



Universidade Federal de Uberlândia
Faculdade de Engenharia Química
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA



**Avaliação do Desempenho do Algoritmo de Evolução
Diferencial na Solução de Problemas de Otimização Dinâmica**

Cleuton Luis Nascentes

Uberlândia – MG

2013



Universidade Federal de Uberlândia
Faculdade de Engenharia Química
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA



Avaliação do Desempenho do Algoritmo de Evolução Diferencial na Solução de Problemas de Otimização Dinâmica

Cleuton Luis Nascentes

Orientadora: Valéria Viana Murata

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Química da Universidade Federal de Uberlândia como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Química

Uberlândia – MG

2013

FICHA CATALOGRÁFICA

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG - Brasil

N244a Nascentes, Cleuton Luis, 1982-
2013 Avaliação do desempenho do algoritmo de evolução diferencial na
solução de problemas de otimização dinâmica / Cleuton Luis Nascentes. -
2013.
139 f. : il.

Orientadora: Valéria Viana Murata.
Dissertação (mestrado) – Universidade Federal de Uberlândia, Pro-
grama de Pós-Graduação em Engenharia Química.
Inclui bibliografia.

1. Engenharia química - Teses. 2. Otimização matemática - Teses. 3.
Algoritmos - Teses. I. Murata, Valéria Viana. II. Universidade Federal
de Uberlândia. Programa de Pós-Graduação em Engenharia Química.
III. Título.

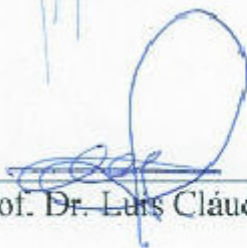
CDU: 66.0

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA DA UNIVERSIDADE FEDERAL DE UBERLÂNDIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO DO TÍTULO DE MESTRE EM ENGENHARIA QUÍMICA, EM 25 DE MARÇO DE 2013.

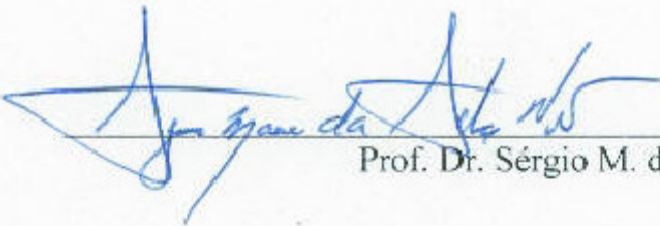
BANCA EXAMINADORA:



Prof. Dra. Valéria Viana Murata
Orientadora (FEQUI/UFU)



Prof. Dr. Luis Cláudio Oliveira Lopes
FEQUI/UFU



Prof. Dr. Sérgio M. da Silva Neiro
FEQUI/UFU



Prof.ª Dr.ª Caliane Bastos Borba Costa
DEQ/UFScar

Agradecimentos

Primeiramente agradeço a Deus pela oportunidade de ter desenvolvido esse trabalho. O meu agradecimento pelo apoio incondicional de meus pais, Diva e Isaías, minhas irmãs Clésia, Claudiene e Rejane, meu irmão Cléver .

Agradecimento especial a minha orientadora Valéria Viana Murata pela dedicação e compreensão durante o desenvolvimento do trabalho. Ao professor Fran Sérgio Lobato pela grande ajuda e disponibilidade no atendimento às dúvidas sobre o tema do trabalho.

Agradecimento aos Professores Sergio Neiro, Luís Cláudio Lopes e Caliane Bastos Borba Costa pela atenção dispensada em relação às correções da dissertação.

Aos amigos Dirney, Diovanina, Rodrigo, Rubens, Mateus, Henrique e a todos aqueles que contribuíram de alguma forma para que esse trabalho fosse realizado.

Agradecimento final a CAPES pela concessão da bolsa de estudo.

“Nunca deixe que lhe digam que não vale
a pena acreditar no sonho que se tem”

Renato Russo

SUMÁRIO

LISTA DE FIGURAS.....	iii
LISTA DE TABELAS.....	vi
LISTA DE ABREVIATURAS.....	viii
RESUMO.....	ix
ABSTRACT	x
INTRODUÇÃO	1
ASPECTOS GERAIS DA OTIMIZAÇÃO DINÂMICA DE PROCESSOS	5
2.1. Introdução.....	5
2.2. Abordagem Algébrico-Diferencial na Solução de Problemas de Otimização Dinâmica.....	6
2.2.1 <i>Índice Diferencial de Equações Algébrico-Diferenciais (EAD)</i>	7
2.3. Métodos Clássicos	7
2.4. Métodos Indiretos	10
2.4.1. <i>Arcos Singulares</i>	20
2.4.2. <i>Função Identificadora de Fases (FIF)</i>	20
2.5. Métodos Diretos.....	21
2.5.1. <i>Métodos Sequenciais</i>	22
2.5.2. <i>Métodos Simultâneos</i>	25
2.6. Programação Dinâmica Iterativa	27
2.7. Métodos Híbridos	31
2.8. Métodos de Otimização Naturais.....	33
REVISÃO DE MÉTODOS DE OTIMIZAÇÃO NATURAIS	35
3.1. Introdução.....	35
3.2. Algoritmos Genéticos.....	37
3.2.1. <i>Seleção</i>	38
3.2.2. <i>Recombinação (Cruzamento)</i>	39
3.2.3. <i>Mutação</i>	40
3.3. Colônia de Formigas.....	43
3.4. Evolução Diferencial	46
3.4.1. <i>Mutação</i>	47
3.4.2. <i>Cruzamento</i>	49
3.4.3. <i>Seleção</i>	51
3.4.4. <i>Estratégias da Evolução diferencial</i>	52
3.4.5. <i>Algoritmo de Evolução Diferencial</i>	53

3.5. Técnicas de manipulação de restrições utilizadas em Algoritmos Evolutivos....	56
3.5.1. Penalidade Estática	57
3.5.2. Penalidade Dinâmica	57
3.5.3. Penalidade Recozimento.....	58
3.5.4. Penalidade Adaptativa	59
3.5.5. Penalidade Coevolutiva.....	59
ANÁLISE DO DESEMPENHO DO ALGORITMO DE EVOLUÇÃO	
DIFERENCIAL NA SOLUÇÃO DE PROBLEMAS DE OTIMIZAÇÃO	
DINÂMICA.....	61
4.1. Caso 1: Reator Batelada com reação consecutiva	62
4.2. Caso 2: Problema de Jacobson et al. (1970) com formulação de índice diferencial 5	68
4.3. Caso 3: Problema de Jacobson et al (1970) com formulação de índice diferencial 3	72
4.4. Caso 4: Problema de tempo final livre: O problema do carro	74
4.5. Caso 5: Problema com restrição de fim na variável de estado considerando tempo final fixo e tempo final livre: Produção de etanol	78
4.6. Caso 6. Problema com três restrições de trajetória na variável de estado e tempo final fixo: Produção de Penicilina	85
4.7. Análise de sensibilidade paramétrica do algoritmo de Evolução Diferencial para uma população com número fixo de indivíduos.....	91
4.7.1. Mistura de catalisadores em um reator PFR	92
4.7.2. Reator Batelada com reação consecutiva	102
CONCLUSÕES E SUGESTÕES	112
5.1 Conclusões.....	112
5.2. Sugestões para trabalhos futuros	114
REFERÊNCIAS BIBLIOGRÁFICAS	115

LISTA DE FIGURAS

1.1 Classificação dos Métodos de Otimização	3
3.1 Representação de um cromossomo com n genes de 5 bits cada.....	37
3.2 Cruzamento de ponto único.....	39
3.3 Cruzamento de dois pontos	39
3.4 Fundamentação Teórica do Algoritmo de Evolução Diferencial	48
3.5 Representação do Processo de Cruzamento Binomial	49
3.6 Representação do Cruzamento Exponencial	51
4.1 Perfil da variável de controle $T(K)$ para 5 estágios de controle($FO=0,609472$)	67
4.2 Perfil da variável de controle $T(K)$ para 10 estágios de controle ($FO=0,610117$) ..	67
4.3 Perfil da variável de controle $T(K)$ para 15 estágios de controle ($FO=0,610347$) ..	67
4.4 Perfil da variável de controle $T(K)$ para 20 estágios de controle ($FO=0,610466$) ..	67
4.5 Perfil das variáveis de estado $C1$ e $C2$ do problema do Reator Batelada com reação consecutiva para $FO=0,610466$	68
4.6 Elementos de Controle discretizados.....	69
4.7 Perfil da variável de controle para o Problema de Jacobson et al (1970) com formulação de índice 5.	71
4.8 Perfis das variáveis de estado para o Problema de Jacobson et al (1970) com formulação de índice 5.	71
4.9 Perfil da variável de controle para o Problema de Jacobson et al. (1970) com formulação de índice 3	73
4.10 Perfis das variáveis de estado para o Problema de Jacobson et al (1970) com formulação de índice 3.	74
4.11 Perfil da variável de estado x (distância percorrida).	76
4.12 Perfil da variável de estado v (velocidade).....	77
4.13 Perfil da variável de controle a (aceleração).	77

4.14 Perfis da variável de controle u (tempo final de 63 horas).....	83
4.15 Perfis da variável de controle u (tempo final livre).....	83
4.16 Perfis das variáveis de estado X,S,P e V (tempo final fixo de 63 horas).	84
4.17 Perfis das variáveis de estado X,S,P e V (tempo final livre).....	84
4.18 Perfis da variável de controle u para o problema de Produção de Penicilina.....	89
4.19 Perfis das variáveis de estado X, P, S e V	89
4.20 Função Objetivo x Número de Gerações para $F=0,2$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR.....	93
4.21 Função Objetivo x Número de Gerações para $F=0,8$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR.....	93
4.22 Função Objetivo x Número de Gerações para $F=1,2$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR.....	94
4.23 Função Objetivo x Número de Gerações para $F=2,0$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR.....	94
4.24 Função Objetivo x Probabilidade de Cruzamento para $F=0,2$; $NG=50$ e 250 para o problema de mistura de catalisadores em um reator PFR	95
4.25 Função Objetivo x Probabilidade de Cruzamento para $F=0,8$; $NG=50$ e 250 para o problema de mistura de catalisadores em um reator PFR	96
4.26 Função Objetivo x Probabilidade de Cruzamento para $F=1,2$; $NG=50$ e 250 para o problema de mistura de catalisadores em um reator PFR	96
4.27 Função Objetivo x Probabilidade de Cruzamento para $F=2,0$; $NG=50$ e 250 para o problema de mistura de catalisadores em um reator PFR	97
4.28 Função Objetivo x Fator de Perturbação para $P_c=0,3$; $NG=50$ para o problema de mistura de catalisadores em um reator PFR	98
4.29 Função Objetivo x Fator de Perturbação para $P_c=0,8$; $NG=50$ para o problema de mistura de catalisadores em um reator PFR	99
4.30 Função Objetivo x Fator de Perturbação para $P_c=0,3$; $NG=250$ para o problema de mistura de catalisadores em um reator PFR	99
4.31 Função Objetivo x Fator de Perturbação para $P_c=0,8$; $NG=250$ para o problema de mistura de catalisadores em um reator PFR	100
4.32 Perfil da variável de controle (u) para o problema de mistura de catalisadores em um reator PFR.....	101

4.33 Perfis das variáveis de estado x_1 e x_2 para o problema de mistura de catalisadores em um reator PFR.....	101
4.34 Função Objetivo x Número de Gerações para $F=0,2$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva	103
4.35 Função Objetivo x Número de Gerações para $F=0,8$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva	103
4.36 Função Objetivo x Número de Gerações para $F=1,2$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva	104
4.37 Função Objetivo x Número de Gerações para $F=2,0$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva	104
4.38 Função Objetivo x Probabilidade de Cruzamento para $F=0,2$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva	105
4.39 Função Objetivo x Probabilidade de Cruzamento para $F=0,8$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva	106
4.40 Função Objetivo x Probabilidade de Cruzamento para $F=1,2$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva	106
4.41 Função Objetivo x Probabilidade de Cruzamento para $F=2,0$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva	107
4.42 Função Objetivo x Fator de Perturbação para $P_c=0,3$ e $NG=50$ para o problema do Reator Batelada com reação consecutiva	108
4.43 Função Objetivo x Fator de Perturbação para $P_c=0,8$ e $NG=50$ para o problema do Reator Batelada com reação consecutiva	108
4.44 Função Objetivo x Fator de Perturbação para $P_c=0,3$ e $NG=250$ para o problema do Reator Batelada com reação consecutiva	109
4.45 Função Objetivo x Fator de Perturbação para $P_c=0,8$ e $NG=250$ para o problema do Reator Batelada com reação consecutiva	109
4.46 Perfil da variável de controle (T) para o problema do Reator Batelada com reação consecutiva	111
4.47 Perfis das variáveis de estado (x_1 , x_2) para o problema do Reator Batelada com reação consecutiva.....	111

LISTA DE TABELAS

3.1 Representação das estratégias da Evolução Diferencial.....	53
4.1 Parâmetros do algoritmo de Evolução Diferencial utilizados na solução do Caso 1: Reator Batelada com reação consecutiva	64
4.2 Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 5 elementos de controle na solução do Caso 1: Reator Batelada com reação consecutiva	64
4.3 Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 10 elementos de controle na solução do Caso 1: Reator Batelada com reação consecutiva	65
4.4 Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 15 elementos de controle na solução do Caso 1: Reator Batelada com reação consecutiva	65
4.5 Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 20 elementos de controle na solução do Caso 1: Reator Batelada com reação consecutiva	66
4.6 Parâmetros utilizados nas soluções dos problemas de Jacobson et al (1970)	70
4.7 Resultados obtidos para o estudo de caso 2: problema de Jacobson com índice diferencial 5	70
4.8 Resultados obtidos para o estudo de caso 3: problema de Jacobson com índice diferencial 3	72
4.9 Parâmetros utilizados na solução do problema do carro	75
4.10 Resultados obtidos para diferentes sementes e diferentes números de gerações: problema do carro	76
4.11 Parâmetros utilizados na solução do problema: Produção de etanol.....	80
4.12 Resultados obtidos para diferentes sementes e diferentes números de gerações(considerando tempo final fixo)	81
4.13 Resultados obtidos para diferentes sementes e diferentes números de gerações (considerando tempo final livre e Ngerações=100 e 200).....	82
4.14 Resultados obtidos para diferentes sementes e diferentes números de gerações (considerando tempo final livre e Ngerações=300 e 400).....	82

4.15 Resultados obtidos por diferentes técnicas de solução (considerando tempo final fixo e tempo final livre).....	85
4.16 Parâmetros utilizados na solução do problema da Produção de Penicilina.....	87
4.17 Resultados obtidos para diferentes sementes e diferentes números de gerações na solução do problema de Produção de Penicilina	88
4.18 Resultados obtidos por diferentes técnicas de solução para o problema de produção de penicilina.....	90
4.19 Valores de Função Objetivo obtidos por diferentes técnicas para o problema de mistura de catalisadores em um reator PFR	101
4.20 Resultados obtidos por diferentes técnicas de solução: Reator batelada com reação consecutiva	110

LISTA DE ABREVIATURAS

COLDAE – *Collocation Differential Algebric Equation Method*

DIRCOL – *Direct Collocation Method*

EAD – Equações Algébrico-Diferenciais

EDO – Equações Diferenciais Ordinárias

ED – Evolução Diferencial

FIF – Função Identificadora de Fases

KKT – Karush-Kunh-Tucher

LJ – Luus- Jakola

NLP – Programação não linear

NG – Número de Gerações

NP – Número da População

PDI – Programação Dinâmica Iterativa

POD – Problemas de Otimização Dinâmica

PVC – Problema de Valor no Contorno

QP – Programação Quadrática

SQP – Programação Quadrática Seqüencial

RESUMO

A solução de problemas de otimização dinâmica (POD) envolve a determinação do perfil de uma ou mais de uma variável de controle que minimize ou maximize um determinado índice de desempenho. Com exceção de poucos problemas em que métodos analíticos são aplicáveis, a maioria dos problemas de otimização dinâmica é não linear e necessita de uma abordagem de solução numérica para a obtenção do perfil da variável de controle. Métodos clássicos de solução numérica fundamentados no princípio do cálculo diferencial e integral utilizam da informação das derivadas da Função Objetivo e das restrições para a determinação de uma solução ótima. Métodos heurísticos de otimização natural apresentam como característica comum o caráter aleatório na busca de uma solução ótima. São métodos que não utilizam a informação das derivadas da Função Objetivo e das restrições.

O algoritmo de Evolução Diferencial desenvolvido por Storn e Price (1995) é um algoritmo de simples implementação que requer apenas a definição de alguns parâmetros. Dentre algumas das características que o algoritmo de Evolução Diferencial apresenta está a capacidade de manipular Funções Objetivo multimodais não lineares e não diferenciáveis, além de ser um método que possui alta probabilidade de encontrar o ótimo global.

No presente trabalho são apresentadas soluções para diferentes problemas de otimização dinâmica. A utilização do algoritmo de Evolução Diferencial simplifica a solução de POD evitando a necessidade, por exemplo, da solução de problemas de valor no contorno ou mesmo manipulações algébricas expressivas na solução de problemas singulares. Os resultados obtidos por Evolução Diferencial são comparados com resultados obtidos por outros autores utilizando tanto métodos clássicos quanto métodos naturais. A análise dos resultados obtidos demonstra a eficiência da aplicação do algoritmo de Evolução diferencial na solução de diferentes problemas de Otimização Dinâmica.

Palavras-chave: Otimização dinâmica, métodos clássicos, métodos naturais, evolução diferencial.

ABSTRACT

The solution of problems of dynamic optimization (POD) involves determining the profile of one or more of a control variable which minimize or maximize a given performance index. Except for a few problems when analytical methods are applicable, the most dynamic optimization problems is nonlinear and requires a numerical solution approach for obtaining the profile of the control variable. Classical methods of numerical solution based on the principle of differential and integral calculus use information derived from Objective Function and constraints to determine an optimal solution. Heuristic natural optimization methods have in common the random nature of the search for an optimal solution. These are methods that do not use information derived from Objective Function and restrictions.

The Differential Evolution algorithm developed by Storn and Price (1995) is of simple implementation and only requires the definition of some parameters. Among some of the features that the algorithm presents Differential Evolution, is the ability to manipulate functions targeted multimodal nonlinear and non-differentiable, and is a method that has a high probability of finding the global optimum.

In this work we present solutions for different dynamic optimization problems. The use of Differential Evolution algorithm simplifies the POD solution avoiding the need, for example, for the solution of problems at the boundary value or significant algebraic manipulations in solving special problems. The results obtained by Differential Evolution are compared with results obtained by other authors using traditional methods and the natural methods. The analysis of the results demonstrate the efficiency of the application of the Differential Evolution algorithm in solution of different dynamic optimization problems.

Keywords: Dynamic optimization, classical methods, natural methods, differential evolution.

Capítulo 1

INTRODUÇÃO

O objetivo da resolução de Problemas de Otimização Dinâmica (POD) é determinar os perfis das variáveis de controle que minimizam ou maximizam uma medida de desempenho. Nos processos químicos as soluções destes problemas são usadas, por exemplo, na determinação de perfis de temperatura em reatores batelada, dos perfis de alimentação de substrato em biorreatores batelada e da distribuição de catalisadores em reatores de leito recheado.

A teoria do controle ótimo resultante do cálculo variacional tem mais de 300 anos de história. Em 1696, Johann Bernoulli desafiou seus contemporâneos para que resolvessem o problema denominado Braquistócrona em no máximo um ano. O objetivo neste problema é encontrar a forma de um fio sem fricção que faz com que um corpo inicialmente em repouso se movimente para um ponto específico no menor tempo possível sob a ação da gravidade. Dentre os cinco matemáticos que responderam ao desafio estavam Leibnitz e Newton. Em 29 de janeiro de 1697, Isaac Newton estabeleceu os princípios do Cálculo Variacional e aplicou-o na solução do problema. Com o desenvolvimento do Cálculo Variacional, as condições necessárias para a otimalidade considerando diferentes tipos de trajetória foram determinadas (SARGENT, 2000). A partir de 1950, com o surgimento do computador digital, problemas mais complexos de otimização dinâmica puderam ser resolvidos.

A formulação de Hamilton-Jacob-Bellman utiliza o princípio da otimalidade para transformar o POD original num problema de resolução de uma equação diferencial parcial. Dentre os métodos que utilizam a formulação de Hamilton-Jacob-Bellman está a Programação Dinâmica (SRINIVASAN et al., 2003).

Os Métodos Indiretos baseiam-se na aplicação do Princípio do Mínimo de Pontryagin, que resulta num problema de valor no contorno (OUELLET e BUI, 1993), (BELL, SARGENT, 2000), (AZHMYAKOV, 2011), (SIMEONI et al., 2012). Os Métodos Diretos transformam o POD em um problema de dimensão finita através da

parametrização das variáveis. Nos métodos sequenciais, apenas a(s) variável(is) de controle é parametrizada, enquanto que nos métodos simultâneos, ambas variáveis de estado e de controle são parametrizadas (FEEHERY,1998), (FIKAR et al.,1998), (OBERLE e SOTHMANN, 1999), (HADIYANTO et al.,2008), (ROHMAN e AZIZ 2011).

Os métodos híbridos combinam a precisão dos métodos indiretos com a maior facilidade dos métodos diretos em encontrar estimativas para algumas das variáveis obtidas através da abordagem indireta (STRYK e BULIRSCH, 1992), (LOBATO, 2004), (PFEIFER, 2007).

O Programa de Pós-Graduação em Engenharia Química da Faculdade de Engenharia Química da Universidade Federal de Uberlândia tem um histórico de trabalhos abordando técnicas de solução de problemas de otimização dinâmica. Gomes (2000) apresentou a solução de problemas algébrico-diferenciais de valor no contorno resultante da aplicação do Princípio de Pontryagin na sua forma original. Lobato (2004) utilizou um método híbrido denominado de método misto na solução de problemas de otimização dinâmica algébrico-diferenciais. Pfeifer (2007) utilizou um método misto para a solução de problemas de otimização dinâmica com restrições de desigualdade. Borges (2008) apresentou um estudo teórico e experimental da otimização dinâmica da fermentação alcoólica em batelada alimentada utilizando a levedura *Saccharomyces cerevisiae* em um fermentador de 5 L.

Os Métodos Indiretos e os Métodos Diretos fazem uso de informações sobre o gradiente da Função Objetivo e das restrições para a determinação da direção de busca do ótimo. De forma geral, eles apresentam uma rápida taxa de convergência quando uma boa estimativa inicial para a solução é fornecida, mas têm dificuldades em lidar com mínimos locais (VANDERPLAATS, 1999).

Os métodos heurísticos que simulam fenômenos biológicos datam dos anos de 1950 e têm como característica comum o caráter aleatório de busca do ótimo. Apenas nos anos de 1980 com um maior desenvolvimento computacional, esses métodos passaram a ser empregados para a otimização de processos. São métodos que não utilizam a informação da derivada da Função Objetivo. Dentre alguns dos métodos de otimização inspirados na natureza tem-se, Recozimento Simulado, Enxame de Partículas, Algoritmos Genéticos, Evolução Diferencial e Colônia de Formigas. Na Figura 1.1 é apresentada uma classificação dos métodos de otimização (Adaptado de Deb, 2001)

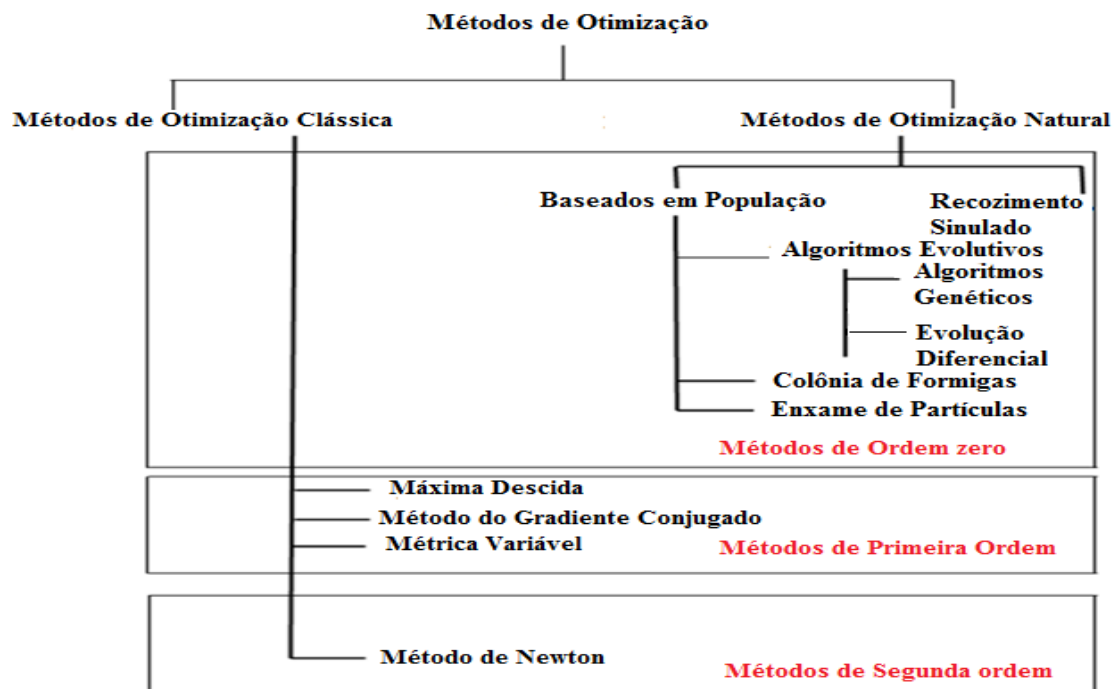


Figura 1.1: Classificação dos Métodos de Otimização

Enquanto os métodos de busca determinísticos partem de um ponto inicial na busca de uma solução ótima e com base em regras pré-especificadas indicam uma direção de busca usando informações locais (OLIVEIRA, 2006), nos métodos heurísticos uma população é gerada aleatoriamente (exceto Recozimento Simulado) e cada indivíduo dessa população é avaliado em relação a sua aptidão (para todos indivíduos da população é calculado um valor de Função Objetivo).

O algoritmo de Evolução Diferencial, desenvolvido por Storn e Price em 1995 surgiu a partir de tentativa de Ken Price em resolver o problema de ajuste polinomial de Chebychev. A idéia inicial foi a de utilizar diferenças vetoriais para perturbar uma população de vetores (STORN, PRICE, 2012). Dentre os vários métodos heurísticos e suas alterações desenvolvidos ao longo dos últimos 30 anos (Recozimento Simulado, KIRKPATRICK et al. (1983), Colônia de Formigas, DORIGO et al. (1996), algoritmos genéticos, enxame de partículas, ENGELBRECHT (2007)), o algoritmo de Evolução Diferencial foi selecionado para a solução de Problemas de Otimização Dinâmica clássicos, devido à sua robustez, fácil implementação computacional e ao histórico de aplicações bem sucedidas em várias outras áreas: projetos de sistemas mecânicos (OLIVEIRA, 2006), otimização de estruturas veiculares (OLIVEIRA et al., 2007), aplicação em problema de controle ótimo multi-objetivo (LOBATO et al., 2007c).

O objetivo deste trabalho é avaliar o desempenho do algoritmo de Evolução Diferencial na solução de Problemas de Otimização Dinâmica, evitando a solução de problemas de valor no contorno gerados pela abordagem indireta e também manipulações algébricas expressivas, especialmente em problemas singulares e problemas de índice diferencial superior. Também é objetivo deste trabalho a determinação do melhor conjunto de parâmetros do algoritmo de Evolução Diferencial aplicado a problemas de índice diferencial superior, através da análise exaustiva de sensibilidade paramétrica.

Esta dissertação está estruturada da seguinte maneira: No Capítulo 2 são apresentados os conceitos fundamentais da Otimização Dinâmica com revisão dos métodos de solução e no Capítulo 3 aspectos gerais dos métodos de otimização naturais, Nestes dois capítulos são apresentadas a revisão e a análise da literatura. No Capítulo 4 o desempenho do Método da Evolução Diferencial aplicado a estudos de caso clássicos da literatura é avaliado quanto à influência de diferentes sementes (geradores randômicos), diferentes números de estágios de controle e número de avaliações de Função Objetivo. Estes resultados são comparados aos obtidos por métodos clássicos e outros métodos naturais. Uma análise comparativa considerando esforço computacional, dimensão das equações, especificação de parâmetros de entrada é apresentada. Para dois casos específicos, é apresentada uma análise da sensibilidade paramétrica considerando os parâmetros do algoritmo de Evolução Diferencial de Storn e Price. As conclusões e sugestões para trabalhos futuros são apresentadas no Capítulo 5.

Capítulo 2

ASPECTOS GERAIS DA OTIMIZAÇÃO DINÂMICA DE PROCESSOS

2.1. Introdução

O interesse em simulação dinâmica e otimização de processos vem aumentando significativamente nas últimas décadas. No âmbito da Engenharia Química, os processos químicos são modelados dinamicamente utilizando Equações Algébrico-Diferenciais (EAD) que representam o comportamento do sistema através de balanços de massa, energia, quantidade de movimento e equações algébricas que representam as relações físicas e termodinâmicas. (BIEGLER, GROSSMANN, 2004).

A formulação geral de um Problema de Otimização Dinâmica pode ser representada pelas Equações (2.1) e (2.2) (BRYSON E HO, 1975):

$$\min J = \Psi(\mathbf{z}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{z}, \mathbf{u}, t) dt \quad (2.1)$$

$$f(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) = 0 \quad (2.2)$$

Sendo $\mathbf{z}$ o vetor de variáveis de estado, $\mathbf{u}$ o vetor de variáveis de controle, Ψ e L a primeira e a segunda parcelas do funcional J (Função Objetivo), e f o sistema de equações algébrico-diferenciais com condições iniciais consistentes dadas por:

$$\varphi(\dot{\mathbf{z}}(t_0), \mathbf{z}(t_0), \mathbf{u}(t_0), t_0) = 0 \quad (2.3)$$

$$\mathbf{z} \in \mathbb{R}^{n_z} \quad (2.4)$$

$$\mathbf{u} \in \mathbb{R}^{n_u} \quad (2.5)$$

$$t \in \mathbb{R}^1 \quad (2.6)$$

$$\Psi: \mathbb{R}^{n_z} \rightarrow \mathbb{R}^1 \quad (2.7)$$

$$L: \mathbb{R}^{n_z + n_u + 1} \rightarrow \mathbb{R}^1 \quad (2.8)$$

$$f: \mathbb{R}^{n_z + n_u + 1} \rightarrow \mathbb{R}^{n_f} \quad (2.9)$$

$$\varphi: \mathbb{R}^{n_z + n_u + 1} \rightarrow \mathbb{R}^{n_\varphi} \quad (2.10)$$

Com exceção de poucos problemas onde os métodos analíticos são aplicáveis (CORREA DO QUINTO, 2010), a grande maioria dos Problemas de Otimização Dinâmica exigem solução numérica. Estes métodos numéricos podem ser classificados em Métodos Clássicos e Métodos Naturais. Nas seções seguintes são apresentados maiores detalhes sobre estes métodos

2.2 Abordagem Algébrico-Diferencial na Solução de Problemas de Otimização Dinâmica

Um sistema de Equações Algébrico-Diferenciais (EAD) pode ser escrito da seguinte maneira:

$$f(\dot{\mathbf{z}}, \mathbf{z}, t) = 0 \quad (2.11)$$

Equações Diferenciais Ordinárias (EDO) são uma classe de Equações Algébrico- Diferenciais. Diferentemente de EDO, uma característica das Equações Algébrico-diferenciais (EAD) é que existem restrições algébricas nas variáveis de estado $\mathbf{z}$ (FEEHERY, 1998).

