



Pós-Graduação em Ciência da Computação

JUDAS TADEU GOMES DE SOUSA

**OTIMIZAÇÃO MULTI-OBJETIVO ON-LINE DOS
PARÂMETROS DE UM CONTROLADOR MFAC
APLICADO A SISTEMAS NÃO-LINEARES
MEDIANTE ALGORITMO DE EVOLUÇÃO
DIFERENCIAL COM IMIGRANTES DIRECIONADOS**



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE
2017

Judas Tadeu Gomes de Sousa

**Otimização Multi-Objetivo *On-Line* dos Parâmetros de um Controlador MFAC
Aplicado a Sistemas Não-lineares Mediante Algoritmo de Evolução Diferencial com
Imigrantes Direcionados**

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

ORIENTADOR: Prof. Dr. Aluizio F. R. Araújo

RECIFE
2017

Catálogo na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S725o Sousa, Judas Tadeu Gomes de
 Otimização multi-objetivo on-line dos parâmetros de um controlador MFAC
 aplicado a sistemas não-lineares mediante algoritmo de evolução diferencial
 com imigrantes direcionados / Judas Tadeu Gomes de Sousa. – 2017.
 172 f.: il., fig., tab.

 Orientador: Aluizio Fausto Ribeiro Araújo.
 Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da
 Computação, Recife, 2017.
 Inclui referências e apêndices.

 1. Inteligência artificial. 2. Otimização. I. Araújo, Aluizio Fausto Ribeiro
 (orientador). II. Título.

006.3

CDD (23. ed.)

UFPE- MEI 2017-215

Judas Tadeu Gomes de Sousa

**Otimização Multi-objetivo on-line dos Parâmetros de um Controlador MFAC
Aplicado a Sistemas não-lineares Mediante Algoritmo de Evolução Diferencial
com Imigrantes Direcionados**

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutora em Ciência da Computação

Aprovado em: 12/06/2017

Orientador: Prof. Dr. Aluizio Fausto Ribeiro Araújo

BANCA EXAMINADORA

Prof. Dr. Adriano Lorena Inacio de Oliveira
Centro de Informática / UFPE

Prof. Dr. Abel Guilhermino da Silva Filho
Centro de Informática / UFPE

Prof. Dr. Renato Tinós
Departamento de Computação e Matemática / USP

Prof. Dr. Carmelo Jose Albanez Filho
Escola Politécnica de Pernambuco / UPE

Prof. Dr. Tiago Alessandro Espinola Ferreira
Departamento de Estatística e Informática / UFRPE

Agradecimentos

Agradeço a minha mãe Maria de Fátima por minha vida e por me ensinar que a educação pode mudar o destino das pessoas.

Agradeço a minha esposa e parceira Karine por sua compreensão e apoio em todos momentos difíceis dessa jornada.

Agradeço as minhas filhas Julia e Lívia, cujo amor é sempre um refúgio para as dificuldades do dia a dia.

Agradeço ao meu irmão Getúlio por acreditar em mim, mesmo quando eu duvidava.

Agradeço ao meu orientador o professor Aluizio Fausto por acreditar no meu potencial, me orientar e me apoiar durante todo o trabalho.

Agradeço aos membros da minha banca examinadora os professores Adriano Lorena, Carmelo José, Renato Tinós, Tiago Alessandro e Abel Guilhermino por suas valiosas sugestões para aprimoramento desta Tese.

Agradeço à UNIVASF ao CIN e à FACEPE por me permitir realizar esse sonho.

Resumo

A demanda pelo tratamento de sistemas não-lineares, resultado do aumento da complexidade dos processos industriais recentes, tem dificultado o uso de técnicas de controle moderno. A teoria de controle moderno é baseada na existência de modelo para representar o processo, no entanto, o uso de modelos complexos pode resultar num controlador complexo e difícil de manter. Técnicas de controle direcionadas por dados estão ganhando destaque em áreas onde a complexidade do sistema, ou mesmo a inexistência de um modelo, podem ser superadas pela disponibilidade de dados do processo, os quais podem ser capturados e usados para calcular diretamente o sinal de controle. Dentre os métodos de controle direcionados por dados, a técnica de Controle Adaptativo Livre de Modelo (MFAC – *Model Free Adaptive Control*) se destaca por características como: ser *on-line*, depender apenas dos dados de entrada e saída medidos da planta e do sinal de referência e por possuir formulações que atendem sistemas com vários graus de não-linearidade. Porém, o MFAC ainda possui questões em aberto, por exemplo, a escolha dos parâmetros do controlador. O ajuste desses parâmetros pode ser transformado num problema de otimização, no entanto, um projeto de controle costuma envolver múltiplos objetivos a serem atendidos. Portanto, neste trabalho serão definidas estratégias e um algoritmo evolucionário multi-objetivo, baseado em evolução diferencial e em imigrantes direcionados, para sintonia dos parâmetros do controlador MFAC. Vários casos de estudos serão testados e duas estratégias de ajustes para os parâmetros serão implementadas: uma estratégia *off-line*, na qual os parâmetros são otimizados em todo intervalo de operação, e outra *on-line*, onde o controlador usa os parâmetros otimizados na estratégia anterior, mas também realiza otimizações em intervalos menores, enquanto o controle atua, quando algumas situações são detectadas. Os resultados obtidos mediante simulações, sugerem que o controlador usando parâmetros otimizados *off-line* tem melhor desempenho do que um com parâmetros encontrados na literatura. Além disso, a estratégia de otimização *on-line* proposta, conseguiu melhorar ou pelo menos manter os benefícios obtidos com a otimização *off-line*.

Palavras-chave: Controle Adaptativo. Controle Direcionado a Dados. Otimização Multiobjetivo. Evolução Diferencial.

Abstract

As a result of the complexity increase in current industrial processes, requests for treatment of nonlinear systems have been overburdening modern control techniques. The modern control theory is based on a model to represent these processes, however complex models can result in a complicated and difficult controller to maintain. Data-Driven Control techniques are getting featured in areas where the system complexity, or the absence of a model, can be overcome by a lot of available data, which can be used to calculate the control signal directly. Among Data-Driven Control methods, the Model Free Adaptive Control (MFAC) technique stands out for characteristics, such as being on-line, using just input and output data from the plant and reference signal, as well as having formulations for systems with varying degrees of non-linearity. However, the MFAC is still making unanswered questions, such as the choice of controller parameters. The tuning of these parameters can be transformed into an optimization problem, nevertheless, a control project usually involves multiple objectives to be attended. Therefore, this work will define strategies and a multi-objective evolutionary algorithm, based on differential evolution and directed immigrants, to adjust the MFAC controller parameters. Several cases will be evaluated and two adaptive strategies for these parameters will be implemented: An off-line strategy, at which the parameters are optimized in all acting period, and another on-line, where the controller uses the optimized parameters obtained in the previous strategy performing optimizations at smaller intervals when some situations are detected. The results obtained through simulations suggest that the controller with optimized parameters off-line is better than parameters found in the literature. In addition, the proposed on-line strategy has been able to improve or at least maintain the benefits of off-line optimization.

Keywords: Adaptive Control. Data Driven Control. Multiobjective Optimization. Differential Evolution.

Lista de Figuras

Figura 1 - Relação do sistema controlado, modelo e o controle no MBC.....	25
Figura 2 - Relação entre o sistema controlado, os dados E/S e o controle.....	26
Figura 3 - Etapas do algoritmo de Evolução Diferencial	45
Figura 4 – Distribuição de indivíduos durante DE	49
Figura 5 – Dinâmica da distribuição dos indivíduos no espaço de soluções.	52
Figura 6 - Representação do modelo proposto de DE.	59
Figura 7 – Esquema para sintonia off-line dos parâmetros do controlador.....	69
Figura 8 – Esquema para sintonia on-line dos parâmetros do controlador.	72
Figura 9 – Fronteira de Pareto e parâmetros em destaque.	90
Figura 10 - Simulação do sistema usando os parâmetros em destaque.	91
Figura 11 – Fronteira de Pareto e parâmetros em destaque.....	93
Figura 12 - Simulação do sistema usando os parâmetros em destaque.	94
Figura 13 – Fronteira de Pareto e parâmetros em destaque	96
Figura 14 - Simulação do sistema usando os parâmetros em destaque.	97
Figura 15 – Fronteira de Pareto e parâmetros em destaque.....	99
Figura 16 - Simulação do sistema usando os parâmetros em destaque	100
Figura 17 – Fronteira de Pareto e parâmetros em destaque.....	102
Figura 18 - Simulação do sistema usando os parâmetros em destaque.	103
Figura 19 – Fronteira de Pareto e parâmetros em destaque	105
Figura 20 - Simulação do sistema usando os parâmetros em destaque	106
Figura 21 - Fronteira de Pareto e parâmetros em destaque.....	108
Figura 22 - Simulação do sistema usando os parâmetros em destaque.	109
Figura 23 - Simulação do sistema usando a otimização off-line e on-line.....	111
Figura 24 - Simulação do sistema usando a otimização off-line e on-line.....	113
Figura 25 - Simulação do sistema usando a otimização off-line e on-line.....	115
Figura 26 - Simulação do sistema usando a otimização off-line e on-line.....	116
Figura 27 - Simulação do sistema usando a otimização off-line e on-line.....	118
Figura 28 - Simulação do sistema usando a otimização on-line.	119
Figura 29 - Simulação do sistema usando a otimização on-line.	120
Figura 30 – Diagrama para SPSA DDC.....	136
Figura 31 – Esquema simplificado de controle UC.....	141
Figura 32 – Diagrama de blocos da planta e o controlador	142
Figura 33 – Esquema Correlation-based Tuning (CbT)	144
Figura 34 - Virtual Reference Feedback Tuning (VRFT)	147
Figura 35 - Noniterative Data-driven Model Reference Control	150
Figura 36 - Iterative Learning Control	155
Figura 37 – Cálculo do sinal de controle	156

Lista de Tabelas

Tabela 1 - Classificação quanto ao uso de dados.	32
Tabela 2 – Resumo Comparativo entre os principais métodos DDC.	34
Tabela 3 - Relação de funções do CEC2013.	76
Tabela 4 - Resultados das simulações para $D = 5$	77
Tabela 5 - Resultados das simulações para $D = 10$	78
Tabela 6 - Resultados das simulações para $D = 30$	79
Tabela 7 - Resultados das simulações para $D = 50$	80
Tabela 8 – Resultado do teste de Wilcoxon para problemas mono-objetivo	81
Tabela 9 - Resultados das simulações para $D = 30$	84
Tabela 10 - Resultado do teste de Wilcoxon para problemas multi-objetivo	84
Tabela 11 – Resultados das otimizações off-line para o primeiro caso de estudo.	89
Tabela 12 - Tabela com o conjunto de parâmetros para o controlador	89
Tabela 13 – Resultados das otimizações off-line para o segundo caso de estudo.	92
Tabela 14 - Tabela com o conjunto de parâmetros para o controlador.	93
Tabela 15 – Resultados das otimizações off-line para o terceiro caso de estudo.	95
Tabela 16 - Tabela com o conjunto de parâmetros para o controlador.	95
Tabela 17 – Resultados das otimizações off-line para o quarto caso de estudo	98
Tabela 18 - Tabela com o conjunto de parâmetros para o controlador.	99
Tabela 19 – Resultados das otimizações off-line para o quinto caso de estudo.	101
Tabela 20 - Tabela com o conjunto de parâmetros para o controlador.	102
Tabela 21 – Resultados das otimizações off-line para o sexto caso de estudo.	104
Tabela 22 - Tabela com o conjunto de parâmetros para o controlador.	105
Tabela 23 – Resultados das otimizações off-line para o décimo caso de estudo.	107
Tabela 24 - Tabela com o conjunto de parâmetros para o controlador.	108
Tabela 25 – Histórico dos parâmetros do controlador para o primeiro caso de estudo.	110
Tabela 26 - Histórico dos parâmetros do controlador para o segundo caso de estudo.	112
Tabela 27 - Histórico dos parâmetros do controlador para o terceiro caso de estudo.	114
Tabela 28 - Histórico dos parâmetros do controlador para o quarto caso de estudo.	116
Tabela 29 - Histórico dos parâmetros do controlador para o quinto caso de estudo.	117
Tabela 30 - Histórico dos parâmetros do controlador para o sexto caso de estudo.	118
Tabela 31 - Histórico dos parâmetros do controlador para o sétimo caso de estudo.	119
Tabela 32 – Quadro resumo com índices de desempenho calculados usando as duas estratégias.	120

Lista de Acrônimos

ADP	<i>Approximate Dynamic Programing</i>
AEMO	<i>Algoritmo Evolucionário Multi-Objetivo</i>
AE	<i>Algoritmo Evolucionário</i>
AG	<i>Algoritmos Genéticos</i>
AGDE	<i>Adaptive Group-Based Differential Evolution</i>
AGS	<i>Algoritmo Genético Simples</i>
BH	<i>Busca Harmônica</i>
CbT	<i>Correlation-Based Tuning</i>
CD	<i>Crowding Distance</i>
CFDL	<i>Compact Form Dynamic Linearization</i>
CMA-ES	<i>Covariance Matrix Adaptation Evolutionary Strategy</i>
DDC	<i>Data Driven Control</i>
DE	<i>Differential Evolution</i>
DE-MPC	<i>Differential Evolution Model Predictive Control</i>
EE	<i>Estratégias Evolucionárias</i>
FFDL	<i>Full Form Dynamic Linearization</i>
FOPDT	<i>First Order Plus Time Delay</i>
HIDE	<i>Harmony Memory Improvement with Differential Evolution</i>
IDE	<i>Improved Differential Evolution</i>
IFT	<i>Iterative Feedback Tuning</i>
ILC	<i>Iterative Learning Control</i>
LabVIEW	<i>Laboratory Virtual Instrument Engineering Workbench</i>
MBC	<i>Model Based Control</i>
McDE	<i>Memetic Compact Differential Evolution</i>
MFAC	<i>Model Free Adaptive Control</i>
MFLAC	<i>Model Free Learning Adaptive Control</i>
MIMO	<i>Multiple Input Multiple Output</i>
MPPT	<i>Maximum Power Point Tracking</i>
NDDMRC	<i>Noniterative Data-Driven Model Reference Control</i>
NSDE	<i>Non-dominated Differential Evolution</i>
NSGA	<i>Non-dominated Sorting Genetic Algorithm</i>
LIT	<i>Linear Invariante no Tempo</i>
LL	<i>Lazy Learning</i>
PE	<i>Programação Evolucionária</i>
PPD	<i>Pseudo Partial Derivative</i>
IDE	<i>Improved Differential Evolution</i>
ILC	<i>Iterative Learning Control</i>
PD	<i>Proportional Derivative</i>
PFDL	<i>Partial Form Dynamic Linearization</i>
PG	<i>Programação Genética</i>
PID	<i>Proportional Integral Derivative</i>
POM	<i>Problema de Otimização Multiobjetivo</i>

SA	<i>Subspace Approach</i>
SISO	<i>Single Input Single Output</i>
SPEA	<i>Strength Pareto Evolutionary Algorithm</i>
SPSA	<i>Simultaneous Perturbation Stochastic Approximation</i>
VRFT	<i>Virtual Reference Feedback Tuning</i>
UC	<i>Unfalsified Control</i>

Sumário

1	INTRODUÇÃO	13
1.1	Objetivos Gerais e Específicos	20
1.2	Estrutura da Tese	21
2	CONTROLE DE SISTEMAS DINÂMICOS VARIANTES NO TEMPO... ..	22
2.1	Definição do Problema	23
2.2	Controle Baseado em Modelos e Controle Direcionado a Dados	25
2.2.1	Definições	26
2.2.2	Classificações.....	31
2.2.3	Vantagens e Limitações	33
2.2.4	Escolha do DDC-MFAC.....	33
2.3	Controle Adaptativo Livre de Modelo – MFAC	35
2.3.1	Modelo Dinâmico Linear da Forma Compacta para Sistemas SISO.....	35
2.3.2	Projeto de um Controlador MFAC-CFDL para Sistemas Não-Lineares SISO....	37
2.3.3	Sintonia dos Parâmetros.....	38
2.4	Algoritmos Evolucionários.....	39
2.4.1	Algoritmos Evolucionários Multi-objetivo.....	42
2.4.2	Evolução Diferencial	44
2.4.3	Non-Dominated Sorting Differential Evolution - NSDE.....	46
2.5	Modelo para Dinâmica da Distribuição da População no Algoritmo DE.....	48
2.5.1	Distribuição de Indivíduos no Espaço de Soluções	48
2.5.2	Modelo Para Dinâmica dos Agrupamentos da População	50
2.6	Evolução Diferencial em Sistemas de Controle DDC	52
2.7	Discussões Finais	57
3	MODELO DE EVOLUÇÃO DIFERENCIAL E CONTROLE ADAPTATIVO EVOLUCIONÁRIO.....	58
3.1	Concepção do Modelo.....	58
3.2	Descrição dos Algoritmos.....	61
3.2.1	Abordagem Mono-Objetivo.....	61
3.2.2	Abordagem Multi-Objetivo	65

3.3	Controle Adaptativo Evolucionário <i>Off-Line</i>	68
3.4	Controle Adaptativo Evolucionário <i>On-Line</i>	69
3.5	Discussões Finais	73
4	RESULTADOS DOS EXPERIMENTOS	74
4.1	Testes para Validação do Algoritmo	75
4.1.1.	Testes para Problemas Mono-Objetivo.....	75
4.1.2.	Testes para Problemas Multi-Objetivo	82
4.2	Sintonia de Parâmetros de um Controlador MFAC-CFDL	85
4.2.1.	Casos de Estudo para Sintonia <i>Off-Line</i>	86
4.2.2.	Casos de Estudo para Sintonia <i>On-Line</i>	110
4.3	Discussões Finais	121
5	CONCLUSÕES	123
5.1	Trabalhos Futuros	126
	REFERÊNCIAS	127
	Apêndice A - Resumo Teórico sobre Técnicas de Controle DDC	136
	Apêndice B - Modelos Lineares Dinâmicos da Planta Usando MFAC	159
	Apêndice C - Projeto do Controlador MFAC para Sistemas Não-Lineares	166

1

INTRODUÇÃO

A teoria de controle moderno tem como fundamento a construção prévia de um modelo para o sistema estudado. As metodologias de controle que seguem essa premissa, costumam ser denominadas de estratégias de Controle Baseado em Modelo – MBC do inglês *Model Based Control*. Atualmente, a teoria para MBC se encontra bem estabelecida e pode ser desmembrada em diversos ramos, tais como, identificação de sistemas, controle adaptativo, controle robusto, controle ótimo, controle com estrutura variável e teoria de sistemas estocásticos (HOU; WANG, 2013). Apesar dos avanços conseguidos, o MBC depende fundamentalmente da precisão dos modelos para representação dos processos ou das plantas. Essa limitação consiste em desafio de ordem teórica e prática no tratamento de sistemas atuais, os quais, com a evolução tecnológica, estão cada vez mais complexos (HOU; JIN, 2014).

Em resposta a essa limitação, a recente abordagem denominada controle direcionado por dados – DDC do inglês *Data-Driven Control*, pode ser entendida como complementar à teoria de controle moderno, tratando desde problemas de grande complexidade, como sistemas não-lineares variantes no tempo, até problemas, cujo modelo para representar o sistema controlado, não está disponível. Portanto, o controle precisa ser realizado com base apenas nos dados de entrada e saída medidos no processo (HOU; JIN, 2014). Atualmente, na literatura, diversas estratégias de controle podem ser classificadas como DDC, a saber, Controle Proporcional Integral Derivativo (PID); Aproximação Estocástica por Perturbação Simultânea (*Simultaneous Perturbation Stochastic Approximation* - SPSA); Controle Não-falseado (*Unfalsified Control* - UC); Controle por Aprendizagem Iterativa

(*Iterative Learning Control* - IL); e Controle Adaptativo Livre de Modelo (*Model Free Adaptive Control* - MFAC).

Particularmente, a técnica DDC *on-line* denominada Controle Adaptativo Livre do Modelo - MFAC, proposta inicialmente por Hou e Huang (1997) para o controle de sistemas não-lineares e variantes no tempo, usa apenas os dados de entrada e saída adquiridos do processo ou da planta durante sua execução para construir, a cada instante de operação, um modelo linear dinâmico para o sistema. O MFAC usa essa relação linear no cálculo do sinal de controle guiando o sistema real em trajetória prescrita. O projeto de um controlador MFAC, no entanto, envolve a determinação de vários parâmetros, os quais dependem, quase exclusivamente, da experiência do projetista. Simulações para problemas de controle encontrados na literatura, demonstram que a saída do sistema controlado apresenta grande variação para diferentes valores do conjunto de parâmetros do controlador (HOU; JIN, 2014). Sendo assim, a sintonia desses parâmetros permanece como questão em aberto e merece um estudo mais aprofundado (JI *et al.*, 2014).

A sintonia de parâmetros de um controlador pode ser transformada num problema de otimização. No entanto, a solução desse problema pode ser desafio bastante complexo, pois o projeto do sistema de controle pode envolver diversos objetivos, os quais podem inclusive serem conflitantes. Essa multiplicidade de objetivos somada à dificuldade de tratamento para sistemas dinâmicos, não-lineares e variantes no tempo, tende a inviabilizar a aplicação de ferramentas convencionais de otimização, favorecendo o uso de meta-heurísticas para solução desse tipo de problema (REYNOSO-MEZA; BLASCO, 2014).

Algoritmos Evolucionários – AEs, são exemplos de meta-heurísticas baseadas na teoria da evolução das espécies de Darwin, cujo princípio fundamental é a competição entre espécies pelos recursos do ambiente, resultando na maior probabilidade de sobrevivência dos indivíduos mais aptos. Costuma-se classificar os algoritmos de computação evolucionária em quatro paradigmas principais: Algoritmos Genéticos – AG, Estratégias de Evolução – EE, Programação Evolucionária – PE e Programação Genética – PG (EIBEN; SMITH, 2003). Dentre as aplicações dos AEs, destaca-se a solução de problemas de otimização, cuja

complexidade da função objetivo, ou até sua indisponibilidade durante a busca, impede o uso de estratégias clássicas, por exemplo, usando métodos baseados no gradiente da função de otimização.

Na literatura de controle de sistemas, os empregos mais frequentes para AEs são: para ajustar os parâmetros de um controlador com estrutura fixa, para definir a estrutura do controlador, ou realizar ambas tarefas (FLEMING, PURSHOUSE, 2002; REYNOSO-MEZA, GARCIA-NIETO, 2013). No primeiro caso, mais comum, os parâmetros do controlador são escolhidos a partir da otimização de um ou mais índices de performance, os quais sintetizam características favoráveis ao projeto de controle (REYNOSO-MEZA; BLASCO, 2014). Apesar de sua capacidade de tratar problemas complexos, em projetos de controle, os AEs costumam ser usados como ferramenta de otimização *off-line*, *i.e.*, a otimização não ocorre simultaneamente ao funcionamento do sistema controlado. A otimização de parâmetros *on-line* usando AEs enfrenta desafios como: tempo necessário para obter a solução ótima e a estocasticidade do algoritmo, o que pode resultar numa solução inviável ou inadequada ao problema. No entanto, avanços no desenvolvimento de novos algoritmos e capacidade de processamento e velocidade dos equipamentos modernos, pode tornar essa abordagem mais comum, viabilizando a realização de novas pesquisas nesse sentido.

Em relação ao uso de AEs e DDC, outro ponto que também deve ser ressaltado é a dificuldade de calcular a aptidão dos membros da população de soluções, quando não existe um modelo teórico para o sistema real e, portanto, é praticamente impossível simular rapidamente o comportamento do sistema para avaliar a performance de cada solução. O desenvolvimento de novas técnicas DDC e o aprimoramento das técnicas existentes baseadas na previsão do comportamento futuro do sistema, *e.g.*, MFAC Preditivo (HOU, JIN, 2014; JUNWEI *et al.*, 2016), podem contribuir na solução desse ponto, mas é uma questão que demanda mais estudos.

Dentre os AEs atuais, o algoritmo de Evolução Diferencial (*Differential Evolution* – DE) proposto por Storn and Price em 1995, destaca-se graças à sua simplicidade, grande capacidade de exploração em profundidade no espaço de

soluções e eficiência na solução de problemas de otimização no domínio dos números reais, características compatíveis com o projeto de um sistema de controle ótimo. Desde sua apresentação, o algoritmo DE tem recebido atenção da comunidade de otimização. Recentemente, Dragoi e Dafinescu (2016) apresentaram um resumo das principais contribuições recentes, abordando aspectos de adaptação e auto-adaptação dos parâmetros da DE e sua hibridização com outros algoritmos evolucionários, visando aprimorar as características da DE.

O uso de algoritmos DE híbridos na solução de problemas de controle, para melhorar as qualidades do DE canônico, é comum na literatura. Por exemplo, em 2010, Neri e Mininno resolveram o problema de otimização complexa para sistemas com limitações de desempenho computacional usando um algoritmo híbrido denominado McDE (*Memetic Compact Differential Evolution*). O McDE combina a versão do algoritmo DE compacto apresentada, com uma busca local em torno das soluções geradas. No trabalho, o McDE é usado para calcular os pesos ótimos de uma rede neural recorrente que funciona como um observador de distúrbio para o sistema de controle de um robô.

Mirkhani *et al.* (2013) propuseram um algoritmo híbrido baseado em DE e no algoritmo de otimização denominado de Busca Harmônica-BH para controle de localização de robôs móveis. O algoritmo BH é uma meta-heurística usada para solução de problemas de otimização, baseada no processo de improvisação musical, no qual os músicos improvisam os sons de seus instrumentos procurando um perfeito estado de harmonia. Infelizmente, o BH muitas vezes, converge lentamente. No algoritmo, cada nova harmônica (solução) é gerada de forma aleatória ou considerando toda a população de harmônicas existentes, característica que ao mesmo tempo dificulta estagnação em ótimo local e desacelera a convergência. Para resolver esse problema, os autores propuseram o algoritmo HIDE – (*Harmony Memory Improvement with Differential Evolution*) que utiliza o esquema DE/*rand-to-best*/1 para também atualizar a população de soluções e assim acelerar o processo de otimização.

Coelho *et al* (2015) apresentaram uma modificação à DE canônica para sintonia de parâmetros de um controlador, aplicada ao controle de processos

químicos, denominada IDE (*Improved DE*). Este algoritmo tem como característica avaliar a diversidade da população em relação aos dois vetores que são sorteados para formar o vetor diferença e, de acordo com a avaliação, modificar o sentido do vetor diferença, promovendo mais exploração ou exploração em profundidade, conforme a diversidade calculada. Juang et al. (2015) propuseram o algoritmo AGDE (*Adaptive Group-Based Differential Evolution*), baseado em evolução diferencial, que introduz o conceito de grupos para geração de vetores mutantes, visando com isso, acelerar a busca por soluções ótimas. O algoritmo foi usado para sintonia dos parâmetros das funções de pertinência de um controlador por lógica difusa no controle de movimentos de um robô hexapode.

O uso de DE para sintonia *on-line* de parâmetros não é muito comum em problemas de controle, mesmo assim, alguns casos foram achados na literatura. Tajuddin et al. (2015) usaram o esquema de mutação DE/*rand*/1/bin para solução do problema de rastreamento *on-line* do ponto de maximização de potência em sistemas fotovoltaicos (MPPT - *Maximum Power Point Tracking*). Neste caso, a estratégia foi adequada, pois a dinâmica do problema MPPT é considerada lenta o suficiente para permitir ao algoritmo encontrar a solução ótima antes que um novo ponto ótimo exista. Mishra et al. (2015) apresentaram o problema de sintonia *on-line* dos parâmetros do controlador PI para compensar o fenômeno de “*stiction*” comum em válvulas de controle pneumáticas. Na otimização *on-line* foi usado o algoritmo DE canônico, executado por poucas gerações (no máximo dez) e, para avaliação *on-line* dos candidatos, um modelo virtual do sistema foi construído usando LABVIEW, permitindo avaliar as aptidões das soluções, antes da solução ótima ser inserida ao sistema real. Recentemente, Negri et al. (2016) usaram com sucesso a DE para otimização *on-line* da função de custo usada no controle preditivo baseado em modelos, denominado de DE-MPC (*Differential Evolution Model Predictive Control*), para um sistema físico de controle de pressão de uma bancada didática.

No presente trabalho, buscou-se desenvolver um AE multi-objetivo capaz de viabilizar a otimização *off-line* e *on-line* dos parâmetros de um controlador, baseado em DDC, a ser aplicado em sistemas não-lineares variantes no tempo. Para tanto, foram inicialmente realizados estudos sobre as principais técnicas DDC, com ênfase no controle *on-line* e escolhida a técnica DDC MFAC. O MFAC é bastante flexível,

pois é adequada ao tratamento de sistemas não-lineares, variantes no tempo, SISO ou MIMO, além disso, conta com um extenso arcabouço teórico com registros recentes de contribuições na literatura de controle e também aplicações práticas (JALALI et al 2013; XUHUI et al, 2013; HOU, JIN, 2014; LENG et al, 2014; XU et al 2014; ZHU, HOU, 2014; PEZESHKI et al 2015; JUNWEI et al, 2016). Depois, buscou-se na literatura exemplos de AEs aplicados com sucesso no projeto de sistemas de controle e adaptáveis a problemas multi-objetivo, atendendo a uma possível diversidade nos critérios a serem definidos no projeto de um sistema de controle para os casos de estudo a serem analisados neste trabalho.

O algoritmo escolhido DE atende aos critérios definidos na pesquisa por AEs e, além disso, apresenta simplicidade de implementação e baixo custo computacional. Procurou-se, então, compreender a dinâmica da população de soluções durante a execução do algoritmo, visando identificar e reforçar características capazes de melhorar a performance da DE. Dentre as características observadas, a tendência a formação de agrupamentos e a direção do percurso seguido pelos centros dos grupos formados, entre as gerações, foram escolhidas para serem reforçadas em um novo algoritmo, mediante a geração externa de novos indivíduos, visando ocupar regiões identificadas como promissoras.

A introdução exógena de indivíduos denominados imigrantes para melhorar a diversidade da população e evitar convergência prematura, já tem registro na literatura (YANG; TINOS, 2007). Nesta Tese, no entanto, esses indivíduos não são introduzidos diretamente na população corrente em substituição a outros membros da população. A população de imigrantes será apenas combinada com a população atual para o sorteio dos indivíduos usados na formação dos vetores mutantes da DE. Neste caso, os imigrantes não precisam ser avaliados e possuem função dupla, a saber: introduzir diversidade à população, pela manutenção de um percentual fixo mínimo de imigrantes aleatórios, enquanto aumenta a probabilidade de exploração de regiões mais favoráveis a busca, mediante inserção de imigrantes direcionados. Com base nessa estratégia, foram elaborados dois algoritmos: um para otimização mono-objetivo e outro para multi-objetivo. Experimentos foram realizados, usando problemas encontrados da literatura, para validar os algoritmos propostos, antes de sua aplicação ao projeto de um o controlador.

No projeto do controlador, buscou-se otimizar a capacidade de rastreamento do sistema de controle para um sinal de referência conhecido. Para tanto, foram adotados dois índices de desempenho, um associado à redução do erro em estado estacionário e o outro relacionado com o valor do sobressinal da resposta do sistema. Em testes realizados mediante simulações verificou-se o conflito na otimização dessas funções objetivo propostas. Também foram desenvolvidas duas estratégias de controle: na primeira, os parâmetros do controlador foram ajustados *off-line*, ou seja, no algoritmo multi-objetivo proposto, cada indivíduo foi avaliado simulando o sistema controlado durante todo período de operação da planta e o resultado ao final da otimização, definirá os parâmetros usados no controlador real para o sistema em funcionamento. Na estratégia *on-line*, o controlador usa durante a operação os parâmetros otimizados na estratégia anterior, mas também realiza otimizações em intervalos menores, enquanto o controle atua, quando algumas situações são detectadas. Por exemplo, alguns instantes antes da função do sinal de referência mudar bruscamente ou no instante que alguma perturbação acentua o sinal de erro.

Experimentos foram realizados controlando diversos sistemas encontrados na literatura e os resultados mostraram que o controlador MFAC com parâmetros otimizados *off-line*, obtiveram resultados melhores, do que os obtidos usando parâmetros definidos na literatura. Inclusive, a versão do controlador usada, o MFAC-CFDL (*Model Free Adaptive Control – Compact Form Dynamic Linearization*), definida mediante a forma mais simplificada do modelo linear dinâmico da planta, foi superior a abordagens mais complexas do MFAC, quando são usados parâmetros escolhidos com base na experiência do projetista.

Também foram efetuados outros experimentos, usando os mesmos sistemas já testados, no entanto, com os parâmetros otimizados *on-line*, durante a operação do sistema. Comparando os resultados obtidos nas duas estratégias de sintonia, verificou-se que a estratégia *on-line* conseguiu melhorar os resultados obtidos com a otimização *off-line*.

1.1 Objetivos Gerais e Específicos

Esta Tese tem como primeiro objetivo geral contribuir com a teoria do algoritmo DE, pelo controle da distribuição da população no espaço de soluções, baseado na introdução de indivíduos produzidos externamente.

Adicionalmente, esta Tese também tem como objetivo geral contribuir para estado da arte do uso de AEs no projeto de sistemas de controle, desenvolvendo estratégias viáveis para a sintonia *off-line* e *on-line* dos parâmetros de um controlador com estrutura fixa.

Para alcançar esses objetivos gerais esta pesquisa tem os seguintes objetivos específicos:

- Compreender a dinâmica da evolução da população durante as etapas de execução do algoritmo DE;
- Observar na dinâmica, características capazes de permitir a identificação de regiões promissoras que deveriam ser povoadas;
- Propor uma política de produção e introdução de indivíduos externos na população que não impacte na quantidade total de avaliações de soluções executadas durante o processo de otimização;
- Desenvolver e validar uma variação do algoritmo DE usando a política de imigrantes proposta;
- Adaptar o algoritmo proposto a problemas de otimização multi-objetivo e validar com problemas da literatura;
- Usar a versão multi-objetivo do algoritmo para sintonia *off-line* dos parâmetros de um controlador MFAC-CFDL
- Comparar os resultados obtidos com os parâmetros ótimos com dados obtidos usando parâmetros encontrados na literatura, ou mesmo outras versões do MFAC;
- Desenvolver uma estratégia de sintonia *on-line* para os parâmetros do controlador e comparar seu desempenho frente à sintonia *off-line*.

1.2 Estrutura da Tese

Visando atingir aos objetivos propostos, este documento foi dividido em mais quatro capítulos. No Capítulo 2, será apresentado um breve estado da arte da teoria de controle de sistemas dinâmicos variantes no tempo ou sujeitos a incertezas, ressaltando as diferenças entre os métodos de controle baseado em modelos e os métodos baseado em dados. Ainda no Capítulo 2, também será apresentado um resumo com as contribuições dos algoritmos evolucionários, com ênfase no algoritmo DE, para o controle de sistemas.

No Capítulo 3, inicialmente, descreve-se o modelo conceitual proposto para melhorar a performance do algoritmo DE. Em seguida, versões para o algoritmo evolucionário proposto para problemas de otimização mono e multi-objetivo, serão apresentadas. Finalmente, os detalhes sobre o processo de sintonia *off-line* e *on-line* para os parâmetros do controlador usado na pesquisa, serão abordados.

Já no Capítulo 4, na primeira parte, os resultados com testes para validação dos algoritmos propostos, usando problemas clássicos da literatura, são mostrados. Na parte final do capítulo, simulações com problemas de controle, usando a estratégia de sintonia dos parâmetros do controlador *off-line* e depois a *on-line*, permitirão a comparação da performance dessas estratégias.

Na Conclusão, primeiramente, uma visão geral do trabalho é apresentada, onde os detalhes das abordagens propostas serão discutidos. Ao final, as propostas para trabalhos futuros serão elencadas para dar andamento a essa linha de pesquisa.

2

CONTROLE DE SISTEMAS DINÂMICOS VARIANTES NO TEMPO

O aumento da complexidade dos processos industriais modernos torna o tratamento de sistemas dinâmicos variantes no tempo, um desafio cada vez mais comum à teoria de sistemas de controle. Por outro lado, a complexidade dos modelos usados para representar esses processos no projeto do controlador, também tem justificado o desenvolvimento de novas estratégias de controle, cuja ênfase está nos dados coletados da planta durante toda a operação. O desenvolvimento de abordagens livre de modelos, também fomenta a aplicação de meta-heurísticas, para problemas de controle ótimo, em virtude de sua flexibilidade para solução de problemas de otimização complexos (HOU; WANG, 2013).

Neste capítulo, inicialmente será apresentada uma descrição dos desafios encontrados pela teoria de controle moderno quando aplicada a sistemas dinâmicos variantes no tempo, com apresentação das principais diferenças entre as estratégias MBC e DDC. Além disso, as técnicas DDC disponíveis atualmente e suas classificações, com ênfase na estratégia Controle Adaptativo Livre de Modelo - MFAC, também serão retratadas. Depois, um breve resumo sobre algoritmos evolucionários aplicados a problemas de otimização, com destaque ao algoritmo DE, é apresentado. Em seguida, detalhes sobre as características da dinâmica da população durante o processo de otimização serão discutidas. Por fim, serão elencadas algumas contribuições relevantes, encontradas na literatura, que mesclam a teoria de controle e os AEs, com foco em DDC e DE.

2.1 Definição do Problema

A última metade do século passado testemunhou importantes avanços na área de controle moderno. Tais avanços, possibilitaram a ramificação da teoria de controle em diversas áreas, a saber, identificação de sistemas, controle clássico, controle adaptativo, controle robusto, controle ótimo, controle com estrutura variável, e controle estocástico (HOU; JIN, 2014). Cada um desses ramos, tem sido objeto de pesquisas e aplicações práticas ao longo dos últimos anos, usos que mostram sua aplicabilidade no controle de processos industriais, aeroespaciais, sistemas de tráfego, dentre outros (HOU; WANG, 2013). Entretanto, a cada nova conquista, novos desafios de ordem teórica e prática, se apresentam no tratamento dos sistemas contemporâneos, cuja dimensão e complexidade também aumentam a passos semelhantes.

Um ponto de interseção a todos os ramos da teoria de controle moderno é a necessidade inicial de modelagem ou identificação do sistema estudado, seguido do projeto do controlador a partir do modelo. Pode-se então dizer que esta teoria é baseada em modelos ou MBC e tem como premissa chave o princípio da equivalência certa, ou seja, o modelo deve representar acuradamente o sistema estudado. Na prática, o processo de modelagem considera abstrações do modelo para o sistema real, nessa etapa, a presença de erros é bem possível. Além disso, dinâmicas e perturbações ao sistema não previstas ocorrem frequentemente em aplicações práticas. Essas características tornam o sistema real em malha fechada com um controlador do tipo MBC inerentemente menos seguro ou robusto (HOU; WANG, 2013).

Existem pesquisas para reduzir tais inconvenientes e preservar as vantagens do MBC. Particularmente, na teoria de controle robusto vários modelos para o erro têm sido considerados, tais como, descrições de erro aditivo, multiplicativo, ou erro dentro de limites predefinidos (ROHRS *et al.*, 1982). Tais descrições, além de serem suposições, podem não produzir resultados consistentes

com os métodos físicos e matemáticos para modelagem de sistemas. Portanto, ainda falta uma descrição precisa e aplicável para incertezas de sistema para uso em controle robusto.

Hou e Wang (2013) citam como outro inconveniente do MBC a relação direta entre a acurácia do modelo e o aumento da complexidade do projeto de controle. Modelos mais acurados, capazes de representar fielmente em todos os detalhes o sistema original durante todo período de operação, tendem a ter uma formulação matemática mais complexa. Infelizmente modelos matemáticos muito complexos normalmente resultam em sistemas de controle também mais complexos. A complexidade do sistema de controle é um fator que costuma elevar os custos com projeto, manutenção e operação do sistema. Uma solução para esses problemas seria simplificar o modelo para facilitar o projeto do controlador, no entanto, isso provavelmente anularia o benefício da acurácia do mesmo.

O aumento da complexidade nos processos modernos vem dificultando o uso de técnicas de controle baseadas no modelo de processos que podem ser caracterizados por gerarem altas quantidades de dados relevantes durante sua operação. Tais dados podem ser empregados *on-line*, *off-line* ou de ambas formas para determinar um controlador, calcular e prever os estados do sistema, tomar decisões, avaliar seu desempenho, ou diagnosticar falhas (HOU; JIN, 2014). Portanto, o desenvolvimento do DDC representa um avanço importante no tratamento dos sistemas complexos

2.2 Controle Baseado em Modelos (MBC) e Controle Direcionado a Dados (DDC)

O diagrama da Figura 1, apresentado por Hou e Wang (2013), resume a interação entre os elementos do projeto de um sistema de controle usando o MBC:

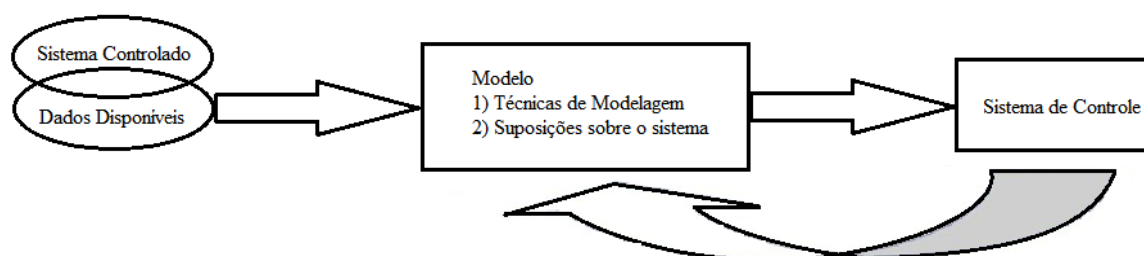


Figura 1 - Relação do sistema controlado, modelo e o controle no MBC (HOU; WANG, 2013).

A Figura 1 mostra que no MBC o sistema de controle é projetado para o modelo e não para o sistema real. O modelo deveria representar corretamente o processo, no entanto, foi construído com base em suposições sobre o sistema real e em técnicas de modelagem, antes de ser validado com os dados reais medidos do sistema controlado. Portanto, o ponto crucial na teoria MBC é saber se as diferenças entre a planta real e seu modelo tendem a desaparecer, após o projeto do controlador, ou se podem aumentar de alguma maneira dificultando o melhor desempenho do sistema controlado.

Em contraposição, o diagrama da Figura 2, também apresentada por Hou e Wang (2013), ilustra a arquitetura DDC, na qual se ressalta que o projeto do controlador DDC e a análise do sistema controlado são ambos realizados usando somente os dados medidos de entrada e saída para o sistema em malha fechada.

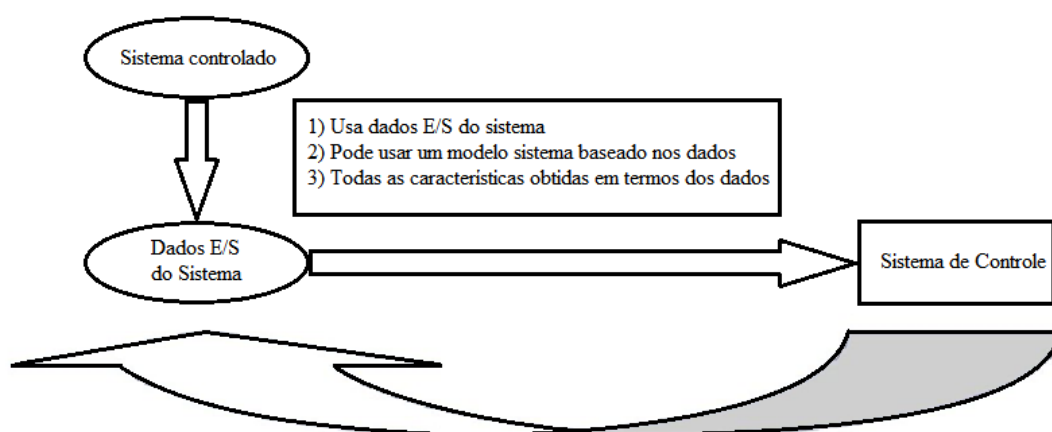


Figura 2 - Relação entre o sistema controlado, os dados E/S e o controle, (HOU; WANG, 2013).

2.2.1. Definições

Hou e Wang (2013) apresentaram algumas definições para a teoria de controle DDC, as quais foram resumidas pelos autores na seguinte definição final:

Controle Direcionado por Dados inclui todas as teorias de controle e métodos, nos quais o controlador é projetado diretamente usando dados E/S on-line ou off-line do sistema controlado ou o conhecimento do processamento de dados processados não empregando qualquer informação explícita de um modelo matemático do processo controlado, e cuja estabilidade, convergência, e robustez podem ser garantidas por análise matemática rigorosa válida sob suposições razoáveis. (HOU; WANG, 2013).

