, failure, most of all.
The greatest teacher, failure is.
Yoda

Those who can imagine
anything,
can create the impossible.
Alan Turing”

RESUMO

PROCEDIMENTO DE PROJETO DE CONTROLADORES ROBUSTOS PARA CONVERSORES DE POTÊNCIA POR MEIO DE OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS

AUTOR: LUCAS CIELO BORIN
ORIENTADOR: VINÍCIUS FOLETTO MONTAGNER

Esta dissertação fornece como contribuição um procedimento de projeto, baseado em otimização por enxame de partículas e no Teorema de Kharitonov, para obter ganhos fixos de controladores robustos para conversores de potência com incertezas paramétricas. Primeiramente, um modelo em função de transferência do conversor com incertezas paramétricas é utilizado e incluído em um modelo politópico. Na sequência, a função de transferência do controlador é escolhida, e seus coeficientes fixos são considerados como as variáveis de decisão de um problema de otimização. Uma função objetivo é então utilizada para mapear cada vetor de coeficientes do controlador em um valor real positivo, considerando os critérios margem de fase, frequência de cruzamento, margem de ganho, máximo sobressinal, erro de regime permanente e amplitude do sinal de controle, para os vértices do modelo politópico. Um fator de penalização baseado no Teorema de Kharitonov é incluído na função objetivo. Um espaço de busca dos coeficientes do controlador é definido de forma sistemática como um hiper-retângulo, a partir da positividade dos coeficientes do polinômio característico, e o algoritmo de otimização por enxame de partículas é então empregado para buscar o vetor de coeficientes do controlador que minimiza a função objetivo. Para comprovar a capacidade de encontrar controladores robustos para aplicações em eletrônica de potência, o procedimento proposto é utilizado para obtenção de controladores PIDs para um conversor *Buck* e de controladores PIs para um motor síncrono de ímãs permanentes, garantindo boa regulação das variáveis de saída em ambas as aplicações. Resultados experimentais para o motor síncrono de ímãs permanentes confirmam a viabilidade prática do procedimento proposto, com desempenho superior em comparação com ganhos obtidos por uma função especializada em projeto de PIDs. O procedimento proposto pode ser considerado uma alternativa a métodos analíticos e a projetos por tentativa e erro, permitindo reduzir o tempo demandado em interação homem-máquina para projetar controladores que atendam a múltiplos objetivos e restrições de ordem prática.

Palavras-chave: Controle robusto, conversores de potência, incertezas paramétricas, inteligência artificial, otimização por enxame de partículas.

ABSTRACT

**ROBUST CONTROLLERS DESIGN
PROCEDURE FOR POWER CONVERTERS BY
MEANS OF PARTICLE SWARM
OPTIMIZATION**

AUTHOR: LUCAS CIELO BORIN
ADVISOR: VINÍCIUS FOLETTO MONTAGNER

This master thesis provides as contribution a design procedure, based on particle swarm optimization and Kharitonov's Theorem, to obtain fixed gains for robust controllers for power converters with parametric uncertainties. Firstly, a transfer function model of the converter with uncertain parameters is used and included in a polytopic model. In the sequence, the transfer function of the controller is chosen, and its fixed coefficients are considered as the decision variables of an optimization problem. An objective function is then used to map each vector of controller coefficients to a real positive value, considering the criteria phase margin, crossover frequency, gain margin, overshoot, steady state error and amplitude of the control signal, for the vertices of the polytopic model. A penalty factor based on the Kharitonov's Theorem is included in the objective function. A search space for the coefficients of the controller is systematically defined as a hyper-rectangle, based on the positivity of the coefficients of the characteristic polynomial, and the particle swarm optimization algorithm is then employed to search for the coefficient vector of the controller that minimizes the proposed objective function. To prove the ability to find robust controllers for applications in power electronics, the proposed procedure is used to obtain PID controllers for a Buck converter and PI controllers for a permanent magnet synchronous motor, ensuring good regulation of the output variables in both applications. Experimental results for the permanent magnet synchronous motor confirm the practical viability of the proposed procedure, with superior performance compared to gains obtained by a function specialized for PID design. The proposed procedure can be considered as an alternative to analytical methods and to trial and error designs, allowing to reduce the time demanded in human-machine interaction in order to design controllers that meet multiple objectives and constraints of interest in practice.

Keywords: Robust control, power converters, uncertain parameters, artificial intelligence, particle swarm optimization.

LISTA DE FIGURAS

| | | |
|------------|---|----|
| Figura 1.1 | – Utilização de meta-heurísticas em problemas de otimização aplicados a conversores de potência, em trabalhos publicados entre os anos de 2002 e 2013. | 16 |
| Figura 2.1 | – Classificação dos métodos de otimização. | 22 |
| Figura 2.2 | – Evolução de uma partícula para um espaço de busca de duas dimensões, com base nas equações (2.3) e (2.4). | 25 |
| Figura 2.3 | – Função (2.6) no espaço de busca $0 \leq x \leq 10$ e $0 \leq y \leq 10$: (a) representação tridimensional; (b) representação em curvas de nível, destacando o ponto de mínimo global. | 28 |
| Figura 2.4 | – Valor do <i>fitness</i> da otimização por meio do PSO para a função (2.6). | 29 |
| Figura 2.5 | – Posição das 10 partículas do enxame nas épocas 1, 5, 10, 15, 20 e 25, mostrando a convergência das partículas durante uma execução do PSO. | 29 |
| Figura 2.6 | – Diagrama de blocos de um sistema de controle em malha fechada. ... | 30 |
| Figura 2.7 | – Valor do <i>fitness</i> para a primeira das execuções apresentadas na Tabela 2.4, da otimização por meio do PSO para a função (2.12). | 32 |
| Figura 2.8 | – Simulações comparativas do ganho da primeira coluna na Tabela 2.4 e do controlador projetado por um método baseado na resposta em frequência: (a) diagrama de Bode do sistema em malha aberta; (b) resposta ao degrau unitário do sistema em malha fechada. | 33 |
| Figura 3.1 | – Fluxograma do procedimento de projeto proposto. | 42 |
| Figura 3.2 | – Circuito de um conversor <i>Buck</i> CC-CC. | 44 |
| Figura 3.3 | – Valor do <i>fitness</i> em cada época da otimização por meio do PSO para o projeto do PID para o conversor <i>Buck</i> | 48 |
| Figura 3.4 | – Polos do sistema em malha fechada localizados no semiplano esquerdo considerando uma discretização fina nos valores dos parâmetros incertos do conversor <i>Buck</i> , confirmando a estabilidade do sistema controlado. | 49 |
| Figura 3.5 | – Diagramas de Bode do sistema em malha aberta para uma discretização fina nos valores dos parâmetros incertos do conversor <i>Buck</i> | 50 |
| Figura 3.6 | – Respostas no domínio do tempo, considerando uma discretização fina nos valores dos parâmetros incertos do conversor <i>Buck</i> : (a) respostas ao degrau unitário do sistema em malha fechada; (b) sinais de controle correspondentes. | 50 |
| Figura 3.7 | – Comparação das respostas ao degrau unitário do sistema em malha fechada para o controlador baseado em PSO e para o controlador projetado com a função <i>pidtune</i> do MATLAB. | 51 |
| Figura 4.1 | – PMSM: diagrama de blocos do sistema de controle, composto por três controladores PIs. | 53 |
| Figura 4.2 | – Valor do <i>fitness</i> em cada época da otimização por meio do PSO para o projeto dos PIs para as malhas de controle de corrente direta, $i_d(s)$, de corrente em quadratura, $i_q(s)$ e de velocidade, $\omega_m(s)$, respectivamente. | 59 |
| Figura 4.3 | – Análises da função objetivo para o projeto de controle da malha de velocidade: (a) parte do espaço de busca dado por $0 \leq K_p \leq 15$ e $0 \leq K_i \leq 15$ que apresenta o comportamento da função objetivo em uma região que inclui o ganho (4.15); (b) curvas de nível da função em (a). | 59 |

| | | |
|------------|---|----|
| Figura 4.4 | – Diagramas de Bode em malha aberta com os ganhos (4.15) para a malha de velocidade. | 61 |
| Figura 4.5 | – Resposta ao degrau unitário do sistema em malha fechada, com os ganhos (4.15), para a malha de velocidade. | 62 |
| Figura 4.6 | – PMSM: protótipo utilizado para obtenção dos resultados experimentais. | 65 |
| Figura 4.7 | – Resultados experimentais para rastreamento de referência com os controladores baseados no procedimento proposto e na função <i>pidtune</i> do MATLAB, sob variações de referência de velocidade: (a) respostas da malha de controle de velocidade (superior) e detalhamento na variação de referência (inferior); (b) malhas de controle das corrente no eixo <i>d</i> (superior) e no eixo <i>q</i> (inferior). | 65 |
| Figura 4.8 | – Resultados experimentais para rastreamento de referência com os controladores baseados em PSO e <i>pidtune</i> sob variações de carga: (a) respostas da malha de controle de velocidade (superior) e detalhamento do comportamento na variação de referência (inferior); (b) malhas de controle das corrente no eixo <i>d</i> (superior) e no eixo <i>q</i> (inferior). | 66 |
| Figura 4.9 | – Resposta ao degrau do sistema com os ganhos (4.13)-(4.15), comparando o resultado do modelo médio linear do MATLAB e o resultado experimental, para a malha de velocidade. | 67 |
| Figura A.1 | – Diagrama de Bode do sistema em malha aberta, comparando o controlador PI em (A.4) e o controlador PID em (A.6). | 79 |

LISTA DE TABELAS

| | | |
|------------|---|----|
| Tabela 2.1 | – Configuração do PSO para busca do ponto de mínimo para a função (2.6)..... | 28 |
| Tabela 2.2 | – Resultados das execuções do PSO para a função (2.6). | 28 |
| Tabela 2.3 | – Configuração do PSO, para o projeto do PID para o Exemplo 6.9 de [Bazanella e Silva 2005]. | 31 |
| Tabela 2.4 | – Resultados de 5 execuções do PSO para o projeto do PID para o Exemplo 6.9, e os respectivos valores obtidos para margem de fase e margem de ganho. | 32 |
| Tabela 3.1 | – Parâmetros da planta (conversor <i>Buck</i>). | 45 |
| Tabela 3.2 | – Especificações da função objetivo para o projeto do controlador PID para o conversor <i>Buck</i> | 46 |
| Tabela 3.3 | – Configuração do PSO, para o exemplo do conversor <i>Buck</i> | 47 |
| Tabela 3.4 | – Valores de pior caso obtidos avaliando os vértices do sistema com os ganhos de controle (3.26). | 47 |
| Tabela 4.1 | – Parâmetros da planta (PMSM)..... | 55 |
| Tabela 4.2 | – Especificações da função objetivo para o projeto dos controladores PIs para as malhas elétricas e mecânica do PMSM. | 57 |
| Tabela 4.3 | – Configuração do PSO para o projeto dos três controladores PIs para o PMSM. | 58 |
| Tabela 4.4 | – Índices medidos nos vértices do modelo politópico, para a malha de velocidade. | 61 |
| Tabela 4.5 | – Comparação dos PIs projetados para a malha de velocidade, mostrando os piores casos das medidas obtidas para os vértices do modelo politópico. | 63 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|-------|---|
| ACO | Otimização por colônia de formigas (do inglês, <i>ant colony optimization</i>) |
| CCM | Modo de condução contínua (do inglês, <i>continuous conduction mode</i>) |
| CC-CC | Circuito com tensão/corrente contínua na entrada e na saída |
| DE | Algoritmo por evolução diferencial (do inglês, <i>differential evolution</i>) |
| DSP | Processador digital de sinais (do inglês, <i>digital signal processor</i>) |
| FOC | Estratégia de controle orientada pelo campo (do inglês, <i>field oriented control</i>) |
| GA | Algoritmo genético (do inglês, <i>genetic algorithm</i>) |
| IAE | Integral do erro absoluto (do inglês, <i>integral of absolute error</i>) |
| IGBT | Transistor bipolar de porta isolada (do inglês, <i>insulated gate bipolar transistor</i>) |
| IM | Máquina de indução (do inglês, <i>induction machine</i>) |
| ISC | Integral do valor quadrático do sinal de controle (do inglês, <i>integral squared of control signal</i>) |
| ISE | Integral do erro quadrático (do inglês, <i>integral squared error</i>) |
| ITAE | Integral do tempo multiplicado pelo valor absoluto do erro (do inglês, <i>integral time absolute error</i>) |
| MPPT | Rastreamento do ponto de potência máxima (do inglês, <i>maximum power point tracking</i>) |
| MSE | Erro médio quadrático (do inglês, <i>mean-squared error</i>) |
| PI | Controlador proporcional-integral (do inglês, <i>proportional-integral controller</i>) |
| PID | Controlador proporcional-integral-derivativo (do inglês, <i>proportional-integral-derivative controller</i>) |
| PMSM | Motor síncrono de ímãs permanentes (do inglês, <i>permanent magnet synchronous motor</i>) |
| PSO | Otimização por enxame de partículas (do inglês, <i>particle swarm optimization</i>) |
| PWM | Modulação por largura de pulso (do inglês, <i>pulse width modulation</i>) |
| SMC | Controle por modos deslizantes (do inglês, <i>sliding mode control</i>) |
| UPS | Fonte ininterrupta de energia (do inglês, <i>uninterruptible power supplies</i>) |
| VSI | Inversor fonte de tensão (do inglês, <i>voltage source inverter</i>) |

LISTA DE APÊNDICES

| | |
|---|----|
| Apêndice A – Resolução baseada em formulação analítica para o problema apresentado na Seção 2.5 | 78 |
|---|----|

SUMÁRIO

| | |
|--|----|
| 1 INTRODUÇÃO | 15 |
| 1.1 MOTIVAÇÃO E JUSTIFICATIVA | 15 |
| 1.2 REVISÃO BIBLIOGRÁFICA | 18 |
| 1.3 OBJETIVOS E CONTRIBUIÇÕES | 20 |
| 1.4 ORGANIZAÇÃO DO TRABALHO | 21 |
| 2 OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS | 22 |
| 2.1 PRINCÍPIO DE FUNCIONAMENTO | 23 |
| 2.2 PSEUDO-CÓDIGO | 25 |
| 2.3 PARÂMETROS DE INICIALIZAÇÃO | 26 |
| 2.4 PSO PARA OTIMIZAÇÃO DE UMA FUNÇÃO DE BENCHMARK | 27 |
| 2.5 APLICAÇÃO EM UM PROBLEMA DE PROJETO DE CONTROLADOR | 30 |
| 2.6 CONSIDERAÇÕES FINAIS | 34 |
| 3 PROCEDIMENTO DE PROJETO PROPOSTO | 35 |
| 3.1 MODELO DA PLANTA COM INCERTEZAS E DO CONTROLADOR ROBUSTO | 35 |
| 3.2 CRITÉRIOS DE DESEMPENHO UTILIZADOS | 36 |
| 3.3 ESTABILIDADE ROBUSTA BASEADA NO TEOREMA DE KHARITONOV | 37 |
| 3.4 DEFINIÇÃO DO PROBLEMA DE PROJETO DO CONTROLADOR | 38 |
| 3.5 FUNÇÃO OBJETIVO E ESPAÇO DE BUSCA | 39 |
| 3.6 SUMARIZAÇÃO DO PROCEDIMENTO DE PROJETO PROPOSTO | 42 |
| 3.7 EXEMPLO PARA O CONTROLE DE TENSÃO DE UM CONVERSOR BUCK | 43 |
| 3.8 CONSIDERAÇÕES FINAIS | 51 |
| 4 VALIDAÇÃO EXPERIMENTAL DO PROCEDIMENTO | 53 |
| 4.1 INFORMAÇÕES PRELIMINARES | 53 |
| 4.2 APLICAÇÃO DO PROCEDIMENTO DE PROJETO PROPOSTO AO PMSM | 56 |
| 4.3 ANÁLISES COMPARATIVAS | 62 |
| 4.4 RESULTADOS EXPERIMENTAIS | 64 |
| 4.5 CONSIDERAÇÕES FINAIS | 67 |
| 5 CONCLUSÕES | 68 |
| 5.1 PERSPECTIVAS | 69 |
| 5.2 PUBLICAÇÕES | 69 |
| REFERÊNCIAS | 71 |
| APÊNDICES | 77 |

1 INTRODUÇÃO

1.1 MOTIVAÇÃO E JUSTIFICATIVA

O termo otimização faz referência a problemas em que se procura minimizar ou maximizar uma função por meio da escolha dos melhores valores de suas variáveis dentro de um conjunto de busca. Isso pode representar o processo de encontrar a melhor maneira de utilizar os recursos disponíveis, respeitando limites sobre estes recursos [Dréo et al. 2006, León-Aldaco, Calleja e Alquicira 2015]. Em engenharia, técnicas matemáticas de otimização podem ser aplicadas para maximizar ou minimizar um índice de desempenho e prover a melhor solução para o problema [Rao 2009].

Um problema de otimização matemática pode ser definido pela busca de valores mínimos ou máximos de uma função objetivo por meio da escolha dos valores de suas variáveis de decisão em um conjunto dado, denominado espaço de busca [Rao 2009]. Em termos técnicos, um problema de otimização pode ser definido por¹

$$\min f(\mathbf{x}) \quad s.a \quad \begin{cases} \mathbf{x} \in \mathcal{X} \\ \mathbf{g}(\mathbf{x}) \leq 0 \\ \mathbf{h}(\mathbf{x}) = 0 \end{cases} \quad (1.1)$$

em que \mathbf{x} é um vetor de variáveis de decisão, \mathcal{X} define o espaço de busca, $\mathbf{g}(\mathbf{x})$ representa um vetor de restrições de desigualdades, $\mathbf{h}(\mathbf{x})$ representa um vetor de restrições de igualdades, e o valor que minimiza $f(\mathbf{x})$ é denominado mínimo global. Por exemplo, em Talbi (2009), o mínimo global \mathbf{x}^* é definido pelo conjunto de variáveis de decisão que possui o melhor valor da função objetivo entre todas as soluções pertencentes a um espaço de busca, i.e., $\forall \mathbf{x} \in \mathcal{X}, f(\mathbf{x}^*) \leq f(\mathbf{x})$.

Segundo Talbi (2009), não existe um método com a capacidade de solucionar todos os problemas de otimização de forma eficiente. Ao longo dos anos, vários métodos de otimização foram desenvolvidos para resolver diferentes tipos de problemas de otimização, e estas abordagens podem ser divididas em métodos de otimização clássicos e métodos de otimização modernos, ou meta-heurísticas.

Os métodos clássicos, como programação linear (e.g., algoritmo simplex) e não-linear (e.g., método de Newton-Raphson, método do gradiente), são abordagens que já foram utilizadas para resolver muitos problemas em engenharia, conforme [Rao 2009, León-Aldaco, Calleja e Alquicira 2015]. No entanto, métodos clássicos apresentam limitações

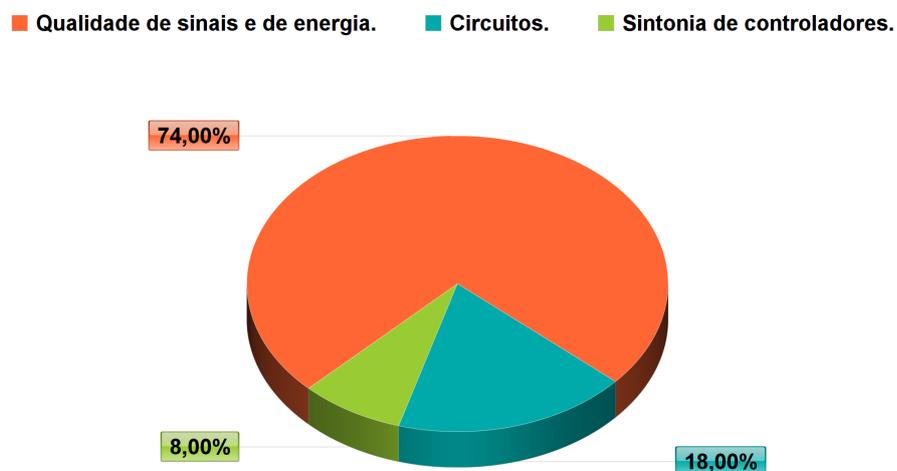
¹Considere a partir deste ponto, para manter a generalidade do problema de otimização, que a minimização de uma função objetivo $f(\mathbf{x})$ é equivalente a maximização do valor da função objetivo $-f(\mathbf{x})$.

para serem aplicados em problemas que envolvem variáveis ou funções que podem não ser contínuas ou diferenciáveis, e também em casos em que há grande dificuldade de expressar o problema analiticamente (e.g., problemas de grande complexidade, com grande número de variáveis, etc.) [Rao 2009, Deb 2001].

Neste contexto, tornam-se interessantes as abordagens meta-heurísticas, que podem ser definidas como procedimentos de alto nível capazes de fornecer soluções suficientemente boas para um problema de otimização. Meta-heurísticas são capazes de tratar problemas que, por exemplo, podem ter função objetivo ou restrições difíceis de serem expressas de forma analítica, sem a necessidade de conhecer a derivada da função objetivo, podendo evoluir com base em simulações numéricas ou em dados experimentais [Haupt e Haupt 2004]. Exemplos de algoritmos muito utilizados na literatura são o algoritmo genético (do inglês, *genetic algorithm* – GA), algoritmo por evolução diferencial (do inglês, *differential evolution* – DE), algoritmo de otimização por colônia de formigas (do inglês, *ant colony optimization* – ACO) e o algoritmo de otimização por enxame de partículas (do inglês, *particle swarm optimization* – PSO) [Serapião 2009].

No contexto de eletrônica de potência, ainda é baixo o número de trabalhos de meta-heurísticas aplicadas na sintonia de controladores. Em [León-Aldaco, Calleja e Alquicira 2015], é apresentado um levantamento considerando trabalhos publicados entre os anos de 2002 e 2013 que utilizam meta-heurísticas aplicadas em otimizações envolvendo conversores de potência. O resultado deste estudo pode ser resumido na Figura 1.1, em que se pode observar a baixa proporção de utilização de meta-heurísticas em problemas de sintonia de controladores em eletrônica de potência, totalizando apenas 8% dos trabalhos publicados.

Figura 1.1 – Utilização de meta-heurísticas em problemas de otimização aplicados a conversores de potência, em trabalhos publicados entre os anos de 2002 e 2013.



Fonte: Adaptado de [León-Aldaco, Calleja e Alquicira 2015].

Estes dados servem de motivação para a realização de estudos mais detalhados de

aplicação de meta-heurísticas para sintonia de controladores para conversores de potência. Neste sentido, dentre as meta-heurísticas existentes, merece destaque o algoritmo PSO, devido à sua possibilidade de aplicação em problemas que podem ser de difícil resolução por meio da utilização de algoritmos clássicos [Eberhart e Kennedy 1995, Sengupta, Basak e Peters 2018, Poli, Kennedy e Blackwell 2007]. O PSO é um algoritmo inspirado em comportamentos coletivos observados na natureza e, em comparação com outras meta-heurísticas, é um algoritmo simples computacionalmente, que exige poucos parâmetros de inicialização, que utiliza apenas duas equações em sua evolução, o que reduz seu tempo de execução, e que pode evitar confinamentos em mínimos locais, na busca pelo ótimo global [Eberhart e Kennedy 1995, Flasiński 2016]. Entretanto, o algoritmo PSO foi pouco utilizado em sintonia de controladores para conversores de potência. Por exemplo, uma pesquisa na base de dados do IEEEExplore, entre os anos 2016 e 2019 aponta 145 artigos com resumo contendo os termos 'enxame de partículas' e 'conversores', sendo apenas 7% destes efetivamente focados em sintonia de controladores, indicando uma lacuna a explorar na literatura.²

Adicionalmente, dentre os trabalhos que utilizam o algoritmo PSO para a sintonia de controladores, muito poucos apresentam um certificado teórico de robustez frente a incertezas paramétricas em conversores. Controladores robustos são reconhecidos como uma alternativa adequada para tratar sistemas afetados por incertezas, como é o caso de conversores de potência cujos parâmetros físicos não são precisamente conhecidos. Neste contexto, o Teorema de Kharitonov fornece um método computacionalmente rápido de certificar teoricamente a robustez de um sistema sujeito a incertezas paramétricas intervalares [Bhattacharyya 2017, Bhattacharyya e Keel 1995, Shin et al. 2003, Bevrani et al. 2010, Kumar e Shreesha 2016, Nazari et al. 2018]. O Teorema de Kharitonov provê um certificado de estabilidade robusta, analisando apenas as raízes de quatro polinômios, que podem ser obtidos a partir dos valores extremos dos coeficientes do polinômio característico do sistema em malha fechada [Bhattacharyya e Keel 1995]. Entretanto, este teorema ainda não foi utilizado de forma extensa na literatura de eletrônica de potência.