Estas restrições podem aparecer explicitamente como:

$$h(\mathbf{x}, \mathbf{y}, t) = 0 \quad (2.12)$$

$$g(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t) = 0 \quad (2.13)$$

Em que $\mathbf{z} = (\mathbf{x}, \mathbf{y})$, ou podem aparecer implicitamente devido à singularidade de $\frac{\partial f}{\partial \mathbf{z}}$.

2.2.1 *Índice Diferencial de Equações Algébrico-Diferenciais (EAD)*

O Índice Diferencial é o número mínimo de vezes que o Sistema Algébrico-Diferencial ou parte dele deve ser diferenciado com relação ao tempo para a obtenção de um sistema puramente diferencial.

Segundo Pantelides et al.(1988), EAD de índice 1 tem comportamento muito similar a um conjunto de equações diferenciais ordinárias (EDO). Entretanto, o comportamento das EAD de índice superior (maior do que um) é qualitativamente diferente daquelas resolvidas como EDO.

Bachmann et al. (1990) apresentaram um algoritmo para a redução do índice de um sistema de equações algébrico-diferenciais. Nesse algoritmo, o índice de um sistema de equações algébrico-diferenciais é reduzido até um índice 1 através da construção de equações algébricas que estão escondidas na formulação de um índice superior. England et al.(2005) apresentaram um procedimento sistemático baseado no conceito de tratabilidade do índice, que permite identificar qual parte do sistema de EAD deve ser diferenciado e quantas vezes, evitando a necessidade de se diferenciar o sistema completo. Os autores estudaram três casos envolvendo reatores químicos, sendo dois casos de índice 1 e o outro de índice 3. O problema de otimização dinâmica original foi transformado em um sistema de EAD usando uma formulação variacional, introduzindo fatores de perturbação constantes e multiplicadores de Lagrange nas equações diferenciais e nas restrições de desigualdade. Por fim, as restrições de desigualdade são eliminadas introduzindo novas variáveis.

Um dos grandes problemas na redução do índice está no fato de que uma solução para o sistema reduzido não necessariamente é uma solução para o sistema de equações algébrico-diferenciais. Embora qualquer solução definida pelo sistema de equações algébrico-diferenciais seja também uma solução para o sistema reduzido, apenas as soluções que satisfaçam os valores iniciais consistentes também são soluções para a EAD original. (SØRENSEN et al. , 2006).

2.3. Métodos Clássicos

Os métodos clássicos de solução são fundamentados em princípios do cálculo diferencial e integral, ou seja, tanto os métodos indiretos quanto os métodos diretos fazem uso de informações sobre o gradiente da Função Objetivo e das restrições para a determinação da direção de busca do ótimo. De forma geral a taxa de convergência

depende de uma da estimativa inicial da trajetória ótima. Como tais métodos investem todo esforço em um único ponto do espaço de projeto, ao se depararem com mínimos locais eles convergem prematuramente para um mínimo local (VANDERPLAATS, 1999).

Em um método baseado no gradiente da Função Objetivo, dada uma estimativa inicial para um vetor de variáveis de decisão $\mathbf{z}$ e considerando k iterações, uma direção de busca $\mathbf{p}_k$ e um comprimento do passo α_k são determinados. A direção de busca fornece uma direção ao longo da qual o valor atual $\mathbf{z}_k$ é modificado enquanto o tamanho do passo fornece a magnitude da mudança de $\mathbf{z}_k$. A atualização da variável $\mathbf{z}$ é feita iterativamente conforme a Equação (2.14):

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \alpha_k \mathbf{p}_k \quad (2.14)$$

No caso da minimização, a direção de busca é escolhida de maneira que diminua suficientemente a Função Objetivo de forma que:

$$f(\mathbf{z}_{k+1}) \leq f(\mathbf{z}_k) + K \alpha_k \nabla f^T(\mathbf{z}_k) \alpha_k \quad (2.15)$$

Os principais métodos de solução de problemas de Programação Não Linear (NLP) baseados em gradiente são apresentados na sequência:

- Programação Quadrática (QP): caracterizada pela seguinte função objetivo e restrições:

$$\min S(\mathbf{x}) = \mathbf{c}^T \mathbf{z} + \frac{1}{2} \mathbf{z}^T \mathbf{Q} \mathbf{z} \quad (2.16)$$

$$\text{sujeito a : } \mathbf{A} \mathbf{z} \leq \mathbf{b} \quad (2.17)$$

$$\mathbf{z} \geq 0 \quad (2.18)$$

Sendo $\mathbf{A}$ uma matriz $m \times n$, $\mathbf{c}$ um vetor de coeficientes e $\mathbf{Q}$ uma matriz simétrica $n \times n$.

Sendo as restrições lineares e linearmente independentes, as condições de qualificação de segunda ordem são sempre satisfeitas, logo, as condições de primeira ordem de Karush-Kuhn-Tucker (KKT) são as condições necessárias para a otimalidade, ou:

$$\nabla_z L(z^*, \lambda^*, \mu^*) = \nabla S(z^*) + (\lambda^*)^T \nabla h(z^*) + (\mu^*)^T \nabla g(z^*) = 0 \quad (2.19)$$

$$h(z^*) = 0 \quad (2.20)$$

$$g(z^*) \leq 0 \quad (2.21)$$

$$\mu_j^* g_j(z^*) = 0 \quad (2.22)$$

$$\mu^* \geq 0 \quad (2.23)$$

Sendo L a função de Lagrange. As variáveis λ e μ são os multiplicadores de Lagrange e estão associadas às restrições incorporadas à Função Objetivo. O sinal de asterisco (*) apresentado nas Equações (2.19) até (2.23) representa o ponto ótimo.

- Programação Quadrática Sequencial (SQP): a ideia em SQP é basicamente resolver as equações de Karush-Kuhn-Tucker (KKT) aproximando de métodos utilizados em problemas sem restrições (por exemplo, o Método de Newton) que apresentam convergência quadrática.

Em SQP são feitas aproximações da matriz Hessiana da função de Lagrange utilizando métodos de atualização tipo quasi-Newton (possuem convergência entre ordem1 e ordem2). Assim, o problema de programação quadrática resultante é associado a uma busca em linha, sendo então resolvido para se determinar uma direção de busca.

Aplicando as condições KKT ao problema de programação não linear:

$$\min S(\mathbf{z}) \quad (2.24)$$

$$\text{sujeito a : } h_j(\mathbf{z}) = 0, j = 1, 2, \dots, m \quad (2.25)$$

$$g_j(\mathbf{z}) \leq 0, j = 1, 2, \dots, p \quad (2.26)$$

Através da função de Lagrange:

$$L(\mathbf{z}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = S(\mathbf{z}) + \boldsymbol{\lambda}^T \nabla h(\mathbf{z}) + \boldsymbol{\mu}^T \nabla g(\mathbf{z}) \quad (2.27)$$

Obtém-se o seguinte conjunto de equações:

$$\nabla_z L(z^*, \lambda^*, \mu^*) = \nabla S(z^*) + (\lambda^*)^T \nabla h(z^*) + (\mu^*)^T \nabla g(z^*) = 0 \quad (2.28)$$

$$h(\mathbf{z}) = 0 \quad (2.29)$$

$$g_j(\mathbf{z}) = 0, \quad j \in \{\text{restrições ativas}\} \quad (2.30)$$

$$\mu_i \geq 0, \quad i \in \{\text{restrições inativa}\} \quad (2.31)$$

Que é transformado nos sucessivos problemas de programação quadrática

$$\min q(\mathbf{d}) = \nabla^T S(\mathbf{z}_k) \mathbf{d} + \frac{1}{2} \mathbf{d}^T H(\mathbf{z}^k, \lambda^k, \mu^k) \mathbf{d} \quad (2.32)$$

$$\text{sujeito a : } h(\mathbf{z}^k) + \nabla^T h(\mathbf{z}^k) \mathbf{d} = 0 \quad (2.33)$$

$$g(\mathbf{z}^k) + \nabla^T g(\mathbf{z}^k) \mathbf{d} \leq 0 \quad (2.34)$$

Nas Equações (2.32)-(2.34), $\mathbf{d}$ é o vetor de direção de busca.

2.4. Métodos Indiretos

Estes métodos surgiram com o desenvolvimento do cálculo variacional e requerem a adição de um número de variáveis de coestado, também denominadas variáveis adjuntas, igual ao número de variáveis de estado (CONWAY, 2012).

Apresenta-se a seguir a formulação matemática para o cálculo variacional conforme apresentado em Lobato (2004) e Pfeifer (2007), considerando nessa seção que o vetor inicial das variáveis de estado é conhecido (isto é, o vetor inicial não é determinado pela otimização), as trajetórias de controle não estão sujeitas a restrições e as trajetórias de estado são restritas por EAD:

$$\min J = \Psi(\mathbf{z}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{z}, \mathbf{u}, t) dt \quad (2.35)$$

Sujeito ao sistema de EAD:

$$f(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) = 0 \quad (2.36)$$

Com as seguintes condições iniciais consistentes:

$$\varphi(\dot{\mathbf{z}}(t_0), \mathbf{z}(t_0), \mathbf{u}(t_0), t_0) = 0 \quad (2.37)$$

$$\mathbf{z} \in \mathbb{R}^{n_z} \quad (2.38)$$

$$\mathbf{u} \in \mathbb{R}^{n_u} \quad (2.39)$$

$$t \in \mathbb{R}^1 \quad (2.40)$$

$$\Psi: \mathbb{R}^{n_z} \rightarrow \mathbb{R}^1 \quad (2.41)$$

$$L: \mathbb{R}^{n_z + n_u + 1} \rightarrow \mathbb{R}^1 \quad (2.42)$$

$$f: \mathbb{R}^{n_z + n_u + 1} \rightarrow \mathbb{R}^{n_f} \quad (2.43)$$

$$\varphi: \mathbb{R}^{n_z + n_u + 1} \rightarrow \mathbb{R}^{n_\varphi} \quad (2.44)$$

A função Ψ na equação 2.35 pode ser expressa como:

$$\Psi(\mathbf{z}(t_f), t_f) = \Psi(\mathbf{z}(t_0), t_0) + \int_{t_0}^{t_f} \frac{d\Psi}{dt} \quad (2.45)$$

Desde que seja admitido que o tempo inicial t_0 e as condições de estado $\mathbf{z}(t_0)$ são fixos, a Função Objetivo pode ser expressa como:

$$J = \int_{t_0}^{t_f} \bar{L}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) dt \quad (2.46)$$

sendo:

$$\bar{L}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) = \frac{d\Psi}{dt} + L = \frac{\partial \Psi}{\partial t} + \left[\frac{\partial \Psi}{\partial \dot{\mathbf{z}}} \right]^T \dot{\mathbf{z}} + L \quad (2.47)$$

Uma Função Objetivo aumentada é formada adicionando as restrições à Função Objetivo através do uso das variáveis adjuntas $\lambda(t)$:

$$\bar{J} = \int_{t_0}^{t_f} [\bar{L}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) + \boldsymbol{\lambda}^T(t) f(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t)] dt \quad (2.48)$$

Por conveniência, definimos o *Hamiltoniano*:

$$H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, t) = \bar{L}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) + \boldsymbol{\lambda}^T(t)[\mathbf{f}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t)] \quad (2.49)$$

Para obter as condições necessárias para o ótimo, é necessário definir a variação do funcional:

$$\bar{J} = \int_{t_0}^{t_f} H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, t) dt \quad (2.50)$$

O incremento do funcional será:

$$\begin{aligned} \Delta \bar{J} = & \int_{t_0}^{t_f} [H(\dot{\mathbf{z}} + \delta \dot{\mathbf{z}}, \mathbf{z} + \delta \mathbf{z}, \mathbf{u} + \delta \mathbf{u}, \boldsymbol{\lambda} + \delta \boldsymbol{\lambda}, t) - H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, t)] dt \\ & + \int_{t_f}^{t_f + \delta t_f} H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, t) dt \end{aligned} \quad (2.51)$$

Fazendo a expansão do incremento em série de Taylor ao redor do ponto $(\dot{\mathbf{z}}(t), \mathbf{z}(t), \mathbf{u}(t))$ e extraíndo os termos que são lineares tem-se a variação de J :

$$\delta \bar{J} = \int_{t_0}^{t_f} \left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \delta \dot{\mathbf{z}} + \frac{\partial H}{\partial \mathbf{z}} \delta \mathbf{z} + \frac{\partial H}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial H}{\partial \boldsymbol{\lambda}} \delta \boldsymbol{\lambda} \right] dt + H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, t) \delta t_f \quad (2.52)$$

que pode ser simplificada integrando o primeiro termo por partes para se obter:

$$\begin{aligned} \delta \bar{J} = & \int_{t_0}^{t_f} \left[\left[\frac{\partial H}{\partial \dot{\mathbf{z}}} - \frac{d}{dt} \left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \right] \right] \delta \mathbf{z} + \frac{\partial H}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial H}{\partial \boldsymbol{\lambda}} \delta \boldsymbol{\lambda} \right] dt + \left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \right]_{t=t_f} \delta \mathbf{z}(t_f) \\ & + H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, t) \delta t_f \end{aligned} \quad (2.53)$$

Usando a relação definida a seguir:

$$\delta \mathbf{z}(t_f) = \delta \mathbf{z}_f - \dot{\mathbf{z}} \delta t_f \quad (2.54)$$

Substituindo a Equação 2.54 na Equação 2.53 temos:

$$\begin{aligned} \delta \bar{J} = & \left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \right]_{t=t_f} \delta \mathbf{z}_f + \left[H - \frac{\partial H}{\partial \dot{\mathbf{z}}} \dot{\mathbf{z}} \right]_{t=t_f} \delta t_f \\ & + \int_{t_0}^{t_f} \left[\left[\frac{\partial H}{\partial \mathbf{z}} - \frac{d}{dt} \left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \right] \right] \delta \mathbf{z} + \frac{\partial H}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial H}{\partial \lambda} \delta \lambda \right] dt \end{aligned} \quad (2.55)$$

As condições necessárias de primeira ordem para o ótimo são encontradas fazendo a variação de $\bar{J}$ igual a zero. Logo, as condições são dadas por:

$$\left[\frac{\partial H}{\partial \mathbf{z}} \right] - \frac{d}{dt} \left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \right] = 0 \quad (2.56)$$

$$\frac{\partial H}{\partial \mathbf{u}} = 0 \quad (2.57)$$

$$\frac{\partial H}{\partial \lambda} = 0 \quad (2.58)$$

$$\left[\frac{\partial H}{\partial \dot{\mathbf{z}}} \delta \mathbf{z} \right]_{t=t_f} + \left[H - \frac{\partial H}{\partial \dot{\mathbf{z}}} \dot{\mathbf{z}} \right]_{t=t_f} \delta t_f = 0 \quad (2.59)$$

As Equações (2.56 a 2.59) definem um sistema de EAD de valor no contorno. Estas Equações podem ser simplificadas expandindo os termos que incluem Ψ na Equação 2.56.

$$\delta \bar{J} = \frac{\partial}{\partial \mathbf{z}} \left[\frac{\partial \Psi}{\partial \mathbf{z}} \dot{\mathbf{z}} + \frac{\partial \Psi}{\partial t} \right] - \frac{d}{dt} \left[\frac{\partial}{\partial \dot{\mathbf{z}}} \left[\frac{\partial \Psi}{\partial \mathbf{z}} \dot{\mathbf{z}} \right] \right] = \left[\frac{\partial^2 \Psi}{\partial \mathbf{z} \partial t} + \frac{\partial^2 \Psi}{\partial \mathbf{z}^2} \dot{\mathbf{z}} \right] + \left[\frac{\partial^2 \Psi}{\partial t \partial \mathbf{z}} + \frac{\partial^2 \Psi}{\partial \mathbf{z}^2} \dot{\mathbf{z}} \right] = 0 \quad (2.60)$$

Admitindo que as derivadas parciais de segunda ordem são contínuas, a ordem de diferenciação pode ser mudada e a expressão inteira igualada a zero. Substituindo as Equações 2.49 e 2.60 e as Equações (2.56)-(2.59) podem ser simplificadas:

$$\frac{\partial L}{\partial \mathbf{z}} + \lambda^T \frac{\partial f}{\partial \mathbf{z}} - \dot{\lambda}^T \frac{\partial f}{\partial \dot{\mathbf{z}}} - \lambda^T \frac{d}{dt} \left[\frac{\partial f}{\partial \dot{\mathbf{z}}} \right] = 0 \quad (2.61)$$

$$\frac{\partial L}{\partial \mathbf{u}} + \lambda^T \frac{\partial f}{\partial \mathbf{u}} = 0 \quad (2.62)$$

$$f(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) = 0 \quad (2.63)$$

$$\left[\lambda^T \frac{\partial f}{\partial \dot{\mathbf{z}}} + \frac{\partial \Psi}{\partial \mathbf{z}} \right]_{t=t_f} \delta \mathbf{z}_f + \left[\frac{\partial \Psi}{\partial t} + L + \lambda^T f - \lambda^T \frac{\partial f}{\partial \dot{\mathbf{z}}} \dot{\mathbf{z}} \right]_{t=t_f} \delta t_f = 0 \quad (2.64)$$

Estas condições são uma generalização das condições necessárias, também denominadas de equações de Euler-Lagrange para otimização dinâmica de sistemas de EAD.

A aplicação dos métodos indiretos resulta em problema de valor no contorno multipontos originado das condições necessárias para a otimalidade. Os problemas de valor no contorno multipontos podem ser resolvidos por métodos como Método do Simples Chute (SIMEONI et al., 2012), Métodos dos Múltiplos Chutes (KIM et al., 2011), Método da Colocação (POURTAKDOUST e JALALI, 2001), Método da Colocação Ortogonal (OH e LUSS, 1977), Método das Diferenças Finitas (OUELLET e BUI, 1993).

Ouellet e Bui (1993) propuseram um método numérico na solução de um problema de controle ótimo envolvendo processos térmicos industriais. O problema foi formulado através do cálculo variacional utilizando o Princípio do Máximo de Pontryagin, resultando em um problema de valor no contorno em dois pontos. O problema foi discretizado através de um esquema de diferenciação central de Euler e a solução obtida pelo método de Newton-Raphson. Os autores utilizaram um programa denominado COMMINS (Continuous Optimization using the Maximum Principle with Minimum Programming) para a solução do problema. As únicas entradas requeridas pelo COMMINS são a função a ser minimizada ou maximizada, o tempo final, as equações diferenciais envolvendo as variáveis de estado, as condições de contorno e a restrição sobre a variável de controle.

Bell e Sargent (2000) apresentaram uma abordagem de solução para problemas de controle ótimo no qual o comportamento do sistema é descrito por equações algébrico-diferenciais e desigualdades. As desigualdades são transformadas em igualdades utilizando variáveis de folga e as funções barreira são usadas para lidar com os limites das variáveis. Assim, são geradas as condições necessárias de otimalidade desse sistema, levando a um problema de valor no contorno em dois pontos, resolvido usando algoritmos de ponto interior para resolver os problemas teste, incluindo o problema de mistura de catalisadores de índice 3. Dentre as vantagens apontadas pelos autores na utilização desta abordagem está o fato de que ela elimina o problema combinatorial de identificar as restrições ativas, além de eliminar a necessidade de análises preliminares para determinar a estrutura da solução.

Azhmyakov (2011) estudou um tipo específico de processos de controle híbrido (partes contínuas e partes discretas) não estacionário, utilizando o método de Lagrange e a técnica de gradientes reduzidos para obter as condições necessárias de otimalidade de primeira ordem.

Simeoni et al. (2012) analisaram estratégias ideais para implantação de uma nave espacial em uma órbita elíptica através de abordagem indireta. Foi investigado nesse estudo, a influência do nível de impulso na nave e o número de revoluções do desempenho de transferência. Os autores utilizaram a técnica do simples chute onde o problema de valor no contorno multipontos é transformado em uma série de problemas de valor inicial levando a convergência por meio do método de Newton. Os autores comprovaram que a técnica utilizada é rápida e confiável, sendo a convergência quase sempre obtida automaticamente, sem qualquer ação do usuário.

Na sequência, são apresentadas formulações para casos particulares considerando especificações de variáveis de controle ou de estado, restrições de igualdade e de desigualdade e tempos finais fixos e livres.

a) Problemas com o tempo final fixo

Se t_f é fixo, logo δt_f na Equação 2.64 será zero. Caso a variável de estado não seja especificada no tempo final, as condições no ponto final têm de satisfazer a:

$$\left[\lambda^T \frac{\partial f}{\partial \dot{\mathbf{z}}} + \frac{\partial \Psi}{\partial \mathbf{z}} \right]_{t=t_f} \delta \mathbf{z}_f = 0 \quad (2.65)$$

Como δz_f é arbitrário, $\delta z_f \neq 0$ e para satisfazer (2.51) tem-se:

$$\left[\lambda^T \frac{\partial f}{\partial \dot{\mathbf{z}}} + \frac{\partial \Psi}{\partial \mathbf{z}} \right]_{t=t_f} = 0 \quad (2.66)$$

b) Problemas com o tempo final livre

Como t_f é livre, a suposição de que $\delta t_f = 0$ não pode ser feita. Assim, além das condições dadas pelas Equações (2.61)-(2.63) para o caso em que se têm as variáveis fixas no tempo final ou variáveis de estado livre, o sistema deverá atender a seguinte condição:

$$\left[\frac{\partial \Psi}{\partial t} + L + \lambda^T f - \lambda^T \frac{\partial f}{\partial \dot{\mathbf{z}}} \dot{\mathbf{z}} \right]_{t=t_f} = 0 \quad (2.67)$$

c) *Algumas variáveis de estado especificadas no tempo final fixo*

Considere o problema de otimização definido pelas Equações (2.35)-(2.37) com algumas variáveis de estado especificadas em $t=t_f$. Se z_i é o i -ésimo termo componente do vetor de estado $\mathbf{z}$, é definido em $t=t_f$, então como a variação $\delta z_i(t_f)$ na Equação 2.64 não pode ser nula, é necessário que a Equação 2.65 seja satisfeita. A Equação 2.62, $\partial H / \partial \mathbf{u} = 0$ necessita de uma condição adicional para o problema com restrição final. No presente caso, $\delta u(t)$ não é completamente arbitrário e o conjunto admissível de $\delta u(t)$ é sujeito às restrições:

$$\delta \mathbf{z}_i(t_f) = 0 \quad i = 1, \dots, q \quad (2.68)$$

Um conjunto admissível de $\delta u(t)$ pode ser definido com os valores de $\delta u(t)$ que satisfazem todas as restrições do problema, como por exemplo a Equação 2.68. Desde que $z_i(t_f)$ esteja especificado para $i=1, \dots, q$, é consistente considerar que:

$$\varphi = \varphi(z_{q+1}, \dots, z_n)_{t=t_f} \quad (2.69)$$

As Equações (2.61)-(2.64) permanecem inalteradas para este caso, apenas a condição de contorno em $t=t_f$ passa a ser dada por:

$$\lambda_i(t_f) = \begin{cases} 0 & j = 1, \dots, q \\ \left(\frac{\partial \phi}{\partial z_j} \right)_{t=t_f} & j = q + 1, \dots, n \end{cases} \quad (2.70)$$

d) *Sistemas com funções de variáveis de estado especificadas no tempo final fixo*

Seja o problema de otimização definido pelas Equações (2.35)-(2.37) sujeito à restrição definida pela Equação 2.71 de dimensão q , função das variáveis de estado e com o valor definido no tempo final.

$$\varphi(\mathbf{z}(t_f), t_f) = 0 \quad (2.71)$$

A Equação 2.71 pode ser adicionada à função objetivo através de multiplicadores de Lagrange $\mathbf{v}$, um vetor de dimensão q .

$$J = \psi(\mathbf{z}(t_f), t_f) + \mathbf{v}^T \varphi(\mathbf{z}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{z}, \mathbf{u}, t) dt \quad (2.72)$$

Definindo:

$$\Psi = \psi(\mathbf{z}(t_f), t_f) + \mathbf{v}^T \varphi(\mathbf{z}(t_f), t_f) \quad (2.73)$$

O conjunto de parâmetros $\mathbf{v}$ deve ser escolhido para satisfazer a equação (2.71). Logo, as condições necessárias são dadas pelas equações (2.61-2.64) e por:

$$\lambda^T(t_f) = \left(\frac{\partial \Psi}{\partial \mathbf{z}} + \mathbf{v}^T \frac{\partial \varphi}{\partial \mathbf{z}} \right)_{t=t_f} \quad (2.74)$$

A Equação 2.62 determina o vetor $\mathbf{u}(t)$, as equações (2.61)-(2.66) formam um sistema de EAD de valor no contorno com q parâmetros $\mathbf{v}$ para serem determinados na equação 2.74 tal que a Equação 2.71 seja satisfeita.

e) Problemas com Restrições de Igualdade na Variável de Controle

O problema de otimização dinâmica nesse caso é definido como:

$$\min J = \Psi(\mathbf{z}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{z}, \mathbf{u}, t) dt \quad (2.75)$$

$$\varphi(\mathbf{z}(t_f), t_f) = 0 \quad (2.76)$$

$$\mathbf{f}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) = 0 \quad (2.77)$$

$$\mathbf{C}(\mathbf{u}, t) = 0 \quad (2.78)$$

Sendo $\mathbf{C}(\mathbf{u}, t)$ o vetor de restrições de igualdade na variável de controle. Neste caso $\mathbf{u}(t)$ é um vetor de variáveis de controle de dimensão $m \geq 2$ e $\mathbf{C}$ é uma função escalar.

A redefinição do *Hamiltoniano* (Equação 2.49) será:

$$H(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, \lambda, t) = \bar{L}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) + \lambda^T(t) \mathbf{f}(\dot{\mathbf{z}}, \mathbf{z}, \mathbf{u}, t) + \mu C \quad (2.79)$$

As condições necessárias, Equações (2.61, 2.63 e 2.64) ficam inalteradas, no entanto a Equação 2.62 será redefinida como:

$$\frac{\partial L}{\partial \mathbf{u}} + \lambda^T \frac{\partial \mathbf{f}}{\partial \mathbf{u}} + \mu \frac{\partial C}{\partial \mathbf{u}} = 0 \quad (2.80)$$

O sistema formado pelas condições necessárias e pela Equação 2.78 representa as $m+1$ condições para determinar o m -ésimo componente do vetor de controle $\mathbf{u}(t)$ e a função escalar $\mu(t)$.

f) Problemas com Restrições de Igualdade nas Variáveis de Controle e de Estado

Neste caso a Equação 2.78 será substituída por:

$$C(\mathbf{z}, \mathbf{u}, t) = 0 \quad (2.81)$$

As condições obtidas na seção anterior podem ser novamente aplicadas. No entanto, um novo termo será acrescentado a Equação 2.61, sendo reescrita como:

$$\frac{\partial L}{\partial \mathbf{z}} + \lambda^T \frac{\partial \mathbf{f}}{\partial \mathbf{z}} + \mu^T \frac{\partial C}{\partial \mathbf{z}} - \dot{\lambda}^T \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{z}}} - \lambda^T \frac{d}{dt} \left[\frac{\partial \mathbf{f}}{\partial \dot{\mathbf{z}}} \right] = 0 \quad (2.82)$$

g) Problemas com Restrições de Igualdade nas Variáveis de Estado

Caso a restrição não tenha dependência explícita na variável de controle, ocorre uma complexidade adicional. Seja a restrição:

$$S(\mathbf{z}, t) = 0 \quad (2.83)$$

Se a restrição é aplicada sobre todo o intervalo $t_0 \leq t \leq t_f$, a derivada temporal da restrição é nula ao longo da trajetória:

$$\frac{dS}{dt} = \frac{\partial S}{\partial t} + \frac{\partial S}{\partial \mathbf{z}} \dot{\mathbf{z}} = \frac{\partial S}{\partial \mathbf{z}} \mathbf{f}(\mathbf{z}, \mathbf{u}, t) = 0 \quad (2.84)$$

A Equação 2.84 pode ou não revelar a dependência da variável de controle $\mathbf{u}$. Caso a Equação 2.84 revele a dependência de $\mathbf{u}$, então pode ser tratada como uma restrição do tipo da equação 2.81. Para isto, deve-se eliminar um componente de $\mathbf{z}$ como uma função dos $n-1$ componentes remanescentes utilizando a Equação 2.83 como uma condição de contorno em $t = 0$ ou $t = t_f$. Se a Equação 2.84 não revelar a variável de controle explicitamente, deve-se repetir o processo de diferenciar a equação até que a variável de controle $\mathbf{u}$ seja revelada explicitamente. Surge assim o conceito de ordem da restrição de igualdade na variável de controle, que é definida como o número de vezes

que a restrição deve ser diferenciada para que se obtenha a dependência da variável de controle $\mathbf{u}$. A q -ésima derivada temporal da restrição da Equação 2.83 é representada por:

$$S^q(\mathbf{z}, \mathbf{u}, t) = 0, \quad S^q(\mathbf{z}, \mathbf{u}, t) = \frac{d^q S}{dt^q} \quad (2.85)$$

Neste caso, q componentes de $\mathbf{z}$ devem ser eliminados manuseando os $(n-q)$ componentes remanescentes, usando as q relações:

$$\begin{bmatrix} S(\mathbf{z}, t) \\ S^1(\mathbf{z}, t) \\ \vdots \\ S^{q-1}(\mathbf{z}, t) \end{bmatrix} = 0 \quad (2.86)$$

Ou adicionando a Equação 2.86 como um conjunto de condições de contorno em $t=t_0$ ou $t=t_f$.