Considerando essa definição, a seguir um resumo das principais técnicas DDC atuais é apresentado. O Apêndice A deste trabalho, contém mais aspectos teóricos das técnicas elencadas.

a) SPSA-based DDC method (SPSA)

O acrônimo SPSA significa *Simultaneous Perturbation Stochastic Approximation*. Essa técnica DDC proposta inicialmente por Spall (1992) baseia-se no projeto do controlador como sendo uma função de aproximação da planta, para tanto, usa os dados do processo medidos *on-line*. A estrutura do controlador é fixa e

seus parâmetros são calculados otimizando um índice de custo função dos sinais de saída e de controle. No processo de otimização, o algoritmo *SPSA* é usado para estimação do gradiente do índice de custo. Na literatura, aplicações industriais (SPALL; CRISTION, 1997; ZHAO et al, 2015) e em controle de tráfego (SPALL; CHIN, 1997; LU et al, 2015) podem ser listadas.

b) Model-Free Adaptive Control (MFAC)

Apresentada por Hou e Huang (1997), essa estratégia de controle DDC também usa dados medidos *on-line* do processo para o controle. O método se aplica a sistemas discretos e não-lineares e se baseia na construção de um modelo dinâmico linearizado virtual para o comportamento de saída de uma planta, em cada instante do processo, visando substituir a modelagem do sistema real, no projeto do controlador. Para tanto, o autor define o conceito de pseudo-derivada parcial (*Pseudo Partial Derivative* – PPD), que no caso de sistemas SISO (*Single Input Single Output*), pode ser interpretada como estimativas para inclinação da reta tangente na equação real da planta durante o processo. A PPD não pode ser calculada analiticamente, mas pode ser estimada usando os dados medidos de entrada e saída da planta. A técnica MFAC prevê três tipos de modelos lineares, *Compact Form Dynamic Linearization* - CFDL, *Partial Form Dynamic Linearization* - PFDL e *Full Form Dynamic Linearization* – FFDL, para atender aos diversos níveis de não linearidades presentes no sistema real. A descrição dos modelos e sua aplicação no controle de sistemas SISO e MIMO (*Multiple Input Multiple Output*) está detalhada no Apêndice B deste trabalho. Versões desta técnica tem alcançado sucesso no controle de diversos sistemas (COELHO, COELHO, 2009; COELHO et al., 2010; HOU, JIN, 2014).

c) Unfalsified Control (UC)

Inicialmente proposta por Safonov e Tsao (1997), baseia-se no Princípio da Falseabilidade de Karl Popper aplicado ao controle. Segundo esse princípio, para uma afirmação ser falseável, é necessário que exista pelo menos um experimento ou observação possível que, fornecendo determinado resultado, implique a falsidade da afirmação inicial. O UC, por sua vez, atua recursivamente falseando conjuntos de parâmetros, associados à estrutura fixa de um controlador, que falham em satisfazer

especificações de performance adotadas no projeto e com isso determina o conjunto de parâmetros adequados ao controle. O método inclui três elementos básicos: um conjunto de controladores candidatos inversíveis, especificações de custo mensuráveis e um mecanismo comutador. O UC tem sido usado com sucesso em diversos campos tais como, controle de braço de robôs, orientação de mísseis e processos industriais (SAFONOV, 2003).

d) Controlador PID (Proportional Integral Derivative)

Controladores PID atuam tentando minimizar o sinal de erro da planta mediante ações do tipo derivativa, integral e proporcional. É uma estratégia de controle clássica, cuja ação de controle é calculada a partir do sinal de erro medido. O processo de sintonia dos seus parâmetros normalmente é feito a partir de simulações usando um modelo para o sistema controlado, o que poderia classifica-lo como uma técnica MBC, no entanto, considerando dois pontos chave da definição apresentada para DDC, ou seja, projeto do controlador com base nos dados E/S medidos da planta e uso de um modelo do processo apenas de forma implícita, o PID pode ser considerado como primeiro método DDC no mundo. As diversas estratégias de sintonia do PID, inclusive mesclando o controle clássico com técnicas modernas de controle inteligente, ainda podem ser acompanhadas nos periódicos na área de controle atualmente (MOHANTY *et al.*, 2014; SAHU *et al.*, 2013; REYNOSO-MEZA *et al.*, 2012; CHEN, HUANG, 2004; KROHLING, REY, 2001). Portanto, o campo de pesquisas usando PID ainda deve ter um futuro longínquo.

e) Iterative Feedback Tuning – IFT

Apresentado por Hjalmarsson *et al.* (1998) é um método DDC iterativo baseado na medição dos dados *off-line* do sistema em malha fechada usando um controlador de estrutura fixa. Os parâmetros são ajustados a cada iteração usando o método do gradiente visando a minimização de uma função de custo. No processo de otimização a estimativa do gradiente é realizada mediante realização de dois experimentos: no primeiro, os dados são colhidos do sistema usando como sinal de referência a saída desejada, e no segundo, os dados são medidos usando como sinal de referência a diferença entre a saída desejada e a saída obtida no primeiro

experimento. O IFT apresenta aplicações no controle de processos industriais (HJALMARSSON, 2002).

f) Correlation-Based Tuning - CbT

Proposto por Karimi em 2002 (HOW; WANG, 2013) é um método DDC iterativo inspirado na ideia de aproximação por correlação usado na identificação de sistemas. O método consiste em ajustar iterativamente os parâmetros de um controlador com estrutura fixa, tendo por objetivo descorrelacionar, ou reduzir a correlação, entre os sinais de referência e de erro. Para tanto, o sinal de erro é definido como a diferença entre a saída do sistema real controlado e a saída de um modelo ideal para o sistema. O CbT é aplicável a sistemas MIMO e na literatura são encontradas referências de seu uso para o controle de sistemas de suspensão de veículos (MISKOVIC *et al.*, 2003).

g) Virtual Reference Feedback Tuning – VRFT

O VRFT foi inicialmente proposto por Gardabasi e Savaresi (2000), aplicado a sistemas Lineares Invariantes no Tempo - LIT. No VRFT o projeto do controlador é transformado num problema de identificação dos parâmetros do controlador, mediante a aproximação de dois sinais de saída: o primeiro, do sistema real em malha fechada e o segundo, de um modelo de referência adotado. O método define uma função de custo cuja minimização aproxima os dois sistemas. Para compensar a ausência de modelo da planta, VRFT propõe uma solução alternativa para esse problema. Se o modelo de referência for inversível, é possível calcular um conjunto de dados virtuais a partir dos dados de entrada e saída medidos da planta em malha fechada. Com esses dados calculados é possível elaborar um segundo índice de desempenho, cujo mínimo em algumas situações é igual ao da função de custo inicial (HOU; WANG, 2013). Aplicações na literatura são encontradas para sistemas de suspensão ativa (CAMPI *et al.*, 2002), controle de velocidade e aplicações industriais (CAMPI *et al.*, 2002).

h) Noniterative Data-Driven Model Reference Control - NDDMRC

Apresentada por Karimi *et al.* (2007), este método, assim como no VRFT, também transforma o projeto do controlador num problema de identificação. No

entanto, diferente dos problemas de identificação padrão, no NDDMRC o sinal de entrada é afetado pelo ruído e não o de saída. Novamente, deseja-se aproximar a saída do sistema real em malha fechada da saída de um sistema de referência usando uma função de custo. A equação do sistema de referência será obtida pela ligação em malha fechada da planta real e o controlador ideal. Na ausência da equação da planta, não há garantias de que a função de custo seja convexa em relação aos parâmetros do controlador. Entretanto, se o controlador for parametrizável linearmente, o NDDMRC aproxima a função de custo por outra função que é convexa, cujo mínimo, se a planta for estável, pode ser obtido otimizando a norma de um sinal do erro obtido para o sinal de saída da planta ruidoso. Portanto, o problema de sintonia usando NDDMRC resume-se a calcular o sinal de saída do controlador capaz de minimizar o sinal de erro e depois usar esse sinal para identificar os parâmetros do controlador real para planta.

i) Subspace Approach – SA

A ideia fundamental da abordagem SA (HUANG; KADALI, 2008) e de outras duas semelhantes, *the data space approach* (FUJISAKI *et al.*, 2004) e a *data-driven simulation approach* (MARKOVSKY *et al.*, 2005), baseia-se na hipótese de que a dinâmica do sistema pode ser representada por um subespaço num espaço vetorial de dimensão finita, formado pelo conjunto de dados entrada-saída ou entrada-estado-saída do sistema. Na aproximação por subespaço é utilizada a representação por variáveis de estado. Nesse caso, considera-se que a base do espaço vetorial, também chamada de matriz dinâmica, envolve toda a informação dinâmica do sistema LIT. Diferentes técnicas de identificação por subespaço estão disponíveis na literatura, as quais diferem normalmente na forma como a matriz dinâmica é estimada. Ferramentas numéricas usadas para a estimação dessas bases incluem decomposição de valores singulares, decomposição - QR e análise de variáveis canônicas.

j) Approximate Dynamic Programing – ADP

Proposta em 1991 por Werbos (1991), a técnica ADP combina aprendizado por reforço, usando estruturas adaptativas críticas, com programação dinâmica. Os principais esquemas para a ADP incluem: programação dinâmica heurística,

programação dinâmica dual heurística, programação dinâmica heurística ação-dependente (*Q-learning*), e programação dinâmica heurística dual ação-dependente.

k) Iterative Learning Control – ILC

Proposta em 1978 por Uchiyama (HOU, WANG, 2013) aplica-se ao tratamento de sistemas que repetem a mesma tarefa continuamente dentro de um intervalo finito. O ILC usa a repetição das tarefas para melhorar a performance do controlador. Este método requer pouco conhecimento do sistema e pode garantir convergência no erro de aprendizagem quando o número de interações tende ao infinito. O ILC usa dados coletados do sistema *on-line* e *off-line*, nas iterações passadas e nos instantes anteriores da iteração atual, para calcular o sinal de controle ótimo a cada instante. Atualmente o ILC tem sido aplicado em diversas áreas de aplicação (XU; HOU, 2005).

l) Lazy Learning - LL

LL são algoritmos de aprendizagem de máquina supervisionado aplicados inicialmente por Shaal e Atkeson (1994) na solução de problemas de controle. Normalmente a meta desses algoritmos é encontrar uma relação entre dos dados de entrada e saída de um conjunto de treinamento. O LL usa os dados da planta para construir uma relação linear local dinâmica para um sistema não linear e projeta um controlador local para o sistema usando como modelo essa relação.

2.2.2. Classificações

As técnicas DDC atuais podem ser classificadas de acordo com o uso dos dados adquiridos para o controle. Na Tabela 1 é mostrado um quadro com as técnicas apresentadas neste Capítulo seguindo essa classificação.

Tabela 1 - Classificação quanto ao uso de dados.

Classificação de acordo com o uso dos dados I/O medidos	
Uso dos dados	Técnica DDC
Baseado em dados <i>on-line</i>	<i>SPSA-based DDC methods</i> (SPSA)
	<i>Model-free adaptive control</i> (MFAC)
	<i>Unfalsified control methodology</i> (UC)
Baseado em dados <i>off-line</i>	<i>PID control method</i>
	<i>Iterative feedback tuning</i> (IFT)
	<i>Correlation-based tuning</i> (CbT)
	<i>Virtual reference feedback tuning</i> (VRFT)
	<i>Noniterative data-driven model reference control</i> (NDDMRC)
	<i>Subspace approach</i> (SA)
	<i>Approximate dynamic programming</i> (ADP)
Baseado em dados <i>on</i> e <i>off-line</i>	<i>Iterative learning control</i> (ILC)
	<i>Lazy learning</i> (LL)

Analisando a Tabela 1 é possível observar que no DDC é mais comum o uso dos dados de forma *off-line*, e.g., PID, IFT, CbT, VRFT, SA e ADP. Neste caso, no projeto do controlador deve existir uma fase inicial para coleta de dados *off-line*, os quais serão usados no processo de ajuste dos parâmetros do controlador. Ao final do projeto, os parâmetros definidos serão usados durante toda a operação do sistema controlado real. Em contrapartida, em outras técnicas DDC, os dados adquiridos *on-line* podem ser usados para o ajuste em tempo de execução dos parâmetros do sistema de controle, a saber, o MFAC e o SPSA, ou para definir a escolha de qual controlador usar a cada instante, a partir de um banco de controladores disponíveis, como é o caso do UC. Além disso, dados obtidos *on-line* também podem ser usados para ajuste fino de parâmetros já sintonizados mediante uso de dados *off-line*, por exemplo, como é o caso das técnicas ILC e LL.

2.2.3. Vantagens e Limitações

A exclusão da etapa de modelagem do sistema para o projeto do controlador; a igualdade de tratamento para os dados obtidos por uma implementação no laboratório ou em campo; e a possibilidade de cooperação com outras técnicas de controle inclusive MBC são vantagens claras para o uso do DDC num projeto de controle. Além disso, técnicas como o PID contam com praticidade de projeto, facilidade de operação, baixo uso de dados medidos para o controle e aplicações em diversos processos industriais modernos. Ressalte-se, no entanto, que o DDC normalmente apresenta desempenho inferior ao MBC quando existe um modelo acurado para o sistema, portanto sua indicação é para problemas de controle que envolvem modelos com elevado grau de incerteza, ou cujo modelo matemático é muito complicado, ou mesmo quando não existe um para o sistema (HOU; WANG, 2013).

2.2.4. Escolha do DDC-MFAC

Na Tabela 2 é apresentado um quadro resumo comparativo entre as principais técnicas DDC. Neste quadro, é possível observar que teoricamente sistemas não-lineares e variantes no tempo não são uma limitação às estratégias DDC, exceto para IFT, CbT, VRFT, NDMRC e SA, cuja formulação usa como premissa que a planta é um sistema LIT. Além disso, as técnicas MFAC, SPSA, UC e LL contam com a capacidade de adaptação *on-line* ao processo o que é uma vantagem no tratamento de sistemas variantes no tempo. Um problema presente no DDC, no entanto, diz respeito à necessidade de processamento de grandes quantidades de dados, como é o caso das estratégias LL, ILC e SA. Contudo, em técnicas como PID e MFAC o processamento dos dados requeridos pode ser considerado baixo, pois poucos parâmetros precisam ser calculados.

Tabela 2 – Resumo Comparativo entre os principais métodos DDC.

Técnica	Uso de Dados	Tipos de Sistemas Indicados	Processamento dos Dados	Estrutura do Controlador
SPSA	<i>On-line</i>	<i>Não-lineares variantes no tempo</i>	<i>Médio</i>	<i>Fixa</i>
MFAC	<i>On-line</i>	<i>Não-lineares variantes no tempo</i>	<i>Baixo</i>	<i>Desconhecida</i>
UC	<i>On-line</i>	<i>Não-lineares variantes no tempo</i>	<i>Alto</i>	<i>Fixa</i>
PID	<i>Off-line</i>	<i>Lineares invariantes no tempo</i>	<i>Baixo</i>	<i>Fixa</i>
IFT	<i>Off-line</i>	<i>Lineares invariantes no tempo</i>	<i>Alto</i>	<i>Fixa</i>
CbT	<i>Off-line</i>	<i>Lineares invariantes no tempo</i>	<i>Alto</i>	<i>Fixa</i>
VRFT	<i>Off-line</i>	<i>Lineares invariantes no tempo</i>	<i>Alto</i>	<i>Fixa</i>
NDMRC	<i>Off-line</i>	<i>Lineares invariantes no tempo</i>	<i>Alto</i>	<i>Fixa</i>
SA	<i>Off-line</i>	<i>Lineares invariantes no tempo</i>	<i>Intenso</i>	<i>Desconhecida</i>
ADP	<i>Off-line</i>	<i>Não-lineares variantes no tempo</i>	<i>Alto</i>	<i>Fixa</i>
ILC	<i>Off/On-line</i>	<i>Não-lineares variantes no tempo</i>	<i>Intenso</i>	<i>Desconhecida</i>
LL	<i>Off/On-line</i>	<i>Não-lineares variantes no tempo</i>	<i>Intenso</i>	<i>Desconhecida</i>

Examinando a Tabela 2, verifica-se que o MFAC não está limitado à aplicação a sistemas simples ou LIT, usa os dados medidos *on-line* para o controle e requer um baixo processamento de dados. Além disso, seu arcabouço teórico se aplica a sistemas SISO e MIMO e conta com provas para estabilidade do método em problemas de regulação ótima (HOU; JIN, 2014). Portanto, para este trabalho, escolheu-se a técnica DDC-MFAC, pois, além das qualidades já descritas, o problema da sintonia dos parâmetros do controlador MFAC ainda está em aberto e merece a atenção de novos estudos (JI et. al., 2014).

2.3 Controle Adaptativo Livre de Modelo – MFAC

A técnica de controle MFAC tem recebido ao longo dos anos bastante atenção na literatura (HOU, HUANG, 1997; WANG *et al.*, 2014; COELHO, COELHO, 2009; COELHO *et al.*, 2009). O MFAC usa os dados de E/S medidos da planta para construir seu modelo virtual dinâmico linear a cada instante. Para tanto, o MFAC define o conceito de Pseudo-Derivada Parcial (*Partial Pseudo Derivative* – PPD), que pode ser interpretada como estimativa do valor das derivadas da equação do modelo em cada ponto de operação. Infelizmente, a PPD não pode ser calculada analiticamente. Entretanto, pode ser estimada usando os dados medidos do sistema. O MFAC usa esse modelo virtual para calcular o sinal de controle para o sistema real.

Para facilitar a compreensão do trabalho, nas seções subsequentes o DDC-MFAC será abordado usando apenas o modelo de linearização dinâmica compacto – CFDL (HOU; HUANG, 1997). Nos Apêndices B e C, deste trabalho, será apresentado um resumo teórico dos demais modelos lineares usados e as etapas do projeto de um controlador MFAC, respectivamente, aplicado a sistemas SISO e MIMO.

2.3.1. Modelo Dinâmico Linear da Forma Compacta para Sistemas SISO

Considere um sistema não-linear SISO discreto no tempo e descrito pela Eq. (2.1)

$$y(k+1) = f(y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)), \quad (2.1)$$

onde $u(k)$ e $y(k)$ são a entrada de controle e a saída da planta no instante k , respectivamente, n_u e n_y são dois números inteiros positivos desconhecidos, e $f(\dots): R^{n_y+n_u+2} \rightarrow R$ é uma função não-linear desconhecida.

Antes de apresentar o modelo é necessário fazer algumas suposições:

SUPosição 2.1

A derivada parcial de $f(\dots)$ em relação à $(n_y + 2)$ -ésima variável é contínua, para todo k com finitas exceções.

SUPosição 2.2

O sistema representado pela Eq. (2.1) satisfaz a condição generalizada de Lipschitz, para todo k com finitas exceções, isto é,

$$|y(k_1 + 1) - y(k_2 + 1)| \leq b|u(k_1) - u(k_2)| \quad (2.2)$$

para $u(k_1) \neq u(k_2)$ e qualquer $k_1 \neq k_2$, $k_1, k_2 > 0$, onde $y(k_i + 1) = f(y(k_i), \dots, y(k_i - n_y), u(k_i), \dots, u(k_i - n_u))$, $i = 1, 2$ e b é uma constante positiva.

A Suposição 2.1 é uma restrição típica imposta ao projeto de controle para sistemas não-lineares. Já a Suposição 2.2, limita a taxa de variação do sinal de saída do sistema controlado em relação a uma variação simultânea no sinal de controle. Garantindo, portanto, que o sistema não produza uma resposta de amplitude infinita, se a mudança no sinal de controle for finita (HOU;JIN,2014).

Para o modelo linear dinâmico, defina também as seguintes variações, $\Delta u(k) = u(k) - u(k - 1)$ e $\Delta y(k + 1) = y(k + 1) - y(k)$. O Teorema 2.1, a seguir, apresenta o modelo CFDL.

TEOREMA 2.1

Considere um sistema não-linear satisfazendo às Suposições 2.1 e 2.2. Se $|\Delta u(k)| \neq 0$, então deve existir um parâmetro variante no tempo $\phi_c(k) \in R$, denominado Pseudo Derivada Parcial – PPD, tal que o sistema (2.1) possa ser transformado no seguinte modelo Linear Dinâmico da Forma Compacta – CFDL:

$$\Delta y(k + 1) = \phi_c(k)\Delta u(k) \quad (2.3)$$

com $\phi_c(k)$ limitado para todo k . A prova foi apresentada por Hou e Jin (2014).

2.3.2. Projeto de um Controlador MFAC-CFDL para Sistemas Não-Lineares SISO

O MFAC usa um modelo linear dinâmico da planta para o projeto do controlador. O Teorema 2.1 garante que um sistema SISO não-linear, atendidas às Suposições 2.1 e 2.2 e $|\Delta u(k)| \neq 0$, pode ser transformado no seguinte modelo dinâmico linearizado:

$$y(k+1) = y(k) + \phi_c(k)\Delta u(k), \quad (2.4)$$

onde $\phi_c(k) \in R$ é a PPD do sistema.

Adotando como objetivo de controle a minimização do seguinte índice de desempenho:

$$J(u(k)) = |y_d(k+1) - y(k+1)|^2 + \lambda |u(k) - u(k-1)|^2, \quad (2.5)$$

sendo $\lambda > 0$ um parâmetro de ponderação, usado para restringir a taxa de mudança no sinal de controle, e $y_d(k+1)$ o sinal de referência desejada para saída.

Uma lei de controle para a planta pode ser obtida, substituindo a Eq. (2.4) na Eq. (2.5), seguida pela diferenciação do índice de desempenho em relação a $u(k)$, e por fim, igualando o resultado a zero, obtendo:

$$u(k) = u(k-1) + \frac{\rho \phi_c(k)}{\lambda + |\phi_c(k)|^2} (y_d(k+1) - y(k)), \quad (2.6)$$

onde, o parâmetro $\rho \in (0,1]$ foi adicionado para fazer o algoritmo mais geral.

Na Eq. (2.6) apenas a PPD não é conhecida e infelizmente também não pode ser obtida analiticamente. No entanto, é possível usar um algoritmo de estimação de parâmetros variantes no tempo para estimá-lo. Por exemplo, considere a seguinte função de custo para estimação da PPD:

$$J(\phi_c(k)) = |y(k) - y(k-1) - \phi_c(k)\Delta u(k-1)|^2 + \mu |\phi_c(k) - \hat{\phi}_c(k-1)|^2 \quad (2.7)$$

onde $\mu > 0$ é um parâmetro de ponderação e $\hat{\phi}_c(k-1)$ o valor estimado da PPD para o instante $k-1$. A Eq. (2.7) foi elaborada visando reduzir a sensibilidade na estimação da PPD de erros devido a dados inexatos resultado de distúrbios ou falha nos sensores.

Minimizar a Eq. (2.7) em relação a $\phi_c(k)$ produz o seguinte algoritmo de estimação:

$$\hat{\phi}_c(k) = \hat{\phi}_c(k-1) + \frac{\eta \Delta u(k-1)}{\mu + (\Delta u(k-1))^2} (\Delta y(k) - \hat{\phi}_c(k-1) \Delta u(k-1)), \quad (2.8)$$

sendo que, o parâmetro $\eta \in (0,2]$ foi adicionado para fazer o algoritmo mais geral e mais flexível.

Para garantir que as condições exigidas pelo Teorema 2.1 e pelas Suposições 2.1 e 2.2 permaneçam válidas, durante todo o período de atuação do controlador, um mecanismo de reinicialização deve ser incorporado ao algoritmo de estimação da PPD, a saber:

$$\hat{\phi}_c(k) = \hat{\phi}_c(1) \quad \text{se} \quad |\hat{\phi}_c(k)| \leq \varepsilon \quad \text{ou} \quad |\Delta u(k-1)| \leq \varepsilon \quad \text{ou} \quad \text{sign}(\hat{\phi}_c(k)) \neq \text{sign}(\hat{\phi}_c(1)) \quad (2.9)$$

ou seja, se o valor da PPD estimada se tornar insignificante, ou se o incremento no sinal de controle calculado para o sistema for insignificante, ou se o sentido da PPD estimada mudar, situações que afetariam a validade da Suposição 2.2 e do Princípio da Excitação Persistente (ASTROM; WITTENMARK, 2008), o valor da PPD estimada deve ser reinicializado.

2.3.3. Sintonia dos Parâmetros

Conforme pode ser observado nas seções anteriores, o projeto de um controlador MFAC exige a escolha de um certo número de parâmetros que geralmente são selecionados com base numa análise qualitativa da resposta do sistema. Recentemente, algumas estratégias para resolver este problema têm sido

propostas, por exemplo, Wang *et al.* (2014) usaram o conceito de otimização do índice de entropia mínima para sintonizar os parâmetros de um controlador MFAC-PFDL e Roman *et al.* (2016) desenvolveram a técnica VRFT-MFAC, que consiste no uso do algoritmo da técnica DDC VRFT para calcular todos os parâmetros do controlador MFAC aplicado a sistemas MIMO.

A escolha dos parâmetros ideais para o controlador MFAC é um típico problema de otimização, no entanto, técnicas de otimização baseadas no cálculo do gradiente não são indicadas, pois a equação da planta pode ser desconhecida ou muito complexa. Sousa *et al.* (2014) apresentaram o ajuste dos parâmetros do controlador MFAC-CFDL *off-line* usando um algoritmo evolucionário multiobjetivo baseado em evolução diferencial. Os resultados sugerem a validade da estratégia adotada e também encorajam novas pesquisas nessa área, visando agora, a sintonia *on-line* dos parâmetros.

2.4 Algoritmos Evolucionários

A observação da natureza é fonte de inspiração para solução de diversos problemas práticos do mundo real. Algoritmos evolucionários são processos iterativos de busca (otimização) de soluções baseados na teoria da evolução das espécies de Charles Darwin com aplicações em diversas áreas, tais como, planejamento (DHURI; SESHU, 2009), projeto de sistemas (TAYARANI, 2015), simulação e identificação de processos (KRISTINSSON; DUMONT, 1992; SAKAGUCHI; YAMAMOTO, 2003) e controle (SAAD; DARUS, 2012). Um algoritmo evolucionário padrão apresenta as seguintes características: uma população de indivíduos, conjunto de soluções, associados à uma codificação genética; pelo menos um operador de recombinação que permita trocas entre os genes dos indivíduos, os quais representam as variáveis de decisão do problema, para facilitar a propagação a cada geração das qualidades desejadas; pelo menos um operador de mutação, para periodicamente inserir novas informações ao banco genético da população; pelo menos um mecanismo de seleção para certificar que os mais

adaptados sobrevivam com a evolução da população e uma condição de término indicando a melhor solução da população.

A aplicação do conceito de seleção natural na solução de problemas específicos ocasionou a geração dos chamados algoritmos evolucionários canônicos, a saber:

- **Algoritmos Genéticos - AG:** descritos por Holland (1975) os AGs usam a recombinação como principal operador de busca (operador primário) e a mutação tem probabilidade baixa atuando com papel secundário. Na sua proposição inicial, denominada Algoritmo Genético Simples – AGS, usa a codificação binária para representar os indivíduos, a recombinação mediante cruzamento em um ponto e a mutação do tipo *bit flip* (mudança no estado de um bit do cromossomo escolhido de forma aleatória). Atualmente, os AGs admitem diversas formas de representação para os indivíduos, tais como, por números inteiros, números reais e outras representações usadas em problemas de permutações. Além disso, na literatura também são descritas inúmeras variantes para os operadores de cruzamento e mutação (EIBEN; SMITH, 2003). Os AGs adotam um critério de seleção probabilística, proporcional à aptidão dos indivíduos da população, para a escolha dos pais usados na operação de cruzamento e dos indivíduos formadores da população da próxima geração (HOLLAND, 1992).
- **Estratégias Evolucionárias - EE:** desenvolvidas por Rechenberg e Schwefel, as EE usam indivíduos com representação real, operações de recombinação do tipo discreta ou contínua, e mutações baseadas na função de distribuição normal. As EE permitem o ajuste dos parâmetros definidos na operação de mutação durante a busca, para tanto usa esquemas de adaptação, tais como, a regra do um quinto de sucesso, e auto-adaptação, pela inclusão dos parâmetros de mutação na codificação dos cromossomos usados para representar os indivíduos. Nas EE os tamanhos das populações de pais e filhos podem ser distintos e a escolha para a operação de cruzamento é

uniforme considerando toda população de pais. A seleção para formação da população da próxima geração é realizada entre os filhos de determinística, só os mais aptos são escolhidos (EIBEN; SMITH, 2003).

- **Programação Evolucionária - PE:** proposto por Fogel (FOGEL, 2006), originalmente foi concebida para evolução de máquinas de estado finito. No entanto, atualmente é empregada em problemas de otimização no domínio dos números reais. Para tanto, a PE usa indivíduos com representação real e não usa operação de recombinação, portanto, a evolução da busca é baseada principalmente nas mutações. Modernamente, a PE usa um esquema de mutação semelhante ao empregado nas EE, ou seja, baseado na função de distribuição normal e em uma estratégia auto-adaptativa para ajuste dos parâmetros da função de distribuição. A escolha dos indivíduos da população para próxima geração é normalmente feita usando um operador de seleção probabilístico, aplicado à população total formada pelos indivíduos pais e filhos (EIBEN; SMITH, 2003).
- **Programação Genética - PG:** técnica de programação usando algoritmos evolucionários. Na PG as estruturas de dados são representadas utilizando árvores, e os operadores de cruzamento e mutação, são adaptados para operarem com esse tipo e estrutura. O processo de seleção segue o proposto pelos AGs, ou seja, a seleção é probabilística e proporcional a aptidão (KOZA, 1992).

Atualmente, na literatura, uma série de aprimoramentos a estes algoritmos canônicos tem sido discutida, por exemplo, separação da população em ilhas para aumentar o paralelismo (EIBEN; SMITH, 2003), inserção de imigrantes para aumentar a diversidade da população (YANG; TINOS, 2007), e até sua combinação com métodos clássicos de otimização, denominados algoritmos evolucionários híbridos (PIOTROWSKI, 2013). Além disso, essa ferramenta tem sido usada também na solução de problemas cada vez mais complexos, como problemas multiobjetivo (COELLO *et al*, 2007) e problemas dinâmicos (YU *et al.*, 2008).

Portanto, a área de computação evolucionária é um campo bastante fértil para novas pesquisas.

2.4.1. Algoritmos Evolucionários Multi-Objetivo

Problemas do mundo real geralmente envolvem a análise simultânea de múltiplos critérios ou objetivos. Entretanto, esses objetivos estão muitas vezes em conflito entre si (*trade-offs*) e conseqüentemente, a melhora em um objetivo pode causar deterioração em outro. A solução para esse tipo de problema passa a ser, no lugar de uma solução única, encontrar um conjunto de soluções que represente o melhor compromisso entre os diversos critérios propostos. Estas soluções são denominadas Pareto ótimas, pois nenhuma outra solução pode ser encontrada, melhorando um objetivo particular, sem prejuízo de um ou mais dos outros objetivos (COELLO et al, 2007).

Um Problema de Otimização Multiobjetivo - POM pode ser definido como uma busca por soluções $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]^T$ dentro de um espaço de decisão D , as quais estejam dentro de uma região factível FR definida por m restrições de desigualdade:

$$g_i(\mathbf{x}) \geq 0; \ i = 1, \dots, m \quad (2.10)$$

e p restrições de igualdade:

$$h_i(\mathbf{x}) = 0; \ i = 1, \dots, p \quad (2.11)$$

enquanto otimiza um vetor de funções objetivo, portanto, se o objetivo é minimizar todas as funções objetivo então:

$$\text{Min}([f_1(\mathbf{x}) \ f_2(\mathbf{x}) \ \dots \ f_k(\mathbf{x})]^T) \quad (2.12)$$

A comparação entre duas soluções de um POM é realizada com base na denominada fronteira de Pareto, a qual é construída usando o critério de dominância (DEB, 2001).

Uma solução x_1 é dita dominar outra solução x_2 (matematicamente se escreve $x_1 \leq x_2$) se as seguintes condições forem satisfeitas (no caso de minimização de todas as funções objetivo):

$$\forall i \in (1, 2, \dots, k) \mid f_i(x_1) \leq f_i(x_2) \quad (2.13)$$

$$\exists i \in (1, 2, \dots, k) \mid f_i(x_1) < f_i(x_2) \quad (2.14)$$

isto é, a solução x_1 é melhor ou igual a x_2 em todos os elementos do vetor de função objetivo, mas é estritamente melhor em pelo menos um dos objetivos.

No passado, os problemas multiobjetivo costumavam ser tratados como, problemas mono-objetivos pela construção de uma função de objetivo geral que pondera a importância relativa de cada objetivo (DEB, 2001). Entretanto, essa abordagem apresenta algumas dificuldades, por exemplo, como atribuir os valores dos pesos na função de ponderação e esse método é também incapaz de identificar as partes não-convexas da superfície de compromisso no espaço de objetivos, denominada fronteira de Pareto.

AEs têm sido usados com sucesso para aproximar o conjunto Pareto por causa da sua flexibilidade, evoluindo uma população inteira em direção à fronteira de Pareto. Uma revisão abrangente da teoria sobre dos Algoritmos Evolucionários Multiobjetivo - AEMOs foi realizada por Coello *et al.* (2007). O Algoritmo 2.1 demonstra um exemplo de como um AEMO básico pode ser implementado:

ALGORITMO 2.1 – AEMO BÁSICO

- 1: Gere uma população inicial P_0 com NP indivíduos.
- 2: Avalie P_0 .
- 3: Construa um conjunto aproximado de Pareto inicial $X_{P_0}^*$ (apenas indivíduos não-dominados).
- 4: Inicialize o contador de gerações $G = 0$.
- 5: ENQUANTO critério de finalização não for alcançado
- 6: Atualize o contador de gerações $G = G + 1$.
- 7: Gere uma nova população P_G^* aplicando operadores

- evolucionários sobre a população atual P_G .
- 8: Avalie a população P_G^* .
 - 9: Construa uma nova aproximação do conjunto de Pareto $X_{P_G}^*$ usando $X_{P_{G-1}}^*$ e os indivíduos não dominados de P_G^* .
 - 10: Atualize a população P_{G+1} mediante P_G e P_G^* .
 - 11: FIM DO ENQUANTO
 - 12: Retorne a aproximação da fronteira de Pareto $X_{P_G}^*$.

No algoritmo, destaca-se as linhas 3 e 9 nas quais uma aproximação da fronteira de Pareto é realizada, mediante uma busca no conjunto de indivíduos das soluções não-dominadas. Na linha 7 os operadores evolucionários, tais como, recombinação e mutação são usados para gerar novas soluções a partir da população atual. No caso de AEMOs, costuma-se usar como critérios de parada, por exemplo, a máxima quantidade de gerações admitida, ou uma medida da distância entre a fronteira estimada e a fronteira de Pareto real, quando esta é conhecida, ou medidas relacionadas com o espalhamento uniforme das soluções ao longo da fronteira, tais como, o hipervolume calculado a partir da fronteira de Pareto estimada (GARROZI, 2012).

2.4.2. Evolução Diferencial

O algoritmo de Evolução Diferencial (Differential Evolution – DE) proposto por Storn and Price (1995) é um algoritmo de otimização estocástico para parâmetros reais, baseado na evolução de populações de soluções, mediante operações com os indivíduos dessas populações, similares às empregadas pela maioria dos algoritmos evolucionários (DAS, SUGANTHAN, 2011). A DE destaca-se pela simplicidade e eficiência, características desejáveis ao projeto de um sistema de controle ótimo. As etapas de execução da DE mono-objetivo podem ser melhor compreendidas a partir do diagrama apresentado na Figura 3

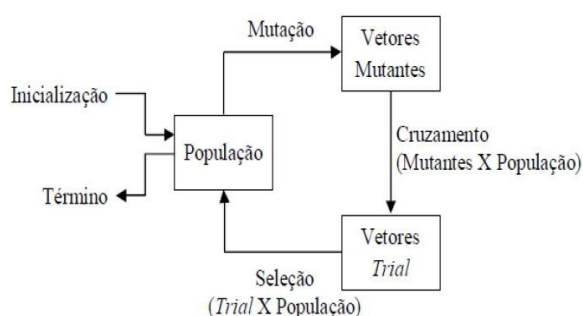


Figura 3 - Etapas do algoritmo de Evolução Diferencial, adaptado de (EIBEN; SMITH, 2003).

No diagrama, inicialmente para cada indivíduo da população, chamado de vetor alvo, são escolhidos de forma aleatória outros três indivíduos distintos, que deverão ser combinados, mediante operações aritméticas simples entre vetores, gerando um vetor mutante ou doador. No passo seguinte, o vetor mutante e seu respectivo vetor alvo são cruzados gerando um indivíduo novo denominado vetor tentativa (*Trial*). Em seguida, as aptidões dos vetores alvo e tentativa são comparadas e aquele que apresentar a melhor aptidão passará para a próxima geração na população de soluções. Todo o processo é repetido para todos os indivíduos da população, definindo-se, com isso, uma nova geração. Finalmente, essa rotina é executada até que alguma condição de parada seja alcançada, por exemplo, uma quantidade máxima de gerações seja atingida; ou um valor específico para função de otimização seja obtido; ou se nenhuma mudança significativa no resultado da função objetivo seja encontrada pelas soluções obtidas com o passar das gerações.

O Algoritmo 2.2 foi construído com base no diagrama da Figura 3 e demonstra o funcionamento do DE no seu formato original (PRICE *et al.*, 2005):

ALGORITMO 2.2 – EVOLUÇÃO DIFERENCIAL CANÔNICA

- 1: Defina os parâmetros do algoritmo: F = fator de mutação, Cr = fator de cruzamento;
- 2: Gere uma população inicial P_0 com NP indivíduos.
- 3: Avalie P_0 usando a função objetivo $f(X_{i,0})$.
- 4: Inicialize o contador de gerações $G = 0$.
- 5: ENQUANTO critério de finalização não for alcançado.

- 6: PARA $i = 1$ ATÉ NP (Para cada indivíduo da população).
- 7: Sorteie outros três indivíduos distintos $(X_{r0,G}, X_{r1,G}, X_{r2,G})$.
- 8: Gere um vetor mutante como uma combinação
 $V_{i,G} = X_{r0,G} + F \times (X_{r1,G} - X_{r2,G})$.
- 9: Gere um vetor tentativa cruzando o indivíduo atual com
o vetor mutante $U_{i,G} = \text{cross}(Cr, X_{i,G}, V_{i,G})$.
- 10: SE $f(U_{i,G}) \leq f(X_{i,G})$ ENTÃO.
- 11: $X_{i,G+1} = U_{i,G}$.
- 12: FIM DO SE.
- 13: FIM DO PARA.
- 14: Atualize o contador de gerações $G = G + 1$.
- 15: Avalie o critério de Parada
- 16: FIM DO ENQUANTO.
- 17: Retorne o ponto ótimo $X_{P_G}^*$.

Tendo por base o algoritmo original, na literatura podem ser encontrados vários esquemas para DE. Portanto, costuma-se usar a operação de mutação para diferenciar um esquema do outro. A convenção geral usada é nomear o esquema como DE / x / y / z, em que DE significa "Evolução Diferencial", x representa como o vetor base é escolhido, y o número de vetores de diferença usado para calcular o vetor mutante e z, significa o tipo de cruzamento que está sendo usado. Por exemplo, o algoritmo DE canônico é referido como DE/rand/1/bin.

2.4.3. Non-Dominated Sorting Differential Evolution - NSDE

O algoritmo DE também pode ser aplicado na otimização de múltiplos objetivos. Suganthan e Das (2011) apresentaram uma revisão sobre o estado da arte da DE e elencaram as diversas variações do algoritmo disponíveis para problemas multi-objetivo. Dentre os algoritmos listados, o NSDE, proposto por Inorio e Li (2004), consiste na adaptação do mecanismo de seleção definido para o algoritmo NSGA II (DEB *et al.*, 2001) para ser usado com a DE invariante a rotação

em problemas multi-objetivo. No algoritmo, inicialmente, uma população aleatória de NP indivíduos é gerada dentro da região factível. Em seguida, esta população é usada para gerar NP filhos, usando para isso as operações do algoritmo DE/*current to rand/1* invariante a rotação ($Cr = 1$). Depois, dentre os membros da população de pais e filhos são selecionados apenas NP indivíduos, usando como critério de seleção o *ranking* de soluções não-dominadas e o valor da distância de multidão (*crowding distance* - CD), calculada para os indivíduos com o mesmo *ranking*, conforme mecanismo de seleção definido no algoritmo NSGA II. Os indivíduos resultantes comporão a população da próxima geração. Todo o processo é repetido com a população resultante até algum critério de parada ser alcançado. Na sua apresentação, o NSDE destacou-se por obter resultados superiores ao NSGA II num conjunto de problemas com rotação e forte interdependência entre as variáveis (MEZURA-MONTES, *et al.*, 2008).

O Algoritmo 2.3 apresenta o funcionamento do algoritmo NSDE

ALGORITMO 2.3 – NSDE

- 1: Defina os parâmetros do algoritmo: F = fator de mutação, K = nível de combinação entre $X_{i3,G}$ e o vetor alvo;
- 2: Gere uma população inicial P_P^0 com NP indivíduos (PAIS).
- 3: Avalie P_P^0 usando o vetor de funções objetivo $f(\cdot)$.
- 4: Inicialize o contador de gerações $G = 0$.
- 5: ENQUANTO critério de Parada não for alcançado.
- 6: PARA $i = 1$ ATÉ NP (Para cada indivíduo da população).
- 7: Sorteie três indivíduos distintos ($X_{r1,G}, X_{r2,G}, X_{r3,G}$).
- 8: Gere um vetor mutante/filho com a combinação

$$V_{i,G} = X_{i,G} + K \times (X_{r3,G} - X_{i,G}) + F \times (X_{r1,G} - X_{r2,G})$$
- 9: Avalie o vetor mutante usando o vetor de funções
objetoivo $f(V_{i,G})$.
- 10: FIM DO PARA.
- 11: Junte a população de Pais e Filhos ($P_P^0 \cup P_F$)
- 12: Selecione NP indivíduos para próxima geração usando o
ranking de soluções não dominadas e o valor da CD

- 13: Atualize o contador de gerações $G = G + 1$.
- 14: Avalie o critério de Parada
- 15: FIM DO ENQUANTO.
- 16: A População Final como uma estimativa do conjunto de Pareto.

As principais alterações em relação a DE canônica mono-objetivo são o esquema de mutação adotado, *current to rand* (linha 8) e a ausência de cruzamento. No algoritmo DE, quando $CR = 1$ o vetor mutante é usado diretamente como tentativa, o que torna a DE invariante a rotação de eixos (PRICE et al., 2005). Na linha 12, a seleção entre pais e filhos é realizada conforme algoritmo NSGA II.

2.5 Modelo para Dinâmica da Distribuição da População no Algoritmo DE

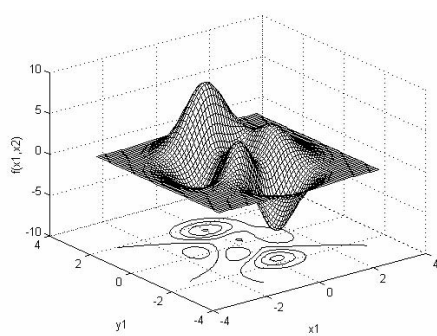
Nesta seção, será analisado o comportamento esperado para a dinâmica da distribuição dos indivíduos da população, durante a evolução até a solução ótima ser encontrada. A exploração desse comportamento será usada como inspiração para a estratégia de geração de imigrantes proposta nesse trabalho.

2.5.1. Distribuição de Indivíduos no Espaço de Soluções

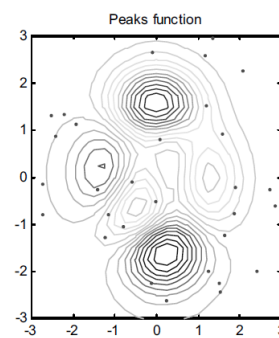
A Figura 4a apresentada em (PRICE et al., 2005) mostra o gráfico tridimensional da função multimodal Peaks expressa pela Eq. (3.3).

$$f(x_1, x_2) = 3(1 - x_1)^2 \exp(x_1^2 + (x_2 + 1)^2) - 10 \left(\frac{x_1}{5} - x_1^3 - x_2^5 \right) \cdot \exp(x_1^2 + x_2^2) - \frac{1}{3} \exp((x_1 + 1) + x_2^2) \quad (3.3)$$

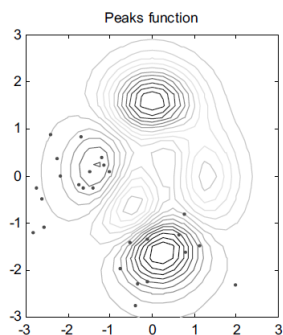
Já as Figuras 4b até a 4h mostram a evolução da distribuição geográfica dos indivíduos da população, pontos escuros, durante a otimização da Eq. (3.3),



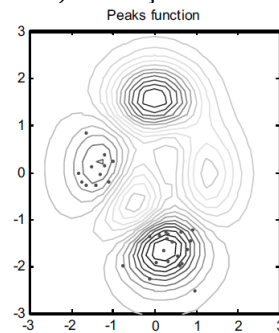
a) Gráfico 3D da função Peaks



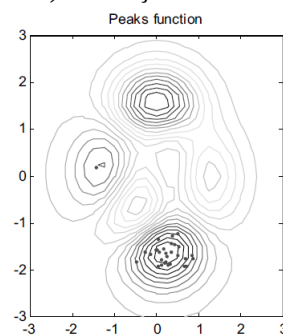
b) Geração 1



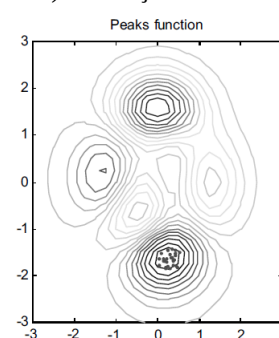
c) Geração 6



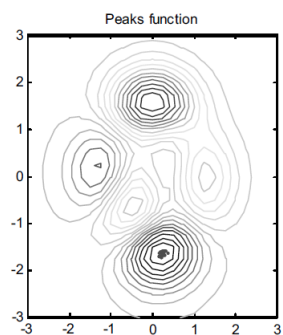
d) Geração 12



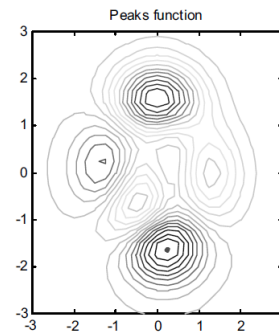
e) Geração 16



f) Geração 20



g) Geração 26



h) Geração 34

Figura 4 – Distribuição de indivíduos durante DE, adaptado de (PRICE *et al.*, 2005).

Analisando todas figuras, observa-se que, à medida que as gerações avançam, os indivíduos tendem a se aglomerar em torno de regiões promissoras. Portanto, a identificação prévia da posição dos centros e da extensão dessas regiões, por exemplo, usando alguma técnica de agrupamento, seguido pela inserção de novos indivíduos distribuídos em torno dessas regiões, poderia aprimorar a fase de exploração do algoritmo e acelerar o processo de otimização.

2.5.2. Modelo para Dinâmica dos Agrupamentos da População

Identificar os agrupamentos presentes na população pode trazer vantagens na otimização, no entanto, entender a dinâmica da região de influência definida por esses grupos durante a evolução também pode ser importante. Ao longo das gerações, a posição dos centros dos grupos tende a se modificar, além disso, o espalhamento dos membros do grupo, ou seja, a extensão de sua área de influência também deve variar. Inclusive, pelo processo de seleção, agrupamentos formados por indivíduos menos aptos tendem a desaparecer com o tempo.

Uma distribuição normal multivariada apresenta uma função de densidade de probabilidade monomodal, em forma de sino, cujas coordenadas do topo do sino, dentro da região do domínio da função, localizam a média da distribuição. Qualquer distribuição normal, representada por $N(\mathbf{m}, \mathbf{C})$, é completamente identificada por sua média $\mathbf{m} \in R^n$ e uma matriz de covariância $\mathbf{C} \in R^{n \times n}$. As matrizes de covariância são simétricas, positiva definidas e têm uma interpretação geométrica associada, ou seja, elas podem ser unicamente identificadas pelo hiper-elipsoide definido por $\{\mathbf{x} \in R^n | \mathbf{x}^T \mathbf{C}^{-1} \mathbf{x} = 1\}$, sendo, os eixos principais do elipsoide e os comprimentos dos eixos quadrados, os autovetores e autovalores de $\mathbf{C}$, respectivamente.

Assumindo que as regiões dos agrupamentos podem ser aproximadas por um hiper-elipsoide, é possível com isso estimar a região de influência de cada grupo encontrado, calculando centro do grupo, e a matriz de covariância dos seus

componentes. O histórico do percurso realizado pelos centros e a matriz de covariância dos membros grupo, obtidos durante a otimização, são informações valiosas para atualizar as características das suas regiões de influência e podem ser usados para aprimorar o processo de busca.