Dado o baixo número de trabalhos que utilizam o algoritmo PSO em sintonia de controladores, e também o pouco frequente emprego do Teorema de Kharitonov para análise de estabilidade robusta em conversores, conforme comprovado pela revisão bibliográfica a seguir, justifica-se explorar a combinação destas técnicas para projetar controladores robustos para conversores, tendo como potencial vantagem do uso do PSO a busca eficiente de ganhos de controle visando atender a múltiplos critérios de desempenho de interesse prático, e como potencial vantagem do uso do Teorema de Kharitonov, a certificação teórica de estabilidade sob incertezas paramétricas no conversor.

²Pesquisa realizada em 10 de novembro de 2019, no site: <https://ieeexplore.ieee.org/search/advanced>, buscando em *abstracts* pelos seguintes termos: "particle swarm" e "converter".

1.2 REVISÃO BIBLIOGRÁFICA

Nesta subseção, apresenta-se uma revisão da literatura sobre a utilização do algoritmo PSO e do Teorema de Kharitonov na sintonia de controladores em aplicações de eletrônica de potência.

Considerando a sintonia de controladores do tipo proporcional-integral (PI), o algoritmo PSO já foi utilizado para obter ganhos para o controle de turbinas eólicas em [Wu et al. 2007], motores de indução em [Sebtahmadi et al. 2017], motores síncronos de ímãs permanentes em [Wang, Ufnalski e Grzesiak 2016], compensadores síncronos estáticos em [Liu e Hsu 2010], conversores *Cúk* em [Thirumeni e Thangavelusamy 2019] e microrredes em [Chung et al. 2010] e [Bevrani et al. 2012]. Da mesma forma, o PSO já foi utilizado para a sintonia de controladores proporcional-integral-derivativo (PID) para reguladores automáticos de tensão em [Gaing 2004], motores CC em [Qi, Shi e Zhang 2019], e para a direção assistida de veículos elétricos em [Hanifah et al. 2018].

É possível notar na literatura uma grande prevalência do emprego do algoritmo PSO para a sintonia de controladores PIs e PIDs, devido ao grande uso deste tipo de controle na indústria [Åström e Hägglund 2006]. Estes controladores também são mais simples de serem sintonizados em relação a controladores de ordem mais elevada. Um ponto interessante nestes trabalhos é que diferentes funções objetivo são utilizadas nos projetos dos ganhos para os PIs e PIDs.

Por exemplo, em [Hanifah et al. 2018], [Liu e Hsu 2010] e [Thirumeni e Thangavelusamy 2019] utilizam-se funções conhecidas na literatura, dadas pelo erro médio quadrático (do inglês, *mean-squared error* – MSE) e pela integral do erro absoluto (do inglês, *integral of absolute error* – IAE). Diferentemente, em [Bevrani et al. 2012] e [Qi, Shi e Zhang 2019], são utilizadas funções objetivo particulares, que consideram critérios pertencentes ao domínio do tempo, como tempo de subida e tempo de acomodação da resposta ao degrau, por exemplo, enquanto em [Wu et al. 2007] é utilizada uma função com critérios pertencentes ao domínio da frequência, como, por exemplo, margem de ganho e margem de fase. Por fim, em [Chung et al. 2010], é utilizado uma função descontínua, que considera critérios pertencentes ao domínio da frequência, com restrições incorporadas como penalidades na função objetivo. Nestes artigos, ao observar os resultados apresentados, pode-se concluir que mais de um tipo de função objetivo permite obter um projeto de controlador viável. Questões como compromisso entre eficiência computacional no projeto do controlador e qualidade dos resultados devem ser investigadas para orientar a escolha da função objetivo.

Outra questão importante dos trabalhos supracitados é que apenas uma pequena parcela informa como é especificado o espaço de busca dos controladores. Destes, apenas em [Sebtahmadi et al. 2017], [Wang, Ufnalski e Grzesiak 2016] e [Gaing 2004] são informados os limites do espaço de busca, porém, sem nenhuma justificativa sobre a escolha dos

limites. Isso ocorre devido à inicialização do espaço de busca, por muitas vezes, ser feita de forma heurística. Outro ponto que se pode observar destes trabalhos é que não há uma indicação de escolha do número de partículas ou do número de épocas para a execução do PSO nos problemas de projeto de controladores. Mesmo que alguns trabalhos indiquem valores para a inicialização das partículas e épocas, como feito nos exemplos numéricos em [Eberhart e Shi 2001, Shi e Eberhart 1998], esta escolha depende da complexidade e da estrutura do problema em casos práticos de aplicação.

Diferentemente dos artigos apresentados acima para PIs e PIDs, existem trabalhos que utilizam o PSO para otimização da sintonia de controladores LQR, aliviando o engenheiro da tarefa de escolher de forma heurística as matrizes de ponderação \mathbf{Q} e \mathbf{R} . Em [Ufnalski, Kaszewski e Grzesiak 2015], um LQR sintonizado pelo PSO é utilizado para o controle de um inversor fonte de tensão com filtro LC, em que a função objetivo considera o erro de rastreamento e um limite para o sinal de controle com a adição de penalizações. Em [Murari et al. 2019], um LQR é utilizado para o controle da potência entregue à rede por um gerador de indução duplamente alimentado, em que a função objetivo é a integral do tempo multiplicado pelo valor absoluto do erro (do inglês, *integral time absolute error* – ITAE). Em [Borin et al. 2019], utiliza-se o PSO para a sintonia de um LQR discreto para o controle de tensão de uma fonte ininterrupta de energia (do inglês, *uninterruptible power supplies* – UPS), em que a função objetivo considera a integral do erro quadrático (do inglês, *integral squared error* – ISE) e a integral do valor quadrático do sinal de controle (do inglês, *integral squared of control signal* – ISC).

O algoritmo PSO também já foi utilizado para a sintonia de controladores tipo III em [Banerjee, Ghosh e Rana 2017], para um conversor *Boost*, em que a função objetivo utilizada é baseada no critério ITAE, e em [Dell’Aquila e Lecci 2008] e [Zobaa e Lecci 2011], em que o PSO é utilizado na sintonia de controladores ressonantes a fim de reduzir o erro de rastreamento na corrente de entrada de conversores de potência, em que uma função objetivo particular que considera critérios pertencentes ao domínio do tempo é proposta. O PSO também já foi utilizado em controladores repetitivos, em [Ufnalski e Grzesiak 2016] para o controle de um inversor fonte de tensão, e em [Stras, Ufnalski e Grzesiak 2018], no controle de um conversor conectado à rede.

Em [Teja, Shanavas e Patnaik 2012], o PSO é utilizado para encontrar os parâmetros de um controle por modos deslizantes (do inglês, *sliding mode control* – SMC) aplicado em um conversor *Buck*, em que a função objetivo utiliza o critério ISE. O PSO também já foi utilizado para alocação de polos em [Yousefi et al. 2008], aplicado em um conversor *Cúk* CC–CC. Em [Galcecki et al. 2018], o PSO foi utilizado para encontrar ganhos por realimentação de estados para um conversor trifásico conectado à rede, em que a função objetivo utilizada é baseada no critério ISE. Por fim, o PSO já foi utilizado na otimização de parâmetros da função de rastreamento do ponto de potência máxima (do inglês, *maximum power point tracking* – MPPT), para sistemas fotovoltaicos, em [Ishaque

et al. 2012, Renaudineau et al. 2015, Silva et al. 2016].

Pode-se notar, nos artigos citados acima, que o PSO não foi utilizado para sintonia de controladores incluindo um certificado teórico de robustez frente a incertezas paramétricas. Especificamente, não se encontram trabalhos para o caso de uso conjunto do PSO com alguma condição capaz de prover, no estágio de projeto de controladores, um certificado teórico de estabilidade robusta frente a incertezas nos parâmetros físicos do conversor. Neste sentido, o Teorema de Kharitonov mostra-se importante [Bhattacharyya 2017]. Quando aplicado ao polinômio característico do sistema em malha fechada, este teorema é capaz de garantir, matematicamente, a estabilidade para todo um domínio de incertezas paramétricas.

O Teorema de Kharitonov já foi empregado para auxiliar o projeto de controladores PIs para um motor de indução em [Shin et al. 2003] e para um conversor *Buck* quadrático em [Bevrani et al. 2010], ambas as aplicações com parâmetros incertos. Em [Pandey et al. 2018], o Teorema de Kharitonov foi utilizado para certificar a estabilidade para um SMC aplicado em um conversor *Boost* CC-CC. O Teorema de Kharitonov também foi utilizado no projeto de controladores PIDs robustos para um reator químico em [Kumar e Shreesha 2016] e na obtenção de parâmetros de um controlador *droop* para estabilidade robusta de microrredes em [Nazari et al. 2018]. Um ponto comum nestes artigos é que todos exploram o Teorema de Kharitonov para a obtenção de uma região de estabilidade, em uma etapa prévia ao projeto dos controladores. Por fim, cabe mencionar que existem trabalhos que já utilizaram o Teorema de Kharitonov combinado com meta-heurísticas, como é o caso de [Elsisi 2019], em que o Teorema de Kharitonov é utilizado junto com um algoritmo genético para a obtenção de um controlador PID, em que os ganhos de controle podem pertencer a uma faixa de valores incertos, abordando o problema da não fragilidade do controlador.

Desta forma, a revisão bibliográfica acima confirma uma lacuna na literatura de trabalhos utilizando o PSO combinado com o Teorema de Kharitonov para prover controladores robustos aplicáveis na prática a conversores de potência com parâmetros incertos, motivando a investigação nesta dissertação.

1.3 OBJETIVOS E CONTRIBUIÇÕES

O objetivo geral desta dissertação é propor um procedimento de projeto de controladores robustos baseado na combinação do algoritmo PSO e do Teorema de Kharitonov para aplicação em conversores de potência.

Os objetivos específicos são:

- fornecer um procedimento para a obtenção de funções de transferências de controladores baseado: i) em um modelo da planta incluindo incertezas paramétricas, ii)

em uma função objetivo que leva em consideração um conjunto de especificações de ordem prática no domínio do tempo e da frequência, iii) em um espaço de busca dos ganhos definido sistematicamente e iv) na execução do Teorema de Kharitonov para certificar teoricamente a estabilidade robusta do sistema em malha fechada;

- validar, por meio de simulações e experimentalmente, o procedimento proposto, a partir de estudos de caso em conversores com parâmetros incertos, comparando a qualidade dos resultados obtidos com o procedimento proposto com os resultados obtidos a partir de métodos similares na literatura.

1.4 ORGANIZAÇÃO DO TRABALHO

Além do Capítulo 1, que contém a motivação, os objetivos e a revisão bibliográfica, esta dissertação esta organizada como segue.

No Capítulo 2, é apresentado o algoritmo por enxame de partículas, detalhando seu funcionamento, pseudo-código, parâmetros de inicialização e exemplos de utilização.

No Capítulo 3, é apresentado o procedimento de projeto proposto, baseado no algoritmo PSO e no Teorema de Kharitonov, detalhando a definição da função objetivo e do espaço de busca, sendo este procedimento a principal contribuição deste trabalho. Também é apresentado neste capítulo um estudo de caso com o projeto de um controlador PID para o controle de tensão de um conversor *Buck*.

No Capítulo 4, é apresentada uma validação experimental do procedimento para um motor síncrono de ímãs permanentes, em que três controladores PIs são projetados de forma independente para o controle de corrente nos eixos direto e quadratura e de velocidade do motor. Resultados de simulação e experimentais, bem como comparações com controladores sintonizados por outros métodos da literatura, são mostrados, ilustrando a superioridade dos controladores robustos obtidos por meio do procedimento proposto.

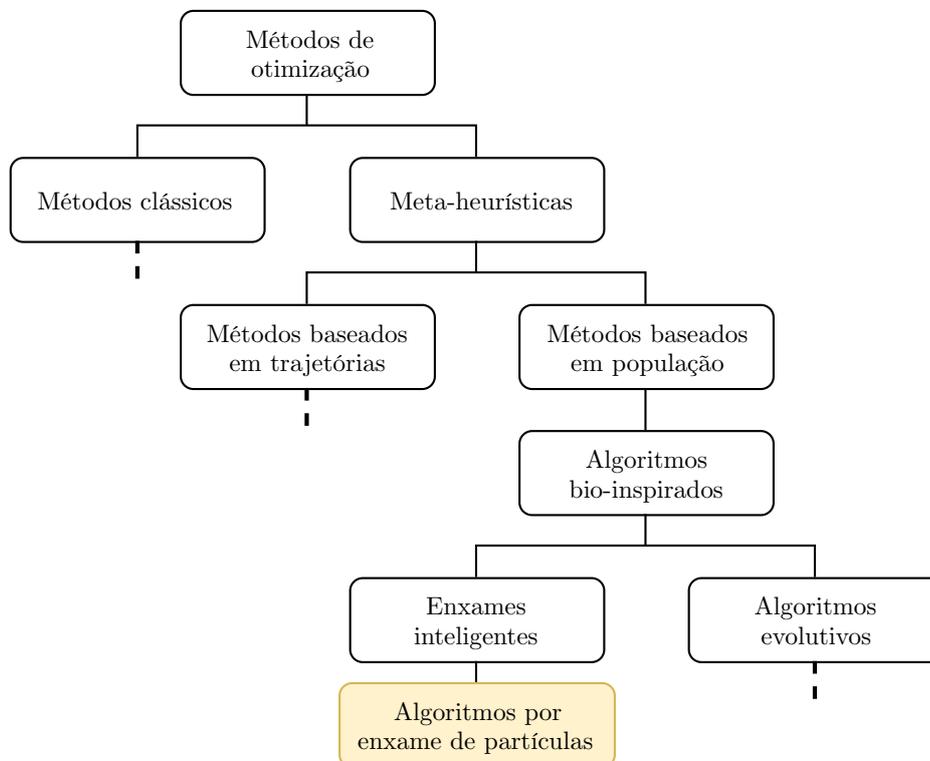
Finalmente, o Capítulo 5 apresenta as conclusões e perspectivas deste trabalho.

2 OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS

Um problema comum em controle aplicado a conversores de potência é a necessidade de levar em consideração diversas especificações de desempenho que, em muitos casos, podem ser conflitantes entre si, como, por exemplo, rápida resposta dinâmica e limitação do sinal de controle. Em um problema de projeto, os coeficientes do controlador podem representar as variáveis de decisão de uma otimização matemática. Neste contexto, a expressão analítica de uma função objetivo e de restrições apropriadas em termos dos coeficientes do controlador pode se tornar um desafio complexo matematicamente [Miranda, Takahashi e Jota 2007]. Uma alternativa para tratar problemas de otimização sem depender da descrição analítica da função objetivo e das restrições é utilizando simulações do conversor em programas especializados (e.g., MATLAB), juntamente com meta-heurísticas [Talbi 2009].

Dentre as meta-heurísticas, destaca-se aqui o algoritmo de otimização por enxame de partículas, proposto por [Eberhart e Kennedy 1995], um algoritmo bio-inspirado pertencente à classe de enxames inteligentes, como pode ser visto na Figura 2.1.

Figura 2.1 – Classificação dos métodos de otimização.



Fonte: Adaptado de [León-Aldaco, Calleja e Alquicira 2015].

O PSO é capaz de fornecer soluções suficientemente boas para um problema de otimização ao evitar confinamento em mínimos locais, em um espaço de busca que pode

ser muito grande para ser investigado por meio de discretização exaustiva [Bianchi et al. 2009, Talbi 2009]. Também, o PSO se destaca entre as meta-heurísticas pela sua simplicidade, em comparação com os algoritmos com operadores genéticos (e.g., evolução diferencial e algoritmos genéticos), e, devido a isso, possui menor tempo de computação e necessita de menos memória para ser executado [Yoshida et al. 2000, Flasiński 2016, Sengupta, Basak e Peters 2018, Poli, Kennedy e Blackwell 2007, Haupt e Haupt 2004].

Como mencionado no Capítulo 1, no contexto de eletrônica de potência, este algoritmo já foi utilizado com sucesso em problemas relacionadas à qualidade de energia e na otimização de componentes de circuitos. Porém, a utilização do PSO em sintonia de controladores ainda é pequena [León-Aldaco, Calleja e Alquicira 2015, Poli, Kennedy e Blackwell 2007].

2.1 PRINCÍPIO DE FUNCIONAMENTO

PSO é uma meta-heurística capaz de investigar grandes espaços de busca com soluções candidatas, sem a necessidade de hipóteses sobre este espaço de busca. Diferentemente dos métodos clássicos de otimização, como o método do gradiente e o método de Newton-Raphson, por exemplo, o PSO não utiliza o gradiente da função objetivo, i.e., não tem a necessidade de calcular a derivada da função objetivo. Este algoritmo distribui um conjunto de agentes (partículas) para encontrar uma solução ótima em um espaço de busca pré-definido em que, a cada nova iteração, a posição de cada partícula evolui, baseada na melhor posição individual e na melhor posição do conjunto até aquela iteração, inspirado em padrões coletivos observados na natureza (e.g., bando de pássaros, cardume de peixes, etc.) [Eberhart e Kennedy 1995].

Algumas definições importantes sobre o algoritmo PSO, desenvolvido por Eberhart & Kennedy (1995), são dadas a seguir:

- partícula - vetor de variáveis de decisão, que representa uma possível solução do problema de otimização;
- enxame - conjunto de partículas;
- época - iteração do algoritmo PSO;
- espaço de busca - conjunto em que são posicionadas as partículas, em geral formado pelo produto cartesiano dos intervalos das variáveis de decisão;
- função objetivo - função que relaciona pontos do espaço de busca com uma medida da sua qualidade;
- *fitness* - valor da função objetivo associado à posição de uma partícula.

No PSO, cada partícula i é associada com uma posição no espaço de busca, e pode ser definida pelo vetor linha dado por

$$\mathbf{s}_i = [s_{i,1} \ s_{i,2} \ s_{i,3} \ \cdots \ s_{i,\mathcal{D}}] \quad , \quad i = 1 \dots N \quad (2.1)$$

em que \mathcal{D} é o número de dimensões do espaço de busca e N é o tamanho do enxame.

Com o vetor \mathbf{s}_i , calcula-se o valor da função objetivo para cada partícula (*fitness*). As partículas então deslocam-se pelo espaço de busca a cada iteração do algoritmo (época k), da posição $\mathbf{s}_i(k)$ para a posição $\mathbf{s}_i(k+1)$, utilizando uma equação de deslocamento, baseada em uma equação recursiva de velocidade.

Da mesma forma que a posição, a velocidade de cada partícula também pode ser definida por um vetor linha dado por [Veerachary e Saxena 2015]

$$\mathbf{v}_i = [v_{i,1} \ v_{i,2} \ v_{i,3} \ \cdots \ v_{i,\mathcal{D}}] \quad , \quad i = 1 \dots N \quad (2.2)$$

A melhor posição que uma partícula i obteve é denominada de $\mathbf{P}_{i.best}$, e a melhor posição obtida pelo enxame, até a época atual k , é denominada de \mathbf{G}_{best} .

As equações que definem a atualização da posição e da velocidade de cada partícula podem ser descritas, respectivamente, por

$$\mathbf{s}_i(k+1) = \mathbf{s}_i(k) + \mathbf{v}_i(k+1) \quad (2.3)$$

$$\mathbf{v}_i(k+1) = \lambda \mathbf{v}_i(k) + \phi_1 \text{rand}_1(\mathbf{P}_{i.best} - \mathbf{s}_i(k)) + \phi_2 \text{rand}_2(\mathbf{G}_{best} - \mathbf{s}_i(k)) \quad (2.4)$$

sendo ϕ_1 e ϕ_2 chamados de coeficientes cognitivo e social, respectivamente, rand_1 e rand_2 valores escalares aleatórios obtidos por meio de uma distribuição uniforme aleatória no intervalo real de 0 a 1, utilizados a fim de criar um pequeno comportamento aleatório nas partículas, e λ o fator de inércia, utilizado para desacelerar as partículas no decorrer das épocas, para uma melhor convergência do algoritmo.

Note que a equação (2.4) possui três parcelas que influenciam o deslocamento da partícula. A primeira parcela é referente à inércia da partícula, calculada a partir da velocidade da partícula na época atual multiplicada pelo fator de inércia λ . Este fator é calculado aqui de forma linear e decrescente com o aumento de k , dado por

$$\lambda = \lambda_{max} - \left(\frac{\lambda_{max} - \lambda_{min}}{M} \right) k \quad (2.5)$$

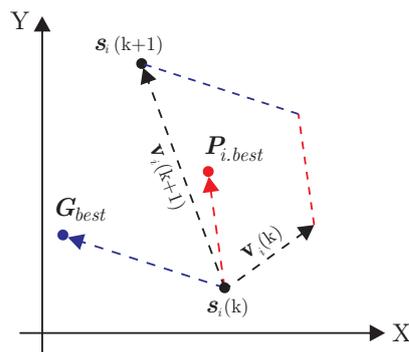
em que k é o valor da época atual, M é o número total de épocas e λ_{max} e λ_{min} são valores máximos e mínimos definidos para o fator de inércia, escolhidos na função *particleswarm* como 0,9 e 0,4, respectivamente.

A segunda parcela é referente à experiência cognitiva da partícula, e é calculada levando em consideração a diferença entre a melhor posição no espaço de busca que

esta partícula já obteve e a sua posição atual. Por fim, a terceira parcela é referente à experiência social do enxame, e é calculada levando em consideração a diferença entre a melhor posição no espaço de busca obtida pelo enxame até a época atual e a posição atual da partícula.

A evolução de uma partícula de uma época para a seguinte, com base em (2.3) e (2.4), é ilustrada na Figura 2.2.

Figura 2.2 – Evolução de uma partícula para um espaço de busca de duas dimensões, com base nas equações (2.3) e (2.4).



Fonte: Adaptado de [Yoshida et al. 2000].

O algoritmo encerra ao atingir um critério de parada, dado por um número máximo de épocas M , um tempo máximo de execução ou pela estagnação do valor de G_{best} , i.e., quando não há melhora do valor de G_{best} por um número pré-determinado de épocas.

2.2 PSEUDO-CÓDIGO

Algoritmo 1: PSO

Entrada: Função objetivo, espaço de busca, N , M , ϕ_1 , ϕ_2

Saída: Melhor ponto do espaço de busca

- 1 **Início**
 - 2 Distribuição das partículas no espaço de busca de forma aleatória
 - 3 **Enquanto** o critério de parada não for atingido, **faça**
 - 4 Cálculo da função objetivo para cada partícula
 - 5 Atualização de $P_{i.best}$ e G_{best}
 - 6 Cálculo e atualização da velocidade $v_i(k+1)$ de cada partícula
 - 7 Cálculo e atualização da posição $s_i(k+1)$ de cada partícula
 - 8 Incremento do número de épocas k
 - 9 **Fim enquanto**
 - 10 **Retorna** G_{best}
 - 11 **Fim**
-

2.3 PARÂMETROS DE INICIALIZAÇÃO

Para a execução do PSO (por exemplo, por meio da função *particleswarm* do MATLAB) é necessário informar os parâmetros descritos a seguir.