A existência de restrições de igualdade nas variáveis de estado em um POD pode aumentar o índice do sistema aumentado formado pelas equações diferenciais e pelas restrições de igualdade. É interessante observar que uma restrição de igualdade pode surgir quando um POD formado por equações diferenciais implícitas do tipo:

$$F(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, p, t) = 0 \quad (2.87)$$

onde $\mathbf{u}$ é a variável de controle, p é o parâmetro e $\mathbf{x}$ é a variável de estado, é reescrito definindo uma nova variável de controle $\mathbf{u}^* = [\mathbf{u}, \mathbf{v}]$ de maneira que o sistema aumentado definido como:

$$\dot{\mathbf{x}} = \mathbf{v} \quad (2.88)$$

$$G(\mathbf{x}, \mathbf{u}, \mathbf{v}, p, t) = 0 \quad (2.89)$$

torna-se um sistema de EAD onde a Equação 2.89 passa ser a nova restrição algébrica.

h) Problemas com Restrições de Desigualdade na Variável de Controle

Neste caso o problema de otimização considerado para simplificar a análise é de tempo fixo e sem restrição definida no ponto final, sujeito a uma restrição de desigualdade do tipo:

$$C(\mathbf{u}, t) \leq 0 \quad (2.90)$$

Se o *Hamiltoniano* for definido por $H^0 = \lambda f + L$, considerando que $\delta z_i = 0$, $\partial H / \partial \lambda = 0$, e para simplificações em que os coeficientes de $\delta \mathbf{z}$ são iguais a zero, pode ser reescrita como:

$$\delta J = \int_{t_0}^{t_f} H_u^0 \delta u dt \triangleq \int_{t_0}^{t_f} H_u^0(\mathbf{z}, \lambda, \mathbf{u}, t) dt \quad (2.91)$$

As condições necessárias para este problema são as Equações (2.61)-(2.64). Para que $\mathbf{u}(t)$ seja minimizado, $\delta J \geq 0$ para todo o conjunto admissível de $\mathbf{u}(t)$. Isto implica que $\delta H^0 \geq 0$ para todo t e todo conjunto admissível $\mathbf{u}(t)$. Os pontos onde ocorrem os valores ótimos de $\mathbf{u}(t)$ têm a seguinte propriedade:

$$\delta H^0 = H_u^0 \delta u \geq 0 \quad (2.92)$$

$$\delta C = C_u \delta u \leq 0 \quad (2.93)$$

2.4.1. Arcos Singulares

O termo singular é utilizado para denotar Problemas de Otimização Dinâmica em que a aplicação do Princípio do Máximo de Pontryagin não fornece uma relação explícita entre as variáveis de controle, as variáveis de estado e de coestado (LAMNABHI-LAGARRIGUE, 1987). Conceitualmente, arcos singulares ocorrem quando a matriz Hessiana da função Hamiltoniano com relação às variáveis de controle é singular (BRYSON E HO, 1975). No caso específico de presença da variável de controle na forma linear, o problema apresenta singularidade.

2.4.2. Função Identificadora de Fases (FIF)

Funções Identificadoras de Fases (Switching Functions) são funções que indicam quando uma restrição que está ativa torna-se inativa e vice versa. Um caso particular e de grande interesse é quando a variável de controle aparece linearmente na função *Hamiltoniano* configurando singularidade na variável de controle. Para problemas assim, em geral não existirá mínimo a menos que restrições de desigualdade no estado ou controle sejam especificadas. Se as restrições de desigualdade são lineares nas variáveis de controle, é razoável esperar que a solução mínima, se existir, exigirá

que as variáveis de controle estejam localizadas em um ponto ou outro do limite da região viável de controle (BRYSON;HO,1975).

Considere o seguinte sistema de equações:

$$\dot{x} = F(x) + g(x)u, \quad x(0) = x_0 \quad (2.94)$$

A variável de controle escalar é dada por:

$$u_{min} \leq u \leq u_{max} \quad (2.95)$$

Sendo a função *Hamiltoniano* definida por:

$$H = \lambda^T (F(x) + g(x)u) \quad (2.96)$$

Para esta classe de controle apresenta-se:

$$\begin{cases} u_{max} & \text{se } \lambda^T g < 0 \\ FIF & \text{se } \lambda^T g = 0 \\ u_{min} & \text{se } \lambda^T g > 0 \end{cases} \quad (2.97)$$

Conforme apresentado na Equação 2.96, o *Hamiltoniano* é linear no controle e a condição estacionária não apresenta a variável de controle u explicitamente. Assim a equação deve ser diferenciada em relação ao tempo para a obtenção de u . A não obtenção de u faz com que esse problema apresente flutuação de seu índice diferencial durante a integração (LOBATO, 2004).

2.5. Métodos Diretos

Nestes métodos o POD é convertido em um problema de otimização paramétrico. Nos métodos Sequenciais apenas a variável de controle é discretizada e nos métodos Simultâneos as variáveis de controle e de estado são discretizadas. Nos métodos diretos, o problema pode ser convertido em um problema de dimensão finita, geralmente um problema de programação não linear (NLP) que pode ser resolvido por um algoritmo de otimização numérico, como por exemplo, Programação Sequencial Quadrática (SQP).

Dentre algumas das vantagens da utilização dos métodos diretos em relação aos métodos indiretos apresentam-se (FEEHERY, 1998):

- os Métodos Diretos são mais facilmente implementados do que os Métodos Indiretos, pois não exigem a geração de equações de coestado adicionais;
- dificuldade em encontrar estimativas para algumas variáveis em problemas de valor no contorno.

2.5.1. Métodos Sequenciais

No método seqüencial direto, o problema de otimização é convertido em um problema de programação não linear pela parametrização dos perfis de controle. Nesta abordagem, apenas as variáveis de controle $u(t)$ são discretizadas sobre elementos finitos utilizando polinômios, sendo os coeficientes dos polinômios e o tamanho dos elementos finitos, as variáveis de decisão de um algoritmo de programação não linear (NLP), enquanto que os perfis das variáveis de estado $z(t)$ são obtidos por integração numérica de um sistema de equações algébricas diferenciais. Uma vantagem que o método seqüencial apresenta é a reduzida dimensão do problema de otimização quando comparado ao método simultâneo. No entanto, as restrições nas variáveis de estado não podem ser aplicadas diretamente no problema de programação não linear. A variável de controle pode ser discretizada utilizando colocação ortogonal em elementos finitos como a seguir:

$$u_k(t) = \sum_{j=1}^k u_{ij} \theta_j(t) \quad (2.98)$$

$$\theta_j(t) = \prod_{k=1, j}^K \frac{t - t_{ik}}{t_{ij} - t_{ik}} \quad (2.99)$$

Sendo K o número de pontos de colocação ortogonal e i o numero de elementos finitos.

Diferentes formas de discretização da variável de controle são apresentadas na literatura de controle ótimo, como por exemplo: Binder et al. (2000) aproximaram a variável de controle por uma expansão polinomial por partes :

$$u_i(t) \approx \bar{u}_i(\mathbf{p}_{\Sigma_i}, t) = \sum_{k \in \Sigma_i} p_{i,k} \phi_{i,k}(t) \quad (2.100)$$

Sendo que Σ_i corresponde ao conjunto de índices que define apropriadamente a parametrização da função $\phi_{i,k}(t)$ e $\mathbf{p}_{\Sigma_i}$ é o vetor de parâmetros associados

Schlegel et al. (2005) expandiram a discretização do perfil de controle como uma combinação linear de β -splines, sendo as funções *spline* uma classe de funções utilizadas freqüentemente para interpolação.

$$u_i(t) = \sum_{j=1}^n \hat{u}_{i,j} \varphi_j^{(m)}(t) \quad (2.101)$$

$$\varphi_j^{(m)} = \frac{t - \tau_j}{\tau_{j+m-1} - \tau_j} \varphi_j^{(m-1)}(t) + \frac{\tau_{j+m} - t}{\tau_{j+m} - \tau_{j+1}} \varphi_{j+1}^{(m-1)}(t) \quad (2.102)$$

Na literatura são apresentados trabalhos utilizando o método direto sequencial, como por exemplo:

Vassiliadis, Sargent, Pantelides (1994) resolveram uma classe de problemas de otimização dinâmica com restrições de trajetória de igualdade e desigualdade. Os autores solucionaram problemas multiestágio, sendo cada estágio representado por um sistema de equações algébrico diferenciais de índice 1. Foi empregada a parametrização da variável de controle acoplado a um método de integração numérico multipasso. Para o cálculo do gradiente da Função Objetivo, os autores utilizaram equações de sensibilidade, que consistem na diferenciação do sistema algébrico diferencial em relação aos parâmetros de otimização.

Feehery (1998) utilizou o método seqüencial para solucionar problemas de otimização dinâmica de diferentes índices diferenciais com restrições de trajetória. O objetivo do trabalho foi melhorar a eficiência em que problemas de otimização dinâmica podem ser resolvidos e desenvolver métodos melhorados para incluir restrições de trajetória. Foi empregada a análise de sensibilidade paramétrica para sistemas híbridos (combinação existente e interação entre fenômenos discretos e contínuos). O autor concluiu, entre outras coisas, que a parametrização do controle é mais atrativa que métodos diretos simultâneos, pois pode “tirar proveito” de solucionadores numéricos eficientes para Problemas de Valor inicial e também manipular restrições diretamente através da parametrização do controle.

Fikar et al. (1998) aplicaram a técnica de parametrização do vetor de controle para a obtenção de perfis de variáveis de controle (vazão de refluxo, concentração do produto de fundo, vazão do destilado e concentração do produto de topo) de uma coluna de destilação binária. Os autores obtiveram perfis semelhantes aos perfis obtidos por programação dinâmica iterativa com o objetivo de comparar os resultados.

Feehery e Barton (1999) solucionaram problemas de otimização dinâmica algébrico-diferenciais de índice superior com restrições de igualdade utilizando parametrização da variável de controle. O objetivo dos autores neste trabalho é resolver simultaneamente um sistema de equações algébrico-diferenciais com restrições de igualdade como um problema de valor inicial dentro do método de parametrização da variável de controle. Os autores concluíram que o método de anexar as restrições de igualdade ao sistema de equações algébrico diferenciais, resolvendo o sistema aumentado diretamente é superior aos métodos indiretos que incluem algumas medidas da violação da restrição em Programação não linear.

Oberle e Sothmann (1999) resolveram um modelo de fermentação de alimentação batelada de um processo que descreve a biosíntese de penicilina. Os autores aplicaram a discretização da variável de controle representando-a por uma função linear por partes, sendo o sistema de equações diferenciais solucionado pelo código RADAU5 (HAIRER, WANNER, 1999). Também foram geradas as condições necessárias para a otimalidade através da aplicação do Princípio do Mínimo de Pontryagin gerando um problema de valor no contorno, solucionado pela técnica de múltiplo chute.

Canto et al. (2002) apresentaram a diferenciação original das equações de sensibilidade de segunda ordem para sistemas de EAD de índice um, realizando a diferenciação das equações de sensibilidade de 1ª ordem (derivada do sistema de EAD em relação aos parâmetros invariantes no tempo). Na parametrização do vetor de controle, o problema de otimização dinâmica original é transformado em problema de programação não linear de dimensão finita onde as variáveis de decisão são os parâmetros invariantes no tempo, definindo a discretização do controle. Neste trabalho foi considerada a parametrização do controle constante por partes definido por elementos de comprimento fixo (malha uniforme). Foi mostrado como um problema de programação não linear pode ser resolvido com eficiência utilizando o gradiente exato e o produto vetorial da matriz Hessiana obtida através da solução de um problema de valor inicial aumentado, que é formado pelas equações diferenciais originais e as equações de sensibilidade de primeira e segunda ordem.

Hadiyanto et al. (2008) aplicaram a técnica de parametrização do vetor de controle com sensibilidade baseada em refinamento no estudo de otimização de processos de panificação da indústria alimentícia. Com o propósito de diminuir o esforço computacional relacionado a um maior refinamento da malha de controle, os

autores propuseram começar a otimização com um baixo número de parâmetros. Quando a otimização atinge um determinado patamar, um refinamento do tamanho do passo é aplicado na próxima iteração para encontrar um melhor desempenho. Em seus estudos, os autores constataram que existem muitos intervalos da variável de controle onde o ajuste dos parâmetros não apresenta efeito significativo na melhoria do índice de desempenho. Desta forma foi utilizado um critério de seleção dos parâmetros calculando um valor limite para a sensibilidade (derivada da Função Objetivo em relação a um determinado parâmetro de controle) separando os parâmetros de controle em dois grupos, sendo um grupo com os parâmetros acima de um valor limite e outro grupo com os parâmetros abaixo de um valor limite. Os parâmetros acima do valor limite são mantidos no processo de otimização e os valores abaixo deste valor limite são excluídos.

2.5.2. *Métodos Simultâneos*

Os Métodos Simultâneos discretizam as variáveis de estado e de controle de um sistema de equações algébrico-diferenciais resultando num problema de Programação Não Linear (NLP).

Como resultado, esses métodos agrupam a solução do sistema de equações algébrico- diferenciais com o problema de otimização; o sistema de equações algébrico diferenciais é resolvido somente uma vez, no ponto ótimo, e, portanto podem evitar soluções intermediárias que podem não existir ou podem requerer grande esforço computacional. (BIEGLER, 2007).

Dentre algumas das vantagens dos métodos simultâneos, são apresentadas (BIEGLER E GROSSMAM, 2004):

- as variáveis de controle podem ser discretizadas com um mesmo nível de precisão das variáveis de estado algébricas e diferenciais;
- podem evitar soluções intermediárias que são difíceis de obter ou requerem excessivo esforço computacional.

A seguir são apresentadas as formas para discretização das variáveis de controle e de estado:

$$z_{k+1}(t) = \sum_{j=0,1}^K z_{ij} \phi_j(t) \quad (2.103)$$

$$\phi_j(t) = \prod_{k=1,j}^K \frac{t - t_{ik}}{t_{ij} - t_{ik}} \quad (2.104)$$

$$u_k(t) = \sum_{j=1}^k u_{ij} \theta_j(t) \quad (2.105)$$

$$\theta_j(t) = \prod_{k=1,j}^K \frac{t - t_{ik}}{t_{ij} - t_{ik}} \quad (2.106)$$

Sendo K o número de pontos de colocação e i é o número de elementos finitos.

Na literatura, apresentam-se trabalhos na década de 80 como, Biegler (1984), Li e Biegler (1988) e Cuthrell e Biegler (1989) obtendo a solução de problemas com restrição nas variáveis de controle e estado utilizando o algoritmo de Programação Quadrática Sucessiva (SQP). Cuthrell e Biegler (1989) apresentaram um método de otimização simultânea para a solução de casos envolvendo a otimização de um reator de alimentação batelada de produção de penicilina. Os problemas envolviam restrições de estado com estratégia de solução baseada em programação quadrática sucessiva e colocação ortogonal sobre elementos finitos com a proposta de manipular os perfis de controle descontínuos. Uma solução com base analítica, através de uma discretização que é análoga às equações adjuntas resultantes da teoria de controle ótimo é apresentada. São apresentadas também quatro soluções numéricas para o problema. As soluções numéricas foram obtidas de acordo com o seguinte esquema: Primeiramente um perfil de controle contínuo foi utilizado juntamente com as condições iniciais do modelo e o tempo inicial da batelada, integrando o modelo utilizando um solucionador de Equações Diferenciais Ordinárias. Na sequência o número de pontos de colocação e o número e a localização dos elementos finitos foram escolhidos. Os coeficientes polinomiais do perfil de estado são inicializados através dos perfis contínuos. Os coeficientes polinomiais nos nós são computados através da solução do sistema linear de equações contínuas. As soluções obtidas foram ligeiramente melhores que soluções

reportadas da literatura. A qualidade da precisão das aproximações do modelo demonstrou a eficiência da utilização de colocação ortogonal e polinômios de Lagrange.

Na década de 90, Cervantes e Biegler (1998) apresentaram a resolução de problemas de otimização dinâmica algébrico- diferenciais resolvendo o problema de programação não linear resultante da aplicação o método de colocação sobre elementos finitos.

Mais recentemente, alguns trabalhos com aplicação específica foram apresentados como, por exemplo, Zavala, Flores-Tlacuahuac e Vivaldo-Lima (2005) realizaram a otimização dinâmica de um reator de copolimerização de poliuretano. As equações algébrico-diferenciais do problema de otimização foram resolvidas utilizando um método de discretização simultânea. Os autores aproximaram os perfis das variáveis diferenciais, controle e algébrica por uma representação de base monomial. O principal objetivo do controle do processo de operação do reator é a maximização do peso molecular do copolímero para um tempo de batelada fixo, evitando atingir o ponto de gelificação. Rohman e Aziz (2011) utilizaram o método de colocação ortogonal aplicado para obter as condições de funcionamento ótimo de um processo de eletrodialise. Os autores utilizaram colocação ortogonal sobre elementos finitos para parametrizar ambas as variáveis de estado e de controle utilizando programação quadrática sequencial para resolver o problema de otimização paramétrico resultante.

2.6. Programação Dinâmica Iterativa

A programação dinâmica iterativa tem atraído a atenção de muitos pesquisadores devido a muitas propriedades favoráveis: é de fácil implementação, robusto, não envolve a solução de programação não linear (NLP), é colocado como capaz de encontrar o ótimo global e não requer qualquer diferenciação das equações do processo. A principal desvantagem do método é a alta dimensão do problema o que acarreta um grande esforço computacional. (RUSNÁK et al., 2001).

Para a apresentação do algoritmo da Programação Dinâmica Iterativa, considere o sistema dinâmico contínuo representado pelo vetor de equações diferenciais:

$$\frac{dx}{dt} = f(x, u, t) \quad (2.107)$$

Com o estado inicial $x(0)$ dado, onde x é um $(n \times 1)$ vetor de variáveis de estado e u é um $(m \times 1)$ vetor de variáveis de controle limitado por:

$$\alpha_j \leq u_j \leq \beta_j, \quad j = 1, 2, \dots, m \quad (2.108)$$

Luus (2000) apresenta o algoritmo para a Programação Dinâmica Iterativa com o uso do controle constante por partes sobre P estágios, cada um de mesmo comprimento. Os 10 passos desse algoritmo são apresentados na sequência:

1. dividir o intervalo de tempo $[0, t_f]$ em P estágios de tempo, cada um de comprimento L ;
2. escolher o número de valores teste para u denotado por R , uma política de controle inicial e a região inicial de tamanho r_{in} ; escolher também a região do fator de contração γ usada após cada iteração e o número de pontos de malha N ;
3. escolher o número de iterações a ser usado em cada passagem e definir o índice de iteração $j=1$;
4. definir o tamanho da região pelo vetor $r^j = r_{in}$;
5. usando a melhor política de controle (a política de controle inicial para a primeira iteração) integrar a Equação (2.107) de $t=0$ a t_f N vezes com diferentes valores para o controle para gerar N x -trajetórias e armazenar os valores de x , no início de cada estágio de tempo, de modo que $x(k-1)$ corresponda ao valor de x no início do estágio k ;
6. começando no estágio P , correspondente ao tempo $t_f - L$, para cada um dos N valores armazenados de $x(P-1)$ através do passo 5 (pontos da malha) integrar a Equação (2.107) de $t_f - L$ até t_f com cada um dos valores de R permitidos para o vetor de controle calculado através de:

$$u(P-1) = u^{*j}(P-1) + D r^j \quad (2.109)$$

Sendo $u^{*j}(P-1)$ o melhor valor obtido na iteração anterior e D uma matriz diagonal de diferentes números aleatórios entre 1 e -1. Para valores de R fora do índice de desempenho, são escolhidos os valores de controle que fornecem o valor mínimo, e armazenam esses valores como $u(P-1)$. Têm-se agora o melhor controle para cada um destes N pontos da malha;

7. retornar ao estágio P-1, correspondente ao tempo $t_f - 2L$, e para cada um dos N pontos da malha fazendo os seguintes cálculos: escolher R valores para $\mathbf{u}(P-2)$ como no passo anterior, e tomando como estado inicial $\mathbf{x}(P-2)$ integrar a Equação(2.107) sobre um estágio de comprimento. Continuar a integração ao longo do último estágio de tempo, utilizando o valor armazenado de $\mathbf{u}(P-1)$ a partir do passo 6 que corresponde ao ponto da malha mais próximo do vetor de variáveis de estado alcançado. Comparar os valores de R do índice de desempenho e armazenar $\mathbf{u}(P-2)$ que fornece o menor valor para o índice de desempenho;
8. continuar o procedimento até que o estágio 1, que corresponde ao tempo inicial $t=0$, e o estado inicial dado sejam alcançados. Este estágio tem apenas um único ponto de malha, uma vez que o estado inicial está especificado. Novamente integrar a equação (2.107) comparando os valores de R do índice de desempenho e armazenando o controle $\mathbf{u}(0)$ que fornece o mínimo do índice de desempenho. Armazenar também a x-trajetória correspondente;
9. reduzir a região para as variáveis de controle permitidas

$$\mathbf{r}^{j+1} = \gamma \mathbf{r}^j \quad (2.110)$$

sendo j o número da iteração;

10. aumentar o número da iteração j para 1 retornando para o passo 5 e continuar o procedimento para determinar o número de iterações.

Na seqüência são apresentadas aplicações da Programação Dinâmica.

Luus e Rosen (1991) analisaram a influência do tamanho do fator de penalidade na solução de três problemas de Otimização Dinâmica com restrição de estado final e concluíram que a taxa de convergência do algoritmo pode ser afetada pelo tamanho do fator penalidade. Foi observado que para um fator penalidade muito grande, a convergência para o ótimo é baixa e um grande número de valores teste para $\mathbf{u}$ é requerido. Se o fator penalidade é razoavelmente pequeno, a taxa de convergência é maior e um número menor de valores testes para a variável de controle $\mathbf{u}$ é requerido.

Luus (1992) considerou a otimização de fermentadores de alimentação batelada com restrições de estado, manipuladas por funções penalidade. Os resultados obtidos para a solução de um problema com tempo final livre e restrições de trajetória nas variáveis de estado são comparados com os obtidos pelo Método Direto utilizando

colocação ortogonal e Programação não Linear realizado por outros autores. Os resultados de Função Objetivo obtidos utilizando Programação Dinâmica Iterativa foram 0,3% superiores aos resultados obtidos por outros autores utilizando colocação ortogonal e Programação não Linear.

Luus (1994) escolheu quatro sistemas de reatores batelada para examinar a viabilidade da utilização de Programação Dinâmica Iterativa (PDI) para sistemas altamente não lineares em engenharia química. Para o primeiro problema solucionado, o autor comparou o número de estágios de controle para o controle constante por partes e linear por partes e observou que com um menor número de estágios de controle linear por partes, o resultado é mais satisfatório devido a natureza suave do perfil de controle. Para o segundo problema solucionado, foi observada a influência do número de estágios de controle constantes por partes no valor da Função Objetivo sendo que o melhor resultado foi obtido para 22,33 e 44 estágios. O objetivo no terceiro problema solucionado é a obtenção do perfil da temperatura de controle de uma reação que maximize um dado produto de interesse em um tempo final a ser determinado. Foi analisada a influência de diferentes números de valores testes para a variável de controle, considerando um determinado tempo final, na convergência do algoritmo para um ótimo global. Foi observado que quando maior o número de valores testes para a variável de controle, maior a possibilidade de se encontrar um ótimo local. O objetivo no quarto problema solucionado é encontrar a temperatura de um reator como uma função do tempo, tal que o rendimento de um dado produto de interesse em um conjunto de reações paralelas seja maximizado. Foi analisada a influencia do número de pontos de malha N na obtenção do ótimo global, observando que para um número maior de pontos de malha, maior a possibilidade de se encontrar o ótimo global.

Ha e Rhee (2002) utilizaram Programação Dinâmica Iterativa para determinar as condições de reação ótimas para minimizar o custo de energia e as quantidades de subprodutos para um processo de produção de poli(etileno tereftalato). O sistema em estudo consiste de três modelos de reação, modelo para a reação de transesterificação semi-batelada, modelo para a reação de prepolimerização semi-batelada e modelo de reação de policondensação. As variáveis de controle consideradas são as temperaturas de reação para o primeiro e segundo reator e a temperatura e a pressão para o terceiro reator. Os autores concluíram que o algoritmo de Programação Dinâmica Iterativa foi aplicado com sucesso para a determinação dos objetivos.

Luus (2009) solucionou um problema de reator batelada não isotérmico com restrição na remoção de calor, apresentando em uma primeira solução a taxa de alimentação e a temperatura de reação como variáveis de controle. O problema foi então reformulado apresentando como variáveis de controle a taxa de alimentação e o calor produzido pela reação. O objetivo é obter os perfis das variáveis de controle que maximizem a produção de um produto desejado em uma reação consecutiva. Na solução, o intervalo de tempo foi dividido em estágios de tempo de comprimento variável, sendo as variáveis de controle e o comprimento de cada estágio de tempo as variáveis de otimização. A solução do problema considerando o calor produzido pela reação como variável de controle apresentou um rendimento superior do produto de interesse.

2.7. Métodos Híbridos

O desenvolvimento de métodos híbridos de solução de problemas de controle ótimo surge para sobrepor as dificuldades inerentes aos métodos indiretos e diretos, aliando as vantagens dos dois métodos na busca de uma solução ótima para o problema de controle.

Enquanto os Métodos Indiretos que utilizam a solução numérica usando chutes múltiplos apresentam dificuldades em encontrar uma estimativa inicial apropriada para as variáveis adjuntas, os Métodos Diretos podem apresentar resultados de baixa precisão.

Stryk e Bulirsch (1992) combinaram o método direto de colocação com a aplicação do método de chute múltiplo para a solução de problemas de controle ótimo. Pela abordagem híbrida apresentada pelos autores, os pontos de discretização obtidos pelo método direto foram uma boa escolha para a posição dos nós no método de chute múltiplo. Também pelo método direto, foram obtidas estimativas para as variáveis adjuntas. A abordagem híbrida foi utilizada para a obtenção da mínima quantidade de calor acumulado na trajetória de descida da cápsula da nave Apollo. O desvio obtido para o valor da função objetivo e do tempo final através da solução pelo método direto comparado com o método de múltiplo chute foi de cerca de 1%.

Bulirsch et al. (1993) aplicaram a abordagem híbrida para determinar o máximo alcance de voo de uma asa delta. Primeiramente, os autores utilizaram a abordagem indireta na tentativa de construir uma trajetória inicial utilizando o método

do chute múltiplo. Para isso foram utilizadas técnicas de homotopia onde é gerada uma família de subproblemas em que a solução de um problema serve como uma estimativa inicial para o problema vizinho. Na sequência é aplicado o método direto de colocação associado com técnicas de homotopia devido a não obtenção direta da convergência para o modelo completo. Por fim os autores concluem que o método direto de colocação pode ser utilizado para estimar as variáveis adjuntas através dos parâmetros de Lagrange em um problema de programação não linear. Para problemas de grau de complexidade moderado, as aproximações de ambas variáveis de estado e variáveis adjuntas fornecidas pelo método direto de colocação são suficientemente precisas para gerar convergência com o método do chute múltiplo.

Lobato (2004) utilizou um método híbrido denominado de método misto na solução de problemas de otimização dinâmica algébrico-diferenciais. No método proposto, o problema original é resolvido inicialmente pelo código computacional DIRCOL (VON STRYK, 1999), buscando uma estimativa dos eventos e do perfil das variáveis adjuntas. O código DIRCOL é um conjunto de sub-rotinas implementado em Fortran com o objetivo de solucionar POD descrito por equações diferenciais de primeira ordem com restrições de igualdade e desigualdades nas variáveis de controle e ou estado. O DIRCOL utiliza um método de colocação direto que discretiza ambas as variáveis de estado e controle por aproximações polinomiais por partes transformando o POD em um problema de Programação não linear solucionado com Programação Quadrática Sequencial (SQP). Com a estimativa dos eventos e análise dos resultados obtidos, são construídas as Funções Identificadoras de Fases (FIF). Os problemas de valor no contorno referentes a cada fase são resolvidos simultaneamente pelo código COLDAE. A subrotina COLDAE implementada em Fortran por ASCHER e SPITERI (1994) tem capacidade de resolver EAD não lineares semi explícitos de índice de no máximo 2 e totalmente implícitos de índice 1. A COLDAE implementa o método de colocação direta em pontos Gaussianos oferecendo opções de seleção do método de projeção da forma de inicialização do sistema, do tamanho da malha e inserção de pontos fixos na malha. Foram solucionados problemas sem restrições, com restrições de igualdade, com restrições de desigualdade, com restrições de igualdade e desigualdade, problemas com restrição de fim e tempo final livre. O autor concluiu que para os problemas sem restrições e com restrições algébricas no sistema original, não houve nenhuma dificuldade de convergência para o método direto ou para o método indireto, independentemente da estimativa do perfil inicial dos valores das variáveis adjuntas,

exceto para um problema de um reator de fluxo pistonado (PFR) onde devido ao mau condicionamento das equações, foi necessário definir as variáveis adjuntas como funções lineares da posição, obtidas a partir dos valores estimados pela DIRCOL. Nos problemas que apresentavam restrição algébrica de desigualdade no sistema original, os valores das variáveis adjuntas estimados pelo método direto foram usados para estabelecer a ordem de grandeza dos valores constantes usados como estimativa inicial para o método indireto.

Pfeifer (2007) apresentou a solução de problemas de otimização dinâmica algébrico-diferencial chaveados com restrições de desigualdade. POD chaveado é um tipo particular de POD híbrido cujo estado contínuo não apresenta saltos nos eventos. Neste trabalho foram resolvidos três estudos de caso, um POD chaveado e dois problemas de otimização dinâmica algébrico-diferencial de reatores batelada onde a variável de controle é a taxa de alimentação de um determinado componente. Para a solução numérica dos três estudos de caso, o método direto foi implementado no código DIRCOL estendido para formulações multifásicas com estimativa dos Eventos e o método indireto com parametrização dos eventos e abordagem algébrico-diferencial foi implementado em um código no software MATLAB.

Maurer e Pesch (2008) analisaram e resolveram um complexo problema de controle ótimo em microeconomia. O problema em questão apresenta 4 variáveis de controle lineares no sistema de equações diferenciais, além de muitas restrições de desigualdade. Primeiramente, o problema é discretizado sendo resolvido por um método de Programação Não Linear (NLP). Em um segundo momento ocorre o refinamento do passo, onde em adição às variáveis de controle e estado discretizadas, a junção entre os tempos dos arcos bang-bang e singular são otimizados. Segundo os autores, as soluções computadas mostraram satisfazer precisamente as condições de otimalidade necessárias do Princípio do Máximo onde as restrições de estado são diretamente acopladas a função *Hamiltoniana*.

2.8. Métodos de Otimização Naturais

O aparecimento de métodos heurísticos que simulam fenômenos naturais datam dos anos de 1950. Uma característica comum a esses métodos de otimização é o caráter aleatório de busca do ótimo. Apenas na década de 1980, com um maior

desenvolvimento computacional, esses métodos passaram a ser empregados para a otimização de processos. Dentre os métodos de otimização heurísticos inspirados na natureza tem-se: Recozimento Simulado, Algoritmos Genéticos, Evolução Diferencial, Colônia de Formigas, Enxame de Partículas, etc.

Enquanto métodos de busca determinísticos partem de um ponto inicial na busca de uma solução ótima, e baseado em regras pré-especificadas o algoritmo indica uma direção através de informações locais (OLIVEIRA, 2006), nos métodos heurísticos uma população é gerada aleatoriamente e cada indivíduo dessa população é avaliado (exceto Recozimento Simulado).