As Figuras 5a-5h exemplificam como a matriz de covariância pode ser adaptada, por conseguinte a região de influência definida pelo grupo, usando a dinâmica do seu centro, para seguir direções mais promissoras à busca, de forma semelhante ao conceito de caminho evolucionário proposto pelo algoritmo de otimização Estratégia Evolucionária com Adaptação da Matriz de Covariância (*Covariance Matrix Adaptation Evolutionary Strategy*- CMA-ES) (HANSEN *et al.*, 2003). Na Figura 5a a população inicial de um grupo de indivíduos está limitada numa região do espaço de soluções, caracterizada por um vetor médio (centro) $\mathbf{m} \in \mathbb{R}^n$ e uma matriz de covariância $\mathbf{C}^G \in \mathbb{R}^{n \times n}$. A ação dos operadores evolucionários e o processo de seleção proporciona o surgimento de novos indivíduos, que modificam a região ocupada pelo grupo e alteram a posição do centro da subpopulação (Figura 5b). A Figura 5c mostra como o vetor $\mathbf{y}_w$, resultado da diferença entre as posições dos centros das regiões final e inicial, pode ser usado para adaptar a matriz de covariância do grupo, mediante $\mathbf{C}^{(G+1)} = (1-\alpha)\mathbf{C}^{(G)} + \alpha \mathbf{y}_w \mathbf{y}_w^T$, sendo $\alpha \in (0,1)$. O novo centro e a matriz de covariância adaptada delimitam uma região, teoricamente, mais promissora para geração de indivíduos do espaço de busca (Figuras 5d – 5e). As Figuras 5f-5h demonstram como encontrar a nova região promissora da próxima iteração e a trajetória dessas regiões no espaço de soluções.

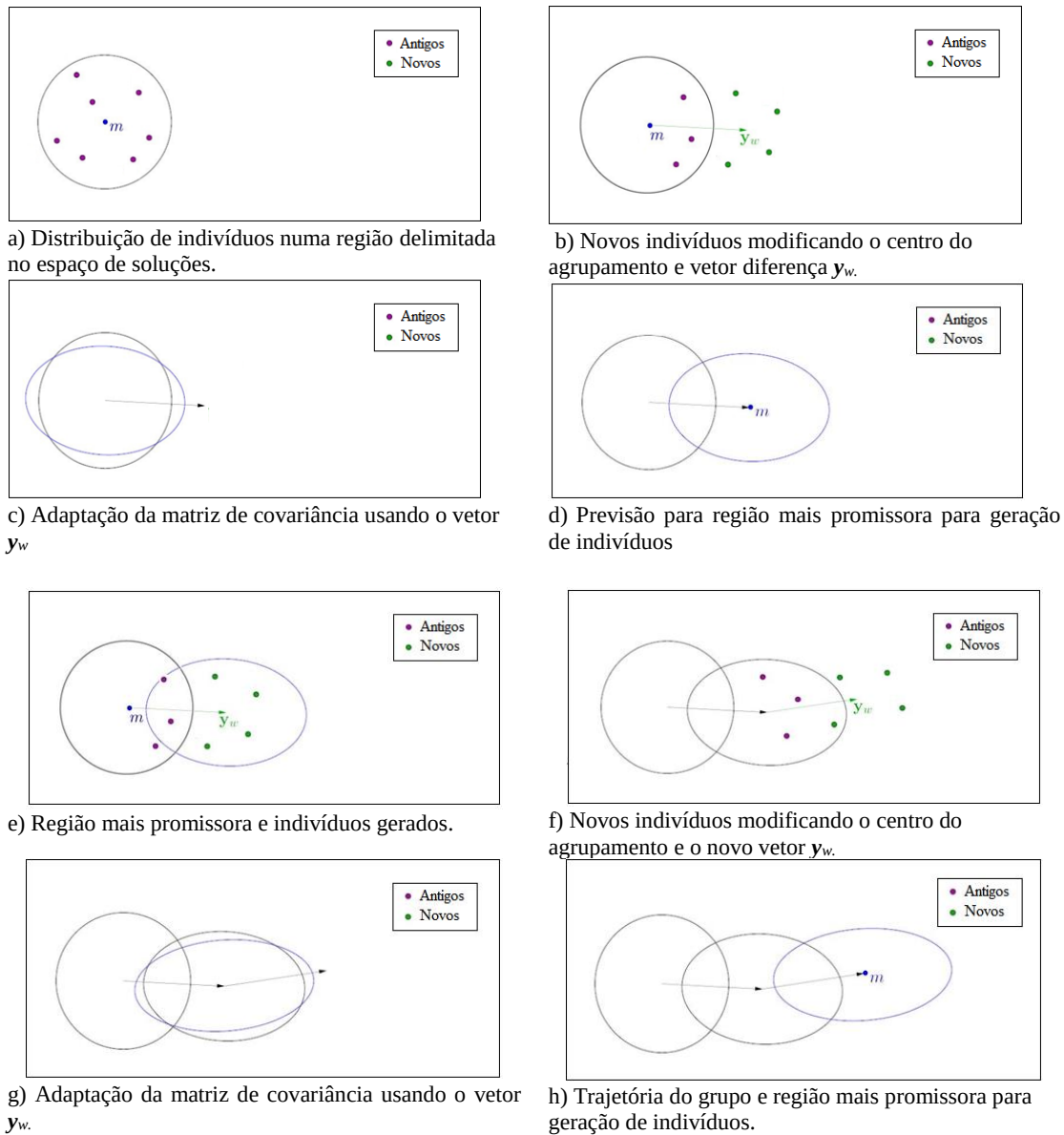


Figura 5 – Dinâmica da distribuição dos indivíduos no espaço de soluções.

2.6 Evolução Diferencial em Sistemas de Controle DDC

Um projeto de um sistema de controle ótimo costuma ser um desafio, pois, normalmente exige que múltiplos critérios de decisão sejam considerados. Essa multiplicidade somada às dificuldades usuais para o tratamento de sistemas dinâmicos, não-lineares e estocásticos, tende a inviabilizar o uso de ferramentas

convencionais de otimização e favorecer o uso meta-heurísticas, na tomada de decisões do projeto de controle (ESPINOSA; AYALA-SOLARES, 2016). Ressalte-se, no entanto, que para problemas bem definidos, para os quais já existem soluções confiáveis, é improvável que um AE produza resultados competitivos com os métodos clássicos de otimização (FLEMING; PURSHOUSE, 2002). Apesar disso, o potencial do AE é imenso, pois, são possíveis múltiplas representações para um mesmo problema, diversas formas de trabalhar com as restrições impostas e é possível introduzir funções de custo para otimizar o projeto de controle. Tudo isso, fornece ao projetista um maior grau de escolha e também de flexibilidade.

Em relação ao uso de AEs em projetos de controle, recentemente, Reynoso-Meza *et al.* (2014) identificaram as tendências atuais e o estado da arte da sintonia de parâmetros para controladores usando AEs multi-objetivo. No artigo, os autores elencam as dificuldades no uso de métodos tradicionais de otimização em problemas de controle e argumentam sobre as vantagens relacionadas com a flexibilidade do uso de AEs. Além disso, também apresentam uma ampla revisão sobre otimização multi-objetivo com aplicações em controle, com ênfase na sintonia de parâmetros para diversos tipos de controladores. Entretanto, os autores concluem que, apesar do constante desenvolvimento dos AEs multi-objetivo, seu uso como ferramenta de otimização para os parâmetros de um controlador, ainda não é tão comum. Os autores apresentam algumas possíveis barreiras para isso, dentre as quais, a inexistência prévia de um problema de otimização multi-objetivo diretamente relacionado ao projeto de controle específico e a disponibilidade ou mesmo a existência de um AEMO que atenda os requerimentos exigidos pelo projetista de controle e produza uma aproximação para a fronteira de Pareto que apresente informações úteis para a tomada de decisão. No trabalho são sugeridas algumas ações para reverter essas barreiras, por exemplo, a escolha da ferramenta de otimização de acordo com a definição do problema de otimização, a fim de obter um melhor desempenho; a definição de critérios de otimização diretamente relacionados ao problema de controle, possivelmente envolvendo medidas no domínio do tempo e frequência, e o uso de metodologias de visualização para o tomador de decisões envolvendo múltiplos critérios visando a escolha da melhor solução para o projeto.

O algoritmo DE tem uma série de características interessantes como: ser de fácil implementação, dispensar codificação adicional para problemas com números reais, realizar busca estocástica mediante mecanismos de seleção natural, apresentar baixa probabilidade de ficar preso em um ótimo local, ter boa performance para resolver problemas de otimização com funções objetivo simples, e não requerer informação sobre derivadas (PRICE, 2005). Por essas propriedades, a DE costuma ser usada na literatura em aplicações para sintonia de parâmetros de controle. Por exemplo, Coelho *et al.* (2010) usou o algoritmo DE/*rand*/1/bin para sintonia dos parâmetros de uma estratégia de controle híbrida MFLAC-NN, resultado da combinação da técnica de controle DDC MFLAC (*Model Free Learning Adaptive Control*), uma variação do MFAC, e um compensador neural, baseado numa rede neural de função de base radial. O algoritmo DE foi usado para ajuste *off-line* dos parâmetros $\phi(1)$, ρ , η , λ e μ , definidos no MFLAC, e dos centros e as variâncias, usadas nas funções de base radial, portanto, trata-se da otimização dos parâmetros de um controlador com estrutura fixa. O compensador neural calcula o incremento ao sinal de controle usando como entrada três vetores normalizados: o erro, $e(k-1)$, a variação do sinal de controle, $\Delta u(k)$, e o sinal de referência $y_r(k-1)$. No artigo, foram analisados dois estudos de casos, um reator de tanque agitado (*stirred-tank reactor*) e uma válvula de controle não-linear. Os resultados numéricos demonstraram que, tanto o MFLAC, quanto o MFLAC-NN, apresentaram um bom desempenho em problemas de regulação. No entanto, o esquema MFLAC-NN foi mais efetivo no controle da válvula de controle não-linear.

O ajuste *on-line* dos parâmetros do controlador, apesar de ser uma característica desejada em projetos de controle industriais, não costuma ser tratado na literatura usando AEs (FLEMING; PURSHOUSE, 2002). No entanto, Reynoso-Meza *et al.* (2012) consideraram a simplicidade e o grau de compactação do algoritmo DE, para usá-lo como ferramenta de otimização na sintonia *on-line* dos parâmetros de um controlador DDC PID. No artigo, os autores sugerem algumas características ideais para uma estratégia de auto-sintonia envolvendo AEs, tais como: o algoritmo deve ser executado a cada instante de amostragem; o algoritmo deve incluir uma etapa de experimentação para adquirir informações sobre o processo; e a cada instante o algoritmo deve capturar dois tipos de informações: o

intervalo de tempo (contado desde o início do experimento) e as medidas que caracterizam o processo. O algoritmo desenvolvido e nomeado como EvoTune tem como objeto a sintonia dos parâmetros de um controlador PID para um sistema identificado como de primeira ordem com atraso de tempo (FOPDT - *First Order Plus Time Delay*), mediante a solução de um problema de otimização não-convexa com restrição. Como objetivo de controle, buscou-se maximizar o ganho integral, considerando três restrições ao problema, as quais estão relacionadas com a máxima margem de estabilidade do sistema, a máxima ação do controlador e os efeitos do ruído introduzido pela otimização no cálculo do sinal de controle. Os resultados obtidos mediante simulações, comprovaram que a estratégia de controle EvoTune é robusta e é capaz de se ajustar diferentes a sistemas com características distintas, tais como, múltiplos pólos, atraso longo e fase não-mínima.

Em relação a sintonia *off-line* de parâmetros de um controlador usando DE, Sahu *et al.* (2013) usaram o algoritmo DE/*rand*/1/bin para ajuste de um controlador PID para controle de um sistema de potência. No trabalho, os autores transformam o projeto de controle, com múltiplos critérios a serem atendidos, num problema de otimização mono-objetivo, combinando os índices de desempenho propostos e uma única função. Já Mohanty *et al.* (2014) testaram vários esquemas de mutação para DE na sintonia *off-line* de parâmetros de controladores do tipo: Integral - I, Proporcional Derivativo - PD e Proporcional Integral Derivativo - PID aplicados a sistemas de potência. O artigo, também apresenta os resultados variando os parâmetros do esquema mutação vencedor, demonstrando a necessidade de ajuste também dos parâmetros internos da DE, por exemplo, o tamanho da população, os fatores de mutação e de cruzamento, para se obter melhores resultados na otimização. Testes realizados com outros controladores e diversos critérios de controle atestaram o método de sintonia.

A capacidade natural dos AEs de serem adaptados para solução de problemas de otimização multi-objetivo, fez surgir diversos algoritmos e estratégias com esse fim (COELLO *et al.*, 2007). Conforme já mencionado, o algoritmo multi-objetivo NSDE combina as operações típicas da DE com a seleção usada no algoritmo NSGA II. Portanto, alia as vantagens como simplicidade e performance, associadas à DE, dispensa do uso de um arquivo externo, e emprega um

mecanismo de seleção menos custoso do que o usado em outros AEMOs, por exemplo, o do SPEA 2. O NSDE tem despertado interesse em diversas áreas de pesquisa, tais como, engenharia (GONG *et al.*, 2009; LI *et al.*, 2013; FLOREA *et al.*, 2016), planejamento (ROSELYN *et al.*, 2014; PENG *et al.*, 2010.), economia (PENG *et al.*, 2012) e controle (SARAVANAN *et al.*, 2008; PANDA, 2011). Essas características vêm ao encontro das necessidades impostas a um projeto de um sistema de controle e merecem ser exploradas nesta Tese.

O tratamento apresentado na literatura para problemas de controle similares envolvendo AEs, somado às tendências e às sugestões elencadas pelos pesquisadores da área, contribuirão, neste trabalho, com o desenvolvimento de estratégias de otimização para serem aplicadas no projeto de controle de sistemas complexos.

2.7 Discussões Finais

Neste capítulo, inicialmente, foram abordadas as principais técnicas recentes de Controle Direcionado a Dados. Em particular, pelas características demonstradas, foi destacado o uso da técnica MFAC para sistemas não-lineares variantes no tempo. No entanto, conforme foi exposto, o ajuste dos parâmetros do controlador dessa técnica ainda é uma questão em aberto e merece mais estudos.

Em seguida, um resumo sobre otimização via AEs foi apresentado com destaque ao algoritmo DE e suas versões para solução de problemas de otimização mono e multi-objetivo. Depois, um modelo para dinâmica da população na DE, durante o processo de evolução, foi apresentado. As características observadas nessa dinâmica serão importantes da definição do modelo de AE a ser proposto neste trabalho.

Finalmente, foram elencados alguns exemplos pesquisados na literatura, que sugerem a viabilidade do uso da DE e NSDE, no projeto de controle de sistemas não-lineares variantes no tempo, portanto, justificando seu uso nesta Tese.

3

MODELO DE EVOLUÇÃO DIFERENCIAL E CONTROLE ADAPTATIVO EVOLUCIONÁRIO

Neste capítulo, inicialmente será apresentado o modelo conceitual para o desenvolvimento do AE proposto. Em seguida, serão elaboradas versões do algoritmo para problemas mono e multi-objetivo. Depois, será abordado como o AE multi-objetivo pode ser empregado no projeto de um sistema de controle. Ao final, será apresentado como pode ser feita a integração entre o algoritmo proposto e o sistema de controle para sintonia *off-line* e *on-line* dos parâmetros do controlador.

3.1 Concepção do Modelo

Desde sua proposição (GREFENSTETTE, 1992), a inserção de imigrantes aleatórios à população atual, mostrou-se inicialmente uma alternativa viável para manutenção da diversidade em EAs, principalmente, no caso de problemas de otimização dinâmicos. Entretanto, imigrantes também podem ser usados para fazer o AE explorar diferentes regiões promissoras em problemas multimodais (HORNBY, 2009). A Figura 6 apresenta o diagrama usado para concepção do modelo proposto

objetivo, a comparação é realizada sistematicamente entre cada membro da população e seu vetor tentativa correspondente de forma elitista, ou seja, dentre os dois, o vetor que obtiver a melhor avaliação ocupará a posição disputada na população da próxima geração. Em problemas multi-objetivo, no trabalho, optou-se por usar o mecanismo de seleção do algoritmo NSGA II que fará a escolha dentre os membros do conjunto $\{P^G \cup P^T\}$ para compor a $\{P^{G+1}\}$. O processo é então repetido para as próximas gerações até que alguma condição de parada seja satisfeita. Como resultado são obtidas estimativas para o ponto ótimo ou para o conjunto de Pareto, caso o problema seja mono ou multi-objetivo, respectivamente.

A segunda parte da Figura 6 mostra a estratégia proposta para criação dos imigrantes direcionados. No esquema, são identificados um número N_c de agrupamentos dentro da população de indivíduos $\{P^{G+1}\}$. No algoritmo, o número de agrupamentos é inicialmente definido pela razão entre o tamanho da população e a dimensão dos indivíduos, no entanto, esse número é atualizado a cada geração, considerando que alguns grupos podem se tornar menos promissores a busca, não fazendo sentido mantê-los. Para cada grupo é definido seu centro $\mathbf{m}_i^{G+1}$, em geral o ponto médio ou o indivíduo melhor avaliado, e calculada a matriz de covariância dos indivíduos do grupo $\mathbf{C}_i^{G+1}$. No diagrama, essa informação é armazenada num banco de memória $\{(\mathbf{m}_i^{G+1}, \mathbf{C}_i^{G+1}), (\mathbf{m}_i^G, \mathbf{C}_i^G)\}, i = 1, \dots, N_c\}$ para ser usada na adaptação da matriz de covariância dos grupos, de forma semelhante ao apresentado no Capítulo 2. A região geométrica no espaço de soluções delimitada pelos hiper-elipsoides, definidos pelos centros e matrizes de covariância adaptados $\{(\mathbf{m}_i^l, \mathbf{C}_i^l), i = 1, \dots, N_c\}$, permitem, teoricamente, estimar regiões promissoras para introduzir novos indivíduos. Para tanto, uma distribuição normal multivariada é usada para gerar externamente essa população de imigrantes $\{P_i\}$. Ressalte-se, que para evitar convergência prematura, para cada região promissora também deverá ser gerado um percentual de indivíduos aleatórios, para manter a capacidade de exploração por novas regiões do algoritmo. A população de imigrantes é então adicionada a população atual para participar do processo de geração dos vetores mutantes.

3.2 Descrição dos Algoritmos

Seguindo as ideias apresentadas para o modelo conceitual de DE, proposto na seção anterior, foram elaborados dois algoritmos, para o tratamento de problemas mono e multi-objetivo.

3.2.1. Abordagem Mono-Objetivo

A abordagem mono-objetivo, tem por base o algoritmo DE, usando o esquema original de mutação o DE/*rand*/1/bin, por sua simplicidade e por ser recorrente em aplicações de controle na literatura (COELHO *et al.*, 2010; SAHU *et al.*, 2013; MOHANTY *et al.*, 2014). No entanto, a estratégia de inserção de imigrantes direcionados concebida, pode ser também adaptada a outros tipos de esquemas da DE e para outros AEs. Antes de apresentar a estratégia é importante tratar algumas questões básicas que a geração de imigrantes para um EA envolve (YU *et al.*, 2008).

- Como gerar imigrantes?
- Qual o tamanho da população de imigrantes?
- Qual a estratégia de substituição de indivíduos na população por imigrantes ou como inserir os imigrantes no processo evolucionário?
- Como aumentar a probabilidade de sobrevivência dos imigrantes recém introduzidos?

Na geração dos imigrantes será usada uma abordagem indireta híbrida, ou seja, uma combinação de imigrantes aleatórios e imigrantes direcionados. A população de imigrantes terá tamanho igual ao da população total NP e será dividida igualmente pelo número de grupos N_c definidos no início da execução. Portanto, para cada grupo haverá uma população de imigrantes vinculada com aproximadamente NP/N_c membros. Além disso, os imigrantes do grupo serão divididos em duas subpopulações: uma de indivíduos gerados aleatoriamente (um

percentual mínimo de 10%, definido após algumas simulações de teste) e outra de imigrantes gerados mediante uma distribuição normal multivariada, usando como base as características atualizadas do grupo, ou seja, o centro e a matriz de covariância adaptada. As duas subpopulações de imigrantes dentro de cada região terão funções distintas: a subpopulação aleatória terá caráter exploratório, já a outra subpopulação, terá função explotatória, pois seus indivíduos estariam dispostos em regiões mais promissoras.

Em relação a substituição de indivíduos da população por imigrantes, uma inovação proposta nesse trabalho, é não inserir os imigrantes gerados diretamente na população, mas sim, usá-los como banco de indivíduos para serem, em conjunto com a população atual, sorteados durante a etapa de geração dos mutantes na DE. Essa estratégia visa reduzir o custo com avaliações da função de aptidão, pois, os imigrantes não são avaliados e contribuem apenas com a informação da sua localização no espaço de busca. Além disso, visando aumentar a probabilidade dos imigrantes contribuírem na geração dos vetores mutantes, optou-se por restringir o sorteio dos vetores base entre os imigrantes e os indivíduos da população vinculados ao mesmo agrupamento do vetor alvo corrente (Algoritmo 3.1).

Para abordagem mono-objetivo do algoritmo, os centros das regiões foram definidos mediante o indivíduo mais bem avaliado de cada agrupamento. Buscando com isso, explorar as regiões que seguem a direção do caminho percorrido pelos indivíduos mais aptos. Outra característica inserida ao modelo de AE proposto diz respeito ao tratamento dos agrupamentos considerados menos viáveis. O número de grupos inicial é definido no início do algoritmo, no entanto, em virtude do processo de seleção, a quantidade de indivíduos de alguns agrupamentos pode diminuir bastante. Esses agrupamentos e seus componentes não são excluídos diretamente, entretanto, como punição deixam de receber imigrantes. Portanto, a probabilidade da região em torno deles seja pesquisada diminui e caso a aptidão dos seus membros não seja alta esses grupos tendem a desaparecer com o tempo.

Os Algoritmos 3.1 e 3.2 apresentam o AE proposto, para problemas mono-objetivo, e a função para geração de imigrantes, respectivamente. Nos experimentos

realizados, o algoritmo de agrupamento utilizado foi o *k-means*, pois para os problemas testados foi eficiente e apresentou baixo custo computacional.

ALGORITMO 3.1 – EVOLUÇÃO DIFERENCIAL COM INSERÇÃO DE IMIGRANTES DIRECIONADOS MONO-OBJETIVO

- 1: Defina os parâmetros do algoritmo DE: F = fator de mutação, Cr = fator de cruzamento e a quantidade de agrupamentos prevista N_c ;
- 2: Gere uma população inicial P_0 com NP indivíduos.
- 3: Avalie P_0 usando a função objetivo $f(X_{i,0})$.
- 4: Identifique na população N_c agrupamentos (e.g., usando o *k-means*)
- 5: Marque os membros do agrupamento com o número do grupo
- 6: Redefina o centro do grupo para o indivíduo mais apto
- 7: Inicialize o contador de gerações $G = 0$.
- 8: ENQUANTO critério de finalização não for alcançado.
- 9: PARA $i = 1$ ATÉ NP (Para cada indivíduo da população atual)
- 10: Sorteie no grupo do vetor alvo i um vetor base ($X_{ib,G}$)
- 11: Sorteie dois vetores distintos na população ($X_{i1,G}, X_{i2,G}$).
- 12: Gere um vetor mutante como uma combinação

$$V_{i,G} = X_{ib,G} + F \times (X_{i1,G} - X_{i2,G}).$$
- 13: Gere um vetor tentativa cruzando o indivíduo atual com
o vetor mutante $U_{i,G} = cross(Cr, X_{i,G}, V_{i,G})$.
- 14: Avalie o vetor tentativa.
- 15: SE $f(U_{i,G}) \leq f(X_{i,G})$ ENTÃO. (Problema de mínimo)
- 16: $X_{i,G+1} = U_{i,G}$.
- 17: Atribua ao vetor tentativa o grupo cujo centro lhe é
mais próximo.
- 18: FIM DO SE.
- 19: FIM DO PARA.
- 20: Atualize o contador de gerações $G = G + 1$.
- 21: Avalie o critério de Parada
- 22: Chame a Função para Geração de Imigrantes
- 23: Mescle as populações de imigrantes e atual para o sorteio
- 24: FIM DO ENQUANTO.

25: Retorne o ponto ótimo X_{PG}^* .

Algoritmo da função para geração de imigrantes

ALGORITMO 3.2 – FUNÇÃO PARA GERAÇÃO DE IMIGRANTES

- 1: Receba os membros, as avaliações e os centros atuais dos grupos;
- 2: Identifique quantos grupos estão ativos ($|Grupo_i| \geq \text{Dim}(X_i)$)
- 3: Atualize N_c
- 4: Defina a quantidade de imigrantes para cada grupo (N_i)
- 5: PARA $i = 1$ ATÉ N_c .
 - 6: Identifique indivíduo melhor avaliado: x_{melhor}
 - 7: Calcule a direção: $y_w = x_{melhor} - \text{centro}_i$
 - 8: Atualize o centro: $\text{centro}_i = x_{melhor}$
 - 9: Calcule a matriz de covariância do agrupamento.
 - 10: Atualiza a matriz de covariância: $C_{atual} = (\alpha-1)C + \alpha(y_w \cdot y_w^T)$
 - 11: Use a distribuição normal $N(\text{centro}_i, C_{atual})$ e gere N_i imigrantes
 - 12: Substitua um percentual dos imigrantes por outros aleatórios
 - 13: Atribua o índice do grupo para os imigrantes gerados
 - 14: FIM DO PARA.
 - 15: Retorne a população de imigrantes e os centros atualizados.

O Algoritmo 3.1 segue com algumas modificações o Algoritmo 2.2 para DE canônica, portanto aqui serão ressaltadas apenas as principais diferenças entre ambos. Nas linhas 4-6, foram incluídas as etapas de agrupamento dos membros da população, do cálculo da matriz de covariância e da definição do centro de cada grupo; na linha 10 o sorteio do vetor base acontece dentro do grupo pertencente ao vetor alvo atual; na linha 17 o vetor tentativa vencedor é associado ao grupo, cujo centro lhe é mais próximo, e finalmente nas linhas 22-23 a função de geração de imigrantes é chamada e a população atual é mesclada com a população de imigrantes.

O Algoritmo 3.2, função para geração de imigrantes, realiza as seguintes ações. Inicialmente, linha 1, recebe a população atual, as avaliações dos indivíduos

e os centros. Em seguida, nas linhas 2-4 é avaliado quantos e quais os grupos são viáveis e qual será a quantidade de imigrantes para cada grupo. Nas linhas 5-14, para cada grupo, a rotina realiza as mesmas operações, ou seja, identifica o melhor indivíduo; calcula a direção entre esse indivíduo e o centro do grupo; redefine o centro como o melhor indivíduo; atualiza a matriz de covariância; gera imigrantes usando a distribuição normal multivariada, com o centro e a matriz de covariância atualizada; substitui uma parcela dos imigrantes por outros aleatórios e vincula os imigrantes ao seu respectivo agrupamento. Finalmente na linha 15, a população de imigrantes e os novos centros são devolvidos ao algoritmo principal.

3.2.2. Abordagem Multi-objetivo

Considerando que o projeto de um sistema de controle normalmente envolve múltiplos objetivos, é importante a extensão da estratégia desenvolvida para problemas de otimização multi-objetivo. O algoritmo proposto usa como base o algoritmo NSDE, que, conforme apresentado no capítulo anterior, consiste numa combinação da DE com o mecanismo de seleção de indivíduos definido pelo algoritmo NSGA II. A união da estratégia proposta e o NSDE visa sobretudo explorar regiões mais promissoras do espaço de busca, usando os imigrantes direcionados, enquanto espalha os indivíduos ao longo da fronteira de Pareto, mediante o mecanismo de seleção do NSGA II.

A escolha do NSDE, nesse trabalho, deve-se a fatores como a sua capacidade para tratamento de problemas com vetores reais, seu mecanismo de seleção de indivíduos ser computacionalmente menos custoso do que o usado por outros algoritmos, por exemplo SPEA2, e uso na literatura da DE com sucesso como ferramenta de otimização. Além disso, Sousa *et al.* (2014) apresentaram um AE multi-objetivo para sintonia de parâmetros de um controlador MFAC, também baseado no NSDE, mas com introdução de imigrantes aleatórios, visando melhorar a diversidade da população e o espalhamento das soluções obtidas ao longo da fronteira de Pareto.

Os Algoritmos 3.3 e 3.4 apresentam a versão multi-objetivo do AE proposto e sua função para geração de imigrantes, respectivamente:

ALGORITMO 3.3 – EVOLUÇÃO DIFERENCIAL COM INSERÇÃO DE IMIGRANTES DIRECIONADOS MULTI-OBJETIVO.

- 1: Defina os parâmetros do algoritmo: F = fator de mutação, Cr = fator de cruzamento, Nc = quantidade de agrupamentos prevista.
- 2: Gere uma população inicial P_0 com NP indivíduos.
- 3: Avalie P_0 usando as funções objetivo.
- 4: Identifique na população Nc agrupamentos: (centros e membros)
- 5: Marque os membros do agrupamento com o número do grupo.
- 6: Inicialize o contador de gerações $G = 0$.
- 7: ENQUANTO critério de finalização não for alcançado.
- 8: PARA $i = 1$ ATÉ NP (Para cada indivíduo da população atual)
- 9: Sorteie no grupo do vetor i um vetor base ($X_{ib,G}$)
- 10: Sorteie dois vetores distintos na população ($X_{i1,G}, X_{i2,G}$).
- 11: Gere um vetor mutante como uma combinação
 $V_{i,G} = X_{ib,G} + F \times (X_{i1,G} - X_{i2,G})$.
- 12: Gere um vetor tentativa cruzando o indivíduo atual com
o vetor mutante $U_{i,G} = cross(Cr, X_{i,G}, V_{i,G})$.
- 13: Avalie o vetor tentativa.
- 14: Atribua ao $U_{i,G}$ o grupo cujo centro lhe é mais próximo
- 15: Adicione $U_{i,G}$ a uma população definida como em
Avanço.
- 16: FIM DO PARA.
- 17: Atualize o contador de gerações $G = G + 1$.
- 18: Mescle a população atual e a população em avanço
- 19: Use o mecanismo de seleção do NSGA II separe NP indivíduos
- 20: Avalie o critério de Parada
- 21: Chame a Função Para Geração de Imigrantes
- 22: Mescle as populações de imigrantes e atual para o sorteio
- 23: FIM DO ENQUANTO.
- 24: A População Final como uma estimativa do conjunto de Pareto.

ALGORITMO 3.4 – FUNÇÃO PARA GERAÇÃO DE IMIGRANTES

- 1: Receba os membros e os centros dos grupos;
- 2: Identifique quantos grupos estão ativos
- 3: Defina a quantidade de imigrantes para cada grupo Nr
- 4: PARA $i = 1$ ATÉ Nc .
- 5: Calcule o ponto médio de cada agrupamento: x_{medio}
- 6: Calcule a direção: $y_w = x_{medio} - centro_i$
- 7: Atualize o centro: $centro_i = x_{medio}$
- 8: Calcule a matriz de covariância do agrupamento.
- 9: Atualiza a matriz de covariância: $C_{atual} = (\alpha-1)C + \alpha(y_w \cdot y_w^T)$
- 10: Use a distribuição normal $N(centro_i, C_{atual})$ e gere Nr imigrantes
- 11: Substitua um percentual dos imigrantes por outros aleatórios
- 12: Atribua o índice do grupo para os imigrantes gerados
- 13: FIM DO PARA.
- 14: Retorne a população de imigrantes e os centros

O Algoritmo 3.3 é semelhante a versão mono-objetivo já apresentada, no entanto, para os centros dos agrupamentos foram definidos os seus pontos médios. Seguindo, o Algoritmo 3.3, nas linhas 4-5, é realizado o agrupamento da população e a definição dos centros como os pontos médios. O sorteio do vetor base para produzir o vetor mutante, na linha 9, é feito entre os indivíduos, da população atual ou imigrantes, que pertençam ao mesmo agrupamento do vetor alvo corrente. Na linha 14 o vetor tentativa é associado a algum dos agrupamentos existentes. Diferente do Algoritmo 3.1, no qual o processo de seleção é uma comparação direta entre o vetor tentativa e o alvo correspondente, na linha 15 o vetor tentativa é adicionado a uma população em avanço, formada pelos vetores tentativa obtidos durante toda a geração. Na linha 18 essa população em avanço é mesclada com a população corrente e na linha 19 o mecanismo de seleção do NSGA II é usado para selecionar os indivíduos que formarão a população na próxima geração. Em seguida, na linha 21, o algoritmo chama a função de geração de imigrantes.

O algoritmo para função de geração de imigrantes é praticamente idêntico ao utilizado no caso mono-objetivo, portanto, não será aprofundado. As únicas

alterações relevantes estão relacionadas ao fato da função não receber como entrada as avaliações dos indivíduos da população e, na linha 5, a definição do centro do agrupamento é feita calculando seu ponto médio.

3.3 Controle Adaptativo Evolucionário *Off-Line*

O problema de sintonia *off-line* dos parâmetros de um controlador pode ser transformado em um típico problema de otimização, na Figura 7 é apresentado um esquema de como esse processo é realizado. No diagrama apresentado, cada conjunto de parâmetros do controlador corresponde a um indivíduo da população e sua aptidão é avaliada pela simulação *off-line* do sistema controlado. A população inicial aleatória de NP indivíduos é gerada dentro da região factível e classificada em grupos. Os indivíduos dessa população formam a primeira população corrente e passam pelas operações de mutação e cruzamento gerando os vetores tentativa, que são classificados de acordo com os agrupamentos existentes. Ao final desse processo, uma nova população de NP indivíduos é gerada, denominada população em avanço. As populações corrente e em avanço são mescladas e os indivíduos avaliados, simulando o sistema controlado. Em seguida, os NP indivíduos da população mesclada são selecionados usando o mecanismo de seleção do algoritmo NSGA II, baseado no *ranking* de soluções não-dominadas e no valor calculado da distância de multidão. Depois, as propriedades dos grupos da população resultante são atualizadas e usadas para gerar a população de imigrantes que são usados na operação de mutação. Todo o processo é repetido até alguma condição de parada ser alcançada. Ao final o critério adotado pelo tomador escolhe no conjunto de Pareto a melhor combinação de parâmetros para ser usado no sistema de controle.

solução de problemas envolvendo múltiplos critérios de decisão, ou cujas equações das funções objetivo são desconhecidas ou muito complexas, o que pode ser útil no projeto de um sistema de controle. Neste trabalho, busca-se analisar as dificuldades elencadas e desenvolver uma estratégia de controle baseada na combinação do controle MFAC, cuja a aplicação em sistemas complexos é viável, e no algoritmo NSDE com inserção de imigrantes direcionados, proposto no trabalho.

A estratégia de controle desenvolvida tem por objetivo ser aplicada principalmente ao controle de sistemas não-lineares variantes no tempo, cuja complexidade do seu equacionamento dificulta o uso de técnicas de controle moderno. Em todos os casos estudados, o objetivo do controle será manter o sistema dentro de uma trajetória prescrita. Para melhor compreensão, a estratégia de controle adaptativo evolucionário proposta será dividida nas etapas a seguir:

- Na etapa *off-line*, de experimentação (REYNOSO-MEZA *et al.*, 2012), os parâmetros do sistema de controle são otimizados, usando o AEMO proposto, considerando que os membros da população serão formados pelos parâmetros do controlador, usados na simulação da planta controlada durante todo o intervalo de operação. A população obtida ao final da otimização *off-line*, que aproxima o conjunto de Pareto, é armazenada para ser usada como população inicial, na estratégia *on-line*. Essa informação será importante para acelerar as otimizações que ocorrerão na segunda etapa.
- Na etapa *on-line*, diferente do sugerido por Reynoso-Meza *et al.* (2012), a estratégia proposta não será executada a cada instante, pois isso consome muito tempo de execução e, em experimentos realizados, observou-se que variações bruscas nos parâmetros do controlador podem resultar em perturbações adicionais ao sistema. No método proposto, o conjunto de parâmetros escolhidos pelo tomador de decisões na primeira etapa será usado como parâmetros iniciais para o controlador. Portanto, o ponto chave da estratégia de ajuste consiste em identificar instantes, dentro do período de operação da planta, nos quais seria interessante uma nova

otimização para definir outros parâmetros do controlador. Quando esse instante é detectado, uma nova otimização é realizada usando o AE e um sistema secundário para simular sistema controlado, dentro de um intervalo de tempo pré-definido (janela) em avanço. No AE, o sistema secundário é usado para avaliar a aptidão dos membros da população durante a evolução. A otimização resultará em novos parâmetros para serem usados no controlador, dentro da janela de tempo usada nas simulações. Ao final desse intervalo, os valores dos parâmetros do controlador devem retornar aos iniciais, pois, estes foram considerados os melhores parâmetros para todo o período de operação do sistema. Portanto, os novos parâmetros têm validade apenas dentro do intervalo usado na nova otimização. Além disso, como medida de segurança, antes de inserir um novo conjunto de parâmetros no sistema de controle, os valores obtidos para o vetor de funções objetivo calculados usando os novos parâmetros e os parâmetros iniciais são comparados, ou seja, os novos só substituirão os iniciais se o sistema obtiver um melhor desempenho usando os últimos.

A Figura 8 apresenta o esquema de funcionamento do controle adaptativo evolucionário. No diagrama, o sistema controlado, a planta e o controlador MFAC reais, opera usando um conjunto de parâmetros iniciais, otimizados mediante a estratégia de sintonia *off-line*. Entretanto, durante a execução, os sinais de referência (em avanço) e de erro são medidos e empregados para definir situações nas quais novos parâmetros ótimos serão calculados. Quando isso ocorre, um sistema secundário, modelo matemático identificado para o sistema controlado, e o AE proposto são usados para encontrar novos parâmetros ótimos para o controlador, considerando um intervalo de tempo futuro. Visando acelerar a busca, o AE usa sempre como população inicial, a cada nova otimização, o conjunto de Pareto obtido na etapa *off-line*. Também no esquema, um mecanismo de comutação é empregado para manter os novos parâmetros calculados válidos apenas dentro da janela de otimização e para, ao final, retornar ao controlador os parâmetros definidos inicialmente. Todo esse processo é realizado durante a operação da planta.

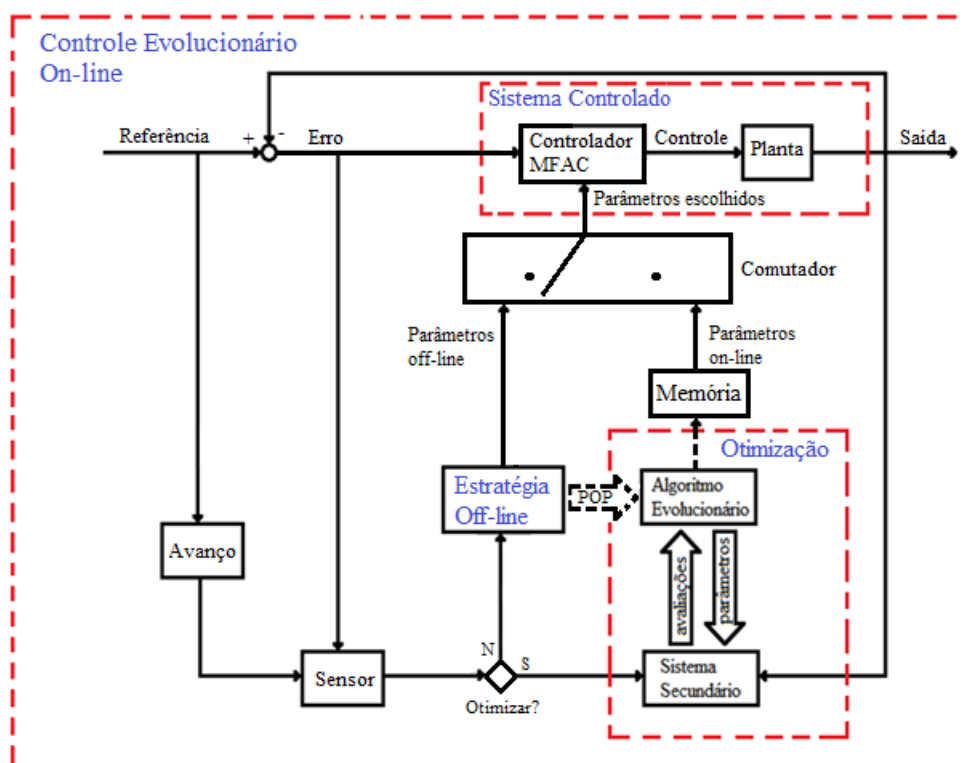


Figura 8 – Esquema para sintonia *on-line* dos parâmetros do controlador.

Na estratégia de otimização *on-line* proposta, duas situações são elegíveis para uma nova otimização. No primeiro caso, considerando que a função de rastreo é conhecida antecipadamente, a estratégia proposta verifica durante a simulação do sistema controlado, porém com alguns instantes de antecedência, se a função de referência apresentará alguma descontinuidade (pontos onde a função não é diferenciável). Portanto, nesse caso, a otimização dos parâmetros do controlador ocorrerá alguns instantes antes do sistema chegar ao ponto de descontinuidade do sinal de referência. Com isso, busca-se reduzir os efeitos nocivos da variação brusca na referência sobre o comportamento do sistema. Outra situação, também passível de nova otimização, ocorre se o erro da resposta do sistema for superior a um determinado limiar pré-fixado, normalmente resultado de não-linearidades intrínsecas ao sistema. Nesse caso, a nova otimização ocorre a partir desse instante de detecção.

3.5 Discussões Finais

Nas seções anteriores, baseado nas características esperadas pela dinâmica de uma população durante o processo simulado de evolução natural, foi inicialmente apresentado um modelo conceitual para a proposta de um AE. Em seguida, com base no modelo, foram elaborados dois algoritmos evolucionários que podem ser usados na solução de problemas de otimização mono e multi-objetivo. Depois, foi mostrado como o AE multi-objetivo proposto pode ser aplicado num projeto de controle, como ferramenta de otimização para o ajuste *off-line* dos parâmetros do controlador. Ao final, também foi proposta uma estratégia de sintonia *on-line*, buscando usar as qualidades do algoritmo evolucionário para solução de problemas complexos e ao mesmo tempo minimizar suas dificuldades em aplicações envolvendo controle. No próximo capítulo serão apresentados resultados para validação dos algoritmos propostos e das estratégias de controle apresentadas.

4

RESULTADOS DOS EXPERIMENTOS

Neste capítulo, inicialmente, serão apresentados testes para validação dos algoritmos evolucionários propostos. Para tanto, foram usados problemas clássicos de otimização, do tipo mono e multi-objetivo, e os resultados permitem a comparação entre o modelo proposto e outros algoritmos evolucionários do estado da arte.

Na segunda parte do capítulo, serão mostrados os resultados dos casos de estudo escolhidos, considerando a sintonia *off-line* e *on-line* de um controlador MFAC-CFDL. A análise dos resultados será feita mediante comparações entre a performance das abordagens *on-line* e *off-line* e os resultados já obtidos na literatura para os mesmos problemas.

Ao final, as conclusões sobre o desempenho do algoritmo proposto, na solução de problemas mono e multi-objetivo e para as estratégias de sintonia para o controlador serão apresentadas.

4.1 Testes para Validação do Algoritmo

Para validação da versão do algoritmo mono-objetivo, foram usadas as 28 funções apresentadas no concurso de otimização do IEEE *Congress on Evolutionary Computation* – CEC (2013). Já para a segunda versão do algoritmo, foram usados os 10 primeiros problemas de otimização multi-objetivo do CEC (2009), nomeados como de otimização sem restrição, cuja definição, com limitação apenas para área factível das soluções, se assemelha mais com os problemas de controle abordados na segunda parte desse capítulo.

4.1.1. Testes para Problemas Mono-Objetivo

Em geral um problema de otimização mono-objetivo estático pode ser enunciado como: minimize (ou maximize) uma função $f(\mathbf{x})$ considerando m restrições de desigualdade, $g_i(\mathbf{x}) \leq 0, i = 1, \dots, m$, e n restrições de igualdade, $h_j(\mathbf{x}) = 0, j = 1, \dots, n$, tal que $\mathbf{x} = [x_1 \dots x_D]^T \in \Omega$, onde $\Omega = [x_{min}, x_{max}]^D$ é o espaço de busca.

O concurso do CEC (2013), consiste em minimizar 28 funções, sem restrições de desigualdade ou igualdade, com o espaço de busca limitado a $[-100, 100]^D$. As 28 funções propostas podem ser agrupadas em três classes: funções unimodais, funções multimodais e funções compostas. A Tabela 3 apresenta um quadro com o tipo e o nome das funções e seu respectivo ponto ótimo (LIANG *et al.*, 2013).

Tabela 3 - Relação de funções do CEC2013.

Tipo	Nº	Nome	Valor Ótimo
Unimodais	1	<i>Sphere Function</i>	-1400
	2	<i>Rotated High Conditioned Elliptic Function</i>	-1300
	3	<i>Rotated Bent Cigar Function</i>	-1200
	4	<i>Rotated Discuss Function</i>	-1100
	5	<i>Different Powers Function</i>	-1000
Multimodais	6	<i>Rotated Rosenbrock's Function</i>	-900
	7	<i>Rotated Schaffers F7 Function</i>	-800
	8	<i>Rotated Ackley's Function</i>	-700
	9	<i>Rotated Weierstrass Function</i>	-600
	10	<i>Rotated Griewank's Function</i>	-500
	11	<i>Rastrigin's Function</i>	-400
	12	<i>Rotated Rastrigin's Function</i>	-300
	13	<i>Non-Continuous Rotated Rastrigin's Function</i>	-200
	14	<i>Schwefel's Function</i>	-100
	15	<i>Rotated Schwefel's Function</i>	100
	16	<i>Rotated Katsuura Function</i>	200
	17	<i>Lunacek Bi_Rastrigin Function</i>	300
	18	<i>Rotated Lunacek Bi_Rastrigin Function</i>	400
	19	<i>Expanded Griewank's plus Rosenbrock's Function</i>	500
	20	<i>Expanded Scaffer's F6 Function</i>	600
Compostas	21	<i>Composition Function 1 (n=5,Rotated)</i>	700
	22	<i>Composition Function 2 (n=3,Unrotated)</i>	800
	23	<i>Composition Function 3 (n=3,Rotated)</i>	900
	24	<i>Composition Function 4 (n=3,Rotated)</i>	1000
	25	<i>Composition Function 5 (n=3,Rotated)</i>	1100
	26	<i>Composition Function 6 (n=5,Rotated)</i>	1200
	27	<i>Composition Function 7 (n=5,Rotated)</i>	1300
	28	<i>Composition Function 8 (n=5,Rotated)</i>	1400

Os resultados obtidos, após 51 repetições, estão dispostos nas Tabelas 4-7 para as dimensões $D = 5, 10, 30, 50$. Como critérios de parada foram adotados o número de avaliações da função (FES) e o valor do erro, ou seja, $FES \geq 10^4 \cdot D$ ou $Erro \leq 10^{-8}$. Conforme orientação do CEC 2013 valores menores ou iguais a 10^{-8} devem ser considerados como zero.

