

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA QUÍMICA
ÁREA DE CONCENTRAÇÃO SISTEMAS DE PROCESSOS
QUÍMICOS E INFORMÁTICA

CONTROLE ÓTIMO NO TEMPO DE
COLUNAS DE DESTILAÇÃO

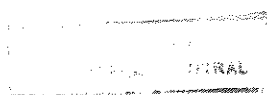
Autor: Andreas Schröder

Orientador: Prof. Dr. Mário de Jesus Mendes

Tese de Doutorado apresentada à Faculdade de Engenharia Química como parte dos requisitos exigidos para a obtenção do título de Doutor em Engenharia Química

Fevereiro de 1999

Campinas - SP



7909652

UNIDADE	BC
N.º CHAMADA:	UNICAMP
	Sch66c
V	Es.
TOMBO 30/	37414
PROC.	229/99
C	<input type="checkbox"/>
D	<input checked="" type="checkbox"/>
PREÇO	R\$ 11,00
DATA	28/04/99
N.º CPU	

CM-00122751-1


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

Sch66c Schröder, Andreas
Controle ótimo no tempo de colunas de destilação /
Andreas Schröder.--Campinas, SP: [s.n.], 1999.

Orientador: Mário de Jesus Mendes.
Tese (doutorado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Química.

1.Otimização matemática. 2. Destilação – Controle
automático. 3. Programação não-linear. I. Mendes, Mário
de Jesus. II. Universidade Estadual de Campinas.
Faculdade de EngenhariaQuímica. IV. Título.

Tese de Doutorado defendida e aprovada em 05 de fevereiro de 1999 pela banca examinadora constituída pelos professores doutores:



Prof. Dr. Mário de Jesus Mendes
Orientador



Prof. Dr. Cláudio Augusto Oller Nascimento



Prof. Dr. Luis Gimeno Latre

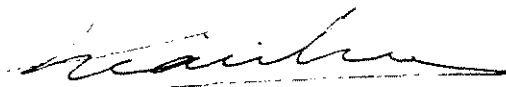


Prof.ª Dr.ª Maria Teresa Moreira Rodrigues



Prof.ª Dr.ª Ana Maria Frattini Fileti

Esta versão corresponde a redação final da Tese de Doutorado em Engenharia Química defendida pelo Eng. Andreas Schröder e aprovada pela Comissão Julgadora em 05.02.1999.



Prof. Dr. Mário de Jesus Mendes
Orientador

Agradecimentos

Ao Prof. Dr. Mário de Jesus Mendes pela orientação e atenção dedicada,

à Profa. Dra. Maria Teresa Moreira Rodríques pelas suas observações,

ao Prof. Dr. L.T.Biegler pela gentileza em ceder seu pacote de SQP,

ao CNPq pelo apoio financeiro,

à Rosa, da Secretaria da Pós-Graduação e

a todos os professores, colegas, amigos, e os demais funcionários do DESQ e da FEQ.

Sumário

Lista de Figuras	I
Lista de Tabelas	III
Nomenclatura	IV
Resumo	IX
<i>Abstract</i>	XI
1. Introdução	1
2. Revisão Bibliográfica	5
2.1. Otimização Dinâmica	5
2.2. Programação Quadrática Seqüencial	9
2.2.1. Programação Quadrática	11
2.2.2. Problemas de Alta Dimensão, usando SQP	13
2.2.3. Otimização Dinâmica, usando SQP	15
2.2.3.1. Método Seqüencial	16
2.2.3.2. Método Simultâneo	17
2.2.3.3. Vantagens e Desvantagens	22
2.3. Programação Dinâmica	23
2.3.1. Discretização no Tempo	26
2.3.2. Discretização dos Domínios	28
2.3.3. Seções de Tempo variável	30
2.3.4. Programação Dinâmica Iterativa	32
2.4. Modelos Simplificados e Modelos de Ordem Reduzida	35
2.5. Controle Ótimo no Tempo de Colunas de Destilação	39
2.6. Formulação Matemática do Problema	42
2.6.1. Parâmetro do Tempo Final	42
2.6.2. Função Objetivo Aumentada	44
2.7. Conclusões	46
3. Métodos Utilizados	47
3.1. Modelo Matemático da Coluna	48
3.2. Formulação do Problema de Controle Ótimo no Tempo	55

3.3. Redução da Ordem do Modelo	58
3.4. Formulação do Método Simultâneo	62
3.5. Algoritmo de IDP	63
4. Resultados	70
4.1. Eficiência do Método de Colocação Ortogonal no Tempo	71
4.2. Eficiência do Pacote de SQP escolhido	76
4.3. Eficiência do Método de Ordem Reduzida, usando SQP	78
4.4. Eficiência do Modelo de Ordem Reduzida, usando IDP	85
4.5. Chute Inicial longe da Solução Ótima, usando SQP	89
4.6. Aplicação Mista, IDP-SQP	91
4.7. Aplicação Mista, SQP-SQP	93
4.8 Análise dos Resultados	96
4.8.1. Custo Computacional e Robustez	97
4.8.2. Processo Transiente Otimizado	100
5. Conclusões e Sugestões	104
Referências Bibliográficas	106
Anexos	114

Lista de Figuras

2.2-1	Colocação Ortogonal; com $K = 3$ Pontos de Colocação internos.	21
2.2-2	Colocação Ortogonal em Elementos Finitos; com $K/K/K=2/2/2$ e $NE=3$	22
2.3-1	Programação Dinâmica	24
2.3-2	Estratégia de Controle Ótima	25
2.3-3	Discretização do Domínio Admissível	29
2.3-4	Determinação do Ponto de Continuação	30
3.1-1	Esquema da Coluna de Destilação Binária Prato-a-Prato	49
3.3-1	Modelo de Ordem Reduzida da Coluna de Destilação	58
3.5-1	Grade de Controle com $M_u = 3$	65
3.5-2	Região Inicial da Grade de Estado	65
3.5-3	Geração das Grades de Estado com $M_x = 5$ e $S = 10$	66
3.5-4	Integração parcial na Seção de Tempo S	66
3.5-5	Integração parcial na Seção de Tempo $S-1$	67
3.5-6	Escolha da Trajetória Ótima obtido ao Final da Iteração	68
4.1-1	Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada	72
4.1-2	Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada	73
4.1-3	Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada	74
4.1-4	Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada	75
4.1-5	Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada	76
4.4-1	Convergência até o Ótimo	86
4.4-2	Erro de Continuação	87
4.4-3	Trajetória Ótima da Fração Molar no Destilado obtida pelo método IDP	88
4.4-4	Estratégia de Controle Ótima, IDP	88

4.6-1 Trajetória Ótima da Fração Molar no Destilado obtida pelo método misto IDP-SQP	93
4.8-1 Custo Computacional com Chute Inicial de $t^{fin} = (7, \dots, 11, \dots, 15)$ min	98
4.8-2 Custo Computacional com Chute Inicial de $t^{fin} = (1, \dots, 41, \dots, 80)$ min	99
4.8-3 Custo Computacional com Chute Inicial de $t^{fin} = (1, \dots, 250, \dots, 500)$ min	100
4.8-4 Trajetória Ótima da Fração Molar no Destilado e Tempo Ótimo	101
4.8-5 Estratégia de Controle Ótima ($K/K/K = 4/3/4$, $N = 7$)	103

Lista de Tabelas

3.1-1	Dados da Coluna	55
4.2-1	Tempo de CPU [s] / Exemplo I	77
4.2-2	Tempo de CPU [s] / Exemplo II	78
4.3-1	Número de Variáveis e (Equações)	79
4.3-2	Tempo de CPU [s]	80
4.3-3	Número de Variáveis e (Equações)	80
4.3-4	Tempo de CPU [s] e (Número de Iterações)	81
4.3-5	Tempo de Transição Ótimo [min]	81
4.3-6	Número de Variáveis e (Equações)	82
4.3-7	Tempo de CPU [s] e (Número de Iterações)	83
4.3-8	Tempo de Transição Ótimo [min]	84
4.4-1	Tempo de CPU [s]	89
4.5-1	Tempo de CPU [s] e (Número de Iterações)	90
4.7-1	Número de Variáveis e de Equações (Etapa de Inicialização)	94
4.7-2	Tempo de CPU [s] e (Número de Iterações) (Etapa de Inicialização)	95
4.7-3	Tempo de CPU [s] e (Número de Iterações); Tempo de Transição Ótimo, t^{fin*} (Etapa de Inicialização)	96

Nomenclatura

Letras Latinas

B	[kmol/min]	Vazão do Produto de Fundo
\mathbf{B}	[-]	Matriz Definida Positiva
C	[-]	Número de Componentes
d	[-]	Direção de Busca
D	[kmol/min]	Vazão do Destilado
f	[-]	Função Objetivo
F	[-]	Função
	[kmol/min]	Vazão da Alimentação
g	[-]	Função
	[-]	Restrição (Inequação)
h	[-]	Restrição (Equação)
I	[-]	Índice de Desempenho, Critério de Otimização
J	[-]	Índice de Desempenho, Critério de Otimização
K	[-]	Número de Pontos de Colocação Internos no Tempo
ℓ	[-]	Largura variável do Intervalo no Tempo
L	[kmol/min]	Vazão do Líquido (Refluxo)
	[-]	Função de Lagrange
	[-]	Largura do Intervalo no Tempo
L_m	[kmol/min]	Vazão do Líquido do Prato m
m	[-]	Número de Prato
m_u	[-]	Número de Variáveis de Controle
m_x	[-]	Número de Variáveis de Estado
M	[kmol]	<i>Holdup</i> do Líquido
M_o	[kmol/min]	<i>Holdup</i> do Líquido acima do vertedouro
M_u	[-]	Número de Variáveis de Controle Permitidos
	[kmol/min]	<i>Holdup</i> do Líquido no vertedouro
M_x	[-]	Número de Pontos de Grade de cada Variável de Estado

N	[-]	Número de Pontos de Colocação Internos ao Longo da Coluna
NE	[-]	Número de Elementos Finitos no Tempo
N_{eq}	[-]	Número de Equações Diferenciais
N_F	[-]	Locação do Prato de Alimentação
N^R	[-]	Número de Pratos da Seção de Retificação
N^S	[-]	Número de Pratos da Seção de Esgotamento
N_T	[-]	Número Total de Pratos (Incluindo Condensador e Refervedor)
P	[-]	Polinômio Discreto
q_F	[-]	Fração do Líquido na Alimentação
S	[-]	Número de Intervalos de Tempo
s	[-]	Ponto de Colocação
t	[min]	Tempo
u	[-]	Variável de Controle
U	[-]	Domínio de Controle
V	{kmol/min}	Vazão do Vapor
x	[-]	Variável de Estado Fração Molar no Líquido do Componente Leve
X	[-]	Domínio de Estado
y	[-]	Fração Molar no Vapor do Componente Leve
z_F	[-]	Fração Molar do Componente Leve na Alimentação

Letras Gregas

α	[-]	Volatilidade relativa;
	[-]	Local de Nós entre Elementos Finitos no Tempo
δ	[-]	Parâmetro da Função Objetivo
Δ	[-]	Primeira Diferença "Forward"
∇	[-]	Primeira Diferença "Backward"
ε	[-]	Erro
ϕ	[-]	Polinômio de Lagrange

γ	[-]	Parâmetro da Função Objetivo
Γ	[-]	Polinômio de Lagrange
φ	[-]	Constante
λ	[-]	Passo Ótimo
	[-]	Efeito Inicial da Variação de Vapor sobre a Vazão do Líquido
nos		
		Pratos
μ	[-]	Fator de Redução
τ	[min]	Tempo Normalizado
τ_L	[min]	Constante de Tempo dos Pratos
ψ	[-]	Polinômio de Lagrange

Sobrescritos

<i>fin</i>	Final
<i>i</i>	Índice dos Instantes no Tempo
<i>j</i>	Índice dos Instantes no Tempo
<i>k</i>	Índice dos Instantes no Tempo
<i>R</i>	Retificação
<i>S</i>	Esgotamento
<i>stat</i>	Estado Estacionário Desejado
<i>T</i>	Transposta
<i>0</i>	Começo
<i>'</i>	Primeira Derivada
<i>•</i>	Primeira Derivada no Tempo
<i>+</i>	Marca para Intervalo no Tempo antes do Ponto
<i>-</i>	Marca para Intervalo do Tempo após do Ponto
<i>o</i>	Inicial
<i>*</i>	Ótimo

Subscritos

B	Produto de Fundo
D	Destilado
F	Alimentação
j	Índice dos Instantes ao Longo do Espaço
k	Índice dos Instantes ao Longo do Espaço
L	Líquido
m	Número de Prato
max	Máximo
min	Mínimo
n	Índice dos Instantes ao Longo do Espaço
o	Vertedouro
s	Ponto de Colocação
T	Total
V	Vapor
0	Começo

Abreviaturas

eq(s).	Equação (Equações)
min	minuto(s)
s	Segundo(s)

Siglas

BFGS	Método de Atualização (Broyden-Fletcher-Goldfarb-Shanno)
BVP	Problema de Valores de Contorno

CPU	Unidade Central de Processamento
DAE	Equações Diferenciais e Algébricas
DFP	Método de Atualização (Davidson-Fletcher-Powell)
DP	Programação Dinâmica
IDP	Programação Dinâmica Iterativa
IVP	Problema de Valor Inicial
NLP	Programação Não-Linear
QP	Programação Quadrática
RND	<i>Range and Null Space Decomposition</i>
SQP	Programação Quadrática Seqüencial

Resumo

Colunas de destilação formam parte dos equipamentos mais importantes da Indústria Química. Processos de troca de produto, ou seja, levando a coluna de um estado estacionário inicial até um novo estado estacionário com novas especificações, levam muito tempo. Tempo perdido para a produção e caro, produzindo lixo e gastando energia e matéria prima. Por conseguinte, o desenvolvimento de métodos para otimizar este tempo não-produtivo foi alvo de vários trabalhos nos últimos anos. Um problema ainda muito freqüente é a velocidade e a robustez dos algoritmos de otimização. O objetivo do presente trabalho é desenvolver um método robusto e rápido para o problema de controle ótimo no tempo de uma coluna de destilação, em processo de troca de produto.

O método de colocação ortogonal é aplicado duas vezes no modelo dinâmico da coluna. Primeiro, na forma discreta, ao longo do espaço da coluna, para aproximar o modelo prato-a-prato nos pontos de colocação, reduzindo assim a ordem do modelo. Segundo, na forma contínua no tempo, para discretizar o modelo dinâmico. Em seguida é escolhido um algoritmo SQP, adequado para problemas de alta dimensão com um número de graus de liberdade pequeno e matriz esparsa, para resolver o problema de otimização. Como o SQP é um método rápido mas pouco robusto, um método IDP é desenvolvido para inicializar o algoritmo SQP. A dupla aplicação da colocação ortogonal ou seja, a possibilidade de reduzir desta maneira a dimensão do problema de otimização, forma a base de um método superior, a autoinicialização do próprio SQP.

Os resultados mostram que o método desenvolvido é muito robusto e converge também quando são usados chutes iniciais extremamente afastados da solução ótima. Os métodos da dupla aplicação da colocação ortogonal e a autoinicialização de SQP se mostram também superiores na velocidade. Acredita-se que o método apresentado neste trabalho é válido também para resolver outros problemas de otimização dinâmica. Pressupõe-se que o modelo dinâmico do

processo ou é simples, ou pode ser simplificado pelas técnicas indicadas. Além disso, deve ser possível reduzir adequadamente a ordem do modelo, para permitir a autoinicialização.

Abstract

Distillation columns are one of the most important equipments of the chemical process industry. Product changing processes, that is, driving a column from an initial steady state to a new desired one with new process specifications, are very time consuming processes. Lost and expensive time, producing waste and consuming energy and raw materials. Therefore, the development of methods, which optimize this unproductive time, was the target of many publications in the last years. A still frequent problem is the velocity and the robustness of the optimization algorithms. The target of the work presented here is the development of a robust and fast method to solve a time optimal control problem of a distillation column in a product changing process.

The orthogonal collocation method is applied two times at the dynamic column model. First, in discrete form about the column space to approximate the tray-to-tray model at the collocation points, reducing the order of the process model. Second, in continuous form about the time to discretize the dynamic model. Afterwards a SQP algorithm, suitable for high dimensioned optimization problems with few degrees of freedom and a sparse matrix, is chosen. As the SQP algorithm is a fast but not robust method, an IDP approach is developed to initialize the SQP algorithm. The double application of orthogonal collocation, which means, the possibility of reduction of the dimension of the optimization problem, leads to a much better approach, the auto-initialized SQP approach.

The results show that the developed method is very robust and converges also when starting out from a very poor initial guess. It is shown that the methods of double orthogonal collocation application and the auto-initialization of SQP are superior also in velocity. It can be assumed that this method will be applicable too to other kinds of dynamic optimization problems, if the process model, which forms the constraints, can be simplified and its order conveniently reduced.

1. Introdução

A concorrência agravada, o constante aumento dos preços das matérias-primas e a crescente exigência na qualidade dos produtos têm levado a uma redução considerável da margem de lucro na produção de produtos químicos. Por outro lado, os altos custos de armazenagem e as baixas durabilidades de muitos produtos, especialmente os derivados de matérias-primas renováveis, levam cada vez mais a uma produção "*just-in-time*" e a unidades multiproduto e, conseqüentemente, a um incremento de troca de produtos e processos de arranques nas instalações químicas. Processos de troca de produtos e de arranque são geralmente períodos não-produtivos cuja duração precisa ser minimizada.

Colunas de destilação formam parte dos aparelhos mais usados na separação de misturas líquidas na prática industrial. Portanto, o desenvolvimento de estratégias operacionais otimizados no tempo para estes aparelhos possui um significado particular do ponto de vista econômico. O rápido alcance do estado estacionário desejado, conforme à especificação do produto exigido, minimiza o tempo necessário para este tipo de processo não-produtivo, além de reduzir os gastos de energia e matéria prima, como também a quantidade dos produtos intermediários que precisam de ser reciclados, depositados ou queimados.

Em vez de se levar, no instante inicial, as entradas do processo aos seus novos valores desejados, por uma variação em salto, como feito freqüentemente anteriormente, uma estratégia modificada, muito usada hoje em dia, foi apresentada por KRUSE *et. al.* (1995) para minimizar o tempo de transição do processo não-econômico da troca de produtos: Para colunas de destilação, a razão de refluxo é levada à zero ou infinito durante a transição. Desta maneira a coluna trabalha com refluxo total ou fluxo do destilado total, dependendo do objetivo da transição, levando o ponto de trabalho de uma concentração do destilado elevada para uma mais baixa ou vice versa. Num dado momento posterior a razão de refluxo é levada finalmente ao seu novo ponto de trabalho desejado e a coluna é operada no seu novo estado estacionário. Além da

razão de refluxo é possível usar a capacidade do refeedor para acelerar o processo de transição.

Um dos problemas deste método de aceleração da transição de um processo de troca de produto está na determinação do tempo ao fim do qual as variáveis de controle devem ser levados dos seus valores extremos para os valores finais e no cálculo do tempo mínimo total. Neste contexto, a implementação de novas estratégias de produção precisa de métodos mais rápidos e robustos para resolver problemas de controle ótimo no tempo.

A otimização do tempo de transição de uma coluna de destilação de um estado estacionário inicial até um novo estado estacionário desejado é um problema de controle ótimo no tempo. Modelos que descrevem o comportamento transiente de colunas de destilação geralmente são sistemas de equações não-lineares contendo equações diferenciais e algébricas, sujeitos a condições de contorno e especificações de processo incorporadas como inequações no modelo. Em consequência, problemas de controle ótimo no tempo de colunas de destilação levam à solução de um modelo dinâmico da coluna, o qual se caracteriza, além da alta complexidade e extrema não-linearidade, pela sua alta dimensão com um número de graus de liberdade relativamente pequeno.

A descrição geral do problema de otimização dinâmica é dada pelas seguintes equações.

$$\min_u f(x, u, t), \quad (1.1-1a)$$

sujeito à:

$$\frac{dx}{dt} = F(x, u, t), \quad (1.1-1b)$$

$$h(\mathbf{x}, \mathbf{u}, t) = 0, \quad (1.1-1c)$$

$$g(\mathbf{x}, \mathbf{u}, t) \leq 0. \quad (1.1-1d)$$

Na função objetivo consideram-se o ponto final e a transição do processo. As restrições representam as equações diferenciais e algébricas do modelo de processo; as inequações são as especificações do processo e os limites das variáveis.

Os modelos dinâmicos que restringem o problema de otimização geralmente possuem um número elevado de equações e variáveis, causando tempos de computação (tempo de CPU) significativos. Para se aplicar métodos de otimização em problemas de controle ótimo no tempo de colunas de destilação é preciso que estes algoritmos sejam rápidos e robustos, ou seja, capazes de convergir a partir de qualquer chute inicial num tempo considerado razoável. Sobretudo a robustez, a capacidade de convergir mesmo partindo de um chute inicial longe da solução ótima, é de uma grande importância, tendo-se em conta que na prática industrial os sistemas de controle de processos são manejados por não-especialistas na área de otimização.

Os métodos usados para resolver pelo menos uma subclasse de (1.1) podem ser classificados nos seguintes tipos básicos: a) Métodos iterativos baseados no Cálculo das Variações; b) Métodos de Programação Não Linear (NLP); c) Métodos de Programação Dinâmica (DP). Os métodos NLP por sua vez podem ainda ser divididos em duas categorias: 1) Métodos NLP de Trajetória Factível ("*feasible path*"), também conhecidos por métodos de Parametrização do Vetor de Controle ("*control vector parametrization*") ou métodos de Modelo Embutido ("*embedded model*"); 2) Métodos NLP Simultâneos.

Basicamente, os métodos NLP de Trajetória Factível usam um perfil discretizado do controle $\mathbf{u}(t)$; a cada iteração o estado $\mathbf{x}(t)$ é determinado por integração do sistema de equações algébrico-diferenciais do modelo por um algoritmo adequado. Já nos métodos NLP simultâneos tanto o perfil do controle $\mathbf{u}(t)$ como o do estado $\mathbf{x}(t)$ são

discretizados; deste modo, as equações algébrico-diferenciais do modelo são transformados num sistema de equações algébricas não lineares, que são tratados como restrições no âmbito do algoritmo NLP (BIEGLER, 1992).

O objetivo deste trabalho é de desenvolver um método rápido e robusto de controle ótimo no tempo para colunas de destilação. O método precisa resolver o problema de otimização dinâmica com um baixo custo computacional e ser suficientemente robusto para encontrar uma solução ótima, mesmo partindo de um ponto inicial longe da mesma. Como se verá adiante, neste trabalho foram usados basicamente dois métodos de otimização dinâmica: Um métodos NLP simultâneo, baseado num algoritmo SQP disponível, e um método IDP ("*iterative dynamic programming*"), desenvolvido no trabalho com base em informações disponíveis na literatura (LUUS, 1990b).

Uma vez apresentada a motivação para este trabalho, parte-se para a explicação da divisão do mesmo. Em seguida à introdução será feita no capítulo 2 uma revisão bibliográfica sobre métodos de otimização e problemas de controle ótimo no tempo, assim como sobre modelos reduzidos de colunas de destilação de pratos. No capítulo 3 será explicada a aplicação dos métodos de otimização e do modelo reduzido, escolhidos neste trabalho, no problema de controle ótimo no tempo de uma coluna de destilação binária. Os resultados obtidos com estes métodos serão apresentados no capítulo 4 e no capítulo 5 será feita uma análise crítica dos resultados, apresentando as conclusões e sugestões para futuros trabalhos.

2. Revisão Bibliográfica

Será feita aqui uma revisão da literatura referente ao tema de controle ótimo no tempo de colunas de destilação. O primeiro subcapítulo é dedicado à otimização dinâmica em geral. Em seguida, nos próximos subcapítulos, será analisada a literatura sobre os dois métodos de otimização considerados os mais adequados, justificando a escolha destes. Logo em seguida, após um breve análise da literatura sobre modelos de ordem reduzida, será discutida a literatura recente sobre problemas de controle ótimo no tempo de colunas de destilação. Os próximos subcapítulos darão uma visão sobre a formulação matemática do problema e no último subcapítulo conclui-se a escolha dos algoritmos e métodos usados.

2.1. Otimização Dinâmica

Sejam eqs. (1.1-1b) e (1.1-1c) o modelo dinâmico do processo a otimizar. $x(t)$ são as variáveis de estado e $u(t)$ as variáveis de entrada, também chamadas de variáveis de controle. Tanto as variáveis de estado como as de controle podem estar sujeitas a limitações (valores limites), sendo x_{\max} e u_{\max} os respectivos limites superiores e x_{\min} e u_{\min} os limites inferiores.

Procura-se então aquele vetor de controle $u(t)$, dentro do seus limites de u_{\min} e u_{\max} , para o qual a função objetivo f toma um valor extremo. O controle u que consegue isso ao mesmo tempo que satisfaz as restrições descritas pelas equações de modelo, chama-se controle ótimo. O correspondente vetor do estado $x(t)$ chama-se trajetória ótima. É comum para modelos dinâmicos escrever a função objetivo f como I ou J , chamando-a de Índice de Desempenho ou Critério de Otimização.

Nem todos os métodos de otimização estática são também apropriados para resolver problemas de otimização dinâmica, descritos pelas eqs. (1.1-1). Problemas de otimização com modelos dinâmicos muitas vezes são também chamados de problemas

de otimização de estrutura (HOFER & LUNDERSTÄDT, 1975). Dependendo das condições de contorno trata-se de um Problema de Valores Iniciais (IVP), de um Problema de Valores Finais ou de um Problema de Condições de Contorno (BVP). Problemas de otimização com modelos dinâmicos quase sempre podem ser transformados numa das seguintes formas gerais:

$$J = f[\mathbf{x}(t^{\text{fin}}), t^{\text{fin}}]; \quad (\text{problema de Mayer}) \quad (2.1-1a)$$

$$J = \int_0^{t^{\text{fin}}} f^0[\mathbf{x}(t), \mathbf{u}(t), t] dt; \quad (\text{problema de Lagrange}) \quad (2.1-1b)$$

$$J = f[\mathbf{x}(t^{\text{fin}}), t^{\text{fin}}] + \int_0^{t^{\text{fin}}} f^0[\mathbf{x}(t), \mathbf{u}(t), t] dt. \quad (\text{problema de Bolza}) \quad (2.1-1c)$$

Problemas de otimização dinâmica exigem, na maioria dos casos a solução de sistemas de Equações Diferenciais e Algébricas (DAE) de índice 1. BRYSON & HO (1975) descreveram diferentes Métodos de Variação para resolver problemas de otimização algébrico-diferenciais. Estes procedimentos analíticos para resolver pequenos problemas de projeto usam algoritmos iterativos baseados em condições de variação. Eles propuseram diferentes algoritmos iterativos do vetor de controle, os quais incluem a solução das equações do modelo "*forward in time*" e as equações adjacentes "*backward in time*", assim como fazem uma atualização periódica do perfil de controle para maximizar a função de Hamilton.

HOFER & LUNDERSTÄDT (1975) usaram na otimização de sistemas dinâmicos o Cálculo de Variações clássico na forma de Hamilton, assim como sua generalização, o Princípio de Máximo de Pontryagin (PONTYAGIN *et al.*, 1962). Eles propuseram o uso da Programação Dinâmica (DP) de acordo com BELLMAN (1957), de preferência para sistemas dados em forma discreta. Para resolver problemas dinâmicos ótimos no tempo os autores propõem o uso do Cálculo de Variações na descrição de Hamilton e o Princípio de Máximo de Pontryagin em exemplos práticos.

Também FÖLLINGER (1988) e PAPAGEORGIOU (1991) descreveram nos seus trabalhos o uso do Princípio de Máximo de Pontryagin em problemas de controle ótimo no tempo. PAPAGEORGIOU (1991) além disso descreveu as tentativas feitas na transferência de diferentes métodos numéricos da otimização estáticas em formulações de problemas dinâmicos. Ele fez um pequeno resumo sobre a tentativa do uso das funções de penalização de multiplicadores e de penalização exata, mas ambas resultaram pouco eficientes em problemas de otimização dinâmica. O método de Programação Quadrática Sucessiva (SQP), um método NLP, o qual teve muito sucesso na otimização estática, tem sido aplicado a problemas dinâmicos, sem chegar a resultados concretos até hoje, segundo o mesmo autor. No seu trabalho, o autor usou métodos de Gradientes e de Gradientes Conjugados em problemas de otimização dinâmica. Ele chegou à conclusão que, entre os testados, os métodos dos Gradientes Conjugados e a eliminação das más condições ao evitar os termos de penalização, levam aos melhores resultados.

Para a solução de problemas de controle ótimo, em geral não-lineares, ROSEN & LUUS (1991) propuseram, em vez dos métodos tradicionais baseados no Princípio de Máximo de Pontryagin, usar a NLP. Eles usaram o método de SQP para resolver problemas de controle ótimo gerais e declararam, ao contrário do que diz PAPAGEORGIOU (1991), que o método SQP é a técnica geral mais eficiente para problemas da NLP.

Um algoritmo misto foi testado com sucesso por HARTIG & KEIL (1993). Os autores utilizaram um algoritmo de Programação Dinâmica Iterativa (IDP) para as primeiras iterações, seguido por um algoritmo SQP para a finalização de um problema de otimização de uma cascada de reatores.

