



PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA - PPGEL
ASSOCIAÇÃO AMPLA CEFET-MG / UFSJ

Marcus Paulo Silva de Abreu

Síntese de Controladores Multivariáveis Baseados em Controle
por Modelo Interno e Otimização Evolucionária Multiobjetivo

Belo Horizonte
2019

Marcus Paulo Silva de Abreu

SÍNTESE DE CONTROLADORES MULTIVARIÁVEIS BASEADOS EM CONTROLE
POR MODELO INTERNO E OTIMIZAÇÃO EVOLUCIONÁRIA MULTIOBJETIVO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, associação ampla entre Centro Federal de Educação Tecnológica de Minas Gerais e Universidade Federal de São João del-Rei como parte dos requisitos exigidos para obtenção do título de Mestre em Engenharia Elétrica.

Área de concentração: Modelagem e Controle de Sistemas.

Linha de pesquisa: Sistemas de controle.

Orientador: Eduardo Nunes Gonçalves

Belo Horizonte
2019

A162s Abreu, Marcus Paulo Silva de
Síntese de controladores multivariáveis baseados em controle por modelo interno e otimização evolucionária multiobjetivo / Marcus Paulo Silva de Abreu. – 2019.
xiv, 72 f.: il., gráfs, tabs.

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica em associação ampla entre a UFSJ e o CEFET-MG.

Orientador: Eduardo Nunes Gonçalves.

Dissertação (mestrado) – Centro Federal de Educação Tecnológica de Minas Gerais.

1. Controle de processo – Teses. 2. Controladores PID – Teses. 3. Programação evolutiva (Computação) – Teses. 4. Otimização matemática – Teses. I. Gonçalves, Eduardo Nunes. II. Centro Federal de Educação Tecnológica de Minas Gerais. III. Universidade Federal de São João del-Rei. IV. Título.

CDD 629.833

DEDICO ESSE TRABALHO AOS
MEUS PAIS, IRMÃOS, NAMORADA E
AMIGOS.

Agradecimentos

Agradeço,

Aos meus pais, Maria das Graças e Rogério de Abreu, pelo incentivo, apoio e suporte que alicerçaram a minha jornada até aqui e por sempre estarem presentes.

Aos meus irmãos, Rafael Pedro e Ana Victória, pela confiança e pela atenção despendida nos momentos que precisei.

À minha namorada, Fabiane Rocha, pelo companheirismo, paciência, compreensão e carinho, fundamentais para que eu tivesse forças para concluir mais essa etapa da minha vida.

Ao meu orientador, Eduardo Nunes Gonçalves, pelos ensinamentos e pela paciência, atenção e serenidade que tornaram mais agradável o percurso realizado no desenvolvimento desse trabalho. Os aprendizados adquiridos com ele, além de imprescindíveis para o êxito dessa jornada, certamente serão importantes para vencer os desafios futuros.

Aos meus colegas de república Edmar e Luccas e a todos os meus colegas de mestrado do PPGEL pelas parcerias e excelente convivência.

Ao CEFET-MG, à FAPEMIG e à CAPES pelo apoio financeiro e pela oportunidade de ingressar no PPGEL.

Não importa o quão difícil a vida possa parecer,
há sempre alguma coisa que você pode fazer e ter
sucesso. Enquanto há vida, há esperança.

Stephen Hawking

Resumo

Os processos industriais, frequentemente, possuem plantas com múltiplas entradas e múltiplas saídas com atrasos no tempo em suas malhas, o que dificulta a realização do controle. Os controladores PID, que a despeito dos avanços nos sistemas de controle digital, são majoritários no ambiente industrial, necessitam do ajuste de três parâmetros: o ganho proporcional, o tempo derivativo e o tempo integral. Com a estrutura de controle por modelo interno, porém, é possível obter controladores PID ou PI sintonizando apenas um parâmetro por malha de controle. Nesse contexto, essa dissertação apresenta uma forma de projeto simples e eficiente de controladores PI, com desacoplador, baseado no controle por modelo interno e em otimização multiobjetivo. Inicialmente, é projetado um desacoplador, utilizando o método Evolução Diferencial para minimizar a diferença entre as respostas transitórias da matriz de transferência diagonal especificada e o sistema desacoplado. A partir do sistema desacoplado, são utilizadas as regras de sintonia IMC para gerar os controladores PI (IMC-PI). É proposto nesse trabalho a aplicação do algoritmo Evolução Diferencial para Otimização Multiobjetivo para determinação dos parâmetros dos controladores IMC-PI, o que torna possível a obtenção de um grupo de soluções eficientes com diferentes compromissos entre o rastreamento da referência e o esforço de controle. Para mostrar a eficácia da metodologia abordada, os controladores IMC-PI propostos são aplicados em três exemplos ilustrativos de sistemas com forte acoplamento, sendo dois deles com elementos de atraso em suas malhas.

Palavras-chave: Controle por modelo interno, controle PID multivariável, otimização evolucionária multiobjetivo.

Abstract

Industrial processes often present plants with multiple input and multiple output with time delays in their loops, which makes it difficult to carry out the control. PID controllers, which in spite of advances in digital control systems, are the majority in the industrial field, need the tuning of three parameters: proportional gain, derivative time and integral time. With the internal model control structure, however, it is possible to obtain PID or PI controllers by tuning only one parameter per control loop. In this context, this dissertation presents a simple and efficient way to design of PI controllers, with decoupler, based on internal model control and multiobjective optimization. First, a decoupler is designed using the Differential Evolution method to minimize the difference between the transient responses of the specified diagonal transfer matrix and the decoupled system. From the decoupled system, the IMC tuning rules are applied to generate the PI controllers (IMC-PI). In this work, it is proposed to apply the Differential Evolution for Multiobjective Optimization algorithm to determine the parameters of the IMC-PI controllers, which makes it possible to obtain a group of efficient solutions with different trade-offs between the set point tracking and the control effort. In order to show the effectiveness of the methodology, the proposed IMC-PI controllers are applied in three illustrative examples of systems with strong coupling, in which two of them have time delay.

Keywords: Internal model control, multivariable PID control, multiobjective evolutionary optimization.

Sumário

Lista de Figuras	xii
Lista de Tabelas	xiii
Lista de Acrônimos	xiv
Lista de Notações	xv
1 Introdução	1
1.1 Revisão Bibliográfica	1
1.2 Motivação	6
1.3 Objetivos do Trabalho	7
1.4 Organização do Documento	8
2 Controle por Modelo Interno - IMC	9
2.1 Introdução	9
2.2 Controle IMC SISO	10
2.3 Projeto de controlador IMC	11
2.4 Controladores IMC-PID e IMC-PI	13
2.4.1 Controlador IMC-PI	14
2.4.2 Controlador IMC-PID	16
2.5 IMC multivariável	18
2.5.1 IMC centralizado	18
2.5.2 IMC descentralizado com desacoplador	19
2.6 Projeto IMC-PI ótimo	21
2.7 Conclusões	24
3 Otimização Multiobjetivo	25
3.1 Introdução	25
3.2 Algoritmo Evolução Diferencial	26
3.2.1 População inicial	27
3.2.2 Mutação diferencial	27
3.2.3 Recombinação	28
3.2.4 Seleção	29
3.3 Algoritmos evolucionários multiobjetivos	30
3.3.1 Algoritmo NSGA-II	32

3.3.2	Algoritmo DEMO	35
3.4	Conclusões	37
4	Exemplos Ilustrativos	38
4.1	IMC centralizado ótimo aplicado ao modelo de quatro tanques	38
4.1.1	Resposta em malha aberta	38
4.1.2	Síntese do controlador IMC centralizado	40
4.2	IMC-PI ótimo descentralizado aplicado ao modelo quatro tanques	44
4.2.1	Desenvolvimento do desacoplador	44
4.2.2	Síntese do controlador IMC-PI	45
4.3	IMC-PI ótimo descentralizado aplicado ao modelo Wood-Berry	48
4.3.1	Resposta em malha aberta	49
4.3.2	Desenvolvimento do desacoplador	50
4.3.3	Síntese do controlador IMC-PI	51
4.4	IMC-PI ótimo descentralizado aplicado ao modelo HVAC	55
4.4.1	Resposta em malha aberta	55
4.4.2	Desenvolvimento do desacoplador	56
4.4.3	Síntese do controlador IMC-PI	57
5	Considerações Finais	63
5.1	Conclusões	63
5.2	Propostas de continuidade	64
	Referências	66

Lista de Figuras

2.1	Representação esquemática do IMC	10
2.2	Representação esquemática da estrutura equivalente do IMC	14
2.3	Estrutura de controle descentralizado com desacoplador	20
2.4	Esquema do IMC com desacoplador	20
3.1	Esquema do mecanismo de mutação diferencial (adaptado de STORN; PRICE (1995))	28
3.2	Ilustração do processo de recombinação para um problema com 7 parâ- metros (adaptado de STORN; PRICE (1997))	29
3.3	Distância de aglomeração (reproduzido de DEB <i>et al.</i> (2002))	33
3.4	Operação de seleção do NSGA-II (baseado em DEB <i>et al.</i> (2002))	35
4.1	Esquema do sistema de quatro tanques (adaptado de JOHANSSON (2000))	39
4.2	Resposta ao degrau do sistema em malha aberta - Modelo quatro tanques	39
4.3	Conjunto de soluções candidatas de Pareto - Modelo quatro tanques . . .	40
4.4	Resposta ao degrau y_1 - Modelo quatro tanques	42
4.5	Resposta ao degrau y_2 - Modelo quatro tanques	42
4.6	Esforço de controle u_1 - Modelo quatro tanques	43
4.7	Esforço de controle u_2 - Modelo quatro tanques	43
4.8	Resposta ao degrau de H , em azul, e \hat{G} , em vermelho - Modelo quatro tanques	45
4.9	Conjunto de soluções candidatas de Pareto para controlador descentrali- zado - Modelo quatro tanques	46
4.10	Resposta ao degrau y_1 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques	47
4.11	Resposta ao degrau y_2 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques	48
4.12	Esforço de controle u_1 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques	48
4.13	Esforço de controle u_2 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques	49
4.14	Resposta ao degrau do sistema em malha aberta - Modelo Wood-Berry . .	49
4.15	Resposta ao degrau de H , em azul, e \hat{G} , em vermelho - Modelo Wood-Berry	51
4.16	Conjunto de soluções candidatas de Pareto - Modelo Wood-Berry	52
4.17	Resposta ao degrau y_1 - Modelo Wood-Berry	53
4.18	Resposta ao degrau y_2 - Modelo Wood-Berry	53

4.19	Esforço de controle u_1 - Modelo Wood-Berry	54
4.20	Esforço de controle u_2 - Modelo Wood-Berry	54
4.21	Resposta ao degrau do sistema em malha aberta - Modelo HVAC	56
4.22	Resposta ao degrau de H , em azul, e \hat{G} , em vermelho - Modelo HVAC	57
4.23	Conjunto de soluções candidatas de Pareto - Modelo HVAC	58
4.24	Resposta ao degrau y_1 - Modelo HVAC	60
4.25	Resposta ao degrau y_2 - Modelo HVAC	60
4.26	Resposta ao degrau y_3 - Modelo HVAC	61
4.27	Resposta ao degrau y_4 - Modelo HVAC	61
4.28	Esforço de controle u_1 - Modelo HVAC	61
4.29	Esforço de controle u_2 - Modelo HVAC	62
4.30	Esforço de controle u_3 - Modelo HVAC	62
4.31	Esforço de controle u_4 - Modelo HVAC	62

Lista de Tabelas

4.1	Desempenho dos controladores - Modelo quatro tanques	41
4.2	Desempenho dos controladores descentralizados IMC-PI - Modelo quatro tanques	47
4.3	Desempenho dos controladores - Modelo Wood-Berry	52
4.4	Constantes de tempo das funções em malha-fechada, T_f - Modelo Wood-Berry	52
4.5	Desempenho dos controladores - Modelo HVAC	59
4.6	Constantes de tempo das funções em malha-fechada, T_f - Modelo HVAC .	59

Lista de Acrônimos

DE	do inglês, <i>Differential Evolution</i>
DEMO	do inglês, <i>Differential Evolution for Multiobjective Optimization</i>
EOTF	do inglês, <i>Effective Open-loop Transfer function</i>
FOPTD	do inglês, <i>First Order Plus Time Delay</i>
HVAC	do inglês, <i>Heating, Ventilation, and Air Conditioning</i>
IMC	do inglês, <i>Internal Model Control</i>
IMC-PI	Controlador PI obtido através das regras de sintonia IMC
IMC-PID	Controlador PID obtido através das regras de sintonia IMC
ISE	do inglês, <i>Integral of the Square of the Error</i>
MIMO	do inglês, <i>Multiple Input, Multiple Output</i>
MOEA	do inglês, <i>Multiobjective Evolutionary Algorithm</i>
MOEA/D	do inglês, <i>Multiobjective Evolutionary Algorithm Based on Decomposition</i>
NPGA	do inglês, <i>Niched Pareto Genetic Algorithm</i>
NSGA	do inglês, <i>Non-dominated Sorting Genetic Algorithm</i>
PAES	do inglês, <i>Pareto Archived Evolution Strategy</i>
PI	Proporcional-Integral
PID	Proporcional-Integral-Derivativo
RNGA	do inglês, <i>Relative Normalized Gain Array</i>
SISO	do inglês, <i>Single Input, Single Output</i>
SPD	Semi-Plano Direito
SPEA	do inglês, <i>Strength Pareto Evolutionary Algorithm</i>
VEGA	do inglês, <i>Vector Evaluation Genetic Algorithm</i>

Lista de Notações

Q	Controlador interno
G	Processo
\tilde{G}	Modelo do processo (modelo interno)
r	Sinal de referência
u	Sinal de controle
d	Sinal de perturbação
y	Sinal de saída
Q	Controlador IMC
\tilde{G}_-	Modelo do processo sem elementos de atraso e zeros não mínimos
$Re(\beta)$	Parte real de β
T_f	Filtro passa-baixa
λ	Constante de tempo de T_f
D	Desacoplador
H	Modelo desejado
$diag(\tilde{G})$	Diagonal principal da matriz \tilde{G}
$\ e(t)\ _2$	Norma 2 de um sinal $e(t)$
$\ u(t)\ _\infty$	Norma infinito do sinal $u(t)$
$\mathcal{U}_{(a,b)}$	Número pseudo-aleatório uniformemente distribuído no intervalo (a,b)
$\mathcal{I}_{(m)}$	Inteiro pseudo-aleatório uniformemente distribuído em $1, \dots, m$
\mathbf{x}^*	Solução ótima
\mathcal{X}^*	Conjunto de soluções candidatas de Pareto
\mathbb{R}	Conjunto dos números reais
\mathbb{Z}	Conjunto dos números inteiros

Introdução

1.1 Revisão Bibliográfica

Sistemas com múltiplas entradas e múltiplas saídas (MIMO - do inglês, *Multiple Input, Multiple Output*) são muito comuns nos processos industriais (JEROME; RAY, 1986). A abordagem de sistemas MIMO é geralmente mais complexa que a abordagem de sistemas com apenas uma entrada e uma saída (SISO - do inglês, *Single Input, Single Output*), principalmente quando existe elementos de atraso no tempo e forte interação entre as malhas de controle. Os elementos de atraso, que decorrem principalmente do transporte de massa, informação ou energia, impõem restrições ao desempenho do sistema de controle (NORMEY-RICO; CAMACHO, 2007). Para lidar com esses processos, os controladores PID são, até os dias atuais, majoritários nas aplicações industriais, a despeito dos avanços nos sistemas de controle digital. A popularidade dos controladores PID se justifica pela sua estrutura simples de fácil implementação, pela baixa quantidade de parâmetros de sintonia e por já serem bem conhecidos pelos operadores (ÅSTRÖM; HÄGGLUND, 1995). Existem na literatura muitas técnicas para determinação dos três parâmetros (ganho proporcional, tempo integral e tempo derivativo) do controlador PID. Nas indústrias, no entanto, muitas vezes o que se vê é uma dispendiosa implementação por tentativa e erro. Nesse caso, o sistema de controle final não apresenta um desempenho ótimo em termos de rastreamento da trajetória, rejeição aos distúrbios e atenuação de ruídos.

A estrutura de controle PID para sistemas MIMO mais simples é o PID descentralizado, onde o controlador é uma matriz de transferência diagonal. Tal estrutura, contudo, pode apresentar um desempenho ruim em sistemas com alta interação entre as malhas, o que pode causar instabilidade em malha fechada (JUNIOR; BARROS, 2015). Uma forma comum de verificação do nível de interação entre as malhas de controle é o método RGA (do inglês, *Relative Gain Array*) (BRISTOL, 1966).

Nesse contexto, destacam-se duas alternativas para lidar com o alto acoplamento: o controlador PID centralizado e o controlador descentralizado com desacoplador. A sinto-

nia do controlador PID centralizado é mais complexa, pois todos os elementos da matriz de transferência são compostos por controladores PID. O controle com o desacoplador, por sua vez, combina a simplicidade do controle PID descentralizado com o desacoplador. O desacoplador é designado de forma que, após ser implementado, precedendo processo, o modelo resultante se aproxime de uma matriz diagonal cujas funções de transferências possam ser controladas individualmente como um sistema de uma entrada e uma saída.

O desacoplador é desenvolvido de forma que o modelo desejado seja aproximadamente igual ao produto entre o desacoplador e o processo. GAGNON; POMERLEAU; DESBIENS (1998) classificam os desacopladores obtidos de forma analítica em três grupos: ideal, simples ou invertido. O desacoplador ideal é dado pela produto entre a inversa do modelo e a matriz de transferência desejada. Apesar da sintonia do controlador ser mais fácil quando se utiliza o desacoplador ideal, o desacoplador, em si, pode ser muito complexo, não realizável ou até mesmo instável. No desacoplador simples os elementos da diagonal são fixados igual a 1 e os elementos diferentes de 1 do desacoplador são funções das frações dos elementos do processo. Embora esse tipo de desacoplador seja mais simples que o ideal, ele resulta em um modelo mais complexo, o que dificulta o projeto do controlador. O desacoplador invertido, por sua vez, busca reunir as principais vantagens dos desacopladores ideal e simples, ou seja, controlador fácil de ser sintonizado e desacoplador simples. O modelo desejado, nesse caso, é dado pela diagonal principal do processo.

A tarefa de desenvolver sistemas de controle desacoplados satisfatórios para processos MIMO constitui um problema que recebeu um grande interesse nas últimas décadas, como pode ser visto em SHEN; CAI; LI (2010); SHEN; SUN; LI (2012); RAJAPANDIYAN; CHIDAMBARAM (2012) e VU; LEE (2013). Nessa situação, principalmente no que se refere a sistemas MIMO, as pesquisas que visam o desenvolvimento de técnicas para síntese de controladores PID com menos parâmetros de ajuste se fazem muito pertinentes. Isso acontece porque, como a quantidade de parâmetros está relacionada a quantidade de malhas do sistema, modelos representados por matrizes de ordem alta implicam em maior dificuldade para a sintonia do controlador. Nesse aspecto, uma abordagem interessante é o controle por modelo interno (IMC - do inglês, *Internal Model Control*).

Introduzida por GARCIA; MORARI (1982), inicialmente a metodologia IMC foi desenvolvida para sistemas SISO. Essa estrutura de controle possui somente um parâmetro de sintonia, a constante de tempo desejada para o sistema em malha fechada. Para a maioria dos modelos comuns nos sistemas de controle, a abordagem IMC conduz a um controlador PID ou PI na forma clássica ou aumentados por um filtro de primeira ordem (RIVERA; MORARI; SKOGESTAD, 1986). Os controladores PI ou PID resultantes da aplicação da metodologia IMC são chamados na literatura de IMC-PI ou IMC-PID. Esses

controladores foram formalizados pela primeira vez por RIVERA; MORARI; SKOGESTAD (1986) para sistemas SISO e, desde então, muitas pesquisas vêm sendo realizadas.

Em se tratando de sistemas MIMO, o controle IMC descentralizado com desacoplador vem sendo muito utilizado. Diante disso, a busca por técnicas de desacoplamento mais eficientes é bastante recorrente. GARRIDO; VÁZQUEZ; MORILLA (2014) propõem o *inverted decoupling* IMC (IDIMC) para processos quadrados, isto é, processos com o mesmo número de entradas e saídas. Nesse método, o bloco Q (discorrido no capítulo 2) é dividido em dois blocos, Q_d e Q_o , sendo o primeiro alocado de forma direta (entre o sinal do erro e o bloco do processo) e o segundo, implementado como realimentação positiva do primeiro. Em seguida é aplicado um filtro para atenuação dos distúrbios. Os autores, no entanto, não fazem menção aos controladores PIDs resultantes da metodologia empregada. JIN *et al.* (2016) também utilizam o desacoplador invertido. Nesse caso, porém, a metodologia é estendida para plantas não quadradas e os procedimentos resultam em controladores IMC-PID acompanhados de um filtro na realimentação. WANG; ZHANG; CHIU (2002) abordam uma técnica de controle IMC com desacoplamento para plantas com múltiplos atrasos. No entanto, os controladores resultantes dessa proposta podem conter atrasos e zeros de fase não mínima. DIAZ *et al.* (2018), por sua vez, propõe um controlador IMC-PID desenvolvido com base em um sistema *benchmark*. JIN *et al.* (2014) propõe um método de projeto de desacoplador utilizando IMC como forma de controle de processos não quadrados para sistemas com múltiplos atrasos. Para tornar o projeto mais simples, uma matriz da função de transferência equivalente, baseada na matriz de ganho normalizada relativa (RNGA - do inglês, *Relative Normalized Gain Array*) (HE *et al.*, 2009), é introduzida como pseudo-inversa da matriz de função de transferência.

Existem também na literatura alguns trabalhos envolvendo IMC-PID centralizado. Dentre os quais, destacam-se VU; LEE; LEE (2007) e JIN; DU; JIANG (2017). O primeiro combina a metodologia IMC com um critério de desempenho que envolve a máxima magnitude da resposta em frequência do sistema em malha fechada. O segundo utiliza um controlador IMC-PID composto por um filtro de matriz completa e um controlador PID diagonal.

Diversas técnicas abordadas recentemente permitem o projeto do controlador IMC-PI e IMC-PID com base em algum critério de desempenho definido pelo usuário. Por exemplo, ALI; MAJHI (2009) propõe controladores projetados de acordo com a especificação do sobresinal para uma entrada de referência tipo degrau. Tal técnica é viabilizada pela determinação de uma relação empírica entre o sobresinal e o valor máximo da função de sensibilidade complementar. Uma estrutura IMC de dois graus de liberdade (2DOF - do inglês, Two Degree of Freedom) cujos parâmetros são definidos de acordo com o nível de robustez desejado é proposta por VILANOVA (2008). Em outro trabalho (MAMAT

et al., 2013), o ajuste dos parâmetros do controlador é dependente do valor de pico e da margem de fase desejadas. Já SAXENA; HOTE (2016), aborda controladores PID através do conceito IMC tradicional. Nesse caso, para converter o controlador IMC em um controlador *feedback* clássico, o atraso do modelo é negligenciado.

Alguns trabalhos mostram que, embora o método IMC convencional possa atenuar de forma satisfatória os distúrbios de saída, ele não fornece uma boa rejeição à perturbações de carga na entrada nos casos que a dinâmica do processo é muito mais lenta do que a dinâmica do sistema em malha fechada (LI; GU, 2012; HORN *et al.*, 1996). Nesse contexto, opções para tratar o problema regulador, podem ser vistas em HORN *et al.* (1996) e SHAMSUZZOHA; LEE (2007), onde, para melhorar a rejeição ao distúrbio, são utilizadas regras alternativas de sintonia para os parâmetros do controlador IMC-PID em série com um filtro. As estruturas propostas, porém, são complexas e demandam alto esforço computacional.

O método tradicional IMC não pode ser aplicado em plantas instáveis, pois, como será apresentado no capítulo 2, para que o sistema como um todo seja estável, é necessário que a planta e o controlador sejam estáveis. Provavelmente o primeiro trabalho a abordar esse problema utilizando controladores IMC-PI e IMC-PID tenha sido realizado por ROTSTEIN; LEWIN (1991). Cabe ressaltar, contudo, que geralmente são realizadas algumas adaptações ao método IMC clássico para tornar o modelo aparente estável. Uma técnica utilizada nesses casos é a implementação de um controlador cuja única incumbência é estabilizar a planta. Esse controlador é aplicado de forma a realimentar diretamente o processo instável, conforme mostra CHIEN (1990) e ISSA (2011). Por sua vez, DEHGHANI; LANZON; ANDERSON (2006) e ALCÁNTARA *et al.* (2011) combinam a simplicidade da técnica IMC com o controle \mathcal{H}_∞ em sistemas SISO. Dessa forma, além de possibilitarem a abordagem de plantas instáveis, obtêm o rastreamento da trajetória com uma boa rejeição ao distúrbio. Já NASUTION; JENG; HUANG (2011), decompõe o controlador IMC-PID por série de Maclaurin com desempenho \mathcal{H}_2 ótimo para plantas instáveis.

Com intuito de atingir o melhor desempenho possível, as técnicas de otimização são frequentemente utilizadas para o projeto de diversos tipos de controlador. IRUTHAYARAJAN; BASKAR (2009) estabelecem uma comparação entre diferentes algoritmos evolucionários utilizados na determinação dos parâmetros de controladores PID aplicados à modelos clássicos da literatura. PANDA (2009) utiliza o algoritmos de evolução diferencial na determinação dos parâmetros do controlador TCSC (do inglês, *Thyristor Controlled Series Capacitor*). GONÇALVES *et al.* (2012) utiliza método cone elipsoidal para o desenvolvimento de controladores PIs desacoplados robustos.

Alguns trabalhos utilizam as técnicas de otimização no projeto de controladores IMC-

PID e IMC-PI. Por exemplo, ALVAREZ-RAMIREZ *et al.* (2004) utilizam sistemas SISO para mostrar a eficácia do controle IMC-PID com parâmetros ótimos. Para tal, realizam uma comparação entre os resultados alcançados analiticamente e os resultados alcançados via técnicas de otimização. JIN; HAO; WANG (2013), por sua vez, utilizam o algoritmo de otimização NPSO (do inglês, *New Particle Swarm Optimization*) (YANG; SIMON, 2005) para determinação dos parâmetros do IMC-PID para processos MIMO. Além disso, um método baseado no EOTF (do inglês, *Effective Open-loop Transfer function*) (XIONG; CAI, 2006), que permite a obtenção de modelos equivalentes para plantas não quadradas, é abordado. Tal método, contudo, mesmo empregado em plantas formadas por funções de transferência de primeira ordem, pode gerar modelos equivalentes de segunda ordem. Assim, os controladores obtidos não são PIs ou PIDs “puros”, pois são sempre acompanhados de um filtro.