Nas Tabelas 4-7 é possível comparar as médias do erro e o desvio padrão, e também a quantidade de média de FES, para cada função, entre o algoritmo proposto, o algoritmo DE/*rand*/1/bin, e o algoritmo CMA-ES. A escolha desses algoritmos para comparação se justifica pela necessidade de avaliar a eficiência do algoritmo proposto frente a um modelo canônico (DE/*rand*/1/bin) e ao algoritmo CMA-ES, que é reconhecidamente eficiente na solução de problemas de otimização mono-objetivo (HANSEN, 2006). Em negrito estão destacados os menores valores médios dos erros obtidos para cada função.

Tabela 4 - Resultados das simulações para D = 5.

F	DE/ <i>rand</i> /1/bin			CMA-ES			DE/ <i>rand</i> /1/bin + Imi. Direcionados		
	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES
1	0,00	0,00	6625,49	0,00	0,00	1029,96	0,00	0,00	4218,63
2	0,00	0,00	18357,84	0,00	0,00	2795,45	0,00	0,00	10245,1
3	0,00	0,00	20615,69	9,37E+02	6,69E+03	5089,88	0,00	0,00	11925,49
4	0,00	0,00	16350,98	0,00	0,00	2766,43	0,00	0,00	9150,98
5	0,00	0,00	7922,55	0,00	0,00	3419,76	0,00	0,00	5097,06
6	1,13E-01	1,71E-01	36017,65	1,54E-01	7,71E-01	3798,12	7,71E-02	5,50E-01	13218,63
7	3,95E-07	2,04E-06	26307,84	8,71E+08	3,83E+09	5,00E+04	9,17E-06	2,83E-05	39433,33
8	1,99E+01	8,86E-01	5,00E+04	2,09E+01	4,30E-01	5,00E+04	1,91E+01	2,89	5,00E+04
9	3,36E-01	7,08E-01	43002,94	6,80	3,58E-01	5,00E+04	1,30	1,14	48573,53
10	4,89E-02	2,09E-02	5,00E+04	5,32E-02	3,41E-02	5,00E+04	9,43E-02	2,52E-02	5,00E+04
11	0,00	0,00	18262,75	9,85E+01	3,69E+01	5,00E+04	3,90E-02	1,95E-01	25950,98
12	5,53E-01	7,32E-01	42204,90	7,22E+01	7,44	5,00E+04	8,55E-01	9,75E-01	45932,35
13	8,25E-01	1,10	39293,14	9,12E+01	4,00E+01	5,00E+04	1,28	1,45	44528,43
14	6,23	1,08E+01	49940,20	3,75E+02	0,00	5,00E+04	5,42E+01	2,80E+01	5,00E+04
15	1,80E+02	9,46E+01	5,00E+04	1,25E+03	5,74E+01	5,00E+04	2,02E+02	9,38E+01	5,00E+04
16	1,27E-01	3,78E-02	5,00E+04	3,52E-01	3,76E-01	48128,31	5,59E-01	1,45E-01	5,00E+04
17	4,94	7,02E-01	5,00E+04	3,96E+01	4,68	5,00E+04	4,71	1,64	5,00E+04
18	6,54	9,33E-01	5,00E+04	3,94E+01	5,78	5,00E+04	7,73	9,20E-01	5,00E+04
19	1,57E-01	1,25E-01	48268,63	4,91E-01	2,72E-01	5,00E+04	2,14E-01	1,49E-01	47922,55
20	3,58E-01	2,48E-01	5,00E+04	NaN	NaN	5,00E+04	3,50E-01	2,80E-01	5,00E+04
21	2,69E+02	7,35E+01	5,00E+04	2,73E+02	1,10E+02	5,00E+04	1,29E+02	6,72E+01	5,00E+04
22	1,53E+02	1,65E+01	5,00E+04	8,83E+02	0,00	5,00E+04	2,26E+02	6,47E+01	5,00E+04
23	3,98E+02	1,05E+02	5,00E+04	1,46E+03	1,23E+02	5,00E+04	3,53E+02	1,39E+02	49604,9
24	1,27E+02	3,79E+01	5,00E+04	2,78E+02	1,82E+01	5,00E+04	1,26E+02	3,33E+01	5,00E+04
25	1,01E+02	1,51	5,00E+04	1,94E+02	1,42E+01	5,00E+04	1,02E+02	2,62	5,00E+04
26	1,01E+02	6,21E-01	5,00E+04	3,22E+02	5,07E+01	5,00E+04	6,72E+01	4,27E+01	5,00E+04
27	3,29E+02	3,05E+01	5,00E+04	3,28E+02	2,89E+01	5,00E+04	3,46E+02	2,62E+01	5,00E+04
28	2,76E+02	6,51E+01	5,00E+04	1,11E+03	2,86E+01	5,00E+04	2,45E+02	9,01E+01	5,00E+04

Tabela 5 - Resultados das simulações para D = 10.

F	DE/rand/1/bin			CMA-ES			DE/rand/1/bin + Imigrantes		
	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES
1	0,00	0,00	29482,35	0,00	0,00	2025,10	0,00	0,00	13562,75
2	1,18E+01	9,30	1,00E+05	0,00	0,00	7194,12	4,24E-01	2,80	96015,69
3	4,46E-01	3,46E-01	1,00E+05	3,73E-01	1,50	26539,22	7,04E-01	1,73	97454,90
4	7,75E-02	4,31E-02	1,00E+05	0,00	0,00	6393,33	0,00	0,00	78945,10
5	0,00	0,00	39500,00	0,00	0,00	10791,57	0,00	0,00	17988,24
6	2,97E-02	5,91E-02	1,00E+05	2,35E-01	9,47E-01	12274,31	2,49E-07	1,29E-06	59339,22
7	3,52E-03	2,35E-03	1,00E+05	1,41E+01	1,90E+01	1,00E+05	3,28E-02	1,47E-01	99725,49
8	2,04E+01	7,07E-02	1,00E+05	2,09E+01	5,94E-01	1,00E+05	2,04E+01	5,66E-02	1,00E5
9	8,91	6,55E-01	1,00E+05	1,21E+01	3,07	1,00E+05	8,91	5,31E-01	1,00E5
10	5,03E-01	7,16E-02	1,00E+05	1,09E-02	9,85E-03	73336,86	2,15E-01	1,91E-01	96125,49
11	1,41E+01	2,74	1,00E+05	2,15E+02	4,26E+01	1,00E+05	1,08E+01	3,28	98747,06
12	2,71E+01	3,69	1,00E+05	1,61E+02	4,01E+01	1,00E+05	2,13E+01	3,98	1,00E5
13	2,66E+01	4,20	1,00E+05	1,66E+02	6,49E+01	1,00E+05	1,99E+01	4,44	1,00E5
14	1,06E+03	1,40E+02	1,00E+05	1,63E+03	1,65E+02	1,00E+05	9,85E+02	1,78E+02	1,00E5
15	1,56E+03	1,38E+02	1,00E+05	1,42E+03	1,79E+02	1,00E+05	1,47E+03	1,31E+02	1,00E5
16	1,01	1,75E-01	1,00E+05	1,48E-01	1,21E-01	1,00E+05	1,12	2,06E-01	1,00E5
17	2,48E+01	2,43	1,00E+05	1,43E+02	2,18E+01	1,00E+05	2,41E+01	2,73	1,00E5
18	3,60E+01	4,13	1,00E+05	1,47E+02	1,12E+01	1,00E+05	3,16E+01	4,00	1,00E5
19	1,95	2,76E-01	1,00E+05	1,16	5,24E-01	1,00E+05	1,74	3,14E-01	1,00E5
20	4,09	2,59E-01	1,00E+05	NaN	NaN	1,00E+05	3,33	3,53E-01	1,00E5
21	3,96E+02	2,80E+01	1,00E+05	4,00E+02	0,00	1,00E+05	3,67E+02	7,40E+01	1,00E5
22	1,31E+03	1,77E+02	1,00E+05	2,82E+03	9,05E+01	1,00E+05	1,16E+03	2,10E+02	1,00E5
23	1,76E+03	1,52E+02	1,00E+05	2,28E+03	2,22E+02	1,00E+05	1,58E+03	1,91E+02	1,00E5
24	2,23E+02	1,37	1,00E+05	3,15E+02	2,93E+01	1,00E+05	2,23E+02	1,66	1,00E5
25	2,23E+02	1,44	1,00E+05	2,79E+02	1,48E+01	1,00E+05	2,22E+02	1,56	1,00E5
26	2,00E+02	7,77E-06	1,00E+05	4,00E+02	1,62E-01	1,00E+05	1,93E+02	2,52E+01	1,00E5
27	5,30E+02	1,45E+01	1,00E+05	4,00E+02	0,00	1,00E+05	5,28E+02	1,74E+01	1,00E5
28	2,96E+02	2,80E+01	1,00E+05	1,16E+03	1,77E+01	1,00E+05	2,76E+02	6,51E+01	1,00E5

Tabela 6 - Resultados das simulações para D = 30.

F	DE/rand/1/bin			CMA-ES			DE/rand/1/bin + Imigrantes		
	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES
1	5,30E-04	1,34E-04	3,00E+05	0,00	0,00	5432,82	0,00	0,00	76858,82
2	3,23E+08	6,59E+07	3,00E+05	0,00	0,00	45558,47	3,06E+06	1,68E+06	3,00E5
3	1,41E+10	4,13E+09	3,00E+05	3,80E-01	2,05	237762,6	2,01E+07	3,53E+07	3,00E5
4	1,64E+05	2,77E+04	3,00E+05	0,00	0,00	29952,04	3,00E+04	6,76E+03	3,00E5
5	8,35E-02	1,89E-02	3,00E+05	0,00	0,00	80867,29	0,00	0,00	112741,18
6	2,51E+01	2,54E-01	3,00E+05	3,13E-01	1,08	66104,16	1,62E+01	1,01	3,00E5
7	2,99E+02	3,21E+01	3,00E+05	1,78E+01	1,09E+01	3,00E+05	1,45E+01	9,48	3,00E5
8	2,09E+01	5,27E-02	3,00E+05	2,15E+01	1,03E-01	3,00E+05	2,10E+01	4,26E-02	3,00E5
9	3,96E+01	1,11	3,00E+05	4,74E+01	3,27	3,00E+05	3,98E+01	1,05	3,00E5
10	1,95E+02	5,25E+01	3,00E+05	2,09E-02	1,71E-02	260035,2	4,15E-02	2,54E-02	3,00E5
11	1,81E+02	1,09E+01	3,00E+05	5,56E+02	2,36E+02	3,00E+05	1,20E+02	1,49E+01	3,00E5
12	2,16E+02	1,09E+01	3,00E+05	9,42E+02	1,96E+01	3,00E+05	1,61E+02	1,06E+01	3,00E5
13	2,17E+02	1,03E+01	3,00E+05	1,09E+03	4,44E+01	3,00E+05	1,59E+02	9,73	3,00E5
14	7,23E+03	2,68E+02	3,00E+05	4,98E+03	3,42E+02	3,00E+05	6,73E+03	2,98E+02	3,00E5
15	7,91E+03	2,44E+02	3,00E+05	5,01E+03	3,22E+02	3,00E+05	7,44E+03	3,03E+02	3,00E5
16	2,46	2,90E-01	3,00E+05	5,88E-02	2,45E-02	3,00E+05	2,44	2,86E-01	3,00E5
17	2,15E+02	1,06E+01	3,00E+05	7,85E+02	1,06E+02	3,00E+05	1,65E+02	1,12E+01	3,00E5
18	2,43E+02	9,36	3,00E+05	6,09E+02	3,16E+02	3,00E+05	1,87E+02	7,89	3,00E5
19	1,81E+01	1,01	3,00E+05	3,82	9,45E-01	3,00E+05	1,25E+01	1,13	3,00E5
20	1,50E+01	6,09E-06	3,00E+05	NaN	NaN	3,00E+05	1,45E+01	6,27E-01	3,00E5
21	3,06E+02	3,65E+01	3,00E+05	3,42E+02	6,61E+01	3,00E+05	2,81E+02	7,13E+01	3,00E5
22	7,61E+03	3,29E+02	3,00E+05	8,02E+03	1,87E+02	3,00E+05	6,98E+03	3,20E+02	3,00E5
23	8,13E+03	2,39E+02	3,00E+05	7,01E+03	3,50E+02	3,00E+05	7,65E+03	3,87E+02	3,00E5
24	3,00E+02	2,55	3,00E+05	6,77E+02	6,94E+01	3,00E+05	3,00E+02	3,06	3,00E5
25	2,99E+02	3,09	3,00E+05	4,41E+02	3,41	3,00E+05	2,99E+02	2,51	3,00E5
26	3,78E+02	3,04E+01	3,00E+05	3,27E+02	7,47	3,00E+05	3,51E+02	5,03E+01	3,00E5
27	1,29E+03	2,82E+01	3,00E+05	5,76E+02	9,66E+01	3,00E+05	1,30E+03	2,73E+01	3,00E5
28	3,03E+02	3,10E-01	3,00E+05	4,82E+03	3,20E+02	3,00E+05	3,00E+02	0,00	3,00E5

Tabela 7 - Resultados das simulações para D = 50.

F	DE/rand/1/bin			CMA-ES			DE/rand/1/bin + Imigrantes		
	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES	Erro (μ)	(σ)	FES
1	2,04E+02	2,76E+01	5,00E+05	0,00	0,00	8253,24	0,00	0,00	173147,06
2	1,67E+09	2,13E+08	5,00E+05	0,00	0,00	121592,06	4,91E+06	2,95E+06	5,00E5
3	1,23E+12	2,07E+11	5,00E+05	6,99E+01	3,95E+02	479692,94	4,66E+07	5,66E+07	5,00E5
4	2,52E+05	4,06E+04	5,00E+05	0,00	0,00	69427,35	7,98E+04	1,41E+04	5,00E5
5	4,00E+02	4,47E+01	5,00E+05	0,00	0,00	221189,71	0,00	0,00	275588,24
6	1,44E+02	1,74E+01	5,00E+05	2,35E-01	9,47E-01	142824,12	4,42E+01	7,08	5,00E5
7	7,37E+02	8,28E+01	5,00E+05	2,31E+01	8,71	5,00E+05	2,82E+01	1,20E+01	5,00E5
8	2,11E+01	4,11E-02	5,00E+05	2,15E+01	7,17E-02	5,00E+05	2,11E+01	3,57E-02	5,00E5
9	7,29E+01	1,22	5,00E+05	6,96E+01	3,39	5,00E+05	7,28E+01	1,48	5,00E5
10	5,69E+03	7,62E+02	5,00E+05	2,67E-02	1,64E-02	4,81E+05	6,05E-02	3,64E-02	5,00E5
11	4,29E+02	1,84E+01	5,00E+05	9,03E+02	3,65E+01	5,00E+05	2,66E+02	1,89E+01	5,00E5
12	5,03E+02	1,82E+01	5,00E+05	1,10E+03	1,26E+01	5,00E+05	3,38E+02	1,40E+01	5,00E5
13	5,05E+02	1,95E+01	5,00E+05	1,32E+03	3,60E+01	5,00E+05	3,36E+02	1,66E+01	5,00E5
14	1,38E+04	3,19E+02	5,00E+05	8,86E+03	2,96E+02	5,00E+05	1,30E+04	3,68E+02	5,00E5
15	1,48E+04	3,46E+02	5,00E+05	9,61E+03	2,64E+02	5,00E+05	1,43E+04	3,51E+02	5,00E5
16	3,36	3,08E-01	5,00E+05	5,33E-02	2,43E-02	5,00E+05	3,29	2,83E-01	5,00E5
17	5,02E+02	1,48E+01	5,00E+05	1,15E+03	2,84E+01	5,00E+05	3,41E+02	1,19E+01	5,00E5
18	5,33E+02	2,13E+01	5,00E+05	8,33E+02	4,46E+02	5,00E+05	3,66E+02	1,59E+01	5,00E5
19	5,87E+02	2,47E+02	5,00E+05	6,57	1,35	5,00E+05	2,19E+01	5,02	5,00E5
20	2,50E+01	2,65E-03	5,00E+05	NaN	NaN	5,00E+05	2,42E+01	9,75E-01	5,00E5
21	8,59E+02	4,71E+02	5,00E+05	9,54E+02	1,42E+02	5,00E+05	6,67E+02	4,37E+02	5,00E5
22	1,42E+04	3,75E+02	5,00E+05	1,38E+04	3,05E+02	5,00E+05	1,31E+04	5,09E+02	5,00E5
23	1,49E+04	3,88E+02	5,00E+05	1,23E+04	4,52E+02	5,00E+05	1,44E+04	3,49E+02	5,00E5
24	3,84E+02	3,83	5,00E+05	1,52E+03	5,03E+01	5,00E+05	3,83E+02	4,01	5,00E5
25	3,82E+02	3,59	5,00E+05	7,19E+02	3,68	5,00E+05	3,82E+02	3,16	5,00E5
26	4,83E+02	7,81	5,00E+05	3,61E+02	9,79	5,00E+05	4,81E+02	1,47E+01	5,00E5
27	2,13E+03	4,18E+01	5,00E+05	8,82E+02	1,09E+02	5,00E+05	2,13E+03	3,35E+01	5,00E5
28	3,03E+02	3,10E-01	5,00E+05	8,92E+03	2,63E+02	5,00E+05	4,60E+02	4,26E+02	5,00E5

Os procedimentos estatísticos usados para analisar experimentos, a depender do tipo de dados considerados, podem ser classificados em dois grupos: testes paramétricos e não-paramétricos. Testes paramétricos costumam ser empregados na avaliação de experimentos com inteligência computacional. Infelizmente tais testes são baseados em suposições, por exemplo, independência, normalidade e homocedasticidade, as quais nem sempre são válidas (DERAC *et al.*, 2011). Uma alternativa a esse problema é usar testes não-paramétricos como ferramenta de avaliação. Neste trabalho, a comparação entre os algoritmos estudados será realizada usando o teste não-paramétrico pareado de Wilcoxon, cujo

procedimento pode ser resumido em: para os n problemas usados na comparação calcule o valor d_i definido como a diferença entre as medidas de performance usadas para avaliar dois algoritmos em relação ao i -ésimo problema. Em seguida, ordene essas diferenças de acordo com seus valores absolutos e gere um *ranking*. Em caso de empate, recomenda-se usar o valor médio para os valores dos *ranks* envolvidos (DERAC *et al.*, 2011). Em seguida calcule os seguintes parâmetros:

$$R^+ = \sum_{d_i > 0} \text{rank}(d_i) + \frac{1}{2} \sum_{d_i = 0} \text{rank}(d_i) \quad (4.1)$$

e

$$R^- = \sum_{d_i < 0} \text{rank}(d_i) + \frac{1}{2} \sum_{d_i = 0} \text{rank}(d_i) \quad (4.2)$$

Seja $T = \min(R^+, R^-)$, se o valor de T é menor ou igual ao obtido na distribuição de Wilcoxon para n graus de liberdade e um certo nível de significância α (SHESKIN, 2004), a hipótese nula é rejeitada, o que significa que um algoritmo supera o outro.

Na Tabela 8 é apresentado os valores calculados para os parâmetros do teste de Wilcoxon, tomando por base os valores dos erros médios obtidos pelos três algoritmos aplicados às funções de teste:

Tabela 8 – Resultado do teste de Wilcoxon para problemas mono-objetivo

Dimensão	Par de Comparação	R+	R-
D = 5	DE/rand/1/bin + Imigrantes versus DE/rand/1/bin	186	237
	DE/rand/1/bin + Imigrantes versus CMAES	378	28
D = 10	DE/rand/1/bin + Imigrantes versus DE/rand/1/bin	372,5	33,5
	DE/rand/1/bin + Imigrantes versus CMAES	334	72
D = 30	DE/rand/1/bin + Imigrantes versus DE/rand/1/bin	379,5	26,5
	DE/rand/1/bin + Imigrantes versus CMAES	151,5	194,5
D = 50	DE/rand/1/bin + Imigrantes versus DE/rand/1/bin	393	13
	DE/rand/1/bin + Imigrantes versus CMAES	192,5	213,5

Para $n = 28$ e $\alpha = 0,01$, $T_{crítico} = 91$ (SHESKIN, 2004, Tabela A5), portanto, observando os dados da Tabela 8, o algoritmo proposto obteve melhor desempenho que a DE/rand/1/bin para $D = 10, 30$ e 50 e foi melhor que o CMA-ES para $D = 5$ e 10 .

Os resultados também mostram que para funções mono-modais, as primeiras 5 funções de teste, e $D = 5$, o modelo proposto têm desempenho equivalente DE/rand/1/bin e superior à CMA-ES. No entanto, a quantidade de avaliações necessárias para o algoritmo proposto alcançar o mínimo foi em média 40% inferior ao exigido pela DE/rand/1/bin, indicando que as alterações incorporadas aceleraram a otimização. Para as demais dimensões, o CMA-ES e a estratégia proposta foram bem superiores à DE canônica, contudo, o CMA-ES foi o que obteve o melhor desempenho com aumento das dimensões.

Para as demais funções, multi-modais e compostas, o desempenho do algoritmo DE/rand/1/bin é o melhor para $D = 5$, ficando em segundo lugar o algoritmo proposto, porém a performance do DE/rand/1/bin decai significativamente com o aumento das dimensões dos problemas. Para $D = 10$ o algoritmo proposto foi o que obteve erros médios menores entre todos os três. Finalmente, para $D > 10$ o CMA-ES e o proposto obtiveram desempenhos similares. Deve-se salientar, no entanto, que no algoritmo CMA-ES é realizada a decomposição da matriz de covariância em autovalores e autovetores, o que é uma tarefa normalmente custosa e exige satisfazer propriedades de condicionalidade e simetria das matrizes calculadas. Isto pode trazer problemas numéricos e inviabilizar a obtenção de uma solução válida. Conforme pode ser observado, na função 20, o algoritmo usado em algumas ocasiões não conseguiu encontrar uma solução numérica viável.

4.1.2. Testes para Problemas Multi-Objetivo

Um problema de otimização multi-objetivo sem restrições pode ser definido como uma busca por soluções $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_D]^T$ dentro de um espaço de decisão

$\Omega = [x_{min}, x_{max}]^D$, enquanto otimiza um vetor de funções objetivo, no caso de minimização, $Min([f_1(x) \ f_2(x) \ \dots \ f_k(x)]^T)$.

O concurso de otimização multi-objetivo sem restrições do CEC2009 (ZHANG *et al.*, 2008), consiste na obtenção do algoritmo que melhor aproxime a fronteira de Pareto para 10 problemas de minimização.

A métrica usada para avaliar os algoritmos foi o IGD dado pela equação:

$$IGD(A, P^*) = \frac{\sum_{v \in P^*} d(v, A)}{|P^*|} \quad (4.3)$$

onde A é o conjunto de pontos da fronteira estimada, P^* é a fronteira de Pareto real e $d(v, A)$ é a distância mínima Euclidiana entre $v \in P^*$ e pontos em A . Se a cardinalidade $|P^*|$ é grande o suficiente para representar P^* muito bem, o IGD pode medir tanto diversidade como a convergência de A . Para que o valor do IGD seja pequeno, o conjunto A deve ser próximo a P^* e não pode faltar qualquer parte de toda P^* . Nos experimentos foram usados 1000 pontos para P^* , e no conjunto A foram usados 100 pontos para problemas com dois objetivos e 150 pontos para problemas com três objetivos.

Os resultados obtidos estão dispostos na Tabela 8, para dimensão $D = 30$, considerando 20 repetições para cada problema, em negrito estão destacados os menores valores médios obtidos para o IGD. Como critério de parada foi adotado o número máximo de gerações 1000. Foram avaliados três algoritmos de otimização multi-objetivo. O primeiro foi o algoritmo NSDE descrito no Capítulo 2, com $K = 1$; o segundo é uma versão do NSDE com imigrantes aleatórios, usados para aumentar diversidade na população; e último algoritmo usa a estratégia multi-objetivo proposta nesse trabalho, descrita no Capítulo 3, que utiliza imigrantes guiados.

Tabela 9 - Resultados das simulações para D = 30.

Função	NSDE sem Imigrantes (K = 1)		NSDE com Imigrantes Aleatórios (K = 1)		Algoritmo Proposto	
	IGD (μ)	(σ)	IGD (μ)	(σ)	IGD (μ)	(σ)
UF1	0,366764	0,080575	0,294505	0,060505	0,102044	0,015446
UF2	0,067765	0,009403	0,064021	0,006516	0,048438	0,006265
UF3	0,386577	0,039247	0,403382	0,030321	0,240794	0,029034
UF4	0,16717	0,006211	0,182781	0,003288	0,149317	0,009455
UF5	1,967161	0,280764	3,257962	0,165556	0,988516	0,124944
UF6	1,852592	0,374601	1,374917	0,165139	0,375564	0,119669
UF7	0,399145	0,090534	0,348144	0,036214	0,258465	0,136166
UF8	0,337653	0,049869	0,255895	0,023202	0,196319	0,027824
UF9	0,442764	0,065406	0,393061	0,040795	0,285179	0,063043
UF10	2,537407	0,192213	2,990309	0,233232	0,58953	0,062097

Na Tabela 10 é apresentado os resultados do teste de Wilcoxon considerando os valores IGD obtidos pelos três algoritmos aplicados às funções de teste:

Tabela 10 - Resultado do teste de Wilcoxon para problemas multi-objetivo

Dimensão	Comparação	R+	R-
D = 30	Algoritmo proposto versus NSDE	55	0
	Algoritmo proposto versus NSDE + imigrantes aleatórios	55	0

Para $n = 10$ e $\alpha = 0,01$, tem-se $T_{crítico} = 3$ (SHESKIN, 2004, Tabela A5), portanto, na comparação direta o algoritmo proposto obteve melhor desempenho. Ou seja, os resultados sugerem que a inserção de imigrantes direcionados, melhorou a performance do NSDE nos problemas testados, inclusive nas funções UF5, UF6 e UF10 essa melhora foi considerável.

4.2 Sintonia de Parâmetros de um controlador MFAC-CFDL

Nesta seção serão apresentados resultados das estratégias para sintonia dos parâmetros de um controlador MFAC-CFDL usando AE proposto. Para tanto, foram escolhidos sete casos de estudo diferentes. O primeiro caso trata-se de uma combinação de dois sub-sistemas não-lineares em série resultando num sistema com estrutura, parâmetros e ordem variante no tempo. Já o segundo, trata-se também de dois sub-sistemas não-lineares em série, no entanto, ambos sub-sistemas são de fase não-mínima. Um sistema é denominado de fase não-mínima se uma realimentação não-linear de estado pode manter a saída do sistema igual a zero, enquanto sua dinâmica interna torna-se instável (KRSTLC *et al.*, 1995).

Para o terceiro caso de estudo, diferente dos dois primeiros, foi escolhido um sistema linear, no entanto, sua estrutura é variável no tempo o que aumenta bastante complexidade do seu tratamento. No quarto caso novamente é analisado um sistema não-linear de estrutura e ordem variante no tempo, mas a função de rastreamento definida no problema é bastante simples, nesse caso, é analisado se isso poderia limitar as vantagens obtidas com otimização.

Já para os casos cinco e seis também foram usados sistemas não-lineares, no entanto, nos problemas estudados o desempenho das estratégias propostas é comparado ao de outras versões de controle baseado no MFAC. Finalmente, no último caso é analisado modelo linear de estrutura fixa, obtido de uma linearização de sistema de servo mecanismo, que usa como função de rastreamento uma função não-linear. Portanto, o MFAC e as estratégias propostas também são adequadas ao controle de sistemas que normalmente seriam tratados por abordagens de controle mais clássicas.

Para todos os casos, inicialmente, os resultados foram obtidos usando parâmetros sintonizados *off-line*, usando o algoritmo de otimização proposto, e parâmetros encontrados na literatura. Em seguida, para os mesmos sistemas, também foi testada estratégia de otimização *on-line* dos parâmetros do controlador.

Os resultados finais permitem a comparação entre o desempenho das duas estratégias propostas.

4.2.1. Casos de Estudo para Sintonia *Off-Line*

O ajuste dos parâmetros do controlador *off-line* para todos os casos de estudo segue o esquema padrão de um problema de otimização multi-objetivo. Ou seja, busca-se encontrar o conjunto de parâmetros reais ótimos, a saber $[\rho \ \eta \ \mu \ \lambda \ \phi(1)]^T$, para um controlador MFAC-CFDL, capaz de encontrar o melhor compromisso entre os seguintes índices de desempenho.

Erro médio quadrático:

$$J_{MSE}(\rho, \lambda, \eta, \mu, \phi(1)) = \frac{1}{N} \sum_{k=1}^N (y_d(k) - y(k))^2 \quad (4.4)$$

sendo, $y_d(k)$, o sinal de saída desejado e $y(k)$ o sinal de saída da planta. O valor do J_{ISE} está associado ao erro em estado estacionário para a resposta do sistema.

Erro médio quadrático da derivada do sinal:

$$J_{MDSE}(\rho, \lambda, \eta, \mu, \phi(1)) = \frac{1}{N-1} \sum_{k=1}^{N-1} [y_d'(k) - y'(k)]^2 \quad (4.5)$$

onde $y_d'(k)$ e $y'(k)$ são as derivadas do sinal desejado e do sinal de saída da planta, respectivamente. Esse índice está relacionado aos valores máximos alcançados para o sobressinal apresentado pela resposta do sistema nos instantes onde o sinal de referência muda bruscamente.

Na simulação do processo de evolução proposto, cada indivíduo da população, formado por um vetor de números reais, representa o conjunto de parâmetros a serem usados no controlador. Portanto, as funções de desempenho são calculadas para os membros da população, simulando a operação do sistema controlado, usando como parâmetros para o controlador os valores dentro do vetor do indivíduo.

Em todos os casos de estudo optou-se por usar os seguintes parâmetros internos para o algoritmo proposto:

- Fator de ponderação da mutação: $F = 0,50$;
- Taxa de cruzamento: $Cr = 0,80$;
- Número de elementos da população: $NP = 50$;
- Número de agrupamentos: 10;
- Número máximo de gerações: $G_{max} = 1500$;
- Número mínimo de gerações: $G_{min} = 1000$.

Tais parâmetros foram escolhidos com base em valores de referência na literatura (PRICE *et al.*, 2005) ou mediante testes realizados na fase de experimentação do trabalho:

Em todos os casos, optou-se por dois critérios de parada. O primeiro, corresponde ao número máximo de gerações G_{max} . Já o segundo, está relacionado à variação do valor do índice de hipervolume das funções normalizadas (GARROZI, 2012). Portanto, quando o valor do hipervolume normalizado se mantém por sucessivas gerações (dez gerações sucessivas nesta Tese), o algoritmo é encerrado. Para evitar que o algoritmo se encerre prematuramente, adotou-se também uma quantidade mínima de gerações que o algoritmo deve ser executado.

A partir da fronteira de Pareto, a escolha da solução adotada, pelo tomador de decisões, foi baseada no critério da menor distância euclidiana entre cada solução na fronteira de Pareto e o ponto utópico (DEB, 2001), que anula as duas funções objetivo normalizadas, ou seja:

$$x_{chosen} = \min_{1 \leq i \leq NP} (d(x_i)) \mid d_i = \sqrt{(f_{N1}(x))^2 + (f_{N2}(x))^2} \quad (4.6)$$

onde x_{chosen} é a solução adotada, e $f_{N1}(x)$ e $f_{N2}(x)$ são as funções objetivo normalizadas.

Para cada caso de estudo foram realizados dez experimentos de otimização. Como resultados iniciais, os valores médios e os desvios padrões obtidos serão apresentados para cada critério. Além disso, também serão mostradas comparações entre os valores médios calculados para os critérios, os melhores valores obtidos

nos experimentos e valores para as funções de desempenho calculadas, usando parâmetros encontrados na literatura.

a) *Primeiro Caso de Estudo*

Os resultados a seguir foram obtidos mediante simulação do seguinte sistema não-linear (HOU; JIN, 2014)

$$y(k+1) = \begin{cases} \frac{y(k)}{1+y^2(k)} + u^3(k) & k \leq 500 \\ \frac{y(k)y(k-1)y(k-2)(y(k-2)-1)+a(k)u(k)}{1+y^2(k-1)+y^2(k-2)} & k > 500 \end{cases} \quad (4.7)$$

observe na Eq. (4.7), que o termo $a(k) = \text{round}(k/500)$, introduz uma perturbação adicional para o sistema.

O controlador MFAC-CFDL foi projetado para fazer a planta seguir o sinal de referência, Eq. (4.8):

$$y_d(k+1) = \begin{cases} 0.5(-1)^{\text{round}(k/100)} & k \leq 300 \\ 0.5 \sin(k\pi/100) + 0.3 \cos(k\pi/50) & 300 < k \leq 700 \\ 0.5(-1)^{\text{round}(k/100)} & k > 700 \end{cases} \quad (4.8)$$

Na simulação do sistema foram usadas as condições iniciais na simulação, $u(1) = u(2) = 0, y(1) = -1, y(2) = 1, \hat{\phi}(1) = 2$ and $\varepsilon = 10^{-5}$.

Os resultados das dez simulações do problema de otimização estão apresentados na Tabela 11. Destacado, em **negrito**, o melhor conjunto de parâmetros, escolhido usando como critério a distância ao ponto utópico.

Tabela 11 – Resultados das otimizações *off-line* para o primeiro caso de estudo.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,9909	1,9914	0,3451	0,1710	0,7727	7,83E-03	8,50E-03
2	0,9918	1,9926	0,2716	0,1585	0,7451	7,32E-03	9,17E-03
3	0,9902	1,9143	0,2261	0,1921	0,7145	7,75E-03	8,89E-03
4	0,9886	1,9763	0,2858	0,1640	0,7503	7,49E-03	9,04E-03
5	0,9930	1,9775	0,2554	0,1730	0,7292	7,41E-03	9,02E-03
6	0,9955	1,9768	0,3606	0,1310	0,8240	7,54E-03	8,54E-03
7	0,9665	1,6054	0,1073	0,2867	0,6684	1,11E-02	9,33E-03
8	0,9950	1,9936	0,2740	0,1574	0,7492	7,31E-03	9,17E-03
9	0,9855	1,9764	0,3492	0,1470	0,7986	7,73E-03	8,52E-03
10	0,9858	1,9954	0,2822	0,1507	0,7525	7,37E-03	9,11E-03
Média $\pm$ Desvio						7,88E-03 $\pm$ 1,14E-03	8,93E-03 $\pm$ 3,02E-04
Média Gerações						1061,6	
Média de Avaliações de Função (FES)						60416	

Na Tabela 12 é possível comparar os resultados dos experimentos, com os valores obtidos mediante simulações usando parâmetros para o controlador encontrados da literatura, parâmetros 3 e 4 (HOU, JIN, 2014) e outros otimizados usando uma versão do algoritmo NSDE, parâmetros 6 a 7 (SOUSA *et al.*, 2014).

Tabela 12 - Tabela com o conjunto de parâmetros para o controlador

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média+Std)					7,88E-03 $\pm$ 1,14E-03	8,93E-03 $\pm$ 3,02E-04
2	Resultado do Melhor do Experimento (Sexto)					7,54E-03	8,54E-03
3	0,6000	1,0000	1,0000	0,1000	2	1,69E-02	1,21E-02
4	0,6000	1,0000	1,0000	2,0000	2	2,65E-02	1,17E-02
5	0,9913	1,5372	0,1733	2,1536	2	1,95E-02	1,14E-02
6	0,9999	0,7628	1,0765	0,0073	2	1,04E-02	1,30E-02
7	0,9568	1,5390	0,0333	4,2184	2	3,83E-02	1,14E-02

A Tabela 12 mostra que o melhor resultado encontrado, obtido usando os parâmetros do Experimento 6 (Tabela 11), foi superior aos resultados das simulações usando parâmetros da literatura e os parâmetros otimizados usando a versão do algoritmo NSDE. Além disso, a média para os índices de desempenho,

nos dez experimentos, também foi melhor que os resultados obtidos com os parâmetros definidos na literatura.

A Figura 9 apresenta a fronteira de Pareto obtida pelo Experimento 6 (Tabela 11). Destacado, em preto, o melhor conjunto de parâmetros encontrado; em vermelho, o resultado do melhor conjunto de parâmetros da literatura; e, em verde, os valores médios obtidos com os dez experimentos.

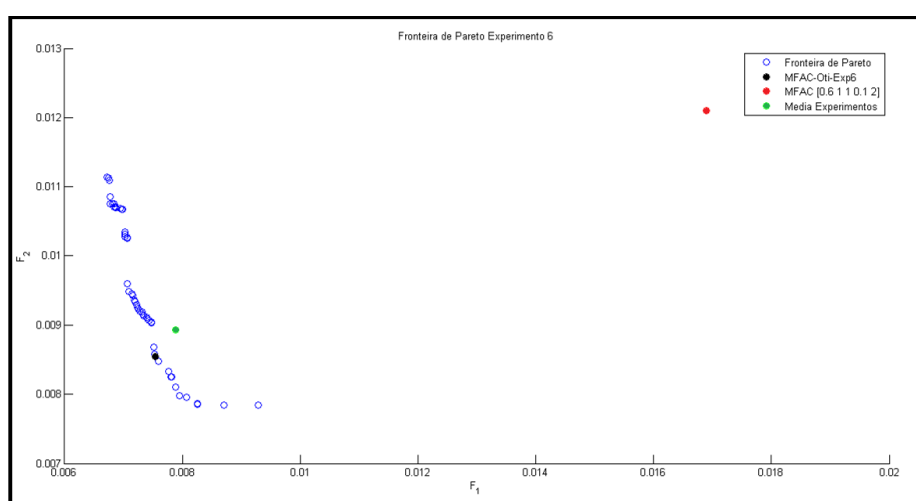
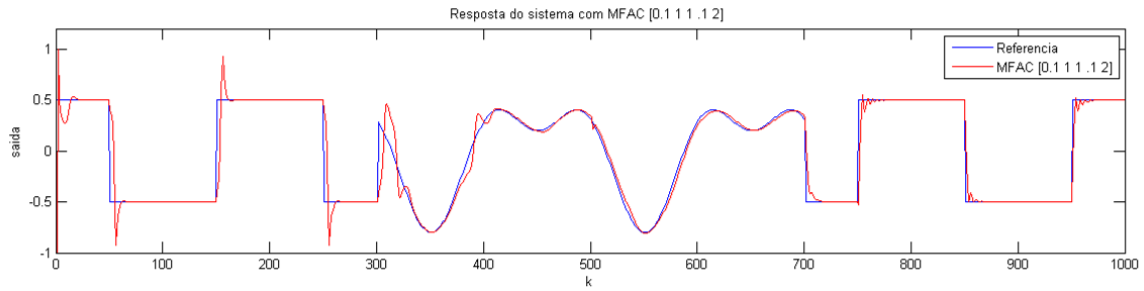


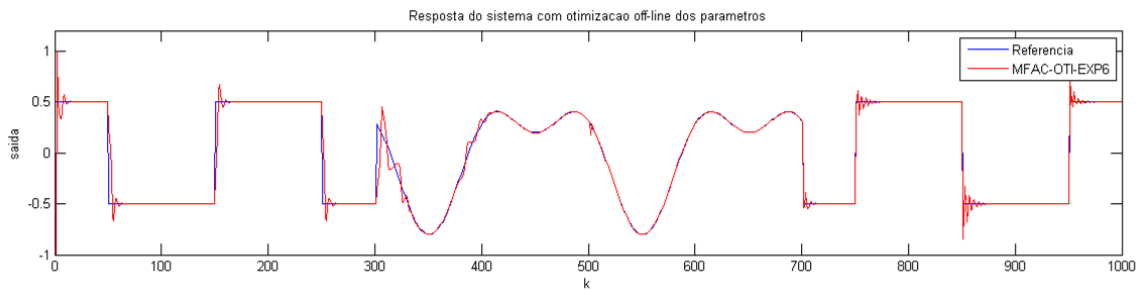
Figura 9 – Fronteira de Pareto e parâmetros em destaque.

Observe na Figura 9 que a melhor solução encontrada na literatura está localizada na região de soluções dominadas.

A Figura 10 a-b apresenta a simulação do sistema não-linear usando dois conjuntos de parâmetros distintos. O primeiro deles (MFAC [0,6 1 1 0,1 2]^T) é o que obteve os melhores resultados encontrados na literatura. Já o segundo (MFAC-OTI-EXP6) foi definido pela otimização.



a) Resposta do sistema com MFAC - CFDL [6 1 1 .1 2]^T



b) Resposta do sistema com MFAC - CFDL - OTI - EXP6

Figura 10 - Simulação do sistema usando os parâmetros em destaque.

No gráfico observa-se que a saída do sistema para o controlador usando o conjunto de parâmetros otimizados, em vermelho, é mais próxima ao sinal de referência, em azul, durante praticamente toda a simulação.

b) Segundo Caso de Estudo

O segundo caso de estudo foi apresentado por Hou e Jin (2014) e está descrito na Eq. (4.9):

$$y(k+1) = \begin{cases} \frac{5y(k)y(k-1)}{1+y^2(k)+y^2(k-1)+y^2(k-2)} + u(k) + 1,1u(k-1), & k \leq 500 \\ \frac{2,5y(k)y(k-1)}{1+y^2(k)+y^2(k-1)} + 1,2u(k) + 1,4u(k-1) + b(k) & k > 500 \end{cases} \quad (4.9)$$

sendo, $b(k) = 0,7\sin((y(k) + y(k-1))/2)\cos((y(k) + y(k-1))/2)$.

O sinal de referência é definido pela Eq. (4.10):

$$y^*(k+1) = \begin{cases} 5\text{sen}\left(\frac{k\pi}{50}\right) + 2\cos\left(\frac{k\pi}{100}\right), & x \leq 300 \\ 5(-1)^{\text{round}(k/100)}, & 300 < x \leq 700 \\ 5\text{sen}\left(\frac{k\pi}{50}\right) + 2\cos\left(\frac{k\pi}{100}\right), & x > 700 \end{cases} \quad (4.10)$$

e para todas as simulações foram usadas seguintes condições iniciais: $u(1) = u(2) = 0, y(1) = -1, y(2) = 1, \hat{\phi}(1) = 2$ and $\varepsilon = 10^{-5}$.

Os resultados das dez simulações do problema de otimização estão apresentados na Tabela 13. Os melhores resultados estão em negritos, no experimento 8 para determinação dos parâmetros ótimos.

Tabela 13 – Resultados das otimizações *off-line* para o segundo caso de estudo.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,9423	1,9651	0,9860	2,6573	2,2938	4,72E-01	4,89E-01
2	0,4737	1,7661	4,4027	1,8847	1,1104	6,35E-01	4,86E-01
3	0,3918	0,9262	2,5928	0,7713	1,3973	5,64E-01	4,96E-01
4	0,8370	1,9979	0,9307	2,0832	2,1973	4,48E-01	4,67E-01
5	0,7474	1,8878	0,4873	1,5188	1,6685	4,02E-01	4,94E-01
6	0,8301	1,9888	0,8982	2,0097	1,7099	4,10E-01	5,10E-01
7	0,8563	1,8511	0,0465	1,7460	1,9611	4,12E-01	4,73E-01
8	0,7331	1,9968	0,3977	1,3789	1,6561	3,95E-01	4,80E-01
9	0,7577	1,9921	0,3683	1,5087	1,9243	4,30E-01	4,47E-01
10	0,4480	0,8087	2,1967	1,5208	1,4014	6,08E-01	4,89E-01
Média ± Desvio						4,78E-01±9,06E-02	4,83E-01±1,75E-02
Média Gerações						1049	
Média de Avaliações de Função (FES)						60420	

Na Tabela 14 seguir é possível comparar os melhores resultados e os valores médios, com os resultados obtidos usando parâmetros encontrados da literatura.

Tabela 14 - Tabela com o conjunto de parâmetros para o controlador.

	Parâmetros					Funções Objetivo	
N	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média+Std)					4,78E-01±9,06E-02	4,83E-01±1,75E-02
2	Resultado do Melhor do Experimento (Oitavo)					3,95E-01	4,80E-01
3	0,6000	1,0000	1,0000	0,1000	2	6,88E-01	9,86E-01
4	0,6000	1,0000	1,0000	2,0000	2	6,08E-01	6,12E-01

Comparando os resultados, observa-se, assim como no primeiro caso, que tanto o melhor experimento como os resultados médios, foram melhores que os valores obtidos para os índices calculados com o controlador usando os parâmetros encontrados na literatura.

A Figura 11 apresenta a fronteira de Pareto obtida do experimento 8 (Tabela 11), avaliado como o que obteve os melhores resultados.

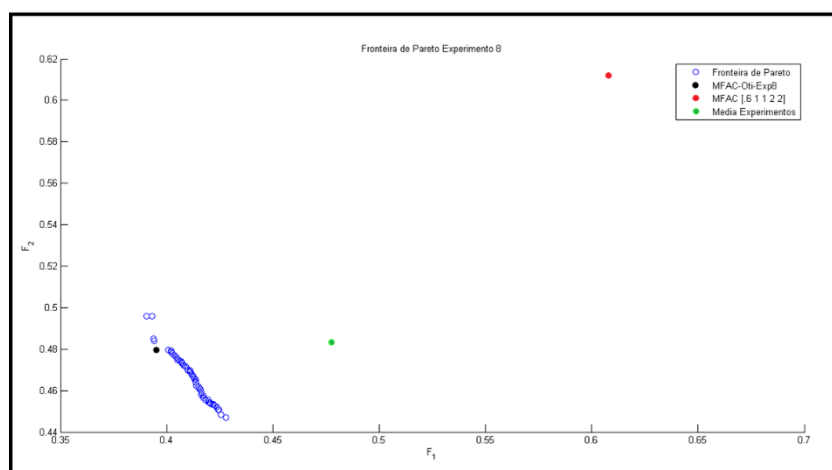


Figura 11 – Fronteira de Pareto e parâmetros em destaque.

Observe na Figura 11 que a melhor solução empregada na literatura se encontra na região de soluções dominadas.

A Figura 12 a-b mostra a saída do sistema usando dois conjuntos de parâmetros distintos. O primeiro, foi definido na literatura (MFAC [0,6 1 1 2 2]^T) e o segundo o melhor dos experimentos de otimização (MFAC-OTI-EXP8).