- **Função objetivo:** esta função pode ser baseada em uma relação matemática entre as variáveis de decisão (e.g., $f(x, y) = x^2 + 3x + 4y + 5$) ou pode ter seu valor obtido baseando-se em simulações computacionais e resultados experimentais (e.g., os valores de saída das funções *margin* ou *step*, do MATLAB). Portanto, é uma função que relaciona um vetor de variáveis de decisão com um valor. Esta função pode ser definida para contemplar critérios usuais em controle (e.g., ITAE, IAE, ISE, ITSE) ou como uma função com métricas dedicadas a um problema específico.
- **Espaço de busca:** é definido por uma região limitada, que restringe a busca do PSO, por exemplo, em um retângulo, para um espaço de duas dimensões, em um paralelepípedo, para um espaço com três dimensões, ou em um hiper-retângulo, para um espaço com mais de três dimensões. Para definir o espaço de busca, deve-se informar os limites inferiores e superiores para todas as variáveis de decisão (e.g., $0 \leq x \leq 1$, $-10 \leq y \leq 10$). Definir um espaço suficientemente grande aumenta as chances de encontrar um vetor de variáveis de decisão associado a um menor valor da função objetivo, ao custo de aumentar potencialmente o número de partículas e de épocas necessárias para a convergência e, conseqüentemente, o tempo de execução do algoritmo. Definir um espaço muito restrito pode auxiliar o PSO a fornecer uma solução rápida, ao custo de uma potencial menor qualidade do resultado. Em um problema de controle, se o projetista tiver conhecimento prévio dos limites dos ganhos do controlador, por exemplo, então o espaço de busca pode ser reduzido, sem prejuízo da qualidade da solução.
- **Número de partículas ou tamanho do enxame (N):** define a quantidade de pontos do espaço de busca que serão investigados simultaneamente em cada época. Caso o número de partículas escolhido seja muito pequeno em relação ao tamanho do espaço de busca, algumas regiões do espaço podem nunca ser investigadas. Em contrapartida, um número excessivo de partículas tende a aumentar de forma desnecessária o tempo de execução do PSO. Segundo Eberhart e Shi (2001), o tamanho do enxame é dependente do problema, e utilizar entre 20 e 50 é uma escolha comum. Ainda assim, o PSO em geral é vantajoso, por necessitar de um menor número de avaliações ($N \times M$) em comparação com outros algoritmos evolutivos [Eberhart e Shi 2001].

- **Número de épocas (M):** é o número de iterações para uma execução completa do PSO. A opção por poucas épocas pode ser insuficiente para a convergência do algoritmo. Segundo Shi & Eberhart (1998), para casos em que um número adequado de épocas é desconhecido (geralmente este é obtido empiricamente pela repetição da execução do PSO), pode-se optar por um grande número de épocas (e.g., 4000 épocas), e escolher um critério de parada por estagnação do valor de G_{best} (e.g., 20 épocas sem variação significativa do valor de G_{best}) [Shi e Eberhart 1998].
- **Coefficientes cognitivo e social (ϕ_1 e ϕ_2):** são fatores que influenciam o deslocamento de cada partícula de uma época para a seguinte. Enquanto o coeficiente cognitivo, ϕ_1 , influencia uma partícula a deslocar-se em direção ao ponto de melhor *fitness* que ela mesma já obteve, o coeficiente social, ϕ_2 , influencia a partícula a deslocar-se em direção ao melhor ponto do espaço encontrado pelo enxame até a época atual. A escolha por valores equilibrados para ambos coeficientes ajuda a manter o conceito de enxame e a liberdade das partículas na exploração do espaço de busca. A soma destes coeficientes também não deve ser maior que 4 [Eberhart e Kennedy 1995].

2.4 PSO PARA OTIMIZAÇÃO DE UMA FUNÇÃO DE BENCHMARK

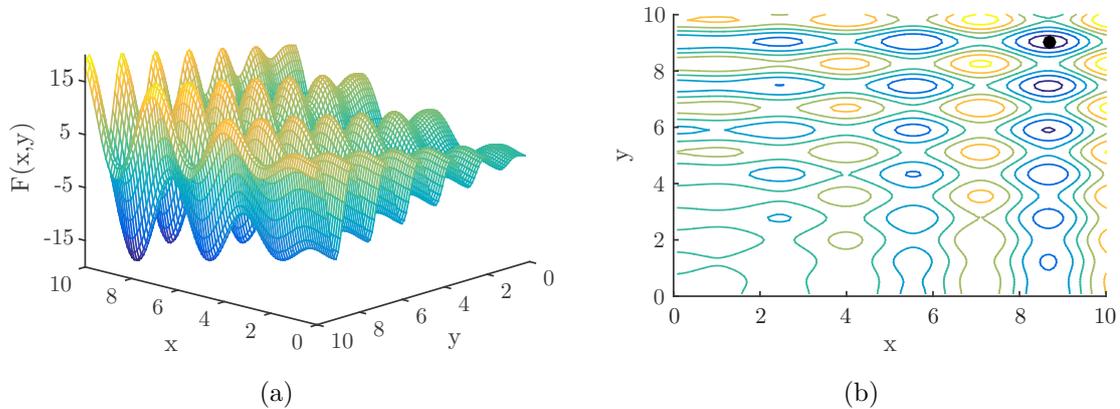
Esta seção apresenta a utilização do PSO para a otimização da função de *benchmark*, constituída por uma soma de senos multiplicada por fatores lineares, apresentada no Capítulo 1 de Haupt & Haupt (2004), descrita como

$$F(x, y) = x \sin(4x) + 1,1 y \sin(2y) \quad (2.6)$$

Funções de *benchmark* como esta são utilizadas por algoritmos de busca e otimização para avaliar características como taxa de convergência, desempenho e precisão dos resultados [Gray 1993].

A Figura 2.3(a) apresenta a superfície tridimensional da função (2.6), para o espaço de busca dado por $0 \leq x \leq 10$ e $0 \leq y \leq 10$. As curvas de nível da função da Figura 2.3(a) são vistas na Figura 2.3(b), em que o mínimo global tem o valor $-18,5547$, no ponto $x = 9,039$ e $y = 8,668$.

Figura 2.3 – Função (2.6) no espaço de busca $0 \leq x \leq 10$ e $0 \leq y \leq 10$: (a) representação tridimensional; (b) representação em curvas de nível, destacando o ponto de mínimo global.



Fonte: Adaptado de [Haupt e Haupt 2004].

As configurações utilizadas para o PSO obter o ponto x e y que minimiza a função (2.6) são dadas na Tabela 2.1. O algoritmo PSO foi executado utilizando a função *particleswarm*, do MATLAB, baseada no algoritmo proposto em [Eberhart e Kennedy 1995] e nas modificações propostas em [Mezura-Montes e Coello 2011, Pedersen 2010].

Tabela 2.1 – Configuração do PSO para busca do ponto de mínimo para a função (2.6).

| Parâmetros do PSO | |
|-------------------------------------|-----|
| Número de partículas (N) | 10 |
| Número de épocas (M) | 25 |
| Coefficiente cognitivo (ϕ_1) | 0,5 |
| Coefficiente social (ϕ_2) | 0,5 |

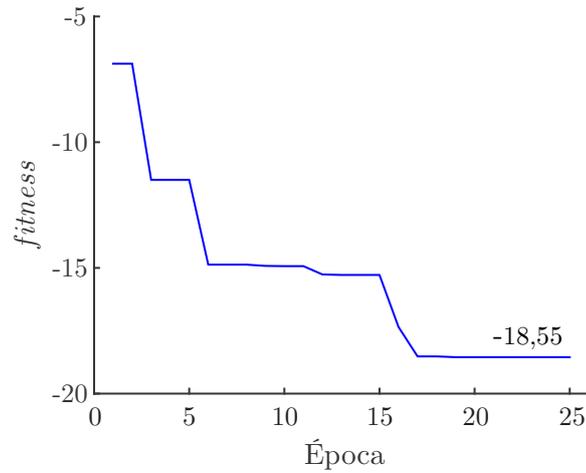
Com esta configuração, o algoritmo PSO foi executado 5 vezes, com o tempo de simulação médio de aproximadamente 3 s, levando aos resultados mostrados na Tabela 2.2. Os resultados confirmam boa repetibilidade na obtenção do mínimo da função objetivo pelo PSO, e também boa convergência para o mínimo global.

Tabela 2.2 – Resultados das execuções do PSO para a função (2.6).

| | | | | | | |
|-----------|--------|--------|--------|--------|--------|--------------------------|
| $F(x, y)$ | -18,55 | -18,53 | -17,73 | -18,52 | -18,55 | $F(x^*, y^*) = -18.5547$ |
| x | 9,038 | 9,025 | 9,011 | 9,020 | 9,043 | $x^* = 9,039$ |
| y | 8,646 | 8,639 | 8,464 | 8,642 | 8,665 | $y^* = 8,668$ |

Para observar a convergência do PSO, a Figura 2.4 apresenta a curva de *fitness* para a primeira das execuções apresentadas na Tabela 2.2, em que se pode observar a convergência do algoritmo até encontrar o mínimo global.

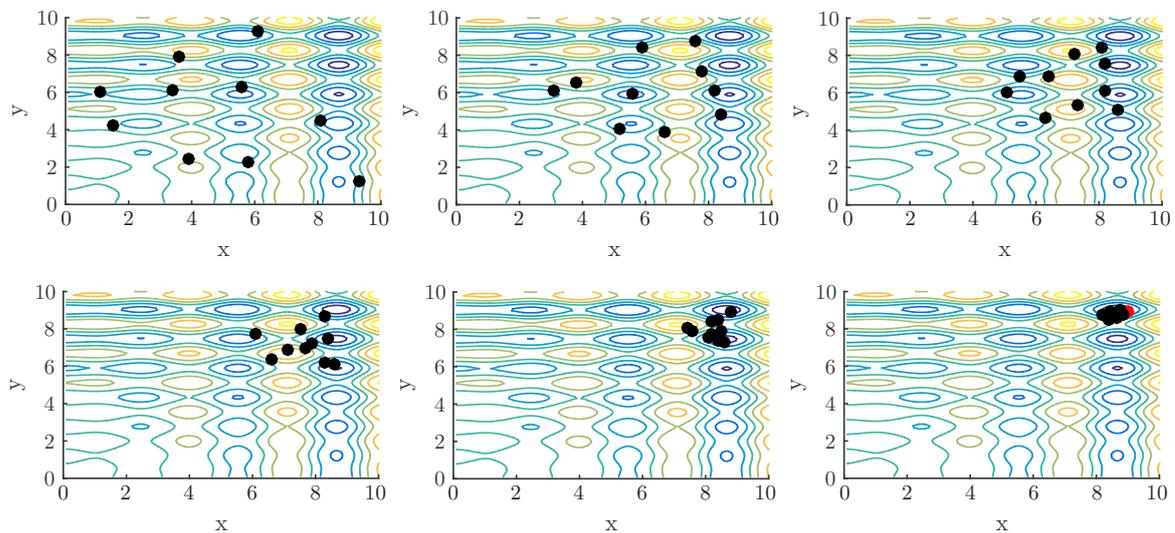
Figura 2.4 – Valor do *fitness* da otimização por meio do PSO para a função (2.6).



Fonte: Elaborado pelo autor.

A convergência das partículas no espaço de busca para esta execução pode ser observada na Figura 2.5, em que é apresentada a posição das partículas a cada 5 épocas do algoritmo. Pode-se notar, na primeira época, a posição aleatória das partículas e, com o decorrer das épocas, a convergência do enxame para a região em que está o ótimo global.

Figura 2.5 – Posição das 10 partículas do enxame nas épocas 1, 5, 10, 15, 20 e 25, mostrando a convergência das partículas durante uma execução do PSO.



Fonte: Elaborado pelo autor.

Resultados similares aos apresentados nas Figuras 2.4 e 2.5 foram obtidos para as demais execuções apresentadas na Tabela 2.2, confirmando a boa repetibilidade das execuções.

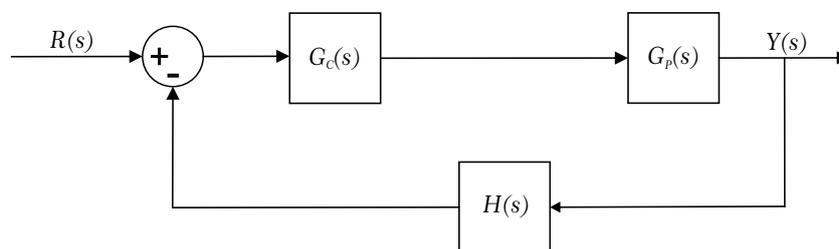
Por fim, o valor do *fitness* encontrado é muito próximo do valor calculado analiticamente para a função (2.6), dado por $-18,5547$. A configuração escolhida para o PSO

mostrou-se adequada, uma vez que os valores ótimos foram encontrados algumas épocas antes de o critério de parada ser atingido (número de épocas $M = 25$).

2.5 APLICAÇÃO EM UM PROBLEMA DE PROJETO DE CONTROLADOR

Esta seção ilustra uma solução utilizando o PSO para o problema de controle apresentado no Exemplo 6.9 em [Bazanella e Silva 2005], em que se deve projetar os ganhos de um controlador PID para o sistema de controle de uma malha, dado na Figura 2.6.

Figura 2.6 – Diagrama de blocos de um sistema de controle em malha fechada.



Fonte: Elaborado pelo autor.

No sistema de controle da Figura 2.6, a planta tem função de transferência dada por

$$G_p(s) = \frac{200}{s^2 + 2s + 100} \quad (2.7)$$

e o sensor tem função de transferência dada por

$$H(s) = \frac{400}{s + 400} \quad (2.8)$$

Segundo Bazanella & Silva (2005), deve-se projetar um controlador PID para este sistema para atender às seguintes especificações: margem de ganho ≥ 3 (9,5 dB); margem de fase $\geq 45^\circ$; estabilidade em malha fechada. Adicionalmente, um polo de alta frequência deve ser fixado em $p = 1000$ rad/s ao projeto do PID, utilizado para tornar a função de transferência do PID realizável na prática.

A função de transferência do controlador PID é então dada por

$$G_c(s) = \frac{K_d s^2 + K_p s + K_i}{s} \cdot \frac{p}{s + p} \quad (2.9)$$

Note que para o cômputo de uma função objetivo há três variáveis de decisão, os ganhos K_p , K_i e K_d , formando um espaço de busca tridimensional, limitado aqui nos intervalos $[-20; 20]$ para cada ganho, para incluir os ganhos de [Bazanella e Silva 2005].

Para computar a função objetivo deste problema sem a necessidade de formulação analítica para cada ganho de controle (K_p , K_i e K_d), acessado pelo espaço de busca,

utilizou-se a função *margin* do MATLAB sobre a função de transferência de malha aberta $G_{ma}(s)$, dada aqui por

$$G_{ma}(s) = G_c(s)G_p(s)H(s) \quad (2.10)$$

e a função *pole*, também do MATLAB, sobre a função de transferência de malha fechada $G_{mf}(s)$, dada aqui por

$$G_{mf}(s) = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)H(s)} \quad (2.11)$$

Assim, a função objetivo para este projeto foi definida como¹

$$F(K_p, K_i, K_d) = \begin{cases} \max(\text{real}(\text{pole}(G_{mf}(s)))) & , \quad \text{se margem de ganho} \geq 3 \\ & \text{e margem de fase} \geq 45^\circ. \\ \max(\text{real}(\text{pole}(G_{mf}(s)))) \times 0, 1 & , \quad \text{caso contrário.} \end{cases} \quad (2.12)$$

Note que, assumindo que o espaço de busca contém controladores que produzem instabilidade e também controladores que garantem estabilidade, esta função objetivo pode fornecer, respectivamente, resultados positivos ou negativos. Para candidatos a controladores que garantem a estabilidade, caso os limites especificados para margem de fase e margem de ganho não sejam satisfeitos, o valor negativo da função objetivo é multiplicado por um fator, a fim de penalizar esta possível solução.

A Tabela 2.3 mostra a configuração do PSO utilizada neste projeto de controlador.

Tabela 2.3 – Configuração do PSO, para o projeto do PID para o Exemplo 6.9 de [Bazanella e Silva 2005].

| Parâmetros do PSO | |
|-------------------------------------|-----|
| Número de partículas (N) | 25 |
| Número de épocas (M) | 100 |
| Coefficiente cognitivo (ϕ_1) | 0,5 |
| Coefficiente social (ϕ_2) | 0,5 |

Para os parâmetros na Tabela 2.3, o algoritmo PSO foi executado 5 vezes, com o tempo de simulação médio de 30 s, levando aos resultados apresentados na Tabela 2.4, indicando boa repetitibilidade dos projetos.

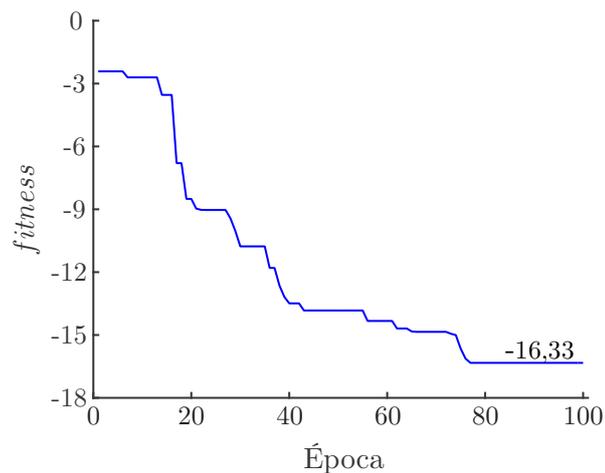
¹ $\max(\text{real}(\text{pole}(G_{mf}(s))))$ é a função que extrai a maior parte real dos polos em malha fechada.

Tabela 2.4 – Resultados de 5 execuções do PSO para o projeto do PID para o Exemplo 6.9, e os respectivos valores obtidos para margem de fase e margem de ganho.

| $F(K_p, K_i, K_d)$ | -16,33 | -15,97 | -16,22 | -16,17 | -16,13 |
|--------------------|--------|---------|---------|--------|---------|
| K_p | 3,139 | 3,020 | 3,104 | 3,089 | 3,074 |
| K_i | 19,99 | 19,14 | 19,87 | 19,61 | 19,64 |
| K_d | 0,215 | 0,211 | 0,213 | 0,213 | 0,212 |
| Margem de fase | 65,2° | 64,9° | 65,2° | 65° | 64,9° |
| Margem de ganho | 30 dB | 29,9 dB | 29,9 dB | 30 dB | 29,9 dB |

A Figura 2.7 mostra a curva de *fitness* para a primeira das execuções apresentadas na Tabela 2.4, em que se pode observar que o PSO converge para o mínimo ao final da execução.

Figura 2.7 – Valor do *fitness* para a primeira das execuções apresentadas na Tabela 2.4, da otimização por meio do PSO para a função (2.12).

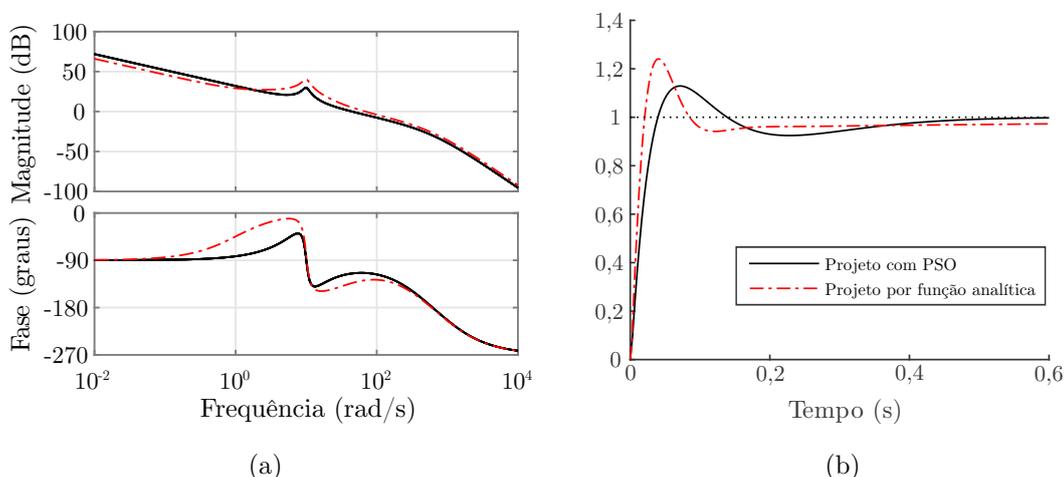


Fonte: Elaborado pelo autor.

O primeiro PID apresentado na Tabela 2.4 foi simulado e comparado com o PID projetado de forma analítica e gráfica, por meio do método da resposta em frequência, utilizado no Exemplo 6.9 em [Bazanella e Silva 2005], mostrado no Apêndice A desta dissertação.

Na Figura 2.8(a) são apresentados os diagramas de Bode do sistema em malha aberta, e, na Figura 2.8(b), são apresentadas as respostas ao degrau unitário do sistema em malha fechada com o PID projetado com o PSO e com o PID projetado de forma analítica.

Figura 2.8 – Simulações comparativas do ganho da primeira coluna na Tabela 2.4 e do controlador projetado por um método baseado na resposta em frequência: (a) diagrama de Bode do sistema em malha aberta; (b) resposta ao degrau unitário do sistema em malha fechada.



Fonte: Elaborado pelo autor.

Tanto o PID projetado de forma analítica quanto os PIDs obtidos pelo PSO atendem às especificações de projeto estabelecidas. Na Figura 2.8(a), o sistema com o controlador calculado de forma analítica possui margem de fase de 52° e margem de ganho de 26 dB, e o sistema controlado pelo PSO possui margem de fase de $65,2^\circ$ e margem de ganho de 30 dB, dadas na primeira coluna da Tabela 2.4.

Na Figura 2.8(b), o tempo de acomodação do controlador obtido de forma analítica é de 924 ms e o máximo sobressinal (*overshoot*) é de 24,1%, enquanto o valor para o sistema controlado com os PID projetado com o PSO é de 419 ms para o tempo de acomodação e 12,9% para o *overshoot*, indicando novamente superioridade do controlador sintonizado por meio do PSO.

Este exemplo mostra que outros critérios além da margem de fase e margem de ganho podem ser levados em consideração no projeto para permitir melhores resultados e também para aumentar as chances de sucesso de uma implementação prática. Por exemplo, juntamente com as margens de fase e de ganho, podem-se utilizar limites para a frequência de cruzamento, *overshoot*, sinal de controle, entre outros, para orientar o PSO na busca de um melhor controlador. Além disso, é interessante adicionar robustez no projeto do controlador, ao considerar que os parâmetros da planta podem assumir valores incertos pertencentes a intervalos.

Neste sentido, o capítulo a seguir traz a principal proposta deste trabalho, que é um procedimento de projeto de controladores robustos baseado no PSO, que investiga múltiplas especificações de controle de interesse prático, de forma simultânea, a fim de garantir controladores com bom desempenho e robustez frente a incertezas paramétricas.

2.6 CONSIDERAÇÕES FINAIS

Este capítulo apresentou o algoritmo PSO, incluindo seu princípio de funcionamento, pseudo-código e inicialização. O PSO foi validado para uma função de *benchmark* da literatura, para a qual demonstrou boa capacidade de investigar um espaço de busca com múltiplos mínimos locais. O PSO também foi aplicado ao projeto de um PID em um exemplo didático, sendo capaz de encontrar ganhos de um controlador com estrutura similar, mas com desempenho superior ao de PIDs projetados de forma analítica para garantir margem de fase e margem de ganho pré-especificadas. Em ambos os exemplos, o PSO demonstrou boa repetibilidade e baixo tempo médio de execução.