Um apanhado acerca das produções mais relevantes envolvendo o método IMC até o ano de 2012 pode ser encontrado em SAXENA; HOTE (2012). Nesse artigo os autores buscaram revisar e explicar tanto as técnicas e desenvolvimentos presentes na literatura, bem como as perspectivas futuras relacionadas ao controle por modelo interno. É pertinente frisar que a simplicidade e eficácia do IMC permite a associação com diversas formas de projeto. Dentre os quais, vale destacar a modelagem por redes neurais (RIVALS; PERSONNAZ, 2000), por lógica *fuzzy* (XIE; RAD, 2000; NATH; DEY; MUDI, 2018) e por séries de Volterra (ZHENG; ZAFIRIOU, 1999). Um trabalho mais recente (NAJAFIZADEGAN; MERRIKH-BAYAT; JALILVAND, 2017) apresenta uma abordagem que consiste em controladores IMC-PID calculados através de um procedimento de modelagem formulado como um problema de autovalor generalizado (GEVP - do inglês, *Generalized Eigenvalue Problem*) de desigualdade matricial linear (LMI - do inglês, *Linear Matrix Inequality*) (BOYD *et al.*, 1994). Uma comparação entre os resultados adquiridos por diferentes técnicas que utilizam o o método IMC no projeto de controladores PID pode ser vista em VISIOLI (2005).

No presente trabalho, a fim de garantir que a matriz desejada tenha apenas elementos de primeira ordem, abriu-se mão de utilizar técnicas analíticas para o desenvolvimento dos desacopladores. Em lugar, serão utilizadas técnicas de otimização. Dessa forma, por consequência, garante-se também que os controladores resultantes sejam IMC-PIs sem filtros adicionais. Em virtude do bom desempenho ratificado por alguns exemplos da literatura, bem como por testes realizados previamente, optou-se por utilizar o algoritmo de evolução diferencial mono-objetivo no projeto do desacoplador e o algoritmo de evolução diferencial para otimização multiobjetivo para o projeto do controlador IMC-PI.

1.2 Motivação

Apesar da simplicidade dos controladores PID e PI, quando o problema de controle envolve sistemas MIMO, a tarefa de sintonia não é tão trivial. Na prática, as dificuldades inerentes ao projeto dos sistemas de controle podem estar relacionadas, dentre outros, a fatores como forte interação entre as malhas, múltiplos elementos de atrasos e um número alto de entradas e saídas. Cada um desses fatores atribui uma complexidade distinta ao projeto. Portanto, processos compostos por um ou mais desses fatores, oferecem um grande espaço para pesquisas e trabalhos que visam tornar mais simples, fácil e com menor custo o desenvolvimento dos sistemas de controle. Paralelo a isso, não se pode abrir mão de um bom desempenho do sistema final sob diversos critérios, tais como velocidade de convergência, esforço de controle, erro em regime permanente etc. Nesse cenário, a principal motivação desse trabalho é desenvolver uma nova metodologia de síntese de sistemas de controle simples, eficaz e flexível à diversos tipos de planta.

A busca pela simplicidade no projeto do controlador, aliada a garantia de um bom desempenho para o sistema de controle, incitou ao estudo do desenvolvimento de controladores desacoplados PI e PID através da metodologia IMC e técnicas de otimização. Utilizando o IMC, o número de parâmetros de sintonia por malha de controle é reduzido, o que simplifica o projeto e diminui o custo computacional. Por sua vez, empregando técnicas de otimização para sintonia desses parâmetros, é possível alcançar um desempenho eficiente para o sistema em malha fechada. Por fim, a utilização de desacopladores, além de tornar o controlador mais simples, diminui as perturbações geradas pelas interações entre as malhas.

Os controladores resultantes da metodologia empregada, os chamados IMC-PIs ou IMC-PIDs, são, dependendo de algumas características do processo, normalmente concebidos acompanhados de filtros. Tal fato, por acrescentar blocos adicionais ao controlador, aumenta a complexidade do sistema, o que contrapõe o intuito principal do presente trabalho. Uma maneira de sanar esse problema pode ser destacada como uma segunda motivação para realização dessa dissertação: a possibilidade de viabilizar uma forma de garantir, ao término da aplicação dos procedimentos da metodologia IMC, controladores PIs (ainda mais simples que os PIDs) sem filtros adicionais, ou seja IMC-PIs “puros”. Para que isso seja possível, é necessário que os elementos da matriz de transferência do processo sejam todos de primeira ordem. Entretanto, muitas plantas não são compostas apenas por elementos de primeira ordem e, além disso, ao se utilizar os desacopladores analíticos em modelos de primeira ordem, frequentemente os resultados são modelos desacoplados compostos por elementos de ordem superior. Nesse contexto, a alternativa proposta é a utilização de técnicas de otimização também no desenvolvimento de desacopladores. Dessa forma, é possível fixar as matrizes de transferência desejadas diagonais de maneira

que sejam compostas apenas por elementos de primeira ordem.

A maioria das técnicas de sintonia de controladores atuais visam atender a necessidades específicas. Em outras palavras, excluindo a sintonia por tentativa e erro, os métodos de projeto geralmente são “engessados” e não oferecem ao usuário a possibilidade de favorecer um critério de desempenho mais importante em detrimento de outro menos importante. Um projetista de um determinado sistema de controle pode, por exemplo, desejar que o tempo de acomodação do sinal de saída do sistema seja menor, mesmo que isso custe um alto esforço de controle. Nessa circunstância, faz-se pertinente o estudo de alternativas concretas para a mudança dos parâmetros do controlador e, ainda, que o resultado dessas mudanças sejam previstos. Posto isso, outra motivação a ser destacada é a escassez de uma técnica de projeto na qual seja possível que o usuário escolha, dentre várias possibilidades disponíveis, os valores dos parâmetros mais convenientes para os requisitos de desempenho em questão.

1.3 Objetivos do Trabalho

O objetivo principal desse trabalho é propor e discutir uma forma de sintonia simples e eficiente de controladores IMC-PI/PID desacoplados em sistemas MIMOs com elementos de atraso no tempo. Objetivo do qual se extrai os seguintes propósitos complementares:

1. Garantir que os controladores IMC-PI desenvolvidos não sejam acompanhados de filtros adicionais. Assim, dentro das variações de controladores PI ou PID possíveis de se alcançar a partir da metodologia IMC, obter a forma mais simples. Para tal, desenvolver desacopladores que resultam em matrizes de transferência compostas apenas por elementos de primeira ordem.
2. Aplicar e discutir a técnica de otimização Evolução Diferencial no desenvolvimento dos desacopladores de primeira ordem referidos no item anterior.
3. Propor e discutir uma técnica que permita ao usuário escolher, dentro de um leque de opções, de acordo com os requisitos de projeto, valores para os parâmetros dos controladores IMC-PI que mais convém às necessidades específicas do sistema em questão.
4. Aplicar e discutir a técnica de Evolução Diferencial Multiobjetivo na determinação dos parâmetros IMC-PI. Através desse algoritmo é gerada uma curva de Pareto com soluções que estabelecem uma relação entre dois ou mais objetivos determinados pelo usuário. Nesse trabalho, os objetivos escolhidos foram a máxima amplitude do sinal de controle e a integral do erro ao quadrado (ISE - do inglês, *Integral of the Square of the Error*) do sinal de saída.

1.4 Organização do Documento

Este trabalho apresenta a seguinte organização:

- **Capítulo 1 - Introdução:** Apresenta uma revisão acerca dos trabalhos mais relevantes e atuais relacionados aos temas abordados; os fatores que motivaram a elaboração desse trabalho e os principais objetivos.
- **Capítulo 2 - Controle por Modelo Interno - IMC:** Consiste em uma fundamentação teórica em torno do método IMC para sistemas SISO e MIMO, bem como dos controladores derivados, IMC-PI e IMC-PID. Além disso, é realizada uma descrição da técnica proposta, na qual, tanto os desacopladores, quanto os controladores IMC-PI são desenvolvidos com o auxílio das técnicas de otimização mono e multiobjetivo.
- **Capítulo 3 - Otimização Multiobjetivo:** - Apresenta uma breve revisão a respeito dos principais trabalhos envolvendo otimização multiobjetivo e uma fundamentação teórica em torno das técnicas de otimização utilizadas nesse trabalho.
- **Capítulo 4 - Exemplos Ilustrativos:** - Esse capítulo contém exemplos da técnica proposta aplicada em modelos de planta clássicos da literatura. É realizada também uma análise dos resultados obtidos com os controladores propostos em comparação com os controladores desenvolvidos em outros trabalhos.
- **Capítulo 5 - Considerações Finais:** - Apresenta as conclusões e propostas de trabalhos futuros.

Controle por Modelo Interno - IMC

2.1 Introdução

Embora outras metodologias de controle conceitualmente similares ao controle por modelo interno (IMC - do inglês *Internal Model Control*) já fossem estudadas em anos anteriores, foi em GARCIA; MORARI (1982) que a técnica foi formalizada. Segundo a teoria proposta pelos autores, “o controle só pode ser alcançado se o sistema de controle envolve, implícita ou explicitamente, alguma representação do processo a ser controlado.”

Embasada na teoria supracitada, a estrutura do IMC foi desenvolvida de forma a envolver diretamente um modelo do processo. ISSA (2011) estabelece uma analogia, em que, visando facilitar a compreensão da utilidade do modelo interno, o mesmo é comparado com a “intuição” de um operador de uma planta:

“Quando o operador, no modo de controle manual, tenta manter uma variável controlada perto de um ponto de ajuste desejado, ele executa um cálculo simples com base em sua representação intuitiva (modelo) do processo para definir o valor adequado da variável manipulada. O operador calcula a diferença entre o valor real da saída controlada e a estimativa (predição) do efeito do valor da variável manipulada na saída do processo. O cálculo dessa diferença é a informação básica sobre a qual se baseia a decisão de definir a amplitude da alteração da variável manipulada que é enviada para a planta.”

Existem dois tipos de problema na área do controle: o problema regulador, onde o controlador é desenvolvido visando rejeitar as perturbações ou atenuar os seus efeitos na resposta do sistema, e o problema servo, em que o projeto do controlador busca fazer com que a saída do sistema rastreie a referência. Ao longo desse capítulo será mostrado que o projeto de controle por IMC lida com os problemas regulador e servo simultaneamente de forma satisfatória. Ademais, a estrutura e os conceitos por trás dessa metodologia de controle serão detalhados.

2.2 Controle IMC SISO

Uma representação esquemática do IMC é apresentada na figura 2.1.

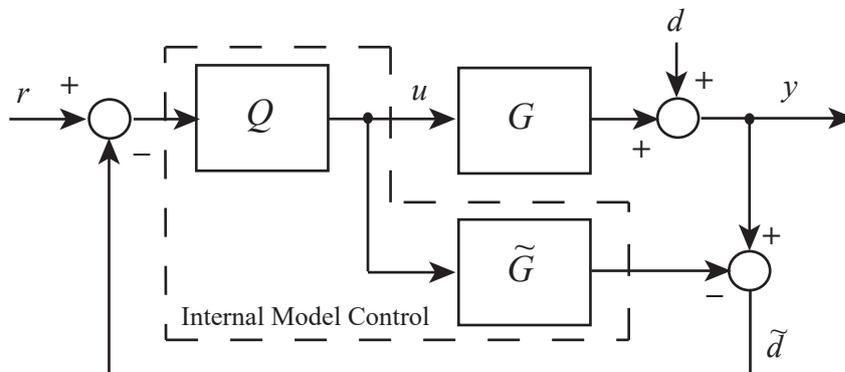


Figura 2.1: Representação esquemática do IMC

Na figura 2.1, G representa o processo real, \tilde{G} é o modelo do processo (modelo interno), r é o sinal de referência, u é o sinal de controle, d é o sinal de perturbação, y é o sinal de saída e Q é o chamado “controlador” IMC. Observa-se que o bloco da função de transferência do processo real e o bloco da função de transferência do modelo estão em paralelo e, portanto, o sinal de controle age igualmente nesses dois blocos. Com essa estrutura, caso o modelo seja perfeito e a perturbação d seja nula, o sinal de realimentação também será nulo, isto é, $\tilde{d} = 0$. Nessa situação, a estrutura assume as características de um sistema em malha aberta. Contudo, caso $d \neq 0$, ainda considerando um modelo perfeito, é trivial concluir que $\tilde{d} = d$. Se, por fim, o modelo não for perfeito, \tilde{d} será dado pela soma da perturbação e o erro entre as saídas do modelo e o processo.

Da figura 2.1, considerando $d = 0$, pode-se definir o sinal de controle:

$$u = Q(r - \tilde{d}) \quad (2.1)$$

mas

$$\tilde{d} = (G - \tilde{G})u \quad (2.2)$$

logo

$$u = Q[r - (G - \tilde{G})u] \quad (2.3)$$

portanto,

$$u = \frac{Q}{1 + Q(G - \tilde{G})} r \quad (2.4)$$

Com a definição do sinal de controle e sabendo que, quando $d = 0$, $y = Gu$, obtém-se a função de transferência que relaciona a referência e a saída do sistema:

$$y_r = \frac{GQ}{1 + Q(G - \tilde{G})} r \quad (2.5)$$

Se a referência for nula e a perturbação $d \neq 0$, o sinal de controle é dado por:

$$u = -Q\tilde{d} \quad (2.6)$$

em que

$$\tilde{d} = (G - \tilde{G})u + d \quad (2.7)$$

portanto, substituindo a equação (2.7) na equação (2.6),

$$u = \frac{-Q}{1 + Q(G - \tilde{G})}d \quad (2.8)$$

Dessa forma, sabendo que, quando $r = 0$, $y = Gu + d$, pode-se determinar o efeito da perturbação na resposta do sistema:

$$y_d = \frac{1 - \tilde{G}Q}{1 + Q(G - \tilde{G})}d \quad (2.9)$$

Para obter a resposta do sistema quando $r \neq 0$ e $d \neq 0$, usando o princípio da superposição, faz-se $y = y_r + y_d$, ou seja:

$$y = \frac{GQ}{1 + Q(G - \tilde{G})}r + \frac{1 - \tilde{G}Q}{1 + Q(G - \tilde{G})}d \quad (2.10)$$

Como já mencionado, quando o modelo é perfeito, o sistema pode ser tratado como um sistema malha aberta. Nessa situação (quando $G = \tilde{G}$), da equação (2.10), vê-se que

$$y = \tilde{G}Qr + (1 - \tilde{G}Q)d \quad (2.11)$$

A equação (2.11) mostra que, se a planta for estável, basta que o controlador Q seja estável para garantir a estabilidade do sistema. A primeira parcela do lado direito da equação (2.11) representa como a estrutura IMC lida com o problema servo e a segunda parcela, como lida com o problema regulador. Ao analisar as parcelas separadamente, nota-se que o controlador se apresenta como um sistema em malha aberta tanto em relação a referência, quanto em relação a perturbação, o que favorece o alcance de um bom desempenho dessa estrutura perante aos dois tipos de problema. Além disso, a estrutura IMC, como será exemplificado na seção 2.4, impõe sempre a presença de um integrador no controle resultante, o que assegura erro nulo entre a referência e a saída do sistema em regime permanente.

2.3 Projeto de controlador IMC

Sabe-se que a função de sensibilidade de um sistema controlado por um controlador C é dada por:

$$S = \frac{1}{1 + CG} = \frac{y}{d} \quad (2.12)$$

A equação (2.11) mostra que, quando o modelo é perfeito ($G = \tilde{G}$),

$$\frac{y}{d} = (1 - \tilde{G}Q) \quad (2.13)$$

Para que se tenha uma perfeita rejeição à perturbação, a função de sensibilidade deve ser nula, isto é, $S = 0$. Para tanto, de acordo com a equação (2.13), basta fazer $Q = \tilde{G}^{-1}$. Assim, além da anulação da perturbação, ocorre um perfeito rastreamento da trajetória pois, da equação (2.11) nota-se que, com $\tilde{G}Q = 1$, $y = r$. Essa, contudo, é uma situação idealizada e impossível de ser implementada na prática. RIVERA; MORARI; SKOGESTAD (1986) enumeram os seguintes empecilhos para implementação de $Q = \tilde{G}^{-1}$:

- **Zeros no semi-plano direito (SPD)** - Uma vez que $Q = \tilde{G}^{-1}$, modelos com zeros no SPD resultam em controladores com polos no SPD, ou seja, controladores instáveis.
- **Atrasos no tempo** - Modelos com atrasos no tempo resultam em controladores $Q = \tilde{G}^{-1}$ com operadores de avanço no tempo e, portanto, não realizáveis.
- **Modelos estritamente próprios** - Modelos estritamente próprios resultam em controladores $Q = \tilde{G}^{-1}$ impróprios. Dessa forma, $\lim_{s \rightarrow \infty} Q(s) = \infty$, ou seja, distúrbios de frequência muito alta obrigam o atuador a produzir sinais de controle excessivamente elevados.

Nesse contexto, o projeto do controlador IMC envolve alguns procedimentos que garantem que o controlador resultante seja causal e estável. O primeiro procedimento consiste em fatorar o modelo da seguinte forma:

$$\tilde{G} = \tilde{G}_+ \tilde{G}_- \quad (2.14)$$

em que \tilde{G}_+ contem os elementos de atraso no tempo e os zeros de fase não mínima, isto é, os zeros no SPD. Assim, a inversa do elemento restante \tilde{G}_- , que não tem atrasos e nem zeros no SPD, não é instável e nem contém operadores de avanço no tempo. Portanto, fazer $Q = \tilde{G}_-^{-1}$ se torna uma alternativa melhor que $Q = \tilde{G}^{-1}$.

Existem duas maneiras de fatorar \tilde{G} (RIVERA, 1999):

$$\tilde{G}_+ = e^{-\theta} \prod_i (-\beta_i s + 1) \quad Re(\beta_i) > 0 \quad (2.15)$$

e

$$\tilde{G}_+ = e^{-\theta} \prod_i \frac{(-\beta_i s + 1)}{(\beta_i s + 1)} \quad \text{Re}(\beta_i) > 0 \quad (2.16)$$

Portanto, com \tilde{G}_+ dado pela equação (2.15) ou (2.16), obtém-se um controlador Q sem a presença de elementos de avanço no tempo ou polos instáveis no SPD. Entretanto, fazer $Q = \tilde{G}_+^{-1}$ ainda não é suficiente para garantir que controlador Q resultante seja próprio. Nesse contexto, faz-se pertinente a inserção de um filtro ao controlador Q de forma a assegurar que o mesmo tenha uma quantidade de polos igual ou superior a quantidade de zeros.

Além de garantir que o “controlador” IMC seja próprio, o filtro inserido ajuda a melhorar a robustez do sistema. Quando o modelo não é perfeito, as diferenças em relação ao processo real ocorrem geralmente em sinais de alta frequência (BROSILOW, 1979). Em virtude disso, com o objetivo de atenuar esse erro, é utilizado um filtro passa-baixa. Portanto, o “controlador” IMC pode ser definido como:

$$Q = \tilde{G}_+^{-1} T_f \quad (2.17)$$

Em que T_f é um filtro passa-baixa dado por

$$T_f = \frac{1}{(\lambda s + 1)^N} \quad (2.18)$$

Na equação (2.18), N é a ordem do filtro e o seu valor deve ser suficientemente alto para garantir que Q seja próprio. Por sua vez, λ é o parâmetro que determina a velocidade de acomodação do sinal de saída. Quanto maior é o valor de λ , maior é a constante de tempo em malha fechada e menor é a velocidade da resposta. Diminuir λ , portanto, torna a resposta mais rápida. O ajuste de λ pode ser realizado on-line para compensar o erro entre a planta e o modelo. Nesse aspecto, o aumento de λ confere maior robustez ao sistema (RIVERA, 1999).

2.4 Controladores IMC-PID e IMC-PI

Existe uma forma alternativa de representar a estrutura IMC que se equivale a estrutura mostrada na figura 2.1. Tal forma é ilustrada na figura 2.2.

Conforme mostra a figura 2.2, é possível obter o controlador clássico a partir da estrutura IMC:

$$C = \frac{Q(s)}{1 - \tilde{G}(s)Q(s)}. \quad (2.19)$$

Os controladores C resultantes da equação (2.19), dependendo de características do modelo, tais como ordem e presença de elementos de atraso, podem, como mostra as subseções 2.4.1 e 2.4.2, ter a estrutura de controladores PI e PID acrescidos ou não de

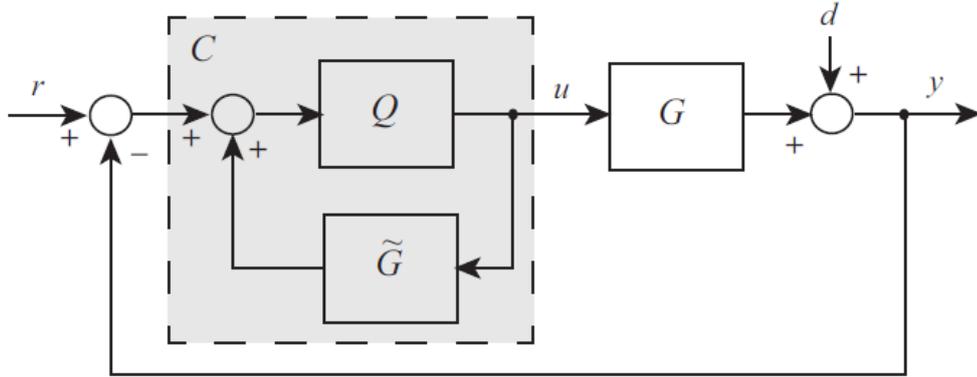


Figura 2.2: Representação esquemática da estrutura equivalente do IMC

um filtro. Tais controladores, por serem oriundos da estrutura IMC e por diferirem dos controladores PI e PID clássicos na quantidade de parâmetros de sintonia, são denominados na literatura como controladores IMC-PID, se possuírem a estrutura de um PID, e IMC-PI, se possuírem a estrutura de um PI.

2.4.1 Controlador IMC-PI

Os exemplos a seguir mostram os controladores com estrutura PI, obtidos da metodologia IMC a partir de diferentes modelos.

$$C_{PI} = k_p \left(1 + \frac{1}{\tau_i s} \right) \quad (2.20)$$

Exemplo 2.1 - Modelo de primeira ordem sem atraso e sem zeros no SPD

Considere o modelo de primeira ordem sem atraso e sem zeros no SPD:

$$\tilde{G} = \frac{K}{\tau s + 1} \quad \tau > 0 \quad (2.21)$$

Como o modelo é de primeira ordem, um filtro (equação (2.18)) com $N = 1$ garante que Q seja próprio. Dessa forma, escolhendo o método de fatoração referido na equação (2.15), o “controlador” IMC é dado por:

$$Q = \tilde{G}^{-1} T_f = \left(\frac{\tau s + 1}{K} \right) \frac{1}{(\lambda s + 1)} \quad (2.22)$$

Assim, substituindo a equação (2.22) na equação (2.19), obtem-se o controlador clássico:

$$C = \frac{\tau}{K\lambda} \left(1 + \frac{1}{\tau s} \right) \quad (2.23)$$

Comparando a equação (2.23) com a estrutura do controlador PI mostrada na equação (2.20), fica fácil inferir que:

$$k_p = \frac{\tau}{K\lambda}; \quad \tau_i = \tau \quad (2.24)$$

Exemplo 2.2 - Modelo de primeira ordem sem atraso e com zero no SPD

Vejamos agora um modelo com um zero no semi plano direito:

$$\tilde{G} = \frac{K(-\beta s + 1)}{\tau s + 1} \quad \tau, \beta > 0 \quad (2.25)$$

Novamente $N = 1$ é o suficiente para garantir que Q seja próprio. Assim, fatorando o modelo pelo critério da equação (2.15), temos:

$$Q = \tilde{G}_-^{-1} T_f = \left(\frac{\tau s + 1}{K} \right) \frac{1}{(\lambda s + 1)} \quad (2.26)$$

Os “controladores” Q exibidos na equação (2.26), em que o modelo tem um zero no SPD, e na equação (2.22), cujo modelo não tem zeros no SPD, são exatamente iguais. Entretanto, o controlador clássico resultante,

$$C = \frac{\tau}{K(\beta + \lambda)} \left(1 + \frac{1}{\tau s} \right), \quad (2.27)$$

quando comparado ao controlador obtido no exemplo 2.1, se difere em virtude da presença do zero em $1/\beta$ no modelo. Assim, de forma análoga ao exemplo 2.1, os parâmetros do controlador IMC-PI resultante são:

$$k_p = \frac{\tau}{K(\beta + \lambda)}; \quad \tau_i = \tau \quad (2.28)$$

Exemplo 2.3 - Modelo de primeira ordem com atraso e sem zeros no SPD

Consideremos um modelo de primeira ordem com atraso com a seguinte forma:

$$\tilde{G} = \frac{K}{\tau s + 1} e^{-Ls} \quad \tau > 0 \quad (2.29)$$

O controlador Q é dado por:

$$Q = \tilde{G}_-^{-1} T_f = \left(\frac{\tau s + 1}{K} \right) \frac{1}{(\lambda s + 1)} \quad (2.30)$$

Novamente, o controlador Q é igual ao obtido no exemplo 2.1. Entretanto, se analisarmos o controlador clássico, veremos que a presença do atraso o modifica. Aplicando

expansão em série de Taylor de primeira ordem para o atraso no tempo, tem-se que $e^{-Ls} \approx 1 - sL$. Dessa forma, da equação (2.19), obtém-se o seguinte controlador:

$$C = \frac{\tau}{K(L + \lambda)} \left(1 + \frac{1}{\tau s} \right) \quad (2.31)$$

ou seja,

$$k_p = \frac{\tau}{K(L + \lambda)}; \quad \tau_i = \tau \quad (2.32)$$

2.4.2 Controlador IMC-PID

Os exemplos listados nessa subseção mostram os “controladores” IMC com estrutura PID, conforme a equação:

$$C_{PID} = k_p \left(1 + \frac{1}{\tau_i s} + \tau_D s \right) \quad (2.33)$$

Exemplo 2.4 - Modelo de segunda ordem sem atraso e com zero no SPD

Consideremos o seguinte modelo de segunda ordem:

$$\tilde{G} = \frac{K(-\beta + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad \tau_1, \tau_2, \beta > 0 \quad (2.34)$$

Como o modelo é de segunda ordem, utilizando um filtro de primeira ordem, o controlador Q resultante é impróprio. No entanto, para implementação na estrutura clássica, faremos $N = 1$. Além disso, permaneceremos com a forma de fatoração referida na equação (2.15). Dessa forma, Q é dado por:

$$Q = \tilde{G}_-^{-1} T_f = \left(\frac{(\tau_1 s + 1)(\tau_2 s + 1)}{K} \right) \frac{1}{(\lambda s + 1)} \quad (2.35)$$

Assim, o controlador clássico é obtido novamente através da equação (2.19):

$$C = \frac{\tau_1 + \tau_2}{K(\beta + \lambda)} \left(1 + \frac{1}{(\tau_1 + \tau_2)s} + \frac{\tau_1 \tau_2}{\tau_1 + \tau_2} s \right) \quad (2.36)$$

Portanto, comparando com a equação (2.33), vemos que controlador PID obtido pela metodologia IMC, contem os seguintes parâmetros:

$$k_p = \frac{\tau_1 + \tau_2}{K(\beta + \lambda)}; \quad \tau_i = \tau_1 + \tau_2; \quad \tau_D = \frac{\tau_1 \tau_2}{\tau_1 + \tau_2} \quad (2.37)$$

Considerando ainda o modelo da equação (2.34), mas utilizando o método de fatoração dado pela equação (2.16), o controlador Q obtido é próprio:

$$Q = \tilde{G}_-^{-1} T_f = \left(\frac{(\tau_1 s + 1)(\tau_2 s + 1)}{K(\beta s + 1)} \right) \frac{1}{(\lambda s + 1)} \quad (2.38)$$

Utilizando a equação (2.19) para definir o controlador clássico, obtém-se um controlador que pode ser caracterizado como um PID acompanhado de um filtro:

$$C = \frac{\tau_1 + \tau_2}{K(2\beta + \lambda)} \left(1 + \frac{1}{(\tau_1 + \tau_2)s} + \frac{\tau_1\tau_2}{\tau_1 + \tau_2}s \right) \left(\frac{1}{\frac{\beta\lambda}{2\beta + \lambda}s + 1} \right) \quad (2.39)$$

Nesse caso, o controlador é analisado sob a seguinte estrutura:

$$C_{PID_f} = k_p \left(1 + \frac{1}{\tau_i s} + \tau_D s \right) f \quad (2.40)$$

Em que f , o filtro anexo ao controlador, e os demais parâmetros são dados por.

$$k_p = \frac{\tau_1 + \tau_2}{K(2\beta + \lambda)}; \quad \tau_i = \tau_1 + \tau_2; \quad \tau_D = \frac{\tau_1\tau_2}{\tau_1 + \tau_2}; \quad f = \frac{1}{\frac{\beta\lambda}{2\beta + \lambda}s + 1} \quad (2.41)$$

Exemplo 2.5 - Modelo de primeira ordem com atraso e sem zeros no SPD

A partir do modelo dado pela equação (2.29), pode-se obter um controlador clássico diferente daquele apresentado na equação (2.31) se, ao invés de usarmos expansão em série de Taylor para substituir o atraso, utilizarmos aproximação de Padé de primeira ordem. Por Padé, o elemento de atraso é aproximado da seguinte forma:

$$e^{-Ls} \approx \frac{1 - \frac{Ls}{2}}{1 + \frac{Ls}{2}} \quad (2.42)$$

Assim, o controlador clássico resultante é dado por:

$$C = \frac{L + 2\tau}{2K(L + \lambda)} \left(1 + \frac{1}{\left(\frac{L+2\tau}{2}\right)s} + \frac{L\tau}{L + 2\tau}s \right) \left(\frac{1}{\frac{\lambda L}{2(L+\lambda)}s + 1} \right) \quad (2.43)$$

O controlador encontrado possui novamente uma estrutura de PID acompanhado por um filtro. Nesse caso, para determinação dos parâmetros, ele é analisado sob a estrutura descrita na equação (2.40). Portanto, o filtro e os parâmetros do controlador são dados por:

$$k_p = \frac{L + 2\tau}{2K(L + \lambda)}; \quad \tau_i = \frac{L + 2\tau}{2}; \quad \tau_D = \frac{L\tau}{L + 2\tau}; \quad f = \frac{1}{\frac{\lambda L}{2(L+\lambda)}s + 1} \quad (2.44)$$

Como pode ser observado em todos os exemplos, os parâmetros dos controladores PI ou PID dependem exclusivamente dos parâmetros da planta e da constante de tempo λ , o único parâmetro de sintonia do projeto IMC.