DADEBO & McAULEY (1995a,b) usaram IDP para resolver problemas de controle ótimo no geral ou controle ótimo no tempo, respectivamente, e propuseram na engenharia química o uso de uma função de penalização de erro absoluto para problemas de otimização dinâmica com restrições. Eles usaram um critério de

supressão de movimento dentro do algoritmo de IDP para conseguir que movimentos de controle demasiado excessivos sejam penalizados. Os autores testaram o algoritmo IDP numa variedade de problemas e compararam os resultados, o número de iterações e os tempos precisos para resolver os problemas, obtendo resultados ainda melhores nos seus exemplos do que com o método SQP, muito usado na literatura recente para este tipo de problema.

Tal como foi proposto por BOJKOV & LUUS (1992), também GUPTA (1995) propôs como princípio não confiar num só método quando se tratar de problemas altamente não-lineares, já que freqüentemente é muito difícil avaliar se o ótimo encontrado é global ou não. Por conseguinte eles propuseram um "*cross check*" com pelo menos mais um método adicional, aplicando para isso IDP ou o método de "*semiexhaustive-search*", método de IDP modificado, apresentado por GUPTA (1995).

LI & WOZNY (1996) aplicaram a Programação Dinâmica Diferencial, um algoritmo baseado no método da DP, e SQP, usando a formulação simultânea, em problemas de otimização de processos em batelada de reatores e colunas de destilação. Os autores chegaram à conclusão que o algoritmo baseado na DP não é adequado para sistemas de alta dimensão. Eles acharam o SQP um método muito rápido mas pouco robusto, precisando de muito cuidado na escolha do chute inicial.

LUUS (1996) mostrou o uso de IDP para resolver problemas de controle ótimo de alta dimensão e um número de graus de liberdade relativamente grande. O autor aplicou o IDP com sucesso em um problema com 250 variáveis de controle e 250 variáveis de estado. O custo computacional de aproximadamente 40 horas deve-se considerar muito alto.

Ao finalizar este subcapítulo serão apontadas aqui umas observações feitas por KEIL (1996). A maior parte dos métodos NLP foram desenvolvidos para achar um extremo local. Os únicos métodos conhecidos, capazes de achar limites inferiores e superiores de um extremo global, comprovado computacionalmente, são os métodos de

intervalos. Para determinados processos a Programação Dinâmica também garante um ótimo global. Consequentemente, neste trabalho foi escolhido, além do método SQP, prometendo ser um método muito rápido, também o método IDP, tanto para testar a sua capacidade em resolver o problema de controle ótimo no tempo de uma coluna de destilação, como para ser usado como método de inicialização, em uma método misto, como proposto por HARTIG & KEIL (1993).

2.2. Programação Quadrática Seqüencial

O SQP, método NLP para problemas com restrições, conhecido na literatura como Programação Quadrática Seqüencial (FRIEDLANDER, 1994), ou Programação Quadrática Sucessiva (KEIL, 1996), método Wilson-Han-Powell (nomes dos inventores do método) ou método Lagrange-Newton (GAL, 1991), é reconhecido como o melhor método de otimização até aproximadamente 100 variáveis (BIEGLER, 1988; EDGAR & HIMMELBLAU, 1989 e GAL 1991).

O método usa a programação quadrática seqüencialmente aproximando localmente a função objetiva por uma função quadrática e as restrições por funções lineares. Por conseguinte é resolvido um subproblema quadrático succesivamente, (2.2-1) e (2.2-2), para determinar a direção de busca d .

$$\min F(d) = \nabla f(x)d^T + \frac{1}{2}d^T B d. \quad (2.2-1)$$

sujeito á:

$$d^T \nabla h(x) + h(x) = 0, \quad (2.2-2a)$$

$$d^T \nabla g(x) + g(x) \leq 0. \quad (2.2-2b)$$

Como derivadas de segunda ordem muitas vezes são difíceis de se calcular, a matriz hessiana é aproximada por uma matriz definida positiva B , a qual é construída por uma fórmula Quase-Newton, o que significa que somente derivadas primeiras são necessárias. Antes do primeiro passo de iteração o B^0 é geralmente definido pela matriz identidade I , para depois ser melhorado em cada um dos passos de iteração seguinte. Convexidade consegue-se através da fórmula de atualização de Davidon-Fletcher-Powell (DFP) ou Broyden-Fletcher-Goldfarb-Shanno (BFGS) (GAL, 1991). Assim evita-se, do mesmo modo que nos métodos Quase-Newton, a difícil averiguação exata da matriz hessiana.

A direção de busca d resulta como solução do problema de otimização com a função objetivo quadrática (2.2-1) e as restrições lineares (2.2-2) no ponto x . Depois da determinação de d calcula-se, tal como na otimização não restringida, o passo ótimo λ , minimizando

$$\min_{\lambda \geq 0} f[x + \lambda d(x)]. \quad (2.2-3)$$

O algoritmo simplificado do método SQP é o seguinte:

Passo 1: Escolher um chute inicial e colocar o número de iterações em 1;

Passo 2: Resolver o subproblema quadrático formado pelas equações (2.2-1) e (2.2-2), obtendo assim uma solução para a direção de busca d e do vetor de multiplicadores de Lagrange associado à solução de d ;

Passo 3: Sendo $d \leq \varepsilon$ o algoritmo pára, caso contrário segue no passo 4;

Passo 4: Calcular o passo ótimo λ , resolvendo a equação (2.2-3);

Passo 5: Calcular o vetor x , usado na iteração seguinte, sendo o vetor o resultado do termo $(x + \lambda d)$;

Passo 6: Calcular B , usado na iteração seguinte, aplicando um método de atualização (por exemplo os métodos DFP ou BFGS);

Passo 7: Aumentar o número de iteração por 1 e seguir com passo 2.

Nos últimos anos foram feitos numerosos avanços no desenvolvimento do algoritmo SQP. Nos próximos subcapítulos serão discutidas algumas publicações, descrevendo avanços no desenvolvimento do método SQP, por exemplo para ser usado na solução de problemas de alta dimensão ou em problemas dinâmicos. Hoje em dia os algoritmos de SQP são usados em pacotes comerciais como ASPEN PLUS (KEIL, 1996), e várias bibliotecas de software matemática como NAG ou IMSL. Amplas descrições dos princípios do SQP, assim como das condições de otimalidade de primeira e segunda ordem, são encontrados em EDGAR & HIMMELBLAU (1989), GAL(1991), FRIEDLANDER (1994) e BAIER *et al.* (1994).

2.2.1. Programação Quadrática

As virtudes de convergência dependem, além do método de determinação do passo ótimo, sobretudo da escolha do algoritmo da Programação Quadrática (QP), sendo a parte principal do método de SQP a solução repetida do subproblema quadrático. Para a QP, como caso especial da NLP caracterizado por uma função objetivo quadrática com restrições lineares, existem diversos algoritmos, como os métodos de Wolfe ou de Hildreth-d'Esopo (BRONSTEIN *et.al.*, 1995) ou o método de Lemke (GAL, 1991), também conhecido como "*Complementary Pivoting*", como também o método "*Range-Null-Space*" (BIEGLER, 1988).

Estão a disposição algoritmos muito eficientes da QP para determinar d , resolvendo o subproblema quadrático (2.2-1) com as restrições (2.2-2) e outras restrições por exemplo da forma:

$$\mathbf{x}_{min} \leq \mathbf{x} + \lambda \mathbf{d} \leq \mathbf{x}_{max} . \quad (2.2-4)$$

BIEGLER (1984) propôs o uso de algoritmos padrões de passos finitos de QP ("*Finite-Steps-Standard*"). O autor recomenda os pacotes VE02AD, desenvolvido por Fletscher (disponível na biblioteca HARWELL) ou, tal como VASANTHARAJAN & BIEGLER (1988), SOL/QPSOL, desenvolvido por Gill, Murray, Saunders e Wright e usado como método QP no pacote E04UCF da biblioteca NAG.

BIEGLER & CUTHRELL (1985) chegaram a conclusão que o pacote de SQP de Gill e Murray é mais apropriado que o VF02AD, geralmente usada até então, o qual usa como método de QP o pacote VE02AD. A razão desta escolha, segundo os mesmos autores, está no elevado custo computacional no processamento da atualização das restrições e na instabilidade freqüentemente aparecendo no VE02AD.

Para resolver problemas quadráticos GAL (1991) propôs aplicar um método de restrições ativas ou métodos que incluem diretamente as condições de Kuhn-Tucker. O autor apresenta o método de Lemke como um método QP adequado.

Mais recentemente, SCHMID & BIEGLER (1994) usaram o pacote QPKWIK, comprovando a superioridade dele sobre outros pacotes aqui mencionados. Eles compararam os pacotes QPSOL, VE17AD e QPKWIK e chegaram à conclusão que o QPKWIK é muito mais rápido do que os outros dois. QPKWIK encontra inicialmente o ótimo não restringido e em seguida acrescenta as restrições para o conjunto ativo, até conseguir uma solução ótima. Em particular, um hessiano definido positivo provoca que a função objetivo diminuía monótona enquanto novas restrições são acrescentadas o que leva ao resultado que uma circulação do conjunto ativo seja evitada e além disso resulta numa redução das restrições. Além disso os autores afirmaram que quando a quantidade das restrições, as quais podem ser ativas, seja limitada pelas variáveis do problema quadrático, o algoritmo utilizado por QPKWIK leva à convergência, com uma

quantidade finita de passos. Mas na prática, isso não é garantido por causa dos erros de arredondamento.

Uma visão geral muito boa sobre os pacotes de QP disponíveis (salvo os mais recentes como QPKWIK), assim como sobre os pacotes contendo um algoritmo completo de SQP, encontra-se no trabalho de LUCIA & XU (1990).

2.2.2. Problemas de Alta Dimensão, usando SQP

A parte do algoritmo SQP que mais tempo de CPU gasta, além do armazenamento dos dados resultando dos cálculos da função e do gradiente, é sem dúvida a solução do problema quadrático. Para aquela operação precisa-se do armazenamento e da atualização da matriz hessiana aproximada, em cada passo de intervalo, o que aumenta o número de variáveis do problema. Para problemas demasiado extensos chamados de problemas de alta dimensão, por exemplo com mais de 100 variáveis, o esforço de computação para resolver o problema quadrático com métodos densos de QP seria muito elevado e ineficiente. Consequentemente, ao contrário do SQP, BIEGLER (1988) achou MINOS, desenvolvido por Murtagh & Saunders, um método adequado para problemas de alta dimensão.

Para que também os métodos SQP trabalhassem eficientemente em problemas de elevada dimensão, BIEGLER (1988) propôs não usar as fórmulas de atualização Quase-Newton as quais aproximam o hessiano da função de Lagrange, e sim formular a matriz hessiana numa forma esparsa e adaptar o método da QP para aproveitar as estruturas esparsas do hessiano e dos gradientes das restrições. Isto não só levaria a uma solução eficiente da QP, mas também a uma melhor aproximação do hessiano e menos iterações do SQP, por causa da introdução de uma estrutura esparsa.

VASANTHARAJAN & BIEGLER (1988) descobriram que uma matriz de Hesse, aproximada por uma fórmula de atualização definida positiva densa, converge muito

lentamente e propuseram a atualização de uma matriz de Hesse projetada, em vez da matriz inteira, através de uma Decomposição "*Range-Null-Space*" (RND). O método baseia-se no princípio de que problemas de otimização de processos geralmente possuem muitas variáveis e equações, porém relativamente poucos graus de liberdade. Por consequência resta resolver um problema quadrático muito mais pequeno se SQP fosse aplicado no espaço nulo das equações de restrição. Com isso o número de variáveis do problema quadrático é igual ao número de graus de liberdade do problema. Os autores apontam que RND é idealmente apropriado para implementações nas quais o jacobiano é grande e esparso e comprovaram no seu trabalho que o algoritmo RND/SQP desenvolvido por eles, funciona tão bem quanto um algoritmo SQP pleno para problemas pequenos. A idéia principal do método de decomposição é reduzir o tamanho do problema da QP, eliminando as variáveis e restrições dependentes. Equações e variáveis independentes, que descrevam o próprio modelo de DAE, são eliminados nos passos de decomposição (BIEGLER, 1992). RND/SQP precisa do mesmo número de iterações que um método SQP pleno, porém com um problema quadrático mais pequeno o que significa um gasto de tempo e armazenagem menor. Uma descrição deste método encontra-se em BIEGLER (1988), (1992), (1994), VASANTHARAJAN & BIEGLER (1988) e GAL (1991). VASANTHARAJAN *et al.* (1990) ainda mostraram, como detalhe muito importante, a superioridade de RND/SQP em comparação com o pacote MINOS.

SCHMID & BIEGLER (1994) expandiram o algoritmo RND/SQP usando um algoritmo de QP chamado por eles de QPKWIK, e o pacote MA28 (biblioteca HARWELL), com subrotinas ligadas. O MA28 é um algoritmo que resolve sistemas de equações algébricas lineares com uma matriz Jacobiana esparsa. Os autores comprovaram no seu trabalho a superioridade do seu algoritmo, aplicado em problemas de alta dimensão, com um número de graus de liberdade pequeno e uma matriz Jacobiana, das restrições, esparsa.

CERVANTES & BIEGLER (1998) classificam os métodos de SQP, adequados para aplicação em problemas de alta dimensão, em dois grupos; "*Full-Space*" e

"*Reduced-Space*", apontando que ambos têm vantagens e desvantagens quando aplicados em certos tipos de problemas. Enquanto métodos de SQP "*Full-Space*" principalmente são muito adequados para problemas dinâmicos com elevado número de graus de liberdade, os métodos de SQP "*Reduced-Space*" são mais adequados para os problemas contendo um número de graus de liberdade pequeno.

Sendo o problema de controle ótimo no tempo de uma coluna de destilação um problema de alta dimensão, resultado do elevado número de equações do modelo da coluna, mas com um número de graus de liberdade pequeno, o algoritmo apresentado por SCHMID & BIEGLER (1994) parece ser um algoritmo muito adequado. Em consequência este algoritmo foi escolhido para a solução do problema de otimização neste trabalho. No capítulo 4.1. comprova-se a sua superioridade na aplicação do problema de minimizar o tempo de transição de uma coluna de destilação binária, comparando-o com um método SQP "*Full-Space*", o pacote E04UCF da NAG, baseado no algoritmo NPSOL.

2.2.3. Otimização Dinâmica, usando SQP

Para transformar um problema dinâmico num problema NLP, os métodos utilizados com mais frequência são os chamados métodos sequencial e simultâneo. A estratégia sequencial, também conhecida como "*Control Vector Parametrization*", aproxima apenas o controle como somatório de funções básicas. O sistema DAE do modelo é integrado usando algum pacote numérico de integração no processamento da otimização. Este método mantém o problema de otimização pequeno, mas a repetitiva integração do modelo pode resultar muito custosa. O método simultâneo aproxima os dois, estado e controle, discretizando as equações diferenciais e transformando-as em restrições algébricas. O modelo então é integrado simultaneamente enquanto o problema converge até o ótimo.

2.2.3.1. Método Seqüencial

A forma mais convencional, usada há varias décadas, é a forma seqüencial. A solução do sistema de equações diferenciais e a atualização da variável de decisão são processos separados. A idéia é discretizar o perfil de controle num polinômio por partes, em elementos finitos, chamado por isso também de "*Control Vector Parametrization*". As equações diferenciais do modelo, para determinado perfil de controle, podem ser resolvidas com qualquer método de integração padrão adequado e os seus resultados são transmitidos como valores da função e dos gradientes para o método de NLP. O método de otimização é aplicado então num "*loop*" externo para a atualização das ações de controle. A vantagem desta estratégia está na dimensão reduzida do problema de otimização. Esta estratégia corrente de otimização de sistemas dinâmicos precisa contudo de soluções repetidas do modelo formado por equações diferenciais e dos seus gradientes, o que pode resultar em alto custo para problemas complexos e de alta dimensão.

BIEGLER (1984) apresentou um trabalho sobre um problema dinâmico de controle ótimo de um reator em batelada usando um modelo matemático de pequena dimensão. O autor aproximou o perfil de controle por uma série de Taylor interrompida após o termo quadrático, usando o método Complex para calcular os coeficientes. Para a integração do modelo dinâmico ele usou o pacote DGEAR da biblioteca IMSL. No seu trabalho o autor testou o método seqüencial comparando-o com outros métodos, concluindo que o método simultâneo, um dos métodos apresentado no mesmo trabalho, é o mais eficiente.

Uma formulação para problemas sensitivos de controle ótimo, com um estado final especificado, foi descrita por ROSEN & LUUS (1989), usando o método seqüencial com o pacote NPSOL da biblioteca NAG como algoritmo de SQP. Eles discretizaram o termo de controle, dividindo o tempo em S intervalos iguais e transformando assim o problema de controle ótimo contínuo num problema de controle ótimo constante por partes. A solução ótima encontra-se com a solução de uma seqüência de problemas ligados com

parâmetros de continuidade. Desta forma os autores aproximaram o vetor de controle utilizando uma função básica enquanto o pacote DVERK foi usado para a integração das equações diferenciais. De uma forma semelhante também ROSEN & LUUS (1991) utilizaram uma formulação com um método de discretização seqüencial, resolvendo problemas de controle ótimo.

Mais recentemente o método seqüencial foi aplicado em um problema de controle ótimo de alta dimensão por FIKAR *et al.* (1998). Eles utilizaram como algoritmo SQP o pacote NLPQL de SCHITTKOWSKI (1985) e o pacote de integração DASSL para resolver as equações diferenciais do modelo. Os autores acharam o método seqüencial superior ao método simultâneo quando aplicado em problemas de alta dimensão, mesmo que o custo computacional, causado pelas repetitivas iterações do modelo, seja muito maior. A conclusão dos autores se baseia no fato de que, quando aplicado em problemas de alta dimensão, o aumento do número de variáveis de otimização causado pela discretização do modelo dinâmico, leva o método simultâneo a um comportamento menos robusto, o que pode provocar a não-convergência do método, em situações nas quais o método seqüencial ainda converge porque possui um número menor de variáveis de otimização.

2.2.3.2. Método Simultâneo

O método simultâneo inclui o modelo dinâmico diretamente no problema de otimização. Neste caso as equações diferenciais do modelo são discretizadas no tempo, formando equações residuais algébricas, por aplicação da colocação ortogonal. As equações discretizadas e os coeficientes do polinômio aproximador são incluídos como equações de restrição e variáveis de decisão, respectivamente, na formulação NLP. O NLP então é resolvido por aplicação de uma estratégia que leva simultaneamente à convergência e otimiza o modelo algébrico.

Um dos primeiros a descrever a aplicação de colocação ortogonal para reduzir problemas de otimização dinâmica num problema de NLP com equações de restrições foi BIEGLER (1984), usando polinômios de Lagrange como função aproximadora. A colocação ortogonal, uma variação do método de resíduos ponderados (FINLAYSON, 1972 e VILLADSEN & MICHELSEN, 1978), resolve as equações diferenciais no tempo transformando-as num conjunto de equações algébricas e as derivadas em somatórios dos valores da função nos pontos de colocação. Ao contrário dos métodos de integração baseados em Diferenças Finitas, a colocação ortogonal usa uma aproximação polinomial e precisa da satisfação da equação apenas nos pontos de colocação discretos, raízes do polinômio ortogonal (FINLAYSON, 1980).

CUTHRELL & BIEGLER (1987) expandiram o método usando colocação ortogonal em elementos finitos e apresentaram diferentes estratégias para a minimização de erros garantindo o aumento da precisão de aproximação na NLP. Isto foi conseguido com a solução adicional das condições suficientes, as quais são escritas como conjunto de equações de localização de pontos de nós que determinam a distribuição dos elementos incluídos no problema de otimização, escolhendo-se desta maneira os locais adequados para os pontos de nós. Além das variáveis do estado e do controle, agora a rotina de otimização determina também o tamanho dos elementos finitos.

O problema de controle ótimo com perfis de controle descontínuos foi tratado por CUTHRELL & BIEGLER (1987 e 1989). Para resolver problemas descontínuos eles incluíram os locais dos pontos de nós de elementos finitos adicionais, chamados de superelementos, nos quais podem-se encontrar as discontinuidades como variáveis de decisão, e apresentaram um método geral para resolver problemas de controle ótimo com perfil descontínuo.

Uma estratégia de minimização de erro apresentado por VASANTHARAJAN & BIEGLER (1990) é composta de dois conceitos diferentes. O primeiro é um critério de erro de aproximação incluído na formulação da NLP, e na verdade uma ampliação do método de equidistribuição de CUTHRELL & BIEGLER (1987), o qual foi aperfeiçoado

com o desenvolvimento de uma estratégia de otimização simultânea e equidistribuição de erro. O segundo conceito é uma estratégia muito mais fácil, na qual o erro de aproximação é controlado diretamente. O método chamado de "*direct error enforcement*", gera um número suficiente de elementos finitos e é incluído como estratégia de adição de elementos ao algoritmo de otimização para obter uma solução ótima e exata. Os métodos de VASANTHARAJAN & BIEGLER (1990) limitam-se a problemas de otimização de parâmetros.

Uma expansão da estratégia de minimização de erros para problemas com perfil de controle foi pesquisado por LOGSDON & BIEGLER (1989). Ao contrário dos problemas de otimização de parâmetros, os problemas de controle ótimo são formulados com DAEs. Portanto, podem aparecer problemas de estabilidade e de precisão em determinadas situações, como na saturação dos perfis de variáveis do estado, os quais não aparecem nos problemas de otimização de parâmetros. Para este tipo de problemas os autores introduziram critérios de erros adicionais e aplicaram uma discretização estável.

Uma boa descrição do método simultâneo para problemas dinâmicos, assim como uma comparação entre o método de variação e NLP, foi apresentada por BIEGLER (1992), que anotou que tal como o método seqüencial, este método não garante estabilidade para alguns casos de BVP ou casos de DAEs com componentes não estáveis (ASCHER *et al.*, 1988). O método de colocação ortogonal pode-se subdividir nas seguintes tarefas (VILLADSEN & MICHELSEN, 1978):

- 1) Escolha do polinômio ortogonal adequado, por exemplo o polinômio de Legendre ou outros como os polinômios de Tschebyshev, Laguerre ou Hermite;
- 2) Determinação dos pontos de colocação;
- 2) Determinação dos coeficientes da matriz de colocação;

4) Cálculo da solução aproximada.

Sendo τ^i os pontos de colocação no tempo normalizado ($\tau=0,\dots,1$), as variáveis de estado são aproximados por um polinômio interpolador de Lagrange da ordem $(K+1)$:

$$\mathbf{x}^i = \mathbf{x}(\tau^i) \approx \sum_{j=0}^{K+1} \mathbf{x}^j \phi^j(\tau^i), \quad (2.2-5a)$$

com:

$$\phi^j(\tau^i) = \prod_{\substack{k=0 \\ k \neq j}}^{K+1} \frac{(\tau^i - \tau^k)}{(\tau^j - \tau^k)}. \quad (2.2-5b)$$

Do mesmo modo, para as variáveis de controle tem-se:

$$\mathbf{u}^i = \mathbf{u}(\tau^i) \approx \sum_{j=0}^{K+1} \mathbf{u}^j \psi^j(\tau^i), \quad (2.2-6a)$$

com:

$$\psi^j(\tau^i) = \prod_{\substack{k=0 \\ k \neq j}}^{K+1} \frac{(\tau^i - \tau^k)}{(\tau^j - \tau^k)}. \quad (2.2-6b)$$

Para problemas da engenharia química os estados e controles são conjuntos fisicamente definidos, como por exemplo temperaturas ou concentrações. Por consequência, o uso do polinômio de interpolação de Lagrange gera coeficientes \mathbf{x}^j e \mathbf{u}^j , o que é útil caso restrições de variáveis sejam fixadas, perfis (do estado ou do controle) inicializados ou perfis de solução interpretados. Para outros tipos de polinômios (por exemplo "*Power Series*" ou " *β -spline*") os coeficientes não têm estas características (CUTHRELL & BIEGLER, 1987). A primeira derivada de \mathbf{x} é aproximada por:

$$\frac{d\mathbf{x}^i}{d\tau} \approx \sum_{j=0}^{K+1} \mathbf{x}^j \dot{\phi}^j(\tau^i), \quad (2.2-7a)$$

onde:

$$\dot{\phi} = \frac{d\phi}{d\tau}. \quad (2.2-7b)$$

Uma vez os pontos τ^i ($i = 0, \dots, K+1$) sejam escolhidos, esta formulação permite resolver problemas de otimização dinâmica gerais. Para escolher os locais dos pontos τ^i , como ilustrado na Figura 2.2-1, são utilizados os zeros de um polinômio ortogonal contínuo:

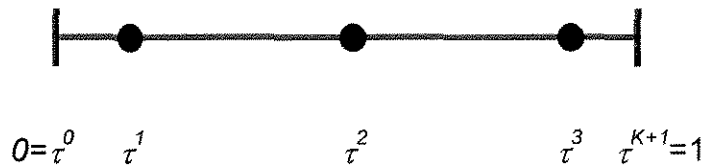


Figura 2.2-1: Colocação Ortogonal; com $K = 3$ Pontos de Colocação internos.

O método da colocação ortogonal global constrói as aproximações do estado e do controle, utilizando polinômios de ordem $(K+1)$. Para funções com comportamento pobre ("*poorly behaved*"), causado por alterações rápidas em domínios pequenos por exemplo, uma aproximação exata precisaria de K muito grande. Por causa disso CUTHRELL & BIEGLER (1987) propuseram o uso de colocação ortogonal em elementos finitos, utilizando aproximações de polinômios por partes, como mostrado em Figura 2.2-2 para 3 elementos finitos. A vantagem deste método está na possibilidade de assentar os pontos de colocação nestas regiões de rápidas alterações, obtendo assim uma melhor aproximação.

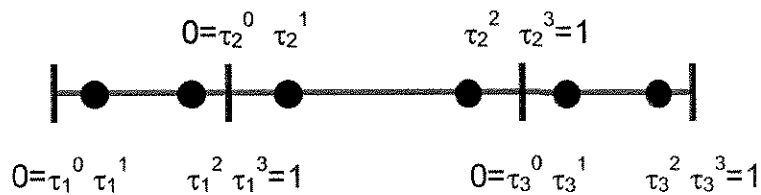


Figura 2.2-2: Colocação Ortogonal em Elementos Finitos; com $K/K/K=2/2/2$ e $NE=3$

2.2.3.3. Vantagens e Desvantagens

A principal desvantagem do método seqüencial é seu alto custo computacional como mostram os resultados obtidos por FIKAR *et al.* (1998) enquanto a principal desvantagem da formulação simultânea é o excessivo aumento do tamanho do problema de otimização e geralmente o alto grau de não-linearidade, o que resulta em que o método pode ser muito sensível em pontos iniciais e precisa de uma inicialização suficientemente cuidadosa. Especialmente quando o modelo dinâmico é de alta dimensão o método seqüencial geralmente é mais robusto do que o método simultâneo. Isso resulta da característica dos dois métodos. O método simultâneo resolve as equações do modelo dinâmico discretizado e converge até um ótimo no mesmo tempo, resultando numa dimensão muito alta para o problema de otimização. No método seqüencial o sistema de equações é resolvido “off-line” resultando assim em um problema pequeno para o método de otimização.

A maior vantagem do método simultâneo sobre o seqüencial é a eliminação de soluções intermediárias de alto custo computacional ou talvez inexecutável. BIEGLER (1984, 1988, 1992); CUTHRELL & BIEGLER (1987, 1989); VASANTHARAJAN & BIEGLER (1990) e TANARTKIT & BIEGLER (1995) propuseram, para resolver

problemas de otimização com DAE, usar o método simultâneo e aplicar SQP como método adequado para resolver problemas de otimização dinâmica.

2.3. Programação Dinâmica

A Programação Dinâmica, método que resolve problemas de otimização através de um processo de decisão ótima em várias etapas (HOFER & LUNDERSTÄDT, 1975), é aplicada há mais de 40 anos numa variedade de problemas. Apesar do entusiasmo inicial com o novo método, já no início da sua aplicação foram descritos as restrições desta técnica numérica para problemas de controle ótimo (LUUS, 1990a). Até o início dos anos 90, poucos avanços foram feitos na área da aplicação de Programação Dinâmica em problemas de controle ótimo não-linear de dimensão superior (LUUS, 1990a).

O método de Programação Dinâmica, chamado por FÖLLINGER (1988) de Otimização Dinâmica, nome que consideramos pouco adequado levando a equívocos, foi desenvolvido por BELLMAN (1957) e é um método altamente flexível e muito adequado para sistemas de otimização dinâmica, ainda que possa também ser aplicado em outros tipos de otimização. Ao contrário do Princípio de Máximo de Pontryagin, ele pode ser facilmente aplicado em problemas com limitações das variáveis do estado. A principal desvantagem deste método é o aumento explosivo da necessidade de capacidade de armazenagem e tempo de computação, no caso do incremento da ordem do sistema.

Ao contrário de outros métodos de otimização, geralmente não é possível aplicar um pacote de "software" com um algoritmo de DP pronto. Cada problema precisa ser formulado individualmente, adaptando-o ao algoritmo adequado de DP. Por este motivo é preciso entrar mais nos detalhes do princípio do método de Programação Dinâmica.

O funcionamento da DP é ilustrado na Figura 2.3-1, como apresentado por FÖLLINGER (1988), e mostra como exemplo, uma rede de caminhos. As estações, chamadas de pontos de grade, são fixadas pelos valores da ordenada x e da abcissa k , sendo k o índice dos instantes de tempo t^k ($k=0, \dots, S$) que limitam as etapas discretas no tempo. Os números escritos nos caminhos representam as unidades do tempo necessário para a respectiva parte do mesmo. Para encontrar o caminho ótimo seria uma consequência lógica calcular todos os caminhos possíveis que levam do ponto inicial t^0 ao ponto final t^{fin} e determinar os tempos necessários. Neste caso existem exatamente $2^3 = 8$ caminhos. É fácil notar que o aumento do número de passos e de variáveis provocaria o crescimento explosivo do número de possíveis caminhos, o que faria o método inexecutável.