3 PROCEDIMENTO DE PROJETO PROPOSTO

O objetivo deste capítulo é apresentar o procedimento de projeto proposto para obtenção de controladores robustos aplicáveis em conversores de potência, baseando-se no uso do algoritmo PSO e no Teorema de Kharitonov. O procedimento permite a obtenção de controladores robustos para conversores de potência que possuem parâmetros incertos, pertencentes a intervalos reais, cujos limitantes são conhecidos. Um exemplo de aplicação a um conversor *Buck* ilustra a utilização do procedimento, garantindo bom desempenho do sistema de controle e um certificado teórico de robustez frente a incertezas paramétricas.

3.1 MODELO DA PLANTA COM INCERTEZAS E DO CONTROLADOR ROBUSTO

Primeiramente, considere o modelo de uma planta com uma entrada e uma saída, descrito em função de transferência como

$$G(s) = \frac{f_0(\mathbf{p}) + f_1(\mathbf{p})s + f_2(\mathbf{p})s^2 + \cdots + f_n(\mathbf{p})s^n}{g_0(\mathbf{p}) + g_1(\mathbf{p})s + g_2(\mathbf{p})s^2 + \cdots + g_n(\mathbf{p})s^n} \quad (3.1)$$

em que os coeficientes f_i e g_i , $i = 0 \cdots n$, são funções de parâmetros físicos incertos \mathbf{p} , assumidos pertencentes a intervalos reais, dados por

$$\mathbf{p} = [p_1, p_2, \cdots, p_\ell] \quad , \quad p_l \in [p_l^-, p_l^+] \quad , \quad l = 1 \cdots \ell \quad (3.2)$$

Este modelo de planta, com coeficientes incertos, pode ser utilizado para representar o comportamento médio linearizado de conversores de potência, em que os parâmetros físicos (por exemplo, indutâncias, capacitâncias, resistências, etc.) não são precisamente conhecidos, mas pertencem a intervalos em que apenas os extremos são informados [Ackermann 2012].

Para cada uma das combinações de máximos e mínimos dos elementos de \mathbf{p} , pode-se escrever uma função de transferência $G_v(s)$, chamada vértice de um modelo politópico, que é dado por [Boyd et al. 1994]

$$G(s) = \sum_{v=1}^V \theta_v G_v(s) \quad , \quad \theta_v \geq 0 \quad , \quad v = 1 \cdots V \quad , \quad \sum_{v=1}^V \theta_v = 1 \quad (3.3)$$

Assim, ao combinar todos os valores extremos dos parâmetros físicos incertos em (3.2), o modelo politópico resultante possuirá $V = 2^\ell$ vértices.

Considere a função de transferência de um controlador, dada por

$$G_c(s) = \frac{x_0 + x_1s + x_2s^2 + \cdots + x_ms^m}{y_0 + y_1s + y_2s^2 + \cdots + y_ms^m} \quad (3.4)$$

com coeficientes fixos, descritos pelo vetor

$$\mathbf{c} = [x_0, x_1, \dots, x_m, y_0, y_1, \dots, y_m] \quad (3.5)$$

Por exemplo, nesta notação, um controlador PI é dado por

$$G_c(s) = \frac{K_p s + K_i}{s} \quad (3.6)$$

com os coeficientes em (3.6) sendo $x_0 = K_i$, $x_1 = K_p$, $y_0 = 0$, $y_1 = 1$, e o vetor dos coeficientes do controlador sendo dado por $\mathbf{c} = [K_i, K_p, 0, 1]$.

Para garantir estabilidade e desempenho do sistema de controle em malha fechada para todo o domínio de incertezas (i.e., para todos os valores que os parâmetros incertos podem assumir), os ganhos fixos de controle serão aqui computados de forma *off-line*, e não haverá alterações dos ganhos na operação do controle em tempo real, implementando a lei de controle de forma simples, evitando estratégias de ganho variável, como estratégias de ganho escalonado, *fuzzy* ou adaptativas.

3.2 CRITÉRIOS DE DESEMPENHO UTILIZADOS

Em procedimentos clássicos de projeto de controladores para conversores de potência, para encontrar os coeficientes da função de transferência do controlador em (3.4) que garantam boas respostas do sistema, é comum especificar o desempenho em termos de critérios pertencentes ao domínio da frequência como, por exemplo, a margem de ganho (do inglês, *gain margin* – GM), a margem de fase (do inglês, *phase margin* – PM) e a frequência de cruzamento (do inglês, *crossover frequency* – ω_{co}) [Dorf e Bishop 2011, Erickson 1997]. Porém, ao tratar com plantas com incertezas, deve-se verificar os valores de pior caso destas métricas (i.e., valores mínimos de GM, PM e ω_{co}), que podem ser obtidos, por exemplo, por meio de uma avaliação de respostas em frequência para uma discretização fina no espaço de parâmetros \mathbf{p} .

Especificações de projeto no domínio do tempo também são relevantes, como as relativas à resposta em malha fechada ao degrau unitário, tais como: máximo sobressinal (do inglês, *overshoot* – OV) e erro em regime permanente (do inglês, *steady state error* – e_{ss}), por exemplo. Uma boa combinação de especificações no domínio do tempo e da frequência é interessante, mas pode ser um desafio que se torna mais complexo de ser solucionado quando incertezas paramétricas e saturação da ação de controle (i.e., um limite superior para a amplitude do sinal de controle u) são levadas em consideração no estágio de projeto do controlador.

Uma forma rápida de estimar os valores de pior caso, isto é, valores mínimos de PM, GM, ω_{co} , e valores máximos de OV, e_{ss} e do sinal de controle, para um conjunto

de plantas que determinam possíveis representações de um conversor com parâmetros incertos (3.1), e para um dado controlador com ganhos fixos (3.4), é utilizando simulações com a representação politópica da planta (3.3). Assim, por meio de simulações para os vértices do modelo politópico, pode-se estimar o desempenho do sistema para todo o domínio de incertezas [Boyd et al. 1994, Ramos, Bretas e Alberto 2002].

Entretanto, no caso da estabilidade, a avaliação apenas nos vértices do modelo politópico não garante que o sistema de controle seja estável para todo o domínio de incertezas paramétricas. Tendo isso em vista, para o projeto de controladores robustos, torna-se importante acrescentar, na etapa de projeto do controlador, um teste que certifica teoricamente a estabilidade robusta do sistema em malha fechada frente às incertezas paramétricas do conversor, sendo aqui utilizado o Teorema de Kharitonov para este fim.

3.3 ESTABILIDADE ROBUSTA BASEADA NO TEOREMA DE KHARITONOV

Para sistemas de controle em que os coeficientes do polinômio característico do sistema em malha fechada não são precisamente conhecidos, mas pertencem a intervalos reais limitados, uma condição para analisar a estabilidade é dada pelo Teorema de Kharitonov [Bhattacharyya, Chapellat e Keel 1995].

Para aplicar o Teorema de Kharitonov ao problema apresentado aqui, considere que o sistema (3.1) pode ser representado pela família de plantas intervalares de ordem n , dada por

$$G_{KT}(s) = \frac{a_0 + a_1s + a_2s^2 + \dots + a_ns^n}{b_0 + b_1s + b_2s^2 + \dots + b_ns^n} \quad (3.7)$$

em que os coeficientes a_i e b_i , $i = 0 \dots n$, são obtidos a partir da avaliação de $f_i(\mathbf{p})$ e $g_i(\mathbf{p})$, $i = 0 \dots n$, em (3.1), levando aos limites dados por

$$a_i \in [a_i^-, a_i^+], \quad b_i \in [b_i^-, b_i^+], \quad i = 0, \dots, n \quad (3.8)$$

Dado o controlador (3.4) e a planta intervalar (3.7), o polinômio característico $D(s)$ do sistema em malha fechada, com realimentação unitária, pode ser escrito como

$$D(s) = d_0 + d_1s + d_2s^2 + \dots + d_{n+m}s^{n+m} \quad (3.9)$$

com os coeficientes intervalares sendo delimitados por

$$d_i \in [d_i^-, d_i^+] \quad , \quad i = 0, \dots, n + m \quad (3.10)$$

Pelo Teorema de Kharitonov, tem-se que se os quatro polinômios [Bhattacharyya

e Keel 1995]

$$\begin{aligned}
 K_1(s) &= d_o^- + d_1^- s + d_2^+ s^2 + d_3^+ s^3 + d_4^- s^4 + \dots, \\
 K_2(s) &= d_o^- + d_1^+ s + d_2^+ s^2 + d_3^- s^3 + d_4^- s^4 + \dots, \\
 K_3(s) &= d_o^+ + d_1^- s + d_2^- s^2 + d_3^+ s^3 + d_4^+ s^4 + \dots, \\
 K_4(s) &= d_o^+ + d_1^+ s + d_2^- s^2 + d_3^- s^3 + d_4^+ s^4 + \dots.
 \end{aligned} \tag{3.11}$$

são Hurwitz, então $D(s)$ em (3.9) é Hurwitz e, portanto, o sistema em malha fechada é estável para todos os valores nos coeficientes intervalares.

Uma prova do Teorema de Kharitonov pode ser encontrada no Capítulo 5, em [Bhattacharyya, Chapellat e Keel 1995].

Definição 1. *Se (3.11) for satisfeito no cenário especificado acima, o sistema incerto (3.1), com controlador (3.4), em malha fechada, será chamado a partir deste ponto de **KT estável**, i.e., estável baseado no Teorema de Kharitonov.*

O Teorema de Kharitonov permite testar, com um número finito de avaliações (apenas quatro testes), se o sistema de controle com um domínio de incertezas de infinitos pontos é estável, evitando utilizar análises baseadas em discretizações exaustivas, que podem demandar muito tempo e não têm garantia teórica sobre a estabilidade.

É importante mencionar que, para o caso em que $D(s)$ em (3.9) possui os coeficientes independentes entre si, o Teorema de Kharitonov fornece uma condição necessária e suficiente para estabilidade robusta em todo intervalo de valores dos coeficientes da planta [Bhattacharyya e Keel 1995].

Ao tratar $D(s)$ com coeficientes que não são independentes entre si, por exemplo, no caso de (3.1) em que os parâmetros físicos podem ter impacto simultâneo em mais de um coeficiente da função, o Teorema de Kharitonov em (3.11) é uma condição suficiente para a estabilidade robusta do sistema em malha fechada. Porém, devido à simplicidade do Teorema de Kharitonov, a sua utilização como uma primeira e rápida avaliação da estabilidade é muito atrativa, e por isso será abordada nesta dissertação [Barmish 1989, Bhattacharyya e Keel 1995].

3.4 DEFINIÇÃO DO PROBLEMA DE PROJETO DO CONTROLADOR

O problema a ser resolvido neste trabalho é fornecer um procedimento capaz de encontrar, *off-line*, um vetor de ganhos fixos de controle (3.5) (i.e., um controlador robusto), que garanta que o sistema de controle com planta (3.1) e controlador (3.4), como na Figura 2.6:¹

- seja estável para todo o conjunto de parâmetros incertos da planta;

¹É suposta realimentação unitária mas, sem perda da generalidade da abordagem, qualquer função de transferência própria do sensor, sem incertezas paramétricas, também pode ser utilizada.

- atenda, para os vértices do modelo politópico, a especificações de desempenho em relação a valores de referência para margem de fase (PM^*) e de frequência de cruzamento (ω_{co}^*);
- respeite, para os vértices do modelo politópico, limite mínimo de margem de ganho (\underline{GM}), e limites máximos de *overshoot* (\overline{OV}), erro em regime permanente ($\overline{e_{ss}}$) e da amplitude do sinal de controle (\overline{u}).

Uma solução sistemática para este problema é fornecida a seguir, utilizando o algoritmo PSO combinado com o Teorema de Kharitonov. Para isso, pode-se definir um espaço de busca para o vetor de ganhos de controle e associar a cada ponto deste espaço um valor positivo. Este valor positivo pode ser interpretado como um custo, que será mais baixo quando: i) a margem de fase e a frequência de cruzamento estiverem mais próximas dos valores de referência, ii) as restrições para valores mínimos de margem de ganho e para valores máximos de *overshoot*, de erro em regime permanente e do sinal de controle não forem violadas, e quando iii) a estabilidade robusta do sistema em malha fechada for garantida pelo Teorema de Kharitonov.

A próxima seção propõe uma função objetivo que leva em consideração os critérios de projeto supracitados e uma forma direta de definir o espaço de busca dos ganhos de controle.

3.5 FUNÇÃO OBJETIVO E ESPAÇO DE BUSCA

Para resolver o problema de projeto como um problema de otimização, uma função objetivo pode ser utilizada para relacionar cada controlador \mathbf{c} com um valor que represente o desempenho do sistema $f(\mathbf{c})$. Para cada candidato a vetor de controle \mathbf{c} , em (3.5), a função objetivo proposta nesta dissertação retorna um valor escalar real positivo, calculado com base em três termos, $\alpha(\mathbf{c})$, $\beta(\mathbf{c})$ e $\gamma(\mathbf{c})$, que medem o desempenho do sistema com este controlador, utilizando a seguinte expressão:

$$f(\mathbf{c}) = \alpha(\mathbf{c}) \beta(\mathbf{c}) \gamma(\mathbf{c}) \quad (3.12)$$

O primeiro termo, $\alpha(\mathbf{c})$, é definido por

$$\alpha(\mathbf{c}) = \max_{j=1, \dots, V} \left(\left| \frac{PM^* - PM_j(\mathbf{c})}{PM^*} \right| + \left| \frac{\omega_{co}^* - \omega_{coj}(\mathbf{c})}{\omega_{co}^*} \right| \right) \quad (3.13)$$

e mede a maior soma dos desvios normalizados em relação à margem da fase e à frequência de cruzamento de referência, considerando todos os vértices do modelo politópico, representados por $j = 1 \dots V$. Os valores de $PM_j(\mathbf{c})$ e $\omega_{coj}(\mathbf{c})$ podem ser facilmente computados para plantas e controladores de ordem arbitrária, por meio de rotinas especializados, como

a função *margin*, do MATLAB.

Observe que avaliar apenas $\alpha(\mathbf{c})$ pode não garantir que o controlador candidato produza boas respostas. Por exemplo, mesmo com $\alpha(\mathbf{c})$ baixo, pode-se produzir respostas no domínio do tempo com alto *overshoot* ou com saturação na amplitude do sinal de controle. A fim de obter mais garantias no projeto, o segundo termo em (3.12), $\beta(\mathbf{c})$, desempenha o papel de um fator de penalização na função objetivo, e é computado da seguinte forma

$$\beta(\mathbf{c}) = \begin{cases} 1, & \text{se } \text{GM}_j(\mathbf{c}) \geq \underline{\text{GM}} \text{ e } \text{OV}_j(\mathbf{c}) \leq \overline{\text{OV}} \text{ e } |e_{ssj}(\mathbf{c})| \leq \overline{e_{ss}} \\ & \text{e } |u_j(\mathbf{c})| \leq \bar{u} \text{ para todo } j = 1, \dots, V \\ 10^6, & \text{caso contrário} \end{cases} \quad (3.14)$$

O termo $\beta(\mathbf{c})$ retorna valor unitário se todas as condições em (3.14) forem satisfeitas, isto é, se o controlador candidato \mathbf{c} garante, para todos os vértices do polítopo, conformidade com o limite inferior $\underline{\text{GM}}$ e com os limites superiores $\overline{\text{OV}}$, $\overline{e_{ss}}$ e \bar{u} . Caso contrário, $\beta(\mathbf{c})$ retorna o valor 10^6 , a fim de penalizar a função objetivo associada com este controlador candidato, por violação de alguma destas restrições de projeto. As medidas em (3.14) podem ser facilmente realizadas, para plantas e controladores de ordens arbitrárias, por exemplo, por meio das funções *margin* e *step*, do MATLAB.

Observe que a escolha dos valores de referência em $\alpha(\mathbf{c})$ e limitantes em $\beta(\mathbf{c})$ não são uma tarefa difícil ao projetista, pois valores típicos podem ser encontrados na literatura ou definidos com base no conversor sob investigação (e.g., PM $\simeq 60^\circ$, GM de 2 até 5, ω_{co} uma década abaixo da frequência de comutação, etc.) [Erickson 1997, Dorf e Bishop 2011].

O terceiro termo da função objetivo proposta, $\gamma(\mathbf{c})$, também desempenha o papel de um fator de penalização, relacionado à comprovação matemática da estabilidade robusta do sistema, dada pelo Teorema de Kharitonov, e é avaliado como

$$\gamma(\mathbf{c}) = \begin{cases} 1, & \text{se o sistema em malha fechada é } KT \text{ estável} \\ 10^6, & \text{caso contrário} \end{cases} \quad (3.15)$$

Note que enquanto o cômputo de $\alpha(\mathbf{c})$ e $\beta(\mathbf{c})$ são feitos a partir da representação politópica da planta, ou seja, para V vértices, o termo $\gamma(\mathbf{c})$ avalia o controlador candidato sempre para os quatro polinômios apresentados em (3.11).

Note também que os termos da função objetivo proposta em (3.12) podem ser de difícil expressão analítica, à medida que aumenta a ordem da planta e do controlador. Devido à não necessidade de definir analiticamente a função objetivo, podendo-se computá-la a partir de simulações com funções especializadas (do MATLAB, por exemplo), reforça-se o uso do algoritmo PSO como uma ferramenta atrativa para encontrar um vetor de ganhos \mathbf{c} de controle que resolva o problema de otimização com eficiência computacional.

Assim, o problema de projeto é encontrar um controlador \mathbf{c} que minimiza a função objetivo (3.12). Este problema pode ser definido por

$$\mathbf{c}^* = \arg \min_{\mathbf{c} \in \mathcal{C}} f(\mathbf{c}) \quad (3.16)$$

em que \mathbf{c}^* é o controlador que minimiza a função objetivo em um espaço de busca dos controladores dado por \mathcal{C} .

Espaço de busca

O espaço de busca dos ganhos de controle é definido aqui como um hiper-retângulo descrito por

$$\mathcal{C} = \left\{ \begin{array}{l} (x_0, x_1, \dots, x_m, y_0, y_1, \dots, y_m) \in \mathcal{R}^{2m+2} \\ x_k^- \leq x_k \leq x_k^+, \quad y_k^- \leq y_k \leq y_k^+, \quad k = 0, \dots, m \end{array} \right. \quad (3.17)$$

Neste trabalho, este espaço é determinado baseando-se na positividade dos coeficientes do polinômio característico (3.9), para todas as combinações possíveis de parâmetros incertos \mathbf{p} . Essa escolha é fundamentada na conhecida condição necessária para a estabilidade do critério de Routh [Dorf e Bishop 2011].

A vantagem é que este espaço pode ser obtido sistematicamente a partir de um conjunto de desigualdades lineares, resolvendo um problema de programação linear e, em seguida, incluindo a região resultante em um hiper-retângulo, conforme descrito em (3.17). A inclusão da região resultante em um hiper-retângulo deve-se ao fato de que no algoritmo PSO são informados apenas os limites inferiores e superiores de cada partícula no espaço de busca.

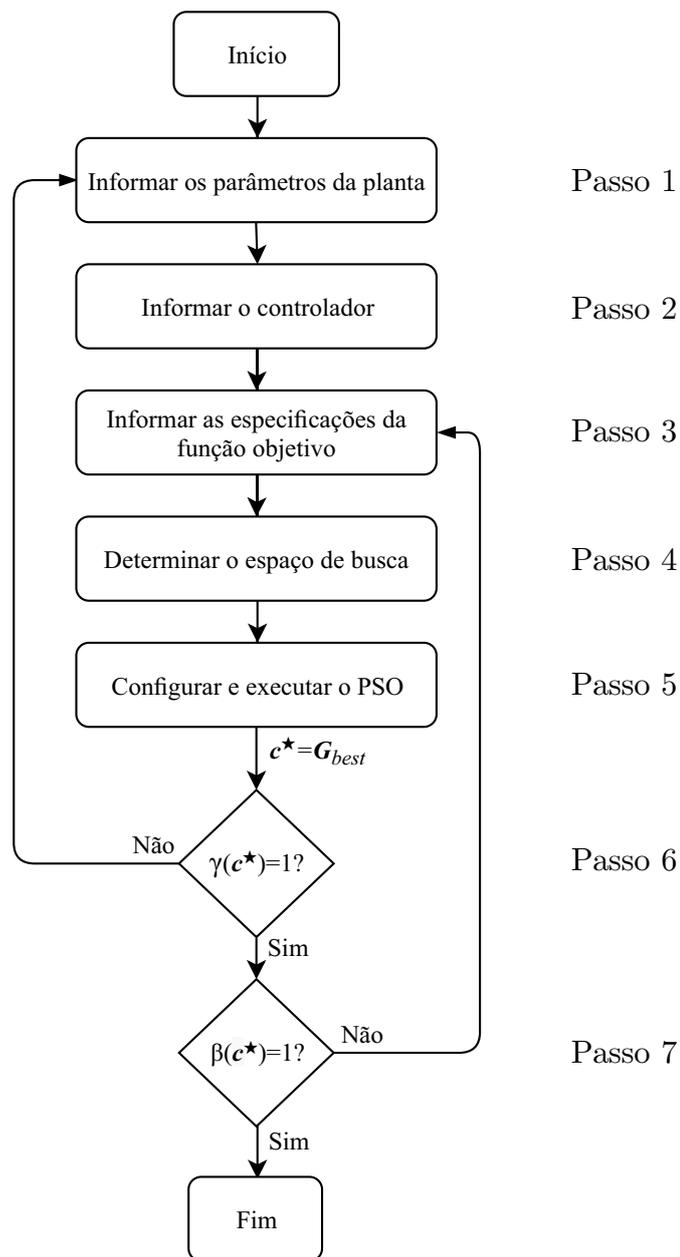
Como \mathcal{C} é definido a partir da positividade dos coeficientes do polinômio característico, pode-se ter um espaço de busca grande, e técnicas de discretização exaustiva do espaço de busca geralmente são inviáveis para uma solução com alta precisão.

Por fim, é importante observar que um espaço de busca mais restrito pode ser obtido aplicando, por exemplo, o critério completo de Routh-Hurwitz [Dorf e Bishop 2011]. Também, cabe mencionar que o Teorema de Kharitonov já foi utilizado na literatura para definir regiões mais precisas para o espaço de busca dos ganhos do controlador, em uma etapa prévia ao projeto do controlador, como em [Shin et al. 2003, Bevrani et al. 2010, Kumar e Shreesha 2016, Nazari et al. 2018]. Em ambos os casos, a inicialização do espaço de busca demanda cálculos mais demorados e complexos. Aqui, opta-se por definir o espaço de busca com cálculos simples e rápidos, tendo como resultado um espaço de busca potencialmente grande, mas que pode ser avaliado devido à capacidade do algoritmo PSO de investigar grandes espaços em tempo computacional viável, como será comprovado nos exemplos a seguir.

3.6 SUMARIZAÇÃO DO PROCEDIMENTO DE PROJETO PROPOSTO

O procedimento de projeto proposto para resolver o problema descrito na Seção 3.4 é resumido no fluxograma da Figura 3.1.

Figura 3.1 – Fluxograma do procedimento de projeto proposto.



Fonte: Elaborado pelo Autor.