2.5 IMC multivariável

Um sistema é dito multivariável se ele possui múltiplas entradas e múltiplas saídas (MIMO - do inglês *multiple input, multiple output*). A metodologia IMC pode ser aplicada em sistemas multivariáveis de forma centralizada e de forma descentralizada. Para o IMC centralizado, deve-se atentar para o desenvolvimento algébrico das equações, pois cada bloco do sistema é representado por uma matriz de transferência. Já no IMC descentralizado, ainda que cada bloco do sistema também seja representado por uma matriz de transferência, através de um bloco desacoplador adicional, o controlador visualiza o processo como um conjunto de sistemas SISO e, portanto, os procedimentos descritos na seção 2.3 podem ser aplicados diretamente para cada malha de controle individualmente.

2.5.1 IMC centralizado

Considerando a estrutura ilustrada na figura 2.1 como um sistema multivariável cujo modelo do processo possui n entradas e m saídas, conforme é descrito a seguir:

$$\tilde{G} = \begin{bmatrix} \tilde{g}_{11} & \tilde{g}_{12} & \cdots & \tilde{g}_{1n} \\ \tilde{g}_{21} & \tilde{g}_{22} & \cdots & \tilde{g}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{g}_{m1} & \tilde{g}_{m2} & \cdots & \tilde{g}_{mn} \end{bmatrix}. \quad (2.45)$$

Tem-se que:

$$u = Q(r - \tilde{d}) \quad (2.46)$$

em que

$$\tilde{d} = (G - \tilde{G})u \quad (2.47)$$

Logo,

$$u = Qr - Q(G - \tilde{G})u \quad (2.48)$$

A partir daqui é necessário atenção para o arranjo algébrico das operações matriciais, pois difere dos sistemas SISO:

$$u = [I + Q(G - \tilde{G})]^{-1}Qr \quad (2.49)$$

Como $y = Gu$,

$$y = G[I + Q(G - \tilde{G})]^{-1}Qr \quad (2.50)$$

Caso o modelo seja igual ao processo,

$$G = \tilde{G} \quad (2.51)$$

$$y = \tilde{G}Qr = \tilde{G}\tilde{G}^{-1}T_f r \quad (2.52)$$

Em que T_f é uma matriz de filtros passa-baixa com a seguinte forma:

$$T_f = \begin{bmatrix} \frac{1}{(\lambda_1 s + 1)^N} & 0 & \dots & 0 \\ 0 & \frac{1}{(\lambda_2 s + 1)^N} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{(\lambda_m s + 1)^N} \end{bmatrix}. \quad (2.53)$$

O controlador centralizado clássico pode ser obtido a partir da metodologia IMC em sistema multivariáveis da seguinte forma:

$$C = Q(s)[I - \tilde{G}(s)Q(s)]^{-1}. \quad (2.54)$$

Embora o desenvolvimento teórico do controlador centralizado seja mais direto, a implementação e a sintonia do controlador descentralizado é mais simples. A dificuldade da abordagem centralizada reside principalmente no fato de que o cálculo da inversa tanto na equação (2.52), quanto na equação (2.54), envolve o cálculo de determinantes de matrizes de funções de transferência cuja complexidade aumenta com as dimensões de $\tilde{G}(s)$. Portanto, caso a degradação do desempenho devido a descentralização do controle não seja significativa, em um sistema de grande porte, é mais conveniente usar uma solução descentralizada (WATANABE *et al.*, 1998). No caso em que o acoplamento entre as malhas de controle seja significativo, uma possibilidade para uso do controlador descentralizado é a utilização de um bloco desacoplador, como será descrito na sequência.

2.5.2 IMC descentralizado com desacoplador

Em sistemas com fortes interações entre malhas de controle, o sinal de referência de uma malha pode causar uma perturbação indesejada nas outras malhas. Nesse contexto, um método bastante empregado é a estrutura de controle descentralizado com desacoplador. Uma ilustração dessa estrutura é mostrada na figura 2.3.

Essa forma de controle consiste em alocar um bloco desacoplador precedendo o processo, gerando um modelo aproximadamente diagonal dominante. Assim, o processo é tratado como um conjunto de sistemas SISO e cada controlador pode ser sintonizado de maneira independente em relação as outras malhas de controle. Para realização dessa abordagem, os procedimentos são divididos em duas partes: primeiramente o sistema é desacoplado para minimizar as interações ou tornar o sistema diagonal dominante. Em seguida, o controlador MIMO é projetado de forma descentralizada (JUNIOR; BARROS, 2015). A figura 2.3 ilustra o controle clássico descentralizado com a presença do desacoplador. Já a forma tradicional da estrutura IMC com desacoplador, é sintetizada conforme a figura 2.4.

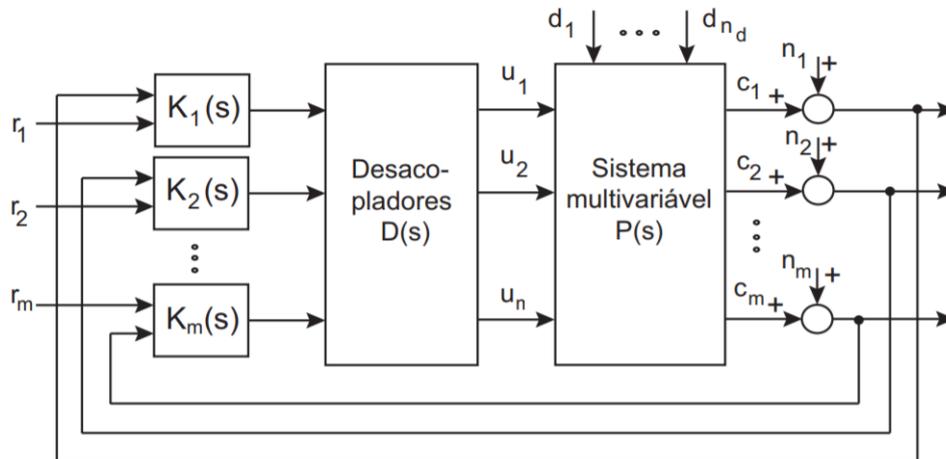


Figura 2.3: Estrutura de controle descentralizado com desacoplador

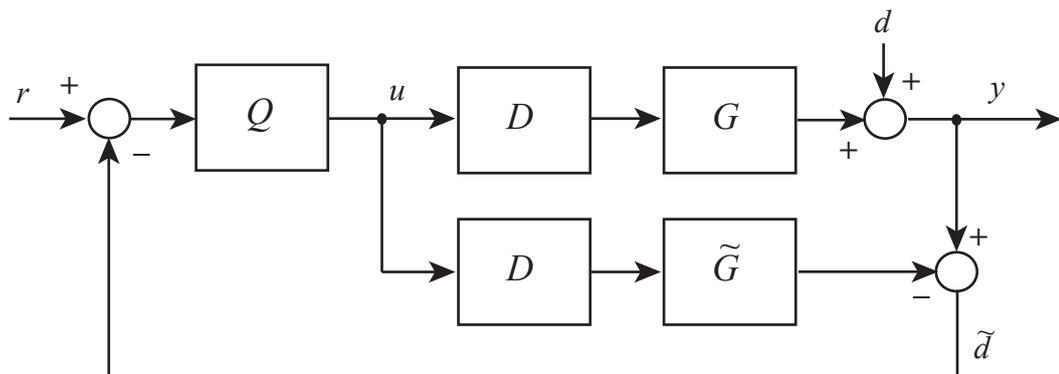


Figura 2.4: Esquema do IMC com desacoplador

A equação (2.11) mostra que, se a perturbação for nula e o modelo for igual ao processo, $y = \tilde{G}Qr$. Assim, se inserirmos um desacoplador D no sistema, obtemos:

$$y = \tilde{G}DQr. \quad (2.55)$$

Considerando o modelo desacoplado dado por $\hat{G} = \tilde{G}D$, temos

$$y = \hat{G}Qr, \quad (2.56)$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} \hat{g}_{11} & 0 & \dots & 0 \\ 0 & \hat{g}_{22} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \hat{g}_{mm} \end{bmatrix} \begin{bmatrix} q_{11} & 0 & \dots & 0 \\ 0 & q_{22} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & q_{mm} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \end{bmatrix}. \quad (2.57)$$

Em que, das equações (2.17) e (2.18), tem-se:

$$q_{ii} = \hat{g}_{-ii}^{-1} T_{fi} = \hat{g}_{-ii}^{-1} \frac{1}{(\lambda_i s + 1)^N}, \quad i = 1, \dots, m \quad (2.58)$$

2.6 Projeto IMC-PI ótimo

Considerando o modelo multivariável dado pela equação (2.45) e a estrutura IMC apresentada na Fig. 2.4, a ideia é encontrar um desacoplador que faça:

$$\hat{G} = \tilde{G}D \approx H \quad (2.59)$$

Em que H é o sistema desacoplado desejado, sem interações entre as malhas de controle, formado por uma matriz de transferência quadrada e diagonal. Quanto maior a proximidade de \hat{G} e H , menor será o acoplamento entre as malhas de controle. O modelo desejado H tem a seguinte forma:

$$H = \begin{bmatrix} h_{11} & 0 & \dots & 0 \\ 0 & h_{22} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & h_{mm} \end{bmatrix}. \quad (2.60)$$

Os elementos da matriz H podem ter a estrutura que for mais conveniente às necessidades do projetista. Aqui, para que seja possível obter um controlador que permita aplicar as regras de sintonia do IMC-PI, como aquele dado no exemplo 2.3, é conveniente que H seja composta por elementos de primeira ordem com atraso (FOPTD - do inglês *first order plus time delay*):

$$h_{jj} = \frac{\bar{K}_j}{\bar{\tau}_j s + 1} e^{-\bar{L}_j s}, \quad j = 1, \dots, m. \quad (2.61)$$

Se a diagonal principal de \tilde{G} for composta por elementos FOPTD, uma alternativa simples, e geralmente eficaz, é escolher H como essa diagonal, ou seja, $H = \text{diag}(\tilde{G})$. Consideremos agora um desacoplador com a seguinte forma:

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1m} \\ d_{21} & d_{22} & \dots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \dots & d_{nm} \end{bmatrix}. \quad (2.62)$$

Os elementos do desacoplador também podem ter as características que forem mais adequadas a cada situação. Aqui, foram trabalhadas três formas de elementos de primeira ordem:

- **Com atraso e com um zero**

$$d_{ij} = \frac{\alpha_{ij}(\gamma_{ij}s + 1)}{\Upsilon_{ij}s + 1} e^{-\theta_{ij}s}. \quad (2.63)$$

- **Com atraso e sem zeros**

$$d_{ij} = \frac{\alpha_{ij}}{\Upsilon_{ij}s + 1} e^{-\theta_{ij}s}. \quad (2.64)$$

- **Sem atraso e com um zero**

$$d_{ij} = \frac{\alpha_{ij}(\gamma_{ij}s + 1)}{\Upsilon_{ij}s + 1}. \quad (2.65)$$

Em que $i = 1, \dots, n$ e $j = 1, \dots, m$.

Existem muitas maneiras de calcular o desacoplador D , mas geralmente elas são difíceis de serem aplicadas em sistemas com $n > 2$. Nesse panorama, a proposta desse trabalho é aplicar o algoritmo de otimização Evolução Diferencial para o cálculo dos parâmetros do desacoplador. Assim sendo, α_{ij} , γ_{ij} , θ_{ij} , e Υ_{ij} são os parâmetros a serem otimizados. A função objetivo a ser minimizada é dada pela somatória das integrais do quadrado do erro entre as respostas de H e \hat{G} , considerando sinais de teste para entrada. Portanto, sendo $e_{D,i}(t)$ o erro entre a i -ésima resposta da saída de H e \hat{G} mediante a entradas de teste e a ISE desse erro dada por:

$$ISE_{D,i} \triangleq \int_0^{\infty} [e_{D,i}(t)]^2 dt = \|e_{D,i}(t)\|_2^2, \quad i = 1, \dots, m. \quad (2.66)$$

O problema de otimização para o cálculo do desacoplador pode ser definido da seguinte forma:

$$\mathcal{X}_D^* = \arg \min_{\mathcal{X}_D} \sum_{i=1}^m ISE_{D,i} \quad (2.67)$$

$$\mathcal{X}_D = \begin{cases} [\alpha_{i,j}, \gamma_{i,j}, \theta_{i,j}, \Upsilon_{i,j}], & \text{para o caso do desacoplador com atraso e com zero} \\ [\alpha_{i,j}, \theta_{i,j}, \Upsilon_{i,j}], & \text{para o caso do desacoplador com atraso e sem zeros} \\ [\alpha_{i,j}, \gamma_{i,j}, \Upsilon_{i,j}], & \text{para o caso do desacoplador sem atraso e com zeros.} \end{cases} \quad (2.68)$$

Com a inserção do desacoplador e com H composta por elementos FOPDT, é possível aplicar a equação (2.31) para calcular de forma independente cada elemento da diagonal do controlador PI descentralizado como um controlador IMC-PI utilizando cada um dos m sistemas SISO da diagonal de H :

$$k_{p,j} = \frac{\bar{\tau}_j}{\bar{K}_j(\bar{L}_j + \lambda_j)}; \quad \tau_{i,j} = \bar{\tau}_j, \quad j = 1, \dots, m. \quad (2.69)$$

Cada vetor $\Lambda = [\lambda_1 \dots \lambda_m]^T$ resulta em um controlador IMC-PI descentralizado diferente:

$$C = \begin{bmatrix} k_{p,1} \left(1 + \frac{1}{\tau_{i,1}s}\right) & 0 & \dots & 0 \\ 0 & k_{p,2} \left(1 + \frac{1}{\tau_{i,2}s}\right) & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & k_{p,m} \left(1 + \frac{1}{\tau_{i,m}s}\right) \end{bmatrix}. \quad (2.70)$$

Para calcular um vetor de constantes de tempo $\Lambda = [\lambda_1 \dots \lambda_m]^T$ ótimo, foi utilizado o algoritmo Evolução Diferencial para Otimização Multiobjetivo (DEMO - do inglês *Differential Evolution for Multiobjective Optimization*). Dessa forma, é possível alcançar diferentes “compromissos” entre a ISE do erro de rastreamento:

$$ISE_i \triangleq \int_0^\infty [(r_i(t) - y_i(t))^2] dt, \quad i = 1, \dots, m, \quad (2.71)$$

$$f_1(\Lambda) \triangleq ISE = \|r(t) - y(t)\|_2^2 = \sum_{i=1}^m ISE_i, \quad (2.72)$$

e o valor absoluto máximo do sinal de controle:

$$f_2(\Lambda) \triangleq \|u(t)\|_\infty = \max_j \sup_{t \in [0, \infty]} |u_j(t)|. \quad (2.73)$$

Portanto, a obtenção dos parâmetros ótimos do controlador IMC-PI é, nesse caso, um problema de otimização multiobjetivo definido da seguinte forma:

$$\Lambda^* = \arg \min_{\Lambda} \begin{bmatrix} \|r(t) - y(t)\|_2^2 \\ \|u(t)\|_\infty \end{bmatrix} \quad (2.74)$$

Ao término da execução do DEMO, o usuário tem a sua disposição um conjunto de soluções ótimas, conhecido como conjunto de Pareto, para escolher a mais apropriada. Soluções que correspondem a sistemas instáveis em malha-fechada serão descartadas no processo de seleção do algoritmo DEMO por apresentarem valores elevados para a função objetivo.

2.7 Conclusões

No presente capítulo, discorreu-se a respeito da metodologia de controle IMC, frisando os detalhes inerentes à sua estrutura, os procedimentos de projeto e, principalmente, a descrição de como os controladores obtidos com essa metodologia podem ter a estrutura de controladores PI ou PID. Para tal, foram abordados exemplos que, além de ilustrar os procedimentos empregados na técnica IMC, permitem observar a relação dos controladores IMC-PI e IMC-PID com as características do modelo e com o parâmetro λ do IMC.

A possibilidade de se obter controladores com estrutura PI ou PID através da metodologia IMC, se mostra vantajosa por diminuir a quantidade de parâmetros a serem definidos por malha de controle. Em sistemas SISO os controladores PID demandam a determinação do ganho proporcional, do tempo integral e do tempo derivativo enquanto, utilizando a metodologia IMC, é necessária apenas a determinação da constante de tempo λ . Dessa forma, em sistemas MIMO descentralizados, o número de parâmetros é multiplicado pelo número de malhas de controle. Em termos práticos, em um sistema com m malhas de controle, são necessários $3m$ parâmetros para os controladores PID e, para os controladores IMC, apenas m parâmetros.

Com intuito de se obter controladores com estrutura PI sem a presença de um filtro adicional, nesse trabalho são abordados somente os sistemas MIMO clássicos da literatura que permitam, após um desacoplamento satisfatório, a obtenção de modelos compostos apenas por funções de transferência de primeira ordem, com atraso, em sua diagonal principal. Portanto, o sistema desejado com a realização do desacoplamento contém apenas modelos como o descrito no exemplo 2.3 da seção 2.4. Em virtude disso, os controladores IMC-PI obtidos nos exemplos ilustrativos têm os parâmetros definidos conforme a equação (2.32). No próximo capítulo serão descritas as técnicas de otimização utilizadas nesse trabalho para resolver os problemas de otimização descritos na seção 2.6.

Otimização Multiobjetivo

3.1 Introdução

Problemas que requerem a melhor decisão, dependendo o menor custo ou tendo o menor ônus, estão presentes no cotidiano das pessoas, estejam elas conscientes disso ou não. VAN VELDHUIZEN; LAMONT (2000) ilustram esse fato, mencionando que os problemas podem estar em uma simples escolha de uma roupa, que pode ser resolvido em poucos segundos, ou em um complexo projeto de engenharia, que pode levar anos para ser resolvido. Como exemplo desse tipo de problema, os autores citam o processo de compra de um carro no qual o comprador deseja minimizar o custo financeiro do veículo e do seguro, bem como deseja maximizar o “prazer” proporcionado pelo veículo. Ademais, o cliente espera que o veículo satisfaça as seguintes condições: tenha acomodações confortáveis para seis adultos, tração nas quatro rodas, um sistema de som com qualidade *premium*, pintura azul ou preta e, por fim, um rendimento de, no mínimo, 75 milhas por galão. Nesse contexto, pode-se dizer que os veículos disponíveis podem ser considerados as *variáveis de decisão*. Já as condições impostas, como quantidade de lugares, tração nas quatro rodas, qualidade do sistema de som, cor da pintura e rendimento, podem ser consideradas as *restrições* do problema. Por sua vez, aquilo que se deseja minimizar ou maximizar são as *funções objetivo*. Observa-se que esse problema possui mais de uma função objetivo e, portanto, é chamado de Problema de Otimização Multiobjetivo (MOP, do inglês, *Multiobjective Optimization Problem*).

TAKAHASHI (2011) formaliza o conceito de otimização como a obtenção da “melhor solução” de problemas que permitem determinar o quão bem cada solução satisfaz à necessidade em questão. Mais abrangente, o problema de otimização multiobjetivo é definido por OSY CZKA (1985) da seguinte forma:

“um vetor de variáveis de decisão que satisfaz as restrições e otimiza uma função vetorial cujos elementos representam as funções objetivo. Essas funções

formam uma descrição matemática dos critérios de desempenho que geralmente estão em conflito uns com os outros. Portanto, o termo “otimizar” significa encontrar uma solução que forneça os valores de todas as funções objetivo aceitáveis para que a decisão seja tomada”.

Nesse trabalho, foi empregado o algoritmo de otimização multiobjetivo DEMO (do inglês, *Differential Evolution for Multiobjective Optimization*), que usa as técnicas do NSGA-II (do inglês, *Non-dominated Sorting Genetic Algorithm*) para realização do truncamento da população durante a seleção. Em virtude disso, tanto o DEMO quanto o NSGA-II serão abordados de forma mais detalhada em subseções específicas. Antes, porém, faz-se pertinente um estudo relacionado ao método de Evolução Diferencial mono-objetivo (DE, do inglês, *Diferencial Evolution*), que juntamente com o NSGA-II, possui conceitos necessários à compreensão do DEMO. Além disso, na proposta abordada aqui, o DE mono-objetivo é utilizado para o desenvolvimento dos desacopladores. Dessa forma, é realizada uma fundamentação do DE na subseção 3.2. Posteriormente, na seção 3.3, uma descrição mais aprofundada acerca do NSGA-II e do DEMO são apresentadas.

3.2 Algoritmo Evolução Diferencial

O algoritmo Evolução Diferencial foi desenvolvido por Storn e Price na década de noventa. A ideia fundamental por trás do DE é o mecanismo de busca através da mutação diferencial, que extrai informações de distância e direção da população e propicia excelentes propriedades de convergência (STORN; PRICE, 1995).

O algoritmo DE, além do mecanismo de mutação diferencial, utiliza operadores que também estão presentes no algoritmo genético, como a recombinação e a seleção. Através desses operadores é alcançada a solução ótima $\mathbf{x}^* \in \mathbb{R}^\eta$, que minimiza $f(\mathbf{x}) : \mathbb{R}^\eta \mapsto \mathbb{R}$ (STORN; PRICE, 1997; DAS; MULLICK; SUGANTHAN, 2016).

Seja $\mathcal{U}_{(a,b)} \in \mathbb{R}$ um número pseudo-aleatório uniformemente distribuído no intervalo (a,b) ; $\mathcal{I}_{(m)} \in \mathbb{Z}^+$ um inteiro pseudo-aleatório uniformemente distribuído em $1, \dots, m$ e a população de soluções $X_G = \{x_{i,G} ; i = 1, \dots, N\}$ da geração G , de tamanho N sendo que, para um problema com η parâmetros, cada solução é dada por:

$$x_{i,G} = \begin{bmatrix} x_{i,1,G} \\ \vdots \\ x_{i,\eta,G} \end{bmatrix} \quad (3.1)$$

O algoritmo DE pode ser dividido nas etapas descritas nas subseções 3.2.1, 3.2.2, 3.2.3 e 3.2.4.

3.2.1 População inicial

No início do DE, é gerada uma população inicial que cobre toda região de busca. Para tal, caso não se tenha conhecimento do sistema, utiliza-se normalmente uma distribuição de probabilidade uniforme aleatória, cujos limites são dados pelo mínimo e máximo de cada variável, $x_{i,j,1} \in [\underline{m}_j, \overline{m}_j]$:

$$x_{i,j,1} = \mathcal{U}_{(\underline{m}_j, \overline{m}_j)}, \quad (3.2)$$

$$i = 1, \dots, N, j = 1, \dots, \eta.$$

Caso não exista um limite para as variáveis, pode-se definir \mathcal{U} como a região inicial de busca.

Segundo STORN (1996), definir a população inicial com $N = 10\eta$ indivíduos garante bons resultados para a maioria dos problemas. Entretanto, CHEN *et al.* (2012) e KAZIMIPOUR; LI; QIN (2013) mostram que populações com um número alto de indivíduos podem trazer desvantagens, como alto esforço computacional e não convergência para o mínimo ótimo. O problema de otimização para definição do desacoplador pode ter, dependendo da estrutura escolhida para os elementos de sua matriz, 3 ou 4 variáveis de otimização por malha de controle, como mostra a seção 2.6 do capítulo 2. Nesse aspecto, com base nas referências mencionadas e em testes realizados previamente, que mostraram bons resultados em termos de esforço computacional e minimização da função objetivo, foi adotado nesse trabalho uma população com $N = 3\eta$ indivíduos.

3.2.2 Mutaç o diferencial

O primeiro passo do algoritmo DE é a mutaç o diferencial, mecanismo que adiciona a diferenç a ponderada de dois indivíduos aleat rios a um terceiro indivíduo. Essa operaç o é realizada N vezes para gerar a populaç o mutante:

$$v_{i,G} = x_{r_1,G} + F_i \cdot (x_{r_2,G} - x_{r_3,G}) \quad (3.3)$$

em que

- $v_{i,G}$ é o i -ésimo vetor mutante da G -ésima geraç o,
- r_1, r_2 e r_3 s o inteiros definidos aleatoriamente de forma que $r_1 \neq r_2 \neq r_3 \neq i$,
- F_i é um n mero real que pondera a diferenç a dos vetores. Nesse trabalho foi adotado $F_i = \mathcal{U}_{(0.5,1)}$, como sugere STORN (1996).