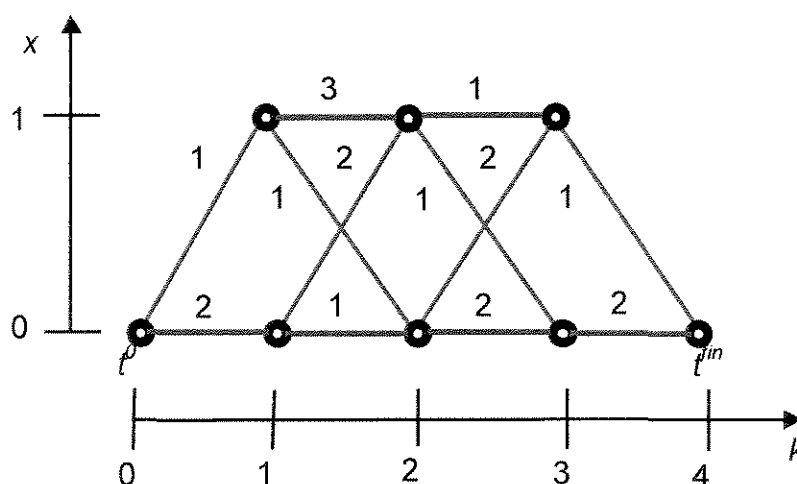


Figura 2.3-1: Programação Dinâmica

Em vez de comparar todos os caminhos possíveis, aplica-se um processo de decisão por etapas que reduz radicalmente o custo de computação. O princípio deste método consiste em decidir o caminho mais rápido de cada ponto (k,x) até o ponto final, começando em $k=S-1$ (no último ponto $k=S$ não é mais preciso tomar uma decisão) e passando pela rede de caminhos respectivamente, utilizando sempre o resultado da

etapa anteriormente já tratado. No exemplo da Figura 2.3-1 a estratégia de controle é representada por:

$$u = \begin{cases} -1 & (\text{"encima, direita"}) \\ 0 & (\text{"direita"}) \\ 1 & (\text{"abaixo, direita"}) \end{cases} \quad (2.3-1)$$

Retrocedendo desta forma até o ponto t^0 , consegue-se a estratégia de controle ótima, mostrada pela linha vermelha na Figura 2.3-2:

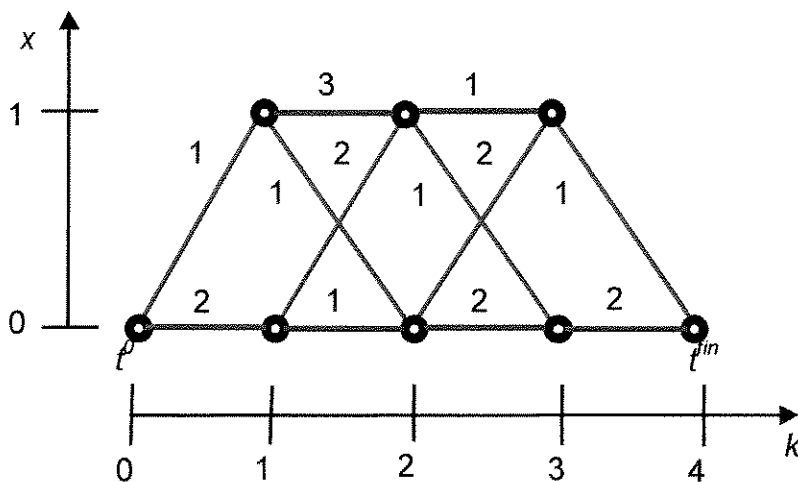


Figura 2.3-2: Estratégia de Controle Ótima

No entanto, a relação entre o ponto x^k e um ponto seguinte x^{k+1} é geralmente descrita por uma função que pode mudar passo a passo (representado pelo índice k de g), da forma:

$$x^{k+1} = g^k(x^k, u^k) \quad (k = 0, 1, \dots, S-1) \quad (2.3-2)$$

2.3.1. Discretização no Tempo

Seja feita aqui uma breve anotação sobre os problemas contínuos no tempo. No caso em que o Índice de Desempenho de um problema contínuo no tempo, por qualquer razão, não pode ser aproximado de forma discreta, o princípio de otimalidade também pode ser aplicado diretamente. Isso levaria a equações diferenciais parciais, com o pressuposto que o Índice de Desempenho a minimizar seja diferenciável continuamente. PAPAGEORGIOU (1991) descreveu as condições de otimalidade necessárias e suficientes para um ótimo global chamando-as, em relação aos problemas reais da física, equações de Hamilton-Jacobi-Bellman, enquanto FÖLLINGER (1988) as chamou apenas como equações da função de Bellman.

Para discretizar a equação diferencial do estado:

$$\frac{dx}{dt} = F(x, u, t), \quad (2.3-3a)$$

com

$$0 \leq t \leq t^{fin}, \quad (2.3-3b)$$

nos instantes de tempo $0 = t^0 < t^1 < \dots < t^S = t^{fin}$, FÖLLINGER (1988) substituiu $x(t)$ por $x(t^k) = x^k$ e $u(t)$ por $u(t^k) = u^k$ ($k = 0, 1, \dots$) e, em vez da derivada, usou o quociente de diferenças:

$$\frac{x^{k+1} - x^k}{t^{k+1} - t^k} = F(x^k, u^k, t^k), \quad (k = 0, 1, \dots, S-1), \quad (2.3-4)$$

ou seja:

$$x^{k+1} = x^k + \Delta t^k F(x^k, u^k, t^k), \quad (k = 0, 1, \dots, S-1). \quad (2.3-5)$$

Se as diferenças $\Delta t^k = t^{k+1} - t^k$ forem suficientemente pequenas, (2.3-4) ou (2.3-5) aproximam a equação diferencial (2.3-3a). Se o lado direito da eq. (2.3-3a) não depende explicitamente de t , além de t^k ser eqüidistante, então eq. (2.3-5) pode-se escrever da forma:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta t \mathbf{F}(\mathbf{x}^k, \mathbf{u}^k). \quad (2.3-6)$$

Por conseqüência, o lado direito é uma função de \mathbf{x}^k e \mathbf{u}^k , idêntica em todos os instantes k , e portanto da forma:

$$\mathbf{x}^{k+1} = \mathbf{g}(\mathbf{x}^k, \mathbf{u}^k), \quad (k = 0, 1, \dots, S-1). \quad (2.3-7)$$

Este caso especial de (2.3-2) apresenta-se em muitas aplicações.

Para a estratégia $\mathbf{u}^k, \mathbf{u}^{k+1}, \dots, \mathbf{u}^{S-1}$, onde k pode tomar qualquer valor, $\mathbf{u}^{k+1}, \dots, \mathbf{u}^{S-1}$ representa uma estratégia restante no sentido da formulação do Princípio de Otimalidade de Bellman, da qual se deduzem as condições necessárias, dizendo:

A trajetória plena somente pode ser ótima se cada trajetória restante, em qualquer estado intermediário, o for.

com:

$$J^{k*}(\mathbf{x}^k) = \left(\min_{\mathbf{u}^k, \dots, \mathbf{u}^{S-1}} \right) J^k(\mathbf{x}^k; \mathbf{u}^k, \dots, \mathbf{u}^{S-1}), \quad (2.3-8)$$

onde $J^k(\mathbf{x}^k; \mathbf{u}^k, \dots, \mathbf{u}^{S-1})$ é o Índice de Desempenho de uma estratégia $(\mathbf{u}^k, \dots, \mathbf{u}^{S-1})$ que leva de \mathbf{x}^k até o estado final \mathbf{x}^S e J^{k*} o seu valor ótimo.

Pelo princípio de otimalidade, o Índice de Desempenho pode-se escrever então da forma:

$$J^{k*}(\mathbf{x}^k) = \min_{\mathbf{u}^k} \left[l^k(\mathbf{x}^k, \mathbf{u}^k) + J^{k+1*}(\mathbf{x}^{k+1}) \right], \quad (2.3-9)$$

onde $l^k(\mathbf{x}^k, \mathbf{u}^k)$ representa o Índice de Desempenho parcial do instante k até $k+1$ e $J^{k+1*}(\mathbf{x}^{k+1})$ o valor mínimo do Índice de Desempenho restante do instante $k+1$ até S . Introduzindo a equação do estado (2.3-2) em (2.3-9), obtém-se a fórmula recursiva de Bellman na forma descrita por FÖLLINGER (1988):

$$J^{k*}(\mathbf{x}^k) = \min_{\mathbf{u}^k} \left[l^k(\mathbf{x}^k, \mathbf{u}^k) + J^{k+1*} \{ \mathbf{g}^k(\mathbf{x}^k, \mathbf{u}^k) \} \right], \quad (k = S-1, \dots, 1, 0). \quad (2.3-10)$$

Para lograr recursivamente a estratégia ótima começa-se agora com $k=S-1$. Começando no último passo, tem-se $J^S(\mathbf{x}^S)=0$ e eq. (2.3-9) pode ser escrita da forma:

$$J^{S-1*}(\mathbf{x}^{S-1}) = \min_{\mathbf{u}^{S-1}} l^{S-1}(\mathbf{x}^{S-1}, \mathbf{u}^{S-1}). \quad (2.3-11)$$

A solução é armazenada e usada no próximo passo ($k=S-2$). Desta maneira e, retrotraindo-nos até chegar a resolver $J^0(\mathbf{x}^0)$, consegue-se a solução do problema de otimização completo.

2.3.2. Discretização dos Domínios

Os domínios admissíveis do estado $X(k) = \mathbf{X}^k = (\mathbf{x}_{\min}, \dots, \mathbf{x}_{\max})$ e do controle $U(k) = \mathbf{U}^k = (\mathbf{u}_{\min}, \dots, \mathbf{u}_{\max})$ são discretizados, pelo uso das respectivas grades de pontos, o que é mostrado na Figura 2.3-3:

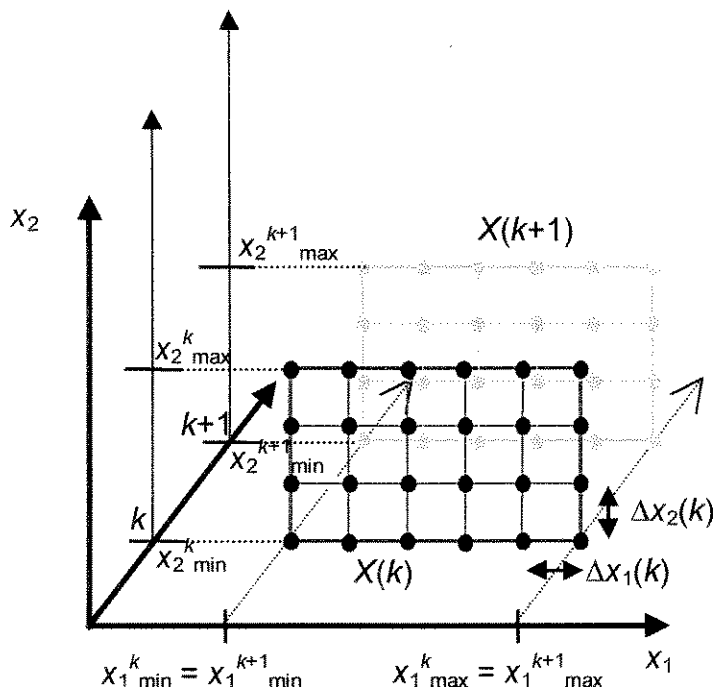


Figura 2.3-3: Discretização do Domínio Admissível do Estado

PAPAGEORGIOU (1991) propôs escolher os intervalos de discretização Δx^k e Δu^k em relação ao tipo de problema e a precisão desejada da solução. No caso em que os domínios admissíveis de estado e controle fossem ilimitados, o autor propôs limitá-los numa forma adequada para conseguir um número finito de pontos de grade. Aplicando todos os valores discretos do controle $u^k \in U^k$ num estado discreto x^k , consegue-se agora um número finito de transições para a próxima etapa ($k+1$). Repetindo este processo para os pontos discretos do estado de todas as seções do tempo, é gerado um processo de decisões discretas multietapas completo.

Durante o processo pode acontecer que o estado x^{k+1} fique fora do domínio do estado admissível X^{k+1} o que significaria que esta transição não precisa mais ser considerada. Outra dificuldade apareceria se o estado x^{k+1} não acertasse num dos pontos do retículo do grau $k+1$, como mostra a Figura 2.3-4. Neste caso, e se resultados mais exatos fossem necessário, PAPAGEORGIOU (1991) recomenda o método de interpolação linear. Para evitar a interpolação e obter a estratégia de controle ótimo

LUUS (1990a, 1990b) usou apenas o ponto de grade do estado mais próximo, como recomendado por DeTREMBLAY & LUUS (1989).

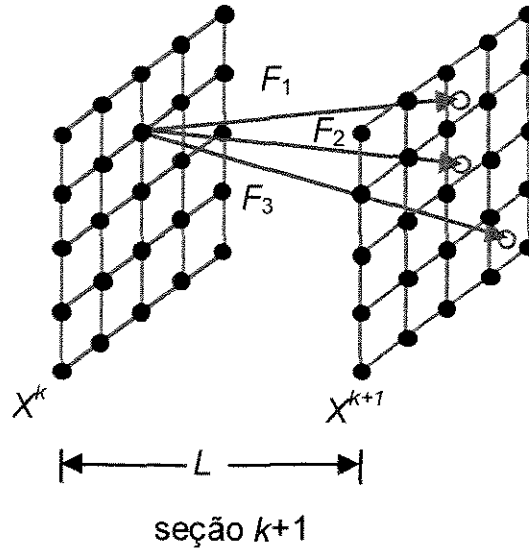


Figura 2.3-4: Determinação do Ponto de Continuação

2.3.3. Seções de Tempo variável

Para problemas de controle ótimo, BOJKOV & LUUS (1994b) utilizaram seções variáveis de tempo. Em vez de dividir o intervalo de tempo $(0, \dots, t^{fin})$ em S etapas de largura igual, eles usaram larguras variáveis, procurando simultaneamente os valores de controle, constantes em cada etapa do tempo, e os valores das larguras variáveis $\ell(k)$, sendo:

$$\ell(k) = t^k - t^{k-1}, \quad (k = 1, 2, \dots, S). \quad (2.3-12)$$

Neste sentido foi transformado o problema de controle contínuo no tempo num problema constante por partes, dividindo o intervalo do tempo $0 \leq t \leq t^{fin}$ em S seções de

tempo de largura variável. Os autores introduziram a variável normalizada do tempo τ da forma:

$$\tau = \frac{t}{t^{\bar{m}}}, \quad (2.3-13)$$

com os valores discretizados:

$$\tau^k = \frac{k}{S}, \quad (k = 0, 1, \dots, S). \quad (2.3-14)$$

Desta maneira, todas as etapas possuem igual largura dentro do domínio do tempo transformado:

$$L = \frac{1}{S}. \quad (2.3-15)$$

As equações diferenciais do sistema transformado, no intervalo do tempo $\tau^k \leq \tau \leq \tau^{k+1}$, podem-se escrever agora da forma:

$$\frac{dx}{d\tau} = \ell(k) S F(x, u), \quad (k = 1, 2, \dots, S). \quad (2.3-16)$$

Sendo constante o controle $u(\tau^k)$, dentro de cada intervalo do tempo, o problema de controle ótimo no tempo consiste agora na determinação de $u(\tau^k)$ e de $\ell(k)$ com $(k = 1, 2, \dots, S)$, satisfazendo as restrições e minimizando $t^{\bar{m}}$.

2.3.4. Programação Dinâmica Iterativa

Por causa do custo de computação, crescendo explosivamente com o aumento de variáveis e seções de tempo (etapas de discretização), a aplicação da Programação Dinâmica se limita a pequenos problemas. PAPAGEORGIOU (1991) anotou que, para poder usar este método na solução de problemas maiores, são aplicadas recentes avanços da computação para reduzir as restrições de dimensão na aplicação de Programação Dinâmica. Realmente a estrutura simples do processo multietapa da Programação Dinâmica oferece a oportunidade de uma paralelização da computação, através de línguas de programação paralelas como ADA.

Outros métodos modificados tem sido propostos na literatura, obtendo reduções consideráveis de computação. Para se evitar a necessidade da grade muito fina, LUUS (1990a, 1990b) usou um método iterativo, o qual começa com uma grade relativamente grossa cujo centro corresponde ao valor ótimo da iteração anterior. Depois de cada iteração o tamanho da grade é reduzido através de um fator, usando a estratégia ótima como centro da grade na próxima iteração. Assim, após um número suficiente de iterações a grade resulta suficientemente fina. Este procedimento iterativo é chamado de Programação Dinâmica Iterativa (LUUS & ROSEN, 1991; BOJKOV & LUUS, 1992; 1994a e 1994b).

LUUS (1990a) mostrou que aplicando IDP com pontos de grade x distribuídos uniformemente e usando uma grade grossa com apenas cinco valores para as variáveis do estado, já pode levar a resultados bem exatos. Segundo o autor, podem aparecer dificuldades na computação se o número de variáveis do estado for superior à 3, pelo aumento explosivo de 5^3 . Para reduzir este problema ele propôs utilizar uma grade que contem apenas pontos admissíveis. Caso as restrições $x_{min} \leq x \leq x_{max}$ fossem violadas, LUUS (1990a) propôs aplicar a técnica de "*clipping*" (2.3-17), a qual substitui o valor calculado, mas fora do domínio, pelo respectivo valor extremo.

$$x = \begin{cases} x_{min} & \text{se } x < x_{min} \\ x & \text{se } x_{min} < x < x_{max} \\ x_{max} & \text{se } x > x_{max} \end{cases} \quad (2.3-17)$$

LUUS (1990b) recomendou reduzir os domínios admissíveis X e U após cada iteração, enquanto LUUS (1990a) e outros autores apenas reduziram o domínio do controle ou, como BOJKOV & LUUS (1994b), as seções do tempo, utilizando a trajetória ótima do passo anterior do processo e o respectivo controle ótimo ou as respectivas larguras das seções, como centros das grades para a iteração seguinte, resultando assim em uma solução cada vez mais afinada.

BOJKOV & LUUS (1992, 1994a) utilizaram, ao em vez da escolha uniforme, um método de escolha casual dos valores de controle admissíveis, o qual tem como vantagem possuir um número pequeno de trajetórias a calcular e comparar, pelo qual o método é adequado para problemas com elevado número de variáveis de controle. Num exemplo de 20 variáveis de estado e 20 variáveis de controle, os autores usaram IDP, aplicando o método de escolha casual e, usando apenas 100 combinações de variáveis de controle escolhidas casualmente em cada ponto de grade, conseguiram uma rápida convergência. Para determinar as variáveis casuais eles utilizaram um gerador ("*built-in random number generator*" RANF de *Cray Research*).

Uma modificação do método IDP, chamada de *Semiexhaustive-Search*, foi apresentada por GUPTA (1995). O novo método utiliza, tal como a IDP, uma redução da grade à volta do domínio da estratégia do controle ótimo da iteração anterior. No entanto o método de busca da estratégia de controle é diferente e não são necessárias as grades das variáveis do estado. Mas a principal diferença está em que o método apresentado pelo autor começa em $t = 0$ e continua até $t = t^n$. Para evitar a enorme acumulação de valores armazenados e do custo de computação ao calculá-los novamente, são armazenados apenas um número determinado dos melhores resultados de cada passo. Num exemplo o autor usou 20 como número máximo de trajetórias, com 5 pontos da grade de controle. Isso significa, começando em x_0 , 5

trajetórias calculadas na primeira seção, 25 na segunda e 100 trajetórias (20 x 5) em cada seção seguinte. O método não vai ser mais aprofundado aqui, porque não apresenta vantagens sobre o IDP na aplicação em problemas de controle ótimo no tempo, mas sim desvantagens.

A principal desvantagem da Programação Dinâmica é a sua limitada aplicabilidade em problemas de alta dimensão. LUUS & SMITH (1991) aplicaram IDP num problema de 6 variáveis de estado e 2 variáveis de controle. Também LUUS (1990a, 1990b) descreveram a aplicação de um método iterativo que utiliza Programação Dinâmica para problemas não-lineares de controle ótimo e de alta dimensão.

Em cada seção a minimização se efetua através da comparação direta de um número finito de transições, o que garante um minimizador global. Por consequência, o processo multietapa leva a um minimizador global do problema completo, porque indiretamente são examinadas todas as combinações de transições possíveis. Evidentemente, trata-se do minimizador global do problema discretizado o que representa uma aproximação da solução do problema anteriormente discreto no tempo (ou discretizado no tempo). Com isso a Programação Dinâmica é superior à aplicação do Princípio de Máximo de Pontryagin, o qual não garante um ótimo global para sistemas altamente não-lineares ou problemas que levam a controles singulares (LUUS, 1990b).

Ainda que seja mais lento que a maioria dos algoritmos baseados em gradientes, DADEBO & McAULEY (1995a) e GUPTA (1995) recomendaram a IDP para o "*cross-checking*" de resultados encontrado por outros métodos, por causa da grande possibilidade de acertar um ótimo global. Porém, também a Programação Dinâmica não é uma garantia absoluta para encontrar um ótimo global, sobre tudo para problemas não limitados (LUUS, 1990b). Em casos duvidosos, e se uma limitação adequada não fosse possível, propõe-se utilizar uma grade suficientemente grande. Uma descrição

detalhada do algoritmo IDP, desenvolvido para este trabalho de controle ótimo no tempo de uma coluna de destilação, encontra-se em capítulo 3.5.

2.4. Modelos Simplificados e Modelos de Ordem Reduzida

A otimização, baseada em equações, requer modelos de processos que descrevam adequadamente o respectivo comportamento físico-químico. A solução de problemas de controle ótimo no tempo de colunas de destilação, descritas por modelos dinâmicos, leva a modelos matemáticos complexos e de alta dimensão. Segundo STØREN & HERTZBERG (1994), o cálculo das propriedades físico-químicas e termodinâmicas na simulação dinâmica de colunas de destilação consome entre 30 e 90% do tempo de CPU necessário, o que mostra o interesse que existe em acelerar este cálculo. A necessidade de modelos dinâmicos de processos mais eficientes, diminuindo os custos computacionais, foi a motivação para o desenvolvimento de técnicas para a simplificação e para redução do tamanho destes modelos.

De uma maneira geral, para a descrição completa do comportamento transiente de uma coluna de destilação são necessárias $N_T(C-1)+2N_T$ equações diferenciais, correspondentes aos balanços de energia e de massa (total e por componente) em cada prato, além de equações algébricas descrevendo as relações estequiométricas entre os componentes e as relações de equilíbrio termodinâmico. Uma das técnicas mais interessantes para a simplificação de modelos dinâmicos de processos químicos é o recurso a modelos termodinâmicos locais. A idéia básica por detrás deste conceito é a de modelar o comportamento termodinâmico complexo através de expressões funcionais simples (os modelos locais) válidas para regiões limitadas de composição, temperatura e pressão (CHIMOWITZ *et al.*, 1985).

Uma outra técnica para a construção de modelos simplificados de colunas de destilação por pratos consiste em desprezar as equações correspondentes à variação de determinadas variáveis de estado. Segundo CHIMOWITZ *et al.* (1985), os modelos

mais simples consideram apenas os balanços de massa por componente na fase líquida. Um modelo muito popular de colunas de destilação considera o "overflow" molar constante, o que só é válido se as mudanças de "holdup" e de energia são rápidas, e os fluxos de líquido e vapor são fixos no tempo.

Neste trabalho é usado basicamente um modelo simplificado de colunas de destilação apresentado por SKOGESTAD & MORARI (1988), expandido considerando, além do balanço de massa por componente, o balanço de massa total em cada prato, durante o transiente. Isso foi considerado necessário porque o desvio do resultado ótimo do tempo de transição, considerando o "holdup" constante, é grande demais, chegando a quase 10 % do valor final, como se mostra ao longo deste trabalho.

SEFERLIS & HRYMAK (1994a) apontam que modelos simplificados precisam de certas propriedades para serem adequados para o uso em otimização em tempo real. Em especial, modelos simplificados precisam conter todas as informações dos gradientes, necessárias para a otimização, para garantir convergência até ao ótimo definido pelo modelo rigoroso. A insuficiente satisfação destas condições pode levar a convergir até uma solução ótima diferente ou simplesmente a não convergir (BIEGLER *et al.* 1985).

Além das técnicas de simplificação dos modelos de colunas descritas resumidamente acima, foram desenvolvidas outras técnicas para diminuir a ordem do modelo do processo, definida como o número de variáveis independentes. Para reduzir a ordem de modelos de colunas de destilação prato-a-prato, descritos por equações às diferenças, vários trabalhos foram publicados aplicando colocação ortogonal. O método descrito por CHO & JOSEPH (1983a,b) transforma o modelo prato-a-prato, formado por equações às diferenças finitas, num sistema de equações diferenciais, tal como proposto por OSBORNE (1971), e resolve este sistema aplicando colocação ortogonal, usando polinômios de Jacobi.

STEWART *et al.* (1985) propuseram a solução das equações às diferenças prato-a-prato usando um método de colocação ortogonal onde os pontos de colocação são os zeros de polinômios ortogonais discretos de Hahn. O uso de polinômios ortogonais discretos de Hahn tem a vantagem, entre outras, de que, no caso em que o número de pontos de colocação é exatamente igual ao número de pratos, as posições dos pontos de colocação são idênticas às dos pratos e desta forma representam melhor as características de uma coluna de destilação prato-a-prato do que um polinômio contínuo como o de Jacobi. Os resultados obtidos pelos autores, comparando o seu método com outros métodos de redução usando polinômios contínuos, mostram a superioridade de um polinômio discreto. Discontinuidades nos perfis de composição e de temperatura causadas pela alimentação e correntes laterais são tratadas pelo recurso à colocação ortogonal discreta por elementos finitos (SEFERLIS & HRYMAK, 1994a).

O método de redução da ordem do modelo prato-a-prato de colunas de destilação usado neste trabalho constitui uma sistematização do método desenvolvido por STEWART *et al.* (1985) e por SEFERLIS & HRYMAK (1994a). Com efeito, considere-se, por exemplo, a equação do balanço de massa por componente em regime estacionário para o prato m :

$$L (x_{m+1} - x_m) - V (y_m - y_{m-1}) = 0. \quad (2.4-1)$$

A variável de estado x_m pode ser considerada como sendo uma função da variável discreta m , o índice do prato. Deste modo, a equação (2.4-1) pode ser escrita na forma simbólica, usando os operadores de diferença para a frente ("*first forward difference*"),

$$\Delta x_m = x_{m+1} - x_m, \quad (2.4-2)$$

e de diferença para trás ("*first backward difference*"),

$$\nabla y_m = y_m - y_{m-1}, \quad (2.4-3)$$

obtendo-se:

$$L \Delta x_m - V \nabla y_m = 0, \quad (2.4-4a)$$

ou:

$$L \Delta x(s) - V \nabla y(s) = 0, \quad (2.4-4b)$$

onde s é uma variável contínua que toma os valores $s=m$ para as posições dos pratos.

À semelhança do que acontece com equações diferenciais, a equação às diferenças (2.4-4b) pode agora ser resolvida pelo método da colocação ortogonal discreta. As funções $x(s)$ e $y(s)$ são aproximadas por um polinômio interpolador de Lagrange:

$$x(s) \approx \sum_{j=0}^{N+1} \Gamma_j(s) x(s_j). \quad (2.4-5)$$

À semelhança das relações (2.2-7a), usadas para aproximar a derivada, tem-se aqui:

$$\Delta x(s) \approx \sum_{j=0}^{N+1} \Delta \Gamma_j(s) x(s_j). \quad (2.4-6)$$

Da mesma forma tem-se:

$$\nabla y(s) \approx \sum_{j=0}^{N+1} \nabla \Gamma_j(s) y(s_j). \quad (2.4-7)$$

Como mostrou CARVALHO (1996), o número total de pontos de colocação, aqui representado por $N+2$, sendo N o número de pontos de colocação internos, depende da estratégia de colocação usada. O mesmo acontece com os valores s_j ($j=0,1,\dots,N+1$), que, no presente caso, são zeros de polinômios de Hahn trasladados.

As equações de balanço por componente na forma (2.4-4) só são válidas para L e V constantes (modelo com "overflow" molar constante, em regime estacionário). Contudo, a introdução das diferenças e a sua aproximação por relações do tipo (2.4-6 e 7) são válidos para modelos prato-a-prato mais gerais (CARVALHO, 1996).

Modelos de ordem reduzida de colunas de destilação de pratos, obtidos pelo método da colocação ortogonal discreta, foram já usados por diversos autores em problemas de simulação e otimização. SWARTZ & STEWART (1986) usaram o modelo de ordem reduzida desenvolvido por STEWART *et al.* (1985) em problemas de projeto ótimo de colunas de pratos. Para isso, o número de pratos na coluna (que aparece como um dos parâmetros dos polinômios de Hahn) foi tratado como sendo uma variável contínua. SEFERLIS & HRYMAK (1994a) usaram modelos de ordem reduzida de colunas de destilação de pratos baseados na estratégia proposta por STEWART *et al.* (1985), mas generalizados pelo recurso à colocação por elementos finitos, para resolver problemas de otimização do estado estacionário. Os resultados da otimização, obtidos usando o pacote MINOS.5, mostraram que os modelos de ordem reduzida usados são mais robustos do que os modelos completos prato-a-prato, conduzindo a convergência mesmo para estimativas iniciais de má qualidade. Os autores mostraram ainda que a solução ótima obtida usando os modelos de ordem reduzida é praticamente idêntica à do modelo completo prato-a-prato, embora o custo computacional seja muito menor. Finalmente, SEFERLIS & HRYMAK (1994b) generalizaram a técnica de colocação adaptativa em elementos finitos para a colocação ortogonal discreta, e aplicaram a técnica em problemas de otimização do projeto do estado estacionário de colunas de destilação.