Dentre algumas das vantagens dos métodos heurísticos em relação aos métodos determinísticos tem-se (BASTOS, 2004):

- não requerem que a Função Objetivo seja contínua ou diferenciável;
- são aplicáveis tanto a problemas com parâmetros contínuos, quanto com parâmetros discretos ou ainda uma combinação deles;
- não há restrição alguma quanto ao ponto de partida dentro do espaço de busca da solução;
- são de fácil implementação computacional.

Entretanto, os métodos heurísticos apresentam uma importante desvantagem em relação aos métodos determinísticos. Por serem considerados métodos de ordem zero, ou seja, não se faz o uso de derivadas da Função Objetivo, métodos heurísticos tendem a apresentar um grande número de avaliações da Função Objetivo sendo necessário muitas das vezes um grande número de iterações para se encontrar a solução ótima.

No Capítulo 3 será apresentada uma revisão de alguns métodos naturais, incluindo o algoritmo de Evolução Diferencial.

Capítulo 3

REVISÃO DE MÉTODOS DE OTIMIZAÇÃO NATURAIS

3.1. Introdução

Os Métodos de Otimização Naturais são métodos de busca do ótimo inspirados em processos que ocorrem na natureza. Dentre estes métodos, estão o algoritmo de Recozimento Simulado (*Simulated Annealing*), que é um algoritmo metaheurístico de busca estocástica que faz uma analogia com o processo de recozimento (*annealing*) da metalurgia. Solução de problemas de Otimização Dinâmica utilizando o algoritmo de Recozimento Simulado podem ser encontrados em HANKE e LI (2000), SARKAR e MODAK (2003) e FABER, JOCKENHÖVEL e TSATSARONIS (2005).

Dentre os métodos com base em população destacam-se os algoritmos Evolutivos (Algoritmos Genéticos, Evolução Diferencial,) que são uma classe de técnicas estocásticas baseadas nas leis da evolução natural das espécies e os algoritmos baseados em inteligência coletiva (Colônia de Formigas, Enxame de Partículas).

O algoritmo de otimização por enxame de partículas é um algoritmo de busca baseado em população com base na simulação do comportamento social de enxame de pássaros, peixes ou insetos. O algoritmo mantém um enxame de partículas, onde cada partícula representa uma solução potencial. Analogamente aos algoritmos evolutivos, um enxame é similar a uma população e uma partícula é similar a um indivíduo (ENGELBRECHT, 2007). No início do processo de busca, uma população de soluções candidatas é criada de forma aleatória no espaço de soluções. Cada partícula está associada a uma velocidade e esta velocidade é ajustada iterativamente de acordo com a “experiência” da partícula e “experiência das partículas que a acompanham. Espera-se assim que as partículas irão se mover para as áreas com as melhores soluções. A aptidão de cada partícula pode ser avaliada de acordo com a Função Objetivo do problema de otimização. (MODARES e SISTANI, 2011)

Dentre alguns trabalhos utilizando a técnica de Enxame de Partículas na resolução de problemas de otimização dinâmica apresentam-se:

Zhang, Xie e Whang (2006) apresentaram uma abordagem para otimização dinâmica com base no princípio de otimização por enxame de partículas. A abordagem apresentada foi implementada no software Matlab® e aplicada para otimizar o indutor e os perfis de alimentação do substrato de um biorreator batelada com o objetivo de maximizar a quantidade produzida de cloranfenciolacetiltransferase.

Skolpap et al. (2008) implementaram o algoritmo de otimização por enxame de partículas para selecionar conjuntos de variáveis de decisão para perfis de alimentação ótima de alimentação batelada de uma cultura de *Bacillus subtilis* recombinante. O algoritmo foi empregado para otimizar a produção volumétrica de α -amilases extracelular recombinante como produto desejado e proteases nativas como produto indesejado. O modelo resolvido consistia de 14 equações diferenciais não lineares. Os autores resolveram o mesmo problema utilizando algoritmos genéticos. Os resultados obtidos foram comparados com resultados experimentais e com resultados obtidos pela aplicação de uma técnica de resolução das equações diferenciais por colocação ortogonal. Os resultados obtidos pelos algoritmos genéticos e otimização por enxame de partículas foram similares sendo superiores em 18% e 3,5% aos resultados experimentais e os resultados obtidos pela técnica de colocação ortogonal respectivamente.

Ferrari, Gutierrez e Biscaia Jr (2010) utilizaram o algoritmo de otimização por enxame de partículas para desenvolver uma estratégia de operação ótima a fim de minimizar a demanda total de aeração ao longo da operação de um reator batelada sequencial. Os autores utilizaram dois tipos de estruturas para a técnica de otimização por enxame de partículas: otimização por enxame de partículas misto-inteiro (mi-PSO) e otimização por enxame de partículas relaxado (r-PSO). A diferença entre as duas estruturas é que enquanto o mi-PSO restringe a natureza de entrada a cada interação no movimento do enxame, r-PSO permite o movimento relaxado da partícula introduzindo a natureza misto-inteiro apenas quando a função objetivo é computada. Os resultados obtidos pela otimização por enxame de partículas foram comparados com resultados obtidos utilizando um método sequencial direto. Em ambas estruturas mi-PSO e r-PSO, os resultados obtidos foram superiores aos obtidos pelo método sequencial direto.

Santos, Lobato e Malagoni (2012) solucionaram um problema de otimização dinâmica de um reator semi-batelada sujeito a limitações na pureza do produto final e em questões de segurança industrial do processo. Os resultados obtidos pelo algoritmo

de enxame de partículas foi comparado com resultados obtidos por outros dois métodos de otimização naturais, colônia de abelhas e colônia de formigas. Os autores concluíram que o algoritmo de enxame de partículas, assim como os outros métodos, foi capaz de estimar satisfatoriamente o perfil de alimentação do reator semi-batelada, quando comparado aos resultados relatados na literatura.

Na sequência deste Capítulo será considerada uma revisão mais detalhada dos métodos evolutivos (algoritmos genéticos e Evolução Diferencial), e Colônia de Formigas com aplicações das técnicas em POD.

3.2. Algoritmos Genéticos

O esquema de representação clássica de um Algoritmo Genético são vetores binários de comprimento fixo. No caso de um espaço de busca n_x -dimensional, cada indivíduo consiste de n_x -variáveis, com cada variável codificada como uma sequência de bits. Se as variáveis têm valores binários, o comprimento de cada cromossomo é n_x bits. Na Figura 3.1 é apresentada uma codificação de um cromossomo que possui n parâmetros com cinco bits cada (adaptado de Oliveira, 2006)

$$cromossomo_d = \left[\begin{array}{ccccc} \overbrace{11010}^{x_1} & \overbrace{01100}^{x_2} & \dots & \overbrace{01010}^{x_i} & \dots & \overbrace{00101}^{x_n} \\ \underbrace{}_{gene_1} & \underbrace{}_{gene_2} & & \underbrace{}_{gene_i} & & \underbrace{}_{gene_n} \end{array} \right]$$

Figura 3.1: Representação de um cromossomo com n genes de 5 bits cada

Caso as variáveis sejam valores nominais, cada valor nominal pode ser codificado como um vetor de bits n_d -dimensional onde 2^{n_d} é o número total de valores nominais discretos para aquela variável. Para a resolução de problemas de otimização com variáveis contínuas, o espaço de busca de problema contínuo pode ser mapeado dentro de um problema de programação discreta. Para isso é necessário converter o espaço binário $\{0,1\}^{n_b}$ para um espaço de variáveis reais $\mathbb{R}^{n_x}$. O domínio do espaço contínuo precisa ser restringido a uma faixa finita, $\{X_{min}, X_{máx}\}$. Um esquema padrão de codificação binária pode ser utilizado para converter um indivíduo de variáveis $x=\{x_1, x_2, \dots, x_j, \dots, x_{n_x}\}$ com $x_j \in \mathbb{R}$, em um indivíduo de variáveis binárias $b=\{b_1, b_2, b_j, \dots, b_{n_x}\}$, onde $b_j = (b_{(j-1)n_d+1}, \dots, b_{jnd})$ com $b_j \in \{0,1\}$ e o número total de bits será igual a $n_b=$

$n \times n_d$. Decodificando cada b_j , a representação que retorna ao ponto flutuante pode ser feita usando a função $\Phi_j: \{0,1\}^{n_d} \rightarrow [x_{min,j}, x_{max,j}]$ de acordo com:

$$\Phi_j(b) = x_{min,j} + \frac{x_{max,j} - x_{min,j}}{2^{n_d} - 1} \left(\sum_{l=1}^{n_d-1} b_{j(n_d-l)} 2^l \right) \quad (3.1)$$

Os Algoritmos Genéticos iniciam com uma escolha aleatória de cromossomos pais através de um espaço de busca para criar uma população. Esta população evolui através dos melhores cromossomos aplicando operadores genéticos que modelam os processos genéticos que ocorrem na natureza (SHOPOVA et al. , 2006).

Os operados genéticos seleção, recombinação e mutação serão apresentados a seguir.

3.2.1. Seleção

A seleção é um dos principais operadores evolutivos relatando diretamente o conceito de Darwin sobre a sobrevivência do mais apto. A proposta deste operador é garantir que as espécies mais aptas perpetuem através de sua descendência e ou tenha grandes chances de ser encontrada na próxima geração (TARCA et al., 2004).

O objetivo principal desse operador é enfatizar melhores soluções. Um pequeno resumo dos principais tipos de operadores é apresentado a seguir:

- seleção aleatória: na seleção aleatória, cada indivíduo tem a mesma probabilidade de ser selecionado. Não se utiliza nenhuma informação sobre a aptidão do indivíduo, significando que os melhores e os piores indivíduos têm exatamente a mesma probabilidade de sobreviverem na próxima geração;
- seleção proporcional: na seleção proporcional, uma distribuição de probabilidade proporcional a aptidão é criada e os indivíduos são selecionados por amostragem;
- seleção por torneio: na seleção por torneio, um grupo de indivíduos é selecionado aleatoriamente através da população. O desempenho dos indivíduos selecionados é comparado e o melhor indivíduo desse grupo é selecionado e retornado pelo operador;
- seleção baseada na posição: a seleção por posição utiliza a posição ordenada dos valores de aptidão para determinar a probabilidade de seleção e não o valor absoluto da aptidão. Assim, esta seleção é independente do atual valor da aptidão, com a vantagem que o melhor indivíduo não irá dominar o processo de seleção;

- elitismo: processo que garante que os melhores indivíduos da população atual sobrevivam na próxima geração. Os melhores indivíduos não passam por mutação sendo copiados para a próxima geração.

3.2.2. Recombinação (Cruzamento)

Através do cruzamento, novos indivíduos são geralmente criados como descendentes de dois pais. Os operadores de cruzamento são classificados com base no esquema de representação usado. Como exemplo, operadores binários específicos têm sido desenvolvidos para representações de cadeias binárias e para operadores específicos de representações de pontos flutuantes.

A maioria dos operadores de cruzamento para representações binárias tem sido aplicada para dois pais selecionados. Dentre alguns dos operadores de cruzamento estão:

- cruzamento em um único ponto: neste cruzamento, segmentos de genes são trocados entre os pais para criar um descendente;
- cruzamento em dois pontos: neste cruzamento duas posições são selecionadas aleatoriamente e a sequência de bits entre esses pontos são trocadas.

Nas Figuras 3.2 e 3.3 é apresentado um esquema com esses dois tipos de cruzamento (adaptado de Engelbrecht, 2007).

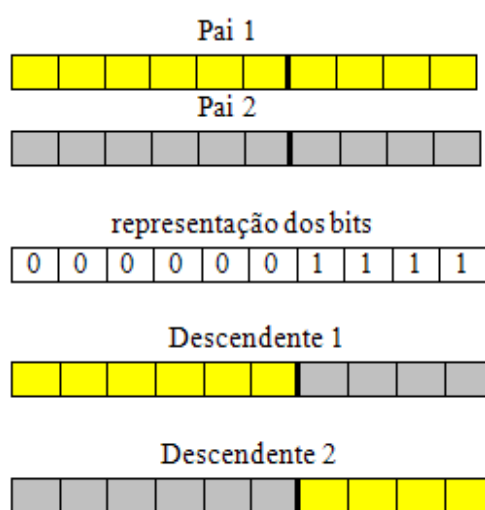


Figura 3. 2-Cruzamento de ponto único

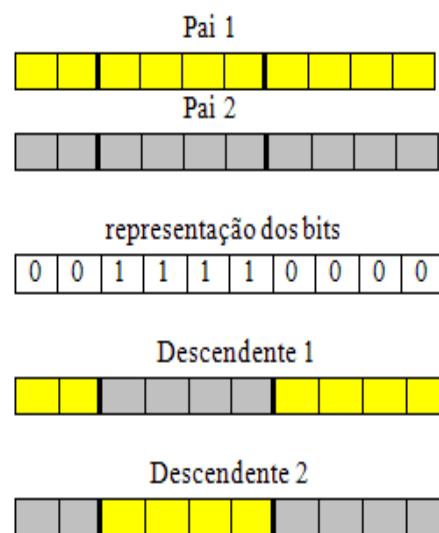


Figura 3.3-Cruzamento de dois pontos

Operadores de cruzamento também podem ser aplicados em representações de pontos flutuantes como estratégias de recombinação discreta. Diferentemente dos

operadores discretos onde a informação é trocada entre os pais, operadores de recombinação intermediária, desenvolvidos especificamente para representações de ponto flutuante, misturam os componentes através dos pais selecionados.

Wright (1991) apud Engelbreth (2007) foi um dos primeiros a propor um operador de cruzamento de ponto flutuante. Através dos pais $x_1(t)$ e $x_2(t)$, três candidatos a filhos são gerados como $(x_1(t) + x_2(t))$, $(1.5x_1(t) + 0.5x_2(t))$ e $(0.5x_1(t) + 1.5x_2(t))$. As duas melhores soluções são selecionadas como filhos.

Deep e Thakur (2007) propuseram um novo operador cruzamento para pontos flutuantes utilizando a distribuição de Laplace. Utilizando este operador, dois descendentes, $y^{(1)} = (y_1^{(1)}, y_2^{(1)}, \dots, y_n^{(1)})$ e $y^{(2)} = (y_1^{(2)}, y_2^{(2)}, \dots, y_n^{(2)})$ são gerados através dos pais $x^{(1)} = (x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)})$ e $x^{(2)} = (x_1^{(2)}, x_2^{(2)}, \dots, x_n^{(2)})$ da seguinte maneira:

Primeiramente um número aleatório $u \in [0,1]$ é gerado. Então, um número aleatório β é gerado pela inversão da função distribuição da distribuição de Laplace:

$$\beta = \begin{cases} a - b \log_e(u), & u \leq \frac{1}{2} \\ a + b \log_e(u), & u > \frac{1}{2} \end{cases} \quad (3.2)$$

Sendo $a \in \mathbb{R}$ um parâmetro local. Os filhos são dados pelas seguintes equações:

$$y_i^1 = x_i^1 + \beta |x_i^1 - x_i^2| \quad (3.3)$$

$$y_i^2 = x_i^2 + \beta |x_i^1 - x_i^2| \quad (3.4)$$

Outros operadores de cruzamento para pontos flutuantes podem ser encontrados em MICHALEWICZ (1992), DEB e AGRAWAL (1995), ONO e KOBAYASHI (1997).

3.2.3. Mutação

No processo de mutação, novo material genético é introduzido em um indivíduo existente, aumentando a diversidade das características genéticas da população. A mutação é aplicada com uma probabilidade p_m em cada gene de um descendente para produzir um descendente modificado.

Dado que cada gene sofra mutação a uma probabilidade $p_m \in [0,1]$, a probabilidade de que um indivíduo sofra mutação será:

$$Prob(\tilde{x}_i(t) \text{ sofrer mutação}) = 1 - (1 - p_m)^{n_x} \quad (3.5)$$

Sendo n_x o número de genes contidos em um indivíduo.

Se o valor da probabilidade atribuída ao operador mutação for baixo demais, ele agirá de forma moderada, e a população não terá diversidade depois de certo número de gerações, estagnando bem rápido devido à convergência genética. No entanto, se o operador receber uma probabilidade alta demais, o algoritmo genético passará a ter um comportamento parecido com um algoritmo aleatório e perderá suas características interessantes (LINDEN, 2006).

Dang e Li (2007) apresentam três operadores de mutação. Para melhor compreensão dos operadores, vamos considerar uma população $P'_g(g) = \{x'_1, x'_2, \dots, x'_n\}$ para n indivíduos dessa população na geração g . Para qualquer indivíduo $x'_i = (x'_{i,1}, x'_{i,2}, \dots, x'_{i,L})$, $i = 1, 2, \dots, n$, a operação de mutação é realizada sobre x'_i que resulta em uma nova geração $x''_i = (x''_{i,1}, x''_{i,2}, \dots, x''_{i,L})$.

- **Mutação Gaussiana:** para um gene de um cromossomo de um indivíduo, é adicionada ao valor original uma variação sujeita a uma distribuição Gaussiana, se o número real aleatório é menor que a probabilidade p_m . Uma operação realizada sobre x'_i resulta em uma nova geração:

$$x''_{i,k} = \begin{cases} x'_{i,k} + \beta_G \xi_k & \text{se } \alpha \leq p_m \\ x'_{i,k} & \text{de outro modo} \end{cases} \quad k = 1, 2, \dots, L \quad (3.6)$$

Sendo $\xi_k \sim N(0, \sigma^2)$ gerada independente de cada gene, σ é o desvio padrão da distribuição Gaussiana, β_G é um parâmetro de escalonamento para a adaptação dos passos de mutação.

- **Mutação Cauchy :** uma função de distribuição Cauchy é dada por:

$$f(x|\alpha_C, \beta_C) = \frac{\beta_C}{\pi[\beta_C^2 + (x - \alpha_C)^2]} \quad (3.7)$$

Sendo α_C e β_C são parâmetros que regulam o ponto central e a largura da distribuição.

- **Mutação na fronteira:** soluções ótimas são geralmente encontradas na fronteira de uma região de solução viável em muitos problemas de otimização. Uma mutação na fronteira tem o intuito de alterar um valor de um alelo para os limites inferiores ou superiores definindo seu campo aleatório. Dado um

indivíduo $x'_i = (x'_{i,1}, x'_{i,2}, \dots, x'_{i,L})$, um gene $x'_{i,k}$ definido sobre o intervalo $[a_k b_k]$ sofre mutação como:

$$x''_{i,k} = \begin{cases} a_k & \text{se } r \leq 0,5 \\ b_k & \text{de outro modo} \end{cases} \quad k = 1, 2, \dots, L \quad (3.8)$$

Outros operadores de mutação podem ser vistos em Deep e Takur (2007).

Dentre alguns dos trabalhos reportados na literatura para a resolução de problemas de Otimização Dinâmica utilizando Algoritmos Genéticos apresentam-se:

Lee et al. (1997) propuseram uma metodologia para obtenção do tempo de transição mínimo e das trajetórias de controle ótimo de entrada durante a partida de um reator de polimerização contínua. A metodologia era dividida em dois níveis:

- No nível superior eram encontrados o contorno superior para o tempo de transição e os valores para as entradas (controle) para o estado estacionário desejado através do algoritmo genético. Os valores para o controle eram mantidos constantes para um intervalo de tempo suficientemente grande.
- No nível inferior era encontrado o tempo de transição mínimo e as trajetórias do controle de entrada simultaneamente. Neste segundo nível, as trajetórias de controle ótimo eram encontradas utilizando a informação através do nível superior e em uma heurística baseada na seguinte informação: no começo, as variáveis de controle na entrada podiam ter alguns valores dentro de uma ampla faixa. Conforme o sistema se aproximava do estado estacionário, as faixas disponíveis para as variáveis de controle na entrada se tornavam cada vez menores, convergindo para alguns valores constantes no estado estacionário desejado. Assim, o tamanho do espaço de busca tinha uma grande redução.

O objetivo dessa divisão em dois níveis foi diminuir o tempo computacional de solução sem perder a propriedade de busca do ótimo global dos algoritmos genéticos. Os autores comparam os resultados obtidos pela aplicação do algoritmo genético (com e sem restrição do espaço de busca) com resultados obtidos pela aplicação da Programação dinâmica Iterativa e Programação Sequencial Quadrática. Foi observado que a aplicação do algoritmo genético com restrição do espaço de busca apresentou o melhor resultado com relação ao tempo mínimo de transição, o tempo computacional e o número de avaliações de Função Objetivo. O pior valor de tempo de transição mínimo encontrado foi através da aplicação da técnica determinística de Programação Quadrática Sucessiva (SQP). Foi demonstrada a alta probabilidade que os métodos

determinísticos possuem em ficarem “presos” em ótimo locais, apesar de requererem um esforço computacional menor e um menor número de avaliações de Função Objetivo.

Sarkar e Modak (2004) determinaram os perfis ótimos da taxa de alimentação de biorreatores batelada com mais de uma taxa de alimentação utilizando algoritmos genéticos. Os dois estudos de caso apresentam singularidade na variável de controle e restrições de desigualdade.

Abo-Hammour et al. (2010) aplicaram um algoritmo genético contínuo para a solução de problemas de controle ótimo. Os autores resolveram três problemas de controle ótimo (Oscilador de Van Der Pol , reator químico com resfriamento e robô de livre flutuação) utilizando um tamanho de população de 600 indivíduos, uma probabilidade de cruzamento de 0,75 e uma probabilidade de mutação também de 0,75. Os autores compararam os resultados obtidos através do algoritmo genético com valores obtidos por técnicas determinísticas (abordagem direta) e abordagem indireta (método do chute). Para os problemas Oscilador de Van Der Pol e robô de livre flutuação, não foram obtidas soluções pela abordagem indireta (método do chute). Em todos os problemas, a soluções obtidas empregando o método não determinístico (algoritmos genéticos) foi superior.

3.3. Colônia de Formigas

Observações do comportamento de formigas reais têm inspirado o desenvolvimento de um grande número de algoritmos baseados em colônias de formigas, utilizados para resolver principalmente problemas de otimização combinatorial definidos sobre espaços de buscas discretos. (ENGELBRECHT, 2007).

Para uma melhor compreensão do algoritmo de colônia de formigas, faz-se necessário o conhecimento do comportamento das formigas. Partindo de um ponto, as formigas têm de decidir qual direção tomar na busca de alimento. Escolhendo aleatoriamente o caminho, as formigas depositam no solo uma substância química denominada feromônio. Após certo período de tempo, a diferença entre a quantidade de feromônio entre os caminhos é suficientemente grande para influenciar a decisão de novas formigas que também estão na dúvida de qual melhor caminho seguir. Assim, as formigas escolhem o caminho com maior quantidade de feromônio.

Dorigo et al. (1996) apresentaram a formulação matemática para o algoritmo de colônia de formigas com o objetivo de resolver o problema do caixeiro viajante. Considere $b_i(t)$ ($i=1, \dots, n$) sendo o número de formigas em um local i no tempo t e considere $m = \sum_{i=1}^n b_i(t)$ sendo o número total de formigas. Cada formiga é um simples agente com as seguintes características:

- escolhe o destino mais próximo com uma probabilidade que é uma função da distância entre os destinos e a quantidade de feromônio sobre os caminhos percorridos;
- cada formiga é obrigada a cumprir as rotas legais, ou seja, não são permitidas transições entre os destinos já visitados até que se complete o percurso completo;
- ao se completar um destino, é estabelecida a presença de feromônio em cada caminho percorrido.

Considere $\tau_{ij}(t)$ como sendo a intensidade de feromônio sobre o caminho (i,j) no tempo t . Cada formiga no tempo t escolhe o próximo destino onde estará no tempo $t+1$. Portanto, se denominarmos uma iteração do algoritmo de colônia de formigas os m movimentos efetuados pelas n formigas no intervalo $(t, t+1)$, após n iterações do algoritmo cada formiga completou um passeio completo. Neste ponto, a intensidade de feromônio é atualizada de acordo com a seguinte fórmula:

$$\tau_{ij}(t+n) = \rho \tau_{ij}(t) + \Delta \tau_{ij} \quad (3.9)$$

Onde ρ é um coeficiente tal que $(1-\rho)$ represente a evaporação do feromônio entre o tempo t e o tempo $t+n$.

$$\Delta \tau_{ij} = \sum_{k=1}^m \Delta \tau_{ij}^k \quad (3.10)$$

Onde $\Delta \tau_{ij}^k$ é a quantidade por unidade de comprimento de feromônio colocado sobre o caminho (i,j) pela k -ésima formiga entre o tempo t e $t+n$ sendo dado por:

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{se a } k - \text{ésima formiga utiliza o caminho } (i,j) \text{ em seu trajeto} \\ 0 & \text{caso contrário} \end{cases} \quad (3.11)$$

Q é uma constante de projeto e L_K é o comprimento do trajeto da k -ésima formiga.

O coeficiente ρ deve ser ajustado para um valor menor que 1 para evitar a acumulação ilimitada de feromônio.

A fim de satisfazer a restrição de que uma formiga visite todos os diferentes destinos, cada formiga é associada a uma estrutura de dados denominada de lista tabu, que salva os destinos já visitados até o tempo t impedindo a formiga de visitar novamente os destinos antes que n iterações sejam concluídas. Quando um passeio estiver concluído, a lista tabu é utilizada para calcular a distância do trajeto percorrido pela formiga. A lista tabu é então esvaziada e a formiga está novamente livre para escolher entre possíveis caminhos até os destinos.

A probabilidade de uma formiga indicar o caminho (i,j) será dada por:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{k \in k_{\text{permitidos}}} [\tau_{ik}(t)]^\alpha [\eta_{ik}]^\beta} & \text{se } j \text{ pertence aos } k \text{ permitidos} \\ 0 & \text{caso contrário} \end{cases} \quad (3.12)$$

Onde α é a ponderação do feromônio ($0 \leq \alpha \leq 1$) e β é a ponderação da informação heurística ($0 \leq \beta \leq 1$), $\eta_{ij} = 1/d_{ij}$ é a visibilidade entre a variável j e a variável i , d_{ij} é a distância Euclidiana entre i e j e τ_{ij} é a intensidade da trilha do caminho (i,j) no tempo t (em $t = 0$ a intensidade da trilha é gerada aleatoriamente com distribuição uniforme).

Dentre algumas aplicações do algoritmo de colônia de formigas na resolução de problemas de otimização dinâmica, apresentam-se:

Rajesh et al. (2001) apresentaram um algoritmo de colônia de formigas para a resolução de seis problemas de otimização dinâmica, sendo que entre eles está o problema de mistura de catalisadores. Pela metodologia apresentada, primeiramente o perfil de controle é aproximado por um perfil linear por partes. Os perfis são gerados aleatoriamente e com esses perfis as equações diferenciais devem ser resolvidas e a Função Objetivo avaliada. As formigas são divididas dentro de uma certa proporção em formigas globais e formigas locais, sendo que somente as formigas globais submetem-se a um processo de busca global. A busca global consiste da atuação de alguns operadores tais como o operador cruzamento e o operador mutação. A busca local é aplicada para a obtenção de uma solução mais precisa e refinada. Os resultados obtidos por este trabalho foram comparados em cada problema com o melhor resultado obtido pela literatura.

Zhang et al. (2005) apresentaram um algoritmo de colônia de formigas iterativo aplicado a otimização dinâmica de processos químicos. A idéia principal neste trabalho é executar iterativamente o algoritmo de colônia de formigas aproximando gradualmente o perfil de controle ótimo. O primeiro passo no algoritmo é discretizar o

intervalo de tempo e a região de controle para que o problema de otimização dinâmica contínuo se torne um problema discreto. Assim, o algoritmo de colônia de formigas é utilizado para buscar o melhor perfil de controle do sistema dinâmico discreto. Por fim, para a obtenção de resultados mais precisos e para uma melhora na robustez do algoritmo, uma estratégia de redução da região de controle foi empregada. Os autores aplicaram o algoritmo em três estudos de caso. Os resultados obtidos foram comparados com resultados obtidos por outro algoritmo de colônia de formigas, Programação Dinâmica Iterativa e algoritmo genético.

3.4. Evolução Diferencial

O algoritmo de Evolução Diferencial (ED) é um método direto de busca estocástico desenvolvido por Storn e Price, que opera sobre uma população de soluções potenciais, aplicando o princípio de sobrevivência do mais apto para gerar uma solução ótima. Computacionalmente, o ED é bastante simples sendo necessária a definição de poucos parâmetros pelo usuário (KAPADI e GUDI, 2004).

Dentre algumas das características do algoritmo de Evolução Diferencial que também são características de outros métodos estocásticos naturais estão:

- capacidade de manipular Funções Objetivo multimodais, não lineares e não diferenciáveis (BABU e ANGIRA, 2006);
- capacidade de solução de problemas de programação não linear mista inteira (ANGIRA e BABU, 2006);
- eficiência quando aplicado na resolução de problemas de otimização tanto estáticos quanto dinâmicos (LOPES CRUZ et. al., 2003);
- não requer uma estimativa inicial, pois a população inicial é escolhida aleatoriamente (CONWAY, 2012);
- possui alta probabilidade de encontrar o ótimo global (CONWAY, 2012).

A Evolução Diferencial utiliza uma população de NP indivíduos em cada geração G, sendo que NP não muda durante o processo de minimização. O vetor população inicial é escolhido aleatoriamente e deve cobrir todo o espaço dos parâmetros. O ED gera novo vetor de parâmetros adicionando a diferença ponderada de dois vetores da população a um terceiro vetor. Esta operação é denominada mutação. Os

parâmetros do vetor mutação são então misturados, com os parâmetros de outro vetor, denominado vetor alvo, gerando assim o vetor experimental. A mistura de parâmetros é freqüentemente referida como operação de cruzamento. Se o vetor experimental levar a um menor valor da Função Objetivo do que o vetor alvo, o vetor experimental substitui o vetor alvo na geração seguinte. Esta operação é denominada seleção. (STORN e PRICE, 1997).

Na sequência serão apresentados os operadores utilizados em Evolução Diferencial. Os operadores de mutação, cruzamento e seleção são utilizados no decorrer das gerações (iterações) buscando melhorar os indivíduos (vetores) da população selecionando os melhores indivíduos para a as próximas gerações.

3.4.1. Mutação

Considere três vetores $X_{r0}^{(q)}$, $X_{r1}^{(q)}$ e $X_{r2}^{(q)}$ ($r0, r1$ e $r2 \in (1..NP)$) da q -ésima geração escolhidos aleatoriamente de uma população de NP indivíduos. NP deve ser maior ou igual a 4 para garantir uma quantidade suficiente de indivíduos para a execução do método. Para a obtenção do vetor doador $V^{(q+1)}$, utiliza-se um par de vetores (X_{r1} , X_{r2}) da q -ésima geração definindo o vetor diferença ($X_{r1} - X_{r2}$). Este vetor diferença é multiplicado por um fator F , sendo denominado diferença vetorial ponderada e será usado para perturbar o terceiro vetor X_{r0} . O fator de perturbação F é um número real positivo pertencente ao intervalo $[0,2]$ e controla a amplitude do vetor diferença.