O procedimento é executado por meio dos seguintes passos:

- **Passo 1:** informar o modelo $G(s)$ e os parâmetros incertos da planta, conforme mostrado em (3.1) e (3.2);

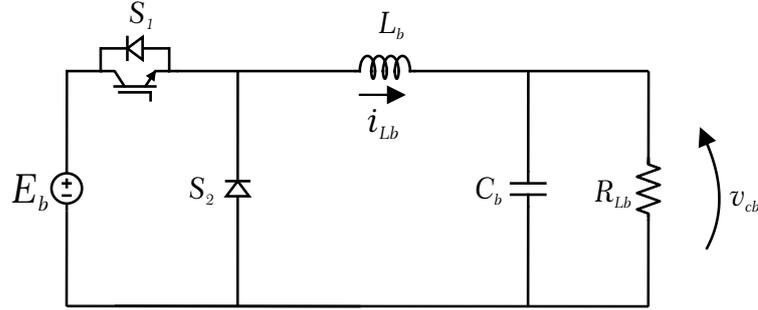
- **Passo 2:** informar $G_c(s)$, definindo o vetor de ganhos do controlador em (3.5);
- **Passo 3:** informar as especificações em (3.13) e (3.14) (i.e., PM^* , ω_{co}^* , \underline{GM} , \overline{OV} , $\overline{e_{ss}}$ e \bar{u}), escolhidas para avaliação da função objetivo (3.12);
- **Passo 4:** determinar o espaço de busca \mathcal{C} (3.17), com base na positividade dos coeficientes do polinômio característico do sistema em malha fechada (3.9);
- **Passo 5:** escolher os parâmetros de configuração do PSO (N , M , ϕ_1 e ϕ_2) e executar o algoritmo. Nesta etapa, a cada iteração do PSO, cada partícula (controlador candidato) é avaliada, com base na função objetivo (3.12), incluindo a avaliação da estabilidade robusta, usando o Teorema de Kharitonov. Ao final deste passo, o PSO fornece o vetor de ganhos de controle \mathbf{c}^* ;
- **Passos 6:** verificar se os ganhos de controle em \mathbf{c}^* , fornecidos pelo PSO, garantem que o sistema em malha fechada é *KT estável*. Em caso positivo, segue-se para o Passo 7. Caso contrário, o procedimento deve ser executado novamente, e as especificações dos parâmetros da planta devem ser atualizadas;
- **Passos 7:** verificar se os ganhos de controle em \mathbf{c}^* , fornecidos pelo PSO, atendem às restrições em (3.14). Em caso positivo, o procedimento termina, fornecendo \mathbf{c}^* como um controlador robusto viável. Caso contrário, o procedimento deve ser executado novamente, e as especificações da função objetivo devem ser atualizadas.

Note que os passos 6 e 7 são incluídos no procedimento para assegurar que \mathbf{c}^* garanta que o sistema em malha fechada é *KT estável*, e que o sistema controlado não viola os limites de projeto, evitando que controladores resultantes que não possuem $\gamma(\mathbf{c}) = 1$ ou $\beta(\mathbf{c}) = 1$ sejam aceitos como solução.

A seguir, para ilustrar a utilização do procedimento de projeto proposto, é apresentado o seu emprego para o projeto de um controlador para um conversor *Buck* CC–CC.

3.7 EXEMPLO PARA O CONTROLE DE TENSÃO DE UM CONVERSOR BUCK

Para detalhar cada etapa do procedimento de projeto proposto, esta seção apresenta como exemplo o projeto de um controlador PID para a regulação da tensão de saída de um conversor *Buck* CC–CC, dado na Figura 3.2 [Erickson 1997].

Figura 3.2 – Circuito de um conversor *Buck* CC-CC.

Fonte: Elaborado pelo Autor.

Passo 1: Modelagem com parâmetros incertos

Supondo o conversor *Buck* em modo de condução contínua (do inglês, *continuous conduction mode* – CCM), este possui duas etapas de operação, conhecidas como etapas de carga e descarga [Erickson 1997].

As equações que definem a etapa de carga podem ser escritas como

$$\frac{di_{Lb}}{dt} = \frac{E_b - v_{cb}}{L_b} \quad \frac{dv_{cb}}{dt} = \frac{i_{Lb}}{C_b} - \frac{v_{cb}}{R_{Lb}C_b} \quad (3.18)$$

e as equações que definem a etapa de descarga podem ser escritas como

$$\frac{di_{Lb}}{dt} = \frac{-v_{cb}}{L_b} \quad \frac{dv_{cb}}{dt} = \frac{i_{Lb}}{C_b} - \frac{v_{cb}}{R_{Lb}C_b} \quad (3.19)$$

A partir das equações (3.18) e (3.19), pode-se obter o modelo médio do conversor e, por meio deste, a função de transferência, considerando como entrada a razão cíclica e como saída a tensão sobre a carga, dada por [Erickson 1997]

$$G(s) = \frac{v_{cb}}{d_b} = \frac{\frac{E_b}{L_b C_b}}{s^2 + \frac{1}{R_{Lb} C_b} s + \frac{1}{L_b C_b}} \quad (3.20)$$

Note que a função em (3.20) é adequada para o modelo da planta definido para o procedimento proposto em (3.1).

Uma vez que o primeiro passo do procedimento consiste em informar os parâmetros incertos $[p_l^-, p_l^+]$ e o modelo do conversor em função de transferência, assumindo, neste exemplo, como parâmetros incertos o valor da tensão de entrada, E_b , e o valor da resistência de carga, R_{Lb} , o modelo do conversor *Buck* pode ser escrito como

$$G(s) = \frac{f_0(p_1)}{g_2 s^2 + g_1(p_2)s + g_0} \quad (3.21)$$

sendo

$$f_0(p_1) = \frac{E_b}{L_b C_b}, \quad g_2 = 1, \quad g_1(p_2) = \frac{1}{R_{Lb} C_b}, \quad g_0 = \frac{1}{L_b C_b}, \quad p_1 = E_b, \quad p_2 = R_{Lb} \quad (3.22)$$

Para este exemplo, são utilizados os parâmetros do conversor *Buck* dado em [Lobato et al. 2017], com a inclusão aqui de incertezas paramétricas, conforme a Tabela 3.1.

Tabela 3.1 – Parâmetros da planta (conversor *Buck*).

| Descrição do sistema | |
|---|------------------------|
| Tensão de entrada (E_b) | 12 V $\pm 10\%$ |
| Tensão de saída (referência) (v_{cb}) | 5 V |
| Frequência de comutação (f_{sb}) | 20000 Hz |
| Capacitor do filtro (C_b) | 10 μ F |
| Indutor do filtro (L_b) | 726 μ H |
| Carga resistiva (R_{Lb}) | 22 Ω $\pm 50\%$ |

Passo 2: Escolha do controlador

O segundo passo do procedimento de projeto proposto é escolher a estrutura do controlador, dada por (3.4) e (3.5). Para controlar de forma adequada a tensão de saída do conversor *Buck*, optou-se por um controlador PID [Guo, Hung e Nelms 2009].

A função de transferência do controlador PID ideal é dada por

$$G_c(s) = \frac{K_d s^2 + K_p s + K_i}{s} \quad (3.23)$$

com $\mathbf{c} = [x_0, x_1, x_2, y_0, y_1, y_2] = [K_i, K_p, K_d, 0, 1, 0]$, ou, de forma mais compacta, pode-se denotar $\mathbf{c} = [K_i, K_p, K_d]$, para PIDs.

Passo 3: Especificações da função objetivo

O terceiro passo do procedimento é escolher as especificações da função objetivo (3.12) para o projeto do controlador PID. Os valores de referência e os limites de desempenho são especificados na Tabela 3.2.

Tabela 3.2 – Especificações da função objetivo para o projeto do controlador PID para o conversor *Buck*.

| Valores de referência para $\alpha(\mathbf{c})$ | |
|---|---------------------|
| Frequência de cruzamento ω_{co}^* | 12560 rad/s (2 kHz) |
| Margem de fase PM* | 60° |
| Valores limites para $\beta(\mathbf{c})$ | |
| Margem de ganho \underline{GM} | 5 (14 dB) |
| <i>Overshoot</i> \overline{OV} | 10% |
| Erro em regime permanente $\overline{e_{ss}}$ | 1% |
| Amplitude do sinal de controle \bar{u} | 1 |

O valor de referência da frequência de cruzamento ω_{co} foi escolhido uma década abaixo do valor da frequência de comutação, f_{sb} , dado na Tabela 3.1 (i.e., $\omega_{co} = 2\pi f_{sb}/10$). O valor escolhido para a máxima amplitude do sinal de controle é unitário, por ser o limite da razão cíclica d_b (sinal de controle). Os valores de referência e limitantes para as demais especificações são típicos, encontrados na literatura [Erickson 1997, Dorf e Bishop 2011].

Passo 4: Definição do espaço de busca

O quarto passo é determinar o espaço de busca, com base no polinômio característico de malha fechada (3.9). A partir de $G(s)$, em (3.20), e com um controlador PID, em (3.23), o polinômio característico do sistema em malha fechada é dado por

$$D(s) = s^3 + \left(\frac{1}{R_{Lb}C_b} + \frac{E_b}{L_bC_b} K_d \right) s^2 + \left(\frac{1}{L_bC_b} + \frac{E_b}{L_bC_b} K_p \right) s + \left(\frac{E_b}{L_bC_b} K_i \right) \quad (3.24)$$

As desigualdades usadas para definir o espaço de busca para os ganhos de controle são então definidas por

$$K_i > 0, \quad K_p > -\frac{1}{E_b}, \quad K_d > -\frac{L_b}{R_{Lb}E_b} \quad (3.25)$$

Note que as desigualdades acima definem intervalos abertos para os coeficientes do controlador. Como critério prático, escolheu-se como limite superior, para todos os coeficientes, o valor 10^4 , por ser um valor significativamente grande para os ganhos de controle.

Passo 5: Configuração e execução do PSO

O quinto passo do procedimento é configurar e executar o PSO. A configuração usada para o PSO neste exemplo é dada na Tabela 3.3.

Tabela 3.3 – Configuração do PSO, para o exemplo do conversor *Buck*.

| Parâmetros do PSO | |
|-------------------------------------|-----|
| Número de partículas (N) | 300 |
| Número de épocas (M) | 50 |
| Coefficiente cognitivo (ϕ_1) | 0,5 |
| Coefficiente social (ϕ_2) | 0,5 |

Cabe lembrar que o Teorema de Kharitonov é testado em cada avaliação da função objetivo, durante a execução do PSO, a partir dos quatro polinômios dados em (3.11).

Para obter uma estatística, o algoritmo PSO foi executado 30 vezes, por meio da função *particleswarm* do MATLAB. Todas execuções foram feitas de forma *off-line* e independentes entre si, tendo convergido em todos os casos, com tempo médio de 9 minutos, para controladores viáveis [Arcuri e Briand 2011].

Uma execução típica do algoritmo encontrou os seguintes ganhos de controle

$$\mathbf{c}^* = [K_i \ K_p \ K_d] = [1334, 163592857 \quad 0,04464179776421 \quad 7,87633899272 \times 10^{-6}] \quad (3.26)$$

A Tabela 3.4 apresenta os valores de pior caso obtidos para as medidas de desempenho, ao avaliar os vértices do modelo politópico do sistema com os ganhos de controle obtidos em (3.26). Consideram-se aqui como valores de pior caso os valores mínimos obtidos para GM, PM e ω_{co} e os valores máximos obtidos para OV, e_{ss} e u , para os testes nos vértices do politopo.

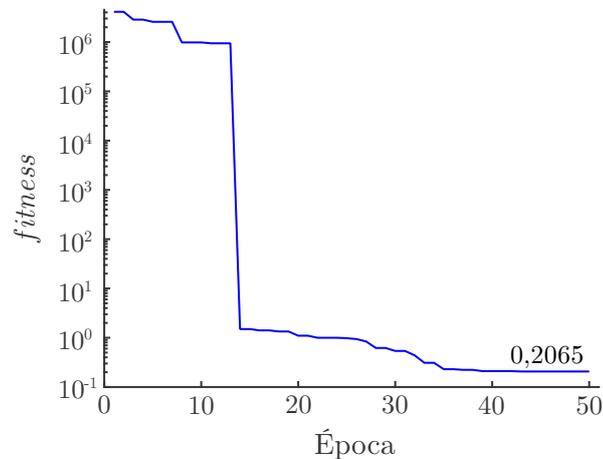
Tabela 3.4 – Valores de pior caso obtidos avaliando os vértices do sistema com os ganhos de controle (3.26).

| Índices de desempenho obtidos a partir do ganho \mathbf{c}^* | |
|--|---------------------------|
| Frequência de cruzamento ω_{co} | 10263,48 rad/s (1,63 kHz) |
| Margem de fase PM | 53,95° |
| Margem de ganho GM | ∞ |
| <i>Overshoot</i> OV | 9,99% |
| Erro em regime permanente e_{ss} | Nulo |
| Amplitude do sinal de controle u | 0,4949 |

Para verificar o processo de otimização fornecido pelo PSO, a Figura 3.3 apresenta o melhor valor do *fitness* encontrado a cada época na execução do algoritmo. Pode-se

notar que, nas primeiras épocas, o *fitness* é da ordem de 10^6 , o que indica que todas as partículas do enxame estavam em uma região penalizada por $\beta(\mathbf{c})$ ou $\gamma(\mathbf{c})$. Entre as épocas 10 e 20, pode-se notar uma redução significativa no valor do *fitness*. Após esta redução, a função objetivo não é mais penalizada (i.e., $\beta(\mathbf{c}) = 1$ e $\gamma(\mathbf{c}) = 1$), e seu valor depende apenas do termo $\alpha(\mathbf{c})$. Ao fim do processo de otimização, pode-se afirmar que o número de épocas escolhido neste exemplo foi suficiente para a convergência, devido ao fato de o valor do *fitness* não apresentar redução significativa nas últimas épocas.

Figura 3.3 – Valor do *fitness* em cada época da otimização por meio do PSO para o projeto do PID para o conversor *Buck*.



Fonte: Elaborado pelo Autor.

Passo 6: Análise de estabilidade robusta

O valor final da função objetivo, encontrado para o vetor de ganhos de controle (3.26) foi de $f(\mathbf{c}^*) = 0,2065$, indicando que $\gamma(\mathbf{c}^*) = 1$ (i.e., sistema *KT estável*). Para confirmar a estabilidade robusta do sistema em malha fechada com o controlador fornecido pelo PSO para toda a região de incertezas paramétricas, os quatro polinômios do Teorema de Kharitonov, para este exemplo, são dados por

$$\begin{aligned}
 K_1(s) &= 1984706171192,70 + 204150332,76s + 17350,91s^2 + s^3 \\
 K_2(s) &= 1984706171192,70 + 218907951,85s + 17350,91s^2 + s^3 \\
 K_3(s) &= 2425751987013,29 + 204150332,76s + 20807,77s^2 + s^3 \\
 K_4(s) &= 2425751987013,29 + 218907951,85s + 20807,77s^2 + s^3
 \end{aligned} \tag{3.27}$$

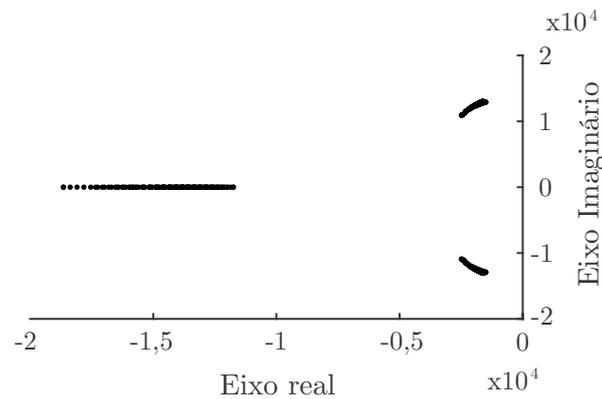
e têm as raízes com parte real negativa (i.e., são polinômios Hurwitz), dadas por

$$\begin{aligned}
 r_1 &= -13248,8, \quad r_2 = -2051 + j12066,3, \quad r_3 = -2051 - j12066,3, \quad \text{para } K_1(s) \\
 r_1 &= -12524,8, \quad r_2 = -2413 + j12354,7, \quad r_3 = -2413 - j12354,7, \quad \text{para } K_2(s) \\
 r_1 &= -17151,1, \quad r_2 = -1828 + j11751,2, \quad r_3 = -1828 - j11751,2, \quad \text{para } K_3(s) \\
 r_1 &= -16461,3, \quad r_2 = -2173 + j11943,1, \quad r_3 = -2173 - j11943,1, \quad \text{para } K_4(s)
 \end{aligned}
 \tag{3.28}$$

Isso confirma que o sistema em malha fechada é *KT estável*.

A Figura 3.4 mostra os polos do sistema em malha fechada para uma discretização fina (passo de 0,01) nos parâmetros R_{Lb} e E_b , sendo possível observar que todos os polos se localizam no semiplano esquerdo do plano complexo, corroborando a estabilidade em malha fechada.

Figura 3.4 – Polos do sistema em malha fechada localizados no semiplano esquerdo considerando uma discretização fina nos valores dos parâmetros incertos do conversor *Buck*, confirmando a estabilidade do sistema controlado.



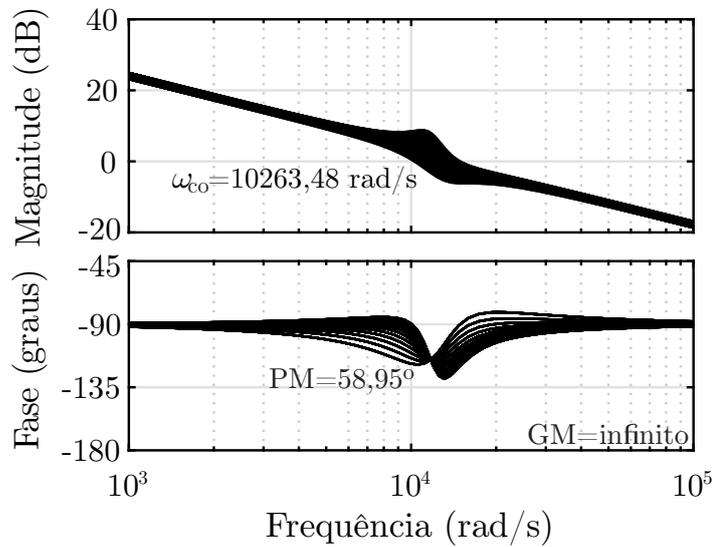
Fonte: Elaborado pelo Autor.

Passo 7: Análise de desempenho robusto

O valor final da função objetivo $f(\mathbf{c}^*) = 0,2065$ também confirma que $\beta(\mathbf{c}^*) = 1$, ou seja, todos os limites de desempenho foram atendidos para os vértices do politopo.

Para comprovar o desempenho do controlador fornecido pelo PSO para toda a faixa de incertezas, a Figura 3.5 apresenta os diagramas de Bode do sistema em malha aberta $G(s)G_c(s)$, para uma discretização fina nos parâmetros incertos. Note que os valores de pior caso para a frequência de cruzamento, margem de fase e margem de ganho são os mesmos encontrados ao analisar apenas os vértices do modelo politópico, como visto na Tabela 3.4, indicando a viabilidade das estimativas dos índices de desempenho por meio dos vértices do modelo politópico, no procedimento proposto.

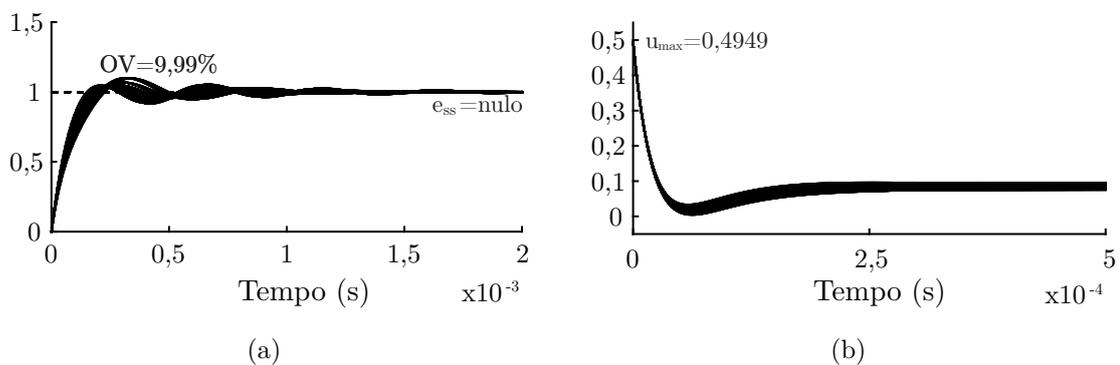
Figura 3.5 – Diagramas de Bode do sistema em malha aberta para uma discretização fina nos valores dos parâmetros incertos do conversor *Buck*.



Fonte: Elaborado pelo Autor.

A Figura 3.6(a) apresenta a resposta ao degrau unitário do sistema em malha fechada e a Figura 3.6(b) mostra o sinal de controle u correspondente, para uma discretização fina nos parâmetros incertos.² Assim como constatado para a simulação na Figura 3.5, os valores de pior caso nas Figuras 3.6(a) e (b) são os mesmos apresentados na Tabela 3.4, novamente confirmando a viabilidade de se utilizar os vértices do modelo politópico para as estimativas de desempenho.

Figura 3.6 – Respostas no domínio do tempo, considerando uma discretização fina nos valores dos parâmetros incertos do conversor *Buck*: (a) respostas ao degrau unitário do sistema em malha fechada; (b) sinais de controle correspondentes.



Fonte: Elaborado pelo Autor.

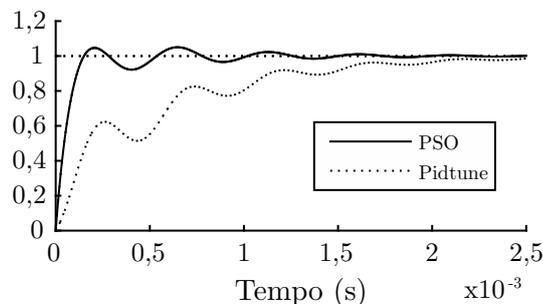
Para uma comparação do controlador (3.26), projetado pelo procedimento pro-

²Para esta simulação, um polo adicional foi fixado próximo a frequência $2\pi f_s/2 \approx 94248 \text{ rad/s}$, para tornar implementável (causal) a função de transferência do controlador em (3.23).

posto, com um controlador obtido por um método estabelecido na literatura, foi utilizada a função *pidtune*, do MATLAB, para fornecer um controlador PID para uma planta nominal, com as condições de projeto ω_{co}^* e PM^* iguais às utilizadas na Tabela 3.2. A planta nominal escolhida foi o caso do modelo com valores máximos de E_b e R_{Lb} , devido ao fato de ser a representação da planta que apresenta o menor fator de amortecimento.

A Figura 3.7 mostra a resposta ao degrau unitário do sistema em malha fechada comparando o controlador obtido com o procedimento proposto, com ganhos em (3.26), com o controlador obtido pela função *pidtune*.³ Pode-se observar que o sistema controlado pelos ganhos obtidos pelo PSO produz uma melhor resposta dinâmica, com tempo de subida e tempo de acomodação significativamente menores.

Figura 3.7 – Comparação das respostas ao degrau unitário do sistema em malha fechada para o controlador baseado em PSO e para o controlador projetado com a função *pidtune* do MATLAB.



Fonte: Elaborado pelo Autor.

Cabe observar também que o controlador obtido pela função *pidtune* do MATLAB não possui um certificado de robustez frente a incertezas paramétricas, pois o mesmo é projetado para apenas uma planta nominal (neste caso, um dos vértices de todo o modelo politópico). O controlador projetado pelo PSO apresenta como vantagens, além de um melhor desempenho, a estabilidade robusta certificada pelo Teorema de Kharitonov, para toda faixa de incertezas nos parâmetros da planta.

3.8 CONSIDERAÇÕES FINAIS

Este capítulo apresentou o procedimento de projeto proposto para obtenção de controladores robustos frente a incertezas paramétricas, para conversores de potência descritos por função de transferência com coeficientes intervalares. Foram propostos: i) uma função objetivo que orienta a busca dos ganhos de controle pela avaliação dos vértices de um modelo politópico da planta, ii) uma maneira sistemática para definir o espaço de busca dos ganhos de controle e iii) a utilização do PSO para buscar os ganhos de

³Ganhos fornecidos pela função *pidtune*: $K_p = 0,0237$, $K_i = 154$, $K_d = 9,09 \times 10^{-7}$.

controle e do Teorema de Kharitonov, para prover um certificado teórico de estabilidade robusta durante o projeto do controlador. Um exemplo é apresentado, a fim de mostrar, passo a passo, um projeto utilizando o procedimento proposto, em que um controlador PID é sintetizado para a regulação da tensão de saída de um conversor *Buck*. Simulações mostraram bom desempenho e robustez frente a incertezas paramétricas com o controlador projetado pelo procedimento proposto, que é capaz de fornecer resultados superiores aos da função *pidtune* do MATLAB, especializada no projeto de PIDs.