2.5. Controle Ótimo no Tempo de Colunas de Destilação

Os problemas de controle ótimo no tempo de colunas de destilação podem ser classificados em três categorias: colunas de batelada; o processo de arranque de uma coluna, também chamado de processo "start-up", e o processo de troca de produto ou

especificação, levando a coluna de um estado estacionário inicial para outro estado estacionário com novas especificações da operação. O processo em batelada pode ser mais complexo, incluindo geralmente ainda o processo de arranque. Enquanto que até a década 90 foram publicados poucos artigos sobre controle ótimo de colunas de destilação, considerando uma ou mais das categorias mencionadas, nos últimos anos o número de publicações tratando do problema de controle ótimo de colunas de destilação vem aumentando.

LOGSDON & BIEGLER (1993) descreveram a otimização de uma coluna de destilação em batelada. Eles utilizaram um algoritmo SQP para resolver o problema de controle ótimo de uma coluna de destilação binária de 12 pratos resultando num custo computacional relativamente alto (mais de 1000 s em todos os exemplos, usando uma Estação de Trabalho). O elevado tempo de computação foi causado pelo grande número de variáveis geradas pela formulação simultânea.

Com métodos fenomenológicos, KRUSE *et al.* (1995) otimizaram o processo de arranque de uma coluna de destilação. Os autores propuseram o uso das concentrações dos produtos de fundo e do destilado como variáveis à controlar e a vazão do destilado e a potência do refeedor como variáveis de controle.

LI & WOZNY (1996) indicam no seu trabalho a solução de um problema de controle ótimo no tempo de uma coluna de destilação em batelada, considerando também o problema da fase de arranque do processo. Utilizando a vazão de destilado, D , como variável de controle, eles aplicaram um algoritmo SQP na formulação simultânea, com o modelo prato-a-prato como restrições discretizadas do problema de otimização. Os autores compararam a aplicação do algoritmo de SQP e do método de Programação Dinâmica Diferencial, concluindo que o SQP, com a formulação simultânea do problema, converge mais rápido. Mesmo assim, para o problema de controle ótimo de uma coluna de destilação em batelada com 40 pratos, o custo de computação pode-se considerar relativamente alto (mais de 1000 s em todos os exemplos, usando uma Estação de Trabalho). Os autores apontaram o problema da

pouca robustez do método SQP simultâneo, a qual vai piorando com o aumento do número de variáveis causado pelo incremento do número de pontos de colocação no tempo.

FLENDER *et al.* (1997) utilizaram SQP para resolver um problema de controle ótimo no tempo de um processo de troca de produto de uma coluna de destilação. Levando a coluna de uma concentração alta do produto de destilado para uma concentração mais baixa, os autores compararam o tempo ótimo, obtido com a aplicação da estratégia ótima, com o tempo necessário para o processo convencional de troca de produto. Os autores consideraram o "*holdup*" constante e usaram como variáveis de controle a razão de refluxo L/D e o calor do refeedor.

Demonstrando a eficiência do algoritmo RND/SQP, por eles desenvolvido, CERVANTES & BIEGLER (1998) comprovaram a superioridade do seu algoritmo, utilizando-o para resolver problemas de controle ótimo de colunas de destilação. Os autores testaram o seu método tanto para resolver um problema em batelada, como para um problema de arranque de processo. O custo computacional, usando como restrições do problema de otimização modelos de colunas com poucos estágios (8 à 15) e trabalhando em uma Estação de Trabalho (HP 777/110), é mais razoável (< 500 s).

FIKAR *et al.* (1998) estudaram um problema de controle ótimo no tempo de um depropanizador multicomponente industrial. Eles utilizaram SQP na formulação seqüencial e o pacote de integração DASSL (BRENAN *et al.*, 1989) para resolver o problema de controle ótimo no tempo da coluna na transição de um estado estacionário para outro. Devido ao grande número de variáveis e equações os autores foram obrigados a usar o método SQP na formulação seqüencial, um algoritmo mais robusto mas também menos veloz como se viu. No caso de problemas de robustez os autores recomendaram inicializar os cálculos com uma precisão menor, aumentando a precisão em cada iteração. O custo computacional que resultou da solução do problema de otimização utilizando o método seqüencial e um algoritmo de otimização NLP, é muito maior do que com o método simultâneo.

2.6. Formulação Matemática do Problema

Para resolver um problema de controle ótimo no tempo de um processo de arranque ou de um processo de transição de uma coluna de destilação de um estado estacionário para outro estado estacionário, recorre-se geralmente a um modelo de estado, formado por um sistema de equações diferenciais e algébricas, da forma das eqs. (1.1-1), e incluindo, além das condições iniciais, também condições finais da forma (problema de condições de contorno):

$$\mathbf{x}(t^{fin}) = \mathbf{x}^{stat} . \quad (2.6-1)$$

O problema de controle ótimo no tempo consiste na determinação do vetor de controle $\mathbf{u}(t)$, limitado por \mathbf{u}_{min} e \mathbf{u}_{max} , de tal maneira que a solução $\mathbf{x}(t)$ correspondente satisfaz as condições de contorno, dentro dos limites \mathbf{x}_{min} e \mathbf{x}_{max} , minimizando ao mesmo tempo o Índice de Desempenho

$$J = \int_0^{t^{fin}} dt . \quad (2.6-2)$$

2.6.1. Parâmetro do Tempo Final

ROSEN & LUUS (1989) propuseram introduzir um parâmetro (t^{fin}) na formulação do problema de controle ótimo:

$$\tau = \frac{t}{t^{fin}} , \quad (2.6-3a)$$

com:

$$0 \leq \tau \leq 1, \quad (2.6-3b)$$

e

$$t^{fin} > 0. \quad (2.6-3c)$$

A troca de variável, proposta pelos autores, é equivalente à forma geral de normalização apresentada por DADEBO & McAULEY (1995b) com $t^0 = 0$:

$$\tau = \frac{t - t^0}{t^{fin} - t^0}. \quad (2.6-4)$$

Com a introdução de um parâmetro representando o tempo de transição à otimizar, o problema geral de controle ótimo, eq. (1.1-1), pode ser rescrito da forma geral:

$$\min_{u, t^{fin}} J(x, u, t^{fin}, \tau), \quad (2.6-5a)$$

sujeito à:

$$\frac{dx}{d\tau} = F(x, u, t^{fin}, \tau), \quad (2.6-5b)$$

$$h(x, u, t^{fin}, \tau) = 0, \quad (2.6-5c)$$

$$x(0) = x^0, \quad (2.6-5d)$$

$$x(1) = x^{stat}, \quad (2.6-5e)$$

$$x_{min} \leq x(\tau) \leq x_{max}, \quad (2.6-5f)$$

$$u_{min} \leq u(\tau) \leq u_{max} , \quad (2.6-5g)$$

$$t_{min}^{fin} \leq t^{fin} \leq t_{max}^{fin} . \quad (2.6-5h)$$

No presente caso, de acordo com (2.6-2), o Índice de Desempenho a minimizar para o problema de controle ótimo no tempo tem a forma:

$$J = t^{fin} . \quad (2.6-6)$$

2.6.2. Função Objetivo Aumentada

Para garantir que as variáveis do estado satisfaçam as condições finais (2.6-5e), vários autores propuseram incorporá-las ao Índice de Desempenho Aumentado. Uma vantagem desta formulação é que pode ser usado um método de integração IVP simples.

Para resolver um problema de controle ótimo no tempo, utilizando o método IDP e começando com o chute inicial $u(0)=u^0$, ROSEN & LUUS (1989) propuseram minimizar o Índice de Desempenho Aumentado (2.6-7) sequencialmente, com $J^{\delta=0}$, $J^{0,1}$, $J^{0,5}$ e com J^1 , se fosse necessário. $x(1)$ representa o estado calculado da respectiva iteração e x^{stat} o estado final desejado.

$$J^{\delta} = \delta t^{fin} + (1-\delta) \sum (x(1) - x^{stat})^2 . \quad (2.6-7)$$

Também LUUS & ROSEN (1991) propuseram utilizar um Índice de Desempenho aumentado por uma função de penalização introduzida. No caso dado de equações (inequações veja LUUS & ROSEN, 1991) de condições finais, eles apresentam para o Índice de Desempenho Aumentado a seguinte forma geral:

$$J = I[x(0), t^{fin}] + F[x(t^{fin})]. \quad (2.6-8)$$

Sendo $F[x(t^{fin})]$ a função de penalização, e δ_i um fator da componente i , os autores escreveram a parte aumentada da forma:

$$F[x(t^{fin})] = \sum_i \delta_i [x_i(t^{fin}) - x^{stat}_i]^2. \quad (2.6-9)$$

Outras publicações descrevem aplicações semelhantes de Funções Objetivos aumentadas pelas condições finais do problema. BOJKOV & LUUS (1994a), por exemplo, aplicaram para problemas de controle ótimo no tempo um Índice de Desempenho Aumentado, relacionado com um tempo final t^{fin} , apresentando um algoritmo modificado da IDP, utilizando um dos seguintes termos:

$$|x(t^{fin}) - x^{stat}| \leq \varepsilon, \quad (2.6-10)$$

ou:

$$\left| \frac{x(t^{fin})}{x^{stat}} - 1 \right| \leq \varepsilon. \quad (2.6-11)$$

CARLOFF & REICHERT (1994) trataram o tema de controle ótimo no tempo de um reator. Mesmo assim, partes das suas formulações podem ser aplicadas em outros problemas, como em problemas de colunas de destilação. Para determinar a trajetória do vetor de controle $u(t)$, a qual otimiza o comportamento de arranque e de transitória de um reator, eles propuseram a função de Bolza como Índice de Desempenho:

$$\min_{u(t)} J = f[x(t^{fin}), t^{fin}] + \int_0^{t^{fin}} F[x(t), u(t), t] dt. \quad (2.6-12)$$

Para F os autores utilizaram a função seguinte (reduzida para a aplicação no problema aqui tratado), sendo δ a potência do fator peso do tempo:

$$F = \left(1 - \frac{\mathbf{x}}{\mathbf{x}^{stat}}\right)^2 t^\delta, \quad (2.6-13a)$$

com:

$$\delta > 0. \quad (2.6-13b)$$

Para problemas de otimização dinâmica DADEBO & McAULEY (1995a) utilizaram uma função de penalização de erro absoluto. O método proposto por eles é difícil de aplicar para SQP, devido à não-diferenciabilidade do Índice de Desempenho aumentado mas, segundo os autores, muito adequado para o algoritmo de IDP que não precisa do cálculo dos gradientes. Um Índice de Desempenho semelhante foi apresentado pelos mesmos autores (DADEBO & McAULEY, 1995b) para ser usado no método IDP com o tempo final como variável de decisão.

2.7. Conclusões

A revisão da literatura recente levou à conclusão que os métodos de otimização SQP e IDP poderiam ser os mais adequados para resolver problemas de controle ótimo de uma forma rápida e robusta. Com base nas experiências publicadas, o método de formulação simultânea do NLP foi utilizado neste trabalho, sendo desta maneira possível desenvolver um algoritmo mais rápido.

O problema da falta de robustez foi enfrentado aplicando um modelo de ordem reduzida da coluna de destilação, como descrito na literatura para outros tipos de problema. A Função Objetivo do problema de controle ótimo no tempo de uma coluna de destilação foi formulada, baseando-se nas recomendações feitas na literatura estudada.

3. Métodos Utilizados

Apresentam-se neste capítulo os métodos e algoritmos que, após o estudo da literatura, foram considerados os mais prometedores para conseguir o objetivo deste trabalho: Desenvolver um algoritmo rápido e robusto para resolver problemas de controle ótimo no tempo de colunas de destilação.

O problema de controle ótimo no tempo, levando uma coluna de destilação de um estado estacionário inicial até um novo estado estacionário, é resolvido usando um modelo de uma coluna de destilação binária prato-a-prato. O modelo descrito por SKOGESTAD & MORARI (1988), representando o sistema de restrições do problema de otimização, apresenta-se no primeiro subcapítulo. Em seguida, no próximo subcapítulo o problema de controle ótimo é formulado, usando uma Função Objetivo adequada, sujeito ao sistema de equações diferenciais e algébricas como as respectivas restrições e limitado por condições de contorno.

No terceiro subcapítulo, a ordem do modelo matemático da coluna é reduzida usando a colocação ortogonal discreta, sendo as raízes dos polinômios de Hahn os pontos de colocação, tal como foi recomendado por STEWART *et al.* (1985). O método da colocação ortogonal é aplicado ao modelo dinâmico, agora de ordem reduzida, para discretizar a parte diferencial no tempo, levando a um modelo algébrico, como mostra a penúltima parte deste capítulo. Consequentemente o sistema de DAE é resolvido nos pontos de colocação e pode ser otimizado com SQP.

A alta dimensão do problema de otimização, especialmente quando resolvido usando o método simultâneo, leva a um problema muito sensível, difícil para se aplicar o método SQP. O algoritmo de IDP tem a vantagem de convergir até o ótimo global mesmo partindo de valores iniciais longe deste ótimo, uma tarefa extremamente difícil e até impossível para o SQP. Por isso será apresentado o desenvolvimento deste algoritmo no último subcapítulo para ser testado como método de inicialização do

problema de controle ótimo no tempo, resultando desta maneira em um algoritmo mais robusto.

3.1. Modelo Matemático da Coluna

Neste trabalho, um modelo de uma coluna de destilação binária prato-a-prato, descrito por SKOGESTAD & MORARI (1988), formará o sistema de restrições do problema de controle ótimo no tempo. A Figura 3.1-1 mostra esquematicamente a coluna, sendo as hipóteses consideradas pelos autores as seguintes:

- volatilidade relativa constante;
- *Holdup* do vapor desprezível;
- *Overflow* molar constante;
- Condensador total;
- Eficiência 100% em cada estágio de separação;
- *Holdup* do líquido no estado estacionário igual em todos os pratos;
- Controles de nível perfeito no refeedor e no tanque de acúmulo de condensado.

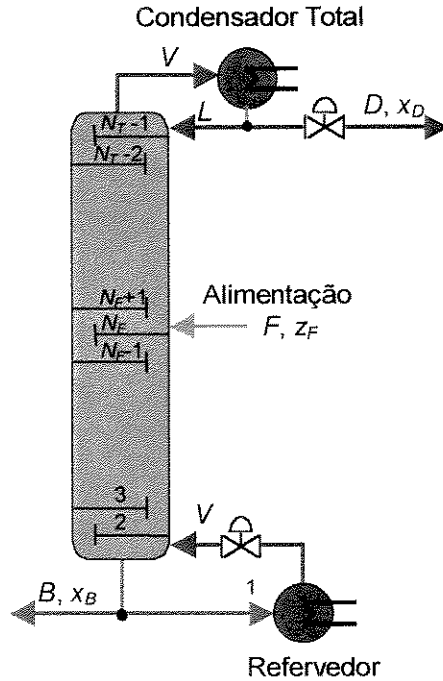


Figura 3.1-1: Esquema da Coluna de Destilação Binária Prato-a-Prato

Da hipótese da volatilidade relativa constante tem-se a relação de equilíbrio:

$$y_m = \frac{\alpha x_m}{1 + (\alpha - 1)x_m}, \quad (m = 1, \dots, N_T - 1), \quad (3.1-1a)$$

para cada prato e

$$y_{N_T-1} = x_D = x_{N_T}, \quad (3.1-1b)$$

para o condensador total. Anota-se aqui, que o subscrito indica o número do prato.

Seja M_m o "holdup" do líquido no prato m ,

$$M_m = M_{um} + M_{om}, \quad (3.1-2)$$

onde M_{om} é a quantidade de líquido acima do vertedouro. Pela relação de Francis tem-se:

$$M_{om} = k L_m^{2/3}. \quad (3.1-3)$$

SKOGESTAD & MORARI (1988) consideram a relação geral

$$dL_m = \lambda dV + \frac{1}{\tau_L} dM_m, \quad (3.1-4)$$

onde λ representa o efeito inicial da variação de vapor sobre a vazão do líquido do prato, e τ_L é a constante de tempo do prato (admitindo uma dinâmica de primeira ordem). Deprezoando o efeito do vapor ($\lambda=0$) tem-se então

$$dL_m = \frac{1}{\tau_L} dM_m, \quad (3.1-5)$$

ou, supondo-se que τ_L é constante e a mesma para todos os pratos:

$$L_m = L_m^0 + \frac{M_m - M_m^0}{\tau_L}, \quad (m = 2, \dots, N_T - 1). \quad (3.1-6)$$

O valor de τ_L é estimado usando a relação de Francis:

$$\frac{dM_m}{dL_m} = \frac{dM_{om}}{dL_m} = \frac{2}{3} \frac{M_{om}}{L_m}, \quad (3.1-7)$$

ou

$$\tau_L = \frac{dM_m}{dL_m} = \frac{2}{3} \frac{M_{om}}{L_m} = \frac{2}{3} \frac{M_{om}}{M_m} \frac{F}{L_m} \frac{M_m}{F}. \quad (3.1-8)$$

Para estimar τ_L usando (3.1-8) serão feitas as aproximações $L_m \approx L_m^0$, $(M_{om}/M_m) \approx 0,5$ e $(M_m/F) \approx 0,5$ min (SKOGESTAD & MORARI, 1988), obtendo-se finalmente:

$$\tau_L = \frac{F}{6L^0}, \text{ [min]}. \quad (3.1-9)$$

Alem das equações algébricas (3.1-1) e (3.1-6), o modelo da coluna é formado por equações diferenciais provenientes dos balanços de massa. Para cada estágio de separação (à exceção do prato de alimentação, do condensador e do refeedor) tem-se:

$$\frac{d(M_m x_m)}{dt} = L_{m+1} x_{m+1} + V y_{m-1} - L_m x_m - V y_m, \quad (3.1-10)$$

e:

$$\frac{dM_m}{dt} = L_{m+1} - L_m. \quad (3.1-11)$$

Rescrevendo a eq. (3.1-10) como:

$$M_m \frac{dx_m}{dt} + x_m \frac{dM_m}{dt} = L_{m+1} x_{m+1} + V y_{m-1} - L_m x_m - V y_m, \quad (3.1-12)$$

e substituindo a eq. (3.1-11) em eq. (3.1-12), obtém-se para o balanço de massa a equação:

$$M_m \frac{dx_m}{dt} = L_{m+1}(x_{m+1} - x_m) + V(y_{m-1} - y_m). \quad (3.1-13)$$

No estado estacionário tem-se:

$$L_m^0 = L^0 + F_L, \quad \text{para } (m \leq N_F), \quad (3.1-14)$$

sendo:

$$F_L = q_F F, \quad (3.1-15a)$$

e

$$F_V = F - F_L. \quad (3.1-15b)$$

Com $q_F = 1$, ou seja alimentação como líquido saturado, tem-se finalmente:

$$L_m^0 = L^0 + F, \quad \text{para } (m \leq N_F), \quad (3.1-16a)$$

e

$$L_m^0 = L^0, \quad \text{para } (m > N_F), \quad (3.1-16b)$$

que são condições iniciais do problema.

Como o "*holdup*" de vapor foi considerado desprezível e a alimentação é totalmente líquida, a vazão de vapor é idêntica em cada estágio de separação. As equações diferenciais provenientes dos balanços de massa, que, juntas com as equações algébricas, formam o sistema de DAE do modelo prato-a-prato da coluna de destilação binária, são as seguintes:

- **Refrervedor**, ($m = 1$):

Com

$$x_1 = x_B, \quad M_1 = M_B, \quad (3.1-17a)$$

e assumindo que o "*holdup*" de líquido no fundo seja constante, tem-se:

$$\frac{dM_B}{dt} = 0, \quad (3.1-17b)$$

o que leva a:

$$M_B = M_B^0. \quad (3.1-17c)$$

O balanço de massa durante a transição escreve-se então da forma:

$$\frac{dx_B}{dt} = \frac{L_2 x_2 - Vy_B - Bx_B}{M_B}. \quad (3.1-17d)$$

- Pratos da seção de esgotamento ($m = 2, \dots, N_F - 1$):

$$\frac{dx_m}{dt} = \frac{L_{m+1}(x_{m+1} - x_m) + V(y_{m-1} - y_m)}{M_m}, \quad (3.1-18a)$$

com:

$$\frac{dM_m}{dt} = L_{m+1} - L_m. \quad (3.1-18b)$$

- Prato de alimentação ($m = N_F$):

$$\frac{dx_{N_F}}{dt} = \frac{L_{N_F+1}(x_{N_F+1} - x_{N_F}) + F(z_F - x_{N_F}) + V(y_{N_F-1} - y_{N_F})}{M_{N_F}}, \quad (3.1-19a)$$

com:

$$\frac{dM_{N_F}}{dt} = L_{N_F+1} + F - L_{N_F}. \quad (3.1-19b)$$

- Pratos da seção de retificação ($m = N_F + 1, \dots, N_T - 1$):

$$\frac{dx_m}{dt} = \frac{L_{m+1}(x_{m+1} - x_m) + V(y_{m-1} - y_m)}{M_m}, \quad (3.1-20a)$$

com:

$$\frac{dM_m}{dt} = L_{m+1} - L_m, \quad (3.1-20b)$$

onde, para $m = N_T - 1$ se tem:

$$L_{m+1} = L_{N_T} = L. \quad (3.1-21)$$

- **Condensador total** ($m = N_T$):

Assumindo que o "holdup" de líquido no tanque do condensado seja constante tem-se ($M_{N_T} = M_D$):

$$\frac{dM_D}{dt} = 0, \quad (3.1-22a)$$

ou:

$$M_D = M_D^0. \quad (3.1-22b)$$

Com L representando a vazão do refluxo de líquido ($L_{N_T} = L$) e:

$$L + D = V, \quad (3.1-22c)$$

o balanço de massa no condensador total toma a forma:

$$\frac{dx_D}{dt} = \frac{V(y_{N_T-1} - x_D)}{M_D}, \quad (3.1-22d)$$

com:

$$x_{N_T} = x_D. \quad (3.1-22e)$$

Os dados técnicos da coluna, assim como a especificação dos estados estacionários inicial e final, são apresentados na Tabela 3.1.-1:

Tabela 3.1-1: Dados da Coluna

Nomenclatura	Dados técnicos	Estado Inicial	Estado Final
q_F	1		
z_F	0,5		
α	1,5		
N_T	41		
N_F	21		
F	1,0 kmol/min		
M_0	0,5 kmol		
V_{min}	3,0 kmol/min		
V_{max}	3,5 kmol/min		
D_{min}	0,0 kmol/min		
D_{max}	0,755 kmol/min		
D		0,555 kmol/min	0,500 kmol/min
V		3,292 kmol/min	3,206 kmol/min
x_D		~0,9	~0,99

3.2. Formulação do Problema de Controle Ótimo no Tempo

Partindo da introdução de um parâmetro, representando o tempo final à minimizar t^{fin} , e gerando o tempo normalizado τ , a Função Objetivo escolhida para formular o

problema de controle ótimo no tempo, é baseada na Função Objetivo, apresentada por ROSEN & LUUS (1989).

As variáveis D e V foram escolhidas como variáveis de controle sendo elas a razão do destilado e do vapor respectivamente, levando a concentração da componente leve do destilado x_D , escolhido como variável de estado, de um estado estacionário inicial até um estado estacionário final. Para garantir convergência até o estado final, a Função Objetivo contem, além do parâmetro a minimizar t^{fin} , o somatório da diferença entre o $x_D(\tau=1)$ e o x_D desejado no estado estacionário novo. Os valores δ e γ são pesos e foram escolhidos $\delta = 10^{-6}$ e $\gamma = 10^5$ como valores adequados. Suficiente para garantir uma diferença entre $x_D(\tau=1)$ e x_D^{stat} de aproximadamente 10^{-6} à 10^{-4} , assumindo um valor ótimo de t^{fin} (t^{fin*}) entre 0,1 min e 1000 min, respectivamente.

Uma vez escolhida a Função Objetivo e tendo descrito o modelo da coluna de destilação que forma o sistema de restrições do problema de otimização, o problema de controle ótimo no tempo pode ser escrito da forma:

$$\min_{D,V,t^{fin}} f(x_D, D, V, t^{fin}, \tau) = \delta t^{fin} + (1-\delta)\gamma (x_D(\tau=1) - x_D^{stat})^2, \quad (3.2-1a)$$

sujeito à (para $m \neq 1, N_F, N_T$):

$$M_m \frac{1}{t^{fin}} \frac{dx_m}{d\tau} = L_{m+1}(x_{m+1} - x_m) + V(y_{m-1} - y_m), \quad (3.2-1b)$$

$$\frac{1}{t^{fin}} \frac{dM_m}{d\tau} = L_{m+1} - L_m, \quad (3.2-1c)$$

$$L_m = L_m^0 + \frac{M_m - M_m^0}{\tau_L}, \quad (3.2-1d)$$

$$y_m = \frac{\alpha X_m}{1 + (\alpha - 1)X_m} \quad , \quad (3.2-1e)$$

$$x_D(\tau = 0) = x_D^0 \quad , \quad (3.2-1f)$$

$$0 \leq x \leq 1 \quad , \quad (3.2-1g)$$

$$3,0 \leq V \leq 3,5 \quad , \quad (3.2-1h)$$

$$0,0 \leq D \leq 0,755 \quad , \quad (3.2-1i)$$

$$t_{min}^{fin} \leq t^{fin} \leq t_{max}^{fin} \quad . \quad (3.2-1j)$$

Observe-se que foram aqui representadas as equações do modelo diferenciais e algébricas correspondentes a um prato genérico de coluna ($m \neq 1, N_F, N_T$). Por conseguinte, às equações citadas acima é necessário ainda acrescentar as correspondentes ao refeedor, prato de alimentação e condensador.

Os valores limites adotados da concentração do destilado são valores físicos de 0 e 1. Já os valores limites adotados para as variáveis de controle não são necessariamente os limites físicos da coluna, devido ao fato que dados reais da coluna não estão a disposição. Para inicializar o NLP, usando o algoritmo SQP, são usados valores de controle muito próximos aos extremos V_{max} e D_{min} , respectivamente. Os limites do intervalo do parâmetro do tempo de transição a minimizar variam e são indicadores de um algoritmo robusto, resolvendo o problema de controle ótimo com um valor inicial do parâmetro (t^{fin}) longe do valor ótimo e num intervalo (t_{min}^{fin} à t_{max}^{fin}) amplo. Porém, os valores extremos de t^{fin} variam de exemplo à exemplo.

3.3. Redução da Ordem do Modelo

Baseado no método recomendado por SEFERLIS & HRYMAK (1994a,b) e sistematizado por CARVALHO (1996), o refeedor, o condensador e o prato de alimentação, são tratados como pratos discretos, e apenas os pratos do Elemento 1 (seção de esgotamento) e do Elemento 2 (seção de retificação), mostrados em Figura 3.3-1, são aproximados usando a colocação ortogonal discreta.