A mutação pode ser escrita como:

$$V^{(q+1)} = X_{r0}^{(q)} + F \left(X_{r1}^{(q)} - X_{r2}^{(q)} \right) \quad (3.13)$$

A Figura 3.4 apresenta a fundamentação teórica do algoritmo de Evolução diferencial representando a etapa de mutação.

Diferentes estratégias de mutação têm sido relatadas na literatura buscando uma melhora do desempenho do algoritmo de Evolução Diferencial.

Fan e Lampinen (2003) propuseram um operador de mutação denominado mutação trigonométrica. O objetivo dos autores foi a de ajustar o equilíbrio entre a taxa de convergência e a robustez do algoritmo de evolução diferencial. O operador mutação trigonométrica pode ser formulado da seguinte maneira:

$$V^{(q+1)} = \frac{X_{r0}^{(q)} + X_{r1}^{(q)} + X_{r2}^{(q)}}{3} + (p_2 - p_1)(X_{r0}^{(q)} - X_{r1}^{(q)}) + (p_3 - p_2)(X_{r1}^{(q)} - X_{r2}^{(q)}) + (p_1 - p_3)(X_{r2}^{(q)} - X_{r1}^{(q)}) \quad (3.14)$$

$$p_1 = \frac{|f(X_{r0}^{(q)})|}{p'} \quad (3.15)$$

$$p_2 = \frac{|f(X_{r1}^{(q)})|}{p'} \quad (3.16)$$

$$p_3 = \frac{|f(X_{r2}^{(q)})|}{p'} \quad (3.17)$$

$$p' = |f(X_{r0}^{(q)})| + |f(X_{r1}^{(q)})| + |f(X_{r2}^{(q)})| \quad (3.18)$$

Bhowmik et al. (2010) apresentaram um algoritmo de evolução diferencial em que o fator de perturbação F varia de acordo com a informação obtida através das gerações anteriores, não sendo necessária a etapa de cruzamento na evolução diferencial. Na proposta do algoritmo apresentado pelos autores, o fator F é adaptado no decorrer das gerações diferentemente da evolução diferencial clássica onde F é mantido constante no decorrer das gerações.

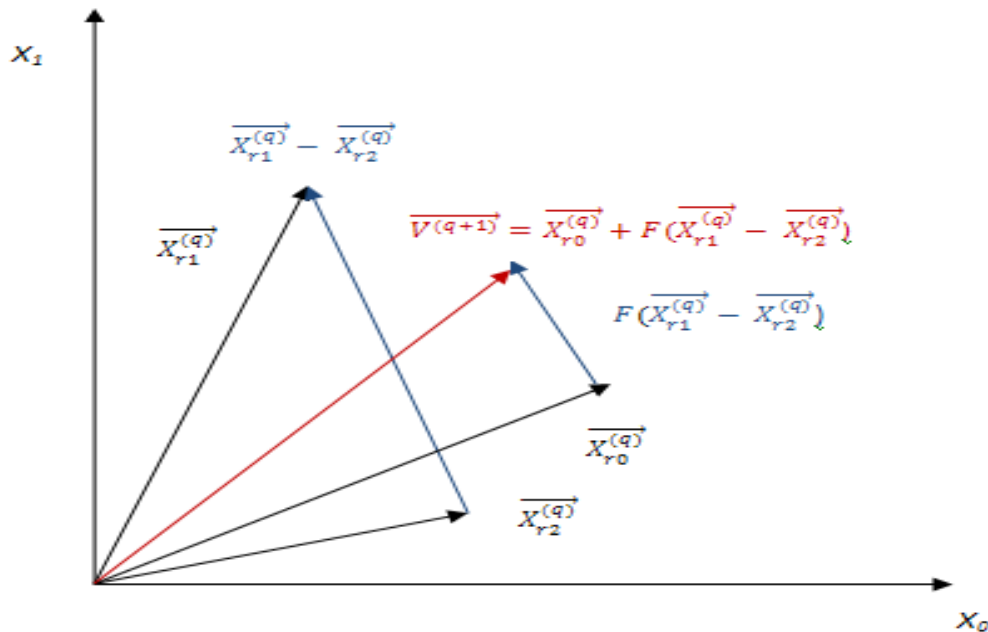


Figura 3.4: Fundamentação Teórica do Algoritmo de Evolução Diferencial

3.4.2. Cruzamento

A operação do cruzamento é introduzida na Evolução Diferencial para aumentar a diversidade dos indivíduos que sofreram a mutação. Aqui, os parâmetros do vetor alvo X_s são misturados com os parâmetros do vetor doador $V^{(q+1)}$ gerando um vetor experimental $U^{(q+1)}$. Em 1995, Storn e Price introduziram o operador cruzamento binomial definido a seguir:

$$u(i)^{(q+1)} = \begin{cases} v(i)^{(q+1)}, & \text{se } rand_i \leq Pc \\ x_s^{(q)}(i), & \text{se } rand_i > Pc, i = 1..n \end{cases} \quad (3.19)$$

Sendo $v(i)^{(q+1)}$ é o $(q+1)$ -ésimo componente do vetor doador $V^{(q+1)}$ e $x_s^{(q)}(i)$ é o q -ésimo componente do vetor alvo X_s , $rand_i$ é um número gerado aleatoriamente no intervalo $[0,1]$, enquanto que Pc pertencente ao intervalo $[0,1]$ é a probabilidade de cruzamento e representa a probabilidade do vetor experimental receber os valores do vetor doador, devendo ser fornecida pelo usuário. O cruzamento binomial será sempre executado em cada variável sempre que um número aleatório $rand \in [0,1]$ for menor que a probabilidade de cruzamento Pc . A Figura 3.5 apresenta a representação do processo de cruzamento binomial (adaptado de Oliveira, 2006).

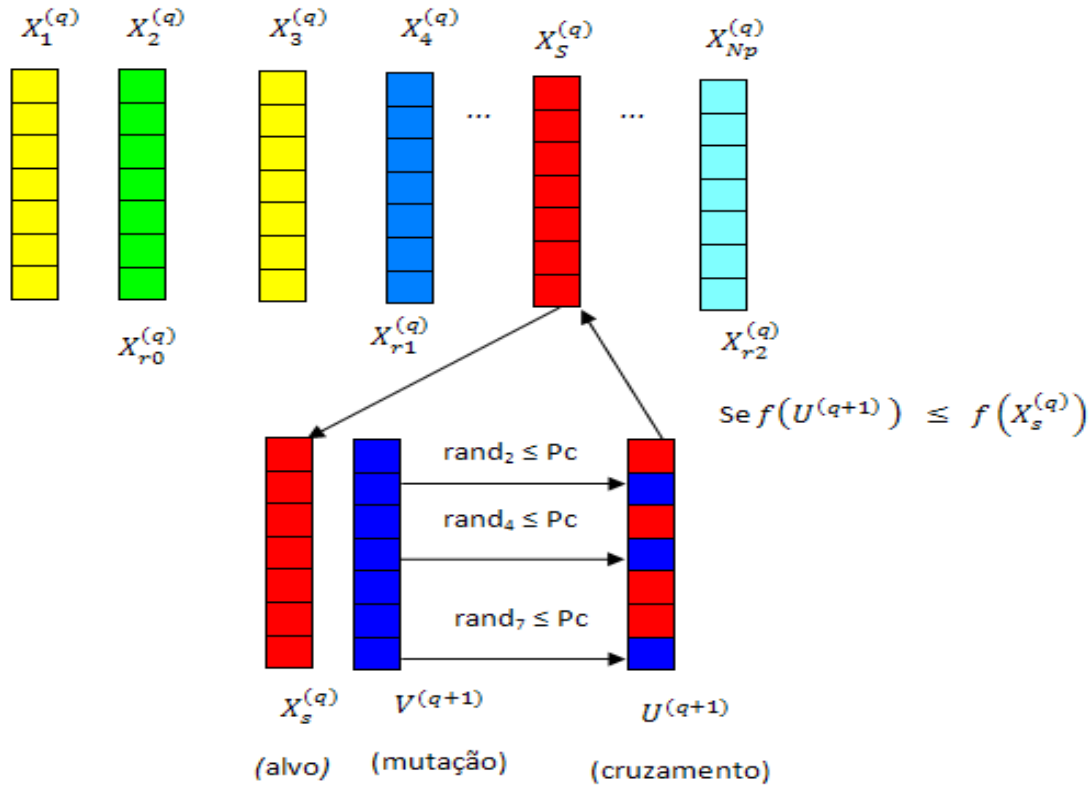


Figura 3.5: Representação do Processo de Cruzamento Binomial

Após dois anos da introdução do cruzamento binomial (1997), Storn e Price introduziram o operador cruzamento exponencial. O operador de cruzamento exponencial é executado nas variáveis enquanto o número aleatório $rand \in [0,1]$ for menor que a probabilidade de cruzamento P_c . Quando o número aleatório ultrapassar o valor de P_c , nenhum cruzamento é executado e as variáveis restantes são deixadas intactas, como podemos ver a seguir:

$$\text{Enquanto } rand_i \leq P_c, u(i)^{(q+1)} = v(i)^{(q+1)}, \quad (3.20)$$

$$\text{Se } rand_i > P_c, u(j)^{(q+1)} = x_s^{(q)}(j), j = (i + 1) \dots n \quad (3.21)$$

A Figura 3.6 apresenta a representação do cruzamento exponencial (adaptado de Oliveira, 2006)

Caso após o cruzamento, um ou mais componentes do vetor experimental estiver fora da região de busca definida pelos limites laterais das variáveis de projeto, devemos ter as seguintes correções:

$$\left\{ \begin{array}{l} \text{Se } u(i) < x(i)^{inf} \text{ então } u(i) = x(i)^{inf} \\ \text{Se } u(i) > x(i)^{sup} \text{ então } u(i) = x(i)^{sup} \end{array} \right. \quad i = 1, \dots, n \quad (3.22)$$

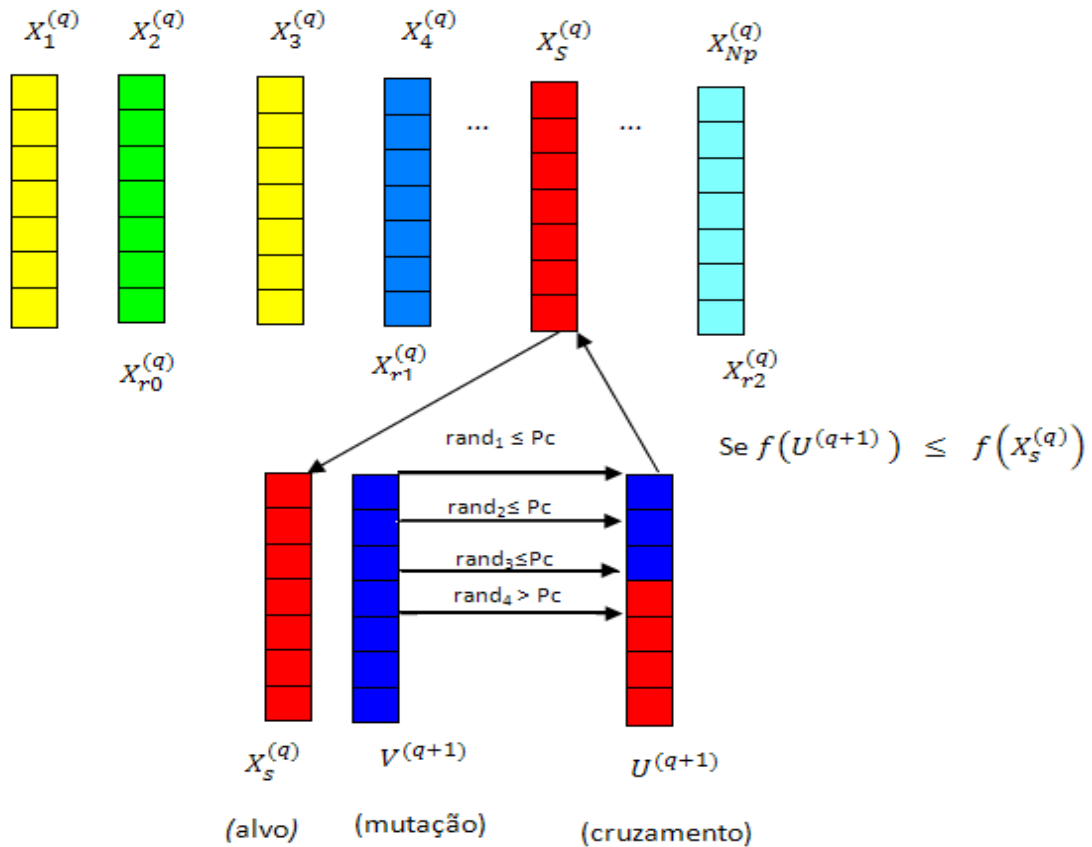


Figura 3.6: Representação do Cruzamento Exponencial

3.4.3. Seleção

A seleção é o processo de gerar melhores descendentes. Na Evolução Diferencial o valor de Função Objetivo do vetor experimental $U^{(q+1)}$ é comparado com o valor de Função Objetivo do vetor alvo $X_s^{(q)}$. Para um processo de minimização, se o valor de Função Objetivo do vetor experimental for menor que o valor de Função Objetivo do vetor alvo, o vetor alvo da próxima geração será o vetor experimental. Caso contrário, se o valor de Função Objetivo do vetor experimental for maior que o valor de Função Objetivo do vetor alvo, o vetor alvo da próxima geração será o vetor alvo da atual geração. Assim, o processo de seleção pode ser descrito como:

$$\left\{ \begin{array}{l} \text{Se } f(U^{(q+1)}) \leq f(X_s^{(q)}) \text{ então } X_s^{(q+1)} = U^{(q+1)} \\ \text{Se } f(U^{(q+1)}) > f(X_s^{(q)}) \text{ então } X_s^{(q+1)} = X_s^{(q)} \end{array} \right\} \quad (3.23)$$

Nos próximos subitens serão apresentas diferentes estratégias utilizadas em Evolução Diferencial e também a descrição passo a passo do algoritmo de Evolução Diferencial.

3.4.4. Estratégias da Evolução diferencial

A Evolução diferencial oferece muitas estratégias para a otimização (MANDAL e CHAKRABORTY, 2012). As estratégias podem variar baseado no vetor a ser perturbado, no número de vetores diferença considerados para a perturbação e pelo operador de cruzamento utilizado.

Elas são classificadas de acordo com a notação:

ED/x/y/z

Sendo:

- x representa o vetor a ser perturbado. Caso este vetor seja escolhido aleatoriamente, este parâmetro da estratégia é classificado como "x=rand". Por outro lado, se o vetor a ser perturbado for o de menor valor de Função Objetivo da população, o parâmetro será representado por "x=best";
- y representa o número de vetores diferença utilizados na perturbação para a obtenção do vetor doador;
- z representa o tipo de cruzamento adotado. Se o cruzamento utilizado for binomial, tem-se o parâmetro "z=bin", se o cruzamento for exponencial, z=exp.

As duas estratégias mais populares são ED/best/1/bin e ED/rand/1/bin (GOUDOS et al., 2011) . Na representação para a obtenção do vetor doador para as duas estratégias tem-se:

$$V^{(q+1)} = X_{best}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)}) \quad (3.24)$$

$$V^{(q+1)} = X_{r0}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)}) \quad (3.25)$$

Lopez Cruz et al, (2003) utilizaram duas diferentes estratégias para o algoritmo de Evolução Diferencial para a resolução de problemas de controle ótimo multimodal.As estratégias utilizadas pelos autores foram :

- ED/rand/1/bin, onde o vetor a ser perturbado é escolhido aleatoriamente, um único vetor diferença é utilizado na perturbação e cruzamento binomial.

- ED/best/2/bin, onde o vetor a ser perturbado é o de menor valor de Função Objetivo da população, dois vetores diferença são utilizados na perturbação e cruzamento binomial.

Babu e Munawar (2007) utilizaram as dez diferentes estratégias (Tabela 3.1, adaptada de Oliveira, 2006) propostas por Price e Storn na obtenção de um projeto ótimo de trocadores de calor casco e tubo.

Tabela 3.1-Representação das estratégias da Evolução Diferencial

Número	Mutação	Notação
1	$V^{(q+1)} = X_{best}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)})$	ED/best/1/exp
2	$V^{(q+1)} = X_{r0}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)})$	ED/rand/1/exp
3	$V^{(q+1)} = X_{old}^q + F(X_{best}^q - X_{old}^q)$	ED/rand-to-best/1/exp
4	$V^{(q+1)} = X_{best}^q + F(X_{r1}^q - X_{r2}^q) + F(X_{r3}^q - X_{r4}^q)$	ED/best/2/exp
5	$V^{(q+1)} = X_{rand}^q + F(X_{r1}^q - X_{r2}^q) + F(X_{r3}^q - X_{r4}^q)$	ED/rand/2/exp
6	$V^{(q+1)} = X_{best}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)})$	ED/best/1/bin
7	$V^{(q+1)} = X_{r0}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)})$	ED/rand/1/bin
8	$V^{(q+1)} = X_{old}^q + F(X_{best}^q - X_{old}^q)$	ED/rand-to-best/1/bin
9	$V^{(q+1)} = X_{best}^q + F(X_{r1}^q - X_{r2}^q) + F(X_{r3}^q - X_{r4}^q)$	ED/best/2/bin
10	$V^{(q+1)} = X_{rand}^q + F(X_{r1}^q - X_{r2}^q) + F(X_{r3}^q - X_{r4}^q)$	ED/rand/2/bin

3.4.5. Algoritmo de Evolução Diferencial

O algoritmo de Evolução Diferencial apresenta os seguintes oito passos para a solução de um problema de otimização:

- Definir os parâmetros Taxa de Perturbação (F), Probabilidade de Cruzamento (Pc), número de gerações (NG), a estratégia a ser utilizada e o número de variáveis de projeto.

- ii. Inicializar a população de candidatos. Antes da inicialização, os contornos inferior e superior ($b_{j,L}$ e $b_{j,U}$) de cada parâmetro devem ser especificados. O valor inicial, geração ($g=0$) do j -ésimo parâmetro do i -ésimo vetor será:

$$x_{j,i,0} = rand_j(0,1) \cdot (b_{j,U} - b_{j,L}) + b_{j,L} \quad (3.26)$$

O número gerador aleatório $rand_j$ retorna uma distribuição uniforme de números aleatórios dentro do intervalo $[0,1]$.

- iii. Escolher um vetor aleatório X_s , por exemplo, ou X_{best} de acordo com a estratégia a ser utilizada para ser substituído (vetor alvo).
- iv. Escolher outros três vetores candidatos X_{r0} , X_{r1} e X_{r2} , $r_0 \neq r_1 \neq r_2$ ou mesmo cinco vetores distintos dependendo da estratégia escolhida em i.
- v. Através da mutação, gera-se um vetor doador $V^{(q+1)}$ de acordo com a estratégia escolhida, por exemplo, através da estratégia 1:

$$V^{(q+1)} = X_{r0}^{(q)} + F(X_{r1}^{(q)} - X_{r2}^{(q)}) \quad (3.27)$$

- vi. Através do cruzamento uniforme, gera-se um vetor $U^{(q+1)}$ a ser comparado a X_{r4} ou X_{best} . A operação de cruzamento se dará de acordo com:

$$u(i)^{(q+1)} = \begin{cases} v(i)^{(q+1)} & \text{se } rand_j < Pc \\ x_s^{(q)}(i) \text{ (ou } x_{best}^{(q)}(i)) & \text{se } rand_j > Pc \end{cases} \quad (3.28)$$

Caso, após o cruzamento, um ou mais componentes de $U^{(q+1)}$ estiver fora da região de busca, faz-se necessária a seguinte correção:

$$\text{Se } u(i) < x(i)^{inf}, \text{ então } u(i) = x(i)^{inf} \quad (3.29)$$

$$\text{Se } u(i) > x(i)^{sup}, \text{ então } u(i) = x(i)^{sup}, i = 1..n \quad (3.30)$$

- vii. Pelo processo de seleção escolhe-se o melhor candidato analisando a função objetivo.

$$\text{Se } f(U^{(q+1)}) \leq f(X_s^{(q)}) \text{ então } X_s^{(q+1)} = U^{(q+1)} \quad (3.31)$$

$$Se f(U^{(q+1)}) > f(X_s^{(q)}) \text{ então } X_s^{(q+1)} = X_s^{(q)} \quad (3.32)$$

- viii. Se algum critério de parada for satisfeito, fim (Os critérios de parada mais comuns são o número máximo de gerações e o número máximo de avaliações da Função Objetivo). Caso contrário, passar para a próxima geração ($q=q+1$) e voltar ao passo iii.

Dentre algumas das aplicações do algoritmo de Evolução Diferencial na solução de POD apresentam-se:

Lopez Cruz, Willigenburg e Straten (2003) apresentaram a solução de dois problemas de controle ótimo multimodais utilizando o algoritmo de evolução diferencial. Primeiramente, os autores resolveram os problemas utilizando a teoria de controle ótimo clássico com o objetivo de mostrar as limitações desta técnica na resolução de problemas multimodais. Em seguida, eles utilizaram duas diferentes estratégias de um algoritmo genético e duas diferentes estratégias de evolução diferencial comparando os resultados obtidos pelos algoritmos evolutivos com resultados obtidos por Programação Dinâmica Iterativa.

Kapadi e Gudi (2004) utilizaram o algoritmo de evolução diferencial para resolver dois estudos de caso envolvendo fermentadores de alimentação batelada com restrições nas variáveis de estado. Os problemas de controle ótimo de dimensão infinita foram transformados em problemas de dimensão finita pela parametrização da variável de controle. O perfil de controle apresentado pelos autores foi aproximado por um valor constante por partes, enquanto que os instantes de atuação de cada estágio de controle foram de tamanhos variáveis. As restrições foram incluídas na função objetivo através de funções penalidade. Neste estudo foram comparados os resultados obtidos considerando a parametrização não uniforme por esse trabalho e resultados obtidos por parametrização uniforme retirados da literatura. Foi observado que os resultados obtidos utilizando parametrização não uniforme foram superiores aos resultados reportados da literatura utilizando parametrização uniforme.

Angira e Santosh (2007) resolveram cinco problemas de controle ótimo não lineares, multidimensionais e multimodais utilizando o algoritmo de evolução diferencial clássico e o algoritmo de evolução diferencial trigonométrica. O objetivo dos autores foi comparar o desempenho do algoritmo de evolução diferencial trigonométrica

com o algoritmo de evolução clássica. Os resultados obtidos pelos dois algoritmos em todos os problemas foram similares, no entanto, o tempo computacional foi no geral maior para algoritmo de evolução diferencial clássica. Os resultados também foram comparados com resultados da literatura e apresentaram boa concordância com os mesmos.

Rahimpour, Parvasi e Setoodeh (2009) otimizaram um reator catalítico de leito esférico de fluxo radial para a síntese de metanol utilizando o algoritmo de evolução diferencial. A otimização do reator foi estudada em quatro abordagens. Na primeira abordagem, o objetivo é obter uma maior produção de metanol através de uma ótima temperatura de entrada no reator. Na segunda abordagem, o objetivo é obter o perfil de temperatura ótimo ao longo do reator durante o período de operação, considerando atingir a taxa de produção máxima de metanol. Na terceira abordagem, o objetivo é otimizar os parâmetros de desativação do catalisador. Por último, o objetivo é a obtenção da razão ótima entre o raio interno e o raio externo do reator para a uma máxima taxa de produção de metanol.

Weschenfelder, Castilhos e Scheer (2012) utilizaram o algoritmo de Evolução Diferencial na otimização dinâmica visando melhorar a programação de tempo-temperatura de um processo de mosturação da cerveja, com base em modelos cinéticos disponíveis na literatura. Os resultados obtidos por Evolução Diferencial foram comparados com resultados obtidos por outros três algoritmos de otimização naturais, colônia artificial de abelhas, enxame de partículas e recozimento simulado. Dos quatro métodos utilizados, a Evolução diferencial foi a que apresentou os melhores resultados.

3.5. Técnicas de manipulação de restrições utilizadas em Algoritmos Evolutivos

A maioria das tarefas de otimização estão envolvidas em encontrar não somente um ótimo, mas também satisfazer uma ou mais restrições. Há muitos caminhos em que um problema de otimização pode ser restringido. O caminho mais comum de incorporar restrições dentro de um algoritmo evolutivo tem sido funções penalidade e serão essas funções abordadas nesta dissertação.

Existem dois tipos de função penalidade consideradas em otimização clássica: exterior e interior. Métodos exteriores iniciam com uma solução inviável que se move

em direção a uma região viável. Nos métodos interiores o termo penalidade é escolhido tal que seu valor será pequeno em pontos fora dos limites da restrição e tenderá a infinito quando esses limites forem abordados.

Coello (2002) apresenta as seguintes funções penalidade:

- penalidade Estática;
- penalidade Dinâmica;
- penalidade recozimento;
- penalidade Adaptativa;
- penalidade Coevolutiva.

3.5.1. Penalidade Estática

Nesta modalidade, o fator penalidade não depende da geração atual, permanecendo constante durante todo o processo evolutivo.

De acordo com a Equação 3.33, define-se $F(\vec{x})$ como a soma da função objetivo $f(\vec{x})$ e o termo penalidade que depende da violação da restrição $\langle g_j(\vec{x}) \rangle$ tem-se:

$$F(\vec{x}) = f(\vec{x}) + \sum_{j=1}^J R_j \langle g_j(\vec{x}) \rangle^2 \quad (3.33)$$

Na Equação 3.33, $\langle \rangle$ representa o valor absoluto e R_j é o parâmetro penalidade da j -ésima restrição de desigualdade. O propósito do parâmetro R_j é fazer a violação da restrição $g_j(\vec{x})$ ser da mesma ordem de magnitude da função objetivo $f(\vec{x})$.

3.5.2. Penalidade Dinâmica

Dentro dessa modalidade considera-se qualquer função penalidade em que o número da geração atual está envolvido no cálculo dos fatores de penalidade, sendo que normalmente a função penalidade aumenta com o número de gerações.

Joines e Houck (1994) propuseram uma técnica em que os indivíduos são avaliados na geração t utilizando:

$$F(\vec{x}) = f(\vec{x}) + (C \times k)^\alpha \times SVC(\beta, \vec{x}) \quad (3.34)$$

Sendo $f(\vec{x})$ a Função Objetivo, k é a geração, C , α e β são constantes definidas pelo usuário (os autores utilizaram $C=0.5$, $\alpha=1$ ou 2 e $\beta=1$ ou 2) sendo o termo $SVC(\beta, \vec{x})$ definido como:

$$SVC(\beta, \vec{x}) = \sum_{i=1}^n D_i^\beta(\vec{x}) + \sum_{j=1}^p D_j(\vec{x}) \quad (3.35)$$

$$D_i(\vec{x}) = \begin{cases} 0, & g_i(\vec{x}) \leq 0 \\ |g_i(\vec{x})|, & \text{caso contrário} \end{cases} \quad 1 \leq i \leq n. \quad (3.36)$$

$$D_j(\vec{x}) = \begin{cases} 0, & -\varepsilon \leq h_j(\vec{x}) \leq \varepsilon \\ |h_j(\vec{x})|, & \text{caso contrário} \end{cases} \quad 1 \leq j \leq p. \quad (3.37)$$

onde ε representa os limites da restrição.

Esta função dinâmica aumenta a penalidade conforme se progride através das gerações.

Kazarlis e Petrides (1998) apresentaram uma técnica que aumenta a eficiência de algoritmos genéticos quando aplicada a problemas de otimização com restrição. A função penalidade dinâmica apresenta pelos autores tem a seguinte forma:

$$F(\vec{x}) = f(\vec{x}) + V(q) \left(A \sum_{i=1}^m (\delta_i \cdot w_i \cdot \Phi(d_i(S))) + B \right) \delta_s \quad (3.38)$$

Sendo A um fator denominado fator de "gravidade", m é o número total de restrições, δ_i recebe o valor de 1 se a restrição é violada e 0 caso contrário, w_i é um fator de ponderação para a restrição i , $d_i(S)$ é a medida do grau da restrição i e Φ_i é a função desta medida, B é um fator penalidade, δ_s é um fator binário (δ_s é 1 se S é inviável e 0 caso contrário) e $V(q)$ é uma função crescente de q (a geração atual) no intervalo de $[0,1]$. Kazarlis e Petrides (1998) experimentaram cinco diferentes formas de $V(q)$ (linear, quadrática, cúbica, potência de 4ª ordem e exponencial) em um problema teste. Eles encontraram que o melhor desempenho foi obtido pela função da forma:

$$V(q) = \left(\frac{q}{G} \right)^2 \quad (3.39)$$

Na Equação 3.39, G é o número de gerações.

3.5.3. Penalidade Recozimento

Apresentado por Michalewicz e Attia (1994) o método é baseado na idéia do recozimento simulado onde os coeficientes penalidade mudam uma vez em muitas gerações. Apenas as restrições ativas são consideradas em cada iteração, e a penalidade é aumentada ao longo do tempo tal que indivíduos inviáveis são pesadamente

penalizados nas últimas gerações. A população evolui de acordo com a seguinte função objetivo expandida:

$$F(\vec{x}) = f(\vec{x}) + \frac{1}{2\tau} \sum_{i \in \mathcal{A}} \phi_i^2(\vec{x}) \quad (3.40)$$

Sendo τ é a temperatura de resfriamento e $\mathcal{A}$ é um conjunto de restrições ativas a ser criado.

$$\phi_i(\vec{x}) = \begin{cases} \max[0, g_i(\vec{x})] & \text{se } 1 \leq i \leq n, \\ |h_i(\vec{x})| & \text{se } n+1 \leq i \leq m \end{cases} \quad (3.41)$$

Sendo m o número de restrições.

Em cada iteração, a temperatura τ é reduzida e a nova população é criada utilizando a melhor solução encontrada em iterações anteriores como passo inicial para a próxima iteração.

Diferentes propostas para a penalidade por recozimento podem ser encontradas em Joines e Houck (1994).

3.5.4. Penalidade Adaptativa

Nanakorn e Meesomklin (2001) apresentaram uma penalidade com a capacidade de se ajustar durante a evolução de tal maneira que o grau de penalidade seja sempre alcançado. A função objetivo expandida é apresentada por:

$$F_i^a = F^a(x_i) = F(x_i) - P(x_i) = F(x_i) - \lambda(t)E(x_i) \quad (3.42)$$

Onde F_i^a representa a função objetivo do i -ésimo indivíduo após a penalidade. $\lambda(t)$ representa um fator de ponderação do erro $E(x_i)$. O fator $\lambda(t)$ varia com a geração, sendo o número da geração denotado por t . O termo $E(x_i)$ é definido por:

$$E(x_i) = \sum_{j=1}^K G_j(x_i) + \sum_{j=1}^P H_j(x_i) \quad (3.43)$$

Sendo $G_j(x_i)$ e $H_j(x_i)$ o grau de violação das restrições de desigualdade e igualdade respectivamente.