A figura 3.1 ilustra o mecanismo de mutaç o diferencial para geraç o do vetor mutante $v_{i,G}$ em um espaç o de par metros bidimensional.

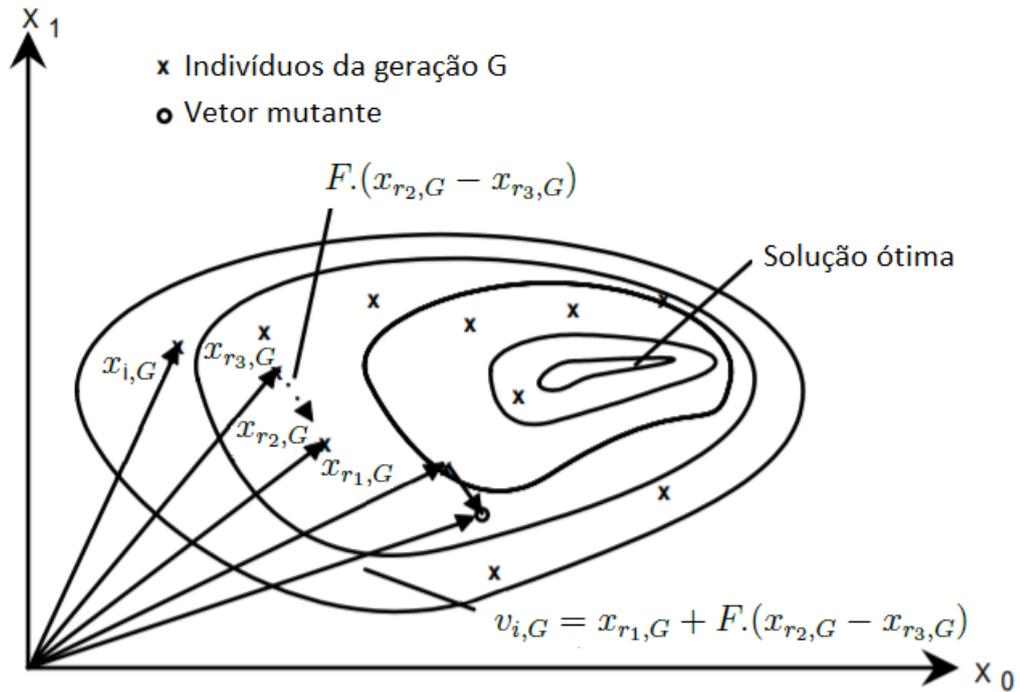


Figura 3.1: Esquema do mecanismo de mutação diferencial (adaptado de STORN; PRICE (1995))

3.2.3 Recombinação

Após o processo de mutação, mediados por um fator aleatório, a população mutante, V_G , é combinada com a população ancestral, X_G , cuja i -ésima solução denominada de vetor alvo, para gerar a população teste U_G em um procedimento conhecido como recombinação ou cruzamento.

A recombinação é descrita por:

$$u_{i,j,G} = \begin{cases} v_{i,j,G} & \text{se } \mathcal{U}_{(0,1)} \leq C \text{ ou } j = \delta_i \\ x_{i,j,G} & \text{caso contrário} \end{cases} \quad (3.4)$$

em que

- $C \in [0, 1]$ é a probabilidade de recombinação escolhida pelo usuário. Quanto mais perto de 1, maior a chance da variável da solução mutante compor a solução teste. De acordo com STORN (1996), $C \in [0.8, 1]$ garante bons resultados para a maioria dos problemas. Portanto, com base em STORN (1996) e em testes que avaliassem a escolha, nesse trabalho foi adotado $C = 0,9$.
- $\delta_i = \mathcal{I}_{(\eta)} \in \{1, \dots, \eta\}$ é um inteiro gerado aleatoriamente, que garante que pelo menos uma variável da solução mutante irá compor a solução teste, isto é, $u_{i,G} \neq x_{i,G}$.

Nota-se que, para que $v_{i,j,G}$, um elemento do vetor contido na população mutante, seja inserido em $u_{i,j,G}$, é necessário que o fator aleatório $\mathcal{U}_{(0,1)}$ seja menor ou igual a C ou que a posição j no vetor seja igual a δ_i . Essa operação é ilustrada na figura 3.2.

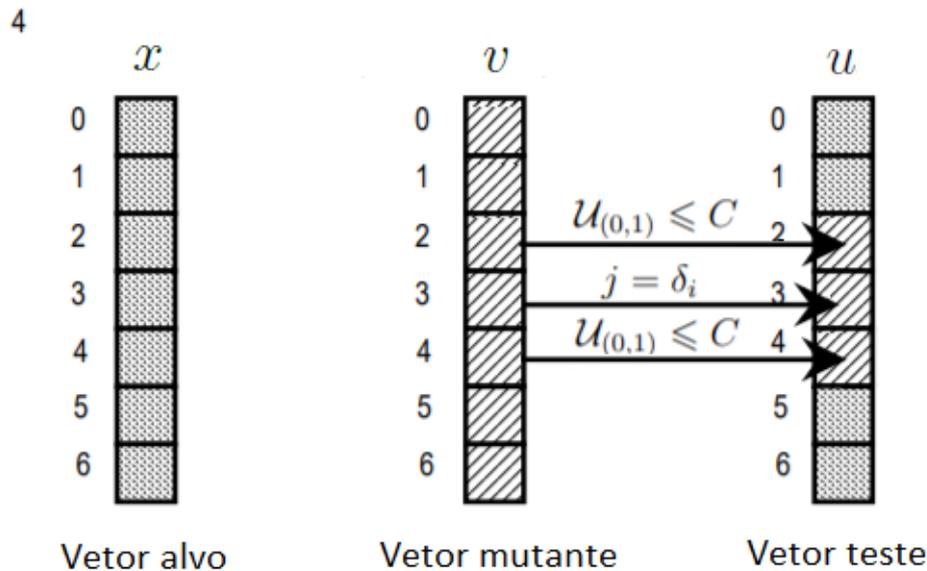


Figura 3.2: Ilustração do processo de recombinação para um problema com 7 parâmetros (adaptado de STORN; PRICE (1997))

A operação de recombinação é ilustrada na figura 3.2. Nessa figura, a solução teste gerada possui três variáveis da solução mutante, sendo duas decorrentes da avaliação da probabilidade e uma forçada a partir do índice aleatório δ_i , nesse caso igual a três.

3.2.4 Seleção

Finalizando a rotina do DE, ocorre o processo de seleção. Nesse processo todos os indivíduos da população formada por vetores teste são comparados com os respectivos indivíduos da população ancestral, ou seja, a população formada por vetores alvo. Se o i -ésimo vetor teste produzir um valor de função objetivo menor do que o i -ésimo vetor alvo da população ancestral, o vetor recém-gerado substituirá o vetor com o qual ele foi comparado. Tal processo é descrito por:

$$x_{i,G+1} = \begin{cases} u_{i,G} & \text{se } f(u_{i,G}) < f(x_{i,G}) \\ x_{i,G} & \text{caso contrário} \end{cases} \quad (3.5)$$

Em que f representa a função objetivo. O resultado desse procedimento é a geração da próxima população ancestral X_{G+1} .

3.3 Algoritmos evolucionários multiobjetivos

As técnicas de otimização multiobjetivo geralmente oferecem um conjunto de soluções eficientes ou não dominadas conhecido como conjunto de Pareto. Uma solução \mathbf{x}^* de um problema multiobjetivo é eficiente se não existe nenhuma outra solução factível \mathbf{x} com $f(\mathbf{x}) \leq f(\mathbf{x}^*)$ e $f(\mathbf{x}) \neq f(\mathbf{x}^*)$. Para os vetores $X = [x_1 \dots x_k]^T$ e $Y = [y_1 \dots y_k]^T$, $X \leq Y$, se $x_i \leq y_i$ com $i = 1, \dots, k$, e $X \neq Y$ se $\exists i \mid x_i \neq y_i$. O problema de otimização multiobjetivo se resume em encontrar o conjunto de Pareto em que:

$$\mathcal{X}^* = \{\mathbf{x}^* \in \mathcal{F}_x : \nexists \mathbf{x} \in \mathcal{F}_x \mid f(\mathbf{x}) \leq f(\mathbf{x}^*); f(\mathbf{x}) \neq f(\mathbf{x}^*)\} \quad (3.6)$$

Sendo \mathcal{F}_x o conjunto de soluções factíveis.

Os algoritmos evolucionários são eficazes em problemas multiobjetivo, principalmente, por lidar simultaneamente com uma população de soluções que podem formar uma amostra do conjunto de Pareto. Assim, é possível encontrar o conjunto ótimo de Pareto executando o algoritmo apenas uma vez, não tendo, portanto, a necessidade de uma série de execuções separadas, como no caso de outras técnicas tradicionais que não consideram população de soluções (COELLO, 1999).

A primeira abordagem de Algoritmo Evolucionário Multiobjetivo (MOEA, do inglês, *Multiobjective Evolutionary Algorithm*) foi feita por SCHAFFER (1985), que propôs o VEGA (do inglês, *Vector Evaluation Genetic Algorithm*). O algoritmo apresentado por Schaffer é uma modificação do algoritmo genético convencional com uma adaptação para múltiplos objetivos. A diferença em relação ao algoritmo original é a criação de um *loop* que faz o procedimento de seleção ser repetido para cada objetivo separadamente até que um determinado número de indivíduos seja alcançado. Em seguida, estes indivíduos são escolhidos de forma aleatória para que sejam realizadas as etapas de recombinação e mutação. O principal problema dessa metodologia é, em virtude da realização da seleção de maneira individual para cada objetivo, a especialização da população (CASTRO, 2001). Tal aspecto se mostra desvantajoso por resultar em curvas de Pareto com baixa diversidade e que, em consequência disso, tem soluções que tendem a satisfazer mais a um objetivo em detrimento dos outros.

HORN; NAFPLIOTIS; GOLDBERG (1994) propuseram o NPGA (do inglês, *Niched Pareto Genetic Algorithm*), que faz uso de um método de seleção por torneio denominado Torneio de Pareto. Tal método consiste em comparar duas soluções com um grupo estabelecido aleatoriamente de forma prévia (t_{don}). Se um indivíduo é dominado e o outro não, a solução não dominada é selecionada. Se ocorre empate o parâmetro de nicho σ_{share} , que é baseado no número de soluções localizadas a uma determinada distancia, é atribuído a todos os membros da população. Desse modo, é selecionada a solução cujo σ_{share} possui

menor valor. Pode-se dizer que a implementação da variável de nicho é o aspecto mais interessante do trabalho, pois visa uma maior diversidade da população. Entretanto, tal aspecto também pode ser encarado como uma desvantagem, uma vez que a incorporação de dois novos parâmetros (t_{don} e σ_{share}) a serem calculados no algoritmo genético cria uma dificuldade a mais para o problema de otimização.

SRINIVAS; DEB (1994) introduziram o NSGA (do inglês, *Non-dominated Sorting Genetic Algorithm*), cuja principal característica é ter o processo de seleção realizado com base em um ordenamento da população de acordo com a dominância. As principais críticas ao NSGA foram enumeradas em DEB *et al.* (2000):

- Alto esforço computacional para o ordenamento
- Falta de elitismo, que, quando ausente, aumenta a possibilidade de se perder boas soluções
- Necessidade de especificar o parâmetro de nicho σ_{share} .

Nesse mesmo trabalho, além de listar os problemas do NSGA, propõem-se o NSGA-II, a evolução do NSGA, que, aqui, será abordado na subseção 3.3.1.

KNOWLES; CORNE (1999) sugeriram um método que usa a estratégia de evolução com elitismo, o PAES (do inglês, *Pareto Archived Evolution Strategy*). Nessa abordagem, o descendente é comparado com os ascendentes. Se o descendente dominar o ascendente, o descendente passa a ser o próximo ascendente e a iteração continua. Por outro lado, se o ascendente dominar o descendente, o descendente é descartado e uma nova solução é estabelecida. No entanto, se o descendente e os ascendentes não se dominam, o descendente é comparado com um conjunto de soluções não dominadas. Se o descendente dominar qualquer indivíduo desse conjunto, ele passa a ser o novo ascendente e o indivíduo dominado é eliminado do conjunto. Se o descendente não dominar nenhum membro do conjunto, é realizada uma análise do espaçamento do ascendente e do descendente em relação as soluções desse conjunto. Se o descendente estiver em uma localização com menor aglomeração de indivíduos, ele será aceito como ascendente e uma cópia será adicionada ao conjunto de soluções não dominadas (DEB *et al.*, 2000).

ZITZLER; LAUMANN; THIELE (2001) propuseram o SPEA-II (do inglês, *Strength Pareto Evolutionary Algorithm*), uma evolução do SPEA, apresentado por ZITZLER; THIELE (1999). No SPEA-II, uma população externa armazena as soluções não dominadas até a iteração corrente. Qualquer membro dessa população é eliminado em duas situações: se for encontrada uma solução que o domine ou se o tamanho máximo da população for excedido. Ser inserido na população externa é, normalmente, a única forma de permanecer no processo ao longo das gerações. A aptidão dos indivíduos é baseada

primeiramente na dominância, isto é, na quantidade de soluções que dominam ou são dominadas por cada indivíduo. As soluções não dominadas recebem o índice 0. As soluções dominadas, por sua vez, recebem um índice igual ao somatório do número de soluções dominadas por cada indivíduo que a domina. Em seguida, para refinar a aptidão atribuída à cada indivíduo, é considerada a densidade de soluções circunvizinhas.

Outro método que vale ser citado é o MOEA/D (do inglês, *Multiobjective Evolutionary Algorithm Based on Decomposition*), que foi desenvolvido por ZHANG; LI (2007). A principal característica desse método é a decomposição do problema de otimização multiobjetivo em N subproblemas de otimização escalar. Dessa forma, todos os subproblemas são otimizados simultaneamente e a melhor solução de cada subproblema forma a população em cada geração. Durante a otimização dos subproblemas são utilizadas apenas informações de alguns subproblemas vizinhos, fazendo com que o MOEA/D demande um menor esforço computacional, até mesmo quando comparado com outros métodos populares, como o NSGA-II.

3.3.1 Algoritmo NSGA-II

DEB *et al.* (2000) apresentou uma versão aperfeiçoada do NSGA, o NSGA-II, cujo princípio básico é a ordenação elitista baseada na dominância. No NSGA-II, a partir de uma população P , por meio de processos presentes nos algoritmos genéticos convencionais, como cruzamento e mutação, é gerada uma população Q . As duas populações são reunidas em um único grupo, $R = P \cup Q$, que é ordenado de acordo com um operador chamado “índice de fronteira”, que armazena a quantidade de membros da população que dominam cada solução. Dessa forma, são estabelecidas fronteiras $F_1, F_2 \dots F_n$, em que cada fronteira tem membros com o mesmo índice de fronteira. Nesse contexto, F_1 contém os indivíduos não dominados pelos demais. F_2 contém os indivíduos dominados por um ou mais indivíduos contidos somente em F_1 , ou seja, não dominados por $R - F_1$. Por sua vez, F_3 , contém indivíduos dominados somente por outros contidos em $F_1 \cup F_2$ e, por consequência, necessariamente não dominados por qualquer indivíduo contido em $R - (F_1 \cup F_2)$. O procedimento de classificação que, quando aplicado a uma população R , retorna uma lista das fronteiras F de acordo com o índice de fronteira de cada solução, é denominado *fast-nondominated-sort* e sua implementação é realizada através do algoritmo 1, apresentado na sequência (DEB *et al.*, 2002). Um segundo critério de seleção, usado para comparar duas soluções na mesma fronteira, é a distância de aglomeração.

As soluções contidas em cada fronteira são inseridas em uma nova população, que formará a próxima geração P . Posto que P_i é a i -ésima população, a adição de fronteiras ocorre enquanto o tamanho de P_{i+1} é menor ou igual ao tamanho de P_i . Se, ao inserir as soluções da n -ésima fronteira F_n , o tamanho de P_{i+1} é excedido em relação ao tamanho

de P_i , utilizando um parâmetro denominado *crowding distance*, ou distância de aglomeração, ocorre um truncamento de F_n para que as soluções menos espaçadas entre si, isto é, localizadas em uma região com maior aglomeração de indivíduos, sejam eliminadas. Tal procedimento consiste em ordenar os membros da fronteira a ser truncada em relação às distâncias de cada indivíduo aos demais. Em seguida, as soluções contidas em uma região com menor densidade de indivíduos são adicionadas à P_{i+1} de forma que, ao término, P_{i+1} e P_i tenham o mesmo tamanho. A distância de aglomeração, que representa a estimativa da aglomeração de soluções em torno de um determinado ponto na população, é dada pela média, considerando todas as funções objetivos, dos perímetros dos maiores cuboides que envolvem o ponto i sem incluir nenhum outro ponto na população. A Figura 3.3 mostra graficamente a distância de aglomeração da i -ésima solução, contida na fronteira delimitada pelos indivíduos identificados por círculos cheios. O cuboide é representado pela linha tracejada. O pseudocódigo para obtenção da distância de aglomeração é apresentado no algoritmo 2. Nele, I é uma determinada fronteira, $I_{d,i}$ é a distância de aglomeração de uma solução i contida na fronteira I , $I_{m,i}$ é o valor de i para a m -ésima função objetivo e os parâmetros f_m^{max} e f_m^{min} , respectivamente, são o maior e menor valor da m -ésima função objetivo.

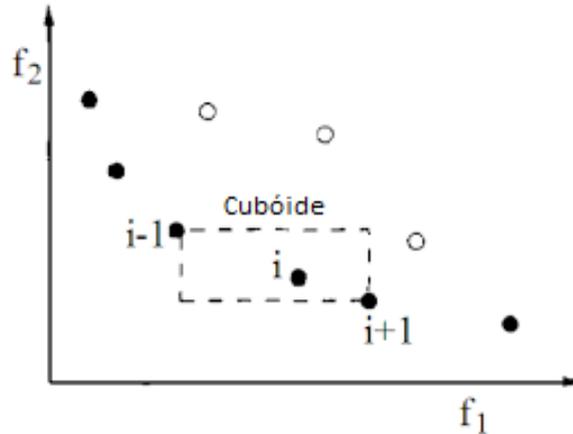


Figura 3.3: Distância de aglomeração (reproduzido de DEB *et al.* (2002))

Considera-se, assumindo que todo indivíduo i possui dois atributos, o índice de fronteira (i_{rank}) e a distância de aglomeração (i_d), o operador \prec definido da seguinte forma:

$$i \prec j \text{ se } (i_{rank} < j_{rank}) \text{ ou } [(i_{rank} = j_{rank}) \text{ e } (i_d > j_d)] \quad (3.7)$$

De uma perspectiva mais geral, o que define quais indivíduos serão inseridos na população P_{i+1} é o processo de seleção, que, como já abordado aqui, no caso do NSGA-II, é baseado no índice de fronteira e distância de aglomeração. Após a seleção e, assim, a formação completa de P_{i+1} , inicia-se a próxima iteração da rotina do NSGA-II, quando,

Algoritmo 1: FAST-NONDOMINATED-SORT

```

1  $F_1 = \emptyset$  ▷ Conjunto de soluções da fronteira 1
2 para cada  $p \in P$  e  $q \neq p$  faça
3    $S_p = \emptyset$  ▷ Conjunto de soluções dominadas por  $p$ 
4    $n_p = 0$  ▷ Número de soluções que dominam  $p$ 
5   para cada  $q \in P$  faça
6     se  $p \prec q$  ▷ Se  $p$  domina  $q$ ,
7       então
8          $S_p = S_p \cup \{q\}$  ▷ adiciona  $q$  ao conjunto de soluções dominadas por  $p$ 
9       fim
10      senão se  $q \prec p$  então
11         $n_p = n_p + 1$  ▷ Incrementa o contador de soluções que dominam  $p$ 
12      fim
13    fim
14    se  $n_p = 0$  então
15       $p_{rank} = 1$  ▷  $p$  pertence à primeira fronteira
16       $F_1 = F_1 \cup p$ 
17    fim
18  fim
19   $i = 1$  ▷ Inicializa o contador de fronteiras
20  enquanto  $F_i \neq \emptyset$  faça
21     $Q = \emptyset$  ▷ Guarda membros da próxima fronteira
22    para cada  $p \in F_i$  faça
23      para cada  $q \in S_p$  faça
24         $n_q = n_q - 1$ 
25        se  $n_q = 0$  então
26           $q_{rank} = i + 1$  ▷  $q$  pertence à próxima fronteira
27           $Q = Q \cup \{q\}$ 
28        fim
29      fim
30    fim
31     $i = i + 1$ 
32     $F_i = Q$ 
33  fim

```

Algoritmo 2: CROWDING DISTANCE

```

1  $l = |I|$  ▷  $l$  Recebe o número de soluções da fronteira  $I$ ,  $|I|$ 
2 para cada  $i$  faça
3    $I_{d,i} = 0$  ▷ Inicializa as distâncias de aglomeração
4 fim
5 para cada objetivo  $m$  faça
6   ordena  $I$  de acordo com o  $m$ -ésimo objetivo
7    $I_{d,1} = I_{d,l} = \infty$ 
8   para  $i = 2$  até  $l - 1$  faça
9      $I_{d,i} = I_{d,i-1} + (I_{m,i} - I_{m,i-1}) / (f_m^{max} - f_m^{min})$ 
10  fim
11 fim

```

novamente através das operações de cruzamento e mutação, a partir de P_{i+1} , é gerado Q_{i+1} . A operação de seleção do NSGA-II pode ser melhor compreendida pela figura 3.4.

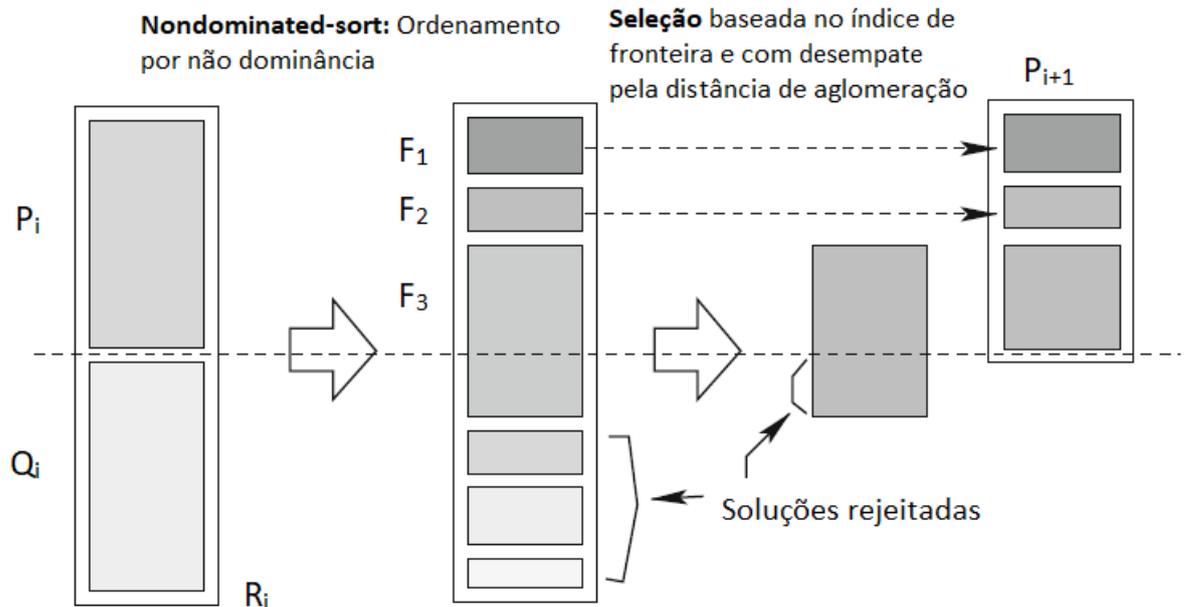


Figura 3.4: Operação de seleção do NSGA-II (baseado em DEB *et al.* (2002))

3.3.2 Algoritmo DEMO

ROBIČ; FILIPIČ (2005) introduziram o DEMO, uma metodologia que, diferente de algumas mencionadas aqui, quais sejam, VEGA, NPGA, PAES, SPEA-II e NSGA-II, não usa algoritmos genéticos para geração de novos indivíduos, mas sim, o algoritmo Evolução Diferencial. A diferença entre o DE mono-objetivo e o DEMO é o processo de seleção:

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G}, & \text{se } f(\mathbf{u}_{i,G}) \leq f(\mathbf{x}_{i,G}) \text{ e } f(\mathbf{u}_{i,G}) \neq f(\mathbf{x}_{k,i}) \\ \mathbf{x}_{i,G}, & \text{caso contrário} \end{cases}, \quad (3.8)$$

A solução teste, $u_{i,G}$, a partir de um indivíduo, a solução alvo, $x_{i,G}$ contido em uma população X_G da geração G , de tamanho N , é gerada pelos processos de mutação e cruzamento do DE. Essa solução teste é a candidata a entrar na população original X_G . Se a solução teste domina $x_{i,G}$, a solução teste substitui $x_{i,G}$. Se $x_{i,G}$ domina a solução teste, a mesma é descartada e $x_{i,G}$ é mantida em X_k . Se a solução teste e $x_{i,G}$, no entanto, não se dominam, a solução teste é inserida na população sem que $x_{i,G}$ seja descartada. Nesse caso, a população original tem o seu tamanho aumentado de 1. Observa-se, portanto, que uma solução pode ser inserida na população por substituição, quando a solução teste domina a solução alvo, ou por adição, quando alvo e teste não se dominam. O funcionamento dessa metodologia pode ser melhor compreendido através do algoritmo 3.

Algoritmo 3: DEMO

```

1  $|X_G| = N$ 
2 Gerar aleatoriamente a população inicial  $X_k$ 
3 enquanto o critério de parada não é satisfeito faça
4   para cada  $x_{i,G} \in X_G$  ( $i = 1, 2 \dots N$ ) faça
5     Cria a solução candidata  $C$  (através do DE)
6     Calcula o valor de cada objetivo em função de  $C$ 
7     se  $C$  domina  $x_{i,G}$  então
8        $C$  substitui  $x_{i,G}$ 
9     fim
10    se  $x_{i,G}$  domina  $C$  então
11       $C$  é descartado
12    fim
13    senão
14       $C$  é adicionado a  $X_{G+1}$  ▷ Aumenta a população de 1
15    fim
16  fim
17  se  $|X_{G+1}| > N$  então
18     $X_{G+1}$  ▷ Utilizando índice de fronteira e distância de aglomeração
19  fim
20 fim

```

Após a operação de seleção, se o número de soluções de X_{G+1} é maior que N , para selecionar as N melhores soluções de X_{G+1} , o algoritmo DEMO aplica o índice de fronteira para ordenar a população dentro de diferentes índices de não-dominância e, quando necessário, a distância de aglomeração para truncar as soluções excedentes da mesma forma que o NSGA-II (DEB *et al.*, 2002).