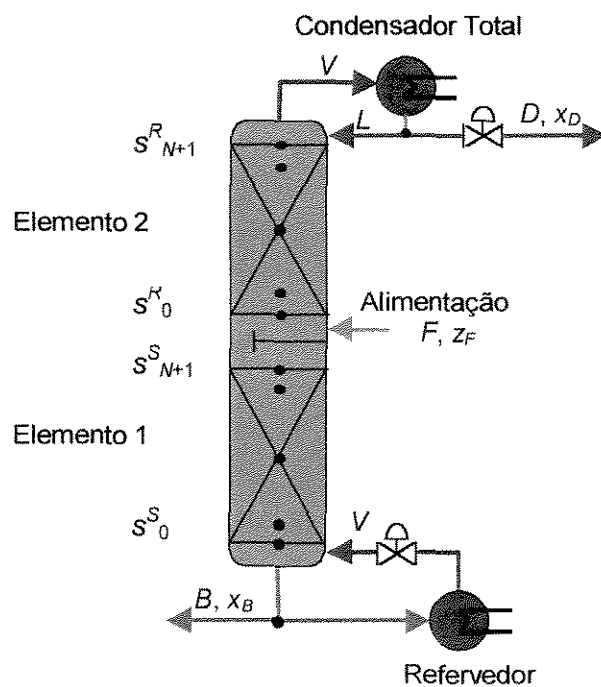


Figura 3.3-1: Modelo de Ordem Reduzida da Coluna de Destilação

Considere-se, por exemplo, o Elemento 2 da coluna (seção de retificação). As equações do balanço de massa (3.1-20) são colocadas na forma simbólica ($m = N_F+1, \dots, N_T-1$):

$$\frac{M_m(\tau)}{t^{fin}} \frac{dx_m(\tau)}{d\tau} = L_{m+1}(\tau) \Delta x_m(\tau) - V(\tau) \nabla y_m(\tau), \quad (3.3-1)$$

$$\frac{1}{t^{\bar{m}}} \frac{dM_m(\tau)}{d\tau} = \Delta L_m(\tau), \quad (3.3-2)$$

onde:

$$\Delta x_m(\tau) = x_{m+1}(\tau) - x_m(\tau), \quad (3.3-3a)$$

$$\nabla y_m(\tau) = y_m(\tau) - y_{m-1}(\tau), \quad (3.3-3b)$$

$$\Delta L_m(\tau) = L_{m+1}(\tau) - L_m(\tau). \quad (3.3-3c)$$

As equações diferenciais às diferenças (3.3-1) e (3.3-2) podem ainda ser colocadas numa forma mais geral, usando uma variável especial s^R que toma valores no intervalo $[0, N^R-1]$ e que na posição de cada prato toma um valor inteiro (0,1,..., N^R-1):

$$\frac{M(s^R, \tau)}{t^{\bar{m}}} \frac{dx(s^R, \tau)}{d\tau} = L(s^R + 1, \tau) \Delta x(s^R, \tau) - V(\tau) \nabla y(s^R, \tau), \quad (3.3-4)$$

$$\frac{1}{t^{\bar{m}}} \frac{dM(s^R, \tau)}{d\tau} = \Delta L(s^R, \tau). \quad (3.3-5)$$

A condição de equilíbrio (3.1-1a) toma então a forma:

$$y(s^R, \tau) = \frac{\alpha x(s^R, \tau)}{1 + (\alpha - 1)x(s^R, \tau)}. \quad (3.3-6)$$

Na estratégia de colocação ortogonal discreta usada neste trabalho, como foi dito, as funções $x(s^R, \tau)$, $y(s^R, \tau)$ e $L(s^R, \tau)$ são aproximadas por polinômios interpoladores de Lagrange:

$$x(s^R, \tau) \approx \sum_{j=0}^{N+1} \Gamma_j(s^R) x(s^R_j, \tau), \quad (3.3-7a)$$

$$y(s^R, \tau) \approx \sum_{j=0}^{N+1} \Gamma_j(s^R) y(s^R_j, \tau), \quad (3.3-7b)$$

$$L(s^R, \tau) \approx \sum_{j=0}^{N+1} \Gamma_j(s^R) L(s^R_j, \tau). \quad (3.3-7c)$$

Neste caso, são considerados N pontos de colocação internos ($j=1, \dots, N$), com $s^R_{j=0}=0$ e $s^R_{j=N+1}=N^R-1$. Os N pontos de colocação internos são zeros de um polinômio de Hahn (CARVALHO, 1996). Usando as aproximações (3.3-7) tem-se (CARVALHO, 1996):

$$\Delta x(s^R, \tau) \approx \sum_{j=0}^{N+1} \Delta \Gamma_j(s^R) x(s^R_j, \tau), \quad (3.3-8a)$$

$$\nabla y(s^R, \tau) \approx \sum_{j=0}^{N+1} \nabla \Gamma_j(s^R) y(s^R_j, \tau), \quad (3.3-8b)$$

$$\Delta L(s^R, \tau) \approx \sum_{j=0}^{N+1} \Delta \Gamma_j(s^R) L(s^R_j, \tau). \quad (3.3-8c)$$

De acordo com o método da colocação ortogonal, o problema definido pelas $2N^R$ equações diferenciais (3.3-4) e (3.3-5) é substituído por um problema de dimensão $2(N+2)$ formado a partir da condição de que as equações obtidas com as aproximações (3.3-7) e (3.3-8) sejam satisfeitas nos $(N+2)$ pontos de colocação s^R_j ($j=0, 1, \dots, N+1$). Para os pontos de colocação internos s^R_n ($n=1, 2, \dots, N$), as $2N$ equações assim obtidas são:

$$\frac{M(s^R_n, \tau)}{t^{\tilde{m}}} \frac{dx(s^R_n, \tau)}{d\tau} = \left[\sum_{j=0}^{N+1} \Gamma_j(s^R_n + 1) L(s^R_j, \tau) \right] \left[\sum_{j=0}^{N+1} \Delta \Gamma_j(s^R_n) x(s^R_j, \tau) \right] - V(\tau) \left[\sum_{j=0}^{N+1} \nabla \Gamma_j(s^R_n) y(s^R_j, \tau) \right], \quad (3.3-9)$$

$$\frac{1}{t^{fin}} \frac{dM(s^R_n, \tau)}{d\tau} = \sum_{j=0}^{N+1} \Delta\Gamma_j(s^R_n) L(s^R_j, \tau). \quad (3.3-10)$$

Para $s^R_0 = 0$ (prato logo acima do prato de alimentação) tem-se:

$$\frac{M(s^R_0, \tau)}{t^{fin}} \frac{dx(s^R_0, \tau)}{d\tau} = \left[\sum_{j=0}^{N+1} \Gamma_j(s^R_0 + 1) L(s^R_j, \tau) \right] \left[\sum_{j=0}^{N+1} \Delta\Gamma_j(s^R_0) x(s^R_j, \tau) \right] - V(\tau) [y(s^R_0) - y_{N_f}] \quad (3.3-11)$$

$$\frac{1}{t^{fin}} \frac{dM(s^R_0, \tau)}{d\tau} = \sum_{j=0}^{N+1} \Delta\Gamma_j(s^R_0) L(s^R_j, \tau). \quad (3.3-12)$$

Por outro lado $s^R_{N+1} = N^R - 1$ é o prato logo abaixo do condensador, tendo-se:

$$\frac{M(s^R_{N+1}, \tau)}{t^{fin}} \frac{dx(s^R_{N+1}, \tau)}{d\tau} = L(\tau) [x_D(\tau) - x(s^R_{N+1}, \tau)] - V(\tau) \left[\sum_{j=0}^{N+1} \nabla\Gamma_j(s^R_{N+1}) y(s^R_j, \tau) \right] \quad (3.3-13)$$

$$\frac{1}{t^{fin}} \frac{dM(s^R_{N+1}, \tau)}{d\tau} = L(\tau) - L(s^R_{N+1}, \tau). \quad (3.3-14)$$

Neste trabalho foi adotado, para o elemento 1 (seção de esgotamento da coluna) o mesmo número N de pontos de colocação internos e o mesmo esquema de colocação. Portanto, se fossem N^S o número de pratos da seção de esgotamento, o número de equações diferenciais do modelo completo prato-a-prato será:

$$N_{eq} = 2N_T - 2 = 2(N^R + N^S + 3) - 2. \quad (3.3-15)$$

Para o modelo de ordem reduzida o número de equações diferenciais obtido será:

$$N_{eq, \text{ modelo reduzido}} = 2[2(N + 2) + 3] - 2, \quad (3.3-16)$$

pois as $2(N + 2)$ equações dos elementos 1 e 2 são resolvidas simultaneamente com as equações do refeedor, do prato de alimentação e do condensador.

3.4. Formulação do Método Simultâneo

Como foi dito, no algoritmo SQP-simultâneo, usado na otimização dinâmica, as equações do modelo dinâmico são discretizadas no tempo usando o método da colocação ortogonal. Deste modo, as equações diferenciais do modelo dinâmico são transformadas em equações algébricas, as quais são consideradas como restrições pelo algoritmo SQP. Considerando uma colocação no tempo em que além dos K pontos de colocação internos são usados os dois pontos extremos do intervalo do tempo normalizado $[0,1]$ e sendo os valores das variáveis, em $\tau=0$, dados pelo estado inicial, as equações diferenciais do balanço de massa (3.3-9) e (3.3-10), para cada valor do ponto de colocação s_n^R ($n = 1, \dots, N$) levam a $2(K+1)$ equações algébricas:

$$0 = \frac{M^i(s_n^R)}{t_{fin}} \sum_{j=0}^{K+1} \phi^j(\tau^i) x^j(s_n^R) - \sum_{j=0}^{N+1} \Gamma_j(s_n^R + 1) L^i(s_j^R) \sum_{j=0}^{N+1} \Delta \Gamma_j(s_n^R) x^j(s_j^R) + V^i \sum_{j=0}^{N+1} \nabla \Gamma_j(s_n^R) y^j(s_j^R), \quad (3.4-1a)$$

$$0 = \frac{1}{t_{fin}} \sum_{j=0}^{K+1} \phi^j(\tau^i) M^j(s_n^R) - \sum_{j=0}^{N+1} \Delta \Gamma_j(s_n^R) L^j(s_j^R), \quad (3.4-1b)$$

Nestas equações foi usada a nomenclatura:

$$M^i(s_n^R) = M(s_n^R, \tau^i), \quad (3.4-2a)$$

$$x^i(s^R_n) = x(s^R_n, \tau^i), \text{ etc,} \quad (i = 1, 2, \dots, K+1). \quad (3.4-2b)$$

As equações (3.4-1) são completadas pelas relações algébricas já apresentadas:

$$L^i(s^R_n) = L^0(s^R_n) + \frac{M^i(s^R_n) - M^0(s^R_n)}{\tau_L}, \quad (3.4-3a)$$

$$y^i(s^R_n) = \frac{\alpha x^i(s^R_n)}{1 + (\alpha - 1)x^i(s^R_n)}. \quad (3.4-3b)$$

O modelo do processo de ordem reduzida, na forma usada no método SQP-simultâneo, é apresentado completo em ANEXO.

3.5. Algoritmo de IDP

Na sua maioria, os métodos de NLP foram desenvolvidos para descobrir ótimos locais. Num problema altamente não-linear e de alta dimensão como o problema de controle ótimo no temo de uma coluna de destilação em transição, podem existir muitos extremos locais e os valores da Função Objetivo podem variar significativamente. Porém, além do problema que a aplicação do algoritmo SQP precisar de um chute inicial muito próximo ao valor ótimo, também precisa-se de um intervalo suficientemente pequeno, incluindo o ótimo global e, preferivelmente, excluindo possíveis ótimos locais. Isso é uma tarefa extremamente difícil e até impossível no caso o resultado ótimo fosse totalmente desconhecido, o que é bastante freqüente.

Um dos poucos métodos que pode garantir um ótimo global (KEIL, 1996) é o método de Programação Dinâmica de Bellman. Neste trabalho um algoritmo de IDP, baseado no algoritmo apresentado por LUUS (1990b) é desenvolvido, permitindo apenas estados acessíveis para a grade da variável do estado, usando a técnica "clipping", descrito no subcapítulo 2.3.4., e usando uma redução iterativa das regiões.

A estratégia de controle contínua é transformado numa estratégia contínua por partes, dividindo o intervalo do tempo $0 \leq t \leq t^{fin}$ em S seções de tempo, cada um de tamanho variável como foi proposto por BOJKOV & LUUS (1994b). Em cada iteração, M_x pontos de grade são calculados gerando as grades de estado para cada um dos m_x valores de estado e M_u valores de cada um dos m_u variáveis de controle, incluindo a variável da seção do tempo, são usados para determinar as trajetórias parciais em cada seção do tempo. Em cada iteração escolhe-se o melhor valor da iteração anterior como ponto central para gerar uma nova grade de estado, reduzindo sucessivamente a região permitida, usando um fator de redução μ . Para evitar o problema de interpolação, para a continuação é aplicado o método recomendado por DeTREMBLAY & LUUS (1989) usando simplesmente a estratégia de controle ótima obtido no ponto da grade de estado mais próximo.

Da mesma maneira é gerado uma nova grade de controle, usando como ponto central a estratégia ótima da iteração anterior. No caso em que a estratégia de controle resulta exatamente no valor de um dos limites da região permitida, os pontos são distribuídos novamente, usando o valor ótimo da iteração anterior como um dos valores extremos da nova região reduzida, ao em vez de ser ponto central dela. Desta forma é garantido que todos os valores de controle sejam diferentes um do outro.

O pacote de *software* LSODE (HINDMARSH, 1980) é usado para gerar a grade de estado inicial e, em cada iteração, para integrar o sistema de equações diferenciais parciais em cada seção de tempo e cada ponto da grade de estado. O algoritmo do método IDP desenvolvido é o seguinte:

Passo 1: Dividir o intervalo do tempo normalizado em S seções.

Passo 2: Escolher os pontos centrais para a grade de controle e os respectivos números de pontos (ilustrado em Figura 3.5-1), tal como a região inicial, formada pelas condições físicas, para as grades de controle e estado (Figura 3.5-2).

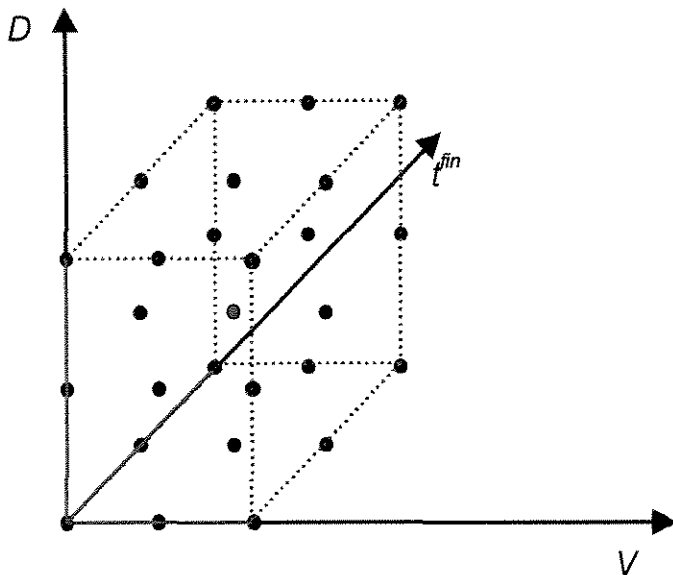


Figura 3.5-1: Grade de Controle com $M_u = 3$

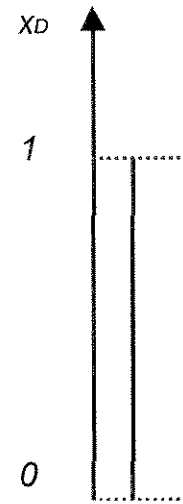


Figura 3.5-2: Região Inicial da Grade de Estado

Passo 3: Escolher M_x combinações dos valores de controle admissíveis e resolver o modelo dinâmico para cada um destes pontos de controle, gerando assim uma grade de estado para cada seção de tempo, como é ilustrado na Figura 3.5-3.

Passo 4: Começando na última seção de tempo, correspondendo ao tempo $\tau = 1 - 1/S$, o modelo dinâmico para cada ponto de grade do estado e integrando de $\tau = 1 - 1/S$ até $\tau = 1$ para todas as $(M_u)^{m_v}$ combinações de variáveis de controle, como é mostrado na Figura 3.5-4 em 2 dos $M_x = 5$ pontos de grade de estado para $(M_u)^{m_v} = 27$ combinações de variáveis de controle. As variáveis de controle que otimizam a Função Objetivo (linhas verde e azul) são armazenados para o próximo passo.

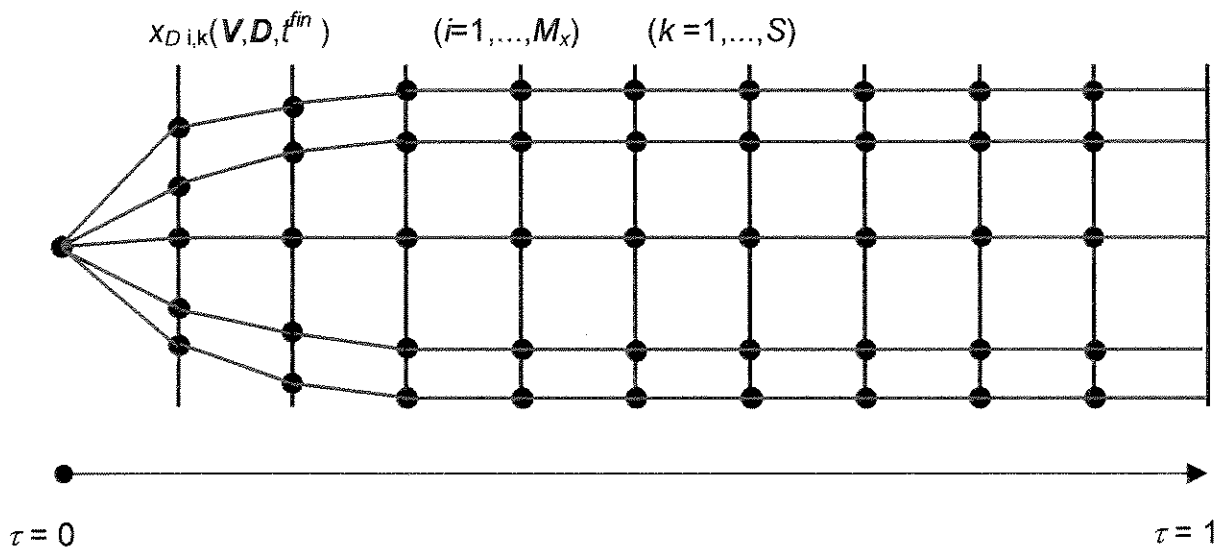


Figura 3.5-3: Geração das Grades de Estado com $M_x=5$ e $S=10$.

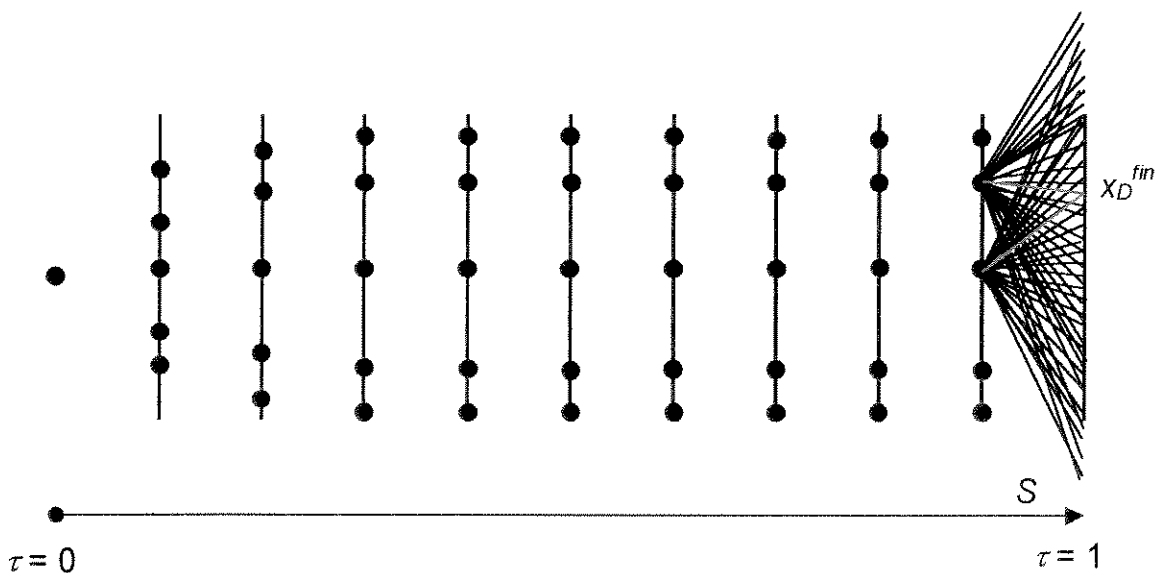


Figura 3.5-4: Integração parcial na Seção de Tempo S .

Passo 5: Passando para a seção $(S-1)$, correspondendo ao tempo $\tau = 1-2/S$, para cada ponto de grade o modelo dinâmico é integrado de $\tau = 1-2/S$ até $\tau = 1-1/S$ para todas as $(M_u)^{m_u}$ combinações de variáveis de controle, como é mostrado na Figura 3.5-5 em 2 dos $M_x = 5$ pontos de grade de estado para $(M_u)^{m_u} = 27$ combinações de variáveis de controle. Para continuar a

integração de $\tau = 1-1/S$ até $\tau = 1$ é usado o controle do passo 4 que corresponde ao ponto da grade mais próximo ao ponto de estado resultando da integração parcial. Os $(M_u)^{m_u}$ valores de controle são comparados e aqueles valores, que otimizam a Função Objetivo, são armazenados.

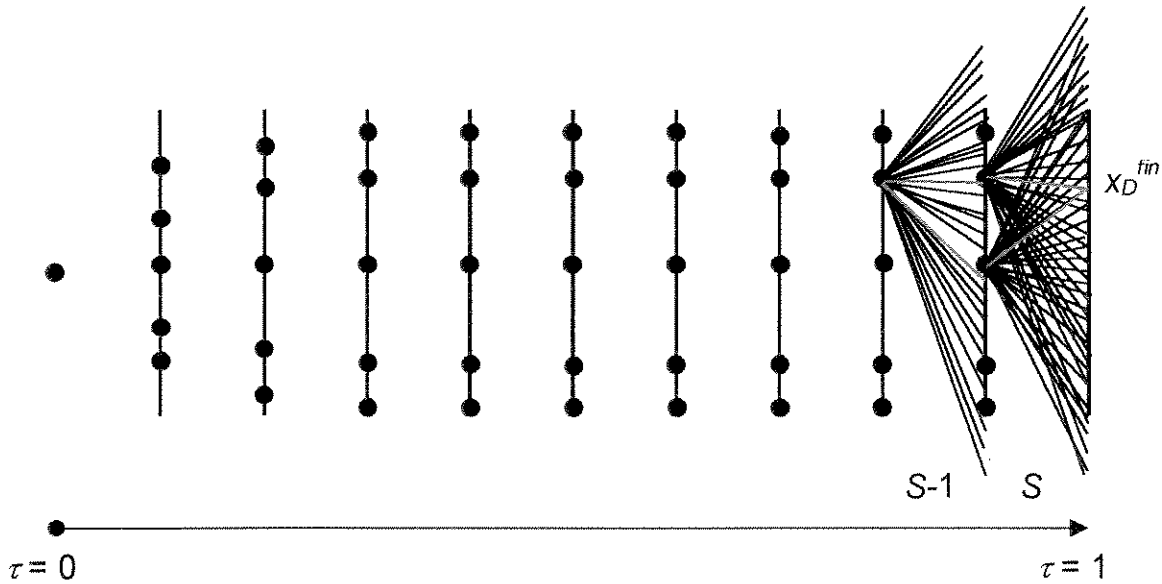


Figura 3.5-5: Integração parcial na Seção de Tempo S-1.

Passo 6: Continuar o processo até a seção de tempo 1, correspondendo ao tempo inicial $\tau = 0$. A estratégia de controle que otimiza a Função Objetivo e a respectiva trajetória de estado, são armazenados. A Figura 3.5-6 mostra este processo, sendo a linha azul a melhor trajetória de estado escolhido.

Passo 7: Reduzir a região para os valores de controle admissíveis, usando um fator de redução μ , sendo os pontos centrais em cada grade aqueles da estratégia de controle do passo 6.

Passo 8: Incrementar o índice da iteração por 1 e continuar o algoritmo no passo 3.

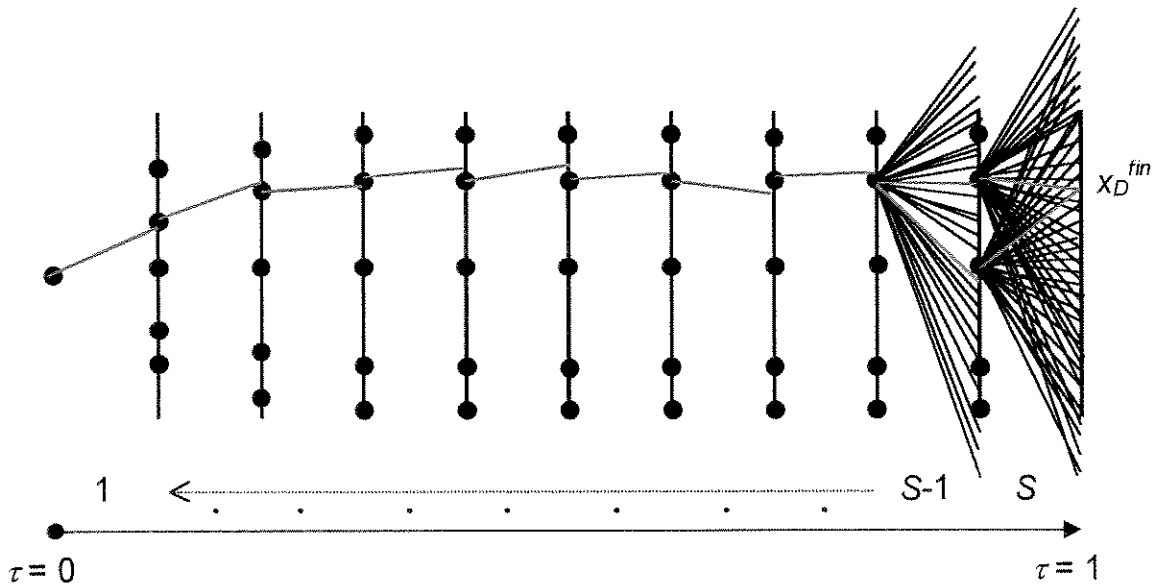


Figura 3.5-6: Escolha da Trajetória Ótima obtido ao Final da Iteração

Para garantir que a trajetória ótima escolhida é suficientemente "contínua", SCHRÖDER & MENDES (1998) desenvolveram um algoritmo que calcula o somatório dos quadrados das diferenças entre o valor da variável de estado no final de cada trajetória parcial e o seu valor no ponto de grade mais próximo usado para a continuação da trajetória na próxima seção de tempo. Em cada seção de tempo temos:

$$\varepsilon = \sum_{i=1}^{S-1} \left(x_D(t^{i+}) - x_D(t^{i-}) \right)^2, \quad (3.5-1)$$

onde o sobrescrito positivo (+) marca o valor de estado no final da trajetória parcial na seção de tempo i e o subscrito negativo (-) o valor no ponto da grade de estado, usado para continuação da trajetória na seção de tempo seguinte ($i+1$). O erro ε foi chamado de erro de continuação.

Baseado no erro de continuação, no somatório quadrático da diferença entre o tempo de transição ótimo, t^{fin*} , calculado na iteração atual (i) e na iteração anterior ($i-1$) e no somatório quadrático entre a concentração do destilado x_D e o seu valor no estado estacionário desejado x_D^{stat} pode-se desenvolver um critério de parada sendo este formado por:

$$\varepsilon \leq \varepsilon^{tol}, \quad (3.5-2a)$$

$$(t^{fin*}_i - t^{fin*}_{i-1})^2 \leq \varepsilon^{tol}, \quad (3.5-2b)$$

$$(x_{Di} - x_D^{stat})^2 \leq \varepsilon^{tol}, \quad (3.5-2c)$$

durante as últimas i iterações.

4. Resultados

Neste capítulo são apresentados os resultados obtidos com a aplicação dos métodos de otimização SQP e IDP, descritos acima, ao problema de controle ótimo no tempo da coluna de destilação binária, usando o modelo de ordem reduzida como apresentado no subcapítulo 3.3. A tolerância de convergência foi escolhida $\leq 10^{-9}$ para os pacotes de SQP, tanto utilizando o modelo completo como de ordem reduzida. Aplicando a colocação ortogonal em elementos finitos no tempo, a tolerância de convergência escolhida foi $\leq 10^{-7}$. A tolerância da solução das restrições, ou seja, a diferença dos resíduos do sistema de equações até zero nos pratos ou pontos de colocação, respectivamente, foi escolhida $\leq 10^{-13}$ ao usar o método SQP. LI & WOZNY (1996) trabalharam com a mesma definição de tolerância, mas com um valor, de $\leq 10^{-6}$, muito menos rígido. Usando o método IDP, o sistema de equações diferenciais foi resolvido com o pacote de integração LSODE, sendo $atol = 10^{-11}$ e $rtol = 10^{-9}$ as respectivas configurações de tolerância. Os tempos de CPU apresentados neste trabalho foram conseguidos com uma SUN-SPARC STATION 20 de 75 MHz.

No primeiro subcapítulo examina-se a eficiência da aplicação de colocação ortogonal contínua no tempo para discretizar as equações diferenciais do modelo. Comprova-se que o método pode ser tão exato quanto um método de integração de um pacote de "software" convencional. O segundo subcapítulo justifica a escolha do algoritmo SQP, mostrando que este é bastante adequado para o problema de otimização dinâmica de alta dimensão com um número de graus de liberdade relativamente pequeno e uma matriz Jacobiana esparsa. O terceiro e o quarto subcapítulo apresentam os resultados do problema de controle ótimo no tempo, comparando o uso dos modelos de ordem completa e reduzida, comprovando desta forma a aplicabilidade e a eficiência do modelo de ordem reduzida quanto usado com os métodos IDP ou SQP.

O quinto subcapítulo mostra a necessidade de um método de inicialização para o SQP, possibilitando a convergência mesmo partindo de pontos iniciais longe da solução

ótima. Os dois subcapítulos seguintes apresentam os resultados das aplicações de IDP e do próprio SQP para a inicialização, conseguindo desta maneira um chute inicial perto da solução ótima que leva o algoritmo SQP à convergência. A análise dos resultados nos últimos subcapítulos leva à conclusão que foi possível desenvolver um método robusto e rápido para o problema de controle ótimo no tempo de uma coluna de destilação.

4.1. Eficiência do Método de Colocação Ortogonal no Tempo

No seu trabalho CARVALHO (1996) demonstrou a eficiência do método de colocação ortogonal discreta ao longo da coluna. Usando $N = 5$ pontos de colocação internos o autor conseguiu uma aproximação bastante boa do modelo de ordem completa da coluna de destilação, tanto no regime estacionário como transiente. A eficiência da colocação ortogonal contínua no tempo será mostrada neste capítulo. Para tal são comparadas as trajetórias $x_D(t)$ obtidas por integração dos modelos dinâmicos, completo e de ordem reduzida, assim como dos modelos discretizados no tempo pela colocação contínua.