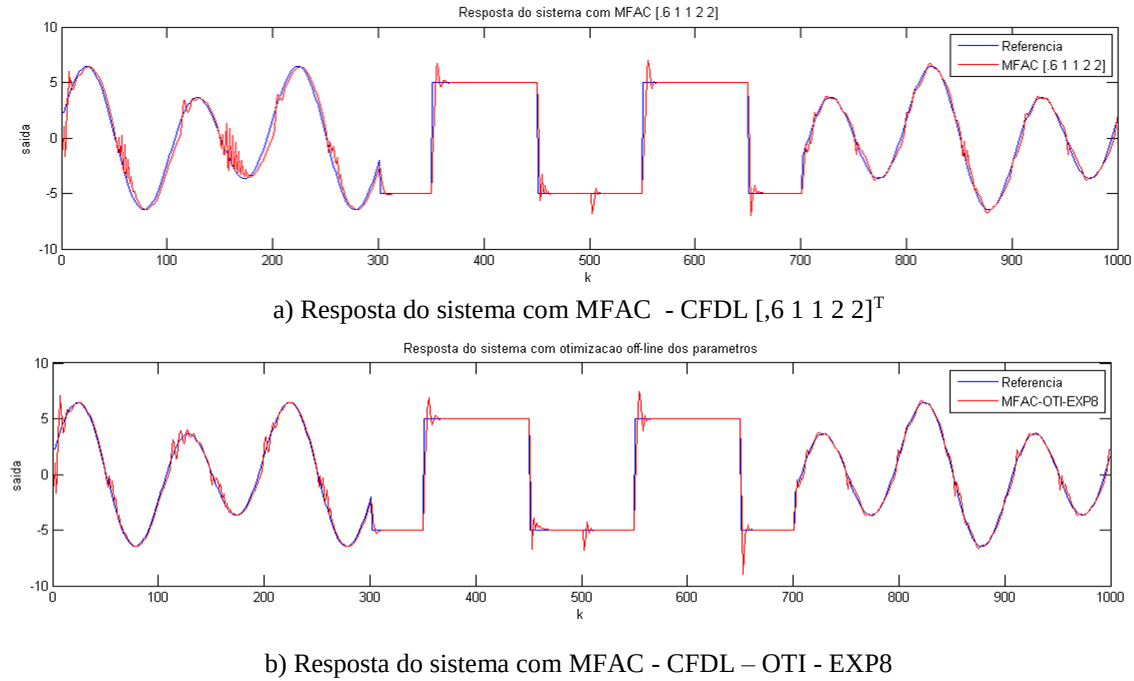


Figura 12 - Simulação do sistema usando os parâmetros em destaque.

O gráfico mostra visualmente que, mais uma vez, a estratégia a otimização *off-line* produziu melhores resultados, no rastreo do sinal de referência na simulação do sistema controlado.

c) Terceiro Caso de Estudo

Sistema apresentado por Hou e Jin (2014) e descrito na Eq. (4.11):

$$y(k+1) = 1,5y(k) - 0,7y(k-1) + 0,1 \times \begin{cases} u(k) + b(k)u(k-1), & 1 \leq k < 200 \\ u(k) + b(k-2)u(k-3), & 200 \leq k < 400 \\ u(k) + b(k-4)u(k-5), & 400 \leq k < 600 \\ u(k) + b(k-6)u(k-7), & 600 \leq k < 800 \\ u(k) + b(k-8)u(k-9), & k \geq 800 \end{cases} \quad (4.11)$$

onde $b(k) = 0,1 + 0,1\text{round}(k/100)$.

A Eq. (4.12) representa o sinal de referência adotado para o problema:

$$y^*(k+1) = 0,5 + 0,5(-1)^{\text{round}(k/200)} \quad (4.12)$$

Os resultados dos dez experimentos de otimização estão na Tabela 15:

Tabela 15 – Resultados das otimizações *off-line* para o terceiro caso de estudo.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,5491	0,1626	0,0058	4,7276	0,4378	1,12E-01	1,31E-02
2	0,3800	1,6194	0,1343	2,1894	0,2524	1,05E-01	1,31E-02
3	0,9412	1,2551	0,1004	5,6486	0,1744	1,15E-01	1,31E-02
4	0,5764	0,6897	0,0539	3,5093	0,1834	1,15E-01	1,31E-02
5	0,5951	1,4875	0,0998	4,2381	0,2391	1,13E-01	1,31E-02
6	0,5999	1,8083	0,1170	4,4463	0,2947	1,07E-01	1,31E-02
7	0,8026	1,6676	0,1137	5,6471	0,2539	1,09E-01	1,31E-02
8	0,5853	0,5936	0,0513	3,2132	0,1798	1,10E-01	1,31E-02
9	0,5067	1,0258	0,0453	5,4691	0,2814	1,23E-01	1,30E-02
10	0,8457	1,3533	0,0895	6,1175	0,2622	1,09E-01	1,31E-02
Média $\pm$ Desvio						1,12E-01 $\pm$ 5,19E-03	1,31E-02 $\pm$ 2,27E-05
Média Gerações						1059,60	
Média de Avaliações de Função (FES)						53080	

Na Tabela 16 é possível comparar os resultados dos experimentos com os valores obtidos usando parâmetros encontrados da literatura.

Tabela 16 - Tabela com o conjunto de parâmetros para o controlador.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média $\pm$ Std)					1,12E-01 $\pm$ 5,19E-03	1,31E-02 $\pm$ 2,27E-05
2	Resultado do Melhor do Experimento (Segundo)					1,05E-01	1,31E-02
3	0,6000	1,0000	1,0000	2,0000	2	1,18E-01	1,7054E-02
4	0,6000	1,0000	1,0000	15,0000	2	1,01E-01	1,35E-02

Comparando os resultados obtidos nas simulações. Verificou-se que o segundo conjunto de parâmetros da literatura ([0,6 1 1 15 2]^T), obteve um resultado melhor, no primeiro índice de desempenho. Entretanto, deve-se ressaltar que como trata-se de um problema de otimização multi-objetivo, as soluções precisam ser avaliadas segundo critério de não-dominância, em relação a fronteira de Pareto encontrada. Na Figura 13, a fronteira de Pareto do Experimento 2 (Tabela 15) é

traçada e o ponto correspondente ao segundo conjunto de parâmetros pode ser melhor avaliado

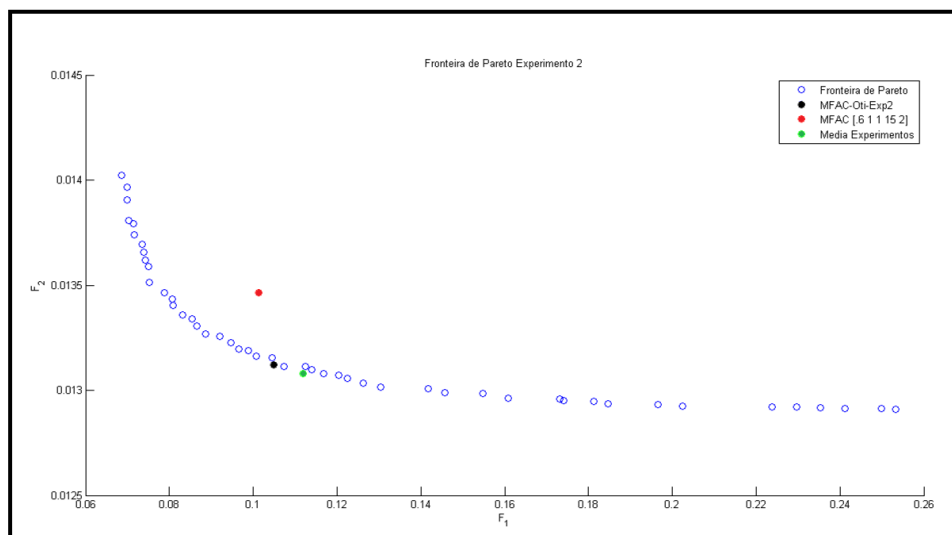
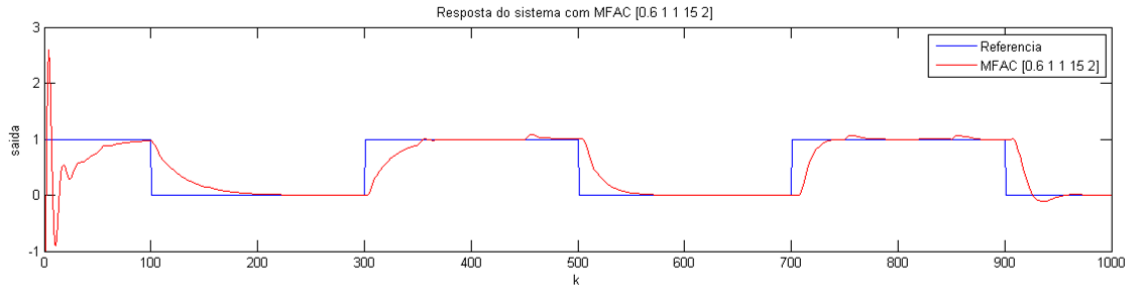


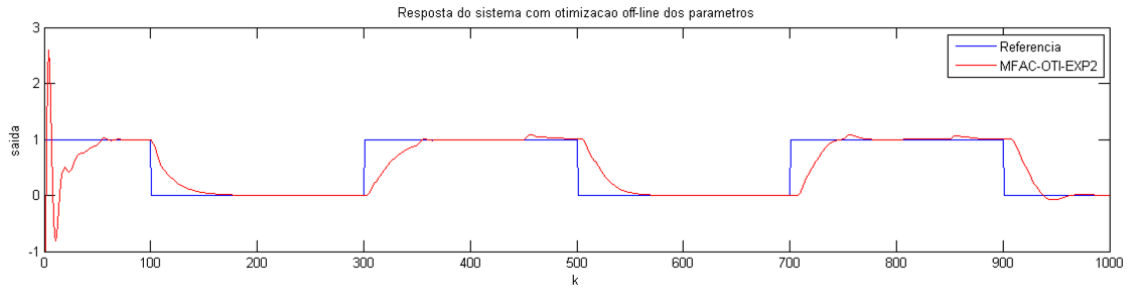
Figura 13 – Fronteira de Pareto e parâmetros em destaque

Na Figura 13, observe que o ponto em vermelho correspondente aos parâmetros da literatura é dominado pela fronteira de Pareto obtida. Além disso, considerando o critério adotado pelo tomador de decisões, distância ao ponto utópico, os parâmetros escolhidos no Experimento 2 (Tabela 13) apresentam melhor desempenho.

A seguir, na Figura 14 a-b, são apresentadas as simulações do sistema com o controlador usando os parâmetros encontrados na literatura e os parâmetros do Experimento 2, respectivamente.



a) Resposta do sistema com MFAC - CFDL $[.6 \ 1 \ 1 \ 15 \ 2]^T$



b) Resposta do sistema com MFAC - CFDL – OTI - EXP2

Figura 14 - Simulação do sistema usando os parâmetros em destaque.

A análise da resposta do sistema controlado mostra que a estratégia adotada (MFAC-OTI-EXP2) novamente obteve no geral desempenho melhor os parâmetros encontrados na literatura, quando são levados em conta o compromisso com os dois objetivos propostos.

d) Quarto Caso de Estudo

Equação do sistema (PANG *et al.*, 2014):

$$y(k+1) = \begin{cases} \frac{y(k)}{1+y(k)^2} + u(k)^3, & 1 \leq x < 100 \\ \frac{y(k)y(k-1)y(k-2)u(k-1)(y(k-2)-1)+1,5u(k)}{1+y(k-1)^2+y(k-2)^2}, & 100 \leq x \leq 200 \end{cases}$$

(4.13)

e sinal de referência:

$$y^*(k+1) = \begin{cases} 1, & 1 \leq k < 100 \\ -1, & 100 \leq k \leq 200 \end{cases} \quad (4.14)$$

Os resultados dos dez experimentos do problema de otimização estão elencados na Tabela 17:

Tabela 17 – Resultados das otimizações *off-line* para o quarto caso de estudo

	Parâmetros					Funções Objetivo	
N	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,7657	1,4198	0,0001	0,7358	2,2283	4,49E-02	1,52E-02
2	0,9875	1,9718	0,0104	0,2457	1,1179	4,82E-02	1,49E-02
3	0,9021	1,6825	0,0014	1,6990	1,9027	4,78E-02	1,46E-02
4	0,9785	1,7405	0,0004	0,5810	0,9246	3,23E-02	1,63E-02
5	0,9751	1,9259	0,0279	3,7490	2,0251	7,24E-02	1,34E-02
6	0,9646	1,8739	0,0191	2,8777	1,8902	6,15E-02	1,36E-02
7	0,9866	1,7856	0,0172	2,9911	2,0010	6,08E-02	1,36E-02
8	0,9867	1,9985	0,0261	3,3445	1,8368	7,45E-02	1,33E-02
9	0,9141	1,9269	0,0201	2,5214	1,8434	5,71E-02	1,39E-02
10	0,9635	1,8758	0,0157	2,7038	1,8381	5,78E-02	1,38E-02
Média ± Desvio						5,37E-02±1,54E-02	1,43E-02±9,74E-04
Média Gerações						1150,40	
Média de Avaliações de Função (FES)						57620	

Na Tabela 18 é possível comparar o melhor resultado obtido nos experimentos, com a média de todos resultados e com parâmetros encontrados da literatura.

Tabela 18 - Tabela com o conjunto de parâmetros para o controlador.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média±Std)					5,37E-02±1,54E-02	1,43E-02±9,74E-04
2	Resultado do Melhor do Experimento (Quarto)					3,23E-02	1,63E-02
3	1	1	1	2	1.6	3,79E-02	1,95E-02

Comparando os resultados obtidos, observa-se que os parâmetros do Experimento 4 (Tabela 17) tiveram desempenho superior aos do encontrado na literatura. Na Figura 15 a fronteira de Pareto do Experimento 4 é apresentada. Em destaque estão os pontos correspondentes aos parâmetros da Tabela 18.

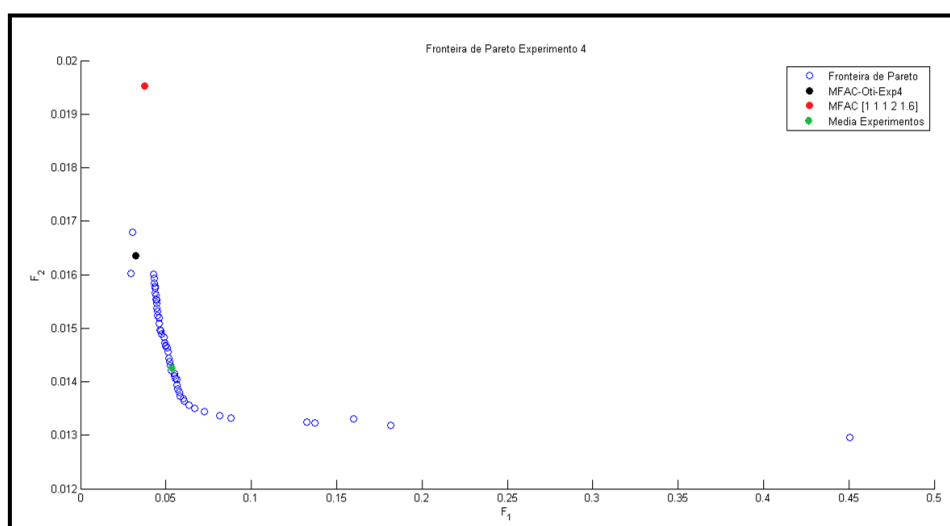
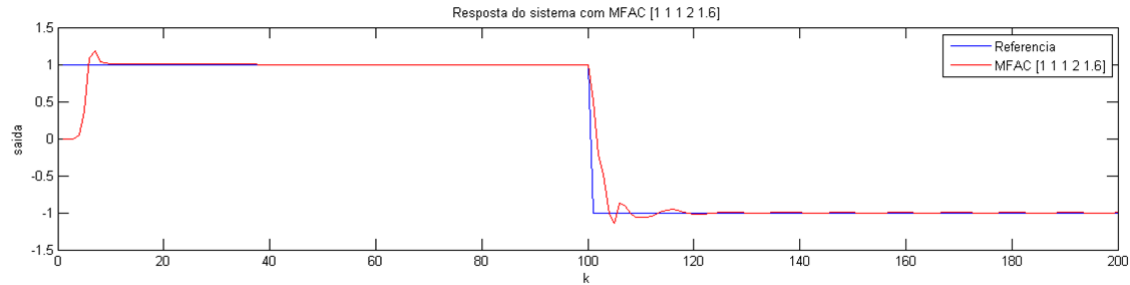
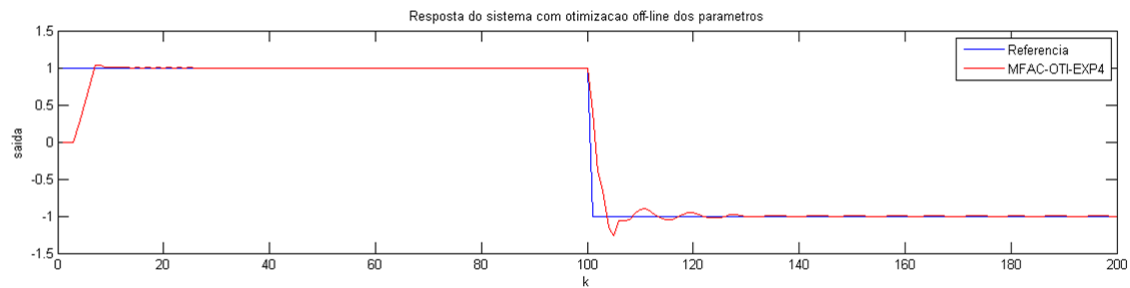


Figura 15 – Fronteira de Pareto e parâmetros em destaque.

Na Figura 16 a-b é apresentado a saída do sistema usando os parâmetros da literatura e os obtidos no Experimento 4, respectivamente.



a) Resposta do sistema com MFAC - CFDL [1 1 1 2 1,6]^T



b) Resposta do sistema com MFAC - CFDL – OTI - EXP4

Figura 16 - Simulação do sistema usando os parâmetros em destaque

Nos gráficos da Figura 16, observa-se que, pela resposta do sistema MFAC-OTI-EXP4, o controlador ótimo apresenta redução no sobressinal e menor erro de estado permanente do sistema na direção do sinal de referência. Portanto, mesmo com uma função de rastreamento bem simples, a estratégia de otimização ainda assim obteve vantagens.

e) Quinto Caso de Estudo

Equação (4.15) descreve o comportamento do sistema (HOU; JIN, 2014):

$$y(k+1) = \begin{cases} \theta_1 + \theta_2, & k \leq 200 \\ \theta_4 + \theta_5, & k > 200 \end{cases} \quad (4.15)$$

onde:

$$\theta_1 = 2,5y(k)y(k-1)/(1+y(k)^2+y(k-1)^2) \quad (4.16)$$

$$\theta_2 = 0,7\text{sen}\left(0,5(y(k)+y(k-1))\right) + 1,4u(k-1) + 1,2u(k) \quad (4.17)$$

$$\theta_3 = -0,1y(k) - 0,2y(k-1) - 0,3y(k-2) \quad (4.18)$$

$$\theta_4 = 0,1u(k) + 0,02u(k-1) + 0,03u(k-2) \quad (4.19)$$

e o sinal de referência é dado por:

$$y^*(k+1) = 5(-1)^{\text{round}(k/80)} \quad (4.20)$$

Na Tabela 19, os resultados das dez simulações do problema de otimização estão apresentados:

Tabela 19 – Resultados das otimizações *off-line* para o quinto caso de estudo.

N	Parâmetros					Funções Objetivo	
	ρ	η	ρ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,3361	1,2776	2,2399	3,5638E-04	0,7556	1,5712	1,0396
2	0,2639	1,6013	2,7880	1,3526E-04	0,5941	1,6298	0,9390
3	0,2570	1,7043	2,6749	2,9542E-04	0,5810	1,6587	0,9252
4	0,9268	0,6802	15,0788	1,3785E-03	2,5539	1,1545	1,0522
5	0,2506	1,7473	2,8192	2,5769E-04	0,5533	1,6803	0,9260
6	0,2637	1,7901	2,5861	1,6851E-04	0,6319	1,5521	0,9629
7	0,2542	1,7536	2,8398	2,6548E-04	0,5556	1,6736	0,9217
8	0,2537	1,7568	1,9667	3,2997E-04	0,6063	1,6417	0,9287
9	0,2769	1,5380	2,3558	6,4725E-04	0,6544	1,6311	0,9549
10	0,2521	1,6526	2,3894	1,6091E-04	0,5706	1,7030	0,9202
Média ± Desvio						1,5896±0,1599	0,9571±0,0490
Média Gerações						1500,00	
Média de Avaliações de Função (FES)						55655,00	

A seguir, na Tabela 20, é possível comparar o resultado dos valores médios para os índices de desempenho, com os valores obtidos pelo melhor Experimento (Sexto) e também com a planta sendo controlada usando um controlador do tipo MFAC-PFDL. Segundo Hou e Jin (2014) a versão do controlador MFAC-PFDL é

mais complexa e mais flexível, pois considera um maior número de parâmetros, e deveria ser usada no tratamento de sistemas cujo grau de complexidade não seria adequado ao uso de um controlador MFAC-CFDL.

Tabela 20 - Tabela com o conjunto de parâmetros para o controlador.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média $\pm$ Std)					1,5896 $\pm$ 0,1599	0,9571 $\pm$ 0,0490
2	Resultado do Melhor do Experimento (Sexto)					1,5521	0,9629
3	MFAC-PFDL					2,0816	1,0529

Observe que os valores obtidos, mediante otimização dos parâmetros do controlador obteve resultados melhores que os com o controlador MFAC-PFDL, a apesar do grau de não-linearidade do sistema, demonstrando que a otimização dos parâmetros pode ser melhor do que usar um controlador MFAC mais complexo.

Na Figura 17 é mostrada a fronteira de Pareto para o Experimento 6 (Tabela 17), a qual apresenta um melhor compromisso entre a minimização dos dois índices de desempenho.

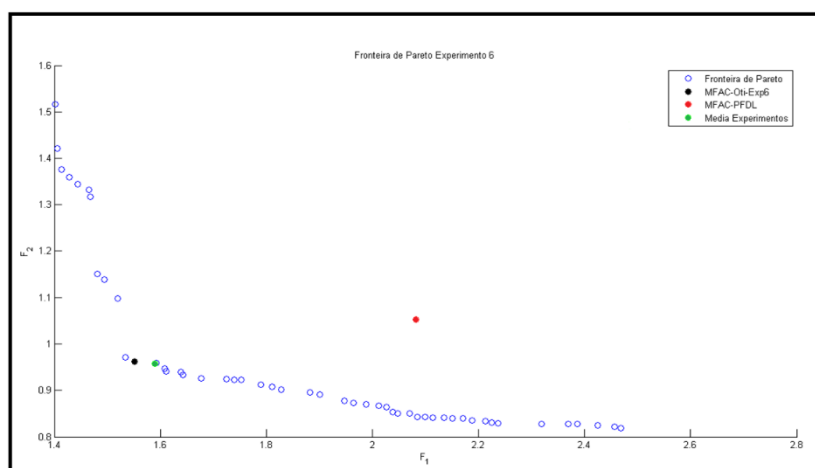


Figura 17 – Fronteira de Pareto e parâmetros em destaque.

Observe na Figura 17 que o resultado obtido usando o MFAC-PFDL encontra-se na região dominada da fronteira de Pareto do Experimento 6.

A Figura 18 a-b mostra as saídas do sistema com o controlador com o controlador MFAC-PFDL, usando parâmetros encontrados na literatura e o MFAC-CFDL, com os parâmetros do Experimento 6.

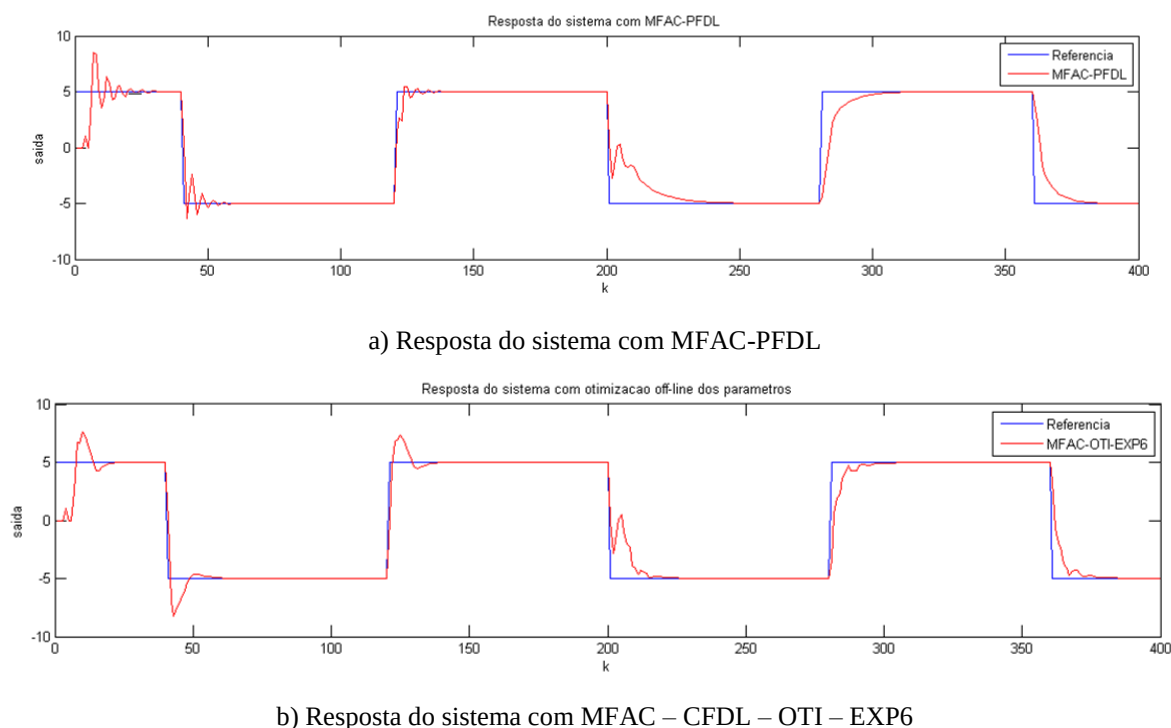


Figura 18 - Simulação do sistema usando os parâmetros em destaque.

A simulações na Figura 18 mostram que, o MFAC-CFDL com parâmetros otimizados, oscila menos nas mudanças bruscas da função de referência e o erro em estado estacionário também é menor. Pode-se notar também que a convergência para o caso MFAC-OTI-EXP6 é mais rápido, em geral.

f) Sexto Caso de Estudo

Equação do sistema proposto por Leng *et al.* (2014):

$$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)+u(k)}{1+y(k-1)^2+0,15y(k-1)^2} \quad (4.21)$$

e sinal de referência:

$$y^*(k+1) = 0,5 + 0,25 \cos(0,01k\pi) + 0,25 \sin(0,02k\pi) \quad (4.22)$$

Na Tabela 21, estão elencados os resultados das dez simulações do problema de otimização:

Tabela 21 – Resultados das otimizações *off-line* para o sexto caso de estudo.

<i>N</i>	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,9541	1,7887	9,0289E-04	0,1560	0,9848	4,1772E-03	7,3504E-04
2	0,9435	1,8910	2,5749E-03	0,1754	0,9375	4,1596E-03	7,4072E-04
3	0,9849	1,8408	8,3908E-04	0,1659	0,9743	4,1415E-03	7,4382E-04
4	0,9952	1,9554	3,7932E-03	0,2646	0,8970	4,1663E-03	7,4213E-04
5	0,9870	1,8457	1,9503E-03	0,2587	0,9319	4,1873E-03	7,3438E-04
6	0,9766	1,7573	8,3586E-04	0,1779	1,0010	4,1680E-03	7,3697E-04
7	0,9097	1,8069	1,7802E-03	0,2269	0,9184	4,2232E-03	7,3095E-04
8	0,9927	1,9599	1,6990E-03	0,3391	0,8482	4,2149E-03	7,3161E-04
9	0,9622	1,8301	1,9654E-03	0,2262	0,9414	4,1880E-03	7,3453E-04
10	0,9821	1,9391	8,8276E-04	0,2629	0,8882	4,1932E-03	7,3388E-04
Média ± Desvio						4,1819E-03±2,4889E-05	7,3641E-04±4,4144E-06
Média Gerações						1330,70	
Média de Avaliações de Função (FES)						66635,00	

Na Tabela 22, a seguir, é possível comparar o resultado dos melhores valores e os médios, para os índices de desempenho, em relação aos calculados, controlando o sistema usando MFAC-CFDL, usando parâmetros $\rho = 1$, $\lambda = 2,5$, $\eta = 1$, $\mu = 1$ $\phi_1(1) = 1$ e também uma estratégia de controle denominada MFAC-CC (Model Free Adaptive Control with Contractive Constraints).

Tabela 22 - Tabela com o conjunto de parâmetros para o controlador.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média±Std)					4,182E-03±2,489E-05	7,364E-04±4,414E-06
2	Resultado do Melhor do Experimento (Terceiro)					4,1415E-03	7,4382E-04
3	1	1	1	0,25	1	4,1110E-03	9,4079E-04
4	MFAC-CC					4,9255E-03	7,4091E-04

Os resultados da média dos experimentos realizados, foi melhor que o obtido com o MFAC-CC. E o resultado do Experimento 3 (Tabela 21) representa um melhor compromisso na minimização dos dois índices de desempenho. A Figura 19 mostra a Fronteira de Pareto obtida no Experimento 3.

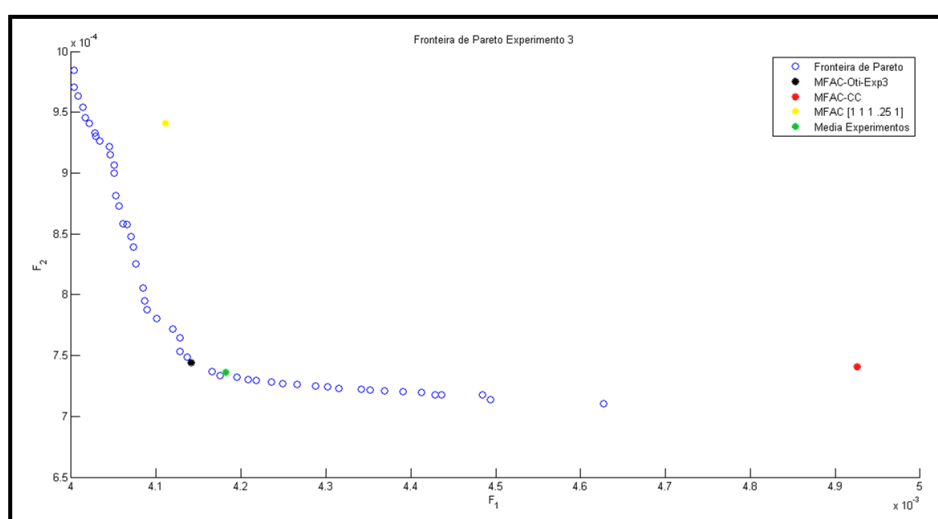
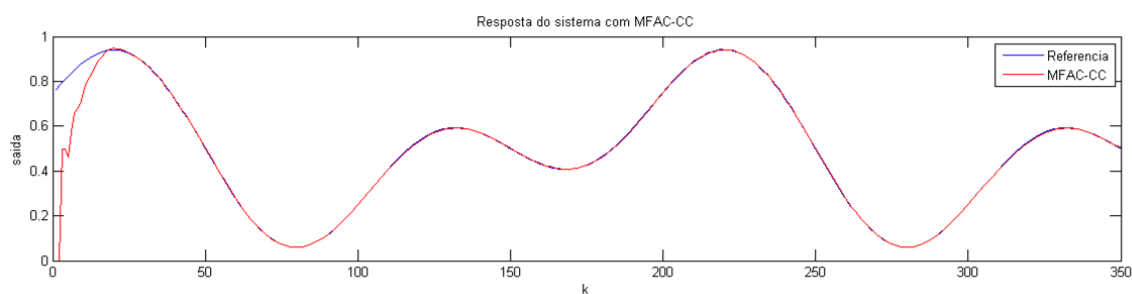


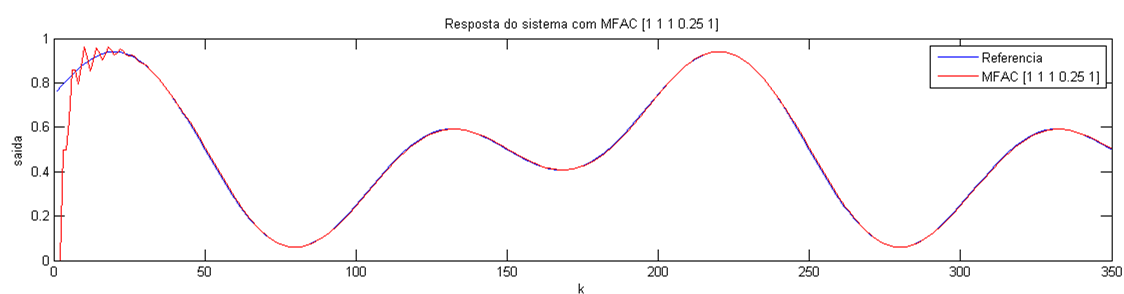
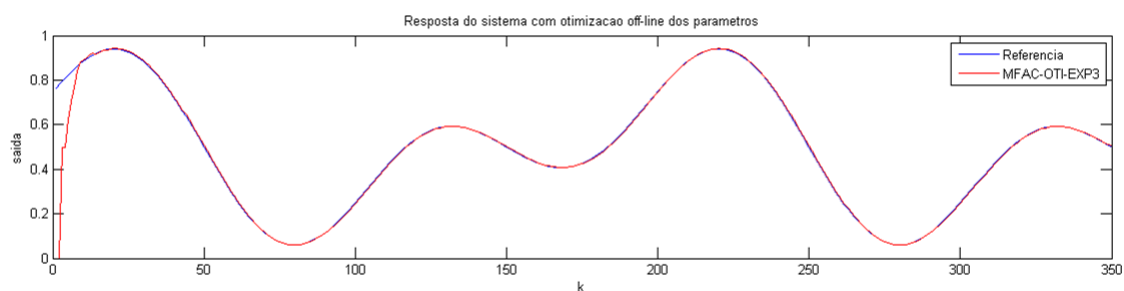
Figura 19 – Fronteira de Pareto e parâmetros em destaque

Observa-se que os índices obtidos com o MFAC-CC e o MFAC-CFDL com parâmetros $([1 \ 1 \ 1 \ 0,25 \ 1]^T)$ estão posicionados na região de soluções dominadas da fronteira de Pareto. Ressaltando, portanto, o sucesso da otimização dos parâmetros.

Nas Figuras 20 a-c são apresentadas as saídas do sistema para comparação da performance dos parâmetros otimizados em relação ao MFAC-CC o MFAC-CFDL com parâmetros da literatura e o MFAC-CFDL com parâmetros otimizados, respectivamente.



a) Resposta do sistema com MFAC - CC

b) Resposta do sistema com MFAC - CFDL $[1 \ 1 \ 1 \ 0,25 \ 1]^T$ 

c) Resposta do sistema com MFAC – CFDL – OTI - EXP3

Figura 20 - Simulação do sistema usando os parâmetros em destaque

Os resultados mostram que a estratégia adotada obteve melhores resultados em termos dos índices de desempenho. Visualmente os controladores apresentaram um bom desempenho, no entanto, o MFAC-CFDL com parâmetros otimizados aproximou a resposta do sistema ao sinal referência mais rápido e sem oscilações.

g) Sétimo caso de Estudo

Sistema apresentado por Junwei *et al.* (2016):

$$y(k+1) = 2,236y(k-1) - 1,64246y(k-2) + 0,385y(k-3) + 0,0018u(k-1) - 0,0036u(k-2) - 0,0025u(k-3) \quad (4.23)$$

e sinal de referência:

$$y^*(k+1) = \text{sen}(\pi t) \quad (4.24)$$

onde $t = kT_s$ e $T_s = 0,001$ (segundos).

Os resultados das dez simulações do problema de otimização estão apresentados na Tabela 23:

Tabela 23 – Resultados das otimizações *off-line* para o décimo caso de estudo.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	0,9987	1,5000	6,79E-04	8,3849	-4,0103	1,71E-04	5,72E-08
2	0,9971	1,9292	7,62E-04	8,5402	-4,1426	1,82E-04	5,53E-08
3	0,9956	1,1163	4,69E-04	8,1577	-3,7032	1,70E-04	5,80E-08
4	0,9940	1,5928	6,39E-04	8,4446	-3,9603	1,80E-04	5,57E-08
5	0,9963	1,3748	5,63E-04	8,2026	-3,9263	1,71E-04	5,76E-08
6	0,9991	1,9371	8,62E-04	8,2335	-3,9651	1,67E-04	5,80E-08
7	0,9996	1,5114	6,36E-04	8,3052	-4,3624	1,69E-04	5,77E-08
8	0,9953	1,0504	4,34E-04	8,5705	-4,0903	1,81E-04	5,53E-08
9	0,9954	1,7033	7,61E-04	8,5608	-4,2574	1,77E-04	5,61E-08
10	0,9965	1,9042	7,91E-04	8,4577	-4,3257	1,76E-04	5,62E-08
Média $\pm$ Desvio						1,74E-04+5,40E-06	5,67E-08+1,10E-09
Média Gerações						1500	
Média de Avaliações de Função (FES)						75100	

Na Tabela 24 é possível comparar os resultados dos experimentos com os valores obtidos usando outros parâmetros encontrados da literatura.

Tabela 24 - Tabela com o conjunto de parâmetros para o controlador.

N	Parâmetros					Funções Objetivo	
	ρ	η	μ	λ	$\phi(1)$	J_{MSE}	J_{MDSE}
1	Resultado da Média dos Experimentos (Média±Std)					1,74E-04+5,40E-06	5,67E-08+1,10E-09
2	Resultado do Melhor do Experimento (Sétimo)					1,69E-04	5,77E-08
3	1	1	1	45	-4	7,49E-04	8,64E-08

Mais uma vez, a comparação entre os resultados obtidos com os parâmetros otimizados e aqueles reportados com parâmetros da literatura, verifica-se a estratégia de otimização obteve desempenho superior.

Fronteira de Pareto obtida (Figura 21) usando considerando o resultado do Experimento 7 (Tabela 23).

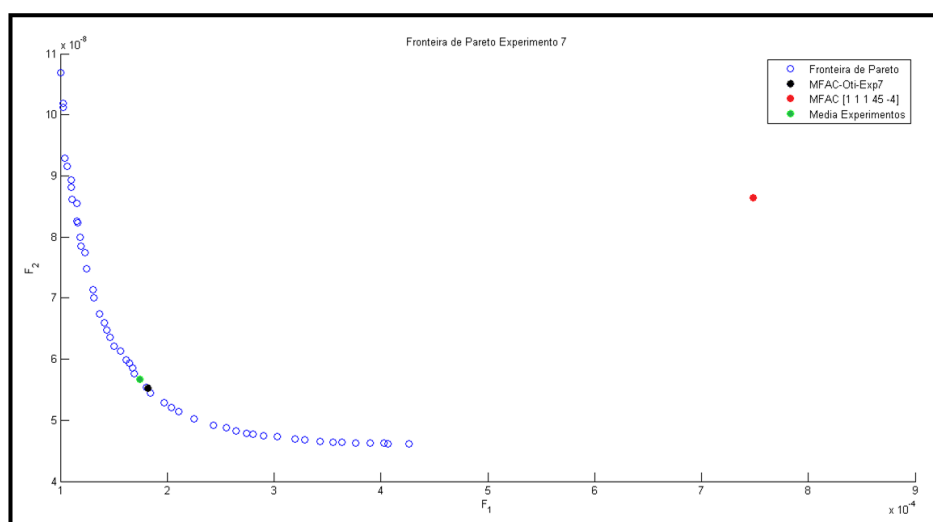
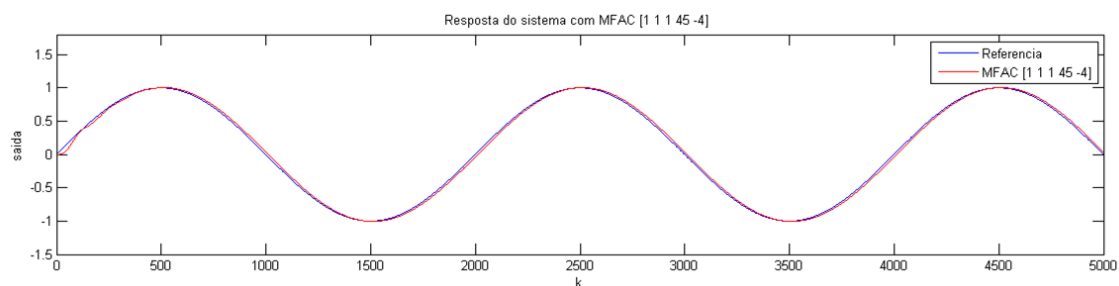


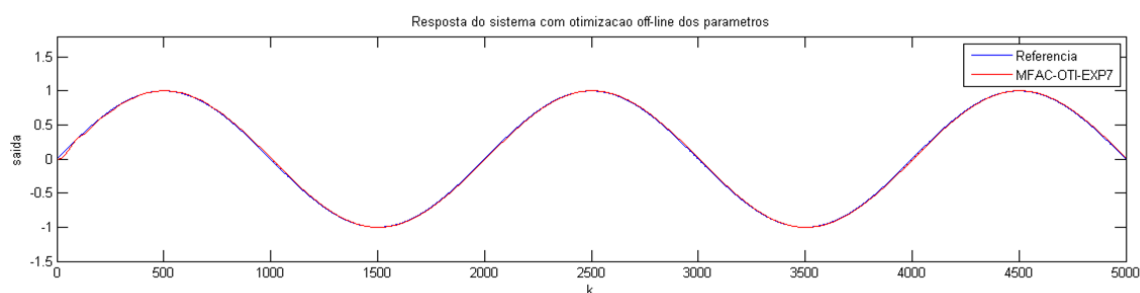
Figura 21 - Fronteira de Pareto e parâmetros em destaque.

Observe, na Figura 21, que a solução apresentada na literatura se encontra novamente na região dominada da fronteira de Pareto obtida.

Na Figura 22 a-b é apresentado a saída do sistema com o controlador usando os parâmetros da literatura e os ótimos.



a) Resposta do sistema com MFAC - CFDL [1 1 1 45 -4]^T



b) Resposta do sistema com MFAC – CFDL – OTI - EXP7

Figura 22 - Simulação do sistema usando os parâmetros em destaque.

A análise da Figura 22 mostra que o MFAC-CFDL, usando tanto os parâmetros da literatura quanto os do Experimento 7, obtiveram um bom resultado em relação ao objetivo principal que é seguir o sinal de referência. No entanto, se observarmos com mais cuidado, é possível verificar que a resposta do sistema para o controlador com os parâmetros otimizados, está mais próxima do sinal de referência.

4.2.2. Casos de Estudo para Sintonia *On-Line*

A estratégia de otimização *on-line* para os parâmetros do controlador apresentado no Capítulo 3, será aplicado aos sete casos de estudo abordados na sintonia *off-line*.

a) Primeiro Caso de Estudo

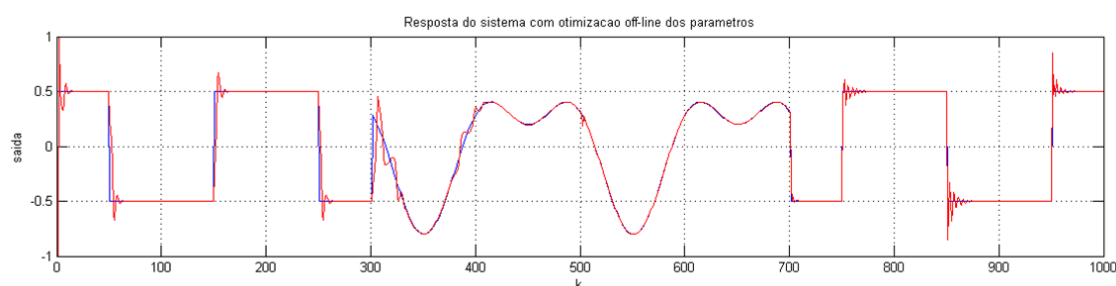
A Tabela 25 mostra um histórico da evolução dos parâmetros do controlador durante a simulação do sistema para a estratégia de controle *on-line* e al final os valores para os índices de desempenhos calculados:

Tabela 25 – Histórico dos parâmetros do controlador para o primeiro caso de estudo.

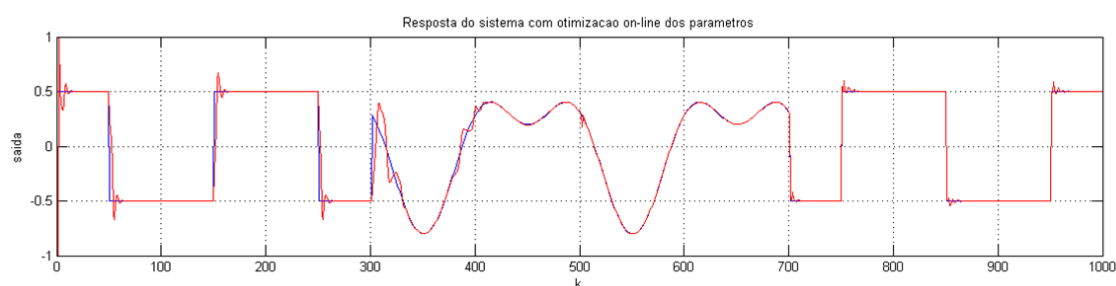
k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-290	0,9955	1,9768	0,3606	0,1310	0,8240
291-329	0,8569	1,5775	2,62E-5	0,4802	0,8240
330-690	0,9955	1,9768	0,3606	0,1310	0,8240
691-729	0,7097	1,7016	0,0002	0,0055	0,8240
730-740	0,9955	1,9768	0,3606	0,1310	0,8240
741-779	0,9395	0,4731	1,01E-5	0,0248	0,8240
780-840	0,9955	1,9768	0,3606	0,1310	0,8240
841-879	0,7020	1,9992	0,0152	0,1009	0,8240
880-940	0,9955	1,9768	0,3606	0,1310	0,8240
941-979	0,9207	1,9040	1,21E-5	0,3363	0,8240
980-1000	0,9955	1,9768	0,3606	0,1310	0,8240
J_{MSE}			7,57E-03		
J_{MDSE}			8,53E-03		

A Figura 23 a-b, a seguir, mostra a resposta do sistema (em vermelho) com sintonia *off-line* e *on-line*. Comparando os gráficos com a variação de parâmetros, na Tabela 23, é possível verificar a ação da estratégia de sintonia *on-line* proposta. Observe que nos instantes $k = 50, 150, 250$, a função de referência sofre variações

bruscas. Portanto, em torno desses pontos provavelmente foram realizadas otimizações, no entanto, como os parâmetros encontrados não devem ter produzido bons resultados, os parâmetros iniciais foram mantidos. Já nos instantes $k = 291$, 691, 741, 841 e 941, que precedem variações no sinal de referência, a Tabela 25 mostra que a partir desses instantes foram realizadas otimizações e novos parâmetros definidos para o controlador. Como resultado disso é possível verificar uma sensível redução no sobressinal e das oscilações em torno desses instantes. Além disso, é possível observar que nos instantes $k = 330$, 730, 780, 880 e 980, após os intervalos de otimização, os parâmetros sempre retornam aos valores iniciais conforme previsto.



a) Simulação com otimização *off-line*



b) Simulação com otimização *on-line*

Figura 23 - Simulação do sistema usando a otimização *off-line* e *on-line*.