Cabe destacar que a utilização do índice de fronteira e da distância de aglomeração durante a etapa de seleção permite um povoamento mais uniforme das soluções resultantes,

gerando, como desejado, uma população mais diversificada (ROBIČ; FILIPIČ, 2005).

3.4 Conclusões

Esse capítulo foi iniciado com uma breve abordagem das idéias que envolvem problemas de otimização mono e multiobjetivos em um âmbito geral. Em seguida, restringindo o conteúdo a um campo mais específico e mais pertinente ao presente trabalho, a discussão foi direcionada para os Algoritmos Evolucionários Multiobjetivos, onde, iniciando a seção 3.3, foi realizada uma revisão acerca de algumas das produções mais relevantes do tema na comunidade científica.

Para solucionar os problemas de otimização presentes nesse trabalho, será utilizado o DE para o problema mono-objetivo, isto é, o desenvolvimento dos desacopladores, e o DEMO para a determinação de parâmetros ótimos para os controladores IMC-PI, que nesse trabalho é um problema multiobjetivo.

Exemplos Ilustrativos

Nesse capítulo, as metodologias de síntese de controladores apresentadas nessa dissertação serão empregadas em plantas multivariáveis clássicas da literatura. Mais precisamente, serão realizadas aplicações da metodologia IMC centralizada e do controlador IMC-PI ótimo descentralizado. Nesse contexto, os procedimentos de síntese desses controladores serão detalhados passo a passo em cada exemplo e, em seguida, os resultados alcançados, bem como as respectivas análises, serão apresentadas.

4.1 IMC centralizado ótimo aplicado ao modelo de quatro tanques

Para mostrar o desempenho do controlador IMC centralizado e, por outro lado, destacar a complexidade e a dificuldade prática de sua aplicação quando comparado ao controle descentralizado com desacoplador, cujas aplicações são abordadas nas seções 4.3 e 4.4, um exemplo desse tipo de controlador será abordado nessa seção. Para tal, será utilizado o exemplo de controle de nível de um sistema com quatro tanques, ilustrado na figura 4.1. As entradas do sistema são as tensões nas bombas v_1 e v_2 e as saídas, os níveis nos tanques 1 e 2 (y_1 e y_2). JOHANSSON (2000) obteve dois modelos referentes a diferentes pontos de operação relacionados às válvulas de três vias presentes no sistema. Como um dos modelos possui um zero de fase não mínima, será considerado nesse trabalho apenas o modelo que possui apenas zeros de fase mínima:

$$\tilde{G} = \begin{bmatrix} \frac{2,6}{62s+1} & \frac{1,5}{(23s+1)(62s+1)} \\ \frac{1,4}{(30s+1)(90s+1)} & \frac{1,6}{91s+1} \end{bmatrix}. \quad (4.1)$$

4.1.1 Resposta em malha aberta

Com intuito de analisar o sistema e, posteriormente, através de comparação, avaliar as melhorias alcançadas com o controlador proposto, o modelo foi simulado em malha

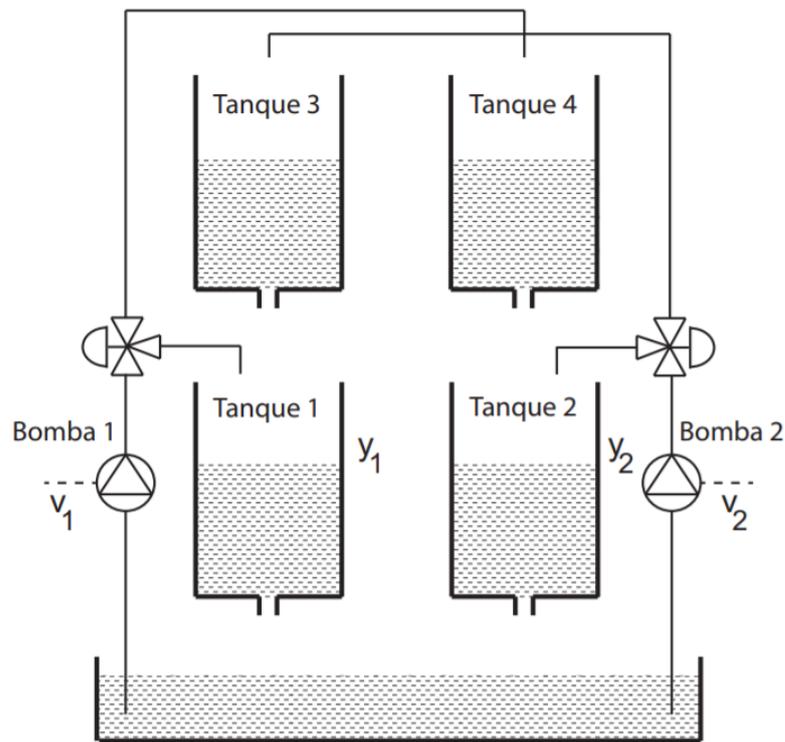


Figura 4.1: Esquema do sistema de quatro tanques (adaptado de JOHANSSON (2000))

aberta. Consideremos $\mathbf{1}(t - \tau)$ a função degrau unitário com $\mathbf{1}(t - \tau) = 0$ para $t < \tau$ e $\mathbf{1}(t - \tau) = 1$ para $t \geq \tau$. Assim sendo, foram aplicados os sinais de entrada $u_1 = \mathbf{1}(t)$ e $u_2 = \mathbf{1}(t - 600)$. A resposta ao degrau do sistema de quatro tanques é mostrada na figura 4.2.

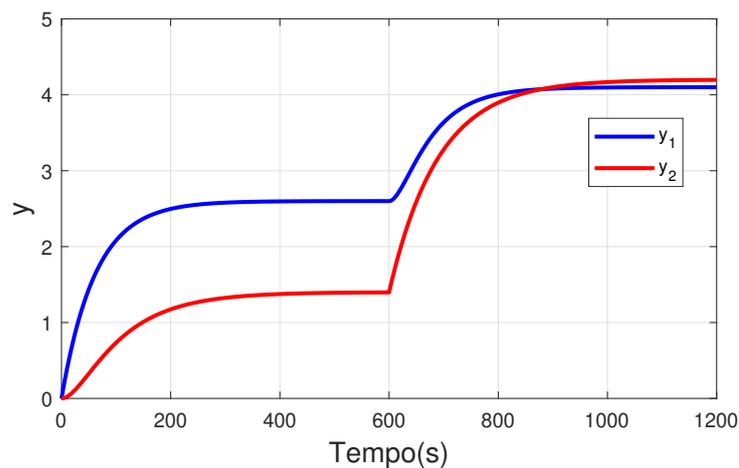


Figura 4.2: Resposta ao degrau do sistema em malha aberta - Modelo quatro tanques

A figura 4.2 mostra que existe interação entre as malhas de controle desse sistema. Haja vista a perturbação sofrida por cada malha quando ocorre variação do sinal de entrada na outra.

4.1.2 Síntese do controlador IMC centralizado

Para o desenvolvimento do controlador Q centralizado, é necessária a definição das constantes de tempo da matriz T_f . Como esse exemplo se trata de um modelo 2×2 , existem duas variáveis a serem determinadas, λ_1 e λ_2 . Para tal, foi utilizado o algoritmo DEMO com uma população inicial estabelecida com $N = 30$ indivíduos, considerando o problema de otimização multiobjetivo definido pela equação (2.74). Com os sinais de referências dados por $r_1 = \mathbf{1}(t)$ e $r_2 = \mathbf{1}(t - 300)$, após a execução do algoritmo, foi gerada a curva de Pareto, cujas soluções estabelecem diferentes compromissos entre dois objetivos de controle em um espaço de objetivos, conforme mostra a figura 4.3.

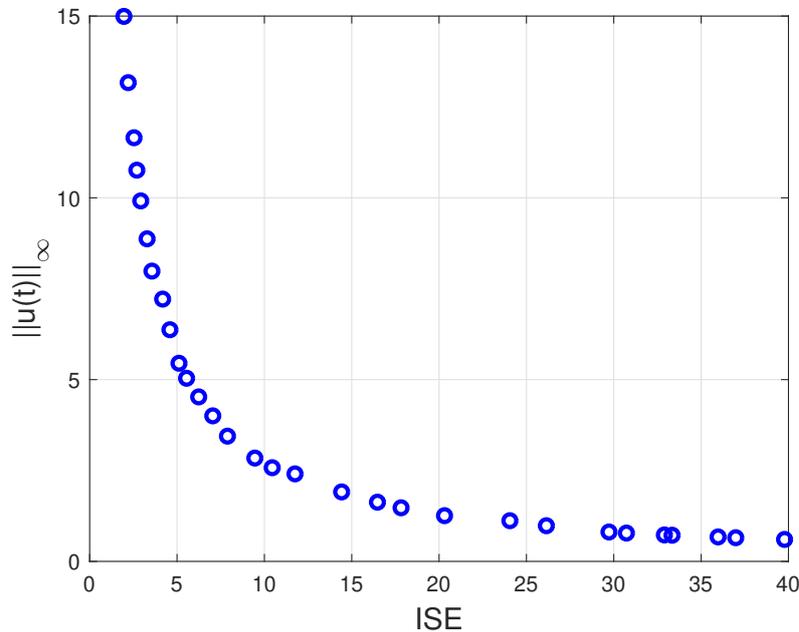


Figura 4.3: Conjunto de soluções candidatas de Pareto - Modelo quatro tanques

A figura 4.3 é composta por pontos que representam os valores das constantes de tempo em malha fechada $\Lambda = [\lambda_1 \ \lambda_2]^T$. O ponto na extremidade superior esquerda corresponde a solução com menor ISE de rastreamento e o maior esforço de controle. A solução na extremidade inferior direita corresponde ao maior ISE e menor esforço de controle.

Com intuito de ilustrar diferenças entre sistemas de controle centralizados e descentralizados, os resultados obtidos serão comparados com o controlador PI descentralizado desenvolvido por JOHANSSON (2000). A tabela 4.1 destaca os resultados alcançados através do controlador referente à 1ª solução mostrada na figura 4.3, que possui a menor ISE; do controlador referente à 15ª solução, que domina o controlador de JOHANSSON (2000), ou seja, possui menor ISE e menor esforço de controle e, por fim, os resultados obtidos com o controlador de JOHANSSON (2000).

Os vetores de constantes de tempo são dados por $\Lambda = [1,6077 \ 2,1059]^T$, referente à

Tabela 4.1: Desempenho dos controladores - Modelo quatro tanques

	1 ^a s. P.	15 ^a s.P.	Johansson (2000)
$\ u(t)\ _\infty$	14,9915	2,8426	3,0000
ISE1	0,8549	4,2453	4,0308
ISE2	1,1038	5,2105	5,6761
ISE	1,9586	9,4558	9,7069

1^a solução, e $\Lambda = [8,3902 \ 10,3206]^T$, referente à 15^a solução. Os “controladores IMC” Q obtidos utilizando essas constantes de tempo são dados por:

$$Q_{1^a} = \begin{bmatrix} \frac{23,85s^3+2,216s^2+0,0641s+0,0005574}{1,608s^3+1,123s^2+0,07847s+0,001031} & \frac{-0,8063s^2-0,03583s-0,0002986}{2,106s^3+1,162s^2+0,07898s+0,001031} \\ \frac{-0,3974s^2-0,02369s-0,0002787}{1,608s^3+1,123s^2+0,07847s+0,001031} & \frac{32,14s^3+2,826s^2+0,07402s+0,0005176}{2,106s^3+1,162s^2+0,07898s+0,001031} \end{bmatrix} \quad (4.2)$$

e

$$Q_{15^a} = \begin{bmatrix} \frac{23,85s^3+2,216s^2+0,0641s+0,0005574}{8,39s^3+1,644s^2+0,08546s+0,001031} & \frac{-0,8063s^2-0,03583s-0,0002986}{10,32s^3+1,793s^2+0,08745s+0,001031} \\ \frac{-0,3974s^2-0,02369s-0,0002787}{8,39s^3+1,644s^2+0,08546s+0,001031} & \frac{32,14s^3+2,826s^2+0,07402s+0,0005176}{10,32s^3+1,793s^2+0,08745s+0,001031} \end{bmatrix}. \quad (4.3)$$

Para obter a forma clássica dos controladores centralizados, basta utilizar a equação (2.54). Foi utilizado a função *balreal* do Matlab para obter a realização balanceada e isolar o vetor de Hankel. Através da função *modred*, o modelo foi reduzido, eliminando os elementos do vetor de Hankel cujos valores eram menor que $1 \cdot 10^{-3}$, pois representam estados com contribuições insignificantes para a resposta do sistema. Os controladores obtidos foram:

$$C_{1^a} = \begin{bmatrix} \frac{14,83s^4+1,379s^3+0,03987s^2+0,0003467s-5,097 \cdot 10^{-12}}{s^4+0,07681s^3+0,001031s^2-1,986 \cdot 10^{-11}s-6,923 \cdot 10^{-23}} \\ \frac{-4,577 \cdot 10^{-8}s^4-0,2472s^3-0,01474s^2-0,0001734s+8,834 \cdot 10^{-12}}{s^4+0,07681s^3+0,001031s^2-1,986 \cdot 10^{-11}s-6,961 \cdot 10^{-22}} \\ \frac{5,654 \cdot 10^6s^4-0,3829s^3-0,01702s^2-0,0001418s+2,73 \cdot 10^{-12}}{s^4+0,07681s^3+0,001031s^2-1,986 \cdot 10^{-11}s-6,923 \cdot 10^{-23}} \\ \frac{15,26s^4+1,342s^3+0,03515s^2+0,0002458s-4,732 \cdot 10^{-12}}{s^4+0,07681s^3+0,001031s^2-1,986 \cdot 10^{-11}s-6,961 \cdot 10^{-22}} \end{bmatrix} \quad (4.4)$$

e

$$C_{15^a} = \begin{bmatrix} \frac{2,842s^4+0,2641s^3+0,00764s^2+6,644 \cdot 10^{-5}s-1,993 \cdot 10^{-13}}{s^4+0,07681s^3+0,001031s^2-3,805 \cdot 10^{-12}s-4,017 \cdot 10^{-23}} \\ \frac{-6,634 \cdot 10^{-10}s^4-0,04737s^3-0,002824s^2-3,322 \cdot 10^{-5}s+3,454 \cdot 10^{-13}}{s^4+0,07681s^3+0,001031s^2-3,805 \cdot 10^{-12}s-2,194 \cdot 10^{-22}} \\ \frac{1,885 \cdot 10^{-7}s^4-0,07812s^3-0,003472s^2-2,893 \cdot 10^{-5}s+1,068 \cdot 10^{-13}}{s^4+0,07681s^3+0,001031s^2-3,805 \cdot 10^{-12}s-4,017 \cdot 10^{-23}} \\ \frac{3,114s^4+0,2738s^3+0,007172s^2+5,015 \cdot 10^{-5}s-1,85 \cdot 10^{-13}}{s^4+0,07681s^3+0,001031s^2-3,805 \cdot 10^{-12}s-2,194 \cdot 10^{-22}} \end{bmatrix}. \quad (4.5)$$

Em virtude da complexidade de Q , que possui elementos de terceira ordem, o cálculo do controlador clássico, por envolver a inversão de matrizes, não é trivial e o seu resultado é difícil de ser implementado na prática. Isso é evidenciado ao observar as equações (4.4) e (4.5), que representam os controladores obtidos.

As respostas ao degrau obtidas com os controladores referidos nas equações (4.4) e (4.5), juntamente com as repostas obtidas por JOHANSSON (2000), são mostradas nas

figuras 4.4 e 4.5. Os gráficos do esforço de controle necessário para gerar as saídas apresentadas nas figuras 4.4 e 4.5 são mostrados nas figuras 4.6 e 4.7.

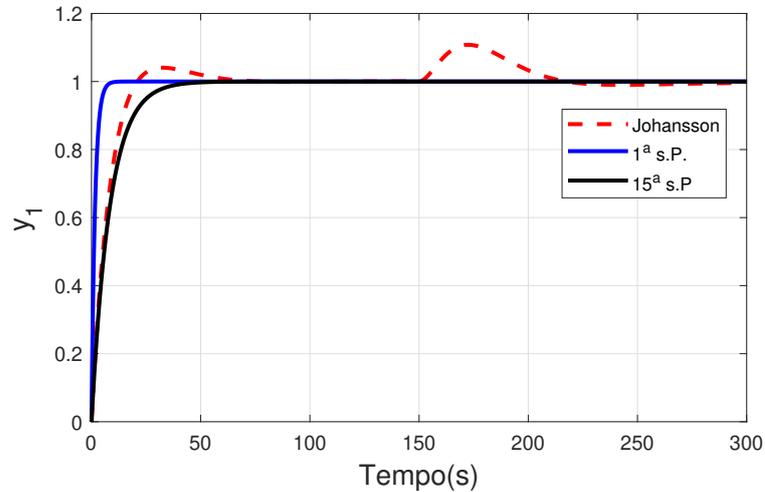


Figura 4.4: Resposta ao degrau y_1 - Modelo quatro tanques

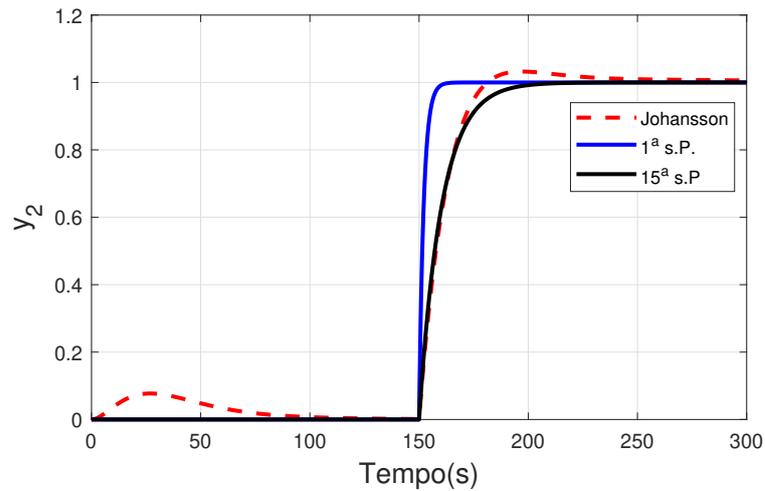
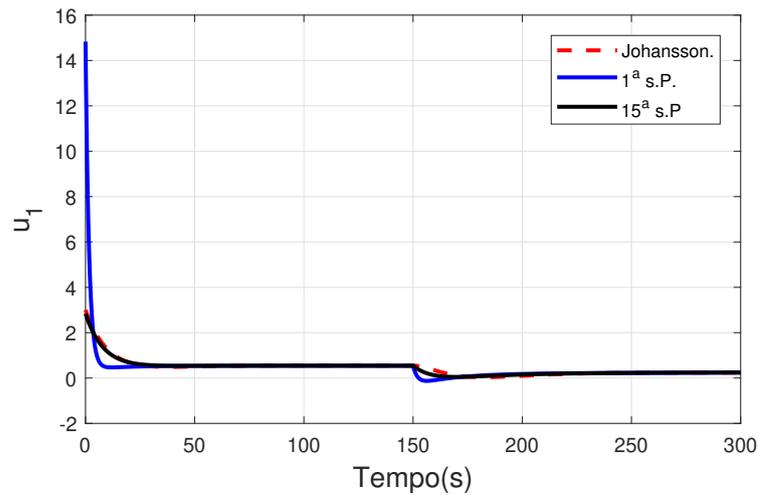
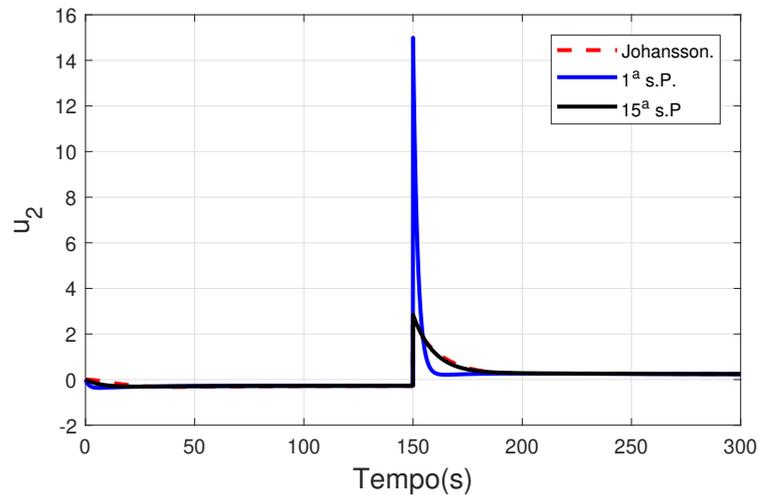


Figura 4.5: Resposta ao degrau y_2 - Modelo quatro tanques

As figuras 4.4 e 4.5 mostram que, quando comparadas às repostas em malha aberta, com a utilização da metodologia IMC, a saída do sistema diminui significativamente o seu tempo de acomodação e tem erro nulo em regime permanente. Ademais, como esse sistema não possui elementos de atraso e zeros de fase não mínima, ao fazer $y = \tilde{G}\tilde{G}_-^{-1}T_f r$, a planta é completamente anulada, o que resulta em $y = T_f r$. Dessa forma, uma vez que T_f é escolhido como uma matriz diagonal composta apenas por funções de transferência de primeira ordem com polos reais estáveis, as respostas do sistema são completamente desacopladas e não possuem *overshoots*. Ademais, nota-se que o desempenho alcançado com o controlador IMC centralizado é superior ao controlador PI descentralizado de JOHANSSON (2000).

Figura 4.6: Esforço de controle u_1 - Modelo quatro tanquesFigura 4.7: Esforço de controle u_2 - Modelo quatro tanques

As figuras 4.6 e 4.7 mostram que, utilizando o 15º controlador, o esforço de controle é pouco menor em relação ao controlador de JOHANSSON (2000). Já com o 1º controlador, a resposta transitória é muito mais rápida, porém a amplitude do sinal de controle alcança valores elevados, podendo gerar saturação do atuador.

É importante observar que a abordagem do IMC centralizado se mostra menos eficaz em plantas com atraso ou zeros de fase não mínima. Como os elementos de atraso e os zeros de fase não mínima não permitem que a planta seja completamente anulada ao se fazer $y = \tilde{G}\tilde{G}^{-1}T_fr$, modelos com interações entre as malhas mantêm um certo grau acoplamento após a implementação do controlador. Outra dificuldade é que, em se tratando de sistemas MIMO, os zeros no semi-plano direito podem não aparecer explicitamente na matriz de transferência. Dessa forma, “separar” os zeros do modelo, conforme a equação (2.14), se torna um procedimento mais complexo. Além disso, plantas representadas por

matrizes de ordem alta demandam maior esforço computacional para se obter a inversa e podem gerar controladores de grande complexidade e difíceis de serem implementados. Por fim, para se obter um controlador clássico a partir do IMC centralizado, caso exista elementos de atraso, é necessário fazer uma aproximação. Essa aproximação, contudo, gera um erro, que pode se propagar durante os cálculos algébricos matriciais (equação (2.54)) que envolvem o desenvolvimento desse controlador e, assim, causar uma resposta indesejada para o sistema. Nesse contexto, faz-se pertinente para esses casos, a utilização do controle descentralizado com desacoplador, que permite, além de obter controladores mais simples, atenuar o acoplamento de forma mais satisfatória. Nas seções seguintes, portanto, controladores IMC-PI com desacoplador serão abordados no sistema de quatro tanque e em outros exemplos contendo atraso e interação entre as malhas.

4.2 IMC-PI ótimo descentralizado aplicado ao modelo quatro tanques

Com intuito de estabelecer uma comparação com o controle IMC centralizado, serão abordados nessa seção os procedimentos e os resultados da aplicação do controlador IMC-PI descentralizado sobre o modelo de quatro tanques apresentado na equação 4.1.

4.2.1 Desenvolvimento do desacoplador

Como mencionado anteriormente, para o projeto do desacoplador, é necessário definir o modelo que se deseja obter através do desacoplamento. Portanto, definindo o modelo desejado, H , como a diagonal principal do modelo original, tem-se:

$$h_{jj} = g_{jj}, \quad j = 1,2 \quad (4.6)$$

$$H = \begin{bmatrix} \frac{2,6}{62s+1} & 0 \\ 0 & \frac{1,6}{91s+1} \end{bmatrix}. \quad (4.7)$$

Considerando que o modelo não possui retardo no tempo, é possível alcançar um bom desacoplamento através da estrutura de desacoplador indicada pela equação ((2.65)). O DE foi aplicado para solucionar o problema de otimização mono-objetivo dado pelas equações (2.68) e (2.67). A estrutura escolhida para o desacoplador demanda 3 variáveis de otimização por malha de controle. Portanto, considerando que esse exemplo se trata de um modelo 2×2 , a população é definida com $N = 3\eta = 3 \times 4 \times 3 = 36$ indivíduos. Dessa forma, o seguinte desacoplador foi obtido:

$$D = \begin{bmatrix} \frac{63,32s+1,411}{68,15s+1} & \frac{-2,346s-0,7966}{42,63s+1} \\ \frac{-2,257s-0,7026}{55,44s+1} & \frac{49,19s+1,392}{54,66s+1} \end{bmatrix}. \quad (4.8)$$

Na figura 4.8 é apresentada a resposta ao degrau do modelo desejado, H , e do modelo obtido, $\hat{G} = \tilde{G}D$. Analisando essa figura, é notório que o desacoplador obtido através do algoritmo de otimização DE assegurou que a resposta ao degrau do modelo desacoplado possua erro quase nulo em relação à resposta ao degrau do modelo desejado. Isso permite que sejam projetados controladores IMC-PI específicos para cada função de transferência da matriz desejada H .