Os modelos dinâmicos da coluna de destilação, completo e de ordem reduzida, foram integrados usando o pacote LSODE. Para resolver o modelo discretizado, formado agora por um sistema de equações algébricas com matriz Jacobiana esparsa, foi aplicada apenas a parte do pacote SQP de SCHMID & BIEGLER (1994) indicada para esta tarefa, e baseada no algoritmo MA28 da biblioteca HARWELL.

Exemplo: Neste exemplo não são usados variáveis de controle. O problema consiste em calcular a trajetória não otimizada, resolvendo apenas o modelo dinâmico. A coluna de destilação binária é levada do seu estado estacionário inicial até um novo estado estacionário, usando as especificações do novo estado estacionário desejado, já deste o começo do processo de transição. Assim os parâmetros do processo, D e V , são fixados nos seus valores finais, representando desta forma um processo de troca

de produto clássico. O parâmetro do tempo de transição, t^{fin} , é fixado em 1000 min para calcular a trajetória durante este período.

As Figuras 4.1-1 e 4.1-2 mostram as curvas da trajetória da concentração no destilado, usando o modelo de ordem completa e comparando os resultados da colocação ortogonal contínua no tempo com os do pacote de integração LSODE. A Figura 4.1-1 mostra o uso da colocação ortogonal contínua com $K = 3$ e 5 pontos de colocação internos no tempo e a Figura 4.1-2 com 7 e 9 pontos.

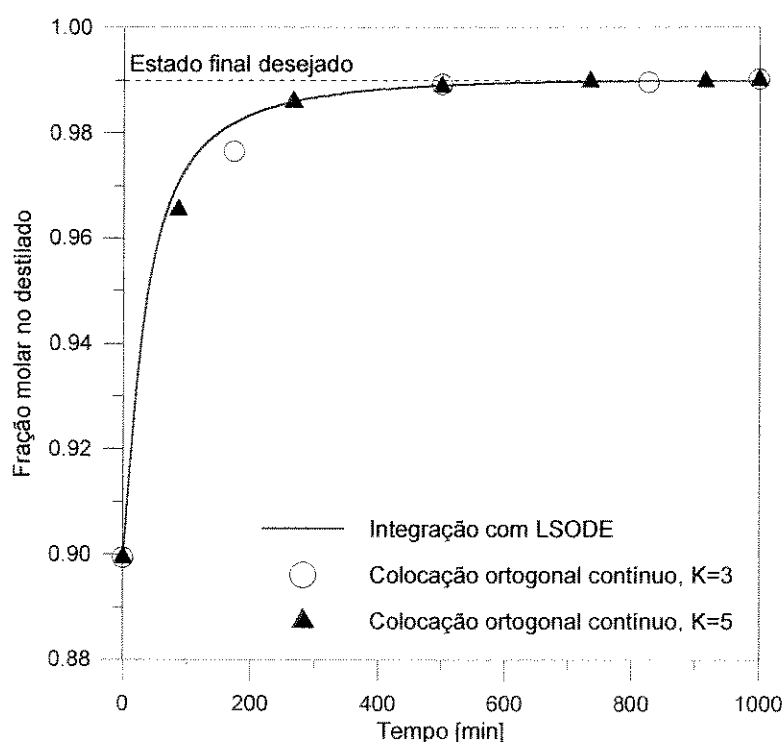


Figura 4.1-1: Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada

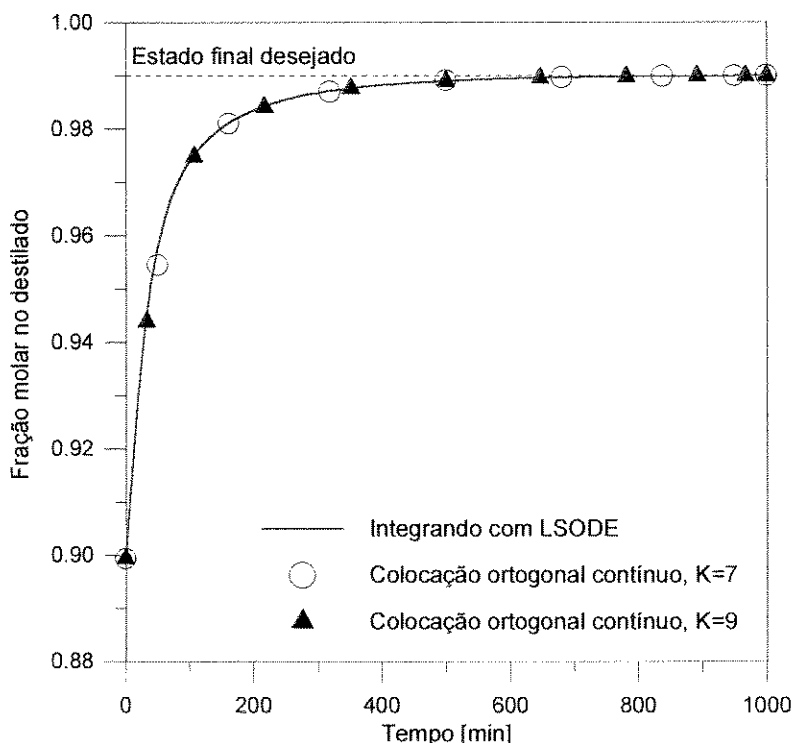


Figura 4.1-2: Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada

A Figura 4.1-1 mostra que o uso de apenas $K = 3$ ou 5 pontos de colocação internos no tempo não é suficiente para aproximar satisfatoriamente a trajetória. Os valores da variável x_D calculados pelo método de colocação ortogonal apresentam grandes diferenças em relação à trajetória calculada pelo pacote de integração. O uso de $K = 7$ pontos de colocação internos (Figura 4.1-2) já mostra resultados mais satisfatórios e com 9 pontos de colocação a aproximação é muito boa.

Para calcular a trajetória num processo não otimizado de troca de produto os resultados já mostrados, usando $K = 7$ ou 9 pontos de colocação internos, podem ser considerados como satisfatórios. Porém, num problema de controle ótimo no tempo o atraso causado pela dinâmica do "holdup" pode ter uma influência considerável. A eficiência da colocação ortogonal contínua no tempo pode ser aumentada recorrendo à colocação em elementos finitos.

As Figuras 4.1-3 e 4.1-4 mostram os resultados usando a colocação ortogonal em elementos finitos, sendo o tamanho do primeiro elemento de ($\tau = 0 - 0,1$), do elemento interno de ($\tau = 0,1 - 0,9$) e do último elemento de ($\tau = 0,9 - 1,0$), com $K/K/K = 2/2/2$, $3/2/3$ pontos de colocação internos em cada elemento finito (Figura 4.1-3) e com $K/K/K = 3/3/3$, $4/3/4$ e $5/5/5$ (Figura 4.1-4), sendo o segundo K o número de pontos internos do elemento interno.

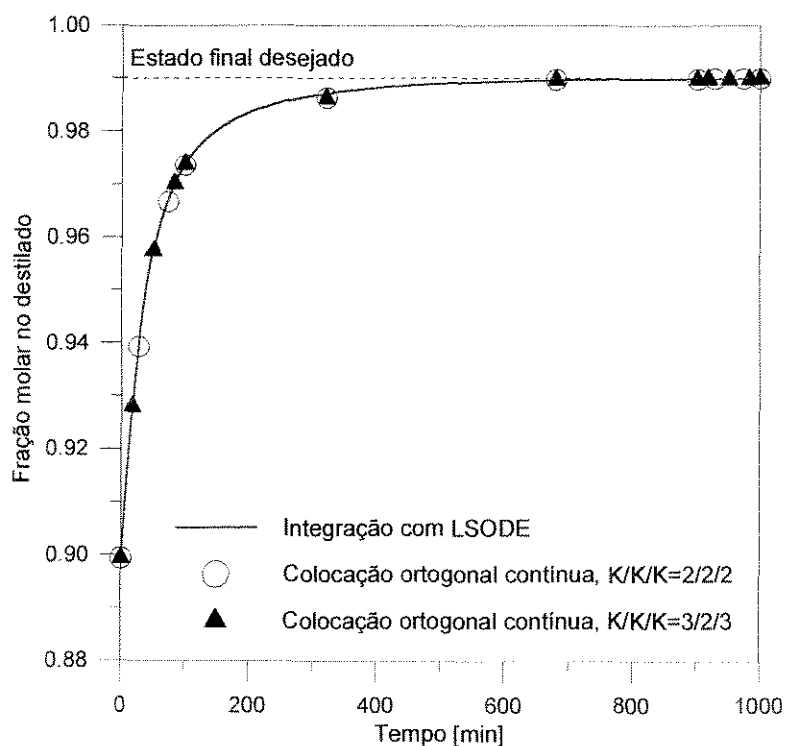


Figura 4.1-3: Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada

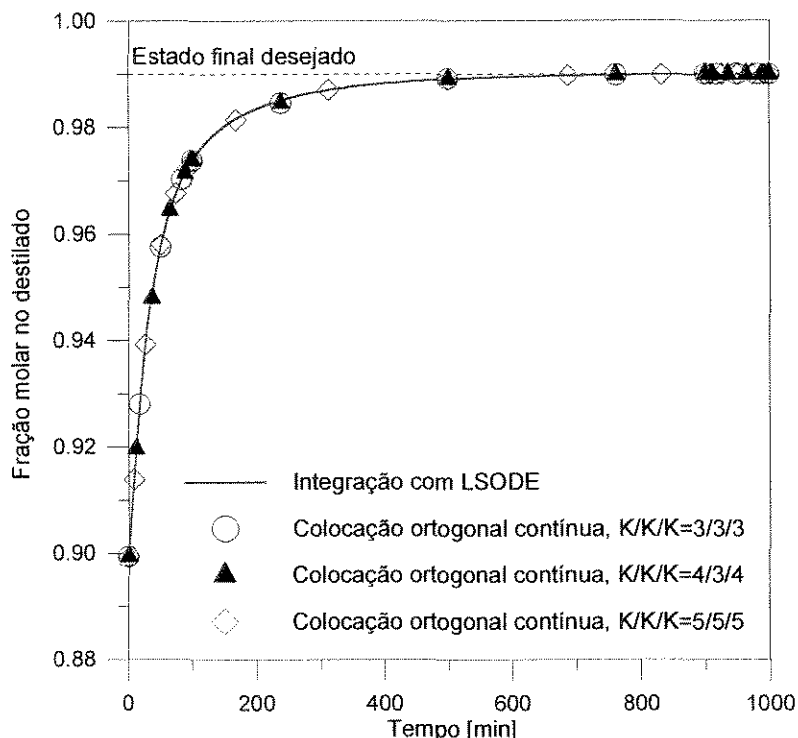


Figura 4.1-4: Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada

Usando colocação contínua por elementos finitos no tempo, com $K/K/K = 4/3/4$ pontos de colocação internos, a Figura 4.1-5 mostra a influência do número N de pontos de colocação internos, na colocação ortogonal discreta, usando o modelo de ordem reduzida. No caso da aproximação da trajetória do processo não otimizado de troca de produto, a escolha do número de N de pontos de colocação internos em cada um dos dois elementos ao longo da coluna não afeta a eficiência do método de colocação ortogonal contínua no tempo.

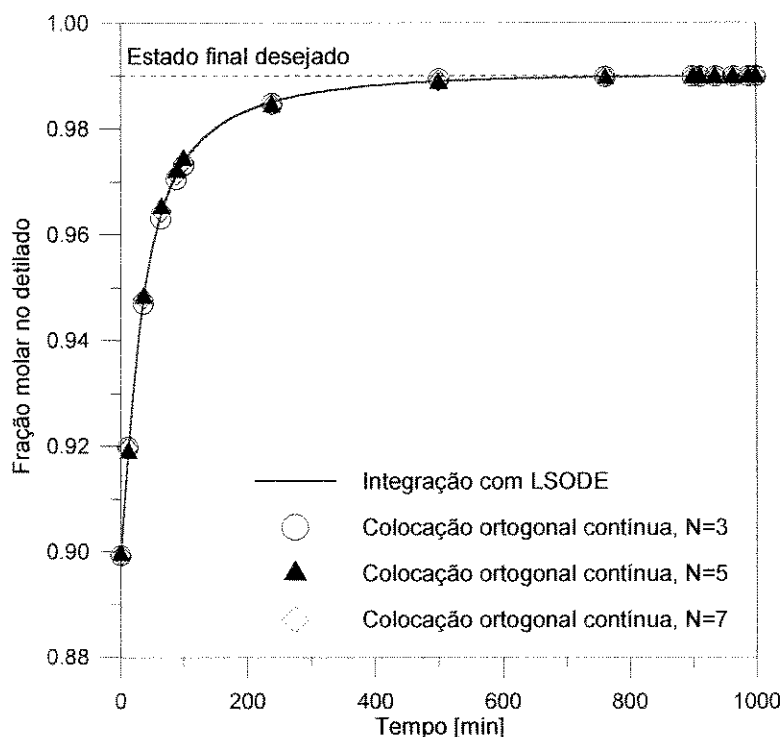


Figura 4.1-5: Cálculo da Trajetória da Fração Molar no Destilado Não-Otimizada

Os resultados obtidos neste subcapítulo mostram que a integração das equações do modelo dinâmico pelo método da colocação é confiável no seu desempenho a algoritmos convencionais como LSODE. Conclui-se então que o uso do método SQP-simultâneo no problema de controle ótimo no tempo de uma coluna de destilação é computacionalmente viável.

4.2. Eficiência do Pacote de SQP escolhido

Para confirmar os testes feitos por SCHMID & BIEGLER (1994) e comprovar a sua validade também no problema aqui tratado, o pacote de SQP desenvolvido por eles foi testado contra o método E04UCF, um pacote comercial da biblioteca NAG e baseado no algoritmo QPSOL. Como se mostra, o método SQP desenvolvido pelos autores para problemas de alta dimensão com um número de graus de liberdade relativamente pequeno e matrizes Jacobianas esparsas, método chamado de RND/SQP, quando

aplicado ao problema de controle ótimo no tempo de uma coluna de destilação, é muito mais rápido e tão exato como um método SQP convencional.

Exemplo I: No primeiro exemplo foi escolhida apenas uma variável de controle, a vazão do destilado D , e o "holdup" do líquido nos pratos foi considerado constante e igual 0,5 kmol durante todo o transiente. Para o chute inicial do parâmetro t^{fin} foi escolhido $t^{fin}_o = 9,5$ min, nos limites de 8 a 10 min. A Tabela 4.2-1 mostra a variação do custo computacional (tempo de CPU em segundos) com o número de variáveis (equações) do problema. Neste exemplo foi usado o modelo completo da coluna (com as simplificações apontadas), sendo o número de equações resultado do valor de K , o número de pontos de colocação internos no tempo. O número de variáveis inclui as variáveis de controle, ou seja, os valores de D nos pontos de colocação.

Tabela 4.2-1: Tempo de CPU [s] / Exemplo I

K	Número de Variáveis e (Equações) do Modelo	E04UCF	RND/SQP
3	168 (164)	119,35	0,70
5	252 (246)	285,10	1,20
7	336 (328)	1874,73	2,01

Exemplo II: No segundo exemplo as variáveis de controle são agora as vazões do destilado D e do vapor V , e o "holdup" do líquido ainda é considerado constante. O chute inicial do parâmetro a otimizar é $t^{fin}_o = 9$ min, nos mesmos limites que no exemplo anterior. Neste exemplo o número de variáveis, como também de graus de liberdade, aumentou devido à introdução da segunda variável de controle V . O custo computacional, dependendo do número de pontos de colocação internos no tempo, e com isso do número de variáveis e equações, é mostrado na Tabela 4.2-2.

Tabela 4.2-2: Tempo de CPU [s] / Exemplo II

K	Número de Variáveis e	E04UCF	RND/SQP
	(Equações) do Modelo		
3	171 (164)	404,05	0,62
5	257 (246)	1069,29	1,43
7	343 (328)	2417,73	10,44

As tabelas mostram o custo computacional usando os dois pacotes de SQP. Nos dois casos o resultado ótimo calculado do parâmetro a otimizar, o tempo de transição do processo de troca de produto t^{fin} , é o mesmo usando E04UCF ou RND/SQP, sendo igual a 9,780 min no Exemplo I, com apenas D como variável de controle, e a 8,994 min no Exemplo II, com duas variáveis de controle, D e V .

Este subcapítulo confirmou os resultados obtidos por SCHMID & BIEGLER (1994). O algoritmo desenvolvido pelos autores mostrou-se muito adequado para problemas de otimização de alta dimensão com muitas variáveis e restrições mas com um número de graus de liberdade relativamente pequeno e uma matriz Jacobiana esparsa. O algoritmo RND/SQP proposto por SCHMID & BIEGLER (1994), chamado de aqui em diante apenas de SQP, é o método usado nos exemplos seguintes.

4.3. Eficiência do Modelo de Ordem Reduzida usando SQP

Neste capítulo será mostrado a eficiência do modelo de ordem reduzida usando o método de otimização SQP com formulação simultânea do problema. Resultados obtidos usando o modelo de ordem reduzida são comparados com os conseguidos usando o modelo de ordem completa. Será comprovada a superioridade do uso do

modelo de ordem reduzida, tanto na velocidade como na robustez. As variáveis de controle são a vazão do destilado, D , e a do vapor, V , em todos os exemplos.

Exemplo I: O primeiro exemplo deste subcapítulo demonstra a maior velocidade de convergência no problema de controle ótimo usando o modelo de ordem reduzida. Para comparar o comportamento geral do método de otimização usando modelos de ordem completa e de ordem reduzida, a dimensão do problema foi diminuída considerando o "holdup" do líquido nos pratos constante, tal como no exemplo anterior. O chute inicial do parâmetro t^{fin} foi escolhido com $t^{fin} = 9$ min no intervalo de 8 à 10 min. O número de pontos de colocação internos, em cada um dos dois elementos ao longo da coluna, foi escolhido com $N = 7$, o que leva a uma redução do número de equações de 47,78%. A Tabela 4.3-1 mostra o número de variáveis e restrições do problema de otimização para os modelos de ordem completa e reduzida.

Tabela 4.3-1: Número de Variáveis e (Equações)

K	Modelo de ordem	Modelo de ordem
	completa	reduzida
3	171 (164)	91 (84)
5	257 (246)	137 (126)
7	343 (328)	183 (168)

O custo computacional para o uso do modelo de ordem reduzida e completo, é mostrado na Tabela 4.3-2.

Tabela 4.3-2: Tempo de CPU [s]

K	Modelo de ordem	Modelo de ordem
	completa	reduzida
3	0,62	0,49
5	1,43	1,46
7	10,44	4,27

Exemplo II: O segundo exemplo mostrará a influência do número de pontos de colocação internos ao longo da coluna (N), ou seja, da ordem do modelo reduzido. Serão comparados os resultados obtidos com o modelo de ordem completa com os obtidos usando o modelo de ordem reduzida com diferentes valores de N .

Neste exemplo a variação do "holdup" do líquido não é mais desprezada. O chute inicial escolhido para o parâmetro do tempo de transição ótimo é de $t^{mo} = 10,0$ min, no intervalo de 9,0 à 11,0 min. Os números de variáveis e de restrições do problema de otimização, usando o modelo de ordem completa e reduzida (com $N = 3, 5$ e 7 pontos de colocação internos), são mostrados na Tabela 4.3-3.

Tabela 4.3-3: Número de Variáveis e (Equações)

K	Modelo de ordem completa	Modelo de ordem reduzida ($N=7$)	Modelo de ordem reduzida ($N=5$)	Modelo de ordem reduzida ($N=3$)
3	327 (320)	167 (160)	135 (128)	103 (96)
5	491 (480)	251 (240)	203 (192)	155 (144)
7	655 (640)	335 (320)	271 (256)	207 (192)

O custo computacional associado ao modelo de ordem reduzida é mostrado na Tabela 4.3-4 e comparado com os resultados obtidos com o modelo de ordem completa. Nota-se uma vantagem ainda maior para problemas de controle ótimo no tempo de elevada dimensão.

Tabela 4.3-4: Tempo de CPU [s] e (Número de Iterações)

K	Modelo de ordem completa	Modelo de ordem reduzida ($N=7$)	Modelo de ordem reduzida ($N=5$)	Modelo de ordem reduzida ($N=3$)
3	4,77 (32)	3,78 (35)	3,13 (43)	3,49 (75)
5	41,36 (153)	27,26 (113)	19,75 (122)	13,96 (137)
7	64,01 (140)	48,15 (99)	36,09 (121)	7,90 (36)

Veja-se claramente, na seguinte Tabela 4.3-5, a influência do número N de pontos de colocação internos ao longo da coluna, usando o modelo de ordem reduzida. O erro entre o valor t^{fin} calculado com o modelo de ordem reduzida, usando $N = 7$ pontos de colocação internos ao longo da coluna, e completa é $<0,1\%$. Este erro aumenta significativamente quando o número de pontos de colocação diminui, ou seja, com a diminuição da ordem do modelo.

Tabela 4.3-5: Tempo de Transição Ótimo [min]

K	Modelo de ordem completa	Modelo de ordem reduzida ($N=7$)	Modelo de ordem reduzida ($N=5$)	Modelo de ordem reduzida ($N=3$)
3	10,080	10,075	10,003	10,576
5	9,968	9,967	9,863	10,472
7	9,941	9,940	9,830	10,446

Exemplo III: Foi comprovado, nos exemplos anteriores, que o modelo de ordem reduzida leva a um algoritmo mais rápido e, dependendo do número de pontos de colocação internos ao longo da coluna, o resultado obtido para o tempo ótimo calculado pode ser quase tão bom como o obtido usando o modelo de ordem completa. Neste exemplo aplica-se agora a colocação ortogonal contínua em elementos finitos no tempo, para a formulação simultânea do problema de otimização, sendo o tamanho do primeiro elemento de $\tau = 0$ até 0,1, do segundo de $\tau = 0,1$ até 0,9 e do último de 0,9 até 1. Em vez de procurar sempre o melhor chute inicial, como feito por SCHRÖDER & MENDES (1998), foi escolhido um chute inicial do parâmetro t^{fin} constante e igual a 10,0 min, no intervalo de 9,0 à 11,0 min. Como mostra a Tabela 4.3-6, o recurso à colocação por elementos finitos no tempo leva a um aumento considerável da dimensão do problema.

Tabela 4.3-6: Número de Variáveis e (Equações)

<i>K/K/K</i>	Modelo de ordem completa	Modelo de ordem reduzida (N=7)	Modelo de ordem reduzida (N=5)	Modelo de ordem reduzida (N=3)
2/2/2	737 (720)	377 (360)	305 (288)	233 (216)
3/2/3	901 (880)	461 (440)	373 (352)	285 (264)
3/3/3	983 (960)	503 (480)	407 (384)	311 (288)
4/3/4	1147 (1120)	587 (560)	475 (448)	363 (336)
5/5/5	1475 (1440)	755 (720)	611 (576)	467 (432)

Alem do aumento da velocidade, mostrado na Tabela 4.3-7, pode-se observar uma maior robustez do método de otimização usando o modelo de ordem reduzida. A

redução da dimensão leva à convergência mesmo, quando o modelo de ordem completa já falha.

Tabela 4.3-7: Tempo de CPU [s] e (Número de Iterações)

<i>K/K/K</i>	Modelo de ordem completa	Modelo de ordem reduzida ($N=7$)	Modelo de ordem reduzida ($N=5$)	Modelo de ordem reduzida ($N=3$)
2/2/2	37,43 (95)	38,55 (112)	31,28 (135)	22,91 (156)
3/2/3	229,04 (381)	55,04 (106)	47,97 (144)	36,59 (161)
3/3/3	209,55 (264)	119,38 (194)	26,61 (59)	26,26 (91)
4/3/4	não converge	157,49 (171)	139,33 (215)	29,85 (67)
5/5/5	não converge	505,43 (285)	170,67 (139)	137,16 (172)

Na Tabela 4.3-8 são mostrados os valores ótimos do tempo de transição, usando os modelos de ordem completa e reduzida, em função do número K de pontos de colocação internos em cada elemento finito no tempo. Nota-se ainda, tal como no exemplo anterior, que o uso de apenas $N = 3$ pontos de colocação internos ao longo da coluna pode ser não suficiente para aproximar bem os resultados obtidos com o modelo de ordem completa.

Tabela 4.3-8: Tempo de Transição Ótimo [min]

<i>K/K/K</i>	Modelo de ordem completa	Modelo de ordem reduzida (<i>N</i> =7)	Modelo de ordem reduzida (<i>N</i> =5)	Modelo de ordem reduzida (<i>N</i> =3)
2/2/2	10,062	10,063	9,937	10,581
3/2/3	9,948	9,947	9,819	10,458
3/3/3	9,932	9,931	9,818	10,441
4/3/4	-	9,996	9,881	10,506
5/5/5	-	9,945	9,829	10,456

O subcapítulo mostrou a alta eficiência do modelo de ordem reduzida na aplicação a problemas de controle ótimo no tempo de colunas de destilação. O uso do modelo de ordem reduzido não só acelera geralmente o algoritmo de otimização, mas também leva à um algoritmo mais robusto.

Neste capítulo foram usadas estimativas iniciais do parâmetro do tempo de transição à otimizar muito próximas à solução ótima e em um intervalo pequeno. Mais em diante vai ser tratado este problema, demonstrando também um aumento da robustez, na aplicação do modelo de ordem reduzida, no caso de um valor inicial longe do ótimo. Um problema sério é o fato que a velocidade e robustez maior é pago com um resultado menos exato. Os resultados obtidos com a ordem do modelo mais reduzida também são os mais longe da solução ótima.

4.4. Eficiência do Modelo de Ordem Reduzida usando IDP

Neste subcapítulo serão mostrados os resultados obtidos aplicando o método IDP ao problema de controle ótimo no tempo de uma coluna de destilação. Será comprovado que o modelo de ordem reduzida é adequado para diminuir o tempo computacional, apresentando um comportamento semelhante ao do modelo de ordem completa na convergência e na aproximação da trajetória. O uso de um chute inicial do tempo ótimo não muito perto da solução, não apresenta nenhum problema para a convergência usando IDP.

Exemplo: O algoritmo de IDP foi aplicado com o modelo completo e o modelo de ordem reduzida com $N = 5$ pontos de colocação internos em cada um dos dois elementos ao longo da coluna. Foram usadas as duas variáveis de controle, a vazão do destilado, D , e do vapor, V . Os números de pontos das grades de estado e de controle escolhidos são $M_x = 5$ e $M_u = 3$, respectivamente. O tempo foi dividido em $S = 10$ intervalos e o fator de redução é $\mu = 0,9$. O valor inicial do parâmetro a otimizar é $t^{fin}_o = 11$ min nos limites da grade de $t^{fin}_{min} = 7$ min e $t^{fin}_{max} = 15$ min. O número de 100 iterações foi escolhido para a aplicação dos dois modelos, o da ordem completa e reduzida.

A Figura 4.4-1 mostra o comportamento na convergência do algoritmo de IDP com os modelos de ordem reduzida e completa. O uso do modelo de ordem reduzida leva o algoritmo até a solução num número de iterações semelhantes ao da ordem completa.

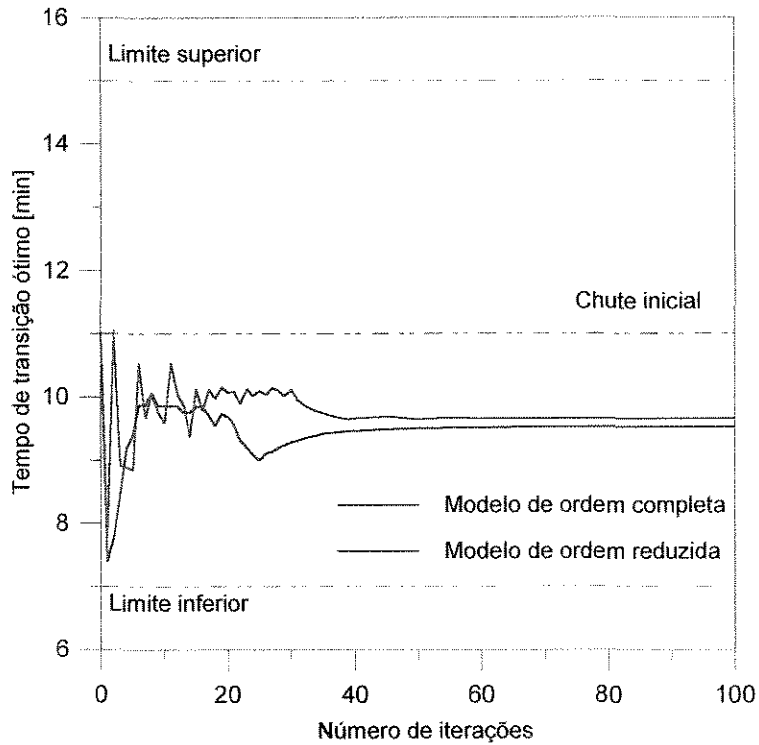


Figura 4.4-1: Convergência até o Ótimo

O comportamento do erro de continuação, descrito no subcapítulo 3.5., é mostrado na Figura 4.4-2. Também aqui pode-se notar que o comportamento observado é semelhante, sendo a trajetória calculada com o modelo de ordem reduzida tão próxima à continuidade como a calculada com o modelo de ordem completa, precisando de um número de iterações semelhantes.