Os gráficos e os índices de desempenho mostram que a estratégia com otimização de parâmetros *on-line* obteve resultados superiores, com redução no sobressinal e oscilações transientes, principalmente a partir do instante $k = 300$.

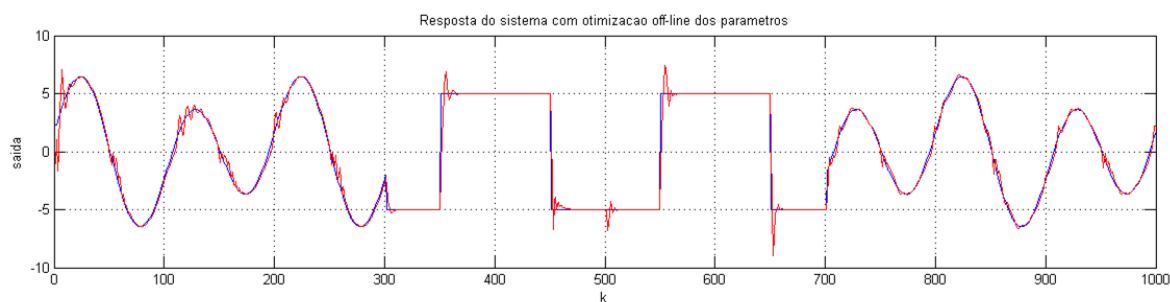
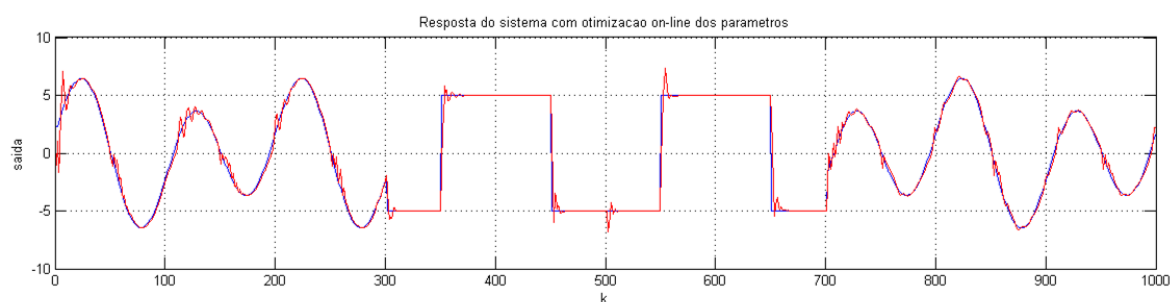
b) Segundo Caso de Estudo

A Tabela 26 novamente mostra um histórico da evolução dos parâmetros do controlador durante a simulação do sistema:

Tabela 26 - Histórico dos parâmetros do controlador para o segundo caso de estudo.

k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-290	0,7331	1,9968	0,3977	1,3789	1,6561
291-319	0,9913	1,9998	0,0027	0,4584	1,6561
320-340	0,7331	1,9968	0,3977	1,3789	1,6561
341-369	0,9910	1,9986	0,1050	1,9813	1,6561
370-440	0,7331	1,9968	0,3977	1,3789	1,6561
441-469	0,9980	1,9994	3,47E-5	2,5289	1,6561
470-540	0,7331	1,9968	0,3977	1,3789	1,6561
541-569	0,9991	1,9980	0,0031	2,0896	1,6561
570-640	0,7331	1,9968	0,3977	1,3789	1,6561
641-669	0,9940	1,7807	0,3181	8,3555	1,6561
670-690	0,7331	1,9968	0,3977	1,3789	1,6561
691-719	0,9882	0,0010	12,3214	0,2242	1,6561
720-1000	0,7331	1,9968	0,3977	1,3789	1,6561
J_{MSE}			4,04E-01		
J_{MDSE}			4,51E-01		

Na Figura 24 a-b são apresentadas as respostas do sistema para sintonia *off-line* e *on-line*, respectivamente. Mais uma vez o gráfico mostra que a estratégia *on-line* conseguiu melhorar a performance do sistema, reduzindo principalmente o sobressinal nos pontos nos quais o sinal de referência muda bruscamente.

a) Simulação com otimização *off-line*b) Simulação com otimização *on-line***Figura 24 - Simulação do sistema usando a otimização *off-line* e *on-line*.**

Também é importante ressaltar, na Figura 24, que a resposta do sistema não-linear de fase não mínima apresenta oscilações intermediárias, independente de variações bruscas na função. Nesses pontos a estratégia também realiza otimizações, entretanto, tornar o sistema muito sensível a essa característica, estabelecendo um limiar para o erro máximo muito pequeno, na estratégia de controle, pode introduzir distúrbios adicionais ao sistema e baixando a performance do controlador.

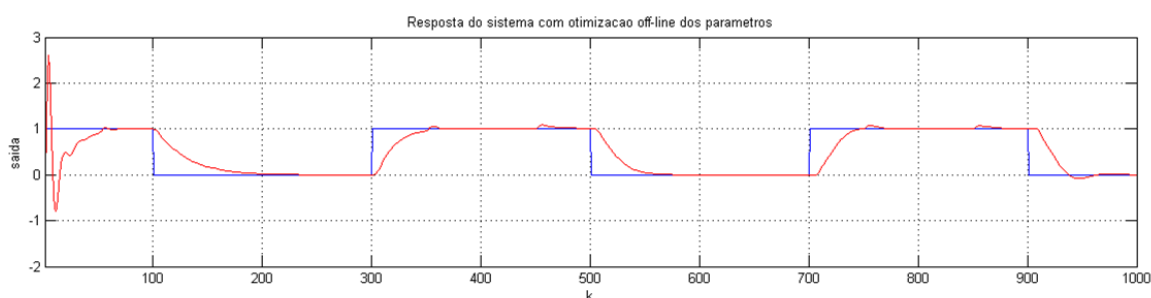
c) *Terceiro Caso de Estudo*

Novamente, na Tabela 27, é apresentado o histórico da evolução dos parâmetros do controlador durante a simulação do sistema:

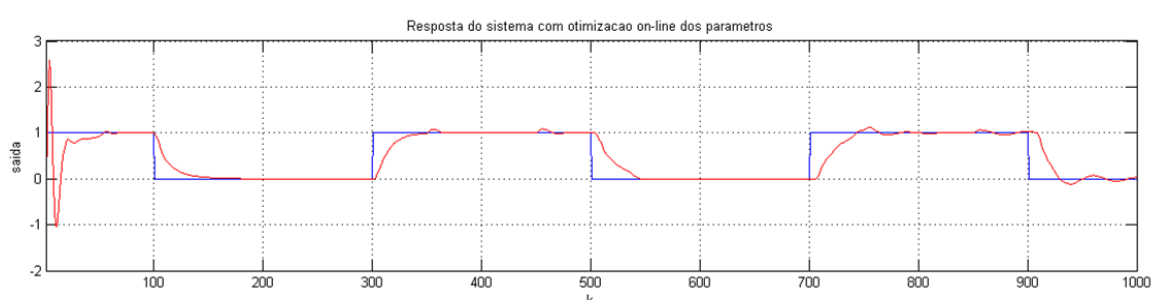
Tabela 27 - Histórico dos parâmetros do controlador para o terceiro caso de estudo.

k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-13	0.3800	1.6194	0.1343	2.1894	0.2524
14-22	0.9530	1.0382	1.3321	0.7412	0.2524
23-90	0.3800	1.6194	0.1343	2.1894	0.2524
91-129	0.8570	1.6442	0.0435	4.1716	0.2524
130-290	0.3800	1.6194	0.1343	2.1894	0.2524
291-329	0.9027	1.3970	0.0310	6.4170	0.2524
330-490	0.3800	1.6194	0.1343	2.1894	0.2524
491-529	0.7286	1.4478	0.0420	12.0573	0.2524
530-690	0.3800	1.6194	0.1343	2.1894	0.2524
691-729	0.7182	1.8131	0.0401	13.6263	0.2524
730-890	0.3800	1.6194	0.1343	2.1894	0.2524
891-929	0.9663	1.5477	0.0680	10.2232	0.2524
930-1000	0.3800	1.6194	0.1343	2.1894	0.2524
J_{MSE}			8,96E-02		
J_{MDSE}			1,37E-02		

A Figura 25 a-b, mostra a resposta do sistema com sintonia *off-line* e *on-line*.



a) Simulação com otimização *off-line*



b) Simulação com otimização *on-line*

Figura 25 - Simulação do sistema usando a otimização *off-line* e *on-line*.

O gráfico mostra que a estratégia *on-line*, tente fazer o sistema controlado retomar a direção da função de rastreamento de forma muito mais rápida, isso pode fazer o sistema oscilar um pouco quando a referência muda bruscamente, o que pode ser observado na Figura 26, a partir do instante $k = 700$. Entretanto, para o exemplo em estudo, a amplitude na oscilação foi muito pequena e foi compensada pela redução do erro médio no cálculo dos índices de desempenho.

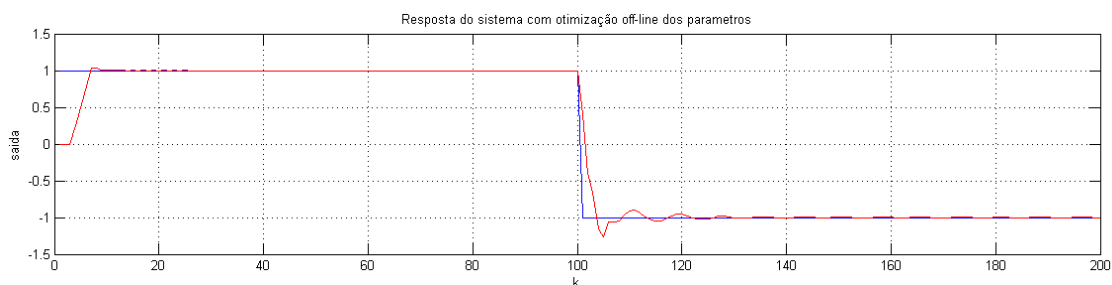
d) Quarto Caso de Estudo

A Tabela 28 mostra um histórico da evolução dos parâmetros do controlador durante a simulação do sistema:

Tabela 28 - Histórico dos parâmetros do controlador para o quarto caso de estudo.

k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-95	0.9785	1.7405	0.0004	0.5810	0.9246
96-134	0.9793	1.9893	0.0016	0.4737	0.9246
135-200	0.9785	1.7405	0.0004	0.5810	0.9246
J_{MSE}			3,06E-02		
J_{MDSE}			1,38E-02		

Na Tabela 28 como era esperado ocorreu apenas uma otimização, no instante $k = 96$. No entanto, examinando a Figura 26 a-b, a seguir, que mostra respectivamente as respostas do sistema para sintonia *off-line* e *on-line*, é possível observar que a estratégia *on-line* melhorou a resposta do sistema controlador reduzindo o sobressinal e as oscilações transientes em torno do instante $k = 100$.



a) Simulação com otimização *off-line*



b) Simulação com otimização *on-line*

Figura 26 - Simulação do sistema usando a otimização *off-line* e *on-line*.

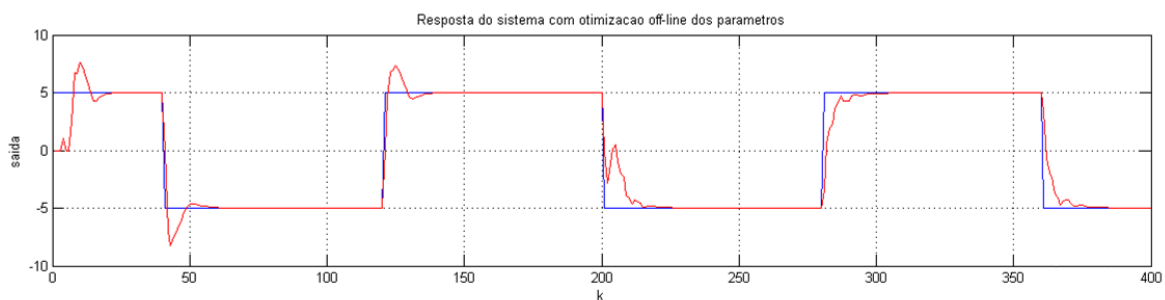
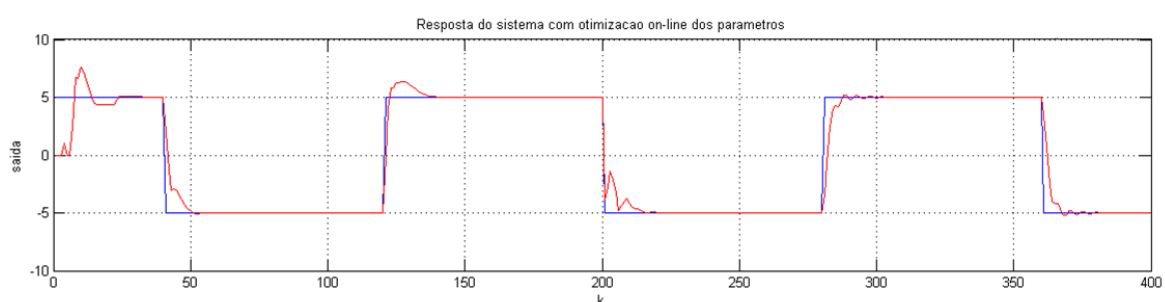
e) *Quinto Caso de Estudo*

A Tabela 29 mostra um histórico da evolução dos parâmetros do controlador durante a simulação do sistema:

Tabela 29 - Histórico dos parâmetros do controlador para o quinto caso de estudo.

k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-12	0,2637	1,7901	2,5861	1,6851E-04	0,6319
13-21	1,0001e-05	1,7698	0,0638	19,9989	0,6319
22-31	0,1272	1,4693	0,0085	0,4658	0,6319
32-35	0,2637	1,7901	2,5861	1,6851e-04	0,6319
36-74	0,9573	1,9884	0,3924	3,4472	0,6319
75-115	0,2637	1,7901	2,5861	1,6851E-04	0,6319
116-154	0,2496	1,9998	0,0648	1,3398e-04	0,6319
155-195	0,2637	1,7901	2,5861	1,6851E-04	0,6319
196-234	0,4652	1,8609	2,3771	4,0193e-04	0,6319
235-275	0,2637	1,7901	2,5861	1,6851E-04	0,6319
276-314	0,9979	0,6205	11,2598	0,0028	0,6319
315-355	0,2637	1,7901	2,5861	1,6851E-04	0,6319
356-394	0,9989	0,6340	15,1193	15,1193	0,6319
395-400	0,2637	1,7901	2,5861	1,6851E-04	0,6319
J_{MSE}			1,3744		
J_{MDSE}			0,8450		

A Figura 27 a-b mostra a resposta do sistema com sintonia *on-line* e *off-line*. Novamente, as figuras sugerem que a resposta da estratégia *on-line* apresentou menor sobressinal e erro em estado estacionário.

a) Simulação com otimização *off-line*b) Simulação com otimização *on-line*Figura 27 - Simulação do sistema usando a otimização *off-line* e *on-line*.f) *Sexto Caso de Estudo*

A Tabela 30 apresenta o histórico da evolução dos parâmetros do controlador durante a simulação do sistema:

Tabela 30 - Histórico dos parâmetros do controlador para o sexto caso de estudo.

k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-350	0,9849	1,8408	8,3908E-04	0,1659	0,9743
J_{MSE}			4,1415E-03		
J_{MDSE}			7,4382E-04		

Como já era esperado o sistema mantém os parâmetros otimizados *off-line* durante todo período de operação, pois não ocorrem descontinuidades no sinal de referência, nem ocorrem variações acentuadas no erro da resposta do sistema durante a simulação. Isto ilustra que a estratégia só atua quando há necessidade, portanto a atuação *on-line* funciona sem mudanças paramétricas desnecessárias. Na Figura 28 mostra a resposta do sistema usando a estratégias *on-line*.

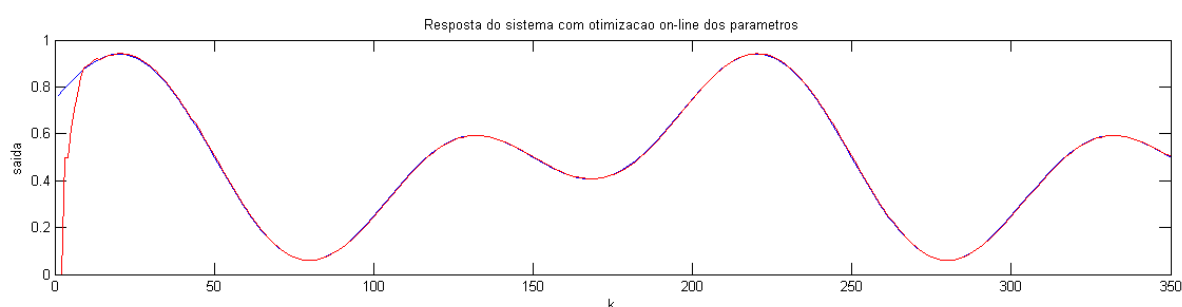


Figura 28 - Simulação do sistema usando a otimização *on-line*.

g) Sétimo Caso de Estudo

A Tabela 31 mostra um histórico da evolução dos parâmetros do controlador durante a simulação do sistema:

Tabela 31 - Histórico dos parâmetros do controlador para o sétimo caso de estudo

k	Parâmetros				
	ρ	η	μ	λ	$\phi(1)$
1-5000	0,9996	1,5114	6,36E-04	8,3052	-4,3624
J_{MSE}			1,69E-04		
J_{MDSE}			5,77E-08		

Novamente, como no caso de Experimento 6, a estratégia de controle manteve os mesmos parâmetros otimizados *off-line*, durante todo período de operação, pois, não ocorrem descontinuidades no sinal de referência, nem ocorrem variações acentuadas no erro da resposta do sistema.

Na Figura 29 é mostrada a resposta do sistema usando a estratégia *on-line*.

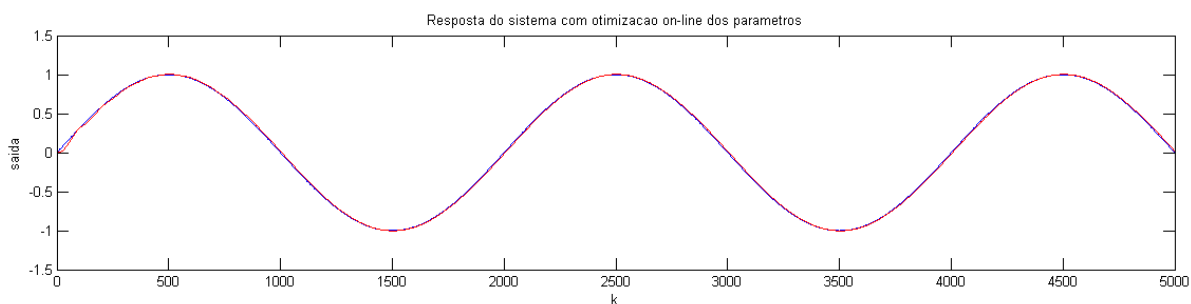


Figura 29 - Simulação do sistema usando a otimização *on-line*.

Para melhor compreensão, na Tabela 32 é apresentado um quadro resumo com os valores para as funções objetivo para todos os casos de estudo considerando as duas estratégias propostas.

Tabela 32 – Quadro resumo com índices de desempenho calculados usando as duas estratégias.

<i>Casos de Estudo</i>	<i>Estratégia Off-line</i>		<i>Estratégia On-line</i>	
	J_{MSE}	J_{MDSE}	J_{MSE}	J_{MDSE}
1	7,88E-03	8,93E-03	7,57E-03	8,53E-03
2	4,78E-01	4,83E-01	4,04E-01	4,51E-01
3	1,12E-01	1,31E-02	8,96E-02	1,37E-02
4	5,37E-02	1,43E-02	3,06E-02	1,38E-02
5	1,5521	0,9629	1,3744	0,8450
6	4,1415E-03	7,4382E-04	4,1415E-03	7,4382E-04
7	1,69E-04	5,77E-08	1,69E-04	5,77E-08

4.3 Discussões Finais

Na primeira parte deste capítulo, foram realizados testes para validação dos modelos de DE propostos. No caso de problemas mono-objetivo, aplicou-se as funções do concurso de otimização do CEC 2013. Para comparação também foram testados os algoritmos *DE/rand/1/bin* e CMA-ES. Os resultados sugerem que o algoritmo proposto apresenta em geral melhor desempenho que a DE canônica em todas as classes de funções testadas (monomodais, multimodais e compostas), exceto no caso de funções monomodais e baixas dimensões ($D = 5$). Já em relação ao CMA-ES, o algoritmo proposto obteve desempenho equivalente, no entanto, o CMA-ES teve dificuldades com a função 20. Entretanto, deve-se ressaltar que, apesar de bem documentada, a formulação do CMA-ES é complexa e está repleta de parâmetros, que precisam estar bem ajustados para o algoritmo ter um bom desempenho. Além disso, O CMA-ES também precisa executar operações computacionalmente custosas, como decomposição de matrizes em autovetores e autovalores, e sujeitas a erros numéricos capazes produzir resultados indesejados e sem significado físico para o problema, por exemplo, autovalores e vetores no domínio dos números complexos. Em contrapartida, a estratégia proposta usa operações simples da DE canônica; faz alguns cálculos estatísticos, média e matriz de covariância dos grupos; usa um algoritmo de agrupamento, que pode ser o mais simples possível; e funções de densidade de distribuição, para geração de imigrantes aleatórios normais e uniformes.

Para validar a versão multi-objetivo da DE proposta foram usadas as funções do CEC 2009. No experimento, foram testados três algoritmos: o NSDE, na forma canônica, uma variação do NSDE com a introdução de imigrantes aleatórios uniformes e o algoritmo proposto com imigrantes direcionados conforme modelo apresentado. Os resultados sugerem que a introdução de imigrantes direcionados ajuda a melhorar o desempenho do algoritmo NSDE canônico. Portanto, validando a aplicação da DE com imigrantes direcionados aos problemas de controle investigados.

Na segunda parte do capítulo, foram apresentados os resultados para estratégia de otimização *off-line* e *on-line* dos parâmetros de um controlador MFAC-CFDL usando o algoritmo evolucionário multi-objetivo concebido. Os dados foram obtidos mediante simulações, usando problemas de controle para sistemas diversos tirados da literatura. Os resultados da sintonia *off-line* indicam que, em geral, o controlador usando os parâmetros otimizados obteve melhor desempenho, quando comparado com o controlador e os parâmetros da literatura, ou seja, reduziu mais o sinal de erro no rastreamento e o sobressinal da resposta nos instantes em que a função de referência variava bruscamente. Além disso, o MFAC-CFDL otimizado também obteve resultados melhores que outras versões do MFAC, indicadas para controle de sistemas mais complexos. Enfatizando a necessidade otimização dos parâmetros em qualquer sistema de controle.

Finalmente, para os mesmos sistemas analisados na sintonia *off-line*, também foi testada a estratégia de sintonia *on-line*. Os resultados sugerem que, em geral, a sintonia *on-line* amplia os benefícios alcançados pelo uso de parâmetros ótimos, quais sejam, diminuição do erro em estado estacionário, redução do sobressinal e atenuação das oscilações nos períodos transientes, tudo isso em tempo de execução.

5

CONCLUSÃO

Nesta Tese, o problema de sintonia dos parâmetros de um controlador com estrutura fixa aplicado a sistemas com diferentes níveis de complexidade, usando um algoritmo evolucionário como método de otimização, é tratado. Para tanto, no controle do sistema, foram analisadas técnicas recentemente desenvolvidas, baseadas nos dados medidos do processo durante sua operação. O controle direcionado a dados tem recebido bastante atenção da área de controle nos últimos anos em virtude do aumento da complexidade dos processos industriais modernos. Dentre as principais técnicas DDC disponíveis atualmente, a técnica DDC MFAC (HOU; JIN, 2014) apresenta uma formulação teórica bastante robusta, é flexível para o tratamento de processos com diferentes níveis de não-linearidade, pode ser usado no controle de sistemas com múltiplas entradas e saídas e, além disso, pode ser combinada com estratégias de controle clássico, ou mesmo outras técnicas DDC. Apesar de todo seu desenvolvimento teórico, a escolha correta dos parâmetros do MFAC é uma questão em aberto (JI *et al.*, 2014). Nesta Tese, estudos são realizados visando contribuir com uma resposta adequada a essa questão.

O projeto de um sistema de controle, e em particular, o ajuste dos parâmetros de um controlador, pode ser transformado num problema de otimização com restrições. Os AEs são ferramentas de otimização flexíveis, cuja aplicação em projetos de controle, encontra registro na literatura (REYNOSO-MEZA *et al.*, 2014). No entanto, questões como: definição correta do problema de otimização, em termos de objetivos e restrições; tratamento da estocasticidade do método; e diminuição do tempo de computação necessário para obtenção dos resultados, precisam ser tratadas logo na fase de pré-projeto, e costumam ser barreiras à sua aplicação.

Durante este trabalho, essas dificuldades foram analisadas e motivaram a elaboração dos métodos para ajuste de parâmetros aqui proposto.

Dentre os AEs disponíveis, o algoritmo DE possui características desejáveis para aplicação em um projeto de controle, tais como, simplicidade, boa performance, realiza operações evolucionárias simples, e tem aplicação direta a problemas com números reais (PRICE *et al.*, 2005). Além disso, sua versão para otimização multi-objetivo, o algoritmo NSDE, goza de características similares e, por ser multi-objetivo, torna-se mais adequado ao projeto de um sistema de controle que normalmente envolve o atendimento a múltiplos critérios de desempenho dentro do projeto. Visando melhorar a performance desses dois algoritmos, a dinâmica da população de indivíduos durante o processo de evolução foi analisada, buscando com isso, extrair características desejáveis à otimização. Como resultado dessa análise, um modelo de AE baseado na introdução de imigrantes direcionados foi elaborado. Os resultados obtidos para os algoritmos propostos em testes com problemas mono e multi-objetivo da literatura, CEC2013 e CEC2009, respectivamente, permitem comparar o desempenho dos algoritmos elaborados com outros elencados na literatura e sugerem que a estratégia proposta pode acelerar e aumentar a eficiência da DE e do NSDE, particularmente em problemas com $D \leq 10$, como é o caso das aplicações em controle estudados.

A versão multi-objetivo do algoritmo proposto foi aplicada a um problema de otimização relacionado como projeto de controle, usando o MFAC-CFDL, para diversos sistemas encontrados na literatura. No problema de controle foram analisadas duas estratégias para ajuste dos parâmetros do controlador. Na primeira, a otimização dos parâmetros foi realizada de forma *off-line*, mediante experimentos de simulação usando diferentes tipos de sistemas. Os resultados obtidos nos testes, indicam que a performance do sistema de controle foi beneficiada com a otimização. Em geral, a capacidade de rastreamento dos sistemas controlados melhorou e podem ser observadas reduções no erro em estado estacionário, nos valores de sobressinal e no período total em que ocorrem oscilações transientes, para praticamente todos os experimentos, com destaque aos casos de estudo 1, 2, 4 e 5, nos quais esses benefícios podem ser melhor visualizados. Além disso, o MFAC-CFDL otimizado também obteve resultados melhores que outras duas versões do

MFAC descritas na literatura, MFAC-PFDL e MFAC-CC, aplicados aos casos de estudo 5 e 6, respectivamente, normalmente indicadas para o controle de sistemas mais complexos. Tudo isso, valida a estratégia usada e enfatiza a necessidade otimização dos parâmetros no projeto de um sistema de controle.

Na segunda estratégia, otimizações podem ser realizadas durante o período de operação da planta, ou seja, quando são detectadas situações de interesse, a saber, variações bruscas na função de rastreamento ou aumento exagerado do erro em estado estacionário, o método proposto realiza uma nova otimização dentro de uma janela de tempo definida, usando um modelo matemático identificado para o sistema controlado e informações como o conjunto de Pareto obtido na otimização *off-line* da fase de experimentação. Esta nova otimização produzirá outros parâmetros que serão usados no controlador real apenas durante o período de otimização considerado, ou seja, no final desse intervalo os parâmetros do controlador retornam aos valores iniciais. Novos resultados para os mesmos casos de estudos testados sugerem que o método de otimização *on-line* consegue, comparado à otimização *off-line*, aumentar ainda mais a capacidade de rastreamento do sistema de controle. Novamente, em geral, para todos os sistemas estudados, houve redução no erro em estado estacionário, diminuição do sobressinal e no período de oscilações durante o regime transiente.

Nesta Tese, buscou-se contribuir com o estado da arte das pesquisas relacionadas aos algoritmos evolucionários e sua aplicação em projetos de sistemas de controle. Os resultados obtidos durante o trabalho indicam que esses objetivos foram atendidos. No entanto, pesquisas futuras ainda são necessárias para aprimorar o modelo de AE proposto, para ajustar melhor seus parâmetros internos e testá-lo com outros problemas de otimização relevantes, principalmente relacionados ao controle. Além disso, novos casos de estudos com diferentes classes de sistemas precisam ser testados e outras restrições, também comuns ao projeto de controle, precisam ser incluídas no problema de otimização apresentado. Essas questões, e outras que surgirão no futuro, deverão ser respondidas com o avanço dessa linha de pesquisa.

5.1 Trabalhos Futuros

Alguns caminhos que podem ser seguidos para o andamento dessa pesquisa, os quais, resumidamente estão elencados a seguir:

- Os resultados obtidos com os algoritmos evolucionários propostos apesar de bons, ainda são inferiores aos melhores resultados das competições, CEC2009 e CEC2013. Portanto, no futuro, são necessários mais estudos para melhorar o ajuste dos parâmetros internos do algoritmo evolucionário proposto, visando aumentar sua eficiência e buscando alcançar os melhores resultados do concurso.
- Testar o modelo de AE proposto com novos problemas de otimização da literatura para identificar possíveis nichos de aplicação.
- Ajustar e até ampliar os critérios definidos na estratégia de otimização *on-line* para as otimizações intermediárias, buscando melhorar a qualidade do rastreamento.
- Buscar novos problemas de controle diversos dos já estudados para aplicar o processo de sintonia.
- Usar a estratégia *on-line* em combinação com outros tipos de controladores, PID, e outras versões do MFAC.
- Propor testes envolvendo ruído e simulando falhas no sistema de aquisição de dados durante a operação do sistema controlado.

REFERÊNCIAS

ALI, M.; SIARRY, P.; PANT, M. An efficient differential evolution based algorithm for solving multi-objective optimization problems. **European journal of operational research**, v. 217, n. 2, p. 404-416, 2012.

ÅSTRÖM, K. J.; WITTENMARK, B. **Adaptive control**. Mineola New York: Dover Publications, 2008.

BÄCK, T.; SCHWEFEL, H. P. An overview of evolutionary algorithms for parameter optimization. **Evolutionary computation**, v. 1, n. 1, p. 1-23, 1993.

CAMPI, M. C.; LECCHINI, A.; SAVARESI, S. M. Virtual reference feedback tuning: a direct method for the design of feedback controllers. **Automatica**, v. 38, n. 8, p. 1337-1346, 2002.

CAMPI, M. C.; LECCHINI, A.; SAVARESI, S. M. An application of the virtual reference feedback tuning method to a benchmark problem. **European Journal of Control**, v. 9, n. 1, p. 66-76, 2003.

CAO, R. M.; HOU, Z. S. Nonparametric model direct adaptive predictive control for linear motor. **Control Theory & Applications**, v. 25, n. 3, p. 587-590, 2008.

CHANG, C. S.; XU, D. Y. Differential evolution based tuning of fuzzy automatic train operation for mass rapid transit system. **IEE Proceedings-Electric Power Applications**, v. 147, n. 3, p. 206-212, 2000.

CHIEN, C. J. A discrete iterative learning control for a class of nonlinear time-varying systems. **IEEE Transactions on Automatic Control**, v. 43, n. 5, p. 748-752, 1998.

CHEN, J.; HUANG, T. C. Applying neural networks to on-line updated PID controllers for nonlinear process control. **Journal of process control**, v. 14, n. 2, p. 211-230, 2004.

COELHO, L. S.; AYALA, H. V. H.; COELHO, A. A. R. Variable structure control optimized by differential evolution approach applied to continuous stirred tank reactor. **Chemical Engineering Research and Design**, v. 100, p. 248-260, 2015.

COELHO, L. S.; COELHO, A. A. R. Model-free adaptive control optimization using a chaotic particle swarm approach. **Chaos, Solitons & Fractals**, v. 41, n. 4, p. 2001-2009, 2009.

COELHO, L. S.; PESSOA, M. W.; SUMAR, R. R.; COELHO, A. A. R. Model-free adaptive control design using evolutionary-neural compensator. **Expert Systems with Applications**, v. 37, n. 1, p. 499-508, 2010.

COELLO, C. A. C.; LAMONT, G. B.; VAN VELDHUIZEN, D. A. **Evolutionary algorithms for solving multi-objective problems**. New York: Springer, 2007.

DAS, S.; SUGANTHAN, P. N. Differential evolution: A survey of the state-of-the-art. **IEEE transactions on evolutionary computation**, v. 15, n. 1, p. 4-31, 2011.

DEB, K. **Multi-objective optimization using evolutionary algorithms**. Chichester: John Wiley & Sons, 2001.

DENAI, M. A.; PALIS, F.; ZEGHBIB, A. Modeling and control of non-linear systems using soft computing techniques. **Applied Soft Computing**, v. 7, n. 3, p. 728-738, 2007.

DERRAC, J.; García, S.; Molina, D.; Herrera, F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. **Swarm and Evolutionary Computation**, v. 1, n. 1, p. 3-18, 2011.

DOTOLIA, M.; FAYB, A.; MIŚKOWICZ, M.; SEATZU, C. A survey on advanced control approaches in factory automation. **IFAC-Papers On Line**, v. 48, n. 3, p. 394-399, 2015.

DRAGOI, E. N.; DAFINESCU, V. Parameter control and hybridization techniques in differential evolution: a survey. **Artificial Intelligence Review**, v. 45, n. 4, p. 447-470, 2016.

EIBEN, A. E.; SMITH, J. E. **Introduction to evolutionary computing**. Heidelberg: Springer, 2003.

ESPINOSA, H. E. P.; AYALA-SOLARES, J. R. The Power of Natural Inspiration in Control Systems. In: **Nature-Inspired Computing for Control Systems**. Springer International Publishing, 2016. p. 1-10.

FLEMING, P. J.; PURSHOUSE, R. C. Evolutionary algorithms in control systems engineering: a survey. **Control engineering practice**, v. 10, n. 11, p. 1223-1241, 2002.

FLOREA, A.; COFARU, I. I.; ROMAN, L.; COFARU, N. Applying the Multi-objective Optimization Techniques in the Design of Suspension Systems. **Journal of Digital Information Management**, v. 14, n. 6, 2016.

FOGEL, D. B. **Evolutionary computation: toward a new philosophy of machine intelligence**. John Wiley & Sons, 2006.

FUJISAKI, Y.; DUAN, Y.; IKEDA, M. System representation and optimal control in input-output data space. **IFAC Proceedings Volumes**, v. 37, n. 11, p. 185-190, 2004.

GARROZI, C., **Algoritmo Evolucionário Multiobjetivo Dinâmico com Inserção de Diversidade Direcionada**. 2012. Tese de Doutorado em Ciências da Computação – Centro de Informática, Universidade Federal de Pernambuco, 2012.

GEORGE, N. V.; PANDA, G. Advances in active noise control: A survey, with emphasis on recent nonlinear techniques. **Signal processing**, v. 93, n. 2, p. 363-377, 2013.

GONG, W.; CAI, Z.; ZHU, L. An efficient multiobjective differential evolution algorithm for engineering design. **Structural and Multidisciplinary Optimization**, v. 38, n. 2, p. 137-157, 2009.

GREFENSTETTE, D. J. J., Genetic Algorithms for Changing Environments, **Parallel Problem Solving from Nature**, Elsevier Science Publishers, The Netherlands, v. 2, p. 137-144, 1992.

GUARDABASSI, G. O.; SAVARESI, S. M. Virtual reference direct design method: an off-line approach to data-based control system design. **IEEE Transactions on Automatic Control**, v. 45, n. 5, p. 954-959, 2000.

HANSEN, N. The CMA evolution strategy: a comparing review. **Towards a new evolutionary computation**, p. 75-102, 2006.

HANSEN, N.; MÜLLER, S. D.; KOUMOUTSAKOS, P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). **Evolutionary computation**, v. 11, n. 1, p. 1-18, 2003.

HJALMARSSON, H. Iterative feedback tuning—an overview. **International journal of adaptive control and signal processing**, v. 16, n. 5, p. 373-395, 2002.

HJALMARSSON, H.; GEVERS, M.; GUNNARSSON, S.; LEQUIN, O. Iterative feedback tuning: theory and applications. **IEEE control systems**, v. 18, n. 4, p. 26-41, 1998.

HJALMARSSON, H.; GUNNARSSON, S.; GEVERS, M. A convergent iterative restricted complexity control design scheme. In: **Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on**. IEEE, 1994. p. 1735-1740.

HOFFMANN, F. Evolutionary algorithms for fuzzy control system design. **Proceedings of the IEEE**, v. 89, n. 9, p. 1318-1333, 2001.

HOLLAND, J. H. **Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence**. MIT press, 1992.

HORNBY, G. S. Steady-state ALPS for real-valued problems. In: **Proceedings of the 11th Annual conference on Genetic and evolutionary computation**. ACM, 2009. p. 795-802.

HOU, Z. S.; HUANG, W. The model-free learning adaptive control of a class of SISO nonlinear systems. In: **American Control Conference, 1997. Proceedings of the 1997**. IEEE, 1997. p. 343-344.

HOU, Z. S.; WANG, Z. From model-based control to data-driven control: survey, classification and perspective. **Information Sciences**, v. 235, p. 3-35, 2013.

HOU, Z. S.; XU, J. X.; YAN, J. An iterative learning approach for density control of freeway traffic flow via ramp metering. **Transportation Research Part C: Emerging Technologies**, v. 16, n. 1, p. 71-97, 2008.

HOU, Z. S.; JIN, S. **Model free adaptive control: theory and applications**. CRC press, 2013.

HUANG, B.; KADALI, R. **Dynamic modeling, predictive control and performance monitoring: a data-driven subspace approach**. Springer, 2008.

JALALI, A., PILTAN, F., GAVAHIAN, A., JALALI, M. Model-free adaptive fuzzy sliding mode controller optimized by particle swarm for robot manipulator. **International Journal of Information Engineering and Electronic Business**, v. 5, n. 1, p. 68, 2013.

JI, C.; WANG, J.; CAO, L.; JIN, Q. Parameters tuning of model free adaptive control based on minimum entropy. **IEEE/CAA Journal of Automatica Sinica**, v. 1, n. 4, p. 361-371, 2014.

JUANG, C. F., CHEN, Y. H.; JHAN, Y. H. Wall-following control of a hexapod robot using a data-driven fuzzy controller learned through differential evolution. **IEEE Transactions on Industrial Electronics**, v. 62, n. 1, p. 611-619, 2015.

JUNWEI, D., RONGMIN, C., ZHONGSHENG, H., YUNJIE, Z. Model-free adaptive predictive control for non-circular cutting derived CNC system. In: **Control and Decision Conference (CCDC), 2016 Chinese**. IEEE, 2016. p. 5772-5777.

KARIMI, A.; MIŠKOVIĆ, L.; BONVIN, D. Iterative correlation-based controller tuning with application to a magnetic suspension system. **Control Engineering Practice**, v. 11, n. 9, p. 1069-1078, 2003.

KARIMI, A.; VAN HEUSDEN, K.; BONVIN, D. Non-iterative data-driven controller tuning using the correlation approach. In: **Control Conference (ECC), 2007 European**. IEEE, 2007. p. 5189-5195.

KOZA, J. R. **Genetic programming: on the programming of computers by means of natural selection**. MIT press, 1992.

KOZA, J. R.; KEANE, M. A.; YU, J.; BENNETT III, F. H.; MYDLOWEC, W. Automatic creation of human-competitive programs and controllers by means of genetic programming. **Genetic Programming and Evolvable Machines**, v. 1, n. 1, p. 121-164, 2000.

KRISTINSSON, K.; DUMONT, G. A. System identification and control using genetic algorithms. **IEEE Transactions on Systems, Man, and Cybernetics**, v. 22, n. 5, p. 1033-1046, 1992.

KROHLING, R. A.; REY, J. P. Design of optimal disturbance rejection PID controllers using genetic algorithms. **IEEE Transactions on evolutionary computation**, v. 5, n. 1, p. 78-82, 2001.

KRSTLC M.; KANELAKOPOULOS, L.; KOKOTOVIC, P. V. **Nonlinear and adaptive control design**. New York: Wiley & Sons, 1995.

LENG, Y., LI, H., WANG, P., QIAO, Z. Model-free adaptive control with contractive constraints for nonlinear systems. In: **Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2014 Sixth International Conference on**. IEEE, 2014. p. 288-291.

LENNON, W. K.; PASSINO, K. M. Intelligent control for brake systems. **IEEE Transactions on control systems technology**, v. 7, n. 2, p. 188-202, 1999.

LI, Y. F.; SANSVINI, G.; ZIO, E. Non-dominated sorting binary differential evolution for the multi-objective optimization of cascading failures protection in complex networks. **Reliability Engineering & System Safety**, v. 111, p. 195-205, 2013.

LIANG J. J.; QU, B. Y.; SUGANTHAN, P. N.; HERNÁNDEZ-DÍAZ, A. G. Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. **Computational Intelligence Laboratory, Zhengzhou**

University, Zhengzhou, China and Nanyang Technological University, Singapore, **Technical Report**, v. 201212, p. 3-18, 2013.

LOSHCHILOV, I. CMA-ES with restarts for solving CEC 2013 benchmark problems. In: **Evolutionary Computation (CEC), 2013 IEEE Congress on**. Ieee, 2013. p. 369-376.

LU, L., XU, Y., ANTONIOU, C., BEN-AKIVA, M. An enhanced SPSA algorithm for the calibration of Dynamic Traffic Assignment models. **Transportation Research Part C: Emerging Technologies**, v. 51, p. 149-166, 2015.

MARKOVSKY, I.; WILLEMS, J., RAPISARDA, P., MOOR, B. Data driven simulation with applications to system identification. **IFAC Proceedings Volumes**, v. 38, n. 1, p. 970-975, 2005.

MEZURA-MONTES, E.; REYES-SIERRA, M.; COELLO, C. Multi-objective optimization using differential evolution: a survey of the state-of-the-art. **Advances in differential evolution**, p. 173-196, 2008.

MISHRA, P.; KUMAR, V.; RANA, K. P. S. An online tuned novel nonlinear PI controller for stiction compensation in pneumatic control valves. **ISA transactions**, v. 58, p. 434-445, 2015.

MIRKHANIA, M.; FORSATIB, R.; SHAHRIC, A. M.; MOAYEDIKIAD A. A novel efficient algorithm for mobile robot localization. **Robotics and Autonomous Systems**, v. 61, n. 9, p. 920-931, 2013.

MISKOVIC, L.; KARIMI, A.; BONVIN, D. Correlation-based tuning of a restricted-complexity controller for an active suspension system. **European journal of control**, v. 9, n. 1, p. 77-83, 2003.

NAKAMOTO, M. An application of the virtual reference feedback tuning for an MIMO process. In: **SICE 2004 Annual Conference**. IEEE, 2004. p. 2208-2213.

NEGRI, G. H.; CAVALCA, M. S. M.; PARPINELLI, R. S. Model-based predictive control using differential evolution applied to a pressure system. **IEEE Latin America Transactions**, v. 14, n. 1, p. 89-95, 2016.

NERI, Ferrante; MININNO, Ernesto. Memetic compact differential evolution for cartesian robot control. **IEEE Computational Intelligence Magazine**, v. 5, n. 2, p. 54-65, 2010.

PANDA, S. Robust coordinated design of multiple and multi-type damping controller using differential evolution algorithm. **International Journal of Electrical Power & Energy Systems**, v. 33, n. 4, p. 1018-1030, 2011.

PENG, C.; SUN, H.; GUO, J. Multi-objective optimal PMU placement using a non-dominated sorting differential evolution algorithm. **International Journal of Electrical Power & Energy Systems**, v. 32, n. 8, p. 886-892, 2010.

PENG, C.; SUN, H.; GUO, J.; LIU, G. Multi-objective optimal strategy for generating and bidding in the power market. **Energy conversion and management**, v. 57, p. 13-22, 2012.

PEZESHKI, S., BADAMCHIZADEH, M. A., GHIASI, A. R., GHAEMI, S. Control of overhead crane system using adaptive model-free and adaptive fuzzy sliding mode controllers. **Journal of Control, Automation and Electrical Systems**, v. 26, n. 1, p. 1-15, 2015.

PIOTROWSKI, A. P. Adaptive memetic differential evolution with global and local neighborhood-based mutation operators. **Information Sciences**, v. 241, p. 164-194, 2013.

PRICE, Kenneth; STORN, Rainer M.; LAMPINEN, Jouni A. **Differential evolution: a practical approach to global optimization**. Berlin: Springer Science & Business Media, 2005.

REYNOSO-MEZA, G.; BLASCO, X.; SANCHIS, J.; MARTÍNEZ, M. Controller tuning using evolutionary multi-objective optimisation: current trends and applications. **Control Engineering Practice**, v. 28, p. 58-73, 2014.

REYNOSO-MEZA, G.; GARCÍA-NIETO, S. Controller tuning by means of multi-objective optimization algorithms: A global tuning framework. **IEEE Transactions on Control Systems Technology**, v. 21, n. 2, p. 445-458, 2013.

REYNOSO-MEZA, G.; SANCHIS, J.; HERRERO, J. M.; RAMOS, C. Evolutionary auto-tuning algorithm for PID controllers. **IFAC Proceedings Volumes**, v. 45, n. 3, p. 631-636, 2012.

REZAYAT, F. On the use of an SPSA-based model-free controller in quality improvement. **Automatica**, v. 31, n. 6, p. 913-915, 1995.

ROHRS, C. E.; VALAVANI, L.; ATHANS, M.; STEIN, G. Robustness of adaptive control algorithms in the presence of unmodeled dynamics. In: **Decision and Control, 1982 21st IEEE Conference on**. IEEE, 1982. p. 3-11.

ROMAN, R. C.; RADAC, M. B.; PRECUP, R. E. Data-driven model-free adaptive control tuned by virtual reference feedback tuning. **Acta Polytech. Hungar**, v. 13, n. 1, p. 83-96, 2016.

ROSELYN, J. P.; DEVARAJ, D.; DASH, S. S. Multi Objective Differential Evolution approach for voltage stability constrained reactive power planning

problem. **International Journal of Electrical Power & Energy Systems**, v. 59, p. 155-165, 2014.

SAFONOV, M. G.; TSAO, T. C. The unfalsified control concept and learning. In: **Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on.** IEEE, 1994. p. 2819-2824.