4 VALIDAÇÃO EXPERIMENTAL DO PROCEDIMENTO

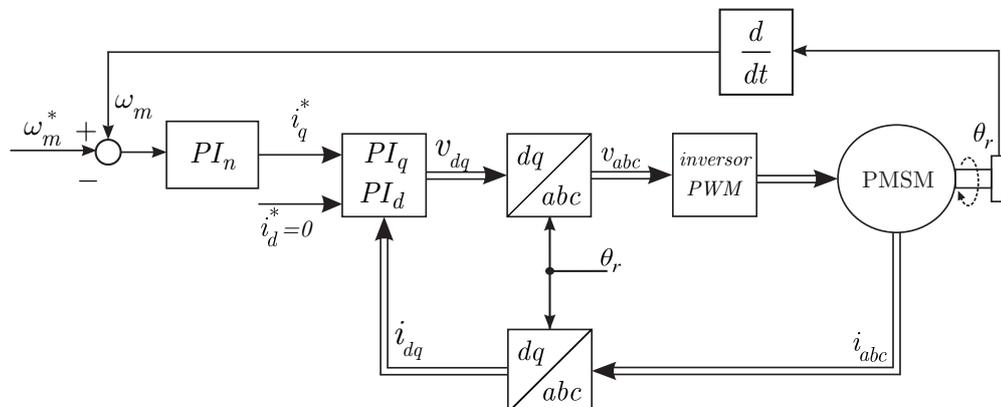
Este capítulo apresenta a validação experimental do procedimento de projeto proposto no Capítulo 3 para uma aplicação ao controle de velocidade de um motor síncrono de ímãs permanentes (do inglês, *permanent magnet synchronous motor* – PMSM). PMSMs são atrativos em vários cenários, como acionamento em robôs industriais, em veículos elétricos, em sistemas eólicos de conversão de energia, entre outros, sendo o controle robusto a incertezas nos parâmetros elétricos e mecânicos um ponto de interesse [Krause, Wasynczuk e Sudhoff 2002].

Neste sentido, o objetivo deste capítulo é projetar controladores PIs robustos para o controle de corrente e de velocidade de um PMSM em uma estratégia orientada pelo campo (do inglês, *field oriented control* – FOC). Assumindo que o motor pode ser representado, de forma simplificada, a partir de modelos lineares de primeira ordem, utiliza-se o procedimento proposto, levando a resultados que mostram boa regulação de velocidade de um PMSM em um cenário de implementação prática.

4.1 INFORMAÇÕES PRELIMINARES

Considere o diagrama de blocos do sistema de controle de um PMSM, dado na Figura 4.1, em que uma estratégia FOC é utilizada [Pillay e Krishnan 1989]. Note que devem ser projetados dois PIs para as malhas internas das correntes em coordenadas dq , chamados PI_d e PI_q , e um PI para a malha de controle externa de velocidade mecânica do motor, chamado de PI_n [Krause, Wasynczuk e Sudhoff 2002, Koch et al. 2018].

Figura 4.1 – PMSM: diagrama de blocos do sistema de controle, composto por três controladores PIs.



Fonte: Adaptado de [Gabbi et al. 2019].

Na Figura 4.1, um compensador de velocidade calcula o erro entre a referência de velocidade ω_m^* e a velocidade real ω_m , para gerar a referência de corrente i_q^* . A estratégia de controle impõe que $i_d^* = 0$, linearizando o torque eletromecânico. Sensores de corrente fornecem as correntes de fase i_{abc} , que são posteriormente convertidas em correntes em referencial síncrono, em coordenadas dq , com base na posição angular θ_r , utilizando a transformação dada em [Krause, Wasynczuk e Sudhoff 2002].

Esta estratégia de controle visa desacoplar o controle de torque e de fluxo. Assim, a corrente i_q^* gerada na saída do controle de velocidade é utilizada para controlar o torque de maneira linear, enquanto a corrente i_d^* é utilizada para controlar o fluxo magnético do motor.

As saídas dos controladores PI_d e PI_q fornecem as tensões v_d e v_q , que são transformadas em tensões em coordenadas abc e convertidas em sinais para a modulação por largura de pulso (PWM), a fim de acionar o PMSM.

O modelo do PMSM no sistema de referência síncrono pode ser dado por [Krishnan 2001]

$$\frac{di_d}{dt} = -\frac{R_s}{L_d}i_d + \frac{L_q}{L_d}\omega_e i_q + \frac{1}{L_d}v_d \quad (4.1)$$

$$\frac{di_q}{dt} = -\frac{R_s}{L_q}i_q - \frac{L_d}{L_q}\omega_e i_d - \frac{\phi_{srm}}{L_q}\omega_e + \frac{1}{L_q}v_q \quad (4.2)$$

em que os subíndices d e q indicam os eixos direto e em quadratura; i_d , i_q , v_d e v_q são as correntes e tensões estatóricas nos eixos dq girando a uma velocidade síncrona; L_d e L_q são as indutâncias dos eixos direto e em quadratura; R_s é a resistência estatórica; ϕ_{srm} é o fluxo magnético dos ímãs e ω_e é a velocidade angular elétrica. Observe que os parâmetros referentes às indutâncias são diferentes para os eixos d e q , devido à configuração magnética interna da máquina.

Considerando os termos dependentes de ω_e em (4.1) e (4.2) como distúrbios e aplicando a transformada de Laplace em (4.1), obtém-se a seguinte função de transferência para a malha de corrente no eixo direto

$$G_d(s) = \frac{i_d(s)}{v_d(s)} = \frac{1}{s + \frac{R_s}{L_d}} \quad (4.3)$$

De mesmo modo, ao aplicar a transformada de Laplace em (4.2), obtém-se a seguinte função de transferência para a malha de corrente no eixo em quadratura

$$G_q(s) = \frac{i_q(s)}{v_q(s)} = \frac{1}{s + \frac{R_s}{L_q}} \quad (4.4)$$

O modelo dinâmico para o comportamento mecânico é descrito por [Krishnan 2001]

$$\frac{dw_m}{dt} = \frac{1}{J} (\tau_m - Bw_m) + \delta(w_m) \quad (4.5)$$

em que J é o momento de inércia do rotor, B é o coeficiente de atrito viscoso e $\delta(w_m)$ representa um termo não linear de norma limitada. τ_m é a diferença entre o torque eletromecânico, dado por $\frac{3}{2}P[(L_d - L_q)i_d + \phi_{srm}]i_q$, e o torque de carga, sendo P o número de pares de polos do motor.

Para obter a função de transferência da parte linear de (4.5), ao aplicar a transformada de Laplace, considerando o termo $\delta(w_m)$ como um distúrbio, o torque mecânico como entrada e a velocidade do rotor como saída, a função de transferência para a planta da malha de velocidade pode ser representada por

$$G_n(s) = \frac{\omega_m(s)}{\tau_m(s)} = \frac{\frac{1}{J}}{s + \frac{B}{J}} \quad (4.6)$$

No caso de incertezas paramétricas, note que os modelos em (4.3), (4.4) e (4.6) são compatíveis com o modelo de função de transferência apresentado no procedimento proposto em (3.1).

Para um estudo de caso, considere os parâmetros na Tabela 4.1, dados em [Koch et al. 2018]. Adicionalmente, as incertezas paramétricas são dadas por Δ_1 e Δ_2 , representando, respectivamente, desvios de 40% e 10% em relação aos valores nominais.

Tabela 4.1 – Parâmetros da planta (PMSM).

| Descrição do sistema | |
|--|---|
| Potência nominal | 11 kW |
| Corrente nominal | 19,2 A |
| Torque nominal | 58,4 Nm |
| Polos (P) | 6 |
| Resistência estatórica (R_s) | 0,475 $\Omega \pm \Delta_1$ |
| Indutância do eixo direto (L_d) | 20,1 mH $\pm \Delta_2$ |
| Indutância do eixo em quadratura (L_q) | 40,9 mH $\pm \Delta_2$ |
| Momento de inércia (J) | 0,03877 kgm ² $\pm \Delta_2$ |
| Coeficiente de atrito viscoso (B) | 0,0194 Nms $\pm \Delta_1$ |
| Máximo fluxo magnético dos imãs | 0,5126 V/rad/s |
| Frequência de comutação (f_s) | 10000 Hz |

4.2 APLICAÇÃO DO PROCEDIMENTO DE PROJETO PROPOSTO AO PMSM

Passo 1: Modelagem com parâmetros incertos

Primeiramente, a partir dos modelos em (4.3), (4.4) e (4.6) e dos parâmetros dados na Tabela 4.1, as funções de transferência podem ser representadas por meio do modelo dado em (3.1), descrito como

$$G(s) = \frac{f_0(p_1)}{s + g_0(p_1, p_2)} \quad (4.7)$$

Nesta representação, para a planta da malha de corrente no eixo direto ($G_d(s)$, em (4.3)), tem-se

$$f_0(p_1) = \frac{1}{L_d}, \quad g_0(p_1, p_2) = \frac{R_s}{L_d}, \quad p_1 = L_d, \quad p_2 = R_s \quad \text{para a malha de } i_d \quad (4.8)$$

Para a planta da malha de corrente no eixo em quadratura ($G_q(s)$, em (4.4)), tem-se

$$f_0(p_1) = \frac{1}{L_q}, \quad g_0(p_1, p_2) = \frac{R_s}{L_q}, \quad p_1 = L_q, \quad p_2 = R_s \quad \text{para a malha de } i_q \quad (4.9)$$

E para a planta de velocidade ($G_n(s)$, em (4.6)), tem-se

$$f_0(p_1) = \frac{1}{J}, \quad g_0(p_1, p_2) = \frac{B}{J}, \quad p_1 = J, \quad p_2 = B, \quad \text{para a malha de } \omega_m \quad (4.10)$$

Passo 2: Escolha do controlador

Como visto na Figura 4.1, serão utilizados três controladores PIs (PI_d , PI_q e PI_n), a serem projetados pelo procedimento proposto, provendo robustez contra as incertezas paramétricas no motor dadas na Tabela 4.1. Para aplicação do procedimento proposto, os PIs serão sintonizados de forma independente entre si, por meio da escolha de bandas passantes significativamente diferentes das malhas elétricas e mecânica.

Controladores PIs robustos provaram ser suficientes em trabalhos anteriores no controle de PMSMs considerando incertezas paramétricas, como em [Jung et al. 2011, Koch et al. 2018, Gabbi et al. 2019]. Em [Koch et al. 2018, Gabbi et al. 2019], os PIs robustos são projetados por meio de LMIs, enquanto em [Jung et al. 2011], utiliza-se o Teorema de Kharitonov para obter regiões de estabilidade robusta para o projeto de controladores de corrente PI do tipo *fuzzy*. Um ponto comum nesses trabalhos é a não utilização simultânea de especificações pertencentes ao domínio do tempo e da frequência para o projeto dos controladores, o que é uma das características do procedimento aqui proposto.

Passo 3: Especificações da função objetivo

As especificações de projeto (valores de referência e limitantes) são dados na Tabela 4.2.

Tabela 4.2 – Especificações da função objetivo para o projeto dos controladores PIs para as malhas elétricas e mecânica do PMSM.

| Valores de referência para $\alpha(\mathbf{c})$ | malhas de corrente | malha de velocidade |
|---|---------------------|---------------------|
| Frequência de cruzamento (ω_{co}^*) | 400 rad/s (63,7 Hz) | 60 rad/s (9,5 Hz) |
| Margem de fase (PM*) | 60° | 60° |
| Valores limitantes para $\beta(\mathbf{c})$ | | |
| Margem de ganho (\underline{GM}) | 5 (14 dB) | 5 (14 dB) |
| <i>Overshoot</i> (\overline{OV}) | 10% | 10% |
| Erro em regime permanente ($\overline{e_{ss}}$) | 1% | 1% |
| Amplitude do sinal de controle (\overline{u}) | 17 | 1 |

O valor de referência da frequência de cruzamento para os controladores de corrente possui a mesma ordem de grandeza dos utilizados em [Koch et al. 2018, Gabbi et al. 2019], que utilizam os mesmos parâmetros da planta na Tabela 4.1 e uma implementação do controle em um processador digital de sinais. O limite da amplitude do sinal de controle \overline{u} para as malhas de corrente foi escolhido abaixo do valor da corrente nominal do sistema (ver Tabela 4.1), para obter uma margem de segurança para o valor desta variável. As demais especificações são valores típicos que podem ser encontrados na literatura [Erickson 1997, Dorf e Bishop 2011].

Passo 4: Definição do espaço de busca

Para as plantas em (4.7)–(4.10) e controladores PIs dados por (3.6), os polinômios característicos do sistema em malha fechada para $G_d(s)$, $G_q(s)$ e $G_n(s)$ são dados, respectivamente, por

$$\begin{aligned}
 D(s) &= s^2 + \left(\frac{R_s}{L_d} + \frac{1}{L_d} K_p \right) s + \frac{1}{L_d} K_i, \text{ para a malha de } i_d \\
 D(s) &= s^2 + \left(\frac{R_s}{L_q} + \frac{1}{L_q} K_p \right) s + \frac{1}{L_q} K_i, \text{ para a malha de } i_q \\
 D(s) &= s^2 + \left(\frac{B}{J} + \frac{1}{J} K_p \right) s + \frac{1}{J} K_i, \text{ para a malha de } \omega_m
 \end{aligned} \tag{4.11}$$

Utilizando a condição necessária de estabilidade do critério de Routh, da positiv-

dade dos coeficientes de (4.11), tem-se que os espaços de busca de cada PI são dados por

$$\begin{aligned}
 K_i > 0 \quad K_p > -R_s \quad , \quad & \text{para a malha de } i_d \\
 K_i > 0 \quad K_p > -R_s \quad , \quad & \text{para a malha de } i_q \\
 K_i > 0 \quad K_p > -B \quad , \quad & \text{para a malha de } \omega_m
 \end{aligned} \tag{4.12}$$

Por razões computacionais, foi utilizado o valor 10^4 como limite superior dos parâmetros de (4.12), tornando o espaço de busca dos ganhos suficientemente amplo.

Passo 5: Configuração e execução do PSO

A configuração utilizada para o PSO neste estudo de caso, para os projetos dos três PIs, é dada na Tabela 4.3.

Tabela 4.3 – Configuração do PSO para o projeto dos três controladores PIs para o PMSM.

| Parâmetros do PSO | |
|------------------------------------|-----|
| Número de partículas (N) | 200 |
| Número de épocas (M) | 50 |
| Coeficiente cognitivo (ϕ_1) | 0,5 |
| Coeficiente social (ϕ_2) | 0,5 |

Para uma análise estatística, para cada PI (i.e., PI_d , PI_q e PI_n), o algoritmo PSO foi executado 30 vezes [Arcuri e Briand 2011], todos de maneira *off-line* e independentes entre si, com um tempo médio de execução de 9 minutos.

Uma execução típica para o projeto do PI da malha de corrente direta resultou nos seguintes ganhos de controle

$$\mathbf{c}^* = [x_0 \ x_1] = [508, 328174521324 \quad 7, 82729852927564] = [K_i \ K_p] \quad , \quad \text{para } PI_d \tag{4.13}$$

Da mesma forma, uma execução típica para o projeto do PI da malha de corrente em quadratura resultou nos seguintes ganhos de controle

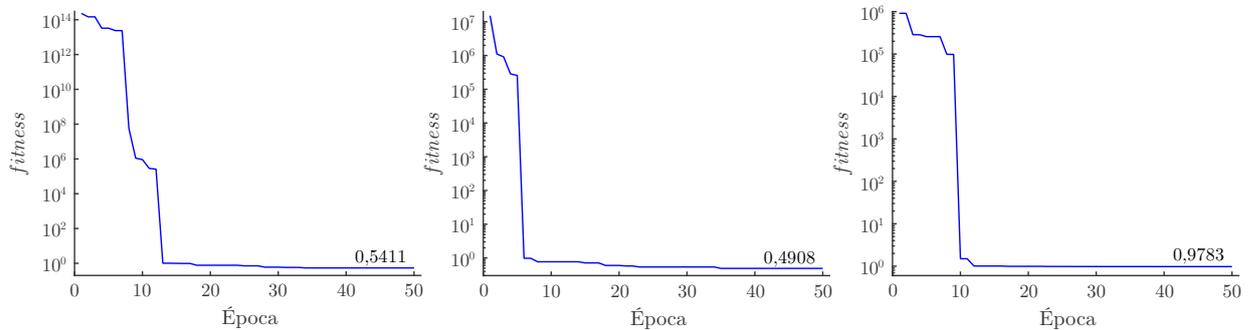
$$\mathbf{c}^* = [x_0 \ x_1] = [1001, 42582632092 \quad 15, 99450844258434] = [K_i \ K_p] \quad , \quad \text{para } PI_q \tag{4.14}$$

Por fim, uma execução típica para o projeto do PI da malha de velocidade resultou nos seguintes ganhos de controle

$$\mathbf{c}^* = [x_0 \ x_1] = [4, 01693568550617 \quad 0, 981429192074386] = [K_i \ K_p] \quad , \quad \text{para } PI_n \tag{4.15}$$

Para as execuções do PSO que resultaram nos ganhos em (4.13)–(4.15), a Figura 4.2 apresenta a evolução do *fitness* no decorrer das épocas.

Figura 4.2 – Valor do *fitness* em cada época da otimização por meio do PSO para o projeto dos PIs para as malhas de controle de corrente direta, $i_d(s)$, de corrente em quadratura, $i_q(s)$ e de velocidade, $\omega_m(s)$, respectivamente.

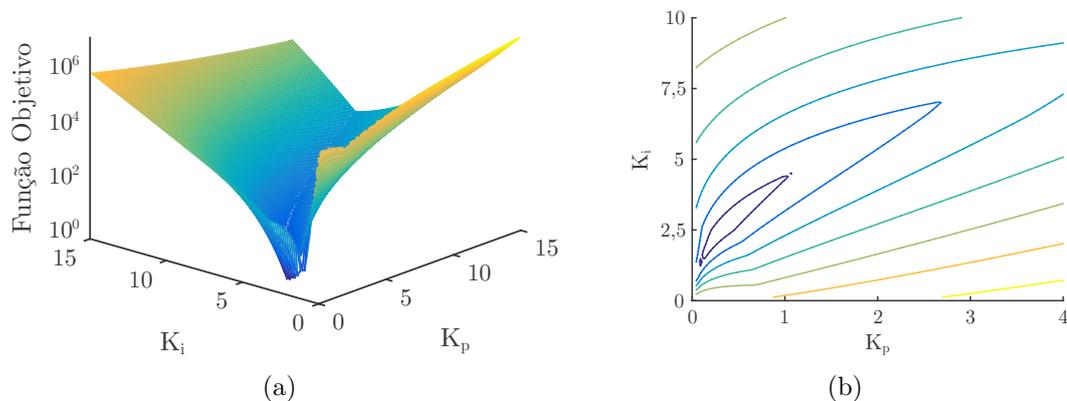


Fonte: Elaborado pelo Autor.

Pode-se observar, na Figura 4.2, pela magnitude do valor final de cada *fitness*, que $\beta(\mathbf{c}^*)$ deve ser igual a 1, indicando conformidade com os limites em (3.14), e que $\gamma(\mathbf{c}^*)$ também deve ser igual a 1, indicando que o sistema em malha fechada é *KT estável*.

Considerando como exemplo o projeto dos ganhos K_p e K_i para a malha de velocidade, a Figura 4.3(a) mostra a função objetivo proposta, para uma discretização fina do espaço de busca dos ganhos de controle. A Figura 4.3(b) apresenta as curvas de nível da função da Figura 4.3(a), mostrando a região em que o ponto de mínimo está localizado.

Figura 4.3 – Análises da função objetivo para o projeto de controle da malha de velocidade: (a) parte do espaço de busca dado por $0 \leq K_p \leq 15$ e $0 \leq K_i \leq 15$ que apresenta o comportamento da função objetivo em uma região que inclui o ganho (4.15); (b) curvas de nível da função em (a).



Fonte: Elaborado pelo Autor.

Cabe mencionar que a área mostrada na Figura 4.3(a) corresponde a apenas 0,01% de todo o espaço de busca dos controladores considerado em (4.12). As Figuras 4.3(a) e (b) foram obtidas por meio de uma discretização fina considerando 10000 pontos, igualmente

distribuídos no espaço de busca. Esta representação gráfica da função objetivo demandou cerca de duas horas para ser realizada.¹ Pode-se estimar então que, uma varredura levando em consideração o espaço de busca completo demandaria aproximadamente 20000 horas para ser realizada, o que indica a inviabilidade prática da obtenção dos ganhos ótimos de controle por um projeto por busca exaustiva.

Nesse sentido, a busca realizada pelo PSO, demandando 9 minutos, é justificada, e o procedimento proposto fornece uma vantagem em relação a procedimentos exaustivos e também sobre procedimentos que usam espaços de busca menores, mas que podem não incluir o ótimo global.

Passo 6: Análise de estabilidade robusta

Para comprovar a estabilidade robusta dos controladores fornecidos pelo PSO para toda faixa de incertezas paramétricas, os quatro polinômios do Teorema de Kharitonov, considerando a planta $G_n(s)$ com coeficientes intervalares, com os ganhos em (4.15), são dados por

$$\begin{aligned} K_1(s) &= 93,23 + 23,02s + s^2 \\ K_2(s) &= 93,23 + 28,59s + s^2 \\ K_3(s) &= 113,96 + 23,02s + s^2 \\ K_4(s) &= 113,96 + 28,59s + s^2 \end{aligned} \quad (4.16)$$

tendo todas as raízes com parte real negativa, dadas por

$$\begin{aligned} r_1 &= -17,78, \quad r_2 = -5,24, \quad \text{para } K_1(s) \\ r_1 &= -24,84, \quad r_2 = -3,75, \quad \text{para } K_2(s) \\ r_1 &= -15,83, \quad r_2 = -7,19, \quad \text{para } K_3(s) \\ r_1 &= -23,80, \quad r_2 = -4,78, \quad \text{para } K_4(s) \end{aligned} \quad (4.17)$$

o que certifica que o sistema em malha fechada é *KT estável*.

As mesmas análises foram feitas para as malhas de corrente e, para ambas, foi comprovada a estabilidade robusta do sistema em malha fechada com os controladores (i.e., o sistema em malha fechada é *KT estável* para cada uma das malhas).

Passo 7: Análise de desempenho robusto

A Tabela 4.4 apresenta os valores das medidas de desempenho para os vértices do modelo politópico da planta, para a malha de velocidade.

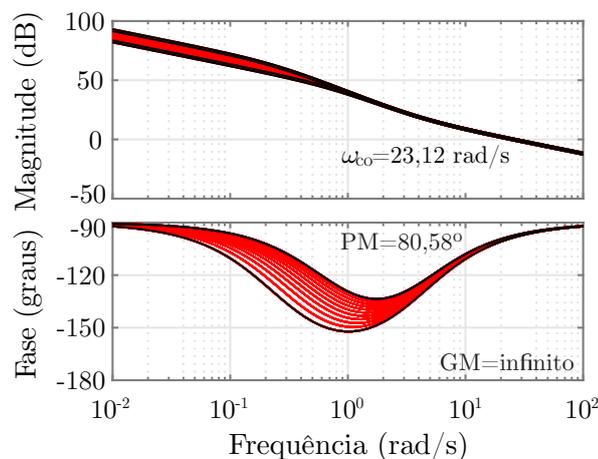
¹Simulação feita em um notebook, com processador core i7 x64 1.8 GHz, com 16 GB RAM.