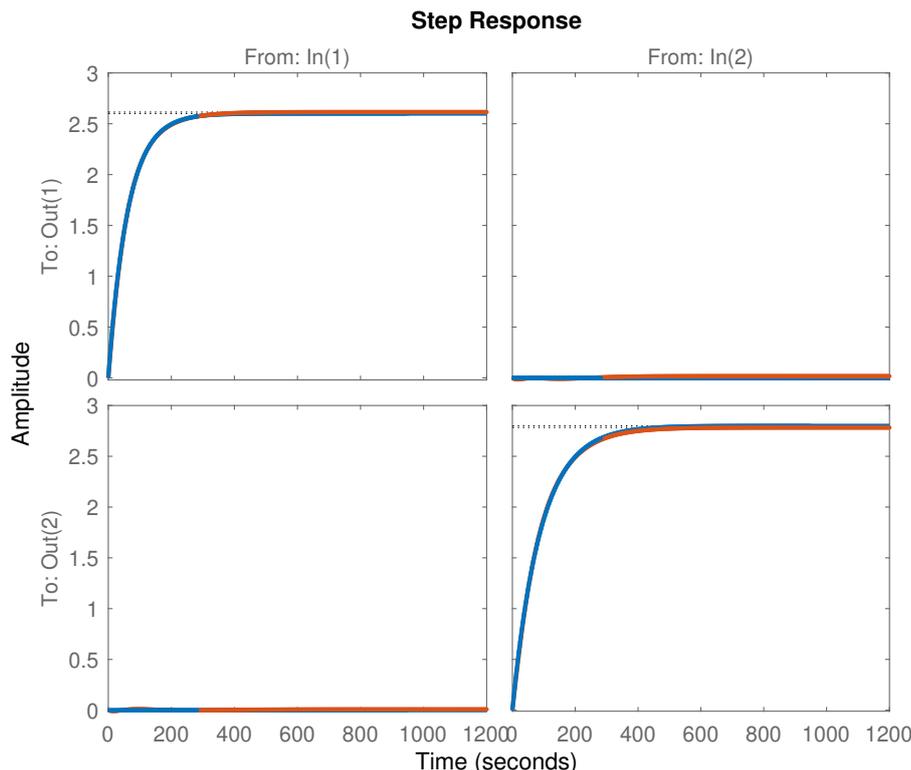


Figura 4.8: Resposta ao degrau de H , em azul, e \hat{G} , em vermelho - Modelo quatro tanques

4.2.2 Síntese do controlador IMC-PI

Utilizando o algoritmo DEMO, com população de 30 indivíduos, e o modelo desejado dado pela equação (4.7), o problema de otimização definido pela equação (2.74) é solucionado. Como resultado, é gerada uma curva de Pareto contendo as soluções candidatas a compor o controlador IMC-PI descentralizado aplicado ao sistema. Cada ponto exibido na figura 4.9 representa um controlador IMC-PI calculado a partir de um vetor $\Lambda = [\lambda_1 \ \lambda_2]^T$. Assim como no controle centralizado, o ponto na extremidade superior esquerda representa a solução que propicia o ISE de rastreamento mínimo em detrimento da minimização do esforço de controle, enquanto a solução na extremidade inferior direita, o contrário, ou seja, assegura o menor pico de esforço de controle, em detrimento da minimização da ISE.

Entre as soluções apresentadas na figura 4.9, a 16ª solução, da esquerda para direita,

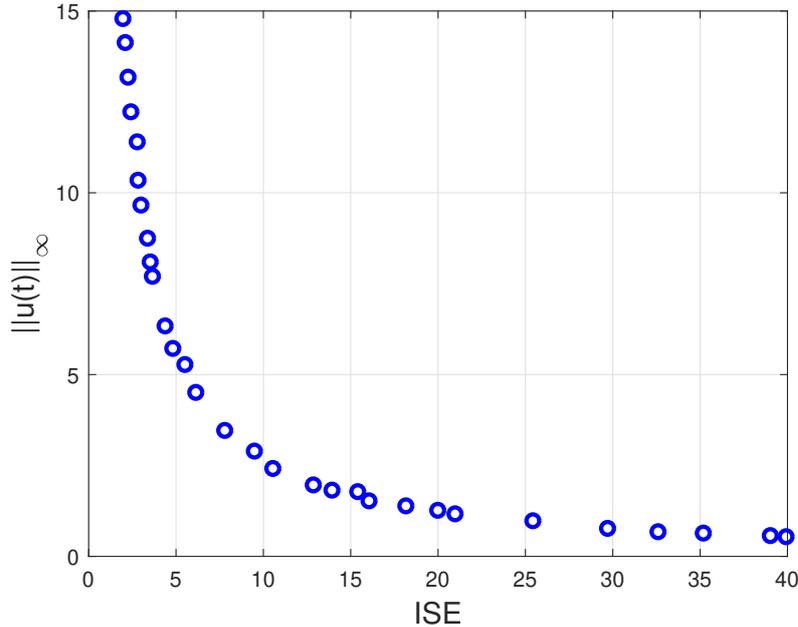


Figura 4.9: Conjunto de soluções candidatas de Pareto para controlador descentralizado - Modelo quatro tanques

$\Lambda = [8,7640 \ 9,1298]^T$, possui menor ISE de rastreamento e menor pico de esforço de controle em relação as respostas obtidas com o controlador de JOHANSSON (2000). A tabela 4.2 mostra os valores da ISE e da $\|u(t)\|_\infty$ proporcionados pelos parâmetros encontrados na 16ª solução do controlador IMC-PI centralizado (figura 4.9). Para comparação, os índices de desempenho obtidos com o controlador de JOHANSSON (2000), bem como os resultados auferidos com a 15ª solução do controlador IMC centralizado (figura 4.3) também serão exibidos na tabela 4.2.

A tabela 4.2 mostra que, embora o controlador IMC-PI descentralizado sintonizado com os parâmetros da 16ª solução da figura 4.2 supere o controlador de JOHANSSON (2000) nos dois objetivos, o desempenho do controlador IMC centralizado com parâmetros da 15ª solução da figura 4.3 é um pouco superior. Essa pequena superioridade era esperada, uma vez que, como o modelo não possui atrasos ou zeros de fase não mínima, como explicado na seção anterior, a planta é completamente anulada e o modelo real se torna exatamente igual ao modelo desejado, que não possui acoplamento. Quando se utiliza o IMC centralizado em plantas com atraso ou zeros não mínimos, contudo, os modelos resultantes mantêm um acoplamento considerável e resultam em uma ISE de rastreamento elevada.

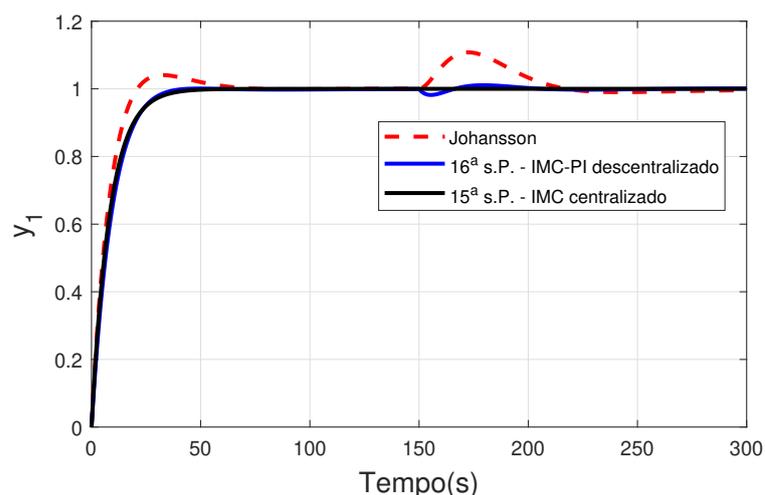
O controlador IMC-PI gerado com os parâmetros referentes a 16ª solução é dado por:

$$C_{16^a} = \begin{bmatrix} 0,36739 \left(1 + \frac{1}{62s}\right) & 0 \\ 0 & -0,1433 \left(1 + \frac{1}{91s}\right) \end{bmatrix} \quad (4.9)$$

Tabela 4.2: Desempenho dos controladores descentralizados IMC-PI - Modelo quatro tanques

	16 ^a s. P. - IMC-PI descentralizado	15 ^a s.P. - IMC centralizado	Johansson (2000)
$\ u(t)\ _\infty$	2,8949	2,8426	3,0000
ISE1	4,6076	4,2453	4,0308
ISE2	4,8786	5,2105	5,6761
ISE	9,4862	9,4558	9,7069

As respostas obtidas com o controlador IMC-PI mostrado na equação (4.9) juntamente com as respostas obtidas com o controlador IMC centralizado e o controlador de JOHANSSON (2000) são exibidas nas figuras 4.10, 4.11, 4.12 e 4.13. É possível notar que os resultados obtidos com o controlador IMC-PI são bem próximos aos resultados alcançados com o controlador IMC centralizado. A principal diferença entre eles está nas pequenas perturbações ocorridas nas malhas de controle quando é utilizado o controlador IMC-PI descentralizado. Tais perturbações, decorrentes do pequeno acoplamento existente entre as malhas, faz com que ISE de rastreamento seja maior no controle IMC-PI descentralizado. Embora o controlador IMC centralizado tenha apresentado melhores resultados sob os critérios analisados ($\|u(t)\|_\infty$ e ISE de rastreamento), ele, além de ser mais complexo, não sustenta esse desempenho em plantas com atraso ou zeros não mínimos, uma vez que não é possível “anular” a planta fazendo $\tilde{G}\tilde{G}^{-1}T_f r$. Nesse contexto, portanto, nos exemplos a seguir, cujos modelos contêm elementos de atraso no tempo, serão abordados apenas os controladores IMC-PI descentralizados com desacoplador.

Figura 4.10: Resposta ao degrau y_1 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques

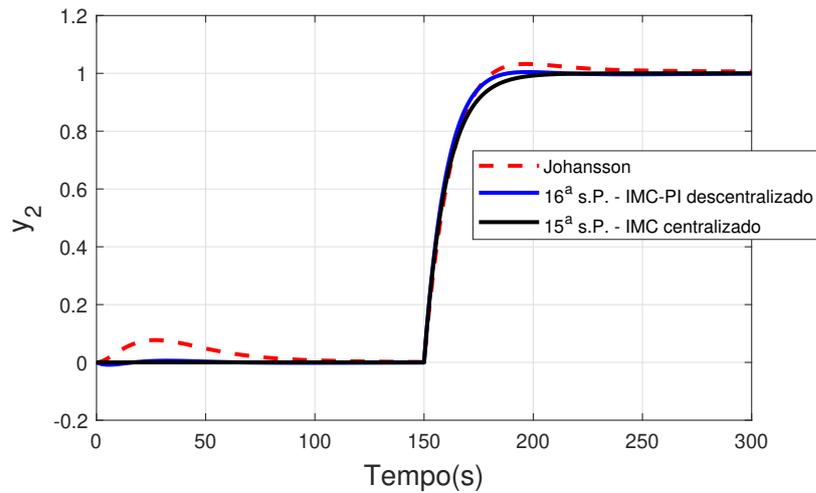


Figura 4.11: Resposta ao degrau y_2 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques

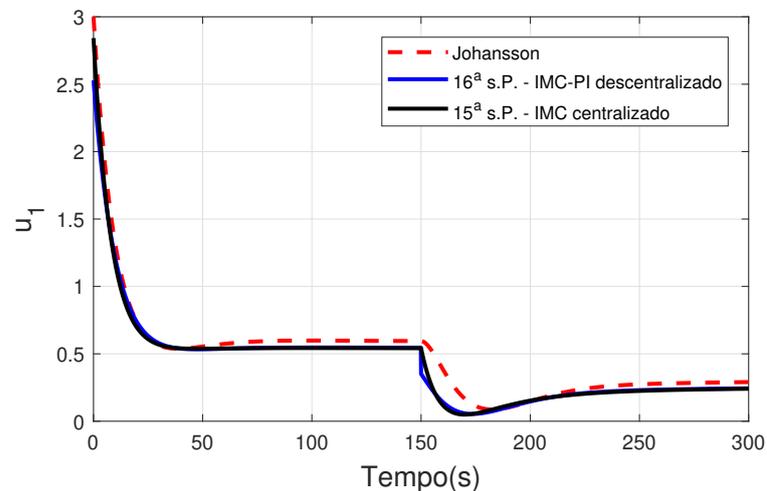


Figura 4.12: Esforço de controle u_1 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques

4.3 IMC-PI ótimo descentralizado aplicado ao modelo Wood-Berry

Apresentado em WOOD; BERRY (1973), o modelo Wood - Berry é uma representação simplificada de uma coluna de destilação que separa metanol e água. O modelo desse sistema é dado por:

$$\tilde{G} = \begin{bmatrix} \frac{12,8}{16,7s+1}e^{-s} & \frac{-18,9}{21s+1}e^{-3s} \\ \frac{6,6}{10,9s+1}e^{-7s} & \frac{-19,4}{14,4s+1}e^{-3s} \end{bmatrix}. \quad (4.10)$$

As entradas do sistema são a vazão mássica de refluxo e a vazão mássica de vapor em lb/min, sendo o minuto a unidade de tempo padrão. As variáveis controladas são as

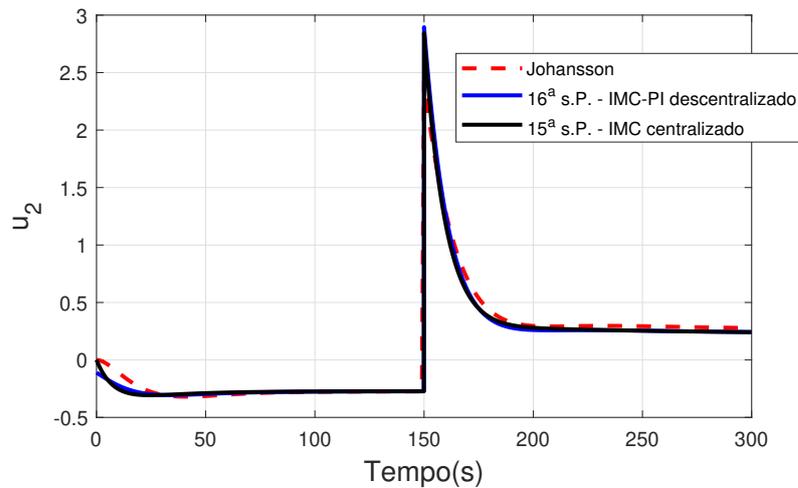


Figura 4.13: Esforço de controle u_2 (Comparação entre IMC-PI descentralizado, IMC centralizado e Johansson) - Modelo quatro tanques

porcentagens mássicas de metanol dos produtos de topo e de fundo. A interação entre as malhas faz com que a aplicação de uma entrada para a concentração de topo perturbe a concentração de fundo e vice-versa.

4.3.1 Resposta em malha aberta

Considerando o sistema em malha aberta e aplicando os sinais de entrada $u_1 = \mathbf{1}(t)$ e $u_2 = \mathbf{1}(t - 150)$, foram obtidos os transitórios apresentados na figura 4.14. Nota-se que a aplicação da entrada em qualquer uma das malhas gera uma grande perturbação na outra, o que evidencia o acoplamento existente nesse sistema.

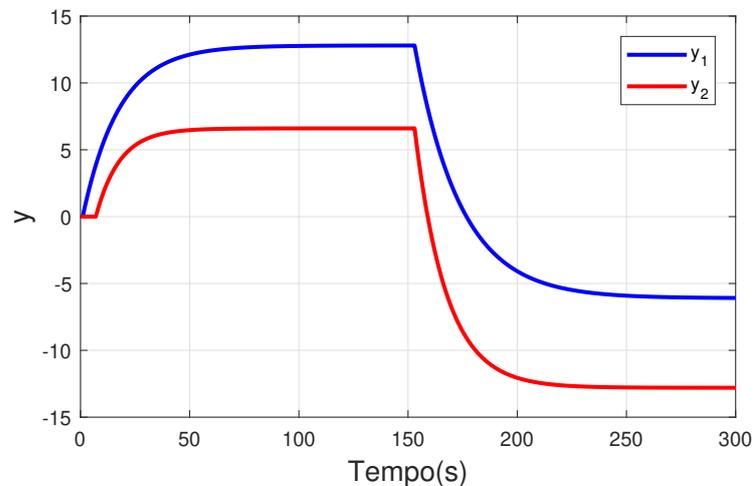


Figura 4.14: Resposta ao degrau do sistema em malha aberta - Modelo Wood-Berry

4.3.2 Desenvolvimento do desacoplador

O modelo para o sistema em malha-aberta desacoplado, H , foi escolhido como a diagonal principal do modelo original:

$$h_{jj} = g_{jj}, \quad j = 1, 2 \quad (4.11)$$

$$H = \begin{bmatrix} \frac{12,8}{16,7s+1}e^{-s} & 0 \\ 0 & \frac{-19,4}{14,4s+1}e^{-3s} \end{bmatrix}. \quad (4.12)$$

Para o modelo Wood-Berry, os melhores resultados foram obtidos implementando o desacoplador com um zero, como aquele descrito na equação (2.63). Dessa forma, o algoritmo DE foi utilizado para resolução do problema de otimização mono-objetivo descrito pelas equações (2.68) e (2.67). Nesse caso, a estrutura escolhida para o desacoplador têm 4 variáveis a serem otimizadas por malha de controle. Assim, sendo o modelo Wood-Berry de ordem 2, a população é definida com $N = 3\eta = 3 \times 4 \times 4 = 48$ indivíduos. O seguinte desacoplador foi obtido:

$$D = \begin{bmatrix} \frac{10,65s+2,007}{11,48s+1}e^{-0,279s} & \frac{15,82s+2,956}{15,55s+1}e^{-2,1s} \\ \frac{0,5988s+0,6836}{5,664s+1}e^{-2,32s} & \frac{7,851s+2,004}{10,16s+1}e^{-0,0405s} \end{bmatrix}. \quad (4.13)$$

Utilizando aproximação de Padé de primeira ordem para os elementos de atraso, obtém-se:

$$\tilde{D} = \begin{bmatrix} \frac{-0,92735(s-7,16)(s+0,1885)}{(s+7,16)(s+0,08709)} & \frac{-1,0176(s-0,9533)(s+0,1868)}{(s+0,9533)(s+0,06432)} \\ \frac{-0,10572(s+1,142)(s-0,8609)}{(s+0,8609)(s+0,1765)} & \frac{-0,77245(s-49,41)(s+0,2553)}{(s+49,41)(s+0,09839)} \end{bmatrix}. \quad (4.14)$$

A figura 4.15 apresenta a resposta ao degrau do modelo desejado, H , e do modelo obtido, $\hat{G} = \tilde{G}\tilde{D}$.

Observando a figura 4.15, nota-se que, como desejado, a implementação do desacoplador ótimo permitiu que as respostas ao degrau de \hat{G} e H fossem satisfatoriamente próximas. Assim, é possível aplicar a estratégia IMC tratando cada função de transferência de H como um sistema SISO.

Existem na literatura alguns métodos analíticos para síntese de desacopladores mais simples que o método via otimização utilizado nesse trabalho. Esses, no entanto, geralmente não permitem escolher as características do modelo desejado H . O desacoplador baseado na matriz adjunta de SHEN; SUN; LI (2012), por exemplo, apesar de possuir melhores resultados de desacoplamento, gera uma matriz H composta por elementos de segunda ordem ou superior. Por esse motivo, o desacoplador de SHEN; SUN; LI (2012) é inadequado para o nosso propósito. Afinal, para que seja possível obter o controla-

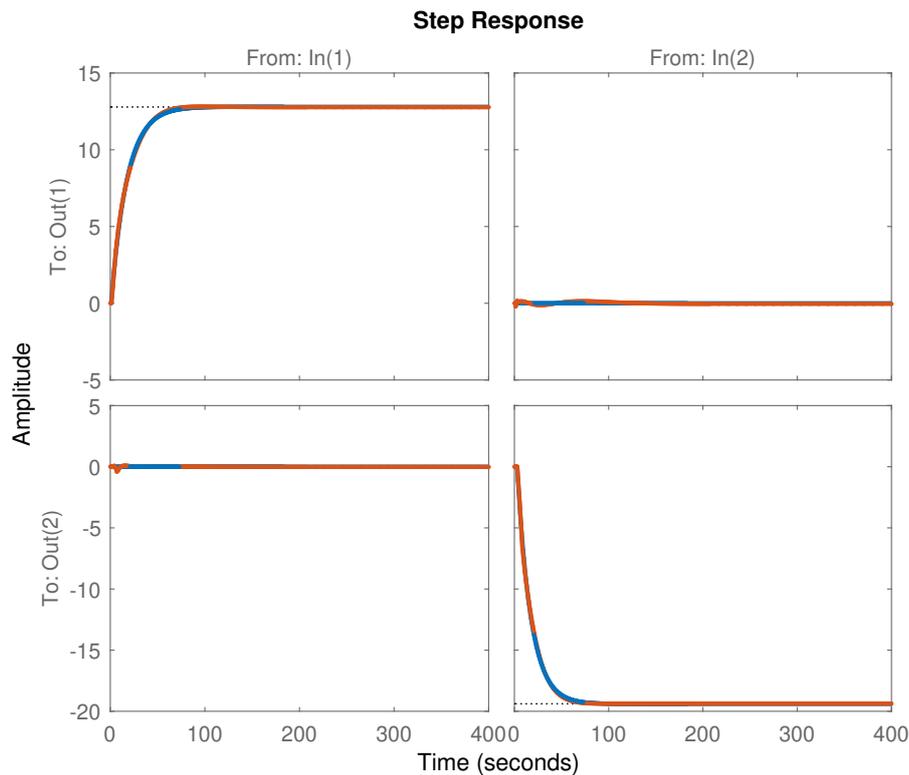


Figura 4.15: Resposta ao degrau de H , em azul, e \hat{G} , em vermelho - Modelo Wood-Berry

dor IMC-PI sem filtros adicionais, a matriz H deve ser composta apenas por funções de transferência de primeira ordem.

4.3.3 Síntese do controlador IMC-PI

Com base no modelo desejado H , aplicando o algoritmo DEMO para o problema de otimização multiobjetivo definido pela equação (2.74), é gerada a curva de Pareto composta pelo conjunto de soluções candidatas. A curva de Pareto encontrada para o modelo Wood-Berry é apresentada na figura 4.16.

Na figura 4.16, quanto mais à direita a solução estiver, menor será o esforço de controle e maior será o amortecimento que o respectivo o controlador irá proporcionar durante o transitório. Por outro lado, quanto mais a esquerda, menor o ISE de rastreamento e maior o esforço de controle. O projetista pode simular as diferentes soluções candidatas de Pareto e escolher aquela que possui o compromisso mais adequado entre a ISE e o esforço de controle.

Para comparação, foi considerado o controlador PID desacoplado baseado na matriz adjunta apresentado em SHEN; SUN; LI (2012). A tabela 4.3 mostra que, dentre as trinta soluções apresentadas na curva de Pareto da figura 4.16, seis (da 6^a solução, da esquerda para direita, à 11^a) possuem menor ISE e menor esforço de controle em relação

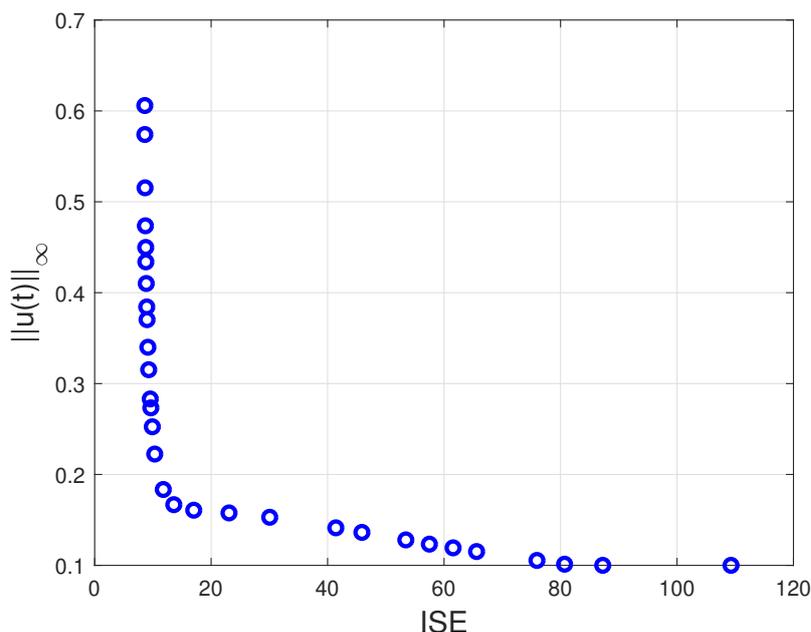


Figura 4.16: Conjunto de soluções candidatas de Pareto - Modelo Wood-Berry

ao resultado obtido em SHEN; SUN; LI (2012).