3.5.5. Penalidade Coevolutiva

Coello (2000) propôs a utilização da função penalidade da seguinte forma:

$$F(\vec{x}) = f(\vec{x}) - (coef \times w_1 + viol \times w_2) \quad (3.44)$$

Onde $f(\vec{x})$ é o valor da função objetivo para um dado conjunto de valores da variável codificada em um cromossomo, w_1 e w_2 são dois fatores de penalidade, $coef$ é a soma de todas as quantidades pelas quais as restrições são violadas (somente restrições de desigualdade são consideradas):

$$coef = \sum_{i=1}^n g_i(\vec{x}) \quad , \quad \forall g_i(\vec{x}) > 0 \quad (3.45)$$

O termo *viol* corresponde a um fator que é inicializado do zero, sendo incrementado em uma unidade para cada restrição que é violada, independentemente da quantidade de violações.

No próximo capítulo serão apresentados alguns estudos de caso que apresentam restrições nas variáveis de estado. Para manipulação dessas restrições, será utilizada a técnica de função penalidade estática.

Capítulo 4

ANÁLISE DO DESEMPENHO DO ALGORITMO DE EVOLUÇÃO DIFERENCIAL NA SOLUÇÃO DE PROBLEMAS DE OTIMIZAÇÃO DINÂMICA

Este capítulo apresenta a avaliação sistemática do Algoritmo de Evolução Diferencial aplicado na solução de Problemas de Otimização Dinâmica. Apesar de largamente empregado em várias áreas como demonstrado na revisão bibliográfica apresentada, a análise do comportamento e do desempenho do Algoritmo de Evolução Diferencial na solução de problemas de Otimização Dinâmica considerando diferentes cenários e diferentes formulações permitirá comprovar a sua adequação e eficiência.

A metodologia adotada considerou problemas com diferentes formulações, cujas soluções por Métodos Indiretos, por Métodos Diretos ou por Métodos Híbridos são bastante complexas devido a flutuações do índice diferencial decorrentes de ativação e desativação de restrições. São considerados problemas com restrições de fim ou de trajetória nas variáveis de estado, resolvidos pela incorporação de Funções Penalidade junto às Funções Objetivo, além de problemas com tempo final fixo e tempo final livre. Para fins de avaliar a sensibilidade da solução aos valores dos parâmetros do Algoritmo de Evolução Diferencial, é feita uma análise exaustiva que considera o efeito de variações dos parâmetros número de gerações, probabilidade de cruzamento e fator de perturbação sobre o valor da Função Objetivo e sobre o esforço computacional para dois estudos de caso.

As soluções foram obtidas para diferentes sementes (geradores de números pseudo-aleatórios distribuídos uniformemente), obtidas utilizando um algoritmo multiplicativo congruente que gera valores de dupla precisão no intervalo $[5 \times 10^{-10}, 1]$. Em todos os estudos de caso apresentados, as sementes utilizadas foram escolhidas aleatoriamente e os resultados obtidos são válidos apenas para o conjunto de sementes referido em cada problema. O critério de parada do algoritmo foi o número máximo de gerações. Os resultados obtidos para todos os estudos de caso são comparados com resultados publicados na literatura para fins de validação e de demonstração de

aplicabilidade do algoritmo de Evolução Diferencial na solução dos problemas apresentados.

4.1. Caso 1: Reator Batelada com reação consecutiva

Este problema de índice algébrico diferencial 1 proposto por Ray (1989) considera uma reação consecutiva $A \rightarrow B \rightarrow C$ ocorrendo num reator batelada. O objetivo é encontrar o perfil ótimo da variável de controle (temperatura) que maximize a produção do produto intermediário B. A reação é de segunda ordem em relação ao consumo de A resultando em B e de primeira ordem em relação ao consumo de B resultando em C.

Os balanços materiais das espécies A e B no reator batelada são dados pelas Equações (4.1) e (4.2) respectivamente:

$$\frac{dC_1}{dt} = -k_1(T)C_1^2, \quad C_1(0) = 1 \quad (4.1)$$

$$\frac{dC_2}{dt} = k_1(T)C_1^2 - k_2(T)C_2, \quad C_2(0) = 0 \quad (4.2)$$

Sendo $C_1 = [A]$ (mol/l), $C_2 = [B]$ (mol/l), $k_i(T) = A_{i0}e^{-E_i/RT}$ $i=1,2$.

Considerando os seguintes limites de temperatura para T:

$$298 \text{ K} \leq T(t) \leq 398 \text{ K} \quad (4.3)$$

e o tempo de batelada de 1 hora, a Função Objetivo é dada por:

$$\max J = C_2(t_f) \quad (4.4)$$

Os parâmetros adicionais requeridos para definir o problema são dados por:

$$\begin{aligned} A_{10} &= 4000 \frac{L}{(mol)(s)} & A_{20} &= 6,2 \times 10^5 / s \\ E_1 &= 5000 \frac{cal}{(g)(mol)} & E_2 &= 10000 \frac{cal}{(g)(mol)} \end{aligned}$$

Renfro, Morshedi e Asbjorsen (1987) aplicaram o método direto simultâneo na resolução deste problema, aplicando o método da colocação ortogonal em 10 subintervalos da variável de controle. O problema foi resolvido em 9 iterações com uma tolerância para a convergência de 1×10^{-4} . Eles obtiveram um valor de 0,610 para a ótima concentração do produto de interesse B, aplicando um perfil constante por partes para a variável de controle (Temperatura).

Dadebo e Mcauley (1995) utilizaram Programação Dinâmica Iterativa, considerando uma aproximação constante por partes para o perfil da variável de controle e avaliando o efeito do número de elementos de controle sobre o máximo rendimento do produto de interesse B e sobre a temperatura inicial. Foi obtido um valor de 0,610070 para 10 estágios de controle e um valor de 0,610453 para 20 estágios de controle. O melhor valor obtido pelos autores foi de 0,610775 para um perfil com 80 estágios de controle.

Rajesh et al. (2001) apresentaram a solução do problema utilizando um algoritmo de colônia de formigas. Os autores adaptaram um perfil de controle linear por partes em que a inclinação dos perfis entre dois intervalos de tempo discretos era utilizado para a solução das equações de balanço de massa do modelo e para a obtenção do rendimento máximo da concentração do produto intermediário B. Foi obtido um valor de 0,61045 para um perfil linear por partes.

Zhang, Chen e Zhao (2005) apresentaram a solução do problema através da aplicação de um algoritmo de colônia de formigas iterativo. Os autores obtiveram um perfil de controle constante por partes executando os três seguintes passos: (i) discretização do intervalo do tempo e da região de controle; (ii) busca do perfil de controle ótimo do sistema discreto através do algoritmo de colônia de formigas; (iii) redução da região de busca e retorno ao passo i para a próxima iteração até a convergência ser atingida. Foi obtido um valor de 0,6100 para 10 estágios de controle e um valor de 0,6104 para 20 estágios de controle.

Neste trabalho, os efeitos da semente, do número de estágios de controle e do número de gerações sobre os resultados obtidos foram avaliados. Foram consideradas 100, 150 e 200 gerações para um número de estágios de controle variando de 5 até 20. Para a solução do sistema de equações diferenciais (Equações 4.1 e 4.2) foi empregado o método de Runge-Kutta 4ª e 5ª ordem com passo de integração variável, tolerância relativa de 10^{-8} e tolerância absoluta de 10^{-4} através da rotina ode45 disponível no software Matlab®. A Tabela 4.1 apresenta os parâmetros do algoritmo de Evolução Diferencial utilizados na solução do problema.

Tabela 4.1- Parâmetros do algoritmo de Evolução Diferencial utilizados na solução do Caso 1- Reator Batelada com reação consecutiva

Número de Indivíduos da População	50
Taxa de Perturbação	0,5
Probabilidade de Cruzamento	0,6
Estratégia	7(ED/best/1/exp)
Vetor de Sementes	[0 1 2 3 4 5 6 7 8 9]

Nas Tabelas 4.2, 4.3, 4.4 e 4.5 são apresentados os resultados obtidos para a Função Objetivo (FO) com diferentes números de estágios de controle e diferentes sementes (geradores randômicos). Neste caso, são necessárias para o N_{gen} utilizadas, 5050, 7550 e 10050 avaliações da Função Objetivo, respectivamente.

Tabela 4.2- Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 5 estágios de controle na solução do Caso 1: Reator Batelada com reação consecutiva

Semente	$N_{gerações}$	FO	$N_{gerações}$	FO	$N_{gerações}$	FO
0	100	0,609472	150	0,609472	200	0,609472
1		0,609472		0,609472		0,609472
2		0,609472		0,609472		0,609472
3		0,609472		0,609472		0,609472
4		0,609472		0,609472		0,609472
5		0,609472		0,609472		0,609472
6		0,609472		0,609472		0,609472
7		0,609472		0,609472		0,609472
8		0,609472		0,609472		0,609472
9		0,609472		0,609472		0,609472
Média		0,609472		0,609472		0,609472

Tabela 4.3-Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 10 estágios de controle na solução do Caso 1: Reator Batelada com reação consecutiva

Semente	$N_{gerações}$	FO	$N_{gerações}$	FO	$N_{gerações}$	FO
0	100	0,610115	150	0,610117	200	0,610117
1		0,610115		0,610117		0,610117
2		0,610115		0,610117		0,610117
3		0,610115		0,610117		0,610117
4		0,610115		0,610117		0,610117
5		0,610115		0,610117		0,610117
6		0,610116		0,610117		0,610117
7		0,610115		0,610117		0,610117
8		0,610115		0,610117		0,610117
9		0,610115		0,610117		0,610117
Média		0,610115		0,610117		0,610117

Tabela 4.4- Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 15 estágios de controle na solução do Caso 1: Reator Batelada com reação consecutiva

Semente	$N_{gerações}$	FO	$N_{gerações}$	FO	$N_{gerações}$	FO
0	100	0,610259	150	0,610333	200	0,610346
1		0,610250		0,610333		0,610345
2		0,610264		0,610333		0,610346
3		0,610176		0,610314		0,610342
4		0,610271		0,610336		0,610347
5		0,610248		0,610329		0,610344
6		0,610256		0,610322		0,610344
7		0,610255		0,610329		0,610344
8		0,610292		0,610334		0,610345
9		0,610248		0,610333		0,610344
Média		0,610252		0,610330		0,610345

Tabela 4.5-Influência das sementes e do número de gerações sobre o valor de Função Objetivo (FO) para 20 estágios de controle na solução do Caso 1: Reator Batelada com reação consecutiva

Semente	N_{gen}	FO	N_{gen}	FO	N_{gen}	FO
0	100	0,610233	150	0,610434	200	0,610465
1		0,610202		0,610435		0,610464
2		0,610176		0,610422		0,610466
3		0,610015		0,610389		0,610460
4		0,610102		0,610412		0,610462
5		0,610082		0,610389		0,610462
6		0,610229		0,610435		0,610464
7		0,610147		0,610407		0,610463
8		0,610098		0,610403		0,610465
9		0,610092		0,610394		0,610465
Média		0,610138		0,610412		0,610464

Os resultados apresentados nas Tabelas 4.2 a 4.5 mostram que de uma maneira geral os geradores randômicos (sementes) tiveram pouca influência nos valores da Função Objetivo. Para 5 estágios de controle, os valores de Função Objetivo foram os mesmos para todas as sementes e número de gerações. Considerando 10 estágios de controle, apenas uma semente apresentou um valor de Função Objetivo diferente das demais para um número de gerações $N_{gerações} = 100$. A influência das sementes foi maior para 15 e para 20 estágios, principalmente para um número de gerações $N_{gerações} = 100$ em que a diferença dos valores obtidos ocorreu na quarta casa decimal.

Observa-se que o melhor valor de Função Objetivo obtido por este trabalho (FO = 0,610466) para 20 estágios de controle, é 0,05% inferior ao melhor valor reportado da literatura por Dadebo e Mcauley (1995) utilizando 80 estágios de controle (FO = 0,610775). Fazendo uma comparação com outro método de otimização natural (Colônia de Formigas), o melhor valor de Função Objetivo obtido por este trabalho para 20 estágios de controle (FO = 0,610466) é 0,011 % superior ao valor encontrado por Zhang et al. (2005) utilizando o mesmo número de estágios de controle (FO = 0,6104).

Os perfis da variável de controle (Temperatura) obtidos para 5, 10, 15 e 20 estágios de controle são apresentados nas Figuras 4.1, 4.2, 4.3 e 4.4 respectivamente. Os perfis foram obtidos para o maior valor de Função Objetivo encontrado nas 30 simulações realizadas para cada número de estágios de controle.

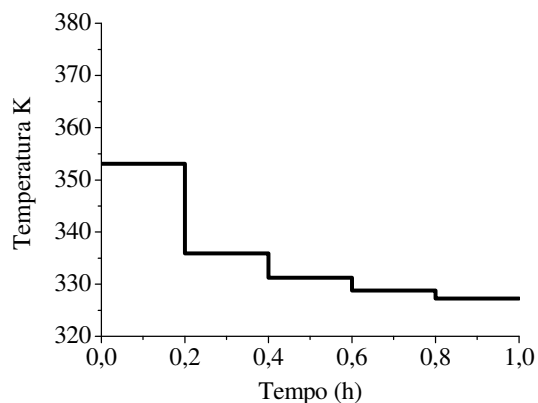


Figura 4.1- Perfil da variável de controle T(K) para 5 estágios de controle (FO=0,609472)

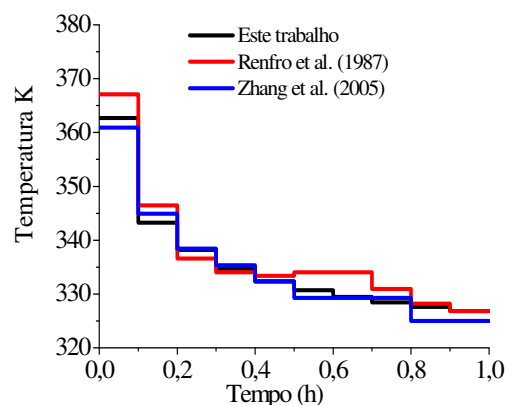


Figura 4.2- Perfil da variável de controle T(K) para 10 estágios de controle (FO=0,610117)

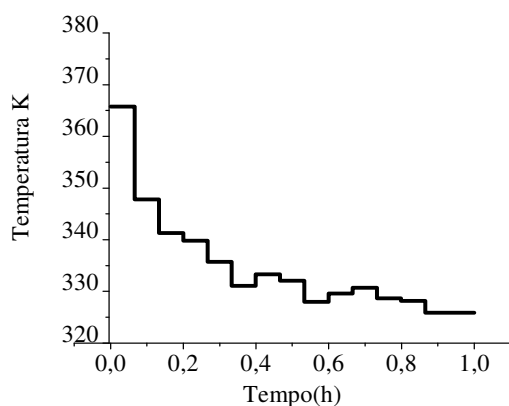


Figura 4.3- Perfil da variável de controle T(K) para 15 estágios de controle (FO=0,610347)

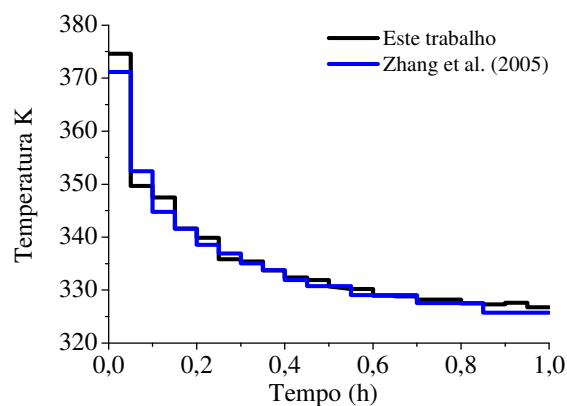


Figura 4.4- Perfil da variável de controle T(K) para 20 estágios de controle (FO=0,610466)

Através da Figura 4.2 observa-se que o perfil da variável de controle obtido por este trabalho apresenta uma boa concordância com os perfis obtidos por diferentes técnicas. A diferença principal é observada nos estágios 6 e 7 da variável de controle.

Considerando 20 estágios de controle na Figura 4.4, o perfil obtido por este trabalho é semelhante ao perfil obtido por Zhang et al. (2005) que utilizaram o algoritmo de Colônia de Formigas. A Figura 4.5 apresenta os perfis das variáveis de estado (C_1 e C_2).

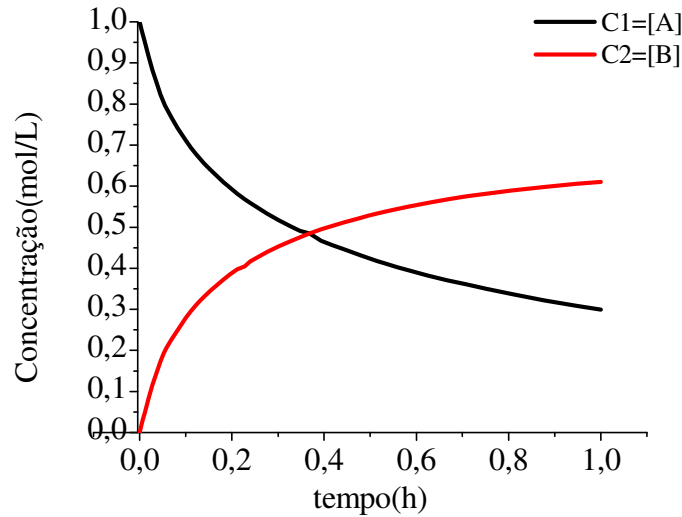


Figura 4.5: Perfil das variáveis de estado C_1 e C_2 do problema do Reator Batelada com reação consecutiva obtidas por este trabalho para $FO=0,610466$

4.2. Caso 2: Problema de Jacobson et al. (1970) com formulação de índice diferencial 5

Neste estudo de caso é considerada a formulação para o problema apresentado originalmente por Jacobson et al. (1970):

$$\min x_3(t_f = 5) \quad (4.5)$$

$$\dot{x}_1 = x_2, \quad x_1(t_0) = 0 \quad (4.6)$$

$$\dot{x}_2 = u, \quad x_2(t_0) = 1 \quad (4.7)$$

$$\dot{x}_3 = x_1^2, \quad x_3(t_0) = 0 \quad (4.8)$$

$$-1 \leq u \leq 1 \quad (4.9)$$

Jacobson et al. (1970) resolveram este problema através da transformação do problema singular em um problema não singular pela adição de um termo quadrático da

variável de controle multiplicado por um parâmetro de perturbação ε , ao termo da integral da Função Objetivo. O resultado do problema não singular é resolvido por um decréscimo da sequência de parâmetros ε_k [$\varepsilon_1 > \varepsilon_2 > \dots > \varepsilon_k > 0$] e como $k \rightarrow \infty$ e $\varepsilon_k \rightarrow 0$, a solução do problema modificado tende à solução do problema original. O problema não singular foi resolvido utilizando o algoritmo de Programação dinâmica diferencial com base no Princípio do Máximo de Pontryagin e as equações diferenciais foram resolvidas por um simples esquema de integração de Euler. Jacobson et al. (1970) reportaram um mínimo de 0,2771

Dadebo e McAuley (1995) utilizaram o método direto através da aplicação do algoritmo de Programação Dinâmica Iterativa (PDI) na solução deste problema, que foi comparado com métodos baseados em gradientes. Com isso, ao recorrerem a métodos que utilizam alguma forma de perturbação na Função Objetivo (entre eles o apresentado por Jacobson et al.(1970)), eles buscaram identificar as limitações destes métodos utilizando PDI. Outras aplicações podem ser vistas em Luus(1995) utilizando Programação Dinâmica Iterativa e Luus(2001) utilizando o algoritmo de Luus-Jakola.

Dadebo e McAuley (1995) encontraram o valor de 0,269 usando 80 estágios de controle de tamanho uniforme. Já Luus (1995), usando programação dinâmica iterativa com 4 estágios de controle variáveis, obteve 0,268395. Luus (2001), usando 5 estágios de controle variáveis e o algoritmo de Luus-Jaakola, encontrou o valor ótimo de 0,268394, com 200000 avaliações da função objetivo FO.

Neste trabalho, o POD singular é transformado em um problema de identificação das variáveis de controle u_i , $1 \leq i \leq n_{\text{control}}$, e do tamanho dessas variáveis(instantes de atuação dos elementos) (j), $1 \leq j \leq n_{\text{elementos}}-1$, totalizando $2n_{\text{elementos}}-1$ variáveis de projeto, como ilustrado a seguir:

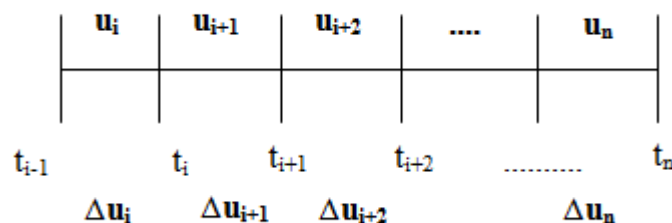


Figura 4.6: Elementos de Controle discretizados

A Tabela 4.6. apresenta os valores dos parâmetros utilizados pelo algoritmo de Evolução Diferencial nas soluções do problemas de Jacobson et al.(1970) (os mesmos

parâmetros também foram utilizados na solução do Caso 3 apresentado na sequência deste trabalho).

Tabela 4.6- Parâmetros utilizados nas soluções dos problemas de Jacobson et al (1970)

Número de Indivíduos da População	100
Número de Gerações	100,200,300,500
Taxa de Perturbação	0,8
Probabilidade de Cruzamento	0,5
Estratégia	7(ED/best/1/exp)
Vetor de Sementes	[0,1 0,2 0,3 0,4 0,5 0,6 0,7 0,8 0,9]

Para a solução do sistema de equações diferenciais foi empregado o método de Runge-Kutta 4ª ordem com 10^4 pontos no domínio do tempo com passo de integração constante.

Neste caso, foram necessárias 10100, 20100, 30100 e 50100 avaliações da Função Objetivo, respectivamente, para o número de gerações de 100, 200, 300, 500. Os resultados obtidos com o algoritmo de Evolução Diferencial são comparados com os resultados obtidos por Luus (2001) usando o algoritmo de Luus-Jaakola.

A Tabela 4.7 apresenta os resultados obtidos com o algoritmo Evolução Diferencial para diferentes sementes, diferentes números de geração (N_{gen}) e para 5 estágios de controle. De modo geral, o aumento do número de gerações diminui o valor da Função Objetivo (FO), pouco influenciado pela escolha da semente.

Tabela 4.7: Resultados obtidos para o estudo de caso 2: problema de Jacobson com índice diferencial 5

Semente	N_{gen}	FO	N_{gen}	FO	N_{gen}	FO	N_{gen}	FO
0	100	0,268628	200	0,268455	300	0,268445	500	0,268442
0,1		0,269749		0,268516		0,268455		0,268439
0,2		0,272885		0,269954		0,268585		0,268442
0,3		0,270878		0,268863		0,268488		0,268447
0,4		0,269513		0,268771		0,268447		0,268441
0,5		0,268512		0,268478		0,268448		0,268444
0,6		0,268766		0,268447		0,268440		0,268439
0,7		0,269074		0,268464		0,268440		0,268439
0,8		0,268683		0,268450		0,268441		0,268439
0,9		0,269251		0,268561		0,268512		0,268504
Média		0,269594		0,268696		0,268470		0,268448

De maneira geral, observa-se que os resultados obtidos com o algoritmo de Evolução Diferencial são compatíveis aos obtidos por Luus (1995, 2001).

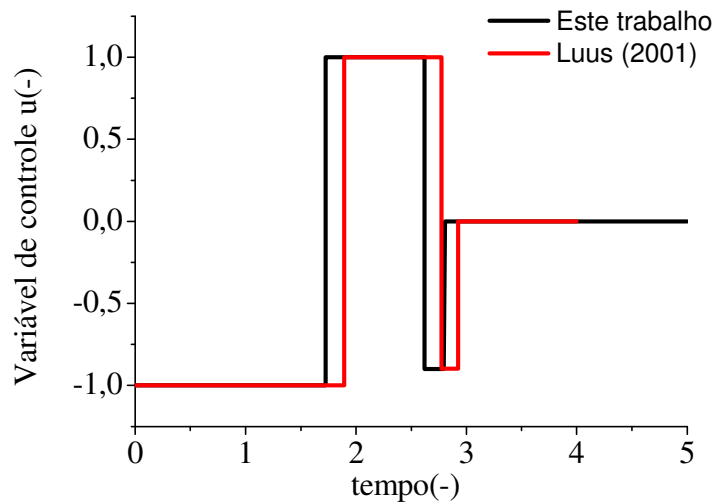


Figura 4.7: Perfil da variável de controle para o Problema de Jacobson et al (1970) com formulação de índice 5.

Observa-se da Figura 4.7 que o perfil da variável de controle obtido por este trabalho é semelhante ao perfil obtido por Luus (2001). Em ambos os perfis, a variável de controle vai do limite mínimo ($u = -1$) até o limite máximo ($u=1$), terminando em $t_f = 5$ com $u=0$. A diferença entre os perfis se apresenta nos instantes de tempo em que a variável de controle apresenta saltos do limite mínimo para o máximo, retornando a um valor próximo do mínimo e novamente um salto até o controle singular ($u=0$).

Na Figura 4.8 são apresentados os perfis das variáveis de estado.

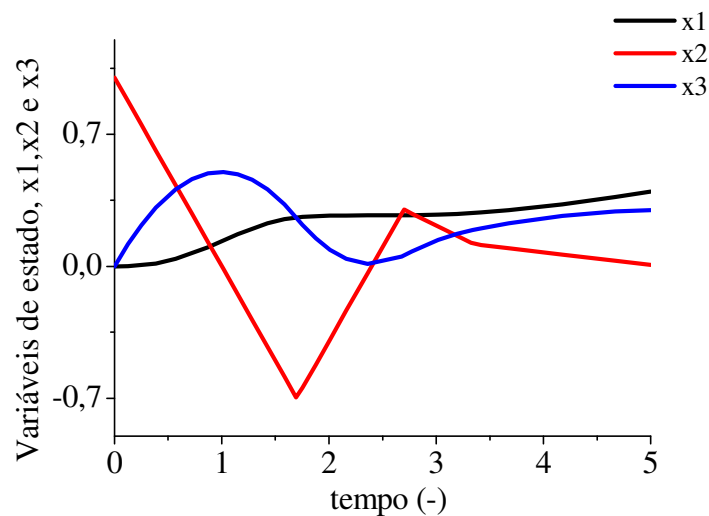


Figura 4.8: Perfis das variáveis de estado obtidas por esse trabalho para o Problema de Jacobson et al (1970) com formulação de índice 5.

4.3. Caso 3: Problema de Jacobson et al (1970) com formulação de índice diferencial 3

O segundo problema de índice 3 é apresentado pelas equações de (4.10) até (4.14).

$$\min x_3(t_f = 5) \quad (4.10)$$

$$\dot{x}_1 = x_2, \quad x_1(t_0) = 0 \quad (4.11)$$

$$\dot{x}_2 = u, \quad x_2(t_0) = 1 \quad (4.12)$$

$$\dot{x}_3 = x_1^2 + x_2^2, \quad x_3(t_0) = 0 \quad (4.13)$$

$$-1 \leq u \leq 1 \quad (4.14)$$

A Tabela 4.8 apresenta os resultados obtidos com o algoritmo de Evolução Diferencial para diferentes sementes, diferentes números de geração (N_{gen}), e para 5 estágios de controle.

Tabela 4.8: Resultados obtidos para o estudo de caso 3: problema de Jacobson com índice diferencial 3

Semente	N_{gen}	FO	N_{gen}	FO	N_{gen}	FO	N_{gen}	FO
0	100	0,756267	200	0,754245	300	0,754171	500	0,754152
0,1		0,754856		0,754470		0,754204		0,754152
0,2		0,766735		0,754812		0,754206		0,754153
0,3		0,755625		0,754447		0,754201		0,754156
0,4		0,755442		0,754242		0,754163		0,754151
0,5		0,760397		0,754236		0,754160		0,754151
0,6		0,759058		0,754171		0,754152		0,754151
0,7		0,759395		0,754877		0,754535		0,754472
0,8		0,759274		0,754262		0,754163		0,754151
0,9		0,756475		0,754161		0,754151		0,754151
Média		0,758352		0,754392		0,754211		0,754184

Para este estudo de caso, Jacobson et al. (1970) reportaram um mínimo de 0,8285. Dadebo e McAuley (1995) encontraram o valor de 0,754016 usando 80 estágios de controle de tamanho uniforme. Luus (1995), usando programação dinâmica iterativa com 10 estágios de controle variáveis, obteve 0,7539848. Já Luus (2001), usando 12

estágios de controle variáveis e o algoritmo de Luus-Jaakola, encontrou o valor ótimo de 0,7539853, com 200000 avaliações da Função Objetivo.

Similarmente ao comportamento observado no caso 4.2, observa-se que os resultados obtidos com o algoritmo de Evolução Diferencial são superiores ao resultado obtido por Jacobson et al. (1970) e compatíveis aos obtidos por Dadebo e Mcauley (1995) e por por Luus (1995, 2001) com um número menor de estágios de controle. Além disso, constata-se que, para o estudo de caso analisado, a escolha da semente inicial tem pouca influência sobre o valor da Função Objetivo. A Figura 4.9 apresenta o perfil da variável de controle.

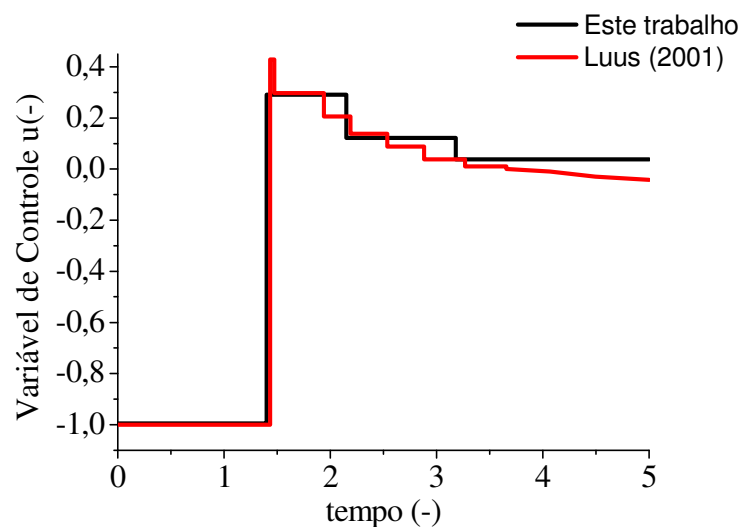


Figura 4.9: Perfil da variável de controle para o Problema de Jacobson et al. (1970) com formulação de índice 3

Observa-se da Figura 4.9 que ambos os perfis (obtidos por este trabalho e por Luus (2001)) iniciam no limite inferior ($u = -1$). O perfil obtido por Luus (2001) apresenta uma maior aproximação em $t_f = 5$ de $u=0$ (controle singular) em relação ao perfil obtido por este trabalho.