Tabela 4.4 – Índices medidos nos vértices do modelo politópico, para a malha de velocidade.

| | Valores nos vértices de $G_n(s)$ com (4.15) | | | | Valores de pior caso |
|-----------------------|---|--------------------------|--------------------------|--------------------|----------------------|
| | $\underline{B}, \underline{J}$ | \underline{B}, \bar{J} | \bar{B}, \underline{J} | \bar{B}, \bar{J} | |
| ω_{co} (rad/s) | 23,13 | 28,13 | 23,12 | 28,13 | 23,12 |
| PM (graus) | 80,58 | 82,23 | 81,82 | 83,25 | 80,58 |
| GM (dB) | ∞ | ∞ | ∞ | ∞ | ∞ |
| OV | 9,96% | 8,64% | 8,38% | 7,30% | 9,96% |
| e_{ss} | Nulo | Nulo | Nulo | Nulo | Nulo |
| \bar{u} | 0,98 | 0,98 | 0,98 | 0,98 | 0,98 |

A malha de controle de velocidade foi simulada, com os ganhos de controle em (4.15), e as respostas no domínio da frequência da função de transferência em malha aberta são apresentadas na Figura 4.4, para uma discretização fina nos parâmetros incertos.

Figura 4.4 – Diagramas de Bode em malha aberta com os ganhos (4.15) para a malha de velocidade.

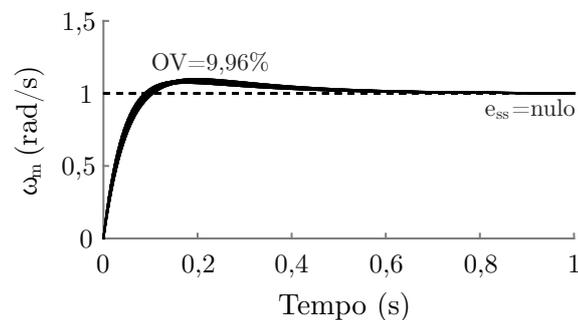


Fonte: Elaborado pelo Autor.

Os valores de pior caso obtidos para margem de fase e frequência de cruzamento na Figura 4.4 foram $PM = 80,58^\circ$ e $\omega_{co} = 23,12$ rad/s. Note que estes valores estão presentes na Tabela 4.4, indicando a viabilidade da estimativa feita nos vértices do modelo politópico.

As respostas ao degrau unitário do sistema em malha fechada, para a mesma discretização fina nos parâmetros, são mostradas na Figura 4.5, em que se pode observar o erro nulo em regime permanente e que o valor de pior caso medido para o *overshoot* é 9,96%. O valor de pior caso medido para a amplitude do sinal de controle também obedece à restrição imposta, novamente indicando a viabilidade do uso do modelo politópico para estimativas de desempenho.

Figura 4.5 – Resposta ao degrau unitário do sistema em malha fechada, com os ganhos (4.15), para a malha de velocidade.



Fonte: Elaborado pelo Autor.

4.3 ANÁLISES COMPARATIVAS

Para análises comparativas, a Tabela 4.5 mostra as medidas de desempenho para o controlador PI projetado para a malha de velocidade, baseado no procedimento de projeto proposto, identificado aqui como (PSO), e de outros controladores PIs obtidos por:

- um algoritmo genético, usado por meio da função *ga* do MATLAB, e que utiliza como função objetivo a função proposta em (3.12), cujos resultados são identificados aqui como GA [Haupt e Haupt 2004];
- uma aplicação da função *pidtune*, do MATLAB;
- um algoritmo PSO, que utiliza como função objetivo o critério ITAE, conforme apresentado em [Banerjee, Ghosh e Rana 2017], identificado aqui como PSO-ITAE.

Note que, das três técnicas escolhidas para a análise comparativa, a primeira tem como objetivo mostrar a qualidade dos resultados obtidos ao utilizar a função objetivo proposta aqui em uma outra meta-heurística similar ao PSO, a segunda tem como objetivo mostrar a qualidade dos resultados obtidos a partir de uma função do MATLAB especializada no projeto de PIDs, e a terceira tem como objetivo mostrar a qualidade dos resultados ao empregar o PSO com outra função objetivo utilizada na literatura, que, neste caso, busca uma otimização da resposta ao degrau e sem restrições, o que pode gerar resultados que demandam um esforço proibitivo na ação de controle.

As medidas dadas na Tabela 4.5 são os valores de pior caso obtidos com base nas simulações no MATLAB, para os vértices do modelo politópico. Para os controladores obtidos a partir do *pidtune* e do PSO-ITAE, que foram projetados utilizando apenas uma planta nominal, as medidas da tabela também foram realizadas ao analisar os vértices do modelo politópico da planta.

Tabela 4.5 – Comparação dos PIs projetados para a malha de velocidade, mostrando os piores casos das medidas obtidas para os vértices do modelo politópico.

| | PSO | GA | PSO-ITAE | <i>pidtune</i> |
|-----------------------|----------|----------|--------------------|----------------|
| PM (graus) | 80,58 | 82,7 | 90 | 60 |
| ω_{co} (rad/s) | 23,12 | 22,09 | $2,83 \times 10^5$ | 9,55 |
| GM (dB) | ∞ | ∞ | ∞ | ∞ |
| OV | 9,96% | 8,14% | 0,0% | 23,4% |
| e_{ss} | Nulo | Nulo | Nulo | Nulo |
| Saturação | Não | Não | Sim | Não |
| KT estável | Sim | Sim | Não | Não |
| K_p | 0,981 | 0,942 | 9999,9 | 0,351 |
| K_i | 4,017 | 2,894 | 9980,2 | 2,05 |

Pode-se observar que os resultados obtidos com o procedimento proposto (primeira coluna) e com o GA (segunda coluna) são semelhantes. Para a execução do algoritmo genético, foi utilizada a função *ga* do MATLAB, com a seguinte configuração: população = 50 cromossomos; gerações = 100; taxa de cruzamento = 0,8; taxa de mutação = 0,2; método de seleção = *selectiontournament*; método de cruzamento = *crossoversinglepoint*; método de mutação = *mutationadaptfeasible* [Haupt e Haupt 2004].

Embora os resultados obtidos com o controlador sintonizado com o PSO e com o GA sejam similares, o PSO destaca-se em relação ao GA pelas seguintes características:

- o PSO tem uma configuração mais simples que o GA;
- o tempo de execução do PSO é aproximadamente metade do tempo exigido pelo GA, para condições semelhantes de projeto (i.e., para o mesmo número de partículas para o PSO e de cromossomos para o GA, e para o mesmo número de épocas para o PSO e gerações para o GA);
- definindo-se como sucesso um caso em que o algoritmo obtém um controlador \mathbf{c}^* que possui as propriedades de $\beta(\mathbf{c}^*) = 1$ e $\gamma(\mathbf{c}^*) = 1$, tem-se que, em 30 execuções dos algoritmos, a taxa de sucesso com o PSO é de 90%, enquanto com o GA é de apenas 40%, neste estudo de caso;
- os resultados do PSO apresentam menor dispersão, definindo-se a dispersão como o desvio padrão do valor de $f(\mathbf{c}^*)$ dividido pelo seu valor médio, para as execuções de sucesso, sendo de 1,29% para a aplicação do PMSM para a malha de velocidade, enquanto a dispersão com o GA foi de 4,52%. Além disso, os valores dos ganhos de controle obtidos pelo procedimento proposto também apresentam baixa dispersão entre as execuções do PSO.

Ao comparar o desempenho do controlador obtido pelo PSO (primeira coluna) com o dos controladores na terceira e quarta coluna (PSO-ITAE e *pidtune*, respectivamente),

pode-se observar que:

- os controladores fornecidos por *pidtune* e PSO-ITAE não garantem que o sistema em malha fechada é *KT estável*, e podem levar à saturação do controle (caso do PSO-ITAE), uma vez que esses métodos não levam em consideração, no projeto do controlador, limites no sinal de controle e um certificado de estabilidade robusta;
- o PSO-ITAE otimiza apenas a resposta no tempo, o que pode aumentar significativamente a frequência de cruzamento, tornando os controladores de resposta muito rápida, mas inviáveis para uma implementação prática em conversores de potência, devido às limitações do sinal de controle em amplitude e frequência;
- os controladores obtidos pela função *pidtune* levam a respostas aceitáveis, porém, estes sistemas possuem desempenho inferior em comparação com os controladores baseados no procedimento proposto, e também não possuem um certificado teórico de estabilidade robusta sob incertezas paramétricas.

A seção a seguir apresenta uma validação experimental dos controladores obtidos pelo procedimento de projeto proposto. Destaca-se que a validação experimental é fundamental, uma vez que, na prática, o sistema de controle pode ser afetado por ruídos, dinâmicas não modeladas e outros efeitos não previstos nas simulações.

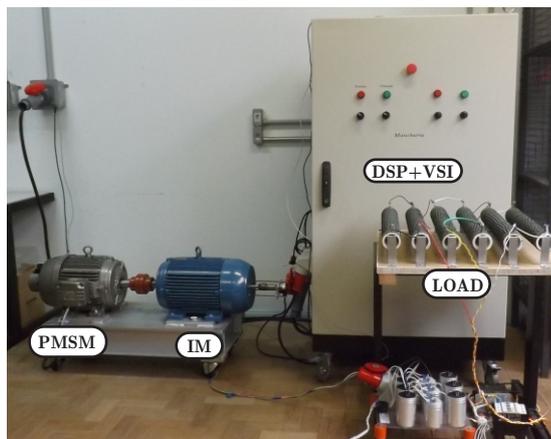
4.4 RESULTADOS EXPERIMENTAIS

Para obter os resultados experimentais, uma plataforma baseada em um PMSM comercial com os parâmetros descritos na Tabela 4.1 foi utilizada [Koch et al. 2018]. O processador digital de sinais (do inglês, *digital signal processor* – DSP) TMS320F28335, da Texas Instruments, foi usado para implementar digitalmente os controladores. O inversor trifásico fonte de tensão (do inglês, *voltage source inverter* – VSI) é baseado em chaves IGBT (do inglês, *insulated gate bipolar transistor*). Sensores de efeito Hall são utilizados para as medições das tensões e correntes. Um *encoder* absoluto TRD-NA256NWD fornece a posição real do rotor. Uma modulação por largura de pulso (do inglês, *pulse width modulation* – PWM) geométrica é adotada. A frequência de comutação utilizada foi de 10 kHz, o período de amostragem de 0,1 ms, e os controladores PIs em (4.13)–(4.15) são discretizados pelo método de Tustin [Ogata 1995].

A Figura 4.6 mostra a plataforma experimental, em que uma máquina de indução (do inglês, *induction machine* – IM) é acoplada ao eixo do PMSM. A carga elétrica mostrada na Figura 4.6 está conectada à IM, para produzir um distúrbio de carga mecânica de aproximadamente 25 Nm para o PMSM.

A Figura 4.7 mostra os resultados experimentais de um teste de rastreamento de referência, com os ganhos projetados pelo PSO em (4.13)–(4.15) e pela função *pidtune*

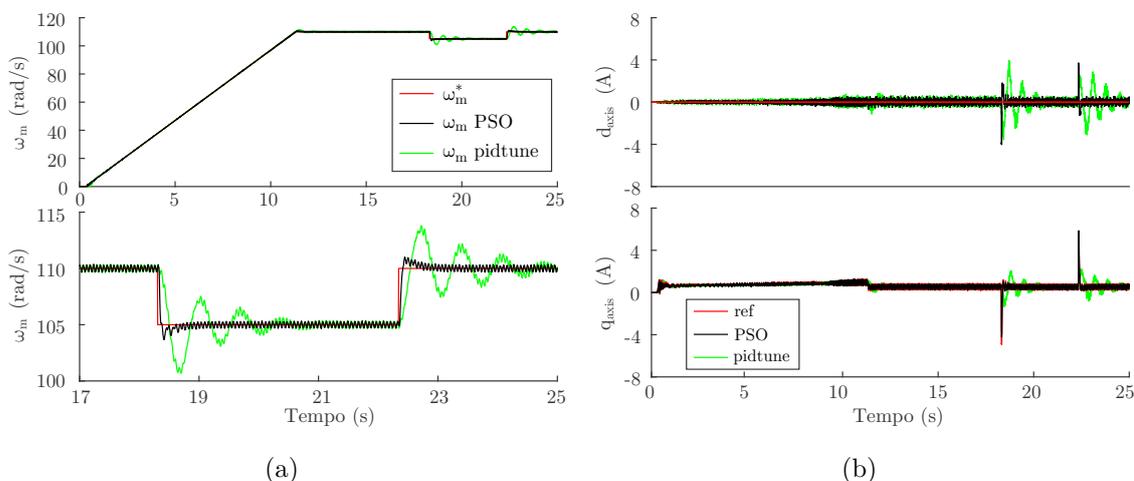
Figura 4.6 – PMSM: protótipo utilizado para obtenção dos resultados experimentais.



Fonte: Adaptado de [Koch et al. 2018].

do MATLAB.² A Figura 4.7(a)-superior apresenta a partida do sistema em rampa, seguido por variações na velocidade, que são detalhadas na Figura 4.7(a)-inferior, em que pode-se observar que o sistema controlado pelos ganhos do PSO possui um rastreamento de referência adequado e um desempenho dinâmico superior ao sistema controlado pelo *pidtune*. A Figura 4.7(b) mostra o comportamento das corrente nos eixos dq , destacando o desempenho superior dos controladores baseados no procedimento proposto em todos as malhas de controle neste ensaio.

Figura 4.7 – Resultados experimentais para rastreamento de referência com os controladores baseados no procedimento proposto e na função *pidtune* do MATLAB, sob variações de referência de velocidade: (a) respostas da malha de controle de velocidade (superior) e detalhamento na variação de referência (inferior); (b) malhas de controle das corrente no eixo d (superior) e no eixo q (inferior).

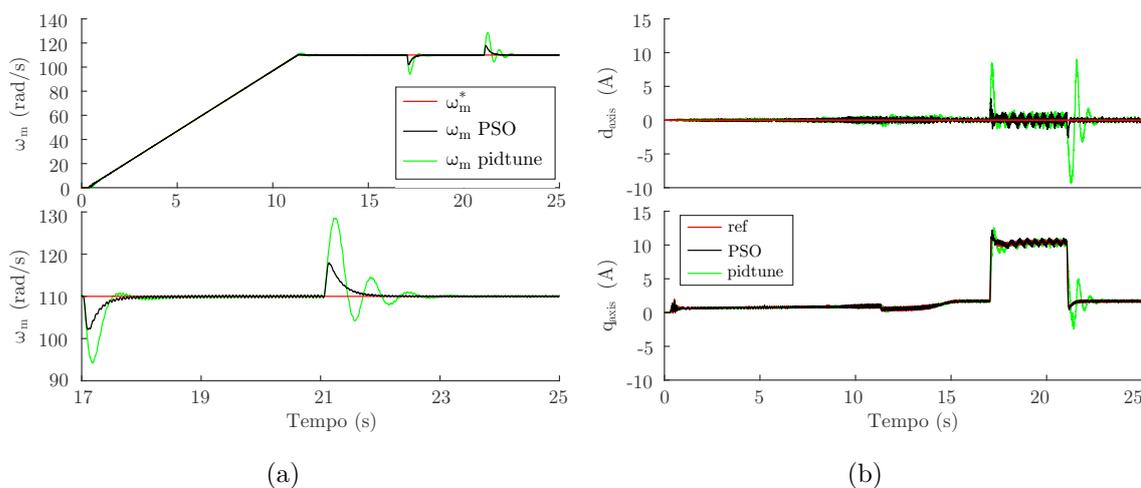


Fonte: Elaborado pelo Autor.

²Ganhos para a malha de velocidade contidos na Tabela 4.5, para a malha de corrente direta: $K_p = 1,09$ e $K_i = 60,6$; e para a malha de corrente em quadratura: $K_p = 2,37$ e $K_i = 107$.

A Figura 4.8 apresenta os resultados experimentais para rejeição de um distúrbio de carga mecânica, gerado pela IM acoplado ao eixo do PMSM, como visto no protótipo da Figura 4.6. O sistema inicia com uma rampa de referência de velocidade e, após atingir o valor de regime de 110 rad/s, são aplicados distúrbios de carga de aproximadamente 25 Nm. A Figura 4.8(a)-superior mostra as respostas da malha de velocidade, em que as variações geradas pelos distúrbios são detalhadas na Figura 4.8(a)-inferior. É possível notar uma boa rejeição de distúrbios no sistema controlado pelos ganhos obtidos com o procedimento proposto, com desempenho superior em comparação aos controladores obtidos utilizando a função *pidtune*. A Figura 4.8(b) mostra as respostas das malhas de corrente, destacando o desempenho superior dos controladores baseados no procedimento proposto em todos as malhas de controle.

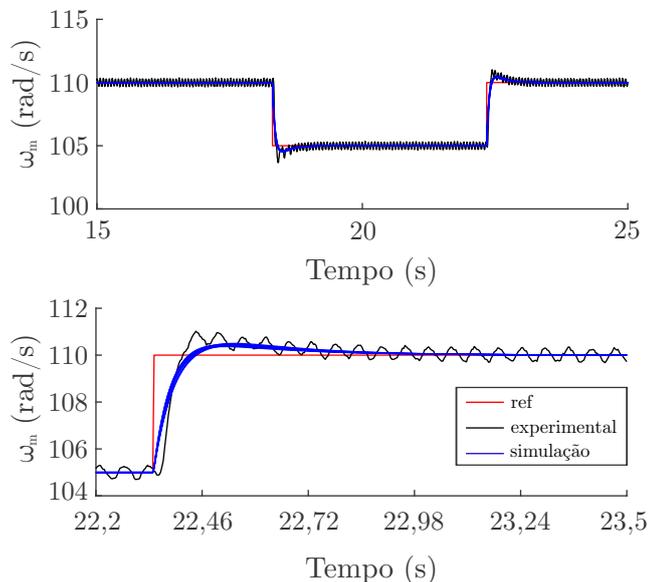
Figura 4.8 – Resultados experimentais para rastreamento de referência com os controladores baseados em PSO e *pidtune* sob variações de carga: (a) respostas da malha de controle de velocidade (superior) e detalhamento do comportamento na variação de referência (inferior); (b) malhas de controle das corrente no eixo *d* (superior) e no eixo *q* (inferior).



Fonte: Elaborado pelo Autor.

A Figura 4.9 apresenta uma comparação para a malha de velocidade entre as respostas de simulação por modelo médio linear, no MATLAB, e do resultado experimental. Pode-se confirmar uma boa correspondência entre o resultado experimental e as simulações, que reproduzem suficientemente bem o comportamento médio da prática. Isso indica a adequação dos modelos lineares de primeira ordem das plantas para orientar os projetos robustos de PIs nesta aplicação. As oscilações observadas nos resultados experimentais são devidas a vibrações mecânicas no acoplamento entre o eixo do PMSM e o *encoder*, bem como à construção não ideal da máquina, que produz oscilações de torque.

Figura 4.9 – Resposta ao degrau do sistema com os ganhos (4.13)-(4.15), comparando o resultado do modelo médio linear do MATLAB e o resultado experimental, para a malha de velocidade.



Fonte: Elaborado pelo Autor.

Por fim, é possível afirmar que os ganhos de controle obtidos pelo procedimento de projeto proposto garantem ao sistema em malha fechada um desempenho adequado, com boas respostas transitórias e em regime permanente, mesmo quando o PMSM é submetido a distúrbios, ruídos, incertezas paramétricas e não linearidade existentes na prática.

4.5 CONSIDERAÇÕES FINAIS

Este capítulo apresentou a validação experimental do procedimento proposto no Capítulo 3 para o controle robusto de um motor síncrono de ímãs permanentes com parâmetros mecânicos e elétricos incertos. Três controladores PIs foram projetados de forma independente entre si para uma estratégia de controle FOC, em que todos os ganhos obtidos respeitaram às especificações de projeto para toda a faixa de incertezas. Os resultados mostraram que o sistema controlado com os PIs projetados pelo procedimento proposto possui bom desempenho quando comparado a outros PIs projetados a partir de métodos tradicionais encontrados na literatura, e também é uma alternativa em relação a controladores robustos projetados a partir de técnicas mais avançadas, como em [Gabbi et al. 2019, Koch et al. 2018]. A validação experimental em um protótipo em escala industrial confirma bom desempenho e a boa correspondência dos resultados experimentais com os resultados obtidos por simulação dos modelos no MATLAB, mostrando que o procedimento proposto pode ser aplicado, na prática, em plantas de interesse em Engenharia Elétrica.

5 CONCLUSÕES

Este trabalho apresentou como contribuição um procedimento de projeto de controladores robustos a incertezas paramétricas aplicado a conversores de potência, baseado no algoritmo PSO, para busca dos ganhos de controle, combinado com o Teorema de Kharitonov, para a certificação teórica da estabilidade robusta do sistema em malha fechada.

No Capítulo 2, é apresentado o algoritmo PSO, detalhando seu princípio de funcionamento e as equações de posição e velocidade das partículas. Ilustrou-se, por meio da otimização de uma função de *benchmark*, que o PSO é capaz de encontrar valores ótimos globais em uma região de busca, para uma função que possui múltiplos pontos de mínimo. Um exemplo de projeto de um controlador PID para uma planta de segunda ordem é mostrado, em que, por meio de uma função objetivo que considera a estabilidade em malha fechada e a não violação das margens de fase e de ganho, o PSO é capaz de encontrar os ganhos do controlador, com desempenho superior àquele obtido por meio de um PID projetado de forma analítica, por um método da literatura, detalhado no Apêndice A.

O Capítulo 3 apresentou o procedimento de projeto proposto nesta dissertação, para obtenção de controladores robustos. Para iniciar o procedimento, é necessária a descrição em função de transferência do conversor, assumindo parâmetros físicos incertos, e a escolha da função de transferência do controlador a ser projetado. Após, devem-se informar as especificações de projeto, que são incluídas na função objetivo, também proposta neste trabalho. Esta função avalia o desempenho do controlador que será projetado, para os vértices de um modelo politópico da planta, e testa a estabilidade robusta em malha fechada por meio dos quatro polinômios do Teorema de Kharitonov. A minimização da função objetivo é feita por meio do algoritmo PSO, que encontra uma solução em um espaço de busca definido sistematicamente com base no polinômio característico do sistema em malha fechada. Para ilustrar o uso do procedimento proposto, foi apresentado um projeto detalhado dos ganhos robustos de um controlador PID para a regulação da tensão de saída de um conversor *Buck* CC-CC, com dois parâmetros incertos: a tensão de entrada e a resistência de carga. Simulações do sistema controlado com o ganho obtido por meio do procedimento proposto, e uma comparação com um PID obtido por meio da função *pidtune* do MATLAB, mostram um desempenho superior, tanto no transitório quanto em regime permanente, para o sistema com o controlador obtido por meio do procedimento proposto.

O Capítulo 4 apresentou uma validação experimental, em uma aplicação prática de controle de velocidade de um motor síncrono de ímãs permanentes, em que três controladores PIs são projetados de forma independente, considerando parâmetros mecânicos e elétricos da planta como incertos. O PSO foi capaz de obter, para as malhas de corrente direta e em quadratura, e para a malha de velocidade, ganhos fixos de controle que respeitam os valores de referência de margem de fase e frequência de cruzamento, e

limitantes para margem de ganho, *overshoot*, erro em regime permanente e do valor do sinal de controle, e também atendendo à estabilidade robusta em malha fechada. Análises comparativas foram apresentadas, para os controladores obtidos pelo procedimento proposto e outros três controladores obtidos por técnicas da literatura (GA, *pidtune* e PSO-ITAE), mostrando vantagens dos controladores robustos obtidos por meio do procedimento proposto. Por fim, resultados experimentais a partir da implementação dos ganhos obtidos pelo procedimento proposto foram apresentados, para um protótipo do PMSM em laboratório, confirmando a viabilidade da modelagem utilizada, e também confirmando robustez para a faixa de incertezas nos parâmetros do motor.