Tabela 4.3: Desempenho dos controladores - Modelo Wood-Berry

	6 ^a s.P.	7 ^a s.P.	8 ^a s.P.	9 ^a s.P.	10 ^a s.P.	11 ^a s.P.	Shen (2012)
$\ u(t)\ _\infty$	0,510	0,481	0,450	0,433	0,396	0,366	0,512
ISE1	3,026	3,106	3,206	3,269	3,428	3,585	2,717
ISE2	5,737	5,718	5,711	5,692	5,673	5,663	6,720
ISE	8,763	8,824	8,917	8,961	9,101	9,248	9,441

A tabela 4.4 mostra o vetor de constantes de tempo Λ de cada solução apresentada na tabela 4.3:

 Tabela 4.4: Constantes de tempo das funções em malha-fechada, T_f - Modelo Wood-Berry

	6 ^a s.P.	7 ^a s.P.	8 ^a s.P.	9 ^a s.P.	10 ^a s.P.	11 ^a s.P.
Λ	2,5513	2,7574	3,0097	3,1608	3,5328	3,8877
	2,1799	2,0372	2,2826	2,0372	2,0762	2,1651

Para que possamos verificar as variações em algumas características das respostas ao escolher controladores diferentes dentre os candidatos do conjunto de Pareto, são apresentados nas figuras 4.17, 4.18, 4.19 e 4.20 as respostas transitórias das saídas e dos sinais de controle do 6^o e do 11^o controladores, bem como do controlador de SHEN; SUN; LI (2012). Os controladores IMC-PI obtidos através da equação (2.70), sendo as constantes de tempo referentes a 6^a e a 11^a soluções, são, respectivamente:

$$C_{6^a} = \begin{bmatrix} 0,36739 \left(1 + \frac{1}{16,7s}\right) & 0 \\ 0 & -0,1433 \left(1 + \frac{1}{14,4s}\right) \end{bmatrix} \quad (4.15)$$

e

$$C_{11^a} = \begin{bmatrix} 0,26693 \left(1 + \frac{1}{16,7s}\right) & 0 \\ 0 & -0,14371 \left(1 + \frac{1}{14,4s}\right) \end{bmatrix}. \quad (4.16)$$

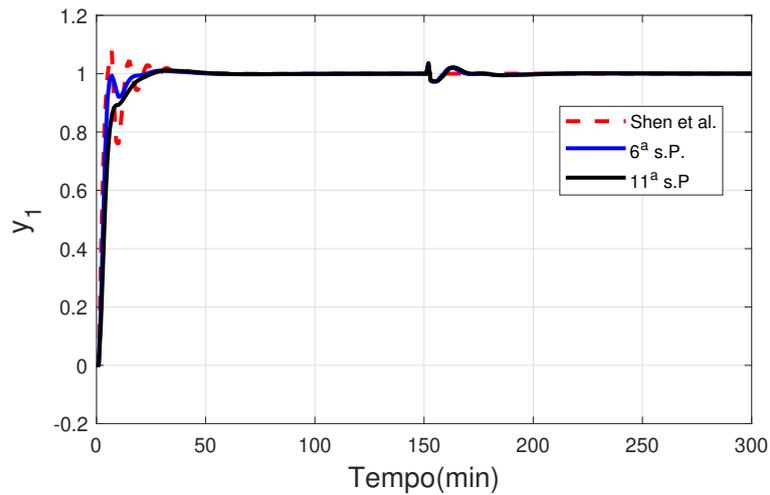


Figura 4.17: Resposta ao degrau y_1 - Modelo Wood-Berry

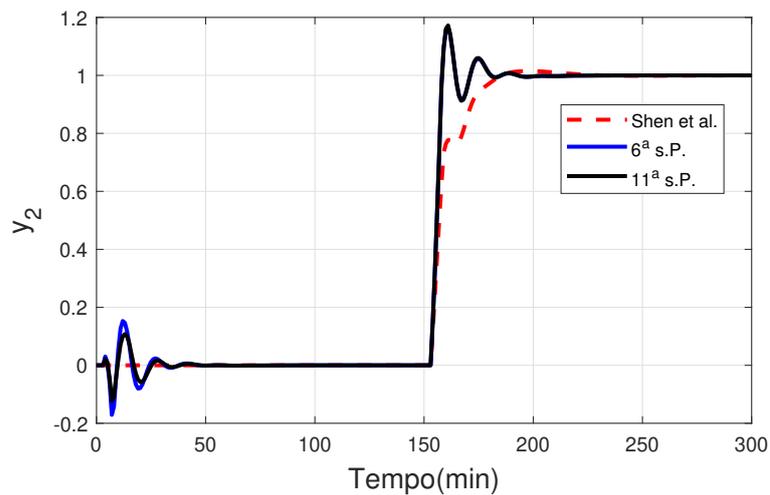
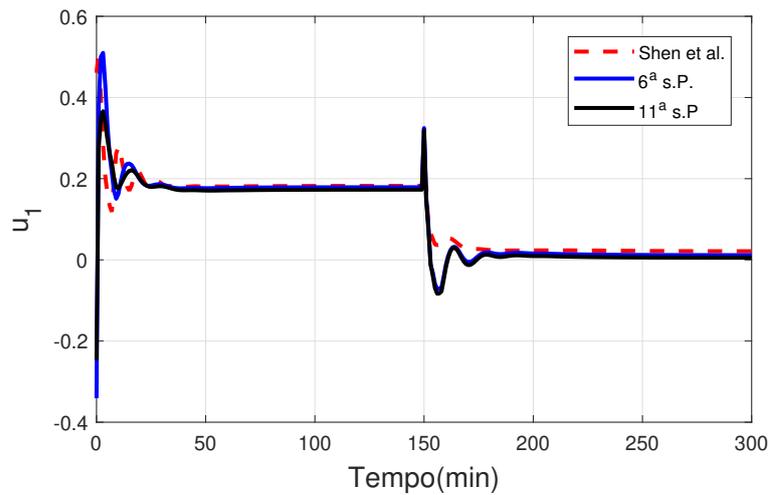
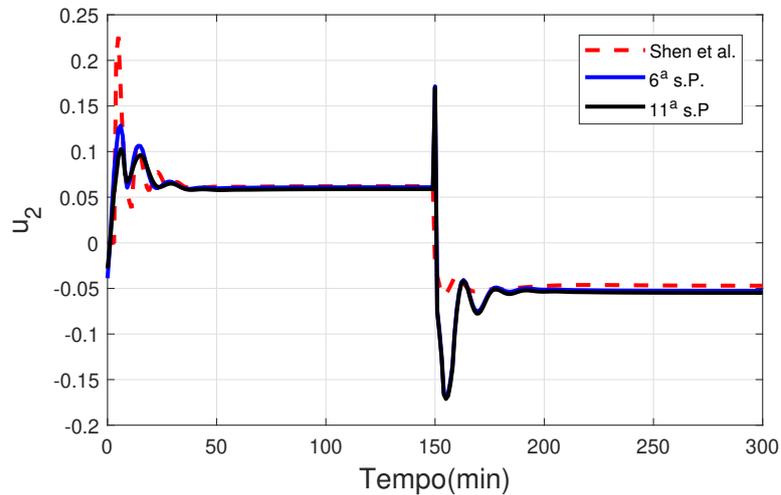


Figura 4.18: Resposta ao degrau y_2 - Modelo Wood-Berry

As figuras 4.17 e 4.18 mostram que, como esperado, com a implementação do controlador IMC-PI, o sinal de saída rastreia a referência com erro nulo em regime permanente. Analisando as figuras 4.19 e 4.20 juntamente com as figuras 4.17 e 4.18, as implicações de se escolher um controlador que exige maior ou menor esforço de controle se tornam mais evidentes. Embora, como já mencionado, ambos os controladores apresentados superem

Figura 4.19: Esforço de controle u_1 - Modelo Wood-BerryFigura 4.20: Esforço de controle u_2 - Modelo Wood-Berry

o controlador de SHEN; SUN; LI (2012) nos dois objetivos, as figuras comprovam que, ao implementar um controlador que exige um maior esforço de controle, o tempo de subida será menor, porém o sistema terá um transitório menos amortecido e, portanto, mais oscilatório. Nesse contexto, a utilização das técnicas de otimização multiobjetivo se mostram muito vantajosas, pois, ao buscar minimizar a ISE e o esforço de controle, que são objetivos concorrentes, de forma simultânea, elas permitem ao usuário escolher um controlador que beneficia suficientemente um determinado objetivo, penalizando o segundo objetivo o mínimo possível.

Cabe frisar também que, apesar do erro entre a resposta ao degrau do modelo desejado H , que representa um desacoplamento perfeito, e \hat{G} , que representa o modelo realmente obtido com o desacoplador, a interação restante entre as malhas se intensifica com o aumento do esforço de controle. Isso é perceptível ao se observar que a perturbação

sofrida pela segunda malha, decorrente da aplicação da referência na primeira, é maior no 6º controlador que no 11º. De forma análoga, ao se aplicar a referência na segunda malha em $t = 150 \text{ min}$, a perturbação sofrida pela primeira malha é maior utilizando o 6º controlador. Isso justifica o comportamento inesperado da segunda malha, onde, diferentemente da primeira malha, como pode ser visto na tabela 4.3 e na figura 4.18, a ISE2 aumenta com o aumento do esforço de controle. Em outras palavras, o aumento da perturbação, decorrente do aumento do esforço de controle, faz com que o erro de rastreamento nessa malha se intensifique quando o esperado era que ele diminuísse.

4.4 IMC-PI ótimo descentralizado aplicado ao modelo HVAC

Esse exemplo trata de um modelo de um sistema centralizado de aquecimento, ventilação e ar condicionado (HVAC - do inglês *heating, ventilation, and air conditioning*) de quatro quartos. As variáveis controladas do sistema são as temperaturas em cada um dos quatro quartos e as variáveis manipuladas são as posições de quatro *dampers* de ar que controlam o fluxo de ar condicionado no ambiente. O objetivo principal de controle é manter a temperatura condicionada de cada sala no valor desejado. O modelo 4x4 do sistema HVAC foi determinado por SHEN; CAI; LI (2010) como se segue:

$$\tilde{G} = \begin{bmatrix} \frac{-0,098}{122s+1}e^{-17s} & \frac{-0,036}{149s+1}e^{-27s} & \frac{-0,014}{158s+1}e^{-32s} & \frac{-0,017}{155s+1}e^{-30s} \\ \frac{-0,043}{147s+1}e^{-25s} & \frac{-0,092}{130s+1}e^{-16s} & \frac{-0,011}{156s+1}e^{-33s} & \frac{-0,012}{157s+1}e^{-34s} \\ \frac{-0,012}{153s+1}e^{-31s} & \frac{-0,016}{151s+1}e^{-34s} & \frac{-0,102}{118s+1}e^{-16s} & \frac{-0,033}{146s+1}e^{-26s} \\ \frac{-0,013}{156s+1}e^{-32s} & \frac{-0,015}{159s+1}e^{-31s} & \frac{-0,029}{144s+1}e^{-25s} & \frac{-0,108}{128s+1}e^{-18s} \end{bmatrix}. \quad (4.17)$$

4.4.1 Resposta em malha aberta

Para simulação do sistema em malha aberta, foram consideradas as seguintes entradas:

- $u_1 = \mathbf{1}(t)$
- $u_2 = \mathbf{1}(t - 2000)$
- $u_3 = \mathbf{1}(t - 4000)$
- $u_4 = \mathbf{1}(t - 6000)$

Pela figura 4.21, nota-se, assim como no sistema Wood-Berry, um grande acoplamento. Isso é evidenciado quando se observa que a variação do sinal de entrada em qualquer uma das malhas gera uma grande perturbação nas demais.

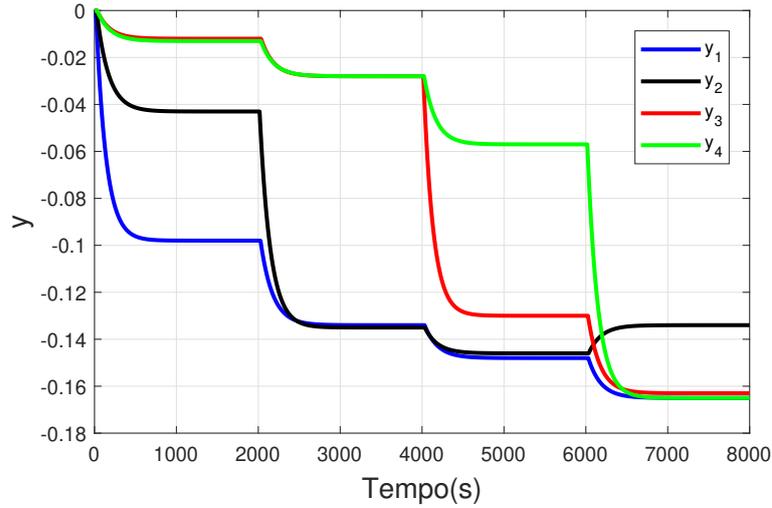


Figura 4.21: Resposta ao degrau do sistema em malha aberta - Modelo HVAC

4.4.2 Desenvolvimento do desacoplador

Novamente o modelo desejado, H , foi escolhido como a diagonal principal do modelo original:

$$h_{jj} = g_{jj}, \quad j = 1, \dots, 4 \quad (4.18)$$

$$H = \begin{bmatrix} \frac{-0,098}{122s+1}e^{-17s} & 0 & 0 & 0 \\ 0 & \frac{-0,092}{130s+1}e^{-16s} & 0 & 0 \\ 0 & 0 & \frac{-0,102}{118s+1}e^{-16s} & 0 \\ 0 & 0 & 0 & \frac{-0,108}{128s+1}e^{-18s} \end{bmatrix}. \quad (4.19)$$

Para o modelo HVAC, foi possível obter bons resultados implementando a estrutura de desacoplador sem zeros descrita na equação (2.64). Além da maior simplicidade de implementação, uma outra vantagem dessa estrutura é a economia de uma variável de otimização para cada elemento da matriz do desacoplador, o que representa, por se tratar de um sistema 4x4, 16 variáveis de otimização a menos em relação a estrutura com zeros. A população, nesse caso, tem $N = 3\eta = 3 \times 16 \times 3 = 144$ indivíduos. Assim, utilizando o DE para o problema de otimização mono-objetivo descrito pelas equações (2.68) e (2.67), o seguinte desacoplador foi obtido:

$$D = \begin{bmatrix} \frac{1,222}{6,834s+1}e^{-3,25s} & \frac{-0,4045}{31,19s+1}e^{-0,0569s} & \frac{-0,06124}{19,99s+1}e^{-14,4s} & \frac{-0,2445}{46,81s+1}e^{-4,48s} \\ \frac{-0,572}{23,49s+1}e^{-6,12s} & \frac{1,189}{5,523s+1}e^{-2,31s} & \frac{-0,1398}{25,92s+1}e^{-6,14s} & \frac{0,2982}{41,34s+1}e^{-13,1s} \\ \frac{-0,03696}{0,4018s+1}e^{-5,04s} & \frac{-0,1095}{18,67s+1}e^{-15,9s} & \frac{1,117}{2s+1}e^{-1,95s} & \frac{-0,367}{27,58s+1}e^{-3,74s} \\ \frac{-0,05819}{23,68s+1}e^{-2,48s} & \frac{-0,08733}{40,02s+1}e^{-5,77s} & \frac{-0,271}{3,186s+1}e^{-13,3s} & \frac{1,084}{1,336s+1}e^{-0,917s} \end{bmatrix}. \quad (4.20)$$

Aproximando os elementos de atraso por Padé de primeira ordem, temos:

$$\tilde{D} = \begin{bmatrix} \frac{-0,17885(s-0,6145)}{(s+0,6145)(s+0,1463)} & \frac{0,012967(s-35,16)}{(s+35,16)(s+0,03206)} & \frac{0,0030632(s-0,1389)}{(s+0,1389)(s+0,05002)} & \frac{0,0052229(s-0,446)}{(s+0,446)(s+0,02136)} \\ \frac{0,024346(s-0,327)}{(s+0,327)(s+0,04257)} & \frac{-0,21524(s-0,8648)}{(s+0,8648)(s+0,181)} & \frac{0,0053946(s-0,3256)}{(s+0,3256)(s+0,03858)} & \frac{-0,0072139(s-0,1526)}{(s+0,1526)(s+0,02419)} \\ \frac{0,091978(s-0,3965)}{(s+2,489)(s+0,3965)} & \frac{0,0058664(s-0,1259)}{(s+0,1259)(s+0,05357)} & \frac{-0,55826(s-1,024)}{(s+1,024)(s+0,5)} & \frac{0,013306(s-0,5349)}{(s+0,5349)(s+0,03626)} \\ \frac{0,0024574(s-0,8061)}{(s+0,8061)(s+0,04223)} & \frac{0,0021822(s-0,3463)}{(s+0,3463)(s+0,02499)} & \frac{0,08505(s-0,1499)}{(s+0,3138)(s+0,1499)} & \frac{-0,81129(s-2,182)}{(s+2,182)(s+0,7483)} \end{bmatrix}. \quad (4.21)$$

A figura 4.22 apresenta a resposta ao degrau do modelo desejado H , e do modelo obtido, $\hat{G} = \tilde{G}\tilde{D}$:

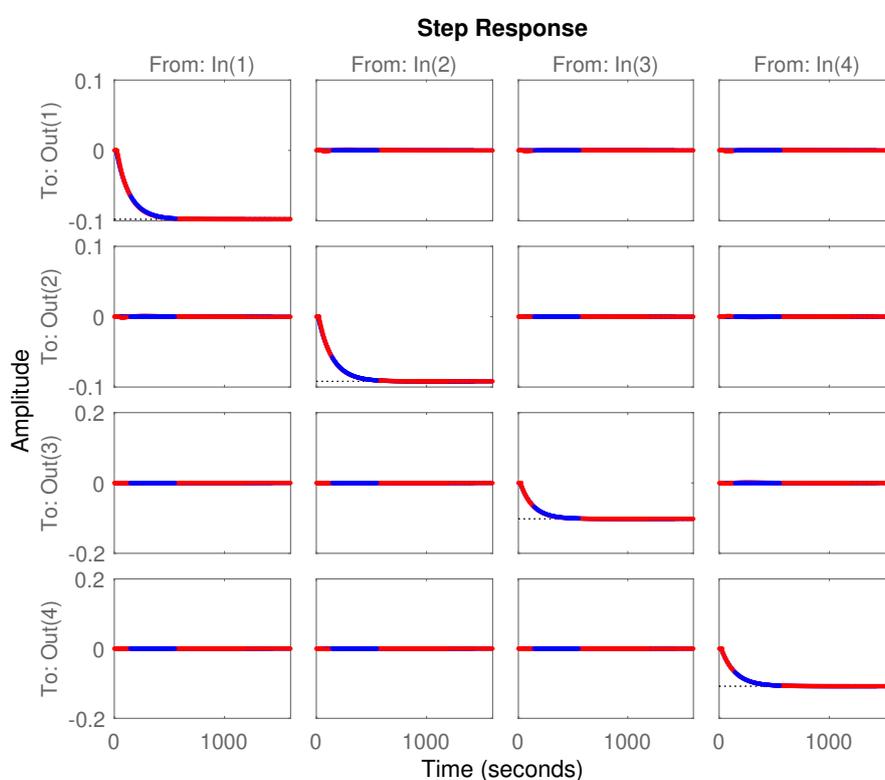


Figura 4.22: Resposta ao degrau de H , em azul, e \hat{G} , em vermelho - Modelo HVAC

Analisando a figura 4.22, fica claro que, mesmo utilizando uma estrutura de desacoplador com menos variáveis de otimização e, ainda, aproximando os elementos de atraso por Padé, o erro entre as respostas ao degrau de \hat{G} e H é quase nulo. Dessa forma, é possível projetar os controladores para cada função de transferência de H de forma independente.

4.4.3 Síntese do controlador IMC-PI

Para a obtenção dos parâmetros do controlador IMC-PI, foram considerados os seguinte sinais de referência:

- $r_1 = \mathbf{1}(t)$
- $r_2 = \mathbf{1}(t - 1000)$
- $r_3 = \mathbf{1}(t - 2000)$
- $r_4 = \mathbf{1}(t - 3000)$

Aplicando o algoritmo DEMO com base no modelo desejado H para solucionar o problema de otimização multiobjetivo dado pela equação (2.74), é gerada a curva de Pareto apresentada na figura 4.23. Como nos exemplos anteriores, cada ponto representa um controlador IMC-PI que estabelece um compromisso entre a ISE do erro de rastreamento e o esforço de controle, calculado a partir de um vetor $\Lambda = [\lambda_1 \ \lambda_2 \ \lambda_3 \ \lambda_4]^T$.

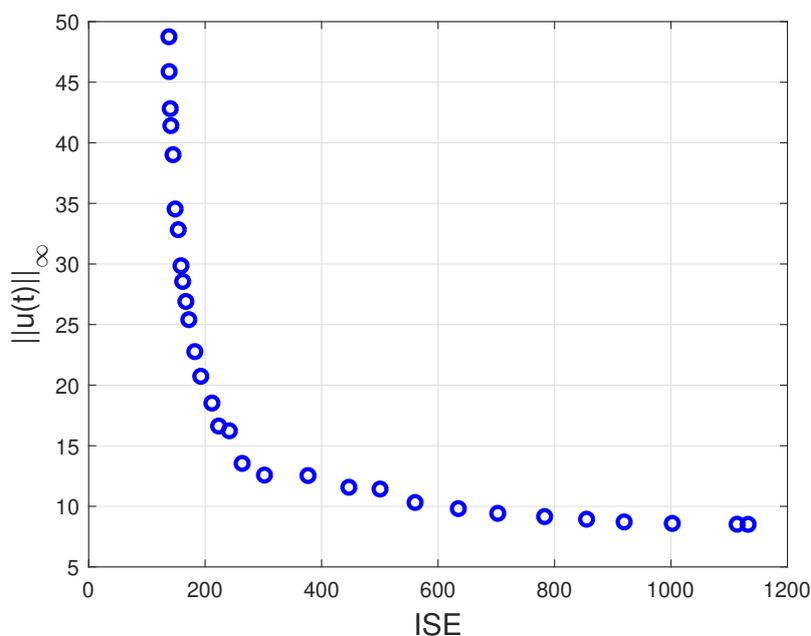


Figura 4.23: Conjunto de soluções candidatas de Pareto - Modelo HVAC

Os resultados obtidos nesse trabalho foram comparados com o controlador desacoplado normalizado apresentado em SHEN; CAI; LI (2010), sintonizado com base na margem de ganho e margem de fase. A tabela 4.5 apresenta os valores da ISE e da $\|u(t)\|_\infty$ da 1ª, 12ª e 13ª soluções da figura 4.23.

Das soluções apresentadas na tabela 4.5, a 1ª possui um valor de ISE muito inferior ao obtido com o controlador de SHEN; CAI; LI (2010). O esforço de controle, contudo, é significativamente maior. Já a 12ª e a 13ª soluções dominam o controlador de SHEN; CAI; LI (2010), ou seja, os superam nos dois objetivos. A tabela 4.6 mostra o vetor de constantes de tempo Λ de cada solução apresentada na tabela 4.5:

Tabela 4.5: Desempenho dos controladores - Modelo HVAC

	1 ^a	12 ^a	13 ^a	Shen (2010)
$\ u(t)\ _\infty$	48,5881	22,7404	20,7198	24,274
ISE1	39,6103	53,0174	56,3122	48,9225
ISE2	35,5193	46,0161	48,3352	48,5594
ISE3	30,9784	42,7471	45,2578	49,7391
ISE4	31,5670	40,5288	42,6284	51,7771
ISE	137,6750	182,3095	192,5336	198,9980

 Tabela 4.6: Constantes de tempo das funções em malha-fechada, T_f - Modelo HVAC

	1 ^a s. P.	12 ^a s.P.	13 ^a s.P.
Λ	$\begin{bmatrix} 20,7051 \\ 18,3503 \\ 13,5539 \\ 10,8600 \end{bmatrix}$	$\begin{bmatrix} 61,4037 \\ 50,3696 \\ 47,0484 \\ 38,3332 \end{bmatrix}$	$\begin{bmatrix} 68,7978 \\ 55,6525 \\ 52,5889 \\ 43,0668 \end{bmatrix}$

Dentre as soluções candidatas obtidas para o sistema HVAC, a 12^a e a 13^a solução, que dominam o controlador com o qual são comparadas, por serem próximas, possuem respostas com características semelhantes. Por esse motivo, apenas uma será discorrida adiante. Portanto, com intuito de permitir a observação das diferenças entre as respostas de soluções distantes no conjunto de Pareto, serão exibidas a seguir as repostas dos sinais de saída e de controle da 1^a e da 12^a solução, bem como do controlador de SHEN; CAI; LI (2010). Utilizando as constantes de tempo exibidas nas colunas referentes à 1^a e à 12^a solução na tabela 4.6, os controladores IMC-PI obtidos são dados por:

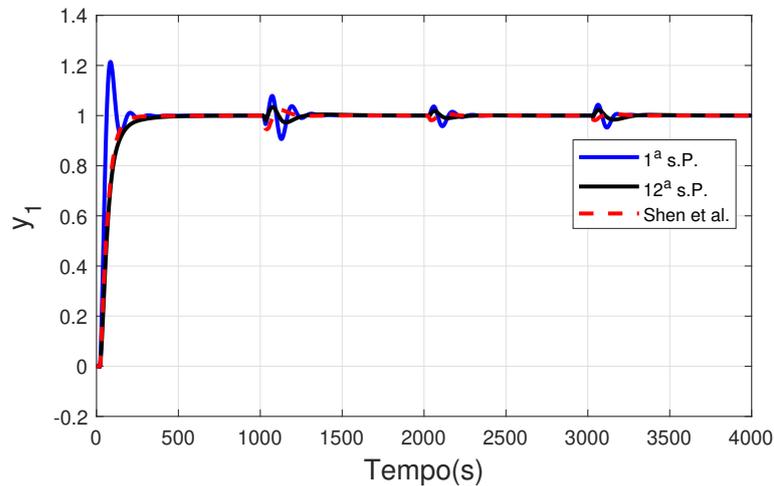
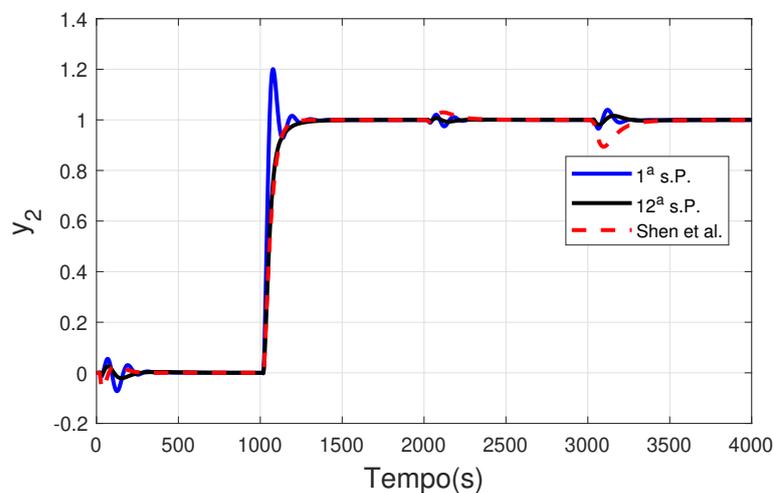
$$C_{1^a} = \begin{bmatrix} -33,017 \left(1 + \frac{1}{122s}\right) & 0 & 0 & 0 \\ 0 & -41,136 \left(1 + \frac{1}{130s}\right) & 0 & 0 \\ 0 & 0 & -39,144 \left(1 + \frac{1}{118s}\right) & 0 \\ 0 & 0 & 0 & -41,067 \left(1 + \frac{1}{128s}\right) \end{bmatrix}. \quad (4.22)$$

e

$$C_{12^a} = \begin{bmatrix} -15,878 \left(1 + \frac{1}{122s}\right) & 0 & 0 & 0 \\ 0 & -21,291 \left(1 + \frac{1}{130s}\right) & 0 & 0 \\ 0 & 0 & -18,349 \left(1 + \frac{1}{118s}\right) & 0 \\ 0 & 0 & 0 & -21,039 \left(1 + \frac{1}{128s}\right) \end{bmatrix}. \quad (4.23)$$

As figuras 4.24 a 4.31 mostram as respostas transitórias dos sinais de saída e dos sinais de controle. Novamente, as melhorias esperadas para a resposta do sistema com

o controlador IMC-PI são facilmente notadas nas figuras que representam as saídas do sistema. Além de diminuir significativamente o tempo de acomodação do sinal de saída, ambos controladores escolhidos garantem, em virtude da presença do integrador, erro nulo em regime permanente.

Figura 4.24: Resposta ao degrau y_1 - Modelo HVACFigura 4.25: Resposta ao degrau y_2 - Modelo HVAC

Pelas figuras 4.24 a 4.31, também é notório que o aumento do esforço de controle aumenta a perturbação decorrente da variação dos sinais de referências nas malhas. Nesse caso, porém, isso não foi suficiente para fazer com que a ISE aumentasse em alguma malha. Portanto, como esperado, em todas as malhas, o erro de rastreamento diminuiu com o aumento do esforço de controle.