Na Figura 4.10 são apresentados os perfis das variáveis de estado.

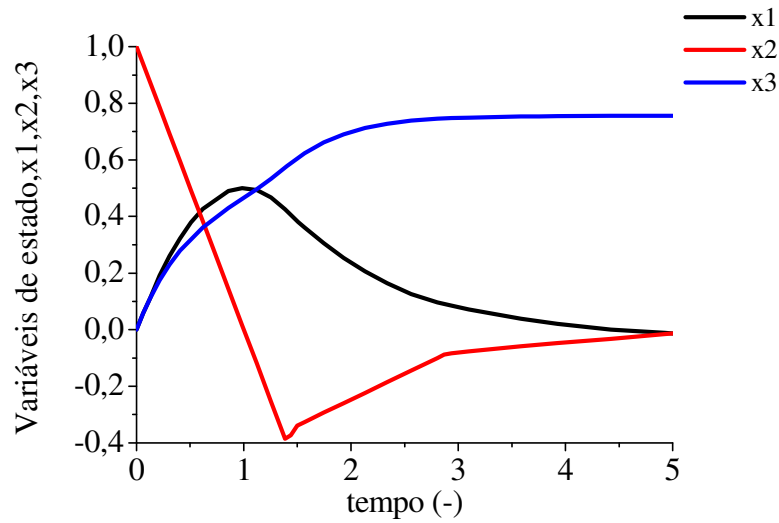


Figura 4.10: Perfis das variáveis de estado obtidas por esse trabalho para o Problema de Jacobson et al (1970) com formulação de índice 3.

4.4. Caso 4: Problema de tempo final livre: O problema do carro

O problema de índice 1 tem como objetivo encontrar o perfil de uma variável de controle (aceleração) que minimize o tempo final (Equação 4.15) de parada de um carro que percorre uma distância de 300 unidades.

Cuthrell e Biegler (1987) solucionaram o problema convertendo equações diferenciais em equações algébricas residuais com coeficientes desconhecidos utilizando colocação ortogonal sobre elementos finitos. Eles utilizaram diferentes números de elementos, comparando os perfis das variáveis de estado obtidas com os perfis analíticos.

As variações da distância percorrida e da velocidade com o tempo são apresentadas nas Equações (4.16) e (4.17). Os limites inferior e superior da variável de controle (aceleração) são apresentados na Equação (4.18).

$$\min t_f \quad (4.15)$$

$$\dot{x} = v, \quad x(t_0) = 0, x(t_f) = 300 \quad (4.16)$$

$$\dot{v} = a, \quad v(t_0) = 0, v(t_f) = 0 \quad (4.17)$$

$$-2 \leq a \leq 1 \quad (4.18)$$

Os parâmetros do algoritmo de Evolução Diferencial utilizados na solução do problema são apresentados na Tabela 4.9.

Tabela 4.9- Parâmetros utilizados na solução do problema do carro

Número de Indivíduos da População	50
Número de Gerações	100, 300, 500
Taxa de Perturbação	0,8
Probabilidade de Cruzamento	0,8
Estratégia	7 (ED/best/1/exp)
Vetor de Sementes	[0 1 2 3 4 5 6 7 8 9 10]

Para a solução do Sistema de Equações Diferenciais (4.16 e 4.17) foi utilizado um método com fórmulas de diferenciação numérica a ré com passo de integração variável, tolerâncias relativa e absoluta de 10^{-10} através da rotina ode15s disponível no software Matlab®. De trabalhos reportados na literatura, observa-se que este problema é do tipo liga-desliga, sendo que a variável de controle assume ou o valor máximo ou o valor mínimo. No algoritmo de Evolução diferencial, as variáveis de projeto são dois estágios de controle ($u \in [-2 \ 1]$) e dois instantes de atuação dos estágios, $t_1 \in [15 \ 25]$, $t_2 \in [25 \ 35]$.

Para a solução de um problema de minimização do tempo final, é necessário que se estabeleça o valor das variáveis de estado em um tempo final. Para isso, foi utilizada a técnica de penalidade estática com o fator penalidade $rp=10^8$ para manipular a restrição de fim, garantindo que os valores para as variáveis de estado no tempo final sejam alcançados.

Na Tabela 4.10 são apresentados os valores de Função Objetivo obtidos para diferentes sementes e diferentes números de gerações.

Tabela 4.10- Resultados obtidos para diferentes sementes e diferentes números de gerações: problema do carro

Semente	$N_{\text{gerações}}$	FO	$N_{\text{gerações}}$	FO	$N_{\text{gerações}}$	FO
0	100	802832,97	300	31,05	500	30,00
1		381733,22		30,08		30,00
2		948184,30		32,24		30,00
3		298143,89		34,54		30,00
4		133271,75		31,18		30,00
5		371928,74		31,29		30,00
6		449497,99		30,90		30,00
7		3225905,54		31,07		30,00
8		1916879,09		30,77		30,00
9		1862682,47		37,26		30,00
10		26302000,01		30,54		30,00
Média		3335732,72		31,90		30,00

Nas Figuras 4.11 e 4.12 são apresentados os perfis das variáveis de estado (distância percorrida e velocidade).

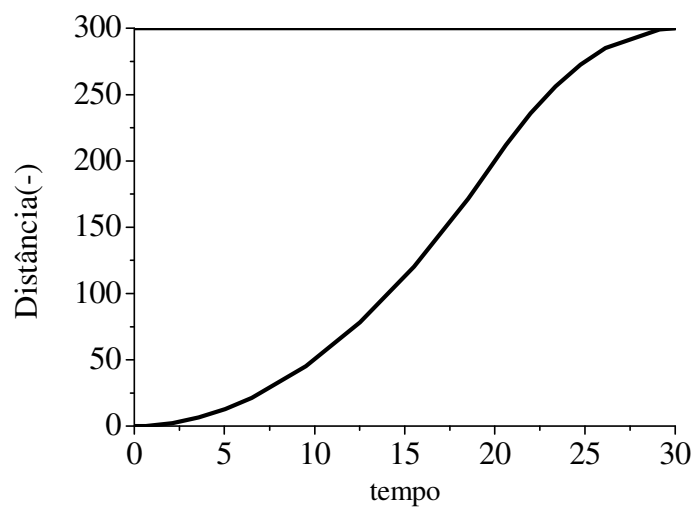


Figura 4.11: Perfil da variável de estado x (distância percorrida).

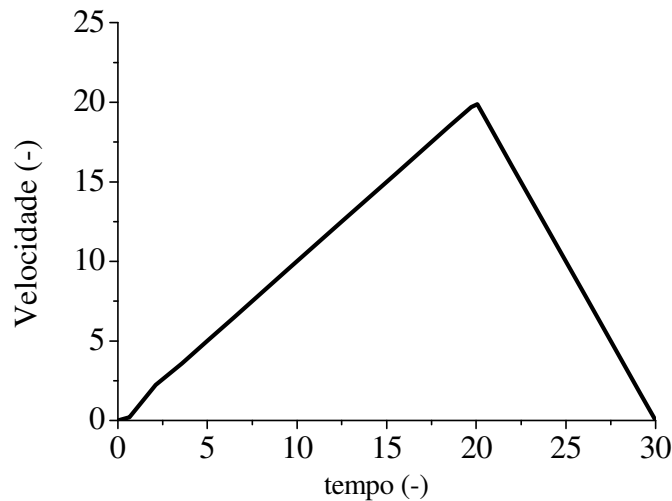


Figura 4.12: Perfil da variável de estado v (velocidade).

Observa-se através da Tabela 4.10 que inicialmente para um número de gerações $N_{\text{gerações}}=100$, todos os indivíduos (vetores) da população violaram alguma(s) restrição de fim para as variáveis de estado (Equações 4.16 e 4.17). Aumentando o número de gerações para $N_{\text{gerações}}=300$, observa-se que os melhores resultados obtidos em cada semente aproxima-se do valor mínimo, já não mais violando a restrição de fim. Para um número de gerações $N_{\text{gerações}}=500$, todas as sementes apresentam o mesmo valor mínimo de Função Objetivo.

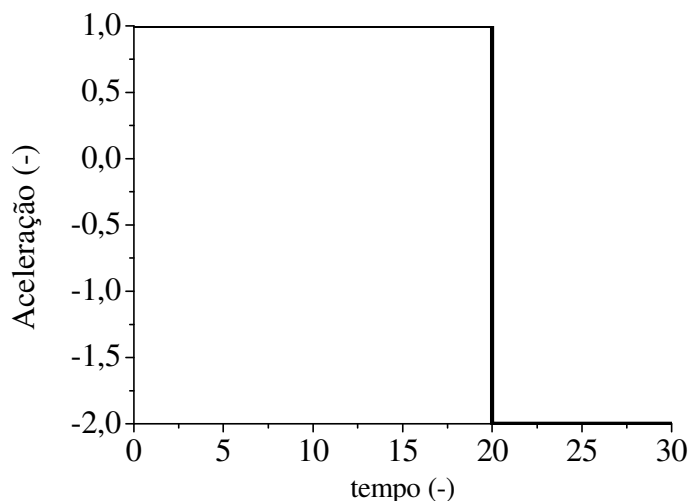


Figura 4.13: Perfil da variável de controle a (aceleração).

O perfil da variável de controle (aceleração) obtido por Evolução Diferencial (Figura 4.13) reafirma os trabalhos reportados na literatura, ou seja, o controle é do tipo liga-desliga, onde o carro apresenta uma aceleração constante até o instante de tempo

($t=20$) e após é desacelerado vindo a parar ($v=0$) em $t=30$. A figura 4.12 apresenta o aumento da velocidade desenvolvida até o instante de tempo ($t=20$) e após esse instante a diminuição da velocidade até parar em $t=30$. Perfis semelhantes aos das Figuras 4.11, 4.12 e 4.13 podem ser observados em Cuthrell e Biegler (1987).

4.5. Caso 5: Problema com restrição de fim na variável de estado considerando tempo final fixo e tempo final livre: Produção de etanol

O problema de índice 2 descrito em Fu e Barford (1993), Bojkov e Luus (1996), Luus e Henessy (1999) e Upreti (2004) tem por objetivo determinar o perfil de uma variável de controle u (vazão de alimentação) que maximize a produção de etanol secretado pelo microorganismo *Saccharomyces cerevisiae*.

Fu e Bardford (1993) sugeriram uma abordagem diferente para formular o problema de controle singular. Os autores aplicaram uma estratégia de controle ótimo não singular oriunda da aplicação do Princípio do Mínimo de Pontryagin. Os resultados apresentados nesse trabalho consideram o tempo final livre.

Bojkov e Luus (1996) utilizaram o algoritmo de Programação Dinâmica Iterativa na solução do problema. Os autores utilizaram estágios de tempo de comprimento variável e os resultados apresentados consideraram o tempo final livre.

Luus e Hennessy (1999) aplicaram o algoritmo de Luus-Jakola para diferentes números de elementos de controle (elementos de tamanho uniforme) observando que quanto maior o número de elementos, mais preciso é o perfil de controle em relação aos eventos (instantes em que o problema apresenta flutuação do índice diferencial). Os resultados apresentados nesse trabalho consideram um tempo final fixo de 63 horas.

Upreti (2004) solucionou o problema utilizando algoritmos genéticos. Ele quantificou o valor da Função Objetivo para noventa sementes (inicializador de geradores randômicos) diferentes, calculando a média e o desvio padrão para os valores obtidos. Os resultados apresentados consideram um tempo final fixo de 63 horas.

O problema pode ser formulado (conforme apresentado em Fu e Barford, 1993) da seguinte maneira:

$$\max J = P(t_f)V(t_f) \quad (4.19)$$

$$\dot{X} = X\mu - \frac{uX}{V} \quad (4.20)$$

$$\dot{S} = -\frac{X\mu}{Y_{xs}} + \frac{u(S_0 - S)}{V} \quad (4.21)$$

$$\dot{P} = X\varepsilon - \frac{uP}{V} \quad (4.22)$$

$$\dot{V} = u \quad (4.23)$$

Os parâmetros cinéticos μ e ε são definidos pelas Eq. 4.24 e Eq.4.25 respectivamente:

$$\mu = \frac{\mu_0 S}{\left(1 + \frac{P}{k_p}\right)(k_s + S)} \quad (4.24)$$

$$\varepsilon = \frac{\varepsilon_0 S}{\left(1 + \frac{P}{k'_p}\right)(k'_s + S)} \quad (4.25)$$

Sendo X , S e P as concentrações mássicas de células, substrato e produto respectivamente, V o volume de caldo dentro do fermentador, μ a taxa de crescimento específico, ε a produtividade específica, Y_{xs} o coeficiente de rendimento e S_0 a concentração de substrato na alimentação.

Para a solução do problema são considerados os seguintes valores para as constantes cinéticas do crescimento do microorganismo *Saccharomyces cerevisiae* em um meio contendo glucose: $\mu_0 = 0,408 \text{ h}^{-1}$; $k_p = 16 \text{ g L}^{-1}$; $k_s = 0,22 \text{ g L}^{-1}$; $\varepsilon_0 = 1,0 \text{ h}^{-1}$; $k'_p = 71,5 \text{ g L}^{-1}$; $k'_s = 44 \text{ g L}^{-1}$; $Y_{xs} = 0,1$. Considera-se que inicialmente o fermentador contém 10 L de caldo em que a concentração mássica de células e substrato é de 1 g L^{-1} e 150 g L^{-1} respectivamente e que não contém nenhuma quantidade de etanol. A capacidade máxima em volume permitida no fermentador é de 200 L e a vazão de alimentação no fermentador é restrita a 12 L h^{-1} no máximo.

O problema pode ser reformulado (formulação utilizada neste trabalho) conforme apresentado em Luus e Henessy (1999)

$$\max J = x_3(t_f)x_4(t_f) \quad (4.26)$$

$$\dot{x}_1 = Ax_1 - u \frac{x_1}{x_4} \quad (4.27)$$

$$\dot{x}_2 = -10Ax_1 + u \left(\frac{150 - x_2}{x_4} \right) \quad (4.28)$$

$$\dot{x}_3 = Bx_1 - u \frac{x_3}{x_4} \quad (4.29)$$

$$\dot{x}_4 = u \quad (4.30)$$

$$x^T(0) = [1 \ 150 \ 0 \ 10] \quad (4.31)$$

$$0 \leq u \leq 12 \quad (4.32)$$

$$x_4(t_f) \leq 200 \quad (4.33)$$

$$A = \left(\frac{0,408}{1 + \frac{x_3}{16}} \right) \left(\frac{x_2}{0,22 + x_2} \right) \quad (4.34)$$

$$B = \left(\frac{1}{1 + \frac{x_3}{71,5}} \right) \left(\frac{x_2}{0,44 + x_2} \right) \quad (4.35)$$

Os parâmetros do algoritmo de Evolução Diferencial utilizados na solução dos problemas de tempo final fixo (63 horas) e tempo final livre são apresentados na Tabela 4.11.

Tabela 4.11- Parâmetros utilizados na solução do problema: Produção de etanol

Número de Indivíduos da População	50
Número de Gerações	100, 200, 300, 400
Taxa de Perturbação	0,5
Probabilidade de Cruzamento	0,8
Estratégia	7 (ED/best/1/exp)
Vetor de Sementes	[0 1 2 3 4 5 6 7 8 9 10]

Para manipular a restrição de fim na variável de estado foi aplicada a técnica de penalidade estática com o fator penalidade $rp = 10^5$. As variáveis a serem otimizadas no algoritmo de Evolução Diferencial são os estágios de controle (20 estágios de tamanho variável) e os instantes de atuação dos estágios (20 instantes de atuação dos estágios) totalizando 40 variáveis de decisão.

Para a solução do Sistema de Equações Diferenciais (Equações 4.27 - 4.30) foi empregado o método de Runge-Kutta 4ª e 5ª ordem com passo de integração variável, tolerância relativa de 10^{-12} e tolerância absoluta de 10^{-12} através da rotina disponível no software Matlab®.

Nas Tabelas 4.12, 4.13 e 4.14 são apresentados os resultados obtidos para a Função Objetivo (Equação 4.26) para diferentes sementes e diferentes números de gerações para um tempo final fixo de 63 horas e para tempo final livre.

Tabela 4.12- Resultados obtidos para diferentes sementes e diferentes números de gerações(considerando tempo final fixo de 63 horas)

Semente	N_{gen}	FO(g)	N_{gen}	FO(g)	N_{gen}	FO(g)	N_{gen}	FO(g)
0	100	20738,77	200	20765,41	300	20765,41	400	20765,41
1		20658,14		20674,93		20758,14		20758,14
2		20720,05		20756,89		20756,89		20756,89
3		20618,76		20730,95		20730,95		20732,26
4		20742,87		20742,87		20793,32		20793,32
5		20645,19		20782,82		20782,82		20782,82
6		20635,41		20663,34		20683,66		20776,49
7		20638,15		20707,71		20707,71		20707,71
8		20741,65		20741,65		20741,65		20741,65
9		20734,96		20734,96		20734,96		20734,96
10		20736,81		20771,36		20771,36		20771,36
Média		20691,89		20733,90		20747,90		20756,45

Da Tabela 4.12 observa-se que o número de gerações apresenta maior influência (para o conjunto de sementes selecionado) sobre os valores de Função Objetivos até 200 gerações. Em apenas três das onze sementes, os valores de Função objetivo foram os mesmos em 100 e 200 gerações. Em oito das onze sementes utilizadas, os valores de Função Objetivo foram os mesmos em 300 gerações e 200 gerações. Da mesma forma, oito sementes apresentaram os mesmos valores de Função Objetivo em 300 e 400 gerações. O melhor resultado obtido considerando um tempo final fixo de 63 horas foi para a semente 4 com 300 gerações (FO=20793,32 g).

Tabela 4.13- Resultados obtidos para diferentes sementes e diferentes números de gerações (considerando tempo final livre e $N_{\text{gerações}}=100$ e 200)

Semente	$N_{\text{gerações}}$	FO(g)	Tempo final (h)	$N_{\text{gerações}}$	FO(g)	Tempo final (h)
0	100	20769,85	62,90	200	20776,68	60,43
1		20735,54	61,27		20737,67	61,14
2		20743,05	60,49		20743,63	60,49
3		20712,91	60,98		20740,63	61,09
4		20683,99	61,07		20784,50	61,66
5		20704,17	61,32		20757,27	62,18
6		20794,36	60,56		20794,36	60,56
7		20737,08	62,61		20737,08	62,61
8		20676,31	61,98		20788,15	61,99
9		20729,59	61,64		20764,41	62,48
10		20711,79	60,29		20751,04	62,07
Média		20725,06			20759,69	

Tabela 4.14- Resultados obtidos para diferentes sementes e diferentes números de gerações (considerando tempo final livre e $N_{\text{gerações}}=300$ e 400)

Semente	$N_{\text{gerações}}$	FO(g)	Tempo final (h)	$N_{\text{gerações}}$	FO(g)	Tempo final (h)
0	300	20776,68	60,43	400	20776,68	60,43
1		20750,10	61,14		20750,10	61,14
2		20758,98	60,49		20758,98	60,49
3		20740,63	61,09		20740,63	61,09
4		20784,50	61,66		20797,91	62,19
5		20757,27	62,32		20757,27	62,32
6		20794,36	60,56		20794,36	60,56
7		20742,39	62,87		20764,50	60,75
8		20788,15	61,99		20788,15	61,99
9		20764,41	62,48		20764,41	62,48
10		20795,68	62,90		20795,68	62,90
Média		20767,07			20771,69	

Das Tabelas 4.13 e 4.14 observa-se de uma maneira geral que o número de gerações apresenta maior influência sobre os valores de Função Objetivo até 200 gerações. Em sete das onze sementes utilizadas, os valores de Função Objetivo e o tempo final obtido foram os mesmos em 300 e 200 gerações. Para um número de gerações $N_{\text{gerações}}=400$, em nove das onze sementes o valor de Função Objetivo e o tempo final obtido foi o mesmo que em 300 gerações. O melhor resultado obtido foi para a semente 4 para um número de gerações $N_{\text{gerações}}=400$ (FO=20797,91 g , tempo final=62,19 h).

Nas Figuras 4.14 e 4.15 são apresentados os perfis da variável de controle para o melhor resultado obtido considerando tempo final fixo de 63 horas e tempo final livre. O perfil da variável de controle obtido por esse trabalho é comparado com o perfil obtido por Luus e Henessy (1999) considerando tempo final fixo, enquanto que o perfil obtido considerando tempo final livre é comparado com Bojkov e Luus (1996). Nas figuras 4.16 e 4.17 são apresentados os perfis das variáveis de estado considerando tempo final fixo e tempo final livre.

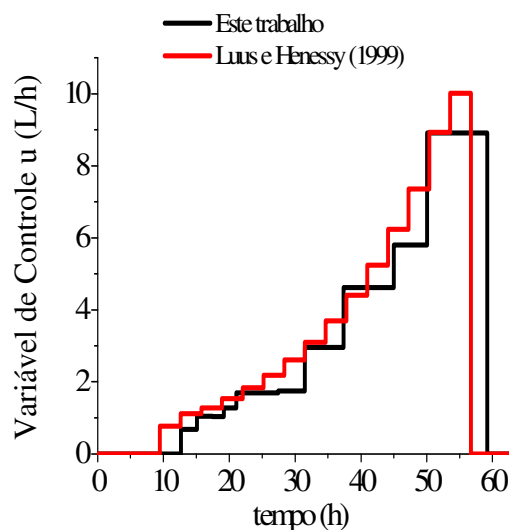


Figura 4.14: Perfis da variável de controle u (tempo final de 63 horas)

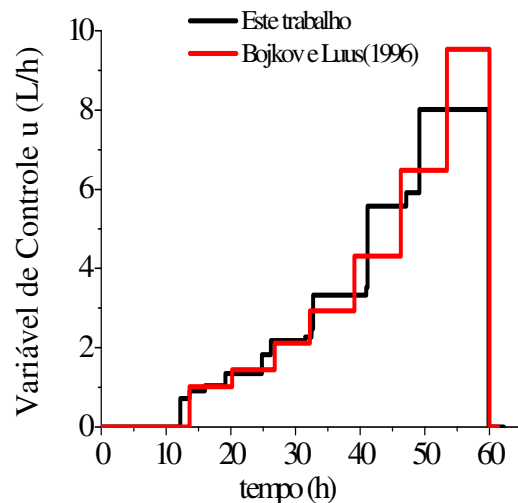


Figura 4.15: Perfis da variável de controle u (tempo final livre)

Observa-se das Figuras 4.14 e 4.15 que o perfil de controle obtido por este trabalho se assemelha aos perfis obtidos por Luus e Henessy (1999) e Bojkov e Luus (1995). As diferenças principais estão no instante em que a variável de controle começa a atuar (instante de tempo em que u começa a aumentar) e o instante em que subitamente o controle para de atuar, retornando para $u=0$. Observa-se também que em ambos os casos (tempo final fixo e tempo final livre) o valor máximo encontrado para a variável de controle é superior em Luus e Henessy (1999) e Bojkov e Luus (1995) ao valor máximo encontrado por este trabalho.

Observa-se através das Figuras 4.16 e 4.17 que a concentração de substrato diminui até aproximadamente 14 horas de batelada. A partir desse momento, através da atuação da variável de controle, a concentração é mantida em níveis aproximadamente

constantes até o instante que a variável de controle para de atuar no final do processo batelada e a concentração de substrato é zero.

Na Tabela 4.15 são apresentados o melhor valor de Função Objetivo obtido por este trabalho e valores de Função Objetivo obtidos por diferentes referências. São apresentados também o número de avaliações de Função Objetivo necessárias para encontrar a solução ótima.

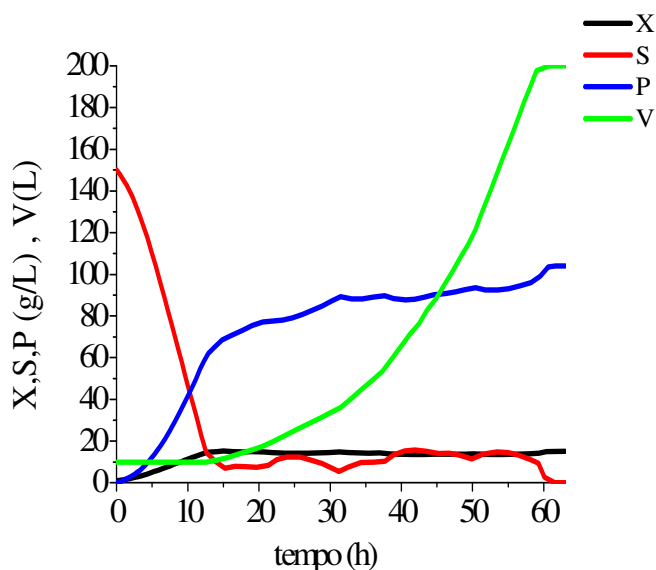


Figura 4.16: Perfis das variáveis de estado X,S,P e V obtidas por esse trabalho(tempo final fixo de 63 horas).

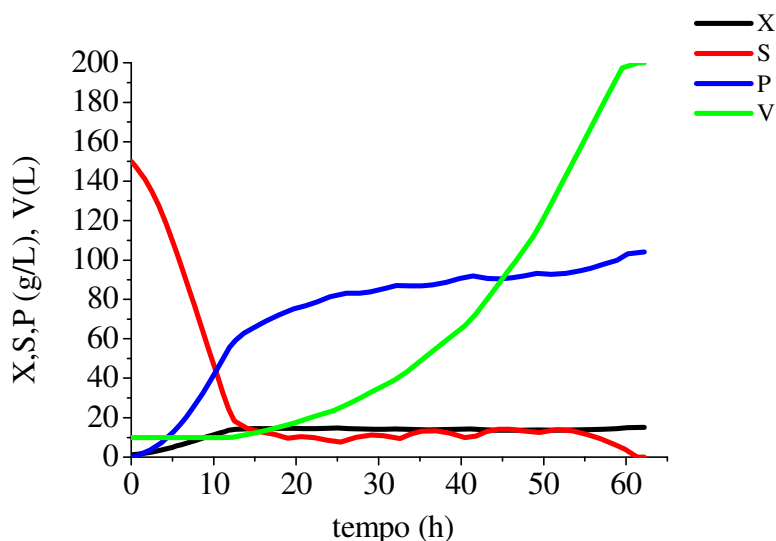


Figura 4.17: Perfis das variáveis de estado X,S,P e V obtidas por esse trabalho (tempo final livre)

Tabela 4.15: Resultados obtidos por diferentes técnicas de solução (considerando tempo final fixo e tempo final livre)

Função Objetivo	Método de Solução	Referência	Nº de Avaliações de Função Objetivo
21140,2	Indireto (PVC)	Fu e Barford(1993)	---
20841,1	Algoritmo L J	Luus e Henessy(1999)	184000-890000
20838	PDI	Bojkov e Luus (1995)	30000-75000
20797,41	ED (tempo livre)	Este trabalho	20050
20793,32	ED (tempo fixo)	Este trabalho	15050-20050
20749,69	Algoritmos Genéticos	Upreti (2004)	1200000

Da Tabela 4.15, o valor de Função Objetivo encontrado por Fu e Barford (1993) foi para um tempo de 56 h de operação batelada. Observa-se da Tabela 4.15 que os resultados obtidos por Luus e Henessy (1999) e Bojkov e Luus (1996) são 0,23% e 0,19% superiores ao valor obtido por este trabalho. O número de avaliações de Função Objetivo necessárias em Luus e Henessy (1999) é cerca de 9 até 44 vezes superior ao número de avaliações realizadas por este trabalho. O número de avaliações de Função Objetivo necessárias em Bojkov e Luus é cerca de 1,5 até 3,5 vezes superior ao número de avaliações realizadas por este trabalho. O resultado obtido por Bojkov e Luus (1996) foi para um tempo de 61,3 horas de operação batelada. O melhor resultado de Função Objetivo obtido por Upreti (2004) é 0,21% inferior ao resultado obtido por este trabalho. O número de avaliações de Função Objetivo em Upreti (2004) é cerca de 60 vezes superior ao número de avaliações realizadas por este trabalho.

4.6. Caso 6. Problema com três restrições de trajetória na variável de estado e tempo final fixo: Produção de Penicilina

O problema de índice 2 descrito em Cuthrell e Biegler (1989), Luus (1992), Luus e Henessy (1999), Upreti (2004) tem por objetivo determinar o perfil de uma variável de controle u (vazão de alimentação) que maximize a produção de penicilina.

Cuthrell e Biegler (1989) utilizaram o método direto simultâneo na solução do problema através de uma estratégia baseada em Programação Quadrática Sucessiva (SQP) e colocação ortogonal sobre elementos finitos.

Luus (1992) solucionou o problema utilizando Programação Dinâmica Iterativa com elementos de controle de mesmo tamanho e funções penalidade para manipular as restrições de estado.

O problema pode ser formulado (conforme apresentado em Cuthrell e Biegler (1989)) da seguinte maneira:

$$\max J = P(t_f)V(t_f) \quad (4.36)$$

$$\dot{X} = \mu(X,S)X - \left(\frac{X}{S_F V}\right)U \quad (4.37)$$

$$\dot{P} = \rho(S)X - K_{degr}P - \left(\frac{P}{S_F V}\right)U \quad (4.38)$$

$$\dot{S} = -\mu(X,S)\left(\frac{X}{Y_{X/S}}\right) - \rho(S)\left(\frac{X}{Y_{P/S}}\right) - \left(\frac{m_s S}{K_m + S}\right) + \left(1 - \frac{S}{S_F}\right)\frac{U}{V} \quad (4.39)$$

$$\dot{V} = \frac{U}{S_F} \quad (4.40)$$

$$\mu(X,S) = \mu_{max} \left(\frac{S}{K_x X + S} \right) \quad (4.41)$$

$$\rho(S) = \rho_{max} \left(\frac{S}{K_p + S \left(1 + \frac{S}{K_{in}} \right)} \right) \quad (4.42)$$

X , P , S e V representam a concentração em massa (g L^{-1}) de biomassa, produto, substrato e volume em litros, $\mu(X,S)$ representa a taxa de crescimento da biomassa (h^{-1}), $\rho(S)$ representa a taxa de produção de penicilina ($\text{g de P} \times \text{g}^{-1} \text{ h}^{-1}$ de X). São considerados os seguintes valores para as constantes cinéticas: $\mu_{max} = 0,11 \text{ h}^{-1}$; $\rho_{max} = 0,0055 (\text{g de P} \times \text{g}^{-1} \text{ h}^{-1} \text{ de X})$; $K_x = 0,006 (\text{g de S} \times \text{g}^{-1} \text{ de X})$; $K_p = 0,0001 (\text{g L}^{-1} \text{ de S})$; $K_{in} = 0,1 (\text{g L}^{-1} \text{ de S})$; $K_{degr} = 0,01 \text{ h}^{-1}$; $K_m = 0,0001 (\text{g L}^{-1} \text{ de S})$; $m_s = 0,029 (\text{g de S} \times \text{g}^{-1} \text{ h}^{-1} \text{ de X})$, $Y_{X/S} = 0,47 (\text{g de X} \times \text{g}^{-1} \text{ de S})$; $Y_{P/S} = 1,2 (\text{g de P} \times \text{g}^{-1} \text{ de S})$. E uma concentração $S_F = 500 \text{ g L}^{-1}$.