SAFONOV, M. G., Data-driven robust control design: unfalsified control, 2003. <<http://routh.usc.edu/pub/safonov/safo03i.pdf>>. Acesso em: 09 jun. 2014

SAKAGUCHI, A.; YAMAMOTO, T. A study on system identification using GA and GMDH network. In: **Industrial Electronics Society, 2003. IECON'03. The 29th Annual Conference of the IEEE.** IEEE, 2003. p. 2387-2392.

SHEKIN, D. J. **Handbook of parametric and nonparametric statistical procedures.** CRC Press, 2004.

SOUSA, J. T. G.; FRANCA, J. E. M.; ARAÚJO, A. F. R. Differential Evolution-Based Parameter Tuning in Model-Free Adaptive Control. In: **Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on.** IEEE, 2015. p. 1726-1731.

SPALL, J. C. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. **IEEE transactions on automatic control**, v. 37, n. 3, p. 332-341, 1992.

SPALL, J. C.; CHIN, D. C. Traffic-responsive signal timing for system-wide traffic control. **Transportation Research Part C: Emerging Technologies**, v. 5, n. 3, p. 153-163, 1997.

SPALL, J. C.; CRISTION, J. A. A neural network controller for systems with unmodeled dynamics with applications to wastewater treatment. **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, v. 27, n. 3, p. 369-375, 1997.

TAJUDDINA, M. F. N.; AYOB, S. M.; SALAM, Z.; SAAD, M. S. Evolutionary based maximum power point tracking technique using differential evolution algorithm. **Energy and Buildings**, v. 67, p. 245-252, 2013.

TAN, K. K.; LEE, T. H.; HUANG, S. N.; LEU, F. M. Adaptive-predictive control of a class of SISO nonlinear systems. **Dynamics and control**, v. 11, n. 2, p. 151-174, 2001.

TAYARANI-N, M. H.; YAO, X.; XU, H. Meta-heuristic algorithms in car engine design: A literature survey. **IEEE Transactions on Evolutionary Computation**, v. 19, n. 5, p. 609-629, 2015.

UCHIYAMA, M. Formation of high-speed motion pattern of a mechanical arm by trial. **Transactions of the Society of Instrument and Control Engineers**, v. 14, n. 6, p. 706-712, 1978.

WERBOS, P. J. A menu of designs for reinforcement learning over time. in: W.T. Miller III, R.S. Sutton, P.J. Werbos (Eds.). **Neural Networks for Control**, MIT Press, p. 67–95. 1991,

XU, J. X.; HOU, Z. S., On learning control: the state of the art and perspective. **Acta Automatica Sinica**, v. 31, n. 6, p. 943–955, 2005.

XUHUI, B.; RUI, W.; RONGHU, C. Model Free Adaptive Control with Control Input Deadzone. **International Journal of Digital Content Technology and its Applications**, v. 7, n. 7, p. 706, 2013.

XU, D.; JIANG, B.; SHI, P. A novel model-free adaptive control design for multivariable industrial processes. **IEEE Transactions on Industrial Electronics**, v. 61, n. 11, p. 6391-6398, 2014.

YANG, S.; TINÓS, R. A hybrid immigrants scheme for genetic algorithms in dynamic environments. **International Journal of Automation and Computing**, v. 4, n. 3, p. 243-254, 2007.

YING, H.; YANG, L.; CIWEI, G. Multi-objective transmission network planning based on non-dominated sorting differential evolution. **Power System Technology**, v. 35, n. 3, p. 85-89, 2011.

YU, X.; TANG, K.; YAO, X. An immigrants scheme based on environmental information for genetic algorithms in changing environments. In: **Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence)**. IEEE Congress on. IEEE, 2008. p. 1141-1147.

ZHAO, Y., DU, X., XIA, G., JIA, R. A novel SPSA kernel wavelet neural network for model-free PID controller. In: **Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE**. IEEE, 2015. p. 001960-001965.

ZHANG, Bin; ZHANG, Weidong. Adaptive predictive functional control of a class of nonlinear systems. **ISA transactions**, v. 45, n. 2, p. 175-183, 2006.

ZHANG, Q.; ZHOU, A.; ZHA, O S.; SUGANTHAN P. N.; LIU , W.; TIWARI, S. Multiobjective optimization test instances for the CEC 2009 special session and competition. **University of Essex, Colchester, UK and Nanyang technological University, Singapore, special session on performance assessment of multi-objective optimization algorithms, technical report**, v. 264, 2008.

Apêndice A - Resumo Teórico sobre Técnicas de Controle DDC

Nas últimas décadas várias opções de DDC foram desenvolvidas. A seguir será apresentado um resumo com as principais técnicas DDC atuais (HOU, WANG, 2013).

a) *Simultaneous Perturbation Stochastic Approximation - SPSA- Based DDC*

Trata-se de uma estratégia DDC *on-line* proposta por Spall (1992), para um controlador de estrutura fixa, na qual os dados de E/S adquiridos do sistema em malha fechada, são usados para sintonizar os parâmetros do controlador. Portanto, o controlador, um aproximador de funções, pode ser caracterizado por uma estrutura fixa com parâmetros sintonizáveis. A Figura 30 apresenta o esquema da técnica SPSA-Based DDC.

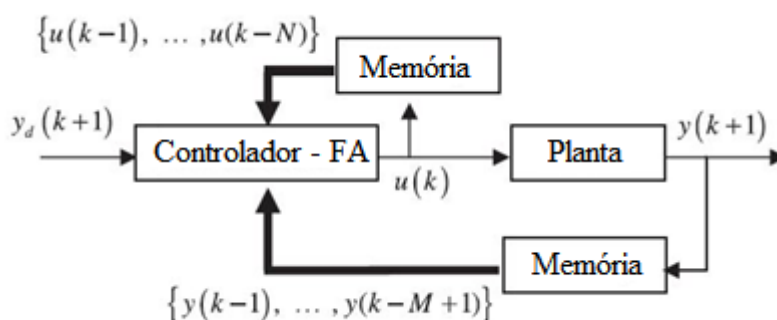


Figura 30 – Diagrama para SPSA DDC, adaptado de (HOU;WANG, 2013)

No instante k os dados de entrada para FA são as saídas do sistema e o sinal de controle, ambos dentro de uma janela de tempo fixa e antes do instante atual, e a saída desejada para a planta a um passo à frente.

$$y(k), y(k-1), \dots, y(k-M+1)$$

$$u(k-1), u(k-2), \dots, u(k-N), y_d(k+1) \quad (\text{A.1})$$

O objetivo do controlador passa a ser então determinar, a cada instante, os parâmetros θ^* ótimos do controlador capazes de minimizar o seguinte índice de desempenho.

$$J_k(\theta_k) = E[(y(\theta_k, k+1) - y_d(k+1))^2] \quad (\text{A.2})$$

Matematicamente, o ponto inicial para resolver esse problema seria conhecer o modelo da planta, para avaliar $\partial y(\theta_k, k+1)/\partial \theta_k$, entretanto essa informação não está disponível. O algoritmo SPSA então é usado para estimar de forma recursiva a sequência do vetor de parâmetros $\{\theta_k\}$ capaz de minimizar Eq. (A.2), mediante:

$$\hat{\theta}_k = \hat{\theta}_{k-1} - a_k \hat{g}_k(\hat{\theta}_{k-1}) \quad (\text{A.3})$$

onde, $\hat{\theta}_k$ é a estimativa do parâmetro na interação corrente, a_k é um parâmetro de escala, e $\hat{g}_k(\hat{\theta}_{k-1})$ é a estimativa da perturbação simultânea $g_k(\hat{\theta}_{k-1})$. Os elementos do vetor $\hat{g}_k(\hat{\theta}_{k-1})$ são calculados como:

$$\hat{g}_{kl}(\hat{\theta}_{k-1}) = \frac{j_k^{(+)} - j_k^{(-)}}{2c_k \Delta_{kl}} \quad (\text{A.4})$$

sendo, $l = 1, 2, \dots, L$, onde L denota o número de parâmetros do controlador, e

$$\hat{j}_{k+1}^{(\pm)} = \left(\hat{y}_{k+1}^{(\pm)} - y_d(k+1) \right)^2 \quad (\text{A.5})$$

calculado usando $\hat{y}_{k+1}^{(\pm)}$, sendo $\hat{y}_{k+1}^{(\pm)}$ a saída medida do sistema quando as entradas forem $u_{k+1}^{(\pm)}$, onde $u_{k+1}^{(\pm)}$ é a entrada gerada pelo controlador quando seus parâmetros forem calculados para $\theta_k = \hat{\theta}_k \pm c_k \Delta_k$, sendo $\Delta_k = \{\Delta_{k1}, \Delta_{k2}, \dots, \Delta_{kL}\}^T$ um vetor estocástico e o coeficiente c_k um fator de escala, considerado normalmente como uma constante ou uma sequência que tende a zero. Pelo procedimento descrito acima só é necessário então dois experimentos em malha fechada na interação antes de estimar os valores de $\hat{g}_k(\hat{\theta}_{k-1})$ de $g_k(\hat{\theta}_{k-1})$ a partir dos dados medidos.

A condição suficiente para convergência do SPSA-Based DDC pode ser achada na literatura (HOU; WANG, 2013). Em resumo, se todas as condições de

operação do sistema se mantém, e θ^* existe então $(\theta_k - \theta^*)$ se aproxima de zero quando k tende ao infinito.

b) Model-Free Adaptive Control - MFAC

A técnica MFAC foi proposta por Hou (HOU; HUANG, 1997). A ideia central do método é construir um modelo de dados dinâmicos linearizado equivalente para a planta, em cada ponto de operação, usando o conceito denominado Pseudo-Derivada Parcial – PPD (*Pseudo Partial Derivative*), cuja estimação é realizada usando apenas os dados de entrada e saída *on-line* do sistema. A partir desse modelo virtual do sistema, é possível projetar uma estratégia de controle adaptativo livre de modelo, para sistemas não-lineares discretos. Na próxima seção serão apresentados maiores detalhes sobre a técnica DDC MFAC, cujo estudo aprofundado é um dos objetivos desse trabalho. Por enquanto, apenas uma ideia central do método vai ser abordada.

Um sistema geral discreto SISO pode ser descrito como:

$$y(k+1) = f(y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)) \quad (\text{A.6})$$

onde, $y(k)$ e $u(k)$ são as entradas e saídas da planta controlada no instante k , e n_y e n_u são as ordens desconhecidas da entrada e da saída e $f(\cdot)$ é uma função não-linear desconhecida.

Se um sistema satisfizer às condições generalizadas de Lipschitz, isto é $|\Delta y(k+1)| \leq b|\Delta u(k)|$, ou condições similares para qualquer k fixo e $|\Delta u(k)| \neq 0$, então a equação geral do sistema pode ser expressa com um dos três tipos de modelos de dados linearizados dinâmicos, e a pseudo-derivada parcial é uniformemente limitada para qualquer k fixo.

- i. Modelo de dados linearizados dinâmico na forma compacta – CFDL (*Compact Form Dynamic Linearization*).

$$y(k+1) = y(k) + \phi(k)\Delta u(k) \quad (\text{A.7})$$

sendo, $\phi(k)$ a pseudo derivada parcial do sistema controlado no instante k

- ii. Modelo de dados linearizados dinâmico na forma parcial – Modelo PFDL (*Partial Form Dynamic Linearization*).

$$y(k+1) = y(k) + \phi^T(k)\Delta u(k) \quad (\text{A.8})$$

sendo,

$$\phi(k) = [\phi_1(k) \dots \phi_L(k)]^T$$

e

$$\Delta u(k) = [\Delta u(k) \dots \Delta u(k-L+1)]^T$$

onde, $\phi(k)$ é o vetor de pseudo derivada parcial do sistema controlado e L é a constante de ordem de linearização da entrada de controle.

- iii. Modelo de dados linearizados dinâmico na forma completa – FFDL (*Full Form Dynamic Linearization*).

$$y(k+1) = y(k) + \phi^T(k)\Delta u(k), \quad (\text{A.9})$$

onde,

$$\phi(k) = [\phi_1(k) \dots \phi_{L_u}(k) \phi_{L_u+1}(k) \dots \phi_{L_u+L_y}(k)]^T$$

e

$$\Delta u(k) = [\Delta u(k) \dots \Delta u(k-L_u+1) \Delta y(k) \dots \Delta y(k-L_y+1)]^T$$

sendo, L_u e L_y as pseudo ordens da entrada e saída do sistema e $\phi(k)$ é o vetor de pseudo derivada parcial do sistema .

Com ajuda da técnica de linearização dinâmica da planta, o projeto do controlador é simplificado. Por exemplo, usando o modelo CFDL é possível calcular o sinal de controle capaz guiar o sistema em uma determinada trajetória desejada, pela minimização de um índice de performance definido por:

$$J(u(k)) = (y_d(k+1) - y(k+1))^2 + \lambda(u(k) - u(k-1))^2 \quad (\text{A.10})$$

onde y_d é o sinal de referência e λ é um fator de ponderação. Para tanto, inicialmente a PPD do modelo deve ser estimada, por exemplo, usando um algoritmo de projeção, ou por mínimos quadrados. Depois o sinal de controle é obtido derivando Eq. (A.10) em relação a $u(k)$ e igualando o resultado a zero. Obtendo assim, o seguinte esquema de controle DDC MFAC – CFDL:

Para cada instante k , calcule a estimativa da PPD, mediante:

$$\hat{\phi}(k) = \hat{\phi}(k-1) + \frac{\eta \Delta u(k-1)}{\mu + \Delta u(k-1)^2} (\Delta y(k) - \hat{\phi}(k-1) \Delta u(k-1)) \quad (\text{A.11})$$

considerando a seguinte condição de reinicialização:

$$\hat{\phi}(k) = \hat{\phi}(1), \text{ if } |\hat{\phi}(k)| \leq \varepsilon, \text{ or } |\Delta u(k-1)| \leq \varepsilon \quad (\text{A.12})$$

Depois calcule o sinal de controle correspondente, usando:

$$u(k) = u(k-1) + \frac{\rho \hat{\phi}(k)}{\lambda + |\hat{\phi}(k)|^2} (y_d(k+1) - y(k)) \quad (\text{A.13})$$

onde ρ e η , λ e μ são fatores ponderadores, e ε uma pequena constante positiva.

c) *Unfalsified Control (UC) DDC*

Proposto em 1995 por Safonov (SAFONOV; TSAO, 1997) a técnica UC baseia-se no objetivo de qualquer teoria de controle para sistemas em malha fechada, qual seja, determinar uma lei de controle, tal que a resposta do conjunto controlador e planta, satisfaça especificações de performance pré-determinadas. O paradigma UC DDC escolhe um controlador, dentro de uma classe de controladores candidatos, a partir da checagem de consistência entre as especificações requeridas e as leis de controle produzidas por cada controlador candidato. O UC recursivamente falseia/invalida um conjunto de parâmetros de controle que falham em satisfazer as especificações exigidas.

Todo esse processo de invalidação dos controladores candidatos é realizado *on-line* com base nos dados de entrada e saída da planta controlada. Trata-se, portanto, de uma espécie de controle por comutação. No caso, o UC pode falsear o controlador que não consegue estabilizar o sistema controlado, antes mesmo dele ser inserido ao laço de realimentação. Os principais elementos de um UC são: um

conjunto de candidatos a controlador que sejam inversíveis, especificações de custo detectáveis como performance, e um mecanismo de comutação.

Um caso simples de UC pode ser observado na Figura 31:

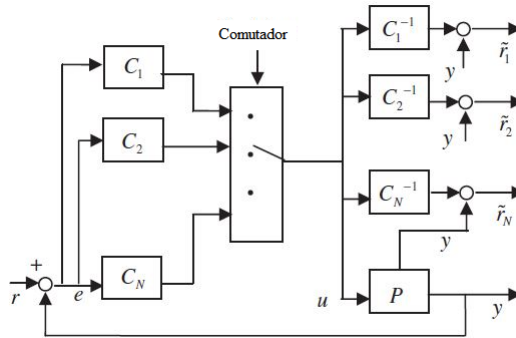


Figura 31 – Esquema simplificado de controle UC, adaptado de (HOU;WANG, 2013)

onde P é a planta a ser controlada desconhecida e $C_j, j = 1, \dots, N$ os controladores candidatos inversíveis. No instante atual, k , os dados da planta e do controlador coletados dentro do intervalo $[0, k - 1]$ são usados para avaliar os controladores candidatos, a partir de seu sinal fictício de referência $\tilde{r}_j$. Assim, quando o controlador ótimo é determinado para o instante k , o sistema é posto em malha fechada.

O sinal fictício de referência para o controlador C_j é calculado mediante:

$$\tilde{r}_j(\tau) = C_j^{-1}(u(\tau)) + y(\tau) \quad (\text{A.14})$$

e cada controlador C_j é avaliado usando um índice de performance $J(u, y, \tilde{r})$ e o conjunto de dados $\{(u(\tau), y(\tau), \tilde{r}_j(\tau)) | \tau \in [0, k - 1]\}$.

Como um exemplo de um índice de performance usado na literatura para o UC, tem-se:

$$J_j(k) = J(u, y, \tilde{r}, k) = \max_{\tau \in [0, k]} \frac{\|u(\tau)\|^2 + \|\tilde{r}_j - y(\tau)\|^2}{\|\tilde{r}_j\|^2 + \alpha}, \quad \alpha > 0 \quad (\text{A.15})$$

Nesse caso, o controlador ideal escolhido para o instante k é obtido calculando $J_j(k), j = 1, \dots, N$ e igualando $j^*(k) = \arg \min_{j=1,2,\dots,N} J_j(k)$, $C_{j^*}(k)$. Trata-se,

portanto, de uma estratégia de decisão minimax, que visa encontrar o controlador que minimize os maiores erros. Em cada instante o UC elimina todos $N - 1$ controladores exceto o considerado ótimo e conecta ele ao sistema.

d) *Iterative Feedback Turning (IFT)*

Proposto em 1994 por Hjalmarsson (HJALMARSSON *et al.*, 1998), o IFT é uma técnica DDC baseada na aquisição de dados *off-line*, que envolve a otimização de parâmetros para um controlador com estrutura fixa de acordo com uma estimativa do gradiente para um critério de desempenho. A cada iteração, a estimativa do gradiente é construída por um conjunto finito de dados, obtidos a partir da realização de dois experimentos distintos, conforme será apresentado a seguir.

Considere um sistema SISO em malha fechada linear invariante no tempo (LIT) é mostrado na Figura 32.

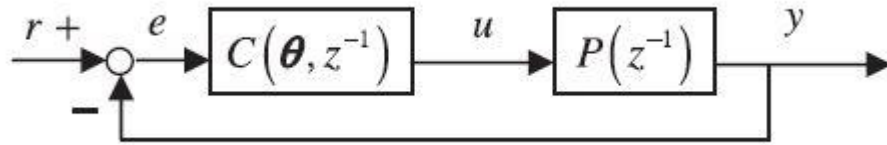


Figura 32 – Diagrama de blocos da planta e o controlador, adaptado de (HOU;WANG, 2013)

onde $P(z^{-1})$ representa a equação da planta, $C(\theta, z^{-1})$ representa um controlador Linear Invariante no Tempo - LIT de estrutura fixa e parâmetros sintonizáveis, θ é o vetor de parâmetros do controlador e os sinais r , u e y são a referência, sinal de controle e saída da planta, respectivamente.

Para o problema de controle, pode-se definir o seguinte índice de performance:

$$J(\theta) = \frac{1}{2N} \sum_{k=1}^N (y(\theta, k) - y_d(k))^2 \quad (\text{A.16})$$

sendo $y(\theta, k)$ a saída do sistema em malha fechada, y_d a saída desejada e N o número de amostras consideradas. O objetivo é encontrar o vetor de parâmetros ótimos θ^* capaz de minimizar $J(\theta)$, ou seja:

$$\theta^* = \arg \min_{\theta} (J(\theta)) \quad (\text{A.17})$$

Se o gradiente $\partial J/\partial \theta$ fosse disponível, então θ^* poderia ser obtido usando o método de otimização de descida de encosta, mediante o seguinte processo iterativo

$$\theta_{i+1} = \theta_i - \gamma_i R_i^{-1} \frac{\partial J(\theta_i)}{\partial \theta} \quad (\text{A.18})$$

onde γ_i é um escalar positivo que representa o tamanho do passo, e R_i é uma matriz positiva definida apropriada.

Derivando o critério de performance escolhido em relação ao vetor de parâmetros, obtém-se

$$\frac{\partial J(\theta_i)}{\partial \theta} = \frac{1}{N} \sum_{k=1}^N (y(\theta_i, k) - y_d(k)) \frac{\partial y(\theta_i, k)}{\partial \theta}. \quad (\text{A.19})$$

Uma vez que a saída em malha fechada $y(\theta_i, k)$ pode ser medida, e $y_d(k)$ é conhecida, somente o termo $\partial y(\theta_i, k)/\partial \theta$ não pode ser calculado quando a planta $P(z^{-1})$ é desconhecida.

A resposta em malha fechada da planta mais o controlador pode ser descrita como:

$$y(\theta) = \frac{C(\theta, z^{-1})P(z^{-1})}{1 + C(\theta, z^{-1})P(z^{-1})} r. \quad (\text{A.20})$$

Derivando Eq. (A.20) em relação ao vetor de parâmetros do controlador, obtém-se:

$$\frac{\partial y(\theta)}{\partial \theta} = \frac{1}{C(\theta, z^{-1})} \frac{\partial C(\theta, z^{-1})}{\partial \theta} \left[\frac{C(\theta, z^{-1})P(z^{-1})}{1 + C(\theta, z^{-1})P(z^{-1})} (r - y(\theta)) \right]. \quad (\text{A.21})$$

Comparando o termo entre colchetes da Eq. (A.21) com o termo também entre colchetes da Eq. (A.20), observa-se que o primeiro termo pode ser obtido aplicando como referência o sinal $(r - y(\theta))$ ao sistema em malha fechada.

Sedo assim, dois experimentos estão envolvidos no algoritmo IFT para cada interação. Primeiro, no experimento chamado normal, são coletadas N amostras da saída da planta $y_1(\theta_i)$ para o sinal de referência igual à saída desejada. No segundo experimento, chamado experimento do gradiente, são coletadas N amostras da saída da planta $y_2(\theta_i)$ considerando como sinal de referência $(r - y_1(\theta_i))$. Com esses dois pares de dados, estima-se o valor de $\partial y(\theta_i, k)/\partial \theta$, ou seja:

$$\frac{\partial \hat{y}(\theta_i)}{\partial \theta} = \frac{1}{c(\theta_i, z^{-1})} \frac{\partial c(\theta_i, z^{-1})}{\partial \theta} y_2(\theta_i) \quad (\text{A.22})$$

Finalmente, usando $\partial \hat{y}(\theta_i, k)/\partial \theta$, $k = 1, \dots, N$, estima-se $\partial \hat{f}(\theta_i)/\partial \theta$ e atualiza os parâmetros do controlador a cada iteração na direção do vetor de parâmetros ótimos.

f) Correlation-based Tuning (CbT)

Proposto por Karimi *et al.* em 2002 (KARIMI *et al.*, 2003) o CbT tem como ideia base ajustar iterativamente os parâmetros do controlador de modo a descorrelacionar o sinal de referência adotado e sinal de erro na saída, calculado pela diferença entre as saídas em malha fechada do sistema real e de um sistema ideal, ou se possível reduzir essa correlação. O diagrama de blocos do método CbT pode ser observado na Figura 33:

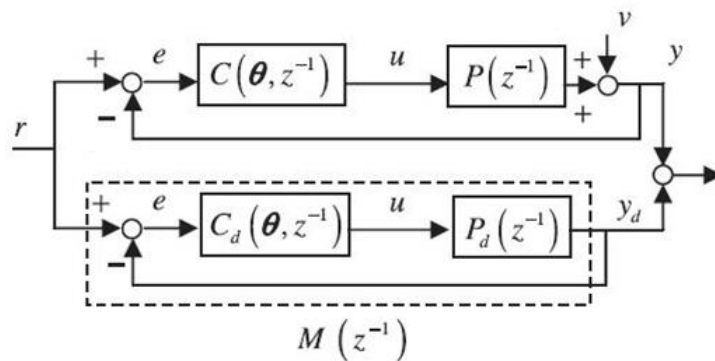


Figura 33 – Esquema Correlation-based Tuning (CbT) , adaptado de (HOU;WANG, 2013)

onde $P(z^{-1})$ representa a equação da planta real e $C(\theta, z^{-1})$ é o controlador LIT real de estrutura fixa, com vetor de parâmetro θ , e os sinais r , u , y , y_d e v são a referência, sinal de controle, saída da planta, a saída do modelo de referência e o distúrbio, respectivamente. No diagrama $C_d(\theta, z^{-1})$ representa o controlador ideal projetado para o modelo da planta $P_d(z^{-1})$, tal que, o bloco $M(z^{-1})$ na Fig.6, modelo da planta em malha fechada com o controlador ideal, é o modelo de referência para o problema.

A resposta do sistema real em malha fechada com o controlador a um sinal de referência r na presença de um distúrbio v , pode ser escrita como:

$$y(\theta) = \frac{C(\theta, z^{-1})P(z^{-1})}{1+C(\theta, z^{-1})P(z^{-1})} r + \frac{1}{1+C(\theta, z^{-1})P(z^{-1})} v \quad (\text{A.23})$$

Entretanto, a resposta desejada para o sistema deve ser calculada usando o modelo de referência na ausência de distúrbio, ou seja:

$$y_d = M(z^{-1})r = \frac{C_d(\theta, z^{-1})P_d(z^{-1})}{1+C_d(\theta, z^{-1})P_d(z^{-1})} r \quad (\text{A.24})$$

O erro da resposta em malha fechada, será a diferença entre os dois sinais de saída, portanto:

$$\varepsilon = y - y_d = \frac{C(\theta, z^{-1})P(z^{-1}) - C_d(\theta, z^{-1})P_d(z^{-1})}{(1+C(\theta, z^{-1})P(z^{-1}))(1+C_d(\theta, z^{-1})P_d(z^{-1}))} r + \frac{1}{1+C(\theta, z^{-1})P(z^{-1})} v \quad (\text{A.25})$$

Observe que o sinal de erro contém termos com contribuições da diferença entre $C(\theta, z^{-1})P(z^{-1})$ e $C_d(\theta, z^{-1})P_d(z^{-1})$, e do distúrbio v . Tendo em vista que, apenas no primeiro termo o erro está correlacionado com o sinal de referência. Se o vetor de parâmetros θ for sintonizado de forma que o sinal de erro não esteja correlacionado com o sinal de referência r , a diferença entre $C(\theta, z^{-1})P(z^{-1})$ e $C_d(\theta, z^{-1})P_d(z^{-1})$ deve se anular. Portanto, na ausência de distúrbio, o sistema real deverá seguir o modelo de referência.

A função de correlação cruzada pode ser escrita como:

$$\xi(\theta) = E\{\hat{\xi}(\theta)\} \quad (\text{A.26})$$

e $E\{\}$ denota a expectativa matemática, sendo $\hat{\xi}(\theta)$ definido como:

$$\hat{\xi}(\theta) = \frac{1}{N} \sum_{k=1}^N \zeta(k) \varepsilon(\theta, k) \quad (\text{A.27})$$

onde $\varepsilon(\theta, k)$ é o erro em malha fechada quando $C(\theta, z^{-1})$ está no laço, $\zeta(k)$ é a variável instrumental correlacionada com $r(k)$ e independente de $v(k)$, e N o número de dados.

Quando o conjunto de controladores, definido pela variação possível do vetor de parâmetros, é grande o suficiente para permitir a perfeita descorrelação

entre os sinais de ε e r , a técnica CbT calcula os parâmetros do controlador, como sendo as raízes da função de correlação cruzada. Entretanto, se os parâmetros calculados estão fora da região factível, o CbT apenas atualiza os parâmetros do controlador, no sentido da minimização da função de correlação cruzada, o que é chamado de redução de correlação. Portanto, o procedimento passa a ser encontrar as raízes da equação de correlação cruzada, ou seja:

$$\xi(\theta) = 0 \quad (\text{A.28})$$

Para o problema, $\hat{\xi}(\theta)$ pode ser visto como uma medida de $\xi(\theta)$ com ruído e a solução da equação acima pode ser obtida usando o Algoritmo de Robbins-Monro, mediante:

$$\theta_{i+1} = \theta_i - \gamma_i \hat{\xi}(\theta_i) \quad (\text{A.29})$$

onde γ_i é um escalar positivo e $\hat{\xi}(\theta)$ pode ser calculado pela Eq. (A.29), usando os dados coletados durante o experimento em malha fechada com $C(\theta, z^{-1})$. Para assegurar a convergência do algoritmo, a variável instrumental é selecionada como estimativa do gradiente da saída da planta com relação aos parâmetros do controlador. Esses parâmetros podem ser calculados usando um para modelo da planta. A exatidão do modelo não afeta o grau de convergência, mas afeta a taxa de convergência.

g) Virtual Reference Feedback Tuning (VRFT)

Proposto por Guardabassi e Savaresi (2000) a técnica DDC VRFT é usada para sintonizar o vetor de parâmetros de um controlador, com estrutura fixa, para um sistema LIT, mediante processo de identificação dos parâmetros do controlador, usando um sinal de referência virtual.

Os diagramas da Figura 34a e Figura 34b apresentam o esquema VRFT, a comparação do modelo de referência com o sistema real e o cálculo do sinal de referência virtual, respectivamente:

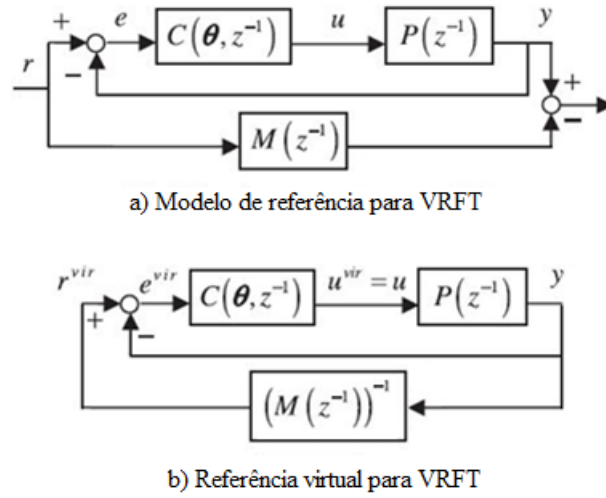


Figura 34 - Virtual Reference Feedback Tuning (VRFT) , adaptado de (HOU;WANG, 2013)

onde $P(z^{-1})$ é a equação da planta desconhecida, $C(\theta, z^{-1})$ é um controlador LIT, com vetor de parâmetros θ , $M(z^{-1})$ é o modelo de referência adotado no projeto de controle, e os sinais r , u , e y são a referência, sinal de controle e saída da planta, respectivamente. O objetivo do controle é aproximar o sinal de saída da planta, em malha fechada com o controlador, do sinal de saída do modelo de referência, ou seja, minimizar o seguinte índice de desempenho:

$$J(\theta) = \left\| \frac{C(\theta, z^{-1})P(z^{-1})}{1+C(\theta, z^{-1})P(z^{-1})}r - M(z^{-1})r \right\|^2 \quad (\text{A.30})$$

Uma vez que a equação $P(z^{-1})$ é desconhecida, a minimização de $J(\theta)$ não pode ser realizada diretamente. No entanto, o algoritmo VRFT propõe uma solução alternativa para esse problema. Seguindo o diagrama da Figura 34b, se o modelo de referência for inversível, é possível calcular um conjunto de dados virtuais $\{(e^{vir}(k), u^{vir}(k))\}_{k=1, \dots, N}$, para o controlador $C(\theta, z^{-1})$ usando o conjunto de dados E/S da planta $\{(u(k), y(k))\}_{k=1, \dots, N}$, mediante:

$$e^{vir}(k) = r^{vir}(k) - y(k) = M^{-1}(z^{-1})y(k) - y(k) \quad (\text{A.31})$$

e

$$u^{vir}(k) = u(k) \quad (\text{A.32})$$

onde $M^{-1}(z^{-1})$ é o inverso de um modelo de referência.

Na Eq. (A.31) o sinal r^{vir} é chamado de referência virtual, pois não existe, sendo calculado via computador com base nos dados $\{(u(k), y(k))_{k=1, \dots, N}\}$ coletados, mediante:

$$r^{vir} = M^{-1}(z^{-1})y. \quad (\text{A.33})$$

Usando o conjunto de dados virtuais calculados, os parâmetros do controlador podem ser identificados, mediante a minimização do seguinte índice de desempenho:

$$J_{VRFT}(\theta) = \|C(\theta, z^{-1})e^{vir} - u^{vir}\|^2 \quad (\text{A.34})$$

Apesar dos dois índices Eq. (A.30) e Eq. (A.34), definidos para identificação do controlador, serem diferentes. O Teorema 1, apresentado por Campi *et al.* (2002) garante que os mínimos de $J_{VRFT}(\theta)$ e $J(\theta)$ são iguais quando $J(\theta^*) = 0$. Nos casos não garantidos pelo Teorema, a Proposição 1 em (CAMPPI *et al.*, 2003) afirma que o projeto de um filtro adequado para os sinais virtuais, pode assegurar o mesmo ponto ótimo pelos dois critérios.

h) Noniterative Data-driven Model Reference Control

Proposto por Van Heusden (KARIMI *et al.*, 2007), trata-se de mais um método DDC para sistemas LIT, que busca ajustar o vetor de parâmetros de um controlador de estrutura fixa, aproximando o sinal de saída da planta e controlador em malha fechada do sinal de saída de um modelo de referência, quando submetidos ao mesmo sinal de referência. Para tanto, o seguinte critério de desempenho deve ser minimizado:

$$J(\theta) = \left\| M(z^{-1})r - \frac{C(\theta, z^{-1})P(z^{-1})}{1+C(\theta, z^{-1})P(z^{-1})}r \right\|^2 \quad (\text{A.35})$$

onde $P(z^{-1})$ é a equação da planta, $C(\theta, z^{-1})$ a equação do controlador, $M(z^{-1})$ é o modelo de referência, θ é o vetor de parâmetros e r é o sinal de referência.

Infelizmente, a condição $J(\theta) = 0$ normalmente não pode ser alcançada. O critério na Eq. (A.35) não é convexo com relação ao vetor de parâmetros θ do controlador. Entretanto, pode-se obter uma aproximação convexa para controladores linearmente parametrizáveis usando o modelo de referência $M(z^{-1})$. Na formulação do problema agora, por simplicidade, toda a notação será encurtada retirando o argumento z^{-1} do equacionamento, logo, a equação do modelo de referência passa ser escrita como:

$$M = \frac{C^*P}{1+C^*P} \quad (\text{A.36})$$

onde C^* é o controlador ideal, o qual pode ser definido indiretamente em função de P e M como:

$$C^* = \frac{M}{P(1-M)}. \quad (\text{A.37})$$

Observe que o controlador só existirá se $M \neq 1$. Usando a Eq. (A.37) para o controlador ideal é possível reescrever a Eq. (A.35), como:

$$J(\theta) = \left\| \frac{C^*P - C(\theta)P}{(1+C^*P)(1+C(\theta)P)} r \right\|^2 \quad (\text{A.38})$$

Agora, substituindo $(1 + C(\theta)P)$ por $(1 + C^*P)$ na Eq. (A.38) e usando a Eq. (A.37), obtém-se a seguinte aproximação:

$$\begin{aligned} \hat{J}(\theta) &= \left\| \frac{C^*P - C(\theta)P}{(1 + C^*P)^2} r \right\|^2 = \|(1 - M)[M - C(\theta)(1 - M)P]r\|^2 \\ &= \|(1 - M)Mr - C(\theta)(1 - M)^2Pr\|^2. \end{aligned} \quad (\text{A.39})$$

Se o controlador é linearmente parametrizado, então $\hat{J}(\theta)$ é convexo para os parâmetros do controlador e um ponto ótimo θ^* pode ser encontrado.

Considerando a situação em que existe ruído na medida do sinal de saída da planta, ou seja, $y(k) = P(z^{-1})u(k) + v(k)$, onde $u(k)$ é a entrada da planta, $v(k)$

é o ruído medido e a planta $P(z^{-1})$ é estável. A solução ótima pode ser encontrada minimizando a norma do seguinte sinal de erro:

$$\begin{aligned}\varepsilon_c(\theta) &= M(1-M)r - C(\theta)(1-M)^2y \text{ ou} \\ &= M(1-M)r - C(\theta)(1-M)^2Pr - C(\theta)(1-M)^2v\end{aligned}\quad (\text{A.40})$$

Os diagramas de blocos da Figura 35a e Figura 35b apresentam as duas versões da Eq. (A.40). Observe, no entanto, que o diagrama da Figura 35b o controlador real é substituído pelo controlador ideal:

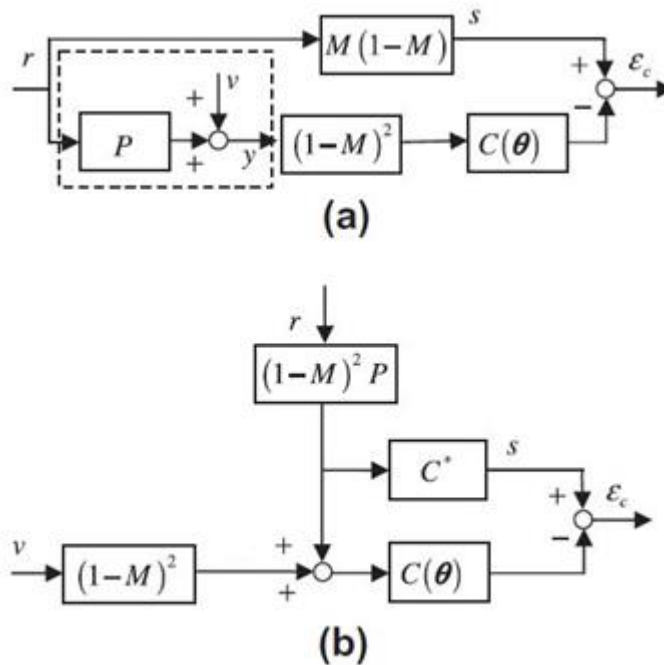


Figura 35 - Noniterative Data-driven Model Reference Control, adaptado de (HOU;WANG, 2013)

Nos dois diagramas a variável $s(k)$ pode ser calculada mediante a seguinte equação:

$$\begin{aligned}s(k) &= (1-M)^2PC^*r(k) \text{ ou} \\ &= M(1-M)r(k)\end{aligned}\quad (\text{A.41})$$

Finalmente, o problema de sintonia de parâmetros para o controlador torna-se, então, um problema de identificação de parâmetros. Ou seja, primeiro o

controlador ideal C^* é identificado, minimizando a norma do erro $\varepsilon_c(\theta)$, usando a primeira versão da Eq. (A.41) para calcular $s(k)$. Depois, o modelo do controlador C a ser acoplado ao sistema é identificado usando os sinais de erros calculados usando o controlador ideal.

i) *Subspace Approach - SA*

Para exemplificar o emprego da técnica SA, será abordada uma aproximação de controle preditivo por subespaço sem distúrbios (MCP *data-driven*). Nesse método, a matriz dinâmica é determinada pelo cálculo da pseudo-inversa de Moore-Penrose, a qual também é usada para elaboração de um previsor para a planta controlada.

A planta controlada via MCP *data-driven* é um sistema discreto LIT livre de ruído descrito pelo seguinte conjunto de equações:

$$x(k+1) = Ax(k) + Bu(k) \quad (\text{A.42})$$

$$y(k) = Cx(k) + Du(k) \quad (\text{A.43})$$

onde A , B , C e D são matrizes invariantes no tempo. Essas equações, considerando o conjunto de dados entrada/estado/saída, também podem ser expressas como:

$$\begin{bmatrix} y(k) \\ y(k+1) \\ \vdots \\ y(k+i) \end{bmatrix} = \begin{bmatrix} C \\ AC \\ \vdots \\ AC^i \end{bmatrix} x(k) + \begin{bmatrix} D & 0 & 0 & 0 \\ CB & D & 0 & 0 \\ \vdots & \dots & \ddots & 0 \\ C^{i-2}B & C^{i-1}B & \dots & 0 \end{bmatrix} u(k) \quad (\text{A.44})$$

A matriz de Hankel H_{ixj} de um sinal $w(k)$, é calculada mediante:

$$H_{ij}(w(k)) = \begin{bmatrix} w(k) & w(k+1) & \dots & w(k+j-1) \\ w(k+1) & w(k+2) & \dots & w(k+j) \\ w(k+2) & w(k+3) & \dots & w(k+j+1) \\ \vdots & \vdots & \ddots & \vdots \\ w(k+i-1) & w(k+i) & \dots & w(k+i+j-2) \end{bmatrix} \quad (\text{A.45})$$

Portanto, as matrizes de dados, calculadas usando a matriz de Hankel para os sinais de entrada e saída, podem ser montadas como segue:

$$U_p = H_{i,j}(u(1)), \quad U_f = H_{i,j}(u(i)), \quad Y_p = H_{i,j}(y(1)), \quad Y_f = H_{i,j}(y(i))$$

Usando a expressão para a saída do sistema, o previsor para o modelo da planta pode ser escrito como:

$$\hat{Y}_f = L_w W_p + L_u U_f \quad (\text{A.46})$$

onde $\hat{Y}_f$ é a saída do modelo previsor dirigido a dados, L_w e L_u são as matrizes dinâmicas, e $W_p = [Y_p^T \ U_p^T]$.

Se as matrizes dinâmicas estão disponíveis, o MCP *data-driven* pode ser aplicado. Portanto, usando a pseudo-inversa de Moore-Penrose e o previsor, Eq. (A.46), as matrizes dinâmicas podem ser calculadas mediante:

$$(L_w \ L_u) = Y_f \begin{pmatrix} W_p \\ U_f \end{pmatrix}^+ = Y_f (W_p^T \ U_f^T) \left(\begin{pmatrix} W_p \\ U_f \end{pmatrix} (W_p \ U_f) \right)^{-1} \quad (\text{A.47})$$

Para o projeto do controlador preditivo, a minimização do seguinte índice de performance quadrático pode ser empregado:

$$J = \sum_{i=1}^{N_y} \|r_{i+1} - \hat{y}_f\|_Q^2 + \sum_{i=1}^{N_y} \|u_{i+1}\|_R^2. \quad (\text{A.48})$$

Otimizar esse índice de desempenho significa produzir via MCP *data-driven* o seguinte sinal de controle:

$$u_f = \arg \min_{u_f} \left\{ (r_f - \hat{y}_f)^T Q (r_f - \hat{y}_f) + u_f^T R u_f \right\} \quad (\text{A.49})$$

Como exemplo, igualando as matrizes $Q=R=I$ (matriz identidade), e usando Eq. (A.46) na Eq. (A.49), resulta no seguinte controlador preditivo de subespaço dirigido a dados:

$$u_f = (\lambda I + L_u^T L_u)^{-1} L_u^T (r_f - L_w W_p) \quad (\text{A.50})$$

Observe que nesse procedimento nenhuma informação explícita do modelo foi incluída. Portanto, a estrutura do modelo é implicitamente envolvida no controlador. Entretanto, sob o ponto de vista teórico, a condição de excitação persistente é outra suposição implícita pois a inversa da matriz é incluída no cálculo do sinal de controle.

j) *Approximate Dynamic Programming (ADP)*

Para o método DDC ADP a técnica de *Q-learning* será aqui tratada em detalhe pois sua aplicação não requer conhecimento do modelo da planta. Originalmente o *Q-learning* foi proposto como solução para processos de decisão de Markov discretos (MDPs) onde o número de pares ação-estado são finitos e o modelo MDP não está disponível.

Considere o seguinte processo determinístico de Markov:

$$x(k+1) = f(x(k), u(k)), \quad (\text{A.51})$$

onde $x(k) \in \mathcal{S}, \forall k \in \mathcal{N}$, é o estado no instante k e $\mathcal{S}$ um conjunto finito de estados; $u(k) \in \mathcal{A}, \forall k \in \mathcal{N}$, é a ação e $\mathcal{A}$ é um conjunto finito de ações; e $f(\cdot)$ é uma função desconhecida. O objetivo do controle é encontrar uma política ótima π^* que minimize a seguinte função de custo.

$$J(x(0), \pi) = \sum_{t=0}^{\infty} \gamma^t r(x(t), u(t)), \quad (\text{A.52})$$

sendo $\gamma \in [0,1]$ é um fator de desconto, $r(\cdot)$ é uma função de custo de estágio simples, e $u(t) = \pi(x(t)), t = 0, \dots, \infty$.

Se $f(\cdot)$ é conhecida, a técnica de programação dinâmica (DP) é uma aproximação geral para resolver esse problema de otimização. O objetivo da DP é obter uma função *cost-to-go*, que pode ser definida como:

$$J^*(x(k)) = \min_{\pi} (\sum_{t=k}^{\infty} \gamma^t r(x(t), \pi(x(t)))) = \sum_{t=k}^{\infty} \gamma^t r(x(t), \pi^*(x(t))), \quad (\text{A.53})$$

onde π^* é a política ótima. Tal que, $J^*(x(0))$ também seja uma solução ótima para a função de custo. Nesse caso, a Eq. (A.53) pode ser reescrita como:

$$J^*(x(k)) = \min_{u(k)} (r(x(k), u(k)) + \gamma J^*(x(k+1))) \quad (\text{A.54})$$

A Eq. (A.54) é conhecida como equação de Bellman, portanto, uma vez conhecidas as funções $f(\cdot)$ e $J^*(\cdot)$, é possível resolver o seguinte problema de decisão de estágio simples *on-line*:

$$\pi^*(x(k)) = \arg \min_{u(k)} (r(x(k), u(k)) + \gamma J^*(f(x(k), u(k)))) \quad (\text{A.55})$$

Nos casos em que a função $f(\cdot)$ é desconhecida, obter $J^*(\cdot)$ não resolve o problema de escolha das ações ótimas, pois a função *cost-to-go* reescrita agora não pode ser resolvida. A solução, nesse caso, pode ser definir uma nova função *cost-to-go* para o problema. Para tanto, a Q-função, pode ser definida como segue:

$$Q(x(k), u(k)) = r(x(k), u(k)) + \gamma J^*(f(x(k), u(k))) \quad (\text{A.56})$$

observe que a função $Q(x(k), u(k))$ é exatamente a quantidade a ser minimizada para solução da equação de Bellman, de forma que a ação ótima $\pi^*(x(k))$ no estado $x(k)$ também pode ser encontrada. Portanto, é possível reescrever a solução do problema de otimização agora em termos da Q-função, como:

$$\pi^*(x(k)) = \arg \min_{u(k)} Q(x(k), u(k)), \quad (\text{A.57})$$

ou seja, usando $Q(\cdot)$ no lugar de $J^*(\cdot)$, é possível selecionar ações ótimas de controle, mesmo sem o conhecimento da função $f(\cdot)$. Além disso, como:

$$J^*(x(k)) = \min_{u(k)} Q(x(k), u(k)), \quad (\text{A.58})$$

é possível reescrever a Q-função como:

$$Q(x(k), u(k)) = r(x(k), u(k)) + \gamma \min_{u'} Q(f(x(k), u(k), u')). \quad (\text{A.59})$$

Esta definição recursiva da função $Q(\cdot)$, é a base para um algoritmo *off-line* que iterativamente aproxima $Q(\cdot)$, ou seja:

$$\hat{Q}_{i+1}(x(k), u(k)) = (1 - \alpha) \hat{Q}_i(x(k), u(k)) + \alpha (r(x(k), u(k)) + \gamma \min_{u'} \hat{Q}_i(x(k+1), u')) \quad (\text{A.60})$$

onde $\hat{Q}_i(\cdot)$ é a aproximação de $Q(\cdot)$ na i -ésima iteração e α é o parâmetro de taxa de aprendizagem definido entre 0 e 1. Usando esse algoritmo $\hat{Q}_i(\cdot)$ converge para $Q(\cdot)$.

j) Iterative Learning Control (ILC)

A ILC tem uma estrutura muito simples e é ideal para sistemas que repetem a mesma tarefa diversas vezes num intervalo de tempo finito. Credita-se que o

controlador ILC consegue aprender com as dinâmicas repetitivas para obter uma melhor performance.