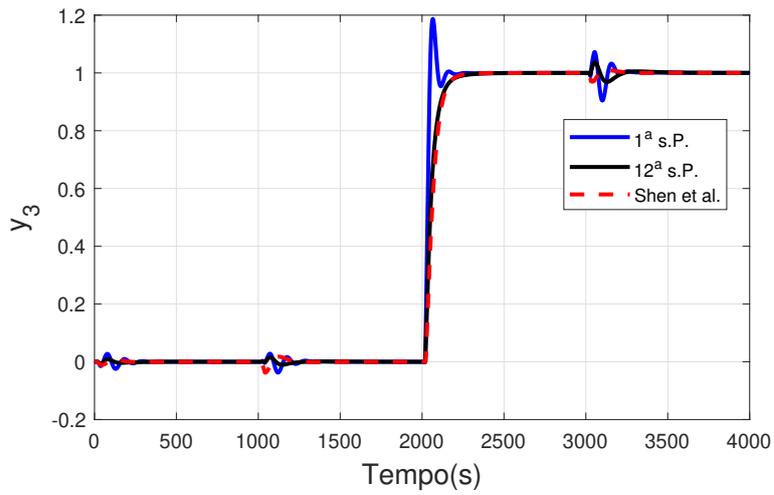


Figura 4.26: Resposta ao degrau y_3 - Modelo HVAC

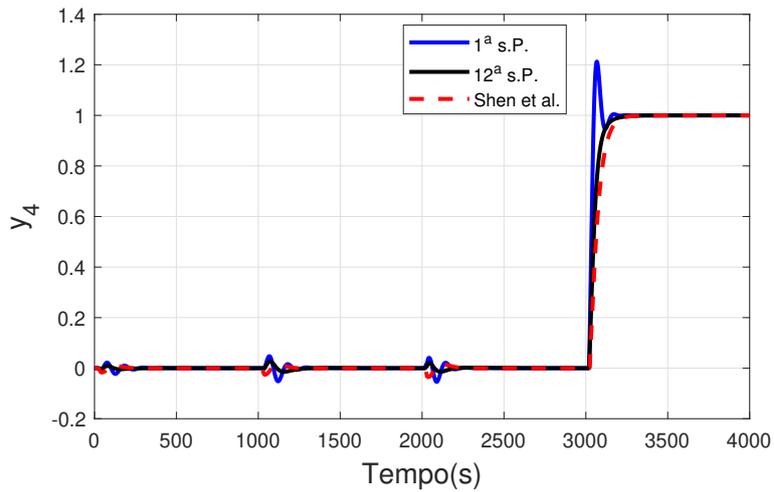


Figura 4.27: Resposta ao degrau y_4 - Modelo HVAC

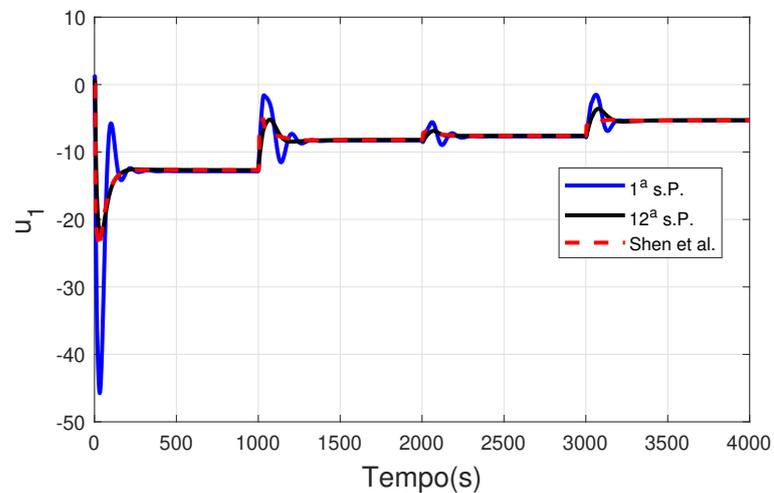


Figura 4.28: Esforço de controle u_1 - Modelo HVAC

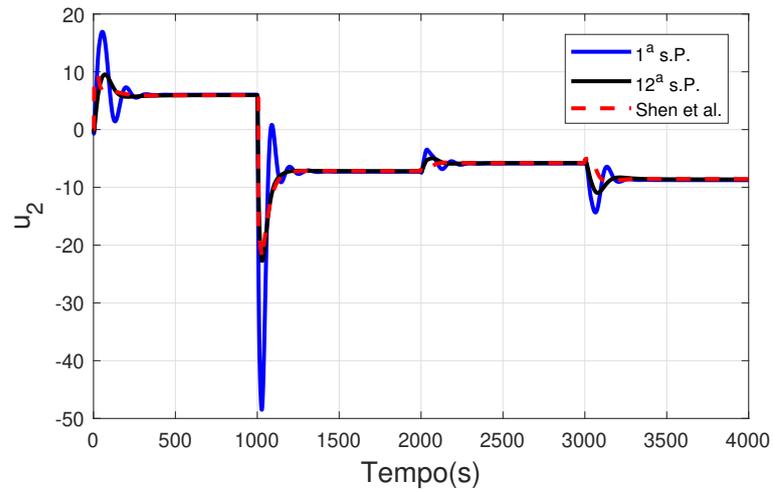


Figura 4.29: Esforço de controle u_2 - Modelo HVAC

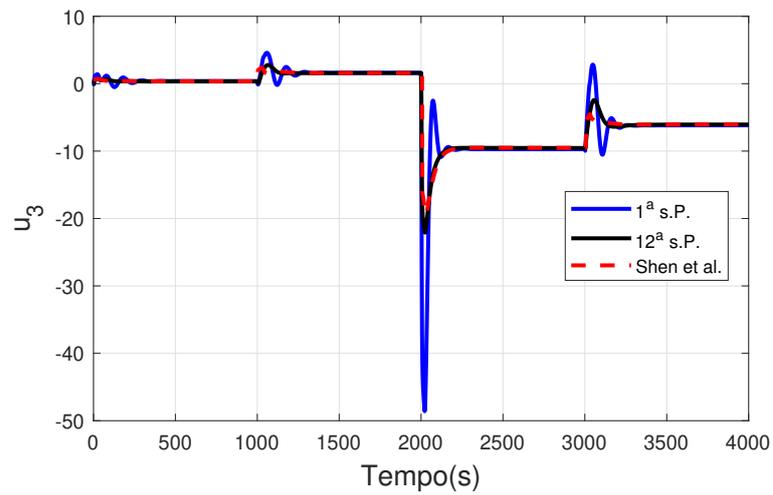


Figura 4.30: Esforço de controle u_3 - Modelo HVAC

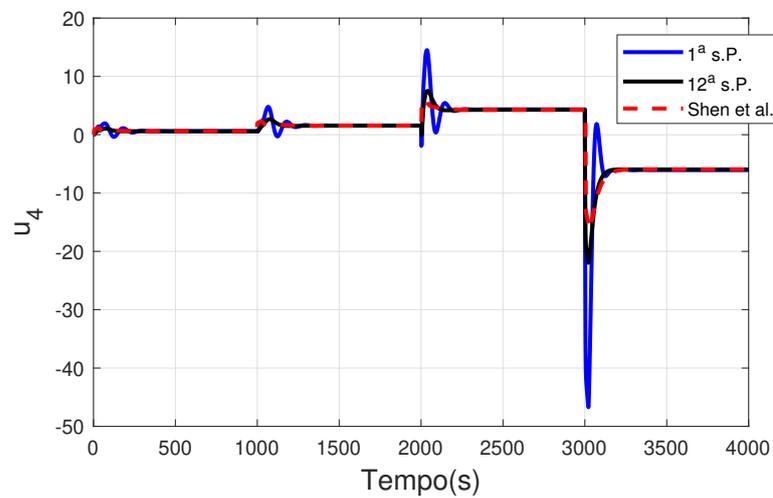


Figura 4.31: Esforço de controle u_4 - Modelo HVAC

Considerações Finais

5.1 Conclusões

Nesse trabalho foi analisada uma metodologia de projeto de controladores IMC-PI para sistemas MIMO. Tal metodologia consiste primeiramente no desenvolvimento de um desacoplador ótimo via algoritmo DE e, em seguida, a sintonia dos parâmetros dos controladores IMC-PI através do algoritmo DEMO. A partir disso, dentro de um conjunto de soluções conhecidas como soluções eficientes ou candidatas a curva de Pareto ótima, o usuário pode escolher os valores dos parâmetros que irão conferir ao controlador IMC-PI a capacidade de atender aos requisitos de desempenho peculiares de cada sistema. Assim, o controle é realizado sobre o modelo desacoplado, que é tratado como um conjunto de sistemas SISO. Apesar da simplicidade da forma em que o controlador é projetado, os resultados apresentados nessa dissertação se mostraram melhores ou parelhos aos resultados com os quais foram comparados.

Embora a necessidade de dois processos de otimização demande um esforço computacional considerável, a metodologia proposta atingiu o seu objetivo principal, que é a obtenção de um controlador simples, eficaz e flexível à diversos tipos de planta. O capítulo 4 mostrou que a utilização de desacopladores desenvolvidos através do algoritmo de otimização DE permite, de forma satisfatória, tratar a função de transferência como uma matriz diagonal composta apenas pelas funções de transferência desejadas pelo usuário. Nesse cenário, a simplicidade do projeto pode ser destacada nas seguintes partes:

- **Desacoplador** - Os desacopladores utilizados nesse trabalho são desenvolvidos como funções de primeira ordem. Cabe destacar que, após a aproximação por Padé para os elementos de atraso presentes no desacoplador, com intuito de os tornarem facilmente aplicáveis, os mesmos passam a ser compostos por funções de segunda ordem sem perda relevante de eficiência.
- **Matriz de transferência desejada** - Viabilizada pela utilização das técnicas de

otimização, a flexibilidade de escolha da matriz de transferência desejada permitiu fixar seus elementos como funções de primeira ordem, o que torna possível a sintonia dos controladores PI pelo método IMC.

- **Controlador** - Uma vez que é possível tratar o modelo como uma matriz diagonal cujos elementos são compostos por funções de primeira ordem, torna-se viável a obtenção de controladores IMC-PIs sem filtros adicionais. Quando comparado com outras metodologias que também utilizam técnicas de otimização para o desenvolvimento de controladores PID ou PI, outro aspecto a ser destacado é a diminuição do esforço computacional despendido na sintonia do controlador, dado que nos controladores IMC-PI é necessário estimar apenas um parâmetro por malha de controle, a constante de tempo em malha fechada, diferentemente dos controladores PID propriamente ditos, que exigem o ajuste de três parâmetros por malha de controle.

Além de garantir a eficiência do controlador, a utilização do algoritmo DEMO permitiu que outro objetivo fosse atingido: dar ao usuário a possibilidade de escolher dentre várias opções disponíveis, os valores dos parâmetros mais apropriados para os requisitos de desempenho em questão. Com a metodologia percorrida aqui, os valores dos parâmetros do controlador podem ser escolhidos através da curva de Pareto, onde cada ponto representa a relação entre as duas funções objetivo especificadas. Nos exemplos ilustrativos apresentados, o usuário pode eleger o melhor controlador dentro de possibilidades que relacionam o ISE do erro e a norma infinita do sinal de controle. Em outras palavras, ele pode definir aquele controlador que terá o menor erro entre a referência e a saída e, ao mesmo tempo, não satura o atuador.

5.2 Propostas de continuidade

Nesse trabalho, para a sintonia do controlador, optou-se por adotar como funções objetivos a serem minimizadas, a somatória das integrais dos erros ao quadrado dos sinais de saída, que corresponde a norma 2 de sinais, e o valor máximo do módulo dos sinais de controle, que corresponde a norma infinito de sinais. Contudo, a depender das características desejadas para o sistema, outras funções objetivo podem ser escolhidas. Pode-se, por exemplo, realizar testes em que as funções objetivos são designadas visando diminuir o *overshoot*, a oscilação da resposta durante o transitório ou a energia do sinal de controle.

Apesar dos bons resultados alcançados com o DE mono e multiobjetivo, um estudo comparativo envolvendo outras técnicas de otimização no desenvolvimento dos desacopladores e dos controladores seria oportuno. Dessa forma, seria possível garantir de forma segura que, dentre as técnicas de otimização vigentes, no que tange aos procedimentos

propostos, o resultado obtido seria o mais eficiente em termos de custo computacional e minimização das funções objetivos.

Com intuito de se obter, a partir da metodologia IMC, os controladores mais simples possíveis, escolhemos fixar os elementos das matrizes desejadas em apenas funções de primeira ordem. Dessa forma, são obtidos controladores PIs na sua forma tradicional. Contudo, os controladores PID ou os filtros, naturalmente gerados quando as funções de transferência ou o método de aproximação do atraso são diferentes dos utilizados nesse trabalho, podem prover resultados melhores em algumas situações. Portanto, um estudo em que se avalia possíveis melhorias dos resultados quando se utiliza controladores IMC-PI ou IMC-PID com filtros, em detrimento de uma pequena perda de simplicidade, é extremamente pertinente.

Como mencionado no parágrafo anterior, as matrizes de transferência desejadas foram fixadas de forma a beneficiar a obtenção de controladores simples. Há, porém, em virtude da utilização de técnicas de otimização no desenvolvimento dos desacopladores, uma flexibilidade na escolha da matriz de transferência. Em outras palavras, avaliar possíveis benefícios alcançados ao se fixar formas diferentes para a matriz desacoplada resultante, justificaria a proposição de novos trabalhos. Porém, deve-se atentar que, nesse caso, considerando que no método IMC o controlador gerado depende do modelo, variações da matriz de transferência desejada podem resultar em controladores mais complexos.

Referências

- ALCÁNTARA, S. *et al.* IMC-like analytical \mathcal{H}_∞ design with S/SP mixed sensitivity consideration: utility in pid tuning guidance. *Journal of Process Control*, [S.l.], v.21, n.6, p.976–985, 2011.
- ALI, A.; MAJHI, S. PI/PID controller design based on IMC and percentage overshoot specification to controller setpoint change. *ISA transactions*, [S.l.], v.48, n.1, p.10–15, 2009.
- ALVAREZ-RAMIREZ, J. *et al.* Optimality of internal model control tuning rules. *Industrial & engineering chemistry research*, [S.l.], v.43, n.24, p.7951–7958, 2004.
- ÅSTRÖM, K. J.; HÄGGLUND, T. *PID controllers: theory, design, and tuning*. [S.l.]: Instrument society of America Research Triangle Park, NC, 1995. v.2.
- BOYD, S. *et al.* *Linear matrix inequalities in system and control theory*. [S.l.]: Siam, 1994. v.15.
- BRISTOL, E. On a new measure of interaction for multivariable process control. *IEEE transactions on automatic control*, [S.l.], v.11, n.1, p.133–134, 1966.
- BROSILOW, C. B. The structure and design of Smith predictors from the viewpoint of inferential control. In: JOINT AUTOMATIC CONTROL CONFERENCE. *Anais...* [S.l.: s.n.], 1979. n.16, p.288.
- CASTRO, R. E. de. *Otimização de estruturas com multi-objetivos via algoritmos genéticos*. 2001. Tese (Doutorado em Ciência da Computação) — Tese de Doutorado, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.
- CHEN, T. *et al.* A large population size can be unhelpful in evolutionary algorithms. *Theoretical Computer Science*, [S.l.], v.436, p.54–70, 2012.
- CHIEN, I.-L. Consider IMC tuning to improve controller performance. *Chem. Eng. Prog.*, [S.l.], v.86, p.33–41, 1990.

- COELLO, C. A. C. A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information systems*, [S.l.], v.1, n.3, p.269–308, 1999.
- DAS, S.; MULLICK, S. S.; SUGANTHAN, P. N. Recent advances in differential evolution – An updated survey. *Swarm and Evolutionary Computation*, [S.l.], v.1, p.1–30, April 2016.
- DEB, K. *et al.* A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: nsga-ii. In: INTERNATIONAL CONFERENCE ON PARALLEL PROBLEM SOLVING FROM NATURE. *Anais...* [S.l.: s.n.], 2000. p.849–858.
- DEB, K. *et al.* A fast and elitist multiobjective genetic algorithm: nsga-ii. *IEEE transactions on evolutionary computation*, [S.l.], v.6, n.2, p.182–197, 2002.
- DEHGHANI, A.; LANZON, A.; ANDERSON, B. D. \mathcal{H}_∞ design to generalize internal model control. *Automatica*, [S.l.], v.42, n.11, p.1959–1968, 2006.
- DIAZ, C. *et al.* IMC based PID Control Applied to the Benchmark PID 18. In: IFAC CONFERENCE IN ADVANCES IN PROPORTIONAL-INTEGRAL-DERIVATIVE CONTROL, PID 2018, 3. *Anais...* [S.l.: s.n.], 2018. p.728–732.
- GAGNON, E.; POMERLEAU, A.; DESBIENS, A. Simplified, ideal or inverted decoupling? *ISA transactions*, [S.l.], v.37, n.4, p.265–276, 1998.
- GARCIA, C. E.; MORARI, M. Internal model control. A unifying review and some new results. *Industrial & Engineering Chemistry Process Design and Development*, [S.l.], v.21, n.2, p.308–323, 1982.
- GARRIDO, J.; VÁZQUEZ, F.; MORILLA, F. Inverted decoupling internal model control for square stable multivariable time delay systems. *Journal of Process Control*, [S.l.], v.24, n.11, p.1710–1719, 2014.
- GONÇALVES, B. M. *et al.* Robust decoupling PI controllers for multi-loop control. In: DECISION AND CONTROL (CDC), 2012 IEEE 51ST ANNUAL CONFERENCE ON. *Anais...* [S.l.: s.n.], 2012. p.1530–1535.
- HE, M.-J. *et al.* RNGA based control system configuration for multivariable processes. *Journal of Process control*, [S.l.], v.19, n.6, p.1036–1042, 2009.
- HORN, I. G. *et al.* Improved filter design in internal model control. *Industrial & engineering chemistry research*, [S.l.], v.35, n.10, p.3437–3441, 1996.

- HORN, J. rey; NAFPLIOTIS, N.; GOLDBERG, D. E. A niched Pareto genetic algorithm for multiobjective optimization. In: IEEE CONFERENCE ON EVOLUTIONARY COMPUTATION, IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE. *Proceedings...* [S.l.: s.n.], 1994. v.1, p.82–87.
- IRUTHAYARAJAN, M. W.; BASKAR, S. Evolutionary algorithms based design of multi-variable PID controller. *Expert Systems with applications*, [S.l.], v.36, n.5, p.9159–9167, 2009.
- ISSA, W. R. A New Approach of Robust Internal Model Control for Nonlinear Pendulum System. *This thesis is submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical Engineering Islamic University of Gaza, Palestine*, [S.l.], 2011.
- JEROME, N.; RAY, W. H. High-Performance multivariable control strategies for systems having time delays. *AIChE Journal*, [S.l.], v.32, n.6, p.914–931, 1986.
- JIN, Q.; DU, X.; JIANG, B. Novel Centralized IMC-PID Controller Design for Multivariable Processes with Multiple Time Delays. *Industrial & Engineering Chemistry Research*, [S.l.], v.56, n.15, p.4431–4445, 2017.
- JIN, Q. *et al.* Decoupling internal model control for non-square processes based on equivalent transfer function. *Transactions of the Institute of Measurement and Control*, [S.l.], v.36, n.8, p.1114–1131, 2014.
- JIN, Q. *et al.* Analytical design 2 DOF IMC control based on inverted decoupling for non-square systems with time delay. *The Canadian Journal of Chemical Engineering*, [S.l.], v.94, n.7, p.1354–1367, 2016.
- JIN, Q.; HAO, F.; WANG, Q. A multivariable IMC-PID method for non-square large time delay systems using NPSO algorithm. *Journal of Process Control*, [S.l.], v.23, n.5, p.649–663, 2013.
- JOHANSSON, K. H. The quadruple-tank process: a multivariable laboratory process with an adjustable zero. *IEEE Transactions on control systems technology*, [S.l.], v.8, n.3, p.456–465, 2000.
- JUNIOR, G. A.; BARROS, P. R. Estudo comparativo de técnicas de projeto de desacoplador para sistemas multivariáveis. *Revista Principia*, [S.l.], v.1, n.28, p.130–143, 2015.

- KAZIMIPOUR, B.; LI, X.; QIN, A. K. Initialization methods for large scale global optimization. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION, 2013. *Anais...* [S.l.: s.n.], 2013. p.2750–2757.
- KNOWLES, J.; CORNE, D. The pareto archived evolution strategy: a new baseline algorithm for pareto multiobjective optimisation. In: EVOLUTIONARY COMPUTATION, 1999. CEC 99. PROCEEDINGS OF THE 1999 CONGRESS ON. *Anais...* [S.l.: s.n.], 1999. v.1, p.98–105.
- LI, S.; GU, H. Fuzzy adaptive internal model control schemes for PMSM speed-regulation system. *IEEE Transactions on Industrial Informatics*, [S.l.], v.8, n.4, p.767–779, 2012.
- MAMAT, R. *et al.* A new tuning method for two-degree-of-freedom internal model control under parametric uncertainty. *Chinese Journal of Chemical Engineering*, [S.l.], v.21, n.9, p.1030–1037, 2013.
- NAJAFIZADEGAN, H.; MERRIKH-BAYAT, F.; JALILVAND, A. IMC-PID controller design based on loop shaping via LMI approach. *Chemical Engineering Research and Design*, [S.l.], v.124, p.170–180, 2017.
- NASUTION, A. A.; JENG, J.-C.; HUANG, H.-P. Optimal H₂ IMC-PID Controller with Set-Point Weighting for Time-Delayed Unstable Processes. *Industrial & Engineering Chemistry Research*, [S.l.], v.50, n.8, p.4567–4578, 2011.
- NATH, U. M.; DEY, C.; MUDI, R. K. Fuzzy-tuned SIMC controller for level control loop. In: *Industry Interactive Innovations in Science, Engineering and Technology*. [S.l.]: Springer, 2018. p.239–245.
- GRIMBLE, M. J.; JOHNSON, M. A. (Ed.). *Control of dead-time processes. - (Advanced textbooks in control and signal processing)*. London: Springer-Verlag, 2007.
- OSY CZKA, A. Multicriteria optimization for engineering design. In: *Design optimization*. [S.l.]: Elsevier, 1985. p.193–227.
- PANDA, S. Differential evolutionary algorithm for TCSC-based controller design. *Simulation Modelling Practice and Theory*, [S.l.], v.17, n.10, p.1618–1634, 2009.
- RAJAPANDIYAN, C.; CHIDAMBARAM, M. Controller Design for MIMO Processes Based on Simple Decoupled Equivalent Transfer Functions and Simplified Decoupler. *Industrial & Engineering Chemistry Research*, [S.l.], v.51, n.38, p.12398–12410, 2012.

- RIVALS, I.; PERSONNAZ, L. Nonlinear internal model control using neural networks: application to processes with delay and design issues. *IEEE Transactions on Neural Networks*, [S.l.], v.11, n.1, p.80–90, 2000.
- RIVERA, D. E. Internal model control: a comprehensive view. *Arizona State University, Tempe, Arizona*, [S.l.], p.85287–6006, 1999.
- RIVERA, D. E.; MORARI, M.; SKOGESTAD, S. Internal model control: pid controller design. *Industrial & engineering chemistry process design and development*, [S.l.], v.25, n.1, p.252–265, 1986.
- ROBIČ, T.; FILIPIČ, B. Differential evolution for multiobjective optimization. In: INTERNATIONAL CONFERENCE ON EVOLUTIONARY MULTI-CRITERION OPTIMIZATION. *Anais. . .* [S.l.: s.n.], 2005. p.520–533.
- ROTSTEIN, G. E.; LEWIN, D. R. Simple PI and PID tuning for open-loop unstable systems. *Industrial & engineering chemistry research*, [S.l.], v.30, n.8, p.1864–1869, 1991.
- SAXENA, S.; HOTE, Y. V. Advances in internal model control technique: a review and future prospects. *IETE Technical Review*, [S.l.], v.29, n.6, p.461–472, 2012.
- SAXENA, S.; HOTE, Y. V. Simple approach to design PID controller via internal model control. *Arabian journal for science and engineering*, [S.l.], v.41, n.9, p.3473–3489, 2016.
- SCHAFFER, J. D. Multiple objective optimization with vector evaluated genetic algorithms. In: FIRST INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS AND THEIR APPLICATIONS, 1985. *Proceedings. . .* [S.l.: s.n.], 1985.
- SHAMSUZZOHA, M.; LEE, M. IMC- PID controller design for improved disturbance rejection of time-delayed processes. *Industrial & Engineering Chemistry Research*, [S.l.], v.46, n.7, p.2077–2091, 2007.
- SHEN, Y.; CAI, W.-J.; LI, S. Normalized decoupling control for high-dimensional MIMO processes for application in room temperature control HVAC systems. *Control Engineering Practice*, [S.l.], v.18, n.6, p.652–664, 2010.
- SHEN, Y.; SUN, Y.; LI, S. Adjoint transfer matrix based decoupling control for multivariable processes. *Industrial & Engineering Chemistry Research*, [S.l.], v.51, n.50, p.16419–16426, 2012.
- SRINIVAS, N.; DEB, K. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, [S.l.], v.2, n.3, p.221–248, 1994.

- STORN, R. On the usage of differential evolution for function optimization. In: NORTH AMERICAN FUZZY INFORMATION PROCESSING. *Proceedings*. . . [S.l.: s.n.], 1996. p.519–523.
- STORN, R.; PRICE, K. Differential evolution—A simple and efficient adaptive scheme for global optimization over continuous spaces [R]. *Berkeley: ICSI*, [S.l.], 1995.
- STORN, R.; PRICE, K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, [S.l.], v.11, n.4, p.341–359, 1997.
- TAKAHASHI, R. H. *Otimização escalar e vetorial*. [S.l.: s.n.], 2011.
- VAN VELDHUIZEN, D. A.; LAMONT, G. B. Multiobjective evolutionary algorithms: analyzing the state-of-the-art. *Evolutionary computation*, [S.l.], v.8, n.2, p.125–147, 2000.
- VILANOVA, R. IMC based Robust PID design: tuning guidelines and automatic tuning. *Journal of process Control*, [S.l.], v.18, n.1, p.61–70, 2008.
- VISIOLI, A. Model-based PID tuning for high-order processes: when to approximate. In: DECISION AND CONTROL, 2005 AND 2005 EUROPEAN CONTROL CONFERENCE. CDC-ECC'05. 44TH IEEE CONFERENCE ON. *Anais*. . . [S.l.: s.n.], 2005. p.7127–7132.
- VU, T. N. L.; LEE, J.; LEE, M. Design of multi-loop PID controllers based on the generalized IMC-PID method with Mp criterion. *International Journal of Control, Automation, and Systems*, [S.l.], v.5, n.2, p.212–217, 2007.
- VU, T. N. L.; LEE, M. An Extended Method of Simplified Decoupling for Multivariable Processes with Multiple Time Delays. *Journal of Chemical Engineering of Japan*, [S.l.], v.46, n.4, p.279–293, 2013.
- WANG, Q.-G.; ZHANG, Y.; CHIU, M.-S. Decoupling internal model control for multi-variable systems with multiple time delays. *Chemical Engineering Science*, [S.l.], v.57, n.1, p.115–124, 2002.
- WATANABE, E. H. *et al.* Tecnologia FACTS-tutorial. *SBA Controle & Automação*, [S.l.], v.9, n.1, p.39–55, 1998.
- WOOD, R.; BERRY, M. Terminal composition control of a binary distillation column. *Chemical Engineering Science*, [S.l.], v.28, n.9, p.1707–1717, 1973.

-
- XIE, W. F.; RAD, A. B. Fuzzy adaptive internal model control. *IEEE Transactions on Industrial Electronics*, [S.l.], v.47, n.1, p.193–202, 2000.
- XIONG, Q.; CAI, W.-J. Effective transfer function method for decentralized control system design of multi-input multi-output processes. *Journal of Process Control*, [S.l.], v.16, n.8, p.773–784, 2006.
- YANG, C.; SIMON, D. A new particle swarm optimization technique. In: NULL. *Anais...* [S.l.: s.n.], 2005. p.164–169.
- ZHANG, Q.; LI, H. MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, [S.l.], v.11, n.6, p.712–731, 2007.
- ZHENG, Q.; ZAFIRIOU, E. A local form of small gain theorem and analysis of feedback Volterra systems. *IEEE transactions on automatic control*, [S.l.], v.44, n.3, p.635–640, 1999.
- ZITZLER, E.; LAUMANN, M.; THIELE, L. SPEA2: improving the strength pareto evolutionary algorithm. *TIK-report*, [S.l.], v.103, 2001.
- ZITZLER, E.; THIELE, L. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE transactions on Evolutionary Computation*, [S.l.], v.3, n.4, p.257–271, 1999.