O problema pode ser reformulado (formulação utilizada neste trabalho) conforme apresentado em Luus e Henessy (1999)

$$\max J = x_2(t_f)x_4(t_f) \quad (4.43)$$

$$\dot{x}_1 = h_1 x_1 - u \frac{x_1}{500 x_4} \quad (4.44)$$

$$\dot{x}_2 = h_2 x_1 - 0,01 x_2 - u \left(\frac{x_2}{500 x_4} \right) \quad (4.45)$$

$$\dot{x}_3 = -\frac{h_1 x_1}{0,47} - \frac{h_2 x_1}{1,2} - \frac{0,029 x_3 x_1}{0,0001 + x_3} + \frac{u}{x_4} \left(1 - \frac{x_3}{500} \right) \quad (4.46)$$

$$\dot{x}_4 = \frac{u}{500} \quad (4.47)$$

$$\mathbf{x}^T(0) = [1,5 \ 0 \ 0 \ 7] \quad (4.48)$$

$$0 \leq u \leq 50 \quad (4.49)$$

$$0 \leq x_1 \leq 40 \quad (4.50)$$

$$0 \leq x_3 \leq 25 \quad (4.51)$$

$$0 \leq x_4 \leq 10 \quad (4.52)$$

$$h_1 = \left(\frac{0,11 x_3}{0,006 x_1 + x_3} \right) \quad (4.53)$$

$$h_2 = \left(\frac{0,0055 x_3}{0,0001 + x_3 (1 + 10 x_3)} \right) \quad (4.54)$$

Os parâmetros do algoritmo de Evolução Diferencial utilizados na solução do problema são apresentados na Tabela 4.16.

Tabela 4.16- Parâmetros utilizados na solução do problema da Produção de Penicilina

Número de Indivíduos da População	50
Número de Gerações	100, 200, 300
Taxa de Perturbação	0,5
Probabilidade de Cruzamento	0,8
Estratégia	7 (ED/best/1/exp)
Vetor de Sementes	[0 1 2 3 4 5 6 7 8 9 10]

Para assegurar que o limite imposto para três das variáveis de estado não fosse ultrapassado, foram utilizadas funções penalidade para cada uma das variáveis com o fator penalidade estática $rp=10^5$. Na solução do problema, as variáveis de decisão para o algoritmo de Evolução Diferencial foram o instante de atuação dos elementos de controle (tempos de atuação) e os elementos de controle.

Para a solução do Sistema de Equações Diferenciais (Equações 4.41 - 4.45) foi empregado o método de Runge-Kutta 4ª e 5ª ordem com passo de integração variável, tolerância relativa de 10^{-12} e tolerância absoluta de 10^{-12} através da rotina ode45 disponível no software Matlab®.

Na Tabela 4.17 são apresentados os resultados obtidos para a Função Objetivo (Equação 4.41) para diferentes sementes e diferentes números de gerações.

Tabela 4.17- Resultados obtidos para diferentes sementes e diferentes números de gerações na solução do problema de Produção de Penicilina

Semente	$N_{\text{gerações}}$	FO(g)	$N_{\text{gerações}}$	FO(g)	$N_{\text{gerações}}$	FO(g)
0	100	87,8427	200	87,8680	300	87,9119
1		87,7045		87,8413		87,8549
2		87,8491		87,8996		87,8996
3		87,8176		87,8816		87,9274
4		87,7304		87,8735		87,8998
5		87,8033		87,8646		87,8858
6		87,8008		87,8338		87,8462
7		87,8300		87,8902		87,9121
8		87,8362		87,8615		87,9104
9		87,8219		87,9012		87,9151
10		87,8199		87,8399		87,9154
Média		87,8051		87,8686		87,8980

Da Tabela 4.17, observa-se que o aumento do número de gerações até $N_{\text{gerações}}=300$ melhora os resultados de Função Objetivo. Exceto para a semente 2 em que o aumento de gerações de duzentos para trezentos não provocou nenhuma alteração, em todos os demais casos houve uma melhora do valor de Função Objetivo. O melhor valor de Função Objetivo encontrado foi para a semente 3 e 300 gerações (FO=87,9274 g).

Na Figura 4.18 é apresentado o perfil da variável de controle obtido por este trabalho, comparado com o perfil obtido por Luus e Henessy (1999) e Luus (1995). Na Figura 4.19 são apresentados os perfis das variáveis de estado.

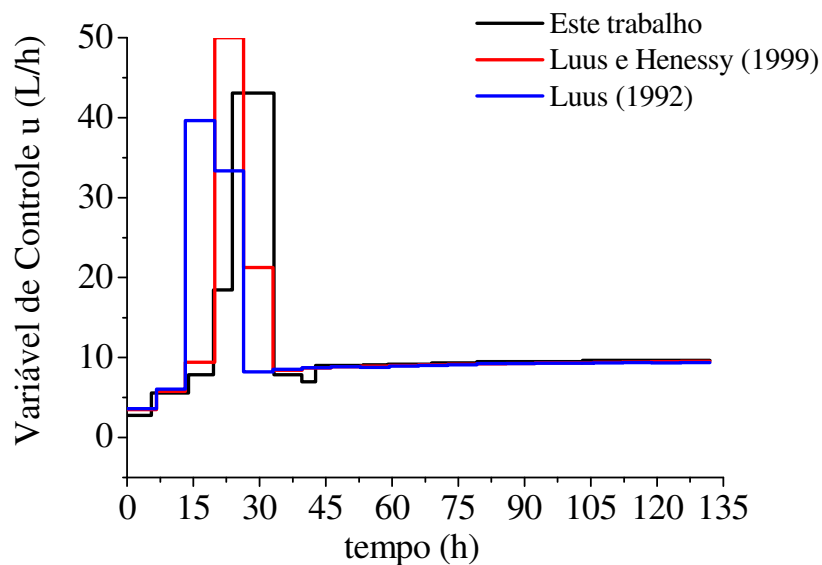


Figura 4.18: Perfis da variável de controle u para o problema de Produção de Penicilina

Observa-se da Figura 4.18, inicialmente os diferentes perfis se assemelham até aproximadamente 15 horas de batelada. Entre 15 e 45 horas os perfis são distintos e após 45 horas os perfis voltam a apresentar uma semelhança.

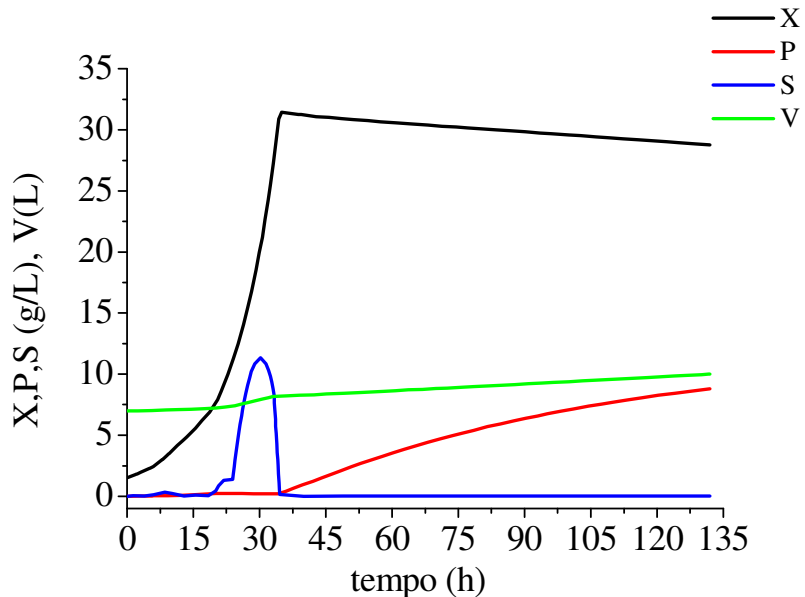


Figura 4.19: Perfis das variáveis de estado X , P , S e V obtidas para esse trabalho

Observa-se na Figura 4.19 que inicialmente há o crescimento da biomassa e as concentrações de produto, substrato e o volume são mantidos aproximadamente constantes. Após aproximadamente 23,5 horas de batelada, a concentração de substrato aumenta até aproximadamente 30 horas. De 30 até aproximadamente 35 horas, a

concentração de substrato diminui e a partir de 35 horas a concentração do produto formado aumenta, aumentando também o volume e diminuindo a concentração de biomassa. Na Tabela 4.18 são apresentados o melhor valor de Função Objetivo obtido por este trabalho e valores de Função Objetivo obtidos por diferentes referências. São apresentados também o número de avaliações de Função Objetivo necessárias para encontrar a solução ótima.

Tabela 4.18: Resultados obtidos por diferentes técnicas de solução para o problema de produção de penicilina

Função Objetivo	Método de Solução	Referência	Nº de Avaliações de Função Objetivo
87,9954	Algoritmo L J	Luus e Henessy(1999)	81000
87,95	PDI	Luus (1992)	---
87,9274	Evolução Diferencial	Este trabalho	15050
87,8729	Algoritmos Genéticos	Upreti (2004)	1866400
87,69	Direto Simultâneo	Cuthrell e Biegler(1989)	32

Observa-se da Tabela 4.18 que os resultados obtidos por Luus e Henessy (1999) e Luus (1992) são 0,08% e 0,026% superiores ao valor obtido por este trabalho. O número de avaliações de Função Objetivo necessárias em Luus e Henessy (1999) é cerca de 5,5 vezes superior ao número de avaliações realizadas por este trabalho. O melhor resultado de Função Objetivo obtido por Upreti (2004) é 0,062% inferior ao resultado obtido por este trabalho. O número de avaliações de Função Objetivo em Upreti (2004) é cerca de 124 vezes superior ao número de avaliações realizadas por este trabalho.

Demonstrada a capacidade do Algoritmo de Evolução Diferencial para resolver Problemas de Otimização Dinâmica fartamente reportados na literatura devido à suas características e dificuldades de solução, através da obtenção de resultados compatíveis com esforço computacional condizente em todos os casos apresentados, será avaliada na sequência uma análise de sensibilidade paramétrica do Algoritmo de Evolução Diferencial. Nesta análise, cujos resultados são restritos aos problemas apresentados, é possível avaliar como o desempenho do algoritmo é afetado por variações nos parâmetros do algoritmo. Desta forma, para cada problema a ser resolvido empregando esta técnica, a correta determinação dos valores destes parâmetros é fundamental para uma solução bem sucedida.

4.7. Análise de sensibilidade paramétrica do algoritmo de Evolução Diferencial para uma população com número fixo de indivíduos

O desempenho do algoritmo de Evolução Diferencial está diretamente associado à definição de parâmetros tais como: Fator de Perturbação ($F \in [0, 2]$), Probabilidade de Cruzamento ($P_c \in [0, 1]$), Número de Gerações (NG). A definição de um conjunto de parâmetros ótimos depende do problema a ser resolvido, podendo variar de problema para problema. Na literatura são apresentados trabalhos onde é realizada a análise de parâmetros do algoritmo de Evolução Diferencial para diferentes problemas.

Oliveira (2006) realizou a análise para a minimização da função de Rastrigin. Zaharie (2009) estudou a influencia da taxa de cruzamento no algoritmo de Evolução Diferencial através de dois problemas teste, a função de Rastrigin e a função de Griewank. Mandal e Chakraborty (2012) realizaram um estudo dos parâmetros para a solução de um problema de scheduling ótimo de sistemas hidrotermais. Com o objetivo de se avaliar a sensibilidade paramétrica do Algoritmo de Evolução Diferencial, foram feitas simulações com uma população NP de 100 indivíduos, um vetor de sementes, $[0, 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9]$ e com estratégia de número 7 (ED/best/1/exp) para dois problemas clássicos, onde o perfil da variável de controle foi obtido através da identificação dos estágios de controle e dos instantes de atuação dos estágios. O primeiro problema é denominado Mistura de Catalisadores e tem índice diferencial 3 - as variáveis de decisão foram compostas por três estágios de controle e dois instantes de atuação em que $u \in [0, 1]$ e $t_1 \in [0,05, 0,5]$ e $t_2 \in [0,51, 1]$. O segundo problema, denominado Reator Batelada com reação consecutiva, tem índice diferencial 1. As variáveis de decisão foram compostas por dez estágios de controle $u \in [300, 400]$ e nove instantes de atuação $t_1 \in [0, 3,33]$, $t_2 \in [3,33, 6,6]$, $t_9 \in [26,6, 30]$.

Para a avaliação do efeito da variação dos parâmetros da Evolução Diferencial foram realizadas 160 simulações para cada problema através do algoritmo implementado em linguagem C. O método utilizado para a solução do sistema de equações diferenciais foi o Runge-Kutta 4ª ordem com passo de integração constante e 1000 pontos no domínio tempo.

Para todas as análises, foram considerados os valores de Função Objetivo obtidos para os seguintes conjuntos: $\{F=0,2; P_c=0,3 ; NG=50\}$, $\{F=0,2; P_c=0,3 ; NG=250\}$, $\{F=0,8 ; P_c=0,3; NG=50\}$, $\{F=0,8; P_c=0,3; NG=250\}$, $\{F=1,2 ; P_c=0,3;$

NG=50}, {F=1,2; Pc=0,3; NG=250}, {F=2,0; Pc=0,3; NG=50}, {F=2,0; Pc=0,3; NG=250}.

4.7.1. Mistura de catalisadores em um reator PFR

O problema de índice diferencial 3 proposto por Gunn e Thomas em 1965 considera um reator PFR recheado com dois catalisadores onde ocorre uma reação do tipo $S_1 \leftrightarrow S_2 \rightarrow S_3$. O objetivo é determinar a melhor mistura entre os dois catalisadores ao longo do comprimento fixo do reator capaz de maximizar a produção de S_2 minimizando $J[u]$ (concentração de S_3). Na formulação apresentada a seguir, x_1 e x_2 representam as concentrações de duas espécies denominadas de S_1 e S_2 respectivamente e a variável de controle u representa a taxa de mistura dos catalisadores.

$$J[u] = 1 - x_1(t_f) - x_2(t_f) \quad (4.55)$$

$$\dot{x}_1 = u(10x_2 - x_1), \quad x_1(0) = 1 \quad (4.56)$$

$$\dot{x}_2 = u(x_1 - 10x_2) - (1 - u)x_2, \quad x_2(0) = 0 \quad (4.57)$$

$$0 \leq u \leq 1 \quad (4.58)$$

Na sequência, os resultados obtidos através da análise serão apresentados através de gráficos boxplot.

- *Análise dos valores de Função Objetivo com o aumento do Número de Gerações (NG) considerando o mesmo Fator de Perturbação(F) para diferentes Probabilidades de Cruzamento (Pc).*

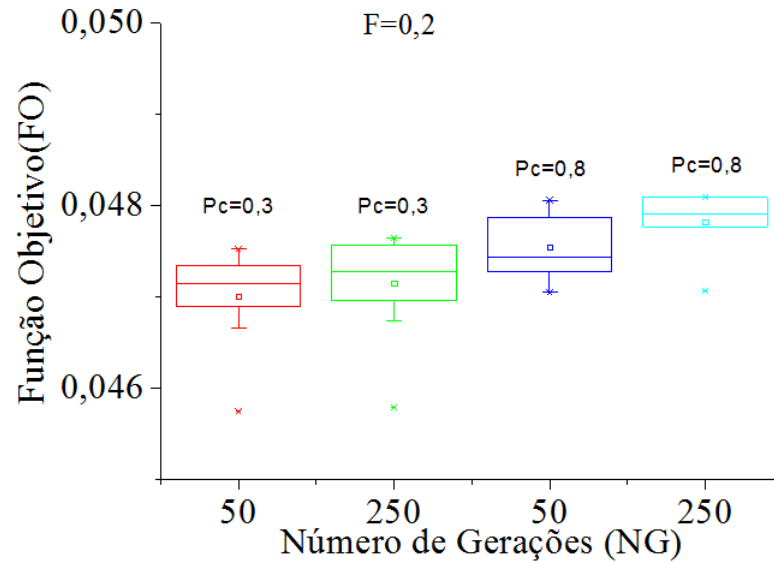


Figura 4.20: Função Objetivo x Número de Gerações para $F=0,2$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR.

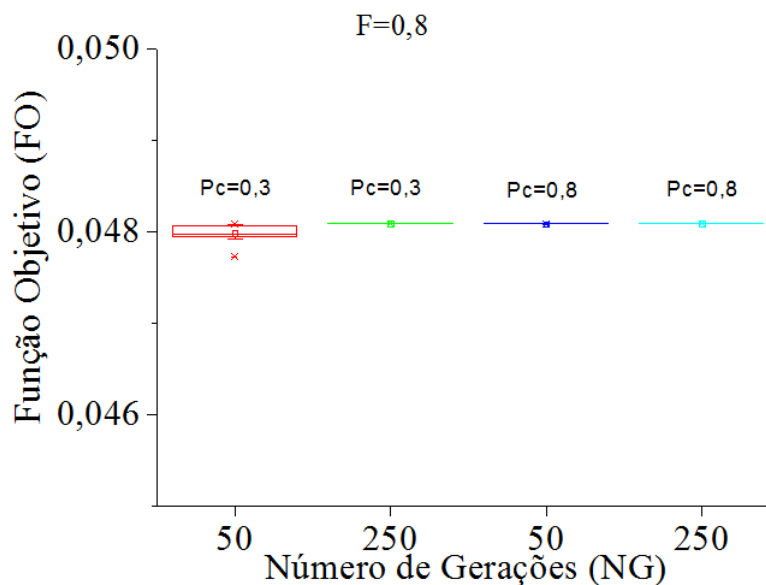


Figura 4.21: Função Objetivo x Número de Gerações para $F=0,8$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR

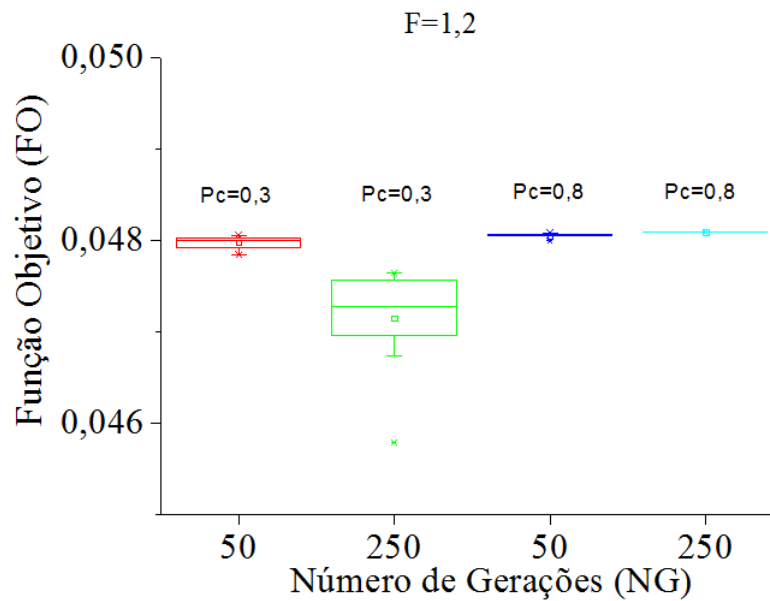


Figura 4.22: Função Objetivo x Número de Gerações para $F=1,2$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR

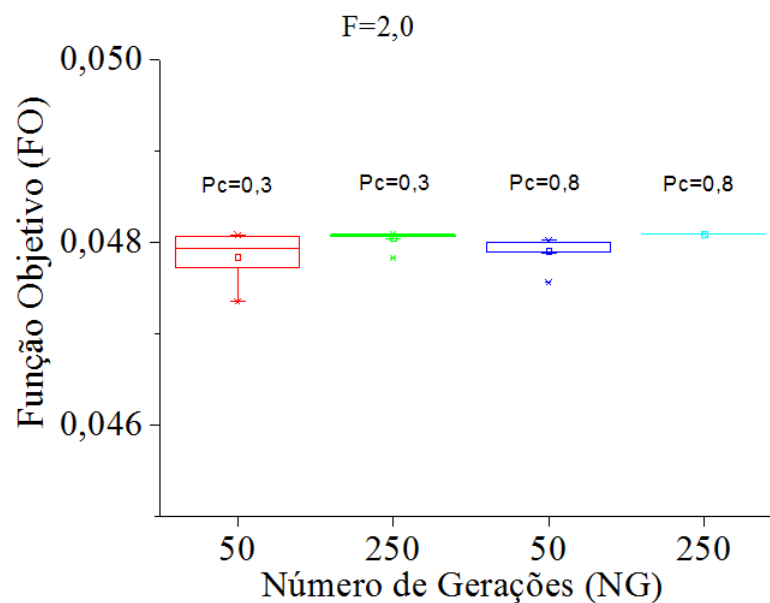


Figura 4.23: Função Objetivo x Número de Gerações para $F=2,0$; $P_c=0,3$ e $0,8$ para o problema de mistura de catalisadores em um reator PFR

- Analisando-se a Figura 4.20 observa-se que independentemente do número de gerações, o valor médio de Função Objetivo obtido para $P_c=0,8$ é melhor que o valor médio para $P_c=0,3$. Considerando uma probabilidade de cruzamento $P_c=0,3$, o aumento do número de gerações apresentou uma maior dispersão dos valores de Função Objetivo. Para uma probabilidade de cruzamento $P_c=0,8$, o

aumento do número de gerações apresentou uma menor dispersão dos valores de Função Objetivo.

- Analisando-se a Figura 4.21 para um número de gerações $NG=50$, o valor médio de Função Objetivo é melhor para uma probabilidade de cruzamento $P_c=0,8$. Para um número de gerações $NG=250$, o valor médio de Função Objetivo é melhor para uma probabilidade de cruzamento $P_c=0,3$. Observa-se que houve dispersão dos valores de Função Objetivo apenas para um número de gerações $NG=50$ e probabilidade de cruzamento $P_c = 0,3$.
- Analisando-se as Figuras 4.22 e 4.23 para um número de gerações $NG=50$, o valor médio da função Objetivo é melhor para uma probabilidade de cruzamento $P_c=0,8$. Para um número de gerações $NG=250$, os valores médios de Função Objetivo são os mesmos para $P_c=0,3$ e $P_c=0,8$. De uma maneira geral, houve uma tendência maior de dispersão dos valores de Função Objetivo para uma probabilidade de cruzamento $P_c=0,3$ independentemente do fator de perturbação e do número de gerações.
- *Análise dos valores de Função Objetivo com o aumento Probabilidade de Cruzamento (P_c) considerando o mesmo Fator de Perturbação(F) para diferentes Números de Gerações (NG).*

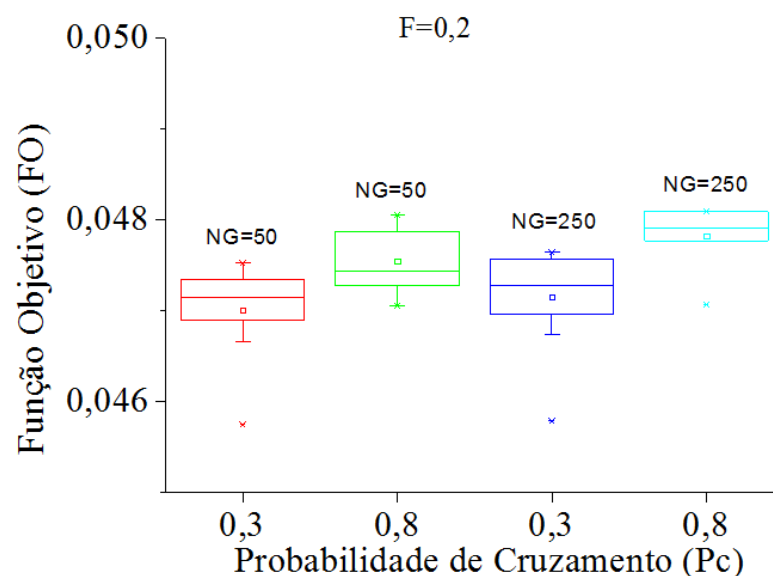


Figura 4.24: Função Objetivo x Probabilidade de Cruzamento para $F=0,2$; $NG=50$ e 250 para o problema de mistura de catalisadores em um reator PFR

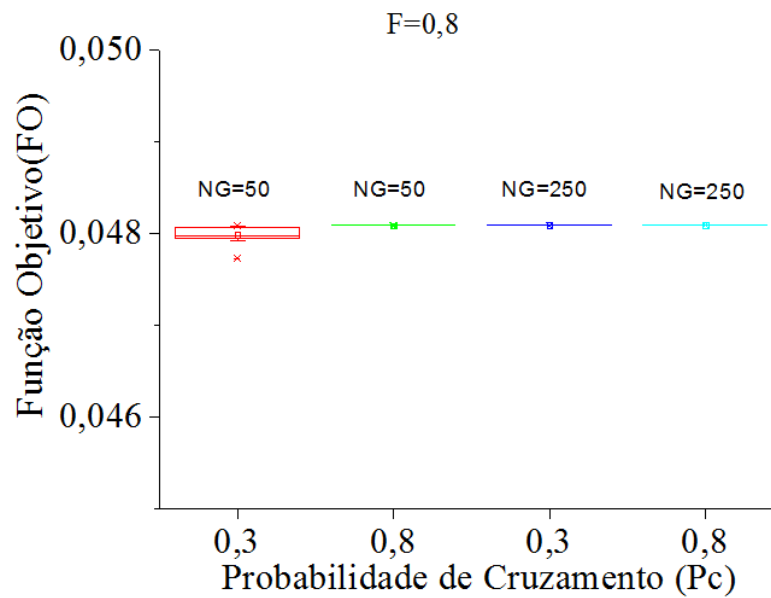


Figura 4.25: Função Objetivo x Probabilidade de Cruzamento para $F=0,8$; NG=50 e 250 para o problema de mistura de catalisadores em um reator PFR

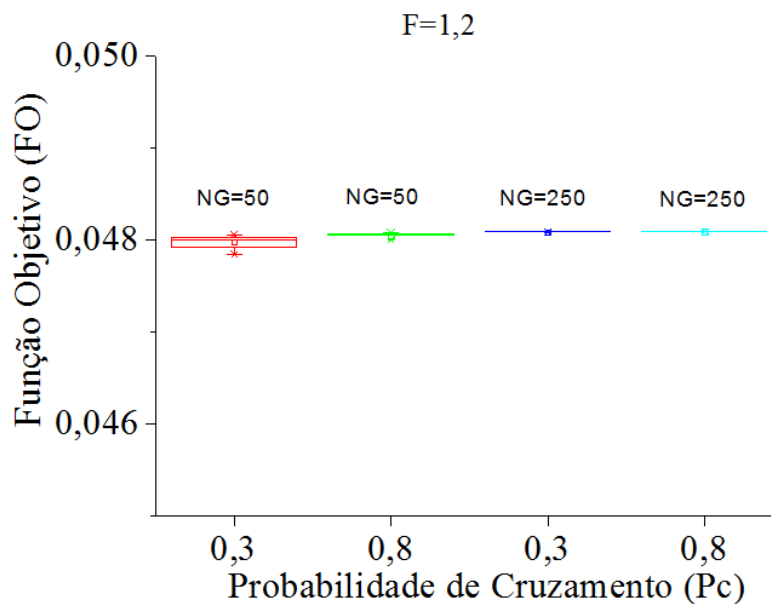


Figura 4.26: Função Objetivo x Probabilidade de Cruzamento para $F=1,2$; NG=50 e 250 para o problema de mistura de catalisadores em um reator PFR

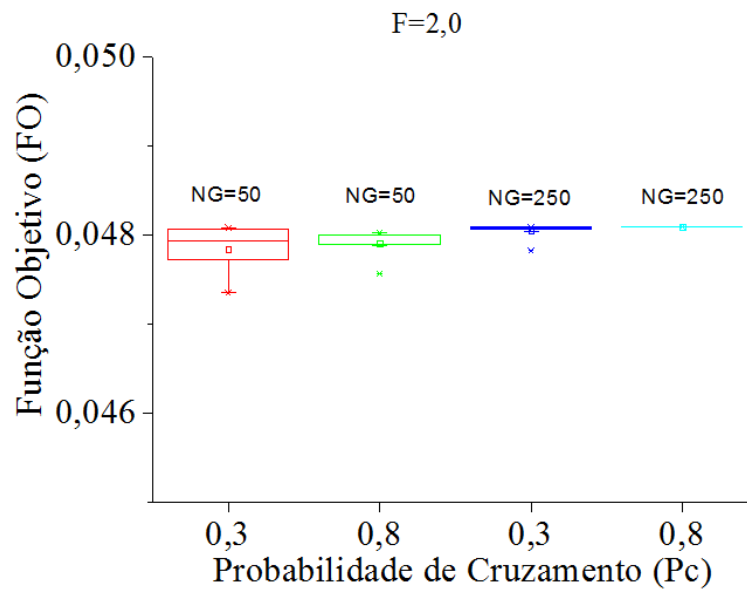


Figura 4.27: Função Objetivo x Probabilidade de Cruzamento para $F=2,0$; $NG=50$ e 250 para o problema de mistura de catalisadores em um reator PFR

- Observa-se na Figura 4.24 que independentemente da probabilidade de cruzamento, o valor médio de Função Objetivo melhorou com o aumento do número de gerações. A dispersão dos valores de Função Objetivo pouco mudou com o aumento do número de gerações para ambas as probabilidades de cruzamento $P_c=0,3$ e $P_c=0,8$.
- Observa-se na Figura 4.25 que para uma probabilidade de cruzamento $P_c=0,8$ o valor médio de Função Objetivo é o mesmo para um número de gerações $NG=50$ e $NG=250$. Observa-se assim que para $F=0,8$ e $P_c=0,8$, um número de gerações $NG=50$ é suficiente para a convergência da solução. Uma pequena dispersão dos valores de Função Objetivo ocorreu para $P_c=0,3$ e $NG=50$.
- Observa-se na Figura 4.26 que o valor médio de Função Objetivo foi o mesmo para ambas as probabilidades de cruzamento considerando um número de gerações $NG=250$. Para um número de gerações $NG=50$ e probabilidade de cruzamento $P_c=0,8$, o valor médio de Função Objetivo foi próximo do obtido para um número de gerações $NG=250$. Novamente uma pequena dispersão é observada em $P_c=0,3$ e $NG=50$.
- Observa-se na Figura 4.27 que independentemente da probabilidade de cruzamento, o valor médio de Função Objetivo foi um pouco melhor com o

aumento do número de gerações. A dispersão foi maior para uma probabilidade de cruzamento $P_c=0,3$ em ambos os números de gerações.

- *Análise dos valores de Função Objetivo com o aumento do Fator de Perturbação (F) para diferentes Probabilidades de Cruzamento (PC) e mesmo Número de Gerações (NG).*