A Figura 36 apresenta o esquema de um sistema ILC. No diagrama dois componentes de memória são usados para gravar os sinais de controle e de saída anteriores ao par de sinais de controle e de saída atuais.

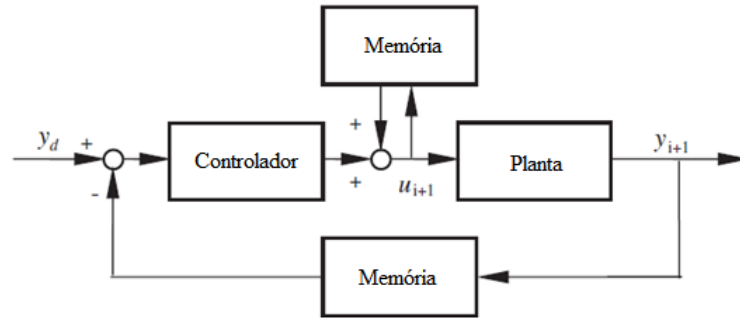


Figura 36 - Iterative Learning Control, adaptado de (HOU;WANG, 2013)

Considere o seguinte sistema dinâmico:

$$\mathbf{x}_i(k+1) = \mathbf{f}(\mathbf{x}_i(k), \mathbf{u}_i(k), k) \quad (\text{A.61})$$

$$\mathbf{y}_i(k) = \mathbf{g}(\mathbf{x}_i(k), \mathbf{u}_i(k), k), \quad (\text{A.62})$$

onde $\mathbf{f}$ e $\mathbf{g}$ são funções globais contínuas de Lipshitz dos argumentos $\mathbf{x}_i$ e $\mathbf{u}_i$; $\mathbf{x}_i(k) \in \mathbf{R}^n$, $\mathbf{y}_i(k) \in \mathbf{R}^m$ e $\mathbf{u}_i(k) \in \mathbf{R}^r$, são o estado, a saída da planta, e o sinal de controle, respectivamente; $k \in \{0, 1, \dots, T\}$ denota um ponto específico no tempo; e $i \in \{0, 1, 2, \dots\}$ denota o número de iterações.

A tarefa do controlador é dirigir a saída da planta $\mathbf{y}_i(k)$ para que ela siga a saída desejada $\mathbf{y}_d(k)$, num intervalo de tempo fixo $k \in [0, T]$, para qualquer instante k , quando a iteração i tende ao infinito. Em outras palavras, que o sinal de erro definido como: $\mathbf{e}_i(k) = \mathbf{y}_d(k) - \mathbf{y}_i(k)$, $\forall k \in [0, T]$, uniformemente convirja para zero quando $i \rightarrow \infty$.

O diagrama esquemático da técnica de controle ILC é mostrado na Figura 37. Observe que o sinal de controle $\mathbf{u}_i(k)$, no instante k da i -ésima iteração, será função de todos os sinais de controle antes do instante k da i -ésima iteração, e dos sinais de controle em todas os instantes das N iterações antes da iteração i , do sinal

de erro para o instante k e para todos os instantes os anteriores da i -ésima iteração, e dos sinais de erro de todos os instantes das N iterações antes da i -ésima iteração.

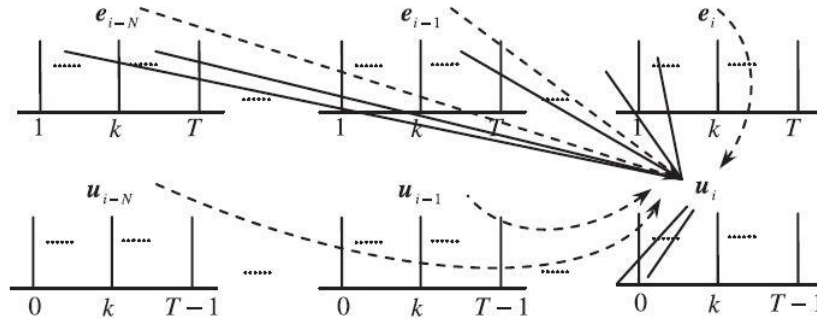


Figura 37 – Cálculo do sinal de controle, adaptado de (HOU;WANG, 2013)

Portanto a lei de aprendizado por iteração pode ser expressa como segue:

$$\mathbf{u}_i(k) = h(\mathbf{u}_i(< k), \mathbf{u}_{i-1}(\cdot), \dots, \mathbf{u}_{i-N}(\cdot), \mathbf{e}_i(\leq k), \mathbf{e}_{i-1}(\cdot), \dots, \mathbf{e}_{i-N}(\cdot)) \quad (\text{A.63})$$

Leis de aprendizagem do tipo P (Proporcional), leis de aprendizagem do tipo D (Derivativas) e leis de aprendizagem do tipo PID (Proporcional Integral e Derivativa), leis de aprendizagem de ordem mais alta, leis de aprendizagem robusta, leis de aprendizagem ótima, e leis de aprendizado realimentado para frente são casos especiais do controle ILC.

l) *Lazy Learning (LL).*

LL são algoritmos de aprendizagem supervisionada, os quais foram aplicados a problemas de controle inicialmente por Shaal e Atkeson em 1994. O objetivo principal desses algoritmos é determinar as relações entre uma entrada e uma saída a partir de uma coleção de entradas e saídas, denominada conjunto de treinamento.

Como exemplo considere uma função não linear $y = f(\phi)$, onde $f: \mathbf{R}^n \rightarrow \mathbf{R}$, $\phi \in \mathbf{R}^n$, e $y \in \mathbf{R}$ e uma coleção de valores de entrada e saída, $\{(\phi_i, y_i)_{i=1, \dots, N}\}$ chamado conjunto de treinamento. Para estimar a saída $y_q \in \mathbf{R}$ para um ponto de pesquisa $\phi_q \in \mathbf{R}^n$, usando o LL, serão necessários três passos:

- i. Passo 1: Regressão linear ponderada usando a geração de um modelo local.

Um modelo local seria uma função linear, do tipo $y = [\phi^T, 1] \cdot \theta$, onde $\theta \in \mathbf{R}^{n+1}$ é um vetor de parâmetros. Para um dado h , uma regressão linear ponderada local é usada para achar a solução ótima $\theta^*(h)$ usando o seguinte critério.

$$J(\theta, h) = \sum_{i=1}^N \left\{ (y_i - [\phi^T, 1] \cdot \theta)^2 \cdot K\left(\frac{D(\phi_i, \phi_q)}{h}\right) \right\} \quad (\text{A.64})$$

onde $D(\phi_i, \phi_q)$ é a função da distância (por exemplo distância euclidiana entre ϕ_i e ϕ_q) e h a largura de banda da função de ponderação $K(\cdot)$, selecionada como segue:

$$K(x) = \begin{cases} 1, & x \leq 1, \\ 0, & x > 1 \end{cases} \quad (\text{A.65})$$

Selecionando diferentes valores para h pode-se montar um conjunto de M candidatos modelo local $\{(\theta^*(h_i))_{i=1, \dots, M}\}$.

- ii. Passo 2: Validação do modelo local.

O critério $J(\theta, h)$ é usado para avaliar cada um dos candidatos do conjunto $\{(\theta^*(h_i))_{i=1, \dots, M}\}$. Entretanto, a validação é tendenciosa pois usa os mesmo dados do conjunto $\{(\phi_i, y_i)_{i=1, \dots, N}\}$ como identificação.

- iii. Passo 3: Seleção do modelo local e estimação da saída.

O modelo local ótimo é obtido mediante:

$$\theta^*(h^*) = \arg \min_{\theta \in \{(\theta^*(h_i))_{i=1, \dots, M}\}} J(\theta, h). \quad (\text{A.66})$$

Finalmente, a estimação da saída no ponto de pesquisa ϕ_q é obtida usando:

$$\hat{y}_q = [\phi_q^T, 1] \cdot \theta^*(h^*) \quad (\text{A.67})$$

Depois de determinar $\hat{y}_q$, o algoritmo LL descarta o modelo ótimo local. Para um novo ponto de pesquisa, os três passos acima são repetidos. Observe também que o LL estima os valores da saída, mas não as funções geradoras. O modelo local

apenas descreve o comportamento do sistema nas proximidades do ponto de pesquisa.

Como o modelo local no LL é selecionado como um modelo linear, sistemas originalmente não lineares tornam-se mais fáceis de trabalhar.

O controle de um sistema usando o método LL aplica a estratégia de dividir para conquistar, ou seja, para cada modelo linear gerado em cada instante da dinâmica do sistema é também projetado um controlador linear para o modelo. Por exemplo, considere a Eq. (A.68) da planta não-linear:

$$y(k+1) = f(y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)) \quad (\text{A.68})$$

onde $y(k) \in \mathbf{R}$ é a saída no instante k , $u(k) \in \mathbf{R}$ é o sinal de controle no instante k , n_y e n_u são as ordens conhecidas da planta e $f: \mathbf{R}^{n_y+n_u} \rightarrow \mathbf{R}$ é uma função não linear desconhecida. Usando como ponto de pesquisa:

$$\phi_q(k) = [y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)]^T \quad (\text{A.69})$$

e o algoritmo LL, um modelo dinâmico local pode ser construído como segue:

$$y(k+1) = [y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u), 1] \cdot \theta^*(h^*) \quad (\text{A.70})$$

para o instante k , onde $\theta^*(h^*) \in \mathbf{R}^{n_y+n_u+1}$. O elemento $u(k)$ no cálculo da distância $D(\phi_i, \phi_q)$, no entanto, deve ser ignorado pois o valor dele não está disponível para ϕ_q e tendo em vista que esse valor é um dos objetivos da técnica de controle. Depois de obter o modelo linear para o sistema, é possível usar técnicas de controle como alocação de pólos ou variância mínima para projetar um controlador local o qual gerará o sinal de controle $u(k)$ no instante k .

Apêndice B - Modelos Lineares Dinâmicos da Planta Usando MFAC

A seguir será abordado os demais modelos de linearização dinâmica usados na teoria MFAC, não descritos no Capítulo 2, para sistemas não-lineares SISO e MIMO (HOU; JIN, 2014).

Um sistema SISO não-linear discreto no tempo pode ser descrito pela Eq. (B.1)

$$y(k+1) = f\left(y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)\right), \quad (\text{B.1})$$

a) Modelo Dinâmico Linear da Forma Parcial para sistemas SISO

Defina $\mathbf{U}_L(k) \in R^L$ como um vetor consistindo de todos os sinais de controle dentro da janela de tempo móvel no intervalo $[k-L+1, k]$, ou seja,

$$\mathbf{U}_L(k) = [u(k), \dots, u(k-L+1)]^T \quad (\text{B.2})$$

onde $\mathbf{U}_L(k) = \mathbf{0}_L$, para $k \leq 0$, e o inteiro L é chamado de constante de comprimento de Linearização – LLC, da entrada e $\mathbf{0}_L$ é o vetor nulo de dimensão L .

SUPosição B.1

As derivadas parciais de $f(\dots)$ na Eq. (B.1) em relação às variáveis entre $(n_y + 2)$ -ésima e a $(n_y + L + 1)$ -ésima variável são contínuas.

SUPosição B.2

O sistema representado pela Eq. (B.1) satisfaz a condição generalizada de Lipschitz, isto é,

$$|y(k_1 + 1) - y(k_2 + 1)| \leq b \|\mathbf{U}_L(k_1) - \mathbf{U}_L(k_2)\| \quad (\text{B.3})$$

para $\mathbf{U}_L(k_1) \neq \mathbf{U}_L(k_2)$ e qualquer $k_1 \neq k_2$, $k_1, k_2 > 0$, onde $y(k_i + 1) = f(y(k_i), \dots, y(k_i - n_y), u(k_i), \dots, u(k_i - n_u))$, $i = 1, 2$ e b é uma constante positiva.

Defina também, $\Delta \mathbf{U}_L(k) = \mathbf{U}_L(k) - \mathbf{U}_L(k - 1)$. A seguir, o Teorema B.1 apresentará o modelo PFDL da Eq. (3.1).

TEOREMA B.1

Considere um sistema não-linear satisfazendo as Suposições B.1 e B.2. Para qualquer L fixo, se $\|\Delta \mathbf{U}_L(k)\| \neq 0$, então deve existir um vetor variante no tempo $\boldsymbol{\phi}_{p,L}(k) \in R^L$, denominado vetor de Pseudo-derivadas parciais – VPPD ou pseudo gradiente – PG, tal que a Eq. (B.1) pode ser transformado no seguinte modelo Linear Dinâmico da Forma Parcial – PFDL:

$$\Delta y(k + 1) = \boldsymbol{\phi}_{p,L}^T(k) \Delta \mathbf{U}_L(k) \quad (\text{B.4})$$

com $\boldsymbol{\phi}_{p,L}(k) = [\phi_1, \dots, \phi_L]^T$ limitado para todo k , a prova em (HOU, JIN, 2014).

b) Modelo Dinâmico Linear da Forma Completa para sistemas SISO

Defina $\mathbf{H}_{L_y, L_u}(k) \in R^{L_y + L_u}$ como um vetor consistindo de todos os sinais de controle, dentro da janela de tempo móvel de sinais de controle no intervalo $[k - L_u + 1, k]$ e todos os sinais de saída dentro da janela de tempo móvel de sinais de saída no intervalo $[k - L_y + 1, k]$

$$\mathbf{H}_{L_y, L_u}(k) = [y(k), \dots, y(k - L_y + 1), u(k), \dots, u(k - L_u + 1)]^T \quad (\text{B.5})$$

onde $\mathbf{H}_{L_y, L_u}(k) = \mathbf{0}_{L_y + L_u}$, para $k \leq 0$, e os inteiros L_y e L_u ($0 \leq L_y \leq n_y$, $0 \leq L_u \leq n_u$) são chamados de pseudo-ordens do sistema ou constantes de comprimento de

Linearização – LLCs, de saída controlada e de entrada de controle, respectivamente, similarmente a constante L do modelo PFDL.

SUPosição B.3

As derivadas parciais de $f(\dots)$ na Eq. (B.1) em relação a todas as variáveis são contínuas.

SUPosição B.4

O sistema representado pela Eq. (B.1) satisfaz a condição generalizada de Lipschitz, isto é,

$$|y(k_1 + 1) - y(k_2 + 1)| \leq b \left\| \mathbf{H}_{L_y, L_u}(k_1) - \mathbf{H}_{L_y, L_u}(k_2) \right\| \quad (\text{B.6})$$

para $\mathbf{H}_{L_y, L_u}(k_1) \neq \mathbf{H}_{L_y, L_u}(k_2)$ e qualquer $k_1 \neq k_2$, $k_1, k_2 > 0$, onde $y(k_i + 1) = f(y(k_i), \dots, y(k_i - n_y), u(k_i), \dots, u(k_i - n_u))$, $i = 1, 2$ e b é uma constante positiva.

Defina também $\Delta \mathbf{H}_{L_y, L_u}(k) = \mathbf{H}_{L_y, L_u}(k) - \mathbf{H}_{L_y, L_u}(k - 1)$. O Teorema B.2 a seguir apresentará o modelo FFDL da Eq. (B.1).

TEOREMA B.2

Considere um sistema não-linear satisfazendo as suposições B.3 e B.4. Para quaisquer $0 \leq L_y \leq n_y$, $0 \leq L_u \leq n_u$ fixos, se $\left\| \Delta \mathbf{H}_{L_y, L_u}(k) \right\| \neq 0$, então deve existir um vetor variante no tempo $\boldsymbol{\phi}_{f, L_y, L_u}(k) \in R^{L_y + L_u}$, denominado vetor de pseudo-derivadas parciais – VPPD ou pseudo gradiente – PG, tal que o sistema (B.1) possa ser transformado no seguinte modelo Linear Dinâmico da Forma Completa – FFDL:

$$\Delta y(k + 1) = \boldsymbol{\phi}_{f, L_y, L_u}(k) \Delta \mathbf{H}_{L_y, L_u}(k) \quad (\text{B.7})$$

com $\boldsymbol{\phi}_{f, L_y, L_u}(k) = [\phi_1, \dots, \phi_{L_y}, \phi_{L_y+1}, \dots, \phi_{L_y+L_u}]^T$ limitado, para todo k , a prova está em (HOU; JIN, 2014).

O método de linearização dinâmica apresentado pode ser estendido a sistemas não-lineares discretos no tempo do tipo MIMO descritos usando a Eq. (B.8), a seguir:

$$\mathbf{y}(k+1) = \mathbf{f}(\mathbf{y}(k), \dots, \mathbf{y}(k-n_y), \mathbf{u}(k), \dots, \mathbf{u}(k-n_u)), \quad (\text{B.8})$$

onde, e $\mathbf{u}(k) \in R^m$ e $\mathbf{y}(k) \in R^m$ são os vetores das entradas e saídas da planta, respectivamente, no instante k , e n_u e n_y são as ordens desconhecidas da entrada e da saída e $\mathbf{f}(\dots) = [f_1(\dots), \dots, f_m(\dots)]^T \in \prod_{n_u+n_y+2} R^m \rightarrow R^m$ é uma função não-linear desconhecida.

c) *Modelo Dinâmico Linear da Forma Compacta para sistemas MIMO*

Novamente, antes de apresentar o modelo é necessário fazer algumas suposições:

SUPosição B.5

As derivadas parcial de $f_i(\dots), i = 1, \dots, m$ em relação a cada entrada da $(n_y + 2)$ -ésima variável $u(k)$ é contínua.

SUPosição B.6

O sistema representado pela Eq. (B.8) satisfaz a condição generalizada de Lipschitz, para todo k com finitas exceções, isto é,

$$\|\mathbf{y}(k_1+1) - \mathbf{y}(k_2+1)\| \leq b \|\mathbf{u}(k_1) - \mathbf{u}(k_2)\| \quad (\text{B.9})$$

para $\mathbf{u}(k_1) \neq \mathbf{u}(k_2)$ e qualquer $k_1 \neq k_2$, $k_1, k_2 > 0$, onde $\mathbf{y}(k_i+1) = \mathbf{f}(\mathbf{y}(k_i), \dots, \mathbf{y}(k_i-n_y), \mathbf{u}(k_i), \dots, \mathbf{u}(k_i-n_u))$ $i = 1, 2$ e b é uma constante positiva.

Para o equacionamento do modelo linear dinâmico também serão definidos os seguintes vetores, $\Delta \mathbf{u}(k) = \mathbf{u}(k) - \mathbf{u}(k-1)$ e $\Delta \mathbf{y}(k+1) = \mathbf{y}(k+1) - \mathbf{y}(k)$. A seguir, o Teorema B.3 apresentará o modelo CFDL da Eq. (B.8).

TEOREMA B.3

Considere um sistema não-linear satisfazendo as Suposições B.5 e B.6. Se $\|\Delta \mathbf{u}(k)\| \neq 0$, então deve existir uma matriz variante no tempo $\Phi_c(k) \in R^{m \times m}$, denominada de matriz de pseudo-derivadas parciais – MPPD ou matriz pseudo jacobiana – MPJ, tal que a Eq. (B8) possa ser transformado no seguinte modelo linear dinâmico de forma completa – CFDL:

$$\Delta \mathbf{y}(k+1) = \Phi_c(k) \Delta \mathbf{u}(k) \quad (\text{B.10})$$

com $\Phi_c(k)$ limitada para todo k , a prova é análoga a do Teorema 3.1.

d) Modelo Dinâmico Linear da Forma Parcial para sistemas MIMO

Defina $\bar{\mathbf{U}}_L(k) \in R^{mL}$ como um vetor consistindo de todos os sinais de controle dentro da janela de tempo móvel no intervalo $[k-L+1, k]$, ou seja,

$$\bar{\mathbf{U}}_L(k) = [\mathbf{u}^T(k), \dots, \mathbf{u}^T(k-L+1)]^T \quad (\text{B.11})$$

onde $\bar{\mathbf{U}}_L(k) = \mathbf{0}_{mL}$, para $k \leq 0$, e o inteiro L a LLC da entrada.

SUPosição B.7

As derivadas parciais de $f_i(\dots)$, $i = 1, \dots, m$, em relação as variáveis entre $(n_y + 2)$ -ésima e a $(n_y + L + 1)$ -ésima variável, a saber $\mathbf{u}(k), \dots, \mathbf{u}(k-L+1)$, são contínuas.

SUPosição B.8

O sistema representado pela Eq. (B.8) satisfaz a condição generalizada de Lipschitz, isto é,

$$\|\mathbf{y}(k_1+1) - \mathbf{y}(k_2+1)\| \leq b \|\bar{\mathbf{U}}_L(k_1) - \bar{\mathbf{U}}_L(k_2)\| \quad (\text{B.12})$$

para $\bar{\mathbf{U}}_L(k_1) \neq \bar{\mathbf{U}}_L(k_2)$ e qualquer $k_1 \neq k_2$, $k_1, k_2 > 0$, onde $\mathbf{y}(k_i+1) = \mathbf{f}(\mathbf{y}(k_i), \dots, \mathbf{y}(k_i - n_y), \mathbf{u}(k_i), \dots, \mathbf{u}(k_i - n_u))$, $i = 1, 2$ e b é uma constante positiva.

Defina também, $\Delta \bar{\mathbf{U}}_L(k) = \bar{\mathbf{U}}_L(k) - \bar{\mathbf{U}}_L(k-1)$. A seguir, o Teorema B.4 apresentará o modelo PFDL da Eq. (B.8).

TEOREMA B.4

Considere um sistema não-linear satisfazendo as suposições B.7 e B.8. Para qualquer L fixo, se $\|\Delta \bar{\mathbf{U}}_L(k)\| \neq 0$, então deve existir uma matriz variante no tempo $\Phi_{p,L}(k) \in R^{m \times mL}$, denominado matriz de pseudo-derivadas parciais – MPPD ou matriz pseudo-jacobiana particionada – PPJM, tal que a Eq. (B.8) possa ser transformado no seguinte modelo linear dinâmico da forma parcial – PFDL:

$$\Delta \mathbf{y}(k+1) = \Phi_{p,L}(k) \Delta \bar{\mathbf{U}}_L(k) \quad (\text{B.13})$$

Com $\Phi_{p,L}(k) = [\Phi_1(k) \dots \Phi_L(k)]$ limitada para todo k , onde $\Phi_i(k) \in R^{m \times m}$, $i = 1, \dots, L$, a prova é análoga a do Teorema B.1.

e) Modelo Dinâmico Linear da Forma Completa para sistemas MIMO

Defina $\bar{\mathbf{H}}_{L_y, L_u}(k) \in R^{m_{L_y} + m_{L_u}}$ como um vetor consistindo de todos os sinais de entrada dentro da janela de tempo móvel de sinais de entrada no intervalo $[k - L_u + 1, k]$ e todos os sinais de saída dentro da janela de tempo móvel de sinais de saída no intervalo $[k - L_y + 1, k]$

$$\bar{\mathbf{H}}_{L_y, L_u}(k) = [\mathbf{y}^T(k), \dots, \mathbf{y}^T(k - L_y + 1), \mathbf{u}^T(k), \dots, \mathbf{u}^T(k - L_u + 1)]^T \quad (\text{B.14})$$

onde $\bar{\mathbf{H}}_{L_y, L_u}(k) = \mathbf{0}_{m_{L_y} + m_{L_u}}$, para $k \leq 0$, e os inteiros L_y e L_u ($0 \leq L_y \leq n_y$, $0 \leq L_u \leq n_u$) são chamados de pseudo-ordens do sistema ou , LLC de saída controlada e LLC de entrada de controle, respectivamente.

SUPosição B.9

As derivadas parciais de $f_i(\dots)$, $i = 1, \dots, m$ em relação a todas as variáveis são contínuas.

SUPosição B.10

O sistema representado pela Eq. (B.8) satisfaz a condição generalizada de Lipschitz, isto é,

$$|\mathbf{y}(k_1 + 1) - \mathbf{y}(k_2 + 1)| \leq b \left\| \bar{\mathbf{H}}_{L_y, L_u}(k_1) - \bar{\mathbf{H}}_{L_y, L_u}(k_2) \right\| \quad (\text{B.15})$$

para $\bar{\mathbf{H}}_{L_y, L_u}(k_1) \neq \bar{\mathbf{H}}_{L_y, L_u}(k_2)$ e qualquer $k_1 \neq k_2$, $k_1, k_2 > 0$, onde $\mathbf{y}(k_i + 1) = \mathbf{f}(\mathbf{y}(k_i), \dots, \mathbf{y}(k_i - n_y), \mathbf{u}(k_i), \dots, \mathbf{u}(k_i - n_u))$, $i = 1, 2$ e b é uma constante positiva.

Defina também $\Delta \bar{\mathbf{H}}_{L_y, L_u}(k) = \bar{\mathbf{H}}_{L_y, L_u}(k) - \bar{\mathbf{H}}_{L_y, L_u}(k - 1)$. Então o Teorema B.5 a seguir apresentará o modelo FFDL da Eq. (B.8).

TEOREMA B.5

Considere um sistema não-linear satisfazendo as Suposições B.9 e B.10. Para quaisquer $0 \leq L_y \leq n_y, 0 \leq L_u \leq n_u$ fixos, se $\left\| \Delta \bar{\mathbf{H}}_{L_y, L_u}(k) \right\| \neq 0$, então deve existir uma matriz variante no tempo $\Phi_{f, L_y, L_u}(k) \in R^{m \times (m L_y + m L_u)}$, denominada matriz de pseudo-derivadas parciais – MPPD ou matriz pseudo-Jacobiana particionada – PPJM, tal que a Eq. (B.8) pode ser transformado no seguinte modelo Linear Dinâmico da Forma Completa – FFDL:

$$\Delta \mathbf{y}(k + 1) = \Phi_{f, L_y, L_u}(k) \Delta \bar{\mathbf{H}}_{L_y, L_u}(k) \quad (\text{B.16})$$

Com $\Phi_{f, L_y, L_u}(k) = [\Phi_1(k) \dots \Phi_{L_y + L_u}(k)]$ limitada, para todo k , onde $\Phi_i(k) \in R^{m \times m}$, $i = 1, \dots, L_y + L_u$, a prova é análoga a do Teorema B.2.

Apêndice C - Projeto do Controlador MFAC para Sistemas Não-Lineares

A seguir será apresentado as etapas necessárias ao projeto do controlador MFAC (HOU;JIN,2014) usando os modelos lineares apresentados no Apêndice B:

a) Projeto de um PFDL-MFAC - SISO

O Teorema B.1 garante que um sistema SISO não linear, atendidas as suposições 3.3 e 3.4 e $\|\Delta \mathbf{U}_L(k)\| \neq 0, \forall k$, pode ser transformado no seguinte modelo PFDL:

$$\Delta y(k+1) = \boldsymbol{\phi}_{p,L}^T(k) \Delta \mathbf{U}_L(k) \quad (\text{C.1})$$

onde $\boldsymbol{\phi}_{p,L}(k) = [\phi_1, \dots, \phi_L]^T \in R^L$ é o PG limitado, $\Delta \mathbf{U}_L(k) = [\Delta u(k), \dots, \Delta u(k-L+1)]^T$ e L é o LLC do sinal de controle.

Para o controle do sistema, procura-se minimiza o seguinte índice de desempenho:

$$J(u(k)) = |y_d(k+1) - y(k+1)|^2 + \lambda |u(k) - u(k-1)|^2, \quad (\text{C.2})$$

Substituindo a Eq. (C.1), na Eq. (C.2), seguida da minimização do índice de desempenho em relação a $u(k)$, obtém-se o seguinte algoritmo de controle:

$$u(k) = u(k-1) + \frac{\rho_1 \phi_1(k)}{\lambda + |\phi_1(k)|^2} (y_d(k+1) - y(k)) - \frac{\phi_1(k)}{\lambda + |\phi_1(k)|^2} \sum_{i=2}^L \rho_i \phi_i(k) \Delta u(k-i+1), \quad (\text{C.3})$$

onde, o fator $\rho_i \in (0,1], i = 1, \dots, L$ foi adicionado para tornar o algoritmo mais flexível.

Para a estimação do PG, adotou-se a seguinte função de custo:

$$J(\boldsymbol{\phi}_{p,L}(k)) = |y(k) - y(k-1) - \boldsymbol{\phi}_{p,L}^T(k)\Delta\mathbf{U}_L(k-1)|^2 + \mu \|\boldsymbol{\phi}_{p,L}(k) - \hat{\boldsymbol{\phi}}_{p,L}(k-1)\|^2 \quad (\text{C.4})$$

onde $\mu > 0$ é um fator de ponderação.

Minimizar a Eq. (C.4) em relação a $\boldsymbol{\phi}_{p,L}(k)$ produz o seguinte algoritmo de estimação da PPD:

$$\hat{\boldsymbol{\phi}}_{p,L}(k) = \hat{\boldsymbol{\phi}}_{p,L}(k-1) + \frac{\eta \Delta\mathbf{U}_L(k-1)}{\mu + \|\Delta\mathbf{U}_L(k-1)\|^2} (\Delta y(k) - \hat{\boldsymbol{\phi}}_{p,L}^T(k-1)\Delta\mathbf{U}_L(k-1)), \quad (\text{C.5})$$

sendo que, o fator $\eta \in (0,2]$ foi adicionado para tornar o algoritmo mais flexível e $\hat{\boldsymbol{\phi}}_{p,L}(k)$ é a estimação do PG.

Para garantir que as condições exigidas pelo Teorema B.1 e pelas Suposições B.1 e B.2 permaneçam válidas, durante todo o período de atuação do controlador, um mecanismo de reiniciação deve ser incorporado ao algoritmo de estimação do PG, a saber:

$$\hat{\boldsymbol{\phi}}_{p,L}(k) = \hat{\boldsymbol{\phi}}_{p,L}(1) \text{ se } \|\hat{\boldsymbol{\phi}}_{p,L}(k)\| \leq \varepsilon \text{ ou } \|\Delta\mathbf{U}_{p,L}(k-1)\| \leq \varepsilon \text{ ou } \text{sign}(\hat{\phi}_1(k)) \neq \text{sign}(\hat{\phi}_1(1)) \quad (\text{C.6})$$

b) Projeto de um FFDL-MFAC - SISO

O Teorema B.2 garante que um sistema SISO não linear, atendidas as Suposições B.3 e B.4 e $\|\Delta\mathbf{H}_{L_y, L_u}(k)\| \neq 0, \forall k$, pode ser transformado no seguinte modelo FFDL:

$$\Delta y(k+1) = \boldsymbol{\phi}_{f, L_y, L_u}^T(k) \Delta\mathbf{H}_{L_y, L_u}(k) \quad (\text{C.7})$$

onde $\boldsymbol{\phi}_{f, L_y, L_u}(k) = [\phi_1, \dots, \phi_{L_y+L_u}]^T \in R^{L_y+L_u}$ é o PG limitado, $\Delta\mathbf{H}_{L_y, L_u}(k) = [\Delta y(k), \dots, \Delta y(k-L_y+1), \Delta u(k), \dots, \Delta u(k-L_u+1)]^T$ e L_y e L_u são chamadas pseudo ordens do sistema.

Para o controle do sistema, inicialmente substitui-se a Eq. (C.7), na Eq. (C.2). A seguir, minimiza a Eq. (C.2) em relação a $u(k)$, resultando no seguinte algoritmo de controle:

$$\begin{aligned}
 u(k) = u(k-1) &+ \frac{\rho_1 \phi_{L_y+1}(k)}{\lambda + |\phi_{L_y+1}(k)|^2} (y_d(k+1) - y(k)) - \\
 &\frac{\phi_{L_y+1}(k)}{\lambda + |\phi_{L_y+1}(k)|^2} \sum_{i=1}^{L_y} \rho_i \phi_i(k) \Delta u(k-i+1) - \\
 &\frac{\phi_{L_y+1}(k)}{\lambda + |\phi_{L_y+1}(k)|^2} \sum_{i=L_y+2}^{L_y+L_u} \rho_i \phi_i(k) \Delta u(k+L_y-i+1), \quad (C.8)
 \end{aligned}$$

onde, o fator $\rho_i \in (0,1], i = 1, \dots, L_y + L_u$ foi adicionado para fazer o algoritmo mais flexível.

Para a estimação do PG, adotou-se a seguinte função de custo:

$$\begin{aligned}
 J(\boldsymbol{\phi}_{f,L_y,L_u}(k)) &= \left| y(k) - y(k-1) - \boldsymbol{\phi}_{f,L_y,L_u}^T(k) \Delta \mathbf{H}_{L_y,L_u}(k-1) \right|^2 + \\
 &\mu \left\| \boldsymbol{\phi}_{f,L_y,L_u}(k) - \hat{\boldsymbol{\phi}}_{f,L_y,L_u}(k-1) \right\|^2 \quad (C.9)
 \end{aligned}$$

onde $\mu > 0$ é um fator de ponderação.

Minimizar a Eq. (C.9) em relação a $\boldsymbol{\phi}_{f,L_y,L_u}(k)$ produz o seguinte algoritmo de estimação da PPD:

$$\begin{aligned}
 \hat{\boldsymbol{\phi}}_{f,L_y,L_u}(k) &= \hat{\boldsymbol{\phi}}_{f,L_y,L_u}(k-1) + \frac{\eta \Delta \mathbf{H}_{L_y,L_u}(k-1)}{\mu + \|\Delta \mathbf{H}_{L_y,L_u}(k-1)\|^2} \left(\Delta y(k) - \right. \\
 &\left. \hat{\boldsymbol{\phi}}_{f,L_y,L_u}^T(k-1) \Delta \mathbf{H}_{L_y,L_u}(k-1) \right), \quad (C.10)
 \end{aligned}$$

sendo que, o fator $\eta \in (0,2]$ foi adicionado para tornar o algoritmo mais flexível e $\hat{\boldsymbol{\phi}}_{p,L}(k)$ é a estimação do PG.

Novamente, para garantir que as condições exigidas pelo Teorema B.2 e pelas Suposições B.3 e B.4 permaneçam válidas, durante todo o período de atuação do controlador, um mecanismo de reiniciação deve ser incorporado ao algoritmo de estimação do PG, a saber:

$$\hat{\boldsymbol{\phi}}_{f,L_y,L_u}(k) = \hat{\boldsymbol{\phi}}_{f,L_y,L_u}(1) \quad \text{se} \quad \left\| \hat{\boldsymbol{\phi}}_{f,L_y,L_u}(k) \right\| \leq \varepsilon \quad \text{ou} \quad \left\| \Delta \mathbf{H}_{L_y,L_u}(k-1) \right\| \leq \varepsilon \quad \text{ou} \\ \text{sign}(\hat{\phi}_{L_y+1}(k)) \neq \text{sign}(\hat{\phi}_{L_y+1}(1)) \quad (\text{C.11})$$

c) *Projeto de um CFDL - MFAC - MIMO*

O sistema MIMO não-linear discreto, Eq. (B.8), segundo o Teorema B.3, satisfeitas as Suposições B.5 e B.6 pode ser transformado no seguinte modelo de dados CFDL, Eq. (C.12):

$$\Delta \mathbf{y}(k+1) = \boldsymbol{\Phi}_c(k) \Delta \mathbf{u}(k) \quad (\text{C.12})$$

onde a matriz

$$\boldsymbol{\Phi}_c(k) = \begin{bmatrix} \phi_{11}(k) & \phi_{12}(k) & \cdots & \phi_{1m}(k) \\ \phi_{21}(k) & \phi_{22}(k) & \cdots & \phi_{2m}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{m1}(k) & \phi_{m2}(k) & \cdots & \phi_{mm}(k) \end{bmatrix} \in R^{m \times m},$$

é a PJM do sistema.

SUPosição C.1

A PJM $\boldsymbol{\Phi}_c(k)$ é uma matriz diagonalmente dominante no seguinte sentido: $|\phi_{ij}(k)| \leq b_1$, $b_2 \leq |\phi_{ii}(k)| \leq \alpha b_2$, $\alpha \geq 1$, $b_2 > b_1(2\alpha + 1)(m - 1)$, $i = 1, \dots, m$, $j = 1, \dots, m$, $i \neq j$, todos os sinais dos elementos de $\boldsymbol{\Phi}_c(k)$ são fixos.

As etapas do projeto do controlador, se resumem a calcular a estimativa da MPJ, usando:

$$\hat{\boldsymbol{\Phi}}_c(k) = \hat{\boldsymbol{\Phi}}_c(k-1) + \frac{\eta(\Delta \mathbf{y}(k) - \hat{\boldsymbol{\Phi}}_c(k-1) \Delta \mathbf{u}(k-1) \Delta \mathbf{u}^T(k-1))}{\mu + \|\Delta \mathbf{u}(k-1)\|^2}, \quad (\text{C.13})$$

considerando também as seguintes condições de reiniciação:

$$\hat{\phi}_{ii}(k) = \hat{\phi}_{ii}(1) \quad \text{se} \quad |\hat{\phi}_{ii}(k)| < b_2 \quad \text{ou} \quad |\hat{\phi}_{ii}(k)| > \alpha b_2 \quad \text{ou} \quad \text{sign}(\hat{\phi}_{ii}(k)) \neq \text{sign}(\hat{\phi}_{ii}(1)) \quad i = 1, \dots, m \quad (\text{C.14})$$

$$\hat{\phi}_{ii}(k) = \hat{\phi}_{ii}(1) \quad \text{se} \quad |\hat{\phi}_{ii}(k)| > b_1 \quad \text{ou} \quad \text{sign}(\hat{\phi}_{ii}(k)) \neq \text{sign}(\hat{\phi}_{ii}(1)), \quad i, j = 1, \dots, m, i \neq j. \quad (\text{C.15})$$

Em seguida calcular o sinal de controle, mediante:

$$\mathbf{u}(k) = \mathbf{u}(k-1) + \frac{\rho \hat{\Phi}_c^T(k)}{\lambda + \|\hat{\Phi}_c(k)\|^2} (\mathbf{y}_d(k+1) - \mathbf{y}(k)), \quad (\text{C.16})$$

onde, $\rho \in (0,1], \mu \in (0,2], \lambda > 0, \mu > 0$.

d) Projeto de um PFDL-MFAC - MIMO

O sistema MIMO não-linear discreto, Eq. (B.8), segundo o Teorema B.4, satisfeitas as Suposições B.7 e B.8 pode ser transformado no seguinte modelo de dados PFDL:

$$\Delta \mathbf{y}(k+1) = \Phi_{p,L}(k) \Delta \bar{\mathbf{U}}_L(k) \quad (\text{C.17})$$

onde $\Phi_{p,L}(k) = [\Phi_1(k) \dots \Phi_L(k)] \in R^{m \times mL}$ é a matriz PPJM do sistema, Eq. (3.10), e

$$\Phi_i(k) = \begin{bmatrix} \phi_{11i}(k) & \phi_{12i}(k) & \dots & \phi_{1mi}(k) \\ \phi_{21i}(k) & \phi_{22i}(k) & \dots & \phi_{2mi}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{m1i}(k) & \phi_{m2i}(k) & \dots & \phi_{mmi}(k) \end{bmatrix} \in R^{m \times m}, i = 1, \dots, L$$

e $\Delta \bar{\mathbf{U}}_L(k) = [\Delta \mathbf{u}^T(k), \dots, \Delta \mathbf{u}^T(k-L+1)]^T$

SUPosição C.2

$\Phi_1(k)$ da PPJM $\Phi_{p,L}(k)$ é uma matriz diagonalmente dominante no seguinte sentido: $|\phi_{ij1}(k)| \leq b_1, \quad b_2 \leq |\phi_{ii1}(k)| \leq \alpha b_2, \quad \alpha \geq 1, \quad b_2 > b_1(2\alpha + 1)(m-1), \quad i = 1, \dots, m, \quad j = 1, \dots, m, \quad i \neq j$, todos os sinais de todos os elementos de $\Phi_1(k)$ ficam inalterados.

As etapas do projeto do controlador, se resumem novamente a calcular a estimativa da PPJM, usando:

$$\hat{\Phi}_{p,L}(k) = \hat{\Phi}_{p,L}(k-1) + \frac{\eta(\Delta y(k) - \hat{\Phi}_{p,L}(k-1)\Delta \bar{U}_L(k-1))\Delta \bar{U}_L^T(k-1)}{\mu + \|\Delta \bar{U}_L(k-1)\|^2}, \quad (C.18)$$

considerando também as seguintes condições de reiniciação:

$$\hat{\phi}_{ii1}(k) = \hat{\phi}_{ii1}(1) \text{ se } |\hat{\phi}_{ii1}(k)| < b_2 \text{ ou } |\hat{\phi}_{ii1}(k)| > \alpha b_2 \text{ ou } \text{sign}(\hat{\phi}_{ii1}(k)) \neq \text{sign}(\hat{\phi}_{ii1}(1)) \quad i = 1, \dots, m \quad (C.19)$$

$$\hat{\phi}_{ij1}(k) = \hat{\phi}_{ij1}(1) \text{ se } |\hat{\phi}_{ij1}(k)| > b_1 \text{ ou } \text{sign}(\hat{\phi}_{ij1}(k)) \neq \text{sign}(\hat{\phi}_{ij1}(1)), \quad i, j = 1, \dots, m, i \neq j. \quad (C.20)$$

Em seguida calcular o sinal de controle, mediante:

$$\mathbf{u}(k) = \mathbf{u}(k-1) + \frac{\hat{\Phi}_1^T(k)\rho_1(\mathbf{y}_d(k+1) - \mathbf{y}(k))}{\lambda + \|\hat{\Phi}_1(k)\|^2} - \frac{\sum_{i=2}^L \rho_i \hat{\Phi}_i(k) \Delta \mathbf{u}(k-i+1)}{\lambda + \|\hat{\Phi}_1(k)\|^2}, \quad (C.21)$$

sendo $\rho_i \in (0,1], i = 1, \dots, L, \eta \in (0,2], \lambda > 0$, e $\mu > 0$.

e) Projeto de um FFDL- MFAC - MIMO

O sistema MIMO não-linear discreto, Eq. (B.8), segundo o Teorema B.5, satisfeitas as Suposições B.9 e B.10 pode ser transformado no seguinte modelo de dados FFDL:

$$\Delta \mathbf{y}(k+1) = \Phi_{f,L_y,L_u}(k) \Delta \bar{\mathbf{H}}_{L_y,L_u}(k) \quad (C.22)$$

sendo $\Phi_{f,L_y,L_u}(k) = [\Phi_1(k) \dots \Phi_{L_y+L_u}(k)] \in \mathbb{R}^{m \times (mL_y+mL_u)}$ a PPJM do sistema, Eq. (B.8), e

$$\Phi_i(k) = \begin{bmatrix} \phi_{11i}(k) & \phi_{12i}(k) & \dots & \phi_{1mi}(k) \\ \phi_{21i}(k) & \phi_{22i}(k) & \dots & \phi_{2mi}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{m1i}(k) & \phi_{m2i}(k) & \dots & \phi_{mmi}(k) \end{bmatrix} \in \mathbb{R}^{m \times m}, i = 1, \dots, L_y + L_u$$

$$\text{e } \Delta \bar{\mathbf{H}}_{L_y,L_u}(k) = [\Delta \mathbf{y}^T(k), \dots, \Delta \mathbf{y}^T(k - L_y + 1), \Delta \mathbf{u}^T(k), \dots, \Delta \mathbf{u}^T(k - L_u + 1)]^T$$

Mais uma vez, as etapas do projeto do controlador, se resumem a calcular a estimativa da PPJM, usando:

$$\hat{\Phi}_{f,L_y,L_u}(k) = \hat{\Phi}_{f,L_y,L_u}(k-1) + \frac{\eta(\Delta y(k) - \hat{\Phi}_{f,L_y,L_u}(k-1)\Delta \bar{H}_{L_y,L_u}(k-1))\Delta \bar{H}_{L_y,L_u}^T(k-1)}{\mu + \|\Delta \bar{H}_{L_y,L_u}(k-1)\|^2}, \quad (C.23)$$

considerando também as seguintes condições de reiniciação:

$$\hat{\phi}_{ii(L_y+1)}(k) = \hat{\phi}_{ii(L_y+1)}(1) \text{ se } |\hat{\phi}_{ii(L_y+1)}(k)| < b_2 \text{ ou } |\hat{\phi}_{ii(L_y+1)}(k)| > \alpha b_2 \text{ ou } \text{sign}(\hat{\phi}_{ii(L_y+1)}(k)) \neq \text{sign}(\hat{\phi}_{ii(L_y+1)}(1)) \quad i = 1, \dots, m \quad (C.24)$$

$$\hat{\phi}_{ij(L_y+1)}(k) = \hat{\phi}_{ij(L_y+1)}(1) \text{ se } |\hat{\phi}_{ij(L_y+1)}(k)| > b_1 \text{ ou } \text{sign}(\hat{\phi}_{ij(L_y+1)}(k)) \neq \text{sign}(\hat{\phi}_{ij(L_y+1)}(1)), i, j = 1, \dots, m, i \neq j. \quad (C.25)$$

Em seguida calcular o sinal de controle, mediante:

$$\mathbf{u}(k) = \mathbf{u}(k-1) + \frac{\hat{\Phi}_{L_y+1}^T(k)\rho_{L_y+1}(\mathbf{y}_d(k+1) - \mathbf{y}(k))}{\lambda + \|\hat{\Phi}_{L_y+1}(k)\|^2} - \frac{\hat{\Phi}_{L_y+1}^T(k)\left(\sum_{i=1}^{L_y} \rho_i \hat{\Phi}_i(k)\Delta \mathbf{y}(k-i+1) + \sum_{i=L_y+2}^{L_y+L_u} \rho_i \hat{\Phi}_i(k)\Delta \mathbf{u}(k-i+1)\right)}{\lambda + \|\hat{\Phi}_{L_y+1}(k)\|^2}, \quad (C.26)$$

onde $\rho_i \in (0,1], i = 1, \dots, L_y + L_u, \eta \in (0,2], \lambda > 0$, e $\mu > 0$.