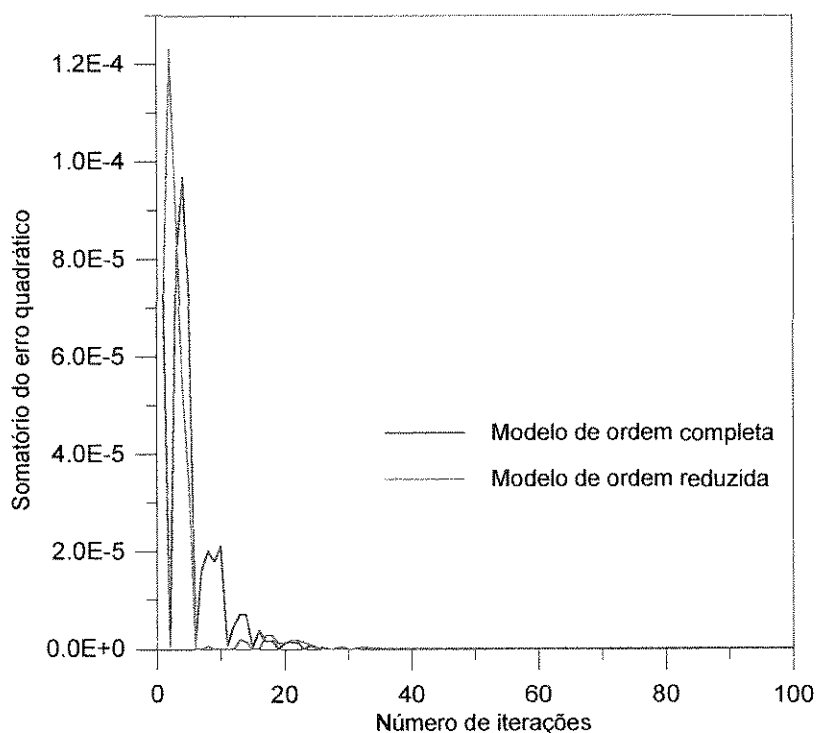


Figura 4.4-2: Erro de Continuação

As trajetórias de x_D , obtidas com o algoritmo IDP usando os modelos de ordem reduzida e completo, são mostradas na Figura 4.4-3 e comparadas com a solução obtida usando o algoritmo SQP e um chute inicial de t^{fin} de 10 min, no intervalo de 9,0 à 11,0 min. A Figura 4.4-4 mostra a estratégia de controle ótima.

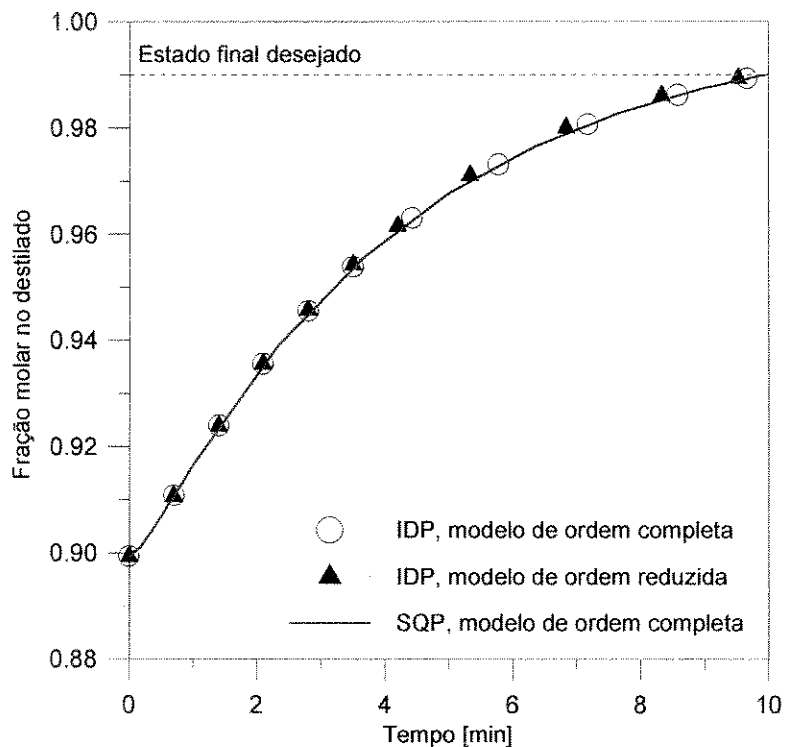


Figura 4.4-3: Trajetória Ótima da Fração Molar no Destilado obtida pelo método IDP.

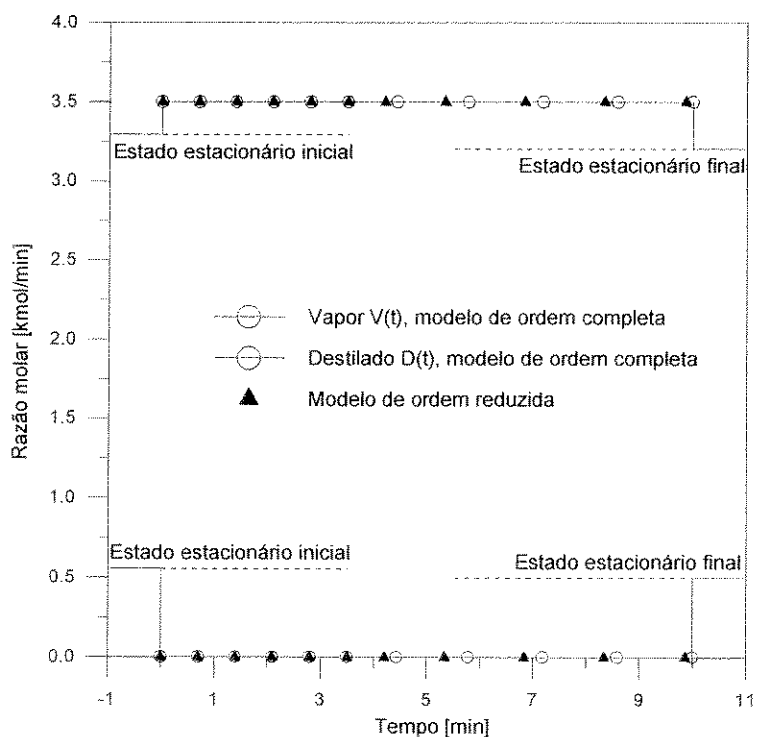


Figura 4.4-4: Estratégia de Controle Ótima, IDP

O custo computacional, aplicando IDP com 100 iterações, é mostrado na Tabela 4.4-1, comparando o uso do modelo de ordem reduzida com o modelo de ordem completa.

Tabela 4.4-1: Tempo de CPU [s]

Modelo de ordem completa	Modelo de ordem reduzida
15893,88	9905,90

Conclui-se que o modelo de ordem reduzida também pode ser aplicado com sucesso no problema de controle ótimo no tempo de uma coluna de destilação, usando o método de otimização IDP, reduzindo significativamente o tempo computacional. As Figuras 4.4-1 e 4.4-2 levam à conclusão que o número de 100 iterações é desnecessário e poderia ser facilmente diminuído, diminuindo assim o tempo computacional. Mesmo assim o método IDP é lento demais para competir com o algoritmo SQP na velocidade. A grande vantagem do IDP é a sua robustez, que leva à convergência mesmo partindo de uma estimativa inicial longe do ótimo.

4.5. Chute Inicial longe da Solução Ótima usando SQP

A convergência do algoritmo SQP não depende apenas da dimensão do problema (subcapítulo 4.3.), mas também do valor escolhido para a estimativa inicial do parâmetro do tempo de transição e do intervalo escolhido para esse tempo. O SQP é considerado um método altamente sensível com respeito à escolha adequada das estimativas iniciais (LI & WOZNY, 1996). A escolha do valor adequado do tempo de transição é obviamente uma tarefa difícil, pois a ordem de grandeza do valor ótimo do tempo de transição é desconhecida. O presente subcapítulo vai-se concentrar na

influência do chute inicial do parâmetro tempo de transição e do intervalo em que ele está contido.

Exemplo: O problema de controle ótimo no tempo de uma coluna de destilação é resolvido com o algoritmo SQP usando a colocação ortogonal discreta com $N=7$ pontos de colocação internos em cada elemento ao longo da coluna e usando colocação ortogonal contínua em elementos finitos no tempo com diferentes valores do número de pontos de colocação K em cada elemento finito no tempo. Os chutes iniciais usados foram 10, 11, 41 e 250 min nos intervalos de 9 à 11, 7 à 15, 1 à 80 e 1 à 500 min respectivamente (Tabela 4.5-1).

A Tabela 4.5-1 mostra claramente as dificuldades associadas à convergência do método. Sendo o problema de alta dimensão e partindo-se de um ponto inicial longe da solução, o algoritmo não consegue mais convergir, mesmo usando o modelo de ordem reduzida.

Tabela 4.5-1: Tempo de CPU [s] e (Número de Iterações)

Chute inicial de t^m no intervalo de seus limites [min]	$K/K/K = 5/5/5$	$K/K/K = 4/3/4$	$K/K/K = 3/3/3$
9,...,10,...,11	505,43 (285)	157,49 (171)	119,12 (194)
7,...,11,...,15	não converge	345,83 (386)	178,05 (296)
1,...,41,...,80	não converge	não converge	não converge
1,...,250,...,500	não converge	não converge	não converge

Partindo-se de estimativas iniciais muito longe da solução ótima e precisando-se de resultados muito exatos, o que obriga a usar muitos pontos de colocação no tempo

(K), o uso do modelo de ordem reduzida pode não ser suficiente para compensar o aumento da região de busca e garantir a convergência.

Para um problema de dimensão alta, causada pela necessidade de uma boa aproximação da trajetória de x_D , e partindo-se de um chute inicial do tempo de transição longe da solução, pode-se dizer que o método SQP-simultâneo usado não resolve satisfatoriamente o problema de controle ótimo no tempo de uma coluna de destilação, nem mesmo usando modelos de ordem reduzida. Será preciso usar um algoritmo mais robusto para inicializar o método, garantindo um valor próximo da solução ótima, para poder partir com um algoritmo mais sofisticado como o SQP desde este ponto, melhorando o resultado até um ponto suficientemente exato. Em princípio isso pode ser conseguido usando um método misto de dois algoritmos.

4.6. Aplicação Mista, IDP-SQP

O IDP tem a vantagem de ser muito robusto e convergir até uma região mais perto da solução em poucas iterações, mesmo partindo de um chute inicial longe da solução e usando uma grade ampla. A principal desvantagem do algoritmo é o seu alto custo computacional, devido a repetitiva solução do modelo dinâmico nos numerosos intervalos de tempo e para vários pontos da grade do estado usando todas as combinações de controle permitido. No exemplo apresentado no subcapítulo 4.4. o algoritmo IDP precisa integrar o modelo dinâmico 135 vezes em cada iteração. Neste capítulo será demonstrado uma aplicação mista, usando o algoritmo IDP numa forma mais reduzida, com apenas poucos iterações, para inicializar o problema de controle ótimo no tempo de uma coluna de destilação, fornecendo o chute inicial para o método SQP.

Exemplo: No exemplo, já apresentado por SCHRÖDER & MENDES (1999), foi usado o modelo de ordem reduzida com $N = 3$ pontos de colocação internos em cada elemento ao longo da coluna. O número de intervalos no tempo do IDP foi reduzido à

apenas $S = 3$, de tamanho variável. Foram usadas as duas variáveis de controle, a vazão do destilado, D , e a do vapor, V . Os números de pontos das grades de estado e de controle escolhidos foram $M_x = 5$ e $M_u = 3$, respectivamente. Partindo de um chute inicial do tempo de transição de 41 min e uma grade de 1 min à 80 min foi usado o IDP com 3 iterações e um fator de redução de $\mu = 0,55$ para aperfeiçoar o chute inicial. A solução "ótima" obtida com o algoritmo IDP, t^{fin*} , foi usada então como chute inicial para o método SQP. Para determinar o intervalo do parâmetro, foram escolhidos $t^{fin}_{min} = t^{fin*} - 0,5$ min e $t^{fin}_{max} = t^{fin*} + 0,5$ min, sempre usando o valor mais próximo da forma 9,0; 9,5; 10,0; As configurações do modelo foram $K/K/K = 4/3/4$ pontos de colocação e $N = 7$. Também foram consideradas as duas variáveis de controle D e V .

A Figura 4.6-1 mostra a trajetória da fração molar no destilado x_D , assim calculada. O tempo de transição "ótimo" calculado, com um custo computacional de 151,71 s, pelo algoritmo de inicialização IDP é $t^{fin*} = 10,1837$ min. Partindo deste novo chute inicial no intervalo de 9,5 à 10,5 min, o algoritmo SQP levou mais 297 iterações ou seja, 275,33 s, para convergir até uma solução ótima com $t^{fin*} = 9,9947$ min.

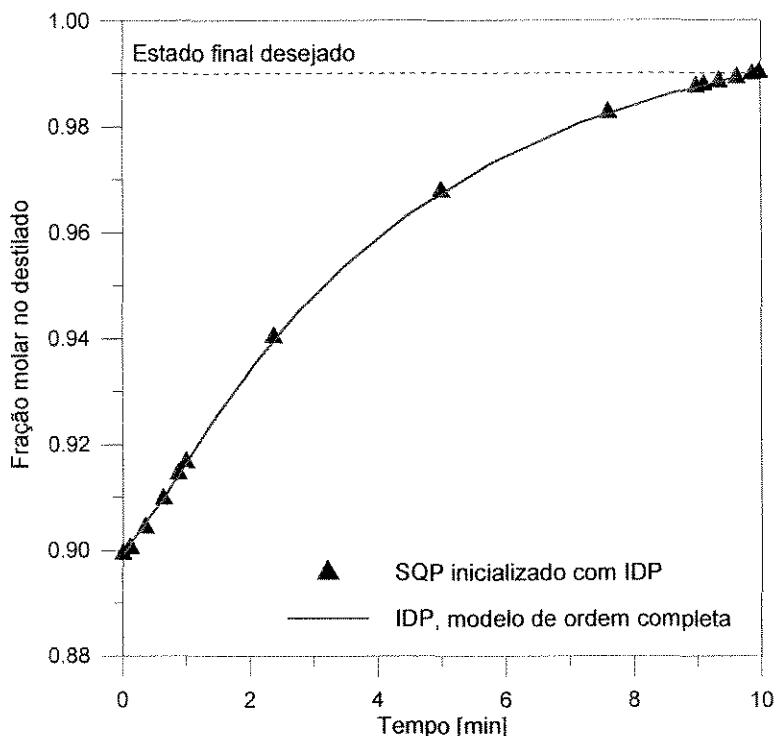


Figura 4.6-1: Trajetória Ótima da Fração Molar no Destilado obtida pelo método misto IDP-SQP.

O método misto IDP-SQP, usando o algoritmo IDP para inicializar e o algoritmo SQP para finalizar o problema de controle ótimo, melhorou a robustez, sendo capaz agora de convergir até a solução ótima, mesmo partindo de uma estimativa inicial do parâmetro t_{fin}^m de 41 min com um intervalo inicial de (1 à 80) min. O tempo computacional do método IDP-SQP, de 427,04 s, pode ser considerado aceitável, comparando-se com os resultados obtidos usando apenas SQP (não convergência) ou usando apenas o IDP (Tabela 4.4-1).

4.7. Aplicação Mista, SQP-SQP

Neste subcapítulo é desenvolvido um método de inicialização do problema de otimização baseado no próprio método SQP, e por isso chamado de método SQP-SQP. A idéia básica é a de resolver o problema de controle ótimo usando um modelo da

coluna de baixa dimensão combinado com o método SQP. Obtém-se deste modo, partindo de uma estimativa inicial de má qualidade, um valor "otimizado" $t_{\text{Inicialização}}^{fin*}$, o qual será usado como chute inicial, $t_{\text{Finalização}}^{fin}$, para uma segunda etapa de otimização. Tal como no subcapítulo anterior, durante a segunda etapa de otimização, a finalização da otimização, o método SQP foi usado em combinação com um modelo da coluna com $N = 7$ pontos de colocação discreta internos por elemento e com $K/K/K = 4/3/4$ pontos de colocação internos por elementos finitos no tempo. D e V foram usadas como variáveis de controle.

Exemplos de Inicialização: A dimensão final do modelo da coluna varia obviamente com a estratégia de colocação usada tanto na colocação ortogonal discreta (redução da ordem) como na colocação ortogonal no tempo. Como mostra a Tabela 4.7-1, na etapa de inicialização foi usado um modelo obtido por colocação ortogonal global no tempo.

Tabela 4.7-1: Número de Variáveis e de Equações;
(Etapa de Inicialização)

Configuração dos métodos de colocação ortogonal	Número de variáveis	Número de equações
$K = 3$		
$N = 5$	135	128
$K = 3$		
$N = 3$	103	96
$K = 1$		
$N = 5$	67	64
$K = 1$		
$N = 3$	51	48

A Tabela 4.7-2 mostra o custo de computação, usando diferentes número de pontos de colocação para o modelo reduzido e a formulação simultânea, em relação ao chute inicial e intervalo do parâmetro do tempo de transição.

Tabela 4.7-2: Tempo de CPU [s] e (Número de Iterações);
(Etapa de Inicialização)

Chute inicial de t^{fin} e valores limites [min]	$K = 3$ $N = 5$	$K = 3$ $N = 3$	$K = 1$ $N = 5$	$K = 1$ $N = 3$
7,...,11,...,15	57,48 (889)	3,18 (80)	2,92 (127)	3,17 (195)
1,...,41,...,80	não converge	não converge	8,79 (363)	18,53 (1150)
1,...,250,...,500	não converge	não converge	14,73 (635)	21,51 (1381)
$t^{fin*}_{Inicialização}$	10,00250	10,57594	9,671455	10,09042

A Tabela 4.7-2 mostra ainda o valor ótimo de t^{fin} , calculado na Etapa de Inicialização e que será usado como chute inicial na Etapa subsequente de Finalização.

Como se vê pela Tabela 4.7-2, o método SQP simultâneo usando um modelo com $K = 1$ e $N = 5$ permite obter um valor de $t^{fin}_{Finalização}$ suficientemente próximo do ótimo. Repare-se que $K = 1$ equivale a aproximar a resposta dinâmica da coluna a degrau por uma curva de 2ª ordem. Os resultados da Tabela 4.7-2 mostram que, usando um modelo de dimensão suficientemente baixa, sem contudo sacrificar as características básicas do comportamento do processo, é possível, na Etapa de Inicialização, chegar a soluções adequadas, mesmo quando o chute de t^{fin} é muito afastado da solução ótima e o seu intervalo muito amplo.

Exemplos de Finalização: A Tabela 4.7-3 mostra o custo computacional da Fase de Finalização e o tempo de transição ótimo t^{fin*} , em função da configuração usada na Etapa de Inicialização.

Tabela 4.7-3: Tempo de CPU [s] e (Número de Iterações);
Tempo de Transição Ótimo, t^{fin*}
(Etapa de Finalização)

Configuração de Inicialização	$t^{fin*}_{Inicialização}$ [min]	Intervalo [min]	Tempo de CPU [s] e (número de iterações)	t^{fin*} [min]
$K = 3$				
$N = 5$	10,00250	(9,5 - 10,5)	143,13 (156)	9,99524
$K = 3$				
$N = 3$	10,57594	(10,0 - 11,0)	214,66 (235)	10,00163
$K = 1$				
$N = 5$	9,671455	(9,0 - 10,0)	34,13 (31)	9,994188
$K = 1$				
$N = 3$	10,09042	(9,5 - 10,5)	244,02 (225)	9,995065

O uso do método SQP para a autoinicialização leva à um método muito mais robusto e ainda mais rápido. Os resultados deste capítulo mostram que o número de pontos de colocação internos dos dois elementos ao longo da coluna apenas pode ser reduzido até $N = 5$, considerado também por CARVALHO (1996) como o número mínimo de pontos de colocação internos para aproximar satisfatoriamente uma coluna prato-a-prato no estado estacionário e transiente.

4.8. Análise dos Resultados

Neste subcapítulo é feita uma análise dos resultado obtidos, comparando os métodos com respeito à velocidade de convergência e robustez quanto iniciados com

um chute inicial longe da solução ótima. Mostra-se que o método de autoinicialização do próprio SQP leva à um método tão robusto como o usando IDP, mas muito mais rápido. No último subcapítulo serão feitos alguns comentários sobre a trajetória ótima.

4.8.1. Custo Computacional e Robustez

Ao longo desta trabalho foram desenvolvidos e apresentados vários métodos para resolver o problema de controle ótimo no tempo de uma coluna de destilação. A sensibilidade do algoritmo SQP leva a sérios problemas no caso em que a solução ótima é completamente desconhecida e portanto, o chute inicial escolhido inadequado. Numa faixa pequena, perto da solução ótima, como apresentado no subcapítulo 4.3. para um chute inicial de $t^{fin}_o = 10$ min no intervalo de 9 à 11 min, com $K = 4$ pontos de colocação internos nos elementos extremos e $K = 3$ no elemento interno, o SQP levou 157,49 s para convergir. Com um chute inicial de $t^{fin} = 11$ min no intervalo de 7 à 15 min o SQP já precisou de mais de 345 s para convergir. Com um chute inicial ainda mais longe da solução ótima, $t^{fin}_o = 41$ min no intervalo de 1 à 80 min, já não convergiu mais. Os métodos inicializados, apresentados nos subcapítulos 4.6 e 4.7, ainda levam ao ótimo, mesmo partindo de estimativas iniciais muito longe da solução ótima.

A Figura 4.8-1 apresenta esquematicamente os custos de computação obtidos para um chute inicial de $t^{fin} = 11$ min no intervalo de 7 à 15 min, já mostrando a superioridade do método de SQP inicializado sobre o SQP não inicializado. O IDP mostrou-se inadequado para resolver o problema de controle ótimo no tempo, resultando num custo computacional de 9905,90 s, considerado alto demais.

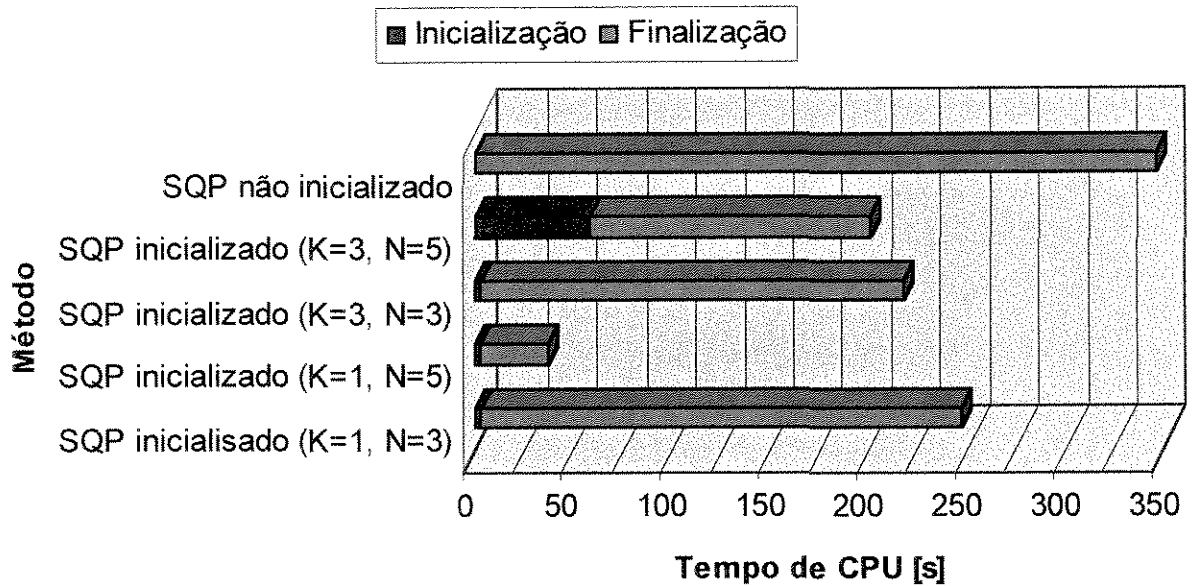


Figura 4.8-1: Custo Computacional com Chute Inicial de $t^{fin} = (7, \dots, 11, \dots, 15)$ min

Partindo de um chute inicial de $t^{fin} = 41$ min no intervalo de 1 à 80 min, não só o SQP mas também o SQP inicializado com $K = 3$, $N = 5$ e $K = 3$, $N = 3$ pontos de colocação internos não converge mais. O método de SQP inicializado com IDP convergiu num tempo razoável, mas não competitivo com os tempos conseguidos usando o próprio SQP para a autoinicialização com $K = 1$, $N = 5$ ou $K = 1$, $N = 3$ pontos de colocação internos (Figura 4.8-2). O método da autoinicialização do SQP pode resultar até 10 vezes mais rápido do que o uso do IDP na inicialização.

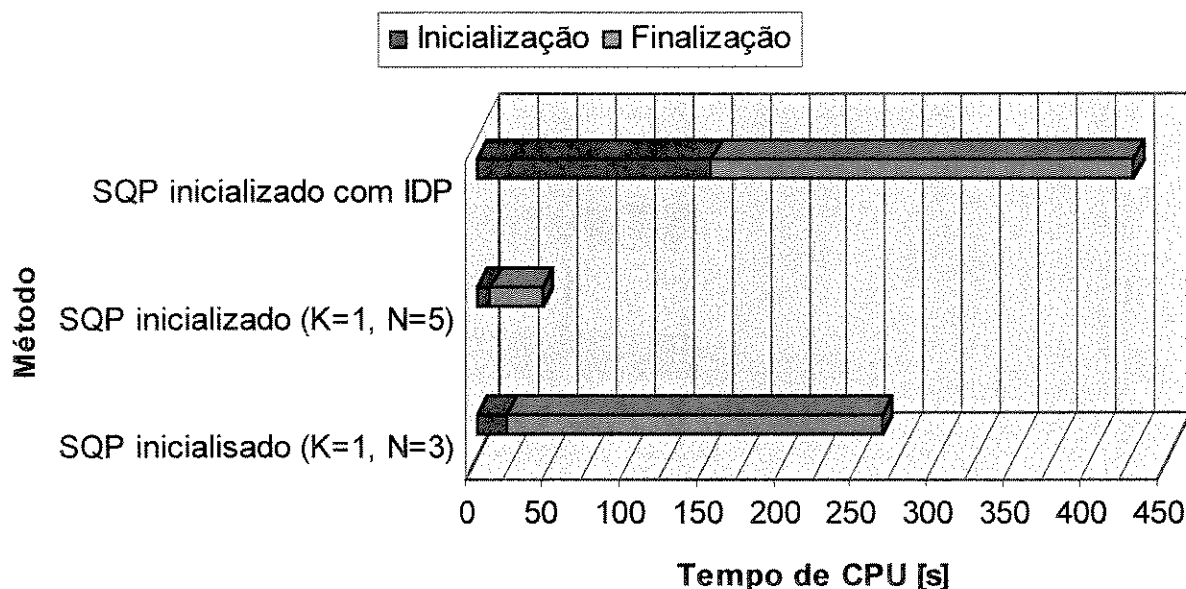


Figura 4.8-2: Custo Computacional com Chute Inicial de $t^{fin} = (1, \dots, 41, \dots, 80)$ min

A autoinicialização do método SQP ainda leva à convergência partindo de um chute inicial extremamente longe da solução ótima, como mostra a Figura 4.8-3 para um chute inicial do parâmetro $t^{fin} = 250$ min e um intervalo de 1 à 500 min. Mostra-se aqui claramente que o método de SQP inicializado usando um modelo com $K = 1$ e $N = 5$ pontos de colocação internos é adequado para encontrar uma solução ótima do problema de controle da coluna de destilação com a configuração dada.

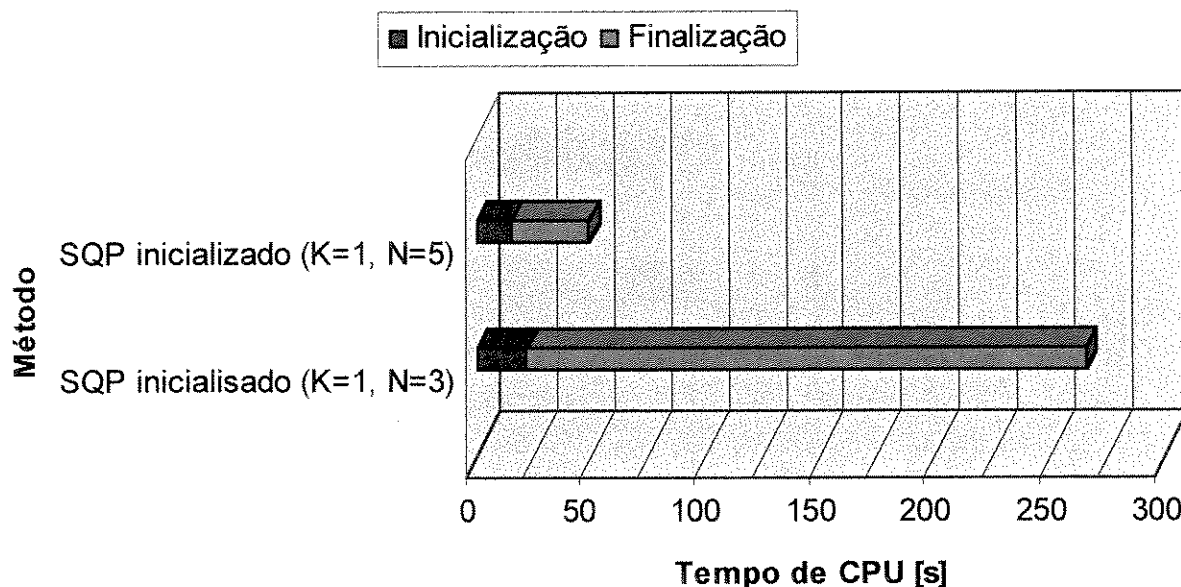


Figura 4.8-3: Custo Computacional com Chute Inicial de $t^{in} = (1, \dots, 250, \dots, 500)$ min

4.8.2. Processo Transiente Otimizado

A eficiência de um processo de troca de produto otimizado pode ser vista claramente comparando o tempo ótimo com o tempo que precisaria este processo se não fosse otimizado. O processo de troca de produto clássico, levando no início do processo de troca de produto todos os parâmetros do processo aos seus valores no novo estado estacionário desejado, leva aproximadamente 800 min para chegar ao seu novo estado estacionário, como mostram as figuras do subcapítulo 4.1. A aplicação de uma estratégia ótima reduz este tempo de uma maneira significativa, precisando aproximadamente 10 min. Neste trabalho foram desenvolvidos métodos adequados para calcular o tempo ótimo usando um modelo de ordem reduzida, resultando em um custo computacional satisfatoriamente baixo e com um tempo ótimo calculado quase tão exato como usando um modelo completo da coluna.