5.1 PERSPECTIVAS

Como perspectivas deste trabalho, consideram-se:

- aplicar o procedimento de projeto proposto em outras plantas de interesse em eletrônica de potência, como, por exemplo, fontes ininterruptas de energia e conversores conectados à rede;
- estender o procedimento proposto para o controle no espaço de estados, buscando ganhos por realimentação de estados por meio do PSO;
- estender o procedimento proposto para o projeto de controle em cascata utilizando PIs e PIDs;
- agregar, ao estágio de projeto, testes em plataformas de simulação em tempo real, por meio da integração do procedimento com outras plataformas de simulação, como o *Hardware-in-the-loop*;
- estender o procedimento para projeto no domínio do tempo discreto.

5.2 PUBLICAÇÕES

As seguintes publicações do autor possuem relação com o tema desta dissertação:

Artigos em periódicos e capítulos de livros

- i. L. C. Borin, I. Cleveston, C. R. D. Osório, G. G. Koch, F. M. Bottega, V. F. Montagner. "Otimização por Enxame de Partículas Aplicada a Controladores de Corrente para Inversores Conectados à Rede", *A Aplicação do Conhecimento Científico na Engenharia Elétrica*, Editora Atena, 2020.

- ii. C. R. D. Osório, G. G. Koch, I. Cleveston, L. C. Borin, F. H. Dupont, R. C. L. F. Oliveira, V. F. Montagner. "Otimização Multiobjetivo para Controle Robusto Aplicado a Inversores Conectados à Rede", *Revista Eletrônica de Potência*, v. 24, p.1-9, 2018.

Trabalhos completos em anais de congressos

- i. L. C Borin, E. Mattos, C. R. D. Osório, G. G. Koch, V. F. Montagner. "Robust PID Controllers Optimized by PSO Algorithm for Power Converters", *5th Southern Power Electronics Conference & 15th Brazilian Power Electronics Conference*, Santos, 2019.
- ii. C. R. D. Osório, L. C Borin, G. G. Koch, V. F. Montagner. "Optimization of Robust PI Controllers for Grid-Tied Inverters", *5th Southern Power Electronics Conference & 15th Brazilian Power Electronics Conference*, Santos, 2019.
- iii. L. C Borin, C. R. D. Osório, G. G. Koch, M. R. Nascimento, F. M. Bottega, V. F. Montagner. "Particle Swarm Optimization for Robust Control Tuning Applied to Uninterruptible Power Supplies", *IEEE PES Innovative Smart Grid Technologies Conference - Latin America*, Gramado, 2019.
- iv. L. C Borin, I. Cleveston, C. R. D. Osório, G. G. Koch, F. M. Bottega, V. F. Montagner. "Algoritmo Genético Aplicado a Controladores de Corrente para Inversores Conectados à Rede", *11º Seminário de Eletrônica de Potência e Controle*, Santa Maria, 2018.
- v. I. Cleveston, L. C Borin, C. R. D. Osório, G. G. Koch, F. M. Bottega, V. F. Montagner. "Otimização por Enxame de Partículas Aplicada a Controladores de Corrente para Inversores Conectados à Rede", *11º Seminário de Eletrônica de Potência e Controle*, Santa Maria, 2018.
- vi. C. R. D. Osório, G. G. Koch, L. C. Borin, I. Cleveston, F. H. Dupont, V. F. Montagner. "Validação Experimental de um Controle por realimentação de Estados Baseado em Algoritmo Genético para Conversores Conectados à rede", *XXII Congresso Brasileiro de Automática*, João Pessoa, 2018.
- vii. G. G. Koch, C. R. D. Osório, L. C. Borin, I. Cleveston, V. F. Montagner. "Projeto e Implementação Prática de Controladores de Corrente para Inversores Trifásicos Conectados à Rede Utilizando Algoritmo PSO", *XXII Congresso Brasileiro de Automática*, João Pessoa, 2018.

REFERÊNCIAS

- ACKERMANN, J. **Robust control: the parameter space approach**. [S.l.]: Springer Science & Business Media, 2012.
- ARCURI, A.; BRIAND, L. A practical guide for using statistical tests to assess randomized algorithms in software engineering. In: IEEE. **2011 33rd International Conference on Software Engineering (ICSE)**. [S.l.], 2011. p. 1–10.
- ÅSTRÖM, K. J.; HÄGGLUND, T. **Advanced PID control**. [S.l.]: ISA - The Instrumentation, Systems and Automation Society Research Triangle, 2006. v. 461.
- BANERJEE, S.; GHOSH, A.; RANA, N. An improved interleaved boost converter with PSO-based optimal type-III controller. **IEEE Journal of Emerging and Selected Topics in Power Electronics**, IEEE, v. 5, n. 1, p. 323–337, 2017.
- BARMISH, B. R. A generalization of Kharitonov’s four-polynomial concept for robust stability problems with linearly dependent coefficient perturbations. **IEEE Transactions on Automatic Control**, v. 34, n. 2, p. 157–165, Feb 1989. ISSN 0018-9286.
- BAZANELLA, A. S.; SILVA, J. M. G. J. da. **Sistemas de controle: Princípios e métodos de projeto**. [S.l.]: UFRGS, 2005.
- BEVRANI, H. et al. Robust control design and implementation for a quadratic buck converter. In: IEEE. **The 2010 International Power Electronics Conference-ECCE ASIA-**. [S.l.], 2010. p. 99–103.
- _____. Intelligent frequency control in an ac microgrid: Online PSO-based fuzzy tuning approach. **IEEE transactions on smart grid**, IEEE, v. 3, n. 4, p. 1935–1944, 2012.
- BHATTACHARYYA, S. Robust control under parametric uncertainty: An overview and recent results. **Annual Reviews in Control**, Elsevier, v. 44, p. 45–77, 2017.
- BHATTACHARYYA, S. P.; CHAPPELLAT, H.; KEEL, L. H. **Robust Control: The Parametric Approach**. Upper Saddle River, NJ, USA: Prentice-Hall Publishing Co., 1995.
- BHATTACHARYYA, S. P.; KEEL, L. H. Robust control: the parametric approach. In: **Advances in Control Education 1994**. [S.l.]: Elsevier, 1995. p. 49–52.
- BIANCHI, L. et al. A survey on metaheuristics for stochastic combinatorial optimization. **Natural Computing**, Springer, v. 8, n. 2, p. 239–287, 2009.
- BORIN, L. et al. Particle swarm optimization for robust control tuning applied to uninterruptible power supplies. In: IEEE. **2019 IEEE PES Innovative Smart Grid Technologies Conference-Latin America (ISGT Latin America)**. [S.l.], 2019. p. 1–6.

- BOYD, S. et al. **Linear matrix inequalities in system and control theory**. [S.l.]: Siam, 1994. v. 15.
- CHUNG, I.-Y. et al. Control methods of inverter-interfaced distributed generators in a microgrid system. **IEEE Trans. Ind. Appl**, v. 46, n. 3, p. 1078–1088, 2010.
- DEB, K. **Multi-objective optimization using evolutionary algorithms**. [S.l.]: John Wiley & Sons, 2001. v. 16.
- DELL’AQUILA, A.; LECCI, A. PSO-based control optimization of power converters. In: IEEE. **2008 International Symposium on Power Electronics, Electrical Drives, Automation and Motion**. [S.l.], 2008. p. 964–968.
- DORF, R. C.; BISHOP, R. H. **Modern control systems**. [S.l.]: Pearson, 2011.
- DRÉO, J. et al. **Metaheuristics for hard optimization: methods and case studies**. [S.l.]: Springer Science & Business Media, 2006.
- EBERHART, R.; KENNEDY, J. A new optimizer using particle swarm theory. In: IEEE. **In Proceedings of the Sixth International Symposium on Micro Machine and Human Science**. [S.l.], 1995. p. 39–43.
- EBERHART, R. C.; SHI, Y. Particle swarm optimization: developments, applications and resources. In: IEEE. **Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)**. [S.l.], 2001. v. 1, p. 81–86.
- ELSISI, M. Optimal design of nonfragile PID controller. **Asian Journal of Control**, Wiley, 2019.
- ERICKSON, R. W. **Fundamentals of Power Electronics**. New York, NY: Chapman & Hall, 1997.
- FLASIŃSKI, M. **Introduction to artificial intelligence**. [S.l.]: Springer, 2016.
- GABBI, T. S. et al. Linear matrix inequalities for digital redesign under delay suitable for PI controllers with application to PMSMs. **Journal of Control, Automation and Electrical Systems**, Springer, p. 1–11, 2019.
- GAING, Z.-L. A particle swarm optimization approach for optimum design of PID controller in AVR system. **IEEE transactions on energy conversion**, IEEE, v. 19, n. 2, p. 384–391, 2004.
- GALCCKI, A. et al. Particle swarm optimization of the state feedback current controller with oscillatory terms for a three-phase grid-tie converter. In: IEEE. **2018 20th European Conference on Power Electronics and Applications (EPE’18 ECCE Europe)**. [S.l.], 2018. p. P–1.
- GRAY, J. The benchmark handbook for database and transaction systems. **Morgan Kaufmann, San Mateo**, 1993.

- GUO, L.; HUNG, J. Y.; NELMS, R. M. Evaluation of DSP-based PID and fuzzy controllers for DC–DC converters. **IEEE transactions on industrial electronics**, IEEE, v. 56, n. 6, p. 2237–2248, 2009.
- HANIFAH, R. A. et al. Swarm-intelligence tuned current reduction for power-assisted steering control in electric vehicles. **IEEE Transactions on Industrial Electronics**, IEEE, v. 65, n. 9, p. 7202–7210, 2018.
- HAUPT, R.; HAUPT, S. **Practical Genetic Algorithms**. [S.l.]: John Wiley, 2004. (Wiley-Interscience publication).
- ISHAQUE, K. et al. An improved particle swarm optimization (PSO)–based MPPT for PV with reduced steady-state oscillation. **IEEE transactions on Power Electronics**, IEEE, v. 27, n. 8, p. 3627–3638, 2012.
- JUNG, J.-W. et al. Fuzzy PI–type current controllers for permanent magnet synchronous motors. **IET electric power applications**, IET, v. 5, n. 1, p. 143–152, 2011.
- KOCH, G. G. et al. Linear matrix inequality based synthesis of PI controllers for PMSM with uncertain parameters. **Eletrônica de Potência**, v. 23, p. 310–319, 2018.
- KRAUSE, P.; WASYNCZUK, O.; SUDHOFF, S. **Analysis of Electric Machinery and Drive Systems**. Second. United States of America: Wiley-IEEE Press, 2002.
- KRISHNAN, R. **Electric motor drives: modeling, analysis, and control**. [S.l.]: Prentice Hall, 2001. ISBN 9780130910141.
- KUMAR, S. S.; SHREESHA, C. Design of robust PID controller for a CSTR plant with interval parametric uncertainty using Kharitonov theorem. In: IEEE. **2016 International Conference on Computation of Power, Energy Information and Commuinca-tion (ICCPEIC)**. [S.l.], 2016. p. 430–433.
- LEÓN-ALDACO, S. E. D.; CALLEJA, H.; ALQUICIRA, J. A. Metaheuristic optimization methods applied to power converters: A review. **IEEE Transactions on Power Electronics**, IEEE, v. 30, n. 12, p. 6791–6803, 2015.
- LIU, C.-H.; HSU, Y.-Y. Design of a self-tuning PI controller for a STATCOM using particle swarm optimization. **IEEE Transactions on industrial Electronics**, IEEE, v. 57, n. 2, p. 702–715, 2010.
- LOBATO, S. de C. et al. Didactic prototype to model and to design linear control applied to a RLC plant. In: IEEE. **2017 Brazilian Power Electronics Conference (COBEP)**. [S.l.], 2017. p. 1–6.
- MEZURA-MONTES, E.; COELLO, C. A. C. Constraint-handling in nature-inspired numerical optimization: past, present and future. **Swarm and Evolutionary Computation**, Elsevier, v. 1, n. 4, p. 173–194, 2011.

MIRANDA, M.; TAKAHASHI, R.; JOTA, F. Hierarchical approach for H-infinity robust control design: S/KS mixed sensitivity with genetic algorithm. **IET Control Theory and Applications**, v. 1, p. 18–24, 01 2007.

MURARI, A. et al. A LQRI power control for DFIG tuned by a weighted-PSO. **Control Engineering Practice**, Elsevier, v. 85, p. 41–49, 2019.

NAZARI, S. et al. Application of Kharitonov's Theorem in robust stability of voltage and frequency of an islanded microgrid. In: IEEE. **2018 Smart Grid Conference (SGC)**. [S.l.], 2018. p. 1–6.

OGATA, K. **Discrete-time control systems**. [S.l.]: Prentice Hall, 1995.

PANDEY, S. K. et al. State and disturbance observer based integral sliding mode controlled boost DC–DC converters. **IEEE Transactions on Circuits and Systems II: Express Briefs**, IEEE, 2018.

PEDERSEN, M. E. H. Good parameters for particle swarm optimization. **Hvass Lab., Copenhagen, Denmark, Tech. Rep. HL1001**, p. 1551–3203, 2010.

PILLAY, P.; KRISHNAN, R. Modeling, simulation, and analysis of permanent-magnet motor drives. I. the permanent-magnet synchronous motor drive. **IEEE Transactions on Industry Applications**, v. 25, n. 2, p. 265–273, 1989.

POLI, R.; KENNEDY, J.; BLACKWELL, T. Particle swarm optimization: An overview. **Swarm intelligence**, Springer, v. 1, n. 1, p. 33–57, 2007.

QI, Z.; SHI, Q.; ZHANG, H. Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-based DC motor subject to stochastic delays. **IEEE Transactions on Industrial Electronics**, IEEE, 2019.

RAMOS, R.; BRETAS, N.; ALBERTO, L. C. Damping controller design for power systems with polytopic representation of operating conditions. In: IEEE. **2002 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No. 02CH37309)**. [S.l.], 2002. v. 2, p. 1517–1521.

RAO, S. S. **Engineering optimization: theory and practice**. [S.l.]: John Wiley & Sons, 2009.

RENAUDINEAU, H. et al. A PSO-based global MPPT technique for distributed PV power generation. **IEEE Transactions on Industrial Electronics**, IEEE, v. 62, n. 2, p. 1047–1058, 2015.

SEBTAHMADI, S. S. et al. A PSO–DQ current control scheme for performance enhancement of Z–source matrix converter to drive IM fed by abnormal voltage. **IEEE Transactions on Power Electronics**, IEEE, v. 33, n. 2, p. 1666–1681, 2017.

- SENGUPTA, S.; BASAK, S.; PETERS, R. Particle swarm optimization: A survey of historical and recent developments with hybridization perspectives. **Machine Learning and Knowledge Extraction**, Multidisciplinary Digital Publishing Institute, v. 1, n. 1, p. 157–191, 2018.
- SERAPIÃO, A. B. d. S. Fundamentos de otimização por inteligência de enxames: uma visão geral. **Sba: Controle & Automação Sociedade Brasileira de Automatica**, SciELO Brasil, v. 20, n. 3, p. 271–304, 2009.
- SHI, Y.; EBERHART, R. C. Parameter selection in particle swarm optimization. In: SPRINGER. **International conference on evolutionary programming**. [S.l.], 1998. p. 591–600.
- SHIN, E.-C. et al. A design method of PI controller for an induction motor with parameter variation. In: IEEE. **IECON'03. 29th Annual Conference of the IEEE Industrial Electronics Society (IEEE Cat. No. 03CH37468)**. [S.l.], 2003. v. 1, p. 408–413.
- SILVA, S. A. O. da et al. Feed-forward DC-bus control loop applied to a single-phase grid-connected PV system operating with PSO-based MPPT technique and active power-line conditioning. **IET Renewable Power Generation**, IET, v. 11, n. 1, p. 183–193, 2016.
- STRAS, A.; UFNALSKI, B.; GRZESIAK, L. Particle swarm optimization-based gain, delay compensation and filter determination of a repetitive controller for a grid-tie converter. In: IEEE. **2018 International Symposium on Industrial Electronics (INDEL)**. [S.l.], 2018. p. 1–7.
- TALBI, E.-G. **Metaheuristics: from design to implementation**. [S.l.]: John Wiley & Sons, 2009. v. 74.
- TEJA, S. V.; SHANAVAS, T.; PATNAIK, S. Modified PSO based sliding-mode controller parameters for buck converter. In: IEEE. **2012 IEEE Students' Conference on Electrical, Electronics and Computer Science**. [S.l.], 2012. p. 1–4.
- THIRUMENI, M.; THANGAVELUSAMY, D. Design and analysis of hybrid PSO–GSA tuned PI and SMC controller for DC–DC cuk converter. **IET Circuits, Devices & Systems**, IET, v. 13, n. 3, p. 374–384, 2019.
- UFNALSKI, B.; GRZESIAK, L. M. Plug-in direct multi-swarm repetitive controller for the sine wave inverter-on keeping particles diversified in a dynamic and noisy environment. In: IEEE. **2016 10th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)**. [S.l.], 2016. p. 484–491.
- UFNALSKI, B.; KASZEWSKI, A.; GRZESIAK, L. M. Particle swarm optimization of the multioscillatory LQR for a three-phase four-wire voltage-source inverter with an LC output filter. **IEEE Transactions on Industrial Electronics**, IEEE, v. 62, n. 1, p. 484–493, 2015.

- VEERACHARY, M.; SAXENA, A. R. Optimized power stage design of low source current ripple fourth-order boost dc–dc converter: A pso approach. **IEEE Transactions on Industrial Electronics**, IEEE, v. 62, n. 3, p. 1491–1502, 2015.
- WANG, X.; UFNALSKI, B.; GRZESIAK, L. M. Adaptive speed control in the PMSM drive for a non-stationary repetitive process using particle swarms. In: IEEE. **2016 10th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)**. [S.l.], 2016. p. 464–471.
- WU, F. et al. Small signal stability analysis and optimal control of a wind turbine with doubly fed induction generator. **IET Generation, Transmission & Distribution**, IET, v. 1, n. 5, p. 751–760, 2007.
- YOSHIDA, H. et al. A particle swarm optimization for reactive power and voltage control considering voltage security assessment. **IEEE Transactions on power systems**, IEEE, v. 15, n. 4, p. 1232–1239, 2000.
- YOUSEFI, M. et al. Particle swarm optimization and genetic algorithm to optimizing the pole placement controller on cuk converter. In: IEEE. **2008 IEEE 2nd International Power and Energy Conference**. [S.l.], 2008. p. 1461–1465.
- ZOBAA, A.; LECCI, A. Particle swarm optimisation of resonant controller parameters for power converters. **IET power electronics**, IET, v. 4, n. 2, p. 235–241, 2011.

APÊNDICES

Apêndice A – Resolução baseada em formulação analítica para o problema apresentado na Seção 2.5

Para a resolução do problema exposto na Seção 2.5 desta dissertação, no Exemplo 6.9 (página 222) em [Bazanella e Silva 2005], é proposta uma solução constituída de duas etapas, em que, em um primeiro momento, projeta-se um controlador PI, e em um segundo momento, projeta-se um controlador PD (para detalhes, veja [Bazanella e Silva 2005]).

Como apresentado na Seção 2.5, as funções de transferências da planta $G_p(s)$ e do sensor $H(s)$, são dadas por

$$G_p(s) = \frac{200}{s^2 + 2s + 100} \quad , \quad H(s) = \frac{400}{s + 400} \quad (\text{A.1})$$

e o controlador PID a ser projetado deve ter um polo fixado na frequência $p = 1000$ rad/s, sendo que o sistema controlado deve atender os requisitos pré-estabelecidos de margem de fase $\geq 45^\circ$ e margem de ganho ≥ 3 (9,5 dB).

Primeiramente, para obtenção do controlador PI, foi incluída uma restrição adicional de projeto, em que o sistema deve seguir uma referência do tipo rampa com erro inferior a 5% ($e_{rampa}(\infty) \leq 0,05$). Para atender a esta especificação de regime permanente, deve-se obter um controlador com ganho tal que a função $G_c(s)G_p(s)H(s)$ satisfaça

$$\lim_{s \rightarrow 0} sL(s) \geq 20 \quad (\text{A.2})$$

conforme o teorema do valor final.

Considerando a função de transferência de um PI sendo dada por $C_{PI}(s) = K \frac{s+z}{s}$, tem-se que o zero do controlador deve ter valor pequeno, de modo que o atraso de fase inserido pelo bloco PI se dê em baixas frequências [Bazanella e Silva 2005].

Deste modo

$$\frac{1}{e_{rampa}(\infty)} = \lim_{s \rightarrow 0} sL(s) = \lim_{s \rightarrow 0} sK \frac{s+z_1}{s} G_p(s)H(s) = K z_1 G_p(0)H(0) = 2K z_1 \quad (\text{A.3})$$

sendo o zero do controlador PI escolhido em $z_1 = 1$, e então o ganho $K = 10$, que satisfaz (A.2).

O controlador PI projetado é dado por

$$C_{PI}(s) = 10 \frac{s+1}{s} \quad (\text{A.4})$$

Note que $G_c(s)G_p(s)H(s)$ tem uma frequência de cruzamento do ganho de aproximadamente $\omega_0 \simeq 45$ rad/s, e a fase da resposta nesta frequência inferior a -180° , o que torna o sistema instável.

Para estabilizar o sistema, e atender às especificações de margem de fase, o bloco PD deve então ser projetado. Em [Bazanella e Silva 2005], considera-se a função de

transferência de um PD sendo $C_{PD}(s) = K \frac{1+Ts}{1}$, que provê o avanço de fase desejado ϕ na frequência de cruzamento do ganho e, uma vez determinado ω_0 , o parâmetro T pode ser calculado como $T = \frac{\tan \phi}{\omega_0}$.

Para encontrar o controlador PD, foi escolhido o valor $\phi = 65^\circ$, para garantir que o valor de margem de fase desejada seja obtido. Assim, tem-se que

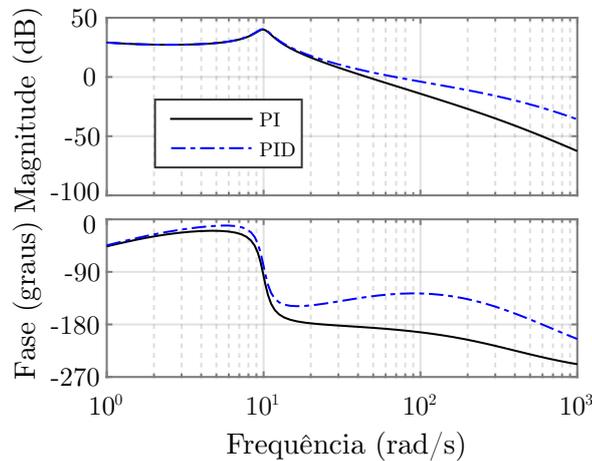
$$|C_{PI}(j\omega_0)G_p(j\omega_0)H(j\omega_0)| = 10 \left| \frac{j\omega_0 + 1}{j\omega_0} \cdot \frac{200}{(j\omega_0)^2 + 2j\omega_0 + 100} \cdot \frac{400}{j\omega_0 + 400} \right| = \cos \phi = 0,42 \quad (\text{A.5})$$

obtendo $\omega_0 = 69 \text{ rad/s}$, $T = \frac{\tan \phi}{\omega_0} = 0,0311 \text{ s}$ ou $z_2 = 31,1 \text{ rad/s}$, e o PID resultante dado por

$$C_{PID}(s) = 10 \frac{(s+1)(31,1s+1000)}{s(s+1000)} = \frac{0,311s^2 + 10,311s + 10}{s} \cdot \frac{1000}{s+1000} \quad (\text{A.6})$$

A resposta em frequência da função $G_c(s)G_p(s)H(s)$ com este controlador é apresentada no diagrama de Bode abaixo, juntamente com a resposta somente com o controlador PI para comparação. Pode-se observar que as margens de estabilidade obtidas são margem de fase de 52° e margem de ganho de 26 dB, atendendo às especificações do problema.

Figura A.1 – Diagrama de Bode do sistema em malha aberta, comparando o controlador PI em (A.4) e o controlador PID em (A.6).



Fonte: Adaptado de [Bazanella e Silva 2005].