Comparando os resultados dos capítulos anteriores, chega-se à conclusão que o uso das duas variáveis de controle D e V em vez de só D não resulta em uma redução muito maior do tempo de transição. Cabe ao aplicador deste método avaliar se o custo

maior de um controle ótimo no tempo, usando também a variável do vapor, pode ser justificado com o valor economizado pela diminuição dos gastos para energia, matéria prima e lixo produzido. O trabalho mostrou que o método de SQP autoinicializado, na formulação simultânea, e usando um modelo de ordem reduzida, resulta em um algoritmo adequado para resolver este problema, mesmo usando as duas variáveis de controle. Outra observação importante é o fato que a dinâmica do "holdup" não pode ser desprezada, recomendando-se não desprezar o "holdup" nem mesmo durante a inicialização, pelo alto desvio causado. A Figura 4.8-4 mostra as trajetórias ótimas calculadas com SQP da formulação simultânea com $K/K/K = 4/3/4$ pontos de colocação internos e o modelo de ordem reduzida com $N = 7$ pontos de colocação internos. Comparam-se as trajetórias calculadas usando uma ou duas variáveis de controle e com "holdup" considerado constante, sendo D e V as variáveis de controle.

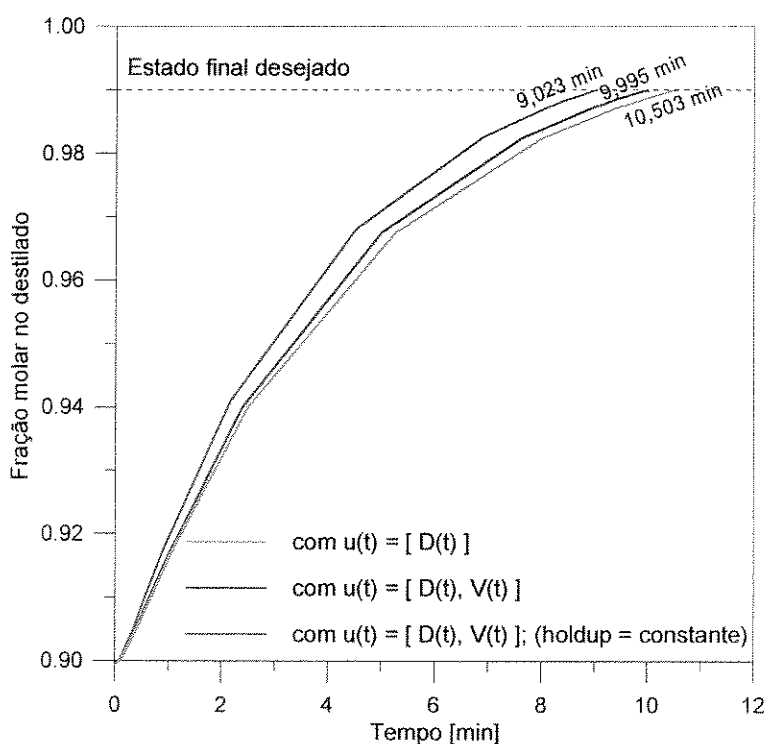


Figura 4.8-4: Trajetória Ótima da Fração Molar no Destilado e Tempo Ótimo

O uso de apenas D como variável de controle resulta em um tempo de transição um pouco maior do que usando também a vazão do vapor V . A diferença entre as duas

soluções leva à conclusão que não é recomendável usar na inicialização apenas uma variável de controle, caso na continuação se queira usar as duas variáveis. O desvio causado desprezando uma variável de controle pode causar um chute inicial inadequado. Ainda pior é o desvio causado quando se despreza a dinâmica do "holdup", o qual chega aqui a cerca de 1 min, longe demais para garantir um algoritmo robusto. Daí recomenda-se para a inicialização apenas a diminuição dos números de pontos de colocação, como feito neste trabalho, sem diminuir o número de variáveis de controle nem desprezar a dinâmica do "holdup".

A Figura 4.8-5 mostra a estratégia de controle ótima das duas variáveis, a vazão do destilado, D , e do vapor, V , que leva a coluna de destilação do seu estado estacionário inicial até um estado estacionário final desejado, num tempo ótimo. A estratégia foi calculada usando SQP da formulação simultânea com $K/K/K = 4/3/4$ pontos de colocação internos nos elementos finitos no tempo e $N = 7$ pontos de colocação internos ao longo da coluna.

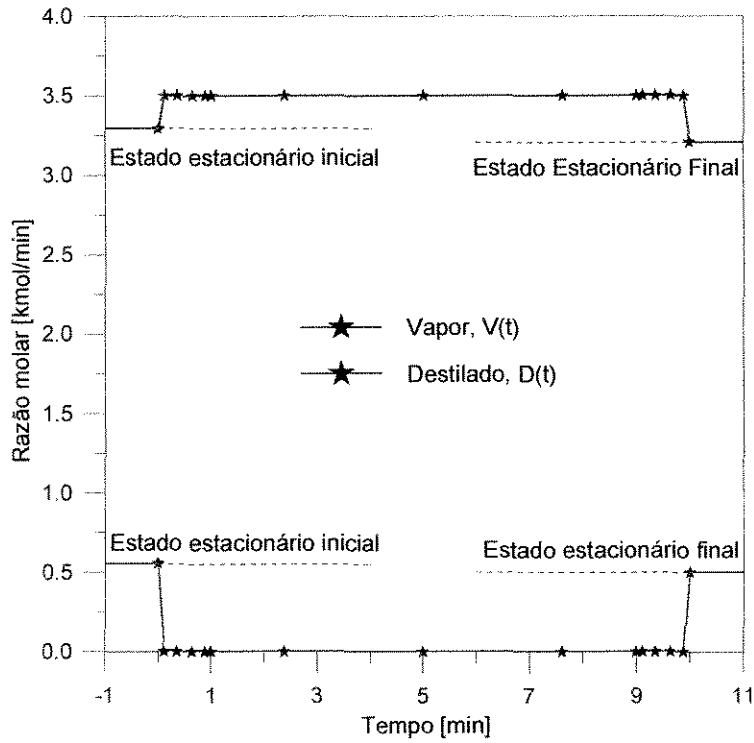


Figura 4.8-5: Estratégia de Controle Ótima ($K/K/K = 4/3/4$, $N = 7$).

Estes resultados são obviamente muito semelhantes aos de Figura 4.4-4. Eventuais diferenças são atribuídas à discretização de $u(t)$ na aproximação NLP.

5. Conclusões e Sugestões

O trabalho tinha como objetivo desenvolver um método robusto e rápido para otimizar o tempo de transição de um processo de troca de produto de uma coluna de destilação binária. Foi possível desenvolver um algoritmo extremamente robusto, convergindo mesmo partindo de um chute inicial muito longe da solução ótima, sem perder em velocidade, muito pelo contrário, os resultados mostram um algoritmo ainda mais rápido.

O método consiste no uso de um modelo de ordem reduzida da coluna, baseado na aplicação da colocação ortogonal discreta, e num algoritmo SQP simultâneo inicializado, discretizando o modelo dinâmico pela aplicação da colocação ortogonal contínua. O método de inicialização é um algoritmo de otimização mais robusto, baseado num modelo bastante simples da coluna, que gera um chute inicial perto da solução ótima para o método de otimização principal. O método de inicialização é capaz de convergir mesmo com chutes iniciais bastante longe do ótimo e contidos num intervalo extremamente amplo.

O uso do método de colocação ortogonal, tanto para reduzir a ordem do modelo como para a discretização da parte dinâmica do processo, mostrou-se altamente vantajoso, viabilizando o algoritmo de autoinicialização do próprio método SQP. Para o problema de controle ótimo aqui tratado, o método de autoinicialização do SQP mostrou-se tão robusto como o uso de IDP para a inicialização, mas muito mais rápido.

Embora o trabalho mostrasse apenas a aplicação deste método a uma coluna de destilação binária, em processo de troca de produto, acredita-se que seja possível aplicar o método também a colunas de destilação multicomponente, em processos de batelada ou de arranque, como também em outros problemas de controle. Uma sugestão para futuros trabalhos, além de estudar a aplicação deste método em outros problemas, é de aumentar a velocidade do método IDP. O uso de um método de escolha casual dos valores de controle, o uso de programação paralela ou a aplicação

da colocação ortogonal em vez do uso do pacote de integração, poderiam levar a um aumento da velocidade deste método.

6. Referências Bibliográficas

ASCHER, U.M., MATHEIJ, R.M. & RUSSEL, R.D., (1988), *Numerical Solution of Boundary Value Problems of Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ.

BAIER, H., SEEßELBERG, C. & SPECHT, B., (1994), *Optimierung in der Strukturmechanik*, Vieweg, Braunschweig/Wiesbaden.

BELLMAN, R., (1957), *Dynamic Programming*, Princeton University Press, Princeton, New Jersey.

BIEGLER, L.T., (1984), *Solution of Dynamic Optimization Problems by Successive Quadratic Programming and Orthogonal Collocation*, *Comput. Chem. Engng.*, 8, pp. 243-248.

BIEGLER, L.T., (1988), *On the Simultaneous Solution and Optimization of Large Scale Engineering Systems*, *Comput. Chem. Engng.*, 12, pp. 357-369.

BIEGLER, L.T., (1992), *Optimization Strategies for Complex Process Models*, *Adv. Chem. Engng.*, 18, pp. 197-256.

BIEGLER, L.T., (1994), *Process Optimization*, Personal Script.

BIEGLER, L.T. & CUTHRELL, J.E., (1985), *Improved Infeasible Path Optimization for Sequential Modular Simulators, Part II., The Optimization Algorithm*, *Comput. Chem. Engng.*, 9, pp. 257-267.

BIEGLER, L.T., GROSSMANN, I.E. & WESTERBERG, A.W., (1985), *A Note of Approximation Techniques used for Process Optimization*, *Comput. Chem. Engng.*, 9, pp. 201-206.

BOJKOV,B. & LUUS,R., (1992), *Use of Random Admissible Values for Control in Iterative Dynamic Programming*, Ind. Eng. Chem. Res., 31, pp. 1308-1314.

BOJKOV,B. & LUUS,R., (1994a), *Application of Iterative Dynamic Programming to Time Optimal Control*, Trans IChemE, 72, pp. 72-80.

BOJKOV,B. & LUUS,R., (1994b), *Time Optimal Control by Iterative Dynamic Programming*, Ind. Eng. Chem. Res., 33, pp. 1486-1492.

BRENAN,K.E., CAMPBELL,S.E. & PETZOLD,L.R., (1989), *Numerical Solution of Initial Value Problems in Differential-Algebraic Equations*, North-Holland, New York.

BRONSTEIN,I.N., SEMENDJAJEW,K.A., MUSIOL,G. & MÜHLIG,H., (1995), *Taschenbuch der Mathematik*, Verlag Harry Deutsch, Thun, Frankfurt am Main.

BRYSON,A.E. & HO,Y.-C., (1975), *Applied Optimal Control*, J.Willey, New York.

CARLOFF,R. & REICHERT,K.H., (1994), *Zeitoptimierte Steuerung des Anfahr- und Übergangsverhaltens der Kontinuierlichen Polymerisation von Methylmethacrylat*, Chem.-Ing.-Tech., 66, pp. 89-92.

CARVALHO,C.R.G., (1996), *Aplicação dos Polinômios Ortogonais Discretos à Modelagem e Simulação de Sistemas de Separação por Estágios*, Universidade Estadual de Campinas, Tese (Mestrado), Campinas.

CERVANTES,A. & BIEGLER,L.T., (1998), *Large-Scale DAE Optimization using a Simultaneous NLP Formulation*, AIChE Journal, 44, pp. 1038-1050.

CHIMOWITZ,E.H., MACCHIETTO,S. & ANDERSON,T.F., (1985), *Dynamic Multicomponent Distillation using Local Thermodynamic Models*, Chem. Engng. Sci., 40, pp. 1974-1978.

CHO,Y.S. & JOSEPH,B., (1983a), *Reduced-Order Steady-State and Dynamic Models for Separation Processes, Part I., Development of the Model Reduction Procedure*, AIChE Journal, 29, pp. 261-269.

CHO,Y.S. & JOSEPH,B., (1983b), *Reduced-Order Steady-State and Dynamic Models for Separation Processes, Part II., Application to Nonlinear Multicomponent Systems*, AIChE Journal, 29, pp. 270-276.

CUTHRELL,J.E. & BIEGLER,L.T., (1987), *On the Optimization of Differential-Algebraic Process Systems*, AIChE Journal, 33, pp. 1257-1270.

CUTHRELL,J.E. & BIEGLER,L.T., (1989), *Simultaneous Optimization and Solution Methods for Batch Reactor Control Profiles*, Comput. Chem. Engng., 13, pp. 49-62.

DADEBO,S.A. & McAULEY,K.B., (1995a), *Dynamic Optimization of Constrained Chemical Engineering Problems using Dynamic Programming*, Comput. Chem. Engng., 19, pp. 513-525.

DADEBO,S.A. & McAULEY,K.B., (1995b), *A Simultaneous Iterative Solution Technique for Time-Optimal Control using Dynamic Programming*, Ind. Eng. Chem. Res., 34, pp. 2077-2083.

DeTREMBLAY,M. & LUUS,R., (1989), *Optimization of Non-Steady-State Operation of Reactors*, Canad. Journal Chem. Engng., 67, pp.494-502.

EDGAR,T.F. & HIMMELBLAU,D.M., (1989), *Optimization of Chemical Processes*, McGraw-Hill.

FIKAR,M., LATIFI,M.A., CREFF,Y., (1998), *Dynamic Optimization of a Highly Nonlinear Depropanizer*, 13th International Congress of Chemical and Process Engineering, E8.1., [1207], Praga

FINLAYSON,B.A., (1972), *The Method of Weighted Residuals and Variational Principles*, Academic Press, New York.

FINLAYSON,B.A., (1980), *Nonlinear Analysis in Chemical Engineering*, McGraw-Hill.

FLENDER,M., LI,P., WOZNY,G. & FIEG.G., (1997), *Modellgestützte Entwicklung einer Zeitoptimalen Produktwechselstrategie für Rektifikationskolonnen*, Chem.-Ing.-Tech., 69, pp. 827-830.

FÖLLINGER,O., (1988), *Optimierung dynamischer Systeme: Eine Einführung für Ingenieure*, R.Oldenbourg Verlag, München; Wien.

FRIEDLANDER,A., (1994), *Elementos de Programação Não-Linear*, Editora da Unicamp, Campinas, SP.

GAL,T. (Hrsg), (1991), *Grundlagen des Operations Research, Part 1*, Springer-Verlag, Berlin; Heidelberg.

GUPTA,Y.P., (1995), *Semiexhaustive Search for Solving Nonlinear Optimal Control Problems*, Ind. Eng. Chem. Res., 34, pp. 3878-3884.

HARTIG,F. & KEIL,F.J., (1993), *Large-Scale Spherical Fixed Bed Reactors; Modeling and Optimization*, Ind. Eng. Chem. Res., 32, pp. 424-437.

HARWELL SUBROUTINE LIBRARY, Atomic Energy Research Establishment, Harwell U.K.

HINDMARSH,A.C., (1980), *LSODE and LSODI, Two new Initial Value Ordinary Differential Equation Solvers*, ACM-SIGMUM Newsletter, 15, pp.10-11.

HOFFER,E. & LUNDERSTÄDT,R., (1975), *Numerische Methoden der Optimierung*, R.Oldenbourg Verlag, München.

IMSL SOFTWARE LIBRARY, (1982).

KEIL,F.J., (1996), *Optimierung Verfahrenstechnischer Prozesse*, Chem.-Ing.-Tech., 68, pp. 639-650.

KRUSE,C., FIEG,G., WOZNY,G. & JEROMIN,L., (1995), *Algorithmus zur Reduzierung des Zeitbedarfs bei Anfahrvorgängen an Rektifikationskolonnen*, Chem.-Ing.-Tech., 67, pp. 458-461.

LI,P. & WOZNY,G. (1996), *Analyse und Anwendung dynamischer Optimierungsalgorithmen für Batch-Prozesse.*, Chem.-Ing.-Tech., (68) 4/96, pp. 382-389.

LOGSDON,J.S. & BIEGLER,L.T., (1989), *Accurate Solution of Differential-Algebraic Optimization Problems*, Ind. Eng. Chem. Res., 28, pp. 1628-1639.

LOGSDON,J.S. & BIEGLER,L.T., (1993), *A Relaxed Reduced Space SQP Strategy for Dynamic Optimization Problems*, Comput. Chem. Engng., 17, pp. 367-372.

LUCIA,A. & XU,J., (1990), *Chemical Process Optimization using Newton-Like Methods*, Comput. Chem. Engng., Vol.14, No.2, pp. 119-138.

LUUS,R., (1990a), *Application of Dynamic Programming to High-Dimensional Non-Linear Optimal Control Problems*, Int. J. Control, 52, pp. 234-250.

LUUS,R., (1990b), *Optimal Control by Dynamic Programming using Systematic Reduction in Grid Size*, Int. J. Control, 51, pp. 995-1013.

LUUS,R., (1996), *Numerical Convergence Properties of Iterative Dynamic Programming when Applied to High Dimensional Systems*, Trans. IChemE., 74, Part A, pp. 55-62.

LUUS,R. & ROSEN,O., (1991), *Application of Dynamic Programming to Final State Constrained Optimal Control Problems*, Ind. Eng. Chem. Res., 30, pp. 1525-1530.

LUUS,R. & SMITH,S.G., (1991), *Application of Dynamic Programming to High-Dimensional Systems Described by Difference Equations*, Chem. Eng. Technol., 14, pp. 122-126.

NAG FORTRAN LIBRARY MANUAL, (1991), Mark 15

OSBORNE,A., (1971), *The Calculations of Unsteady State Multicomponent Distillation using Partial Differential Equations*, AIChE Journal, 17, pp. 696-703

PAPAGEORGIOU,M., (1991), *Optimierung: Statische, dynamische, stochastische Verfahren für die Anwendung*, R.Oldenbourg Verlag, München.

PONTRYAGIN,L.S., BOLTYANSKII,V.G., GAMKRELIDZE,R.V. & MISCHENKO,E.F., (1962), *The Mathematical Theory of Optimal Processes*, Wiley (Interscience), New York.

ROSEN,O. & LUUS,R., (1989), *Sensitivity of Optimal Control to Final State Specification by a Combined Continuation and Nonlinear Programming Approach*, Chem. Eng. Sci., 44, pp. 2527-2534.

ROSEN,O. & LUUS,R., (1991), *Evaluation of Gradients for Piecewise Constant Optimal Control*, Comput. Chem. Engng., 15, pp. 273-281.

- SCHITTKOWSKI, K., (1985), *NLPQL: A FORTRAN Subroutine solving Constrained Nonlinear Programming Problems*, Annals of Operations Research, 5, pp. 485-500.
- SCHMID, C. & BIEGLER, L.T., (1994), *Quadratic Programming Methods for Reduced Hessian SQP*, Comput. Chem. Engng., 18, pp. 817-832.
- SCHRÖDER, A. & MENDES, M.J., (1998), *Time Optimal Control of Distillation Columns*, 13th CHISA Congress, E5.4, [1024], Prague.
- SCHRÖDER, A. & MENDES, M.J., (1999), *Time Optimal Control of Distillation Columns: A mixed IDP-SQP Approach*, ESCAPE-9 Congress, P053, Budapest (accepted).
- SEFERLIS, P. & HRYMAK, A.N., (1994a), *Optimization of Distillation Units using Collocation Models*, AIChE Journal, 40, pp. 813-825.
- SEFERLIS, P. & HRYMAK, A.N., (1994b), *Adaptive Collocation on Finite Elements Models for the Optimization of Multistage Distillation Units*, Chem. Eng. Sci., 49, pp. 1369-1382.
- SKOGESTAD, S. & MORARI, M., (1988), *Understanding the Dynamic Behavior of Distillation Columns*, Ind. Eng. Chem. Res., 27, pp. 1848-1862.
- STEWART, W.E., LEVIEN, K.L. & MORARI, M., (1985), *Simulation of Fractionation by Orthogonal Collocation*, Chem. Eng. Sci., 40, pp. 409-421.
- STØREN, S. & HERTZBERG, T., (1994), *Local Thermodynamic Models applied in Dynamic Process Simulation: A Simplified Approach*, Trans. IChemE., 72, pp. 395-401.
- SWARTZ, C.L.E. & STEWART, W.E., (1986), *A Collocation Approach to Distillation Column Design*, AIChE Journal, 32, pp. 1832-1838.

TANARTKIT,P. & BIEGLER,L.T., (1995), *Stable Decomposition for Dynamic Optimization*, Ind. Eng. Chem. Res., 34, pp. 1253-1266.

VASANTHARAJAN,S. & BIEGLER,L.T., (1988), *Large-Scale Decomposition for Successive Quadratic Programming*, Comput. Chem. Engng., 12, pp. 1087-1101.

VASANTHARAJAN,S. & BIEGLER,L.T., (1990), *Simultaneous Strategies for Optimization of Differential-Algebraic Systems with Enforcement of Error Criteria*, Comput. Chem. Engng, 14, pp. 1083-1100.

VASANTHARAJAN,S., VISWANATHAN,J. & BIEGLER,L.T., (1990), *Reduced Successive Quadratic Programming Implementation for Large-Scale Optimization Problems with Smaller Degrees of Freedom*, Comput. Chem. Engng., 14, pp. 907-915.

VILLADSEN,J. & MICHELSEN,M.L., (1978), *Solution of Differential Equation Models by Polynomial Approximation*, Prentice-Hall, Englewood Cliffs, N.J.

Anexos

Modelo de Ordem Reduzida de Coluna de Destilação.

São apresentadas aqui todas as equações algébricas do modelo dinâmico de ordem reduzida da coluna de destilação, tal como usado com o método SQP-simultâneo.

Refervedor ($s^S = -1$):

$$x_B^i = x^i (s^{S_0} - 1)$$

$$y_B^i = y^i (s^{S_0} - 1)$$

$$\frac{dM_B}{dt} = 0$$

$$L^i(s^{S_0}) = B^i + V^i$$

$$0 = L^i(s^{S_0})x^i(s^{S_0}) - V^i y_B^i - B^i x_B^i - \frac{M_B}{t^m} \sum_{j=0}^{K+1} \dot{\phi}^*(\tau^i) x^j (s^{S_0} - 1)$$

Ponto de colocação extremo inferior do Elemento 1 (Primeiro Prato da seção de esgotamento) ($s^{S_0} = 0$):

$$0 = \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{S_0}) L^i(s^{S_j}) - \frac{1}{t^m} \sum_{j=0}^{K+1} \dot{\phi}^*(\tau^i) M^j(s^{S_0})$$

$$0 = \sum_{j=0}^{N+1} \Gamma_j(s^{S_0} + 1) L^i(s^{S_j}) \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{S_0}) x^i(s^{S_j}) \\ + V^i [y^i(s^{S_0} - 1) - y^i(s^{S_0})] - \frac{M^i(s^{S_0})}{t^{nn}} \sum_{j=0}^{K+1} \phi^{\bullet j}(\tau^i) x^j(s^{S_0})$$

Pontos de colocação internos do Elemento 1 (s^{S_1}, \dots, s^{S_N}):

$$0 = \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{S_k}) L^i(s^{S_j}) - \frac{1}{t^{nn}} \sum_{j=0}^{K+1} \phi^{\bullet j}(\tau^i) M^j(s^{S_k})$$

$$0 = \sum_{j=0}^{N+1} \Gamma_j(s^{S_k} + 1) L^i(s^{S_j}) \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{S_k}) x^i(s^{S_j}) \\ - V^i \sum_{j=0}^{N+1} \nabla \Gamma_j(s^{S_k}) y^i(s^{S_j}) - \frac{M^i(s^{S_k})}{t^{nn}} \sum_{j=0}^{K+1} \phi^{\bullet j}(\tau^i) x^j(s^{S_k})$$

com

$$(k = 1, \dots, N)$$

Ponto de colocação externo superior do Elemento 1 (Prato abaixo do prato de alimentação) ($s^{S_{N+1}} = N^S - 1$):

$$0 = L^i(s^{S_{N+1}} + 1) - L^i(s^{S_{N+1}}) - \frac{1}{t^{nn}} \sum_{j=0}^{K+1} \phi^{\bullet j}(\tau^i) M^j(s^{S_{N+1}})$$

$$0 = L^i(s^{S_{N+1}} + 1) [x^i(s^{S_{N+1}} + 1) - x^i(s^{S_{N+1}})] \\ - V^i \sum_{j=0}^{N+1} \nabla \Gamma_j(s^{S_{N+1}}) y^i(s^{S_j}) - \frac{M^i(s^{S_{N+1}})}{t^{nn}} \sum_{j=0}^{K+1} \phi^{\bullet j}(\tau^i) x^j(s^{S_{N+1}})$$

Prato de alimentação ($s^S = N^S, s^R = -1$):

$$0 = F + L^i(s^{R_0}) - L^i(s^{R_0} - 1) - \frac{1}{t^m} \sum_{j=0}^{K+1} \dot{\phi}(\tau^i)^j M^j(s^{R_0} - 1)$$

$$0 = L^i(s^{R_0}) [x^i(s^{R_0}) - x^i(s^{R_0} - 1)] + V^i [y^i(s^{S_{N+1}}) - y^i(s^{R_0} - 1)] \\ + F [z_F - x^i(s^{R_0} - 1)] - \frac{M^i(s^{R_0} - 1)}{t^m} \sum_{j=0}^{K+1} \dot{\phi}(\tau^i)^j x^j(s^{R_0} - 1)$$

com

$$(s^{R_0} - 1) = (s^{S_{N+1}} + 1)$$

Ponto de colocação extremo inferior do Elemento 2 (Prato acima do prato de alimentação) ($s^{R_0} = 0$):

$$0 = \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{R_0}) L^i(s^{R_j}) - \frac{1}{t^m} \sum_{j=0}^{K+1} \dot{\phi}(\tau^i)^j M^j(s^{R_0})$$

$$0 = \sum_{j=0}^{N+1} \Gamma_j(s^{R_0} + 1) L^i(s^{R_j}) \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{R_0}) x^j(s^{R_j}) \\ + V^i [y^i(s^{R_0} - 1) - y^i(s^{R_0})] - \frac{M^i(s^{R_0})}{t^m} \sum_{j=0}^{K+1} \dot{\phi}(\tau^i)^j x^j(s^{R_0})$$

Pontos de colocação internos do Elemento 2 (s^R_1, \dots, s^R_N):

$$0 = \sum_{j=0}^{N+1} \Delta \Gamma_j(s^{R_k}) L^i(s^{R_j}) - \frac{1}{t^m} \sum_{j=0}^{K+1} \dot{\phi}(\tau^i)^j M^j(s^{R_k})$$

$$0 = \sum_{j=0}^{N+1} \Gamma_j (s^R_k + 1) L^i (s^R_j) \sum_{j=0}^{N+1} \Delta \Gamma_j (s^R_k) x^i (s^R_j) \\ - V^i \sum_{j=0}^{N+1} \nabla \Gamma_j (s^R_k) y^i (s^R_j) - \frac{M^i (s^R_k)}{t^{\eta_n}} \sum_{j=0}^{K+1} \dot{\phi}^j (\tau^i) x^j (s^R_k)$$

onde

$$(k = 1, \dots, N)$$

Ponto de colocação extremo superior do Elemento 2 (Prato abaixo do condensador) ($s^R_{N+1} = N^R - 1$):

$$0 = L^i (s^R_{N+1} + 1) - L^i (s^R_{N+1}) - \frac{1}{t^{\eta_n}} \sum_{j=0}^{K+1} \dot{\phi}^j (\tau^i) M^j (s^R_{N+1})$$

$$0 = L^i (s^R_{N+1} + 1) [x^i (s^R_{N+1} + 1) - x^i (s^R_{N+1})] \\ - V^i \sum_{j=0}^{N+1} \nabla \Gamma_j (s^R_{N+1}) y^i (s^R_j) - \frac{M^i (s^R_{N+1})}{t^{\eta_n}} \sum_{j=0}^{K+1} \dot{\phi}^j (\tau^i) x^j (s^R_{N+1})$$

Condensador total ($s^R_{N+1} + 1$):

$$\frac{dM_D}{dt} = 0$$

$$D^i = V^i - L^i (s^R_{N+1} + 1)$$

$$0 = V^i [y^i (s^R_{N+1}) - x^i (s^R_{N+1} + 1)] - \frac{M_D}{t^{\eta_n}} \sum_{j=0}^{K+1} \dot{\phi}^j (\tau^i) x^j (s^R_{N+1} + 1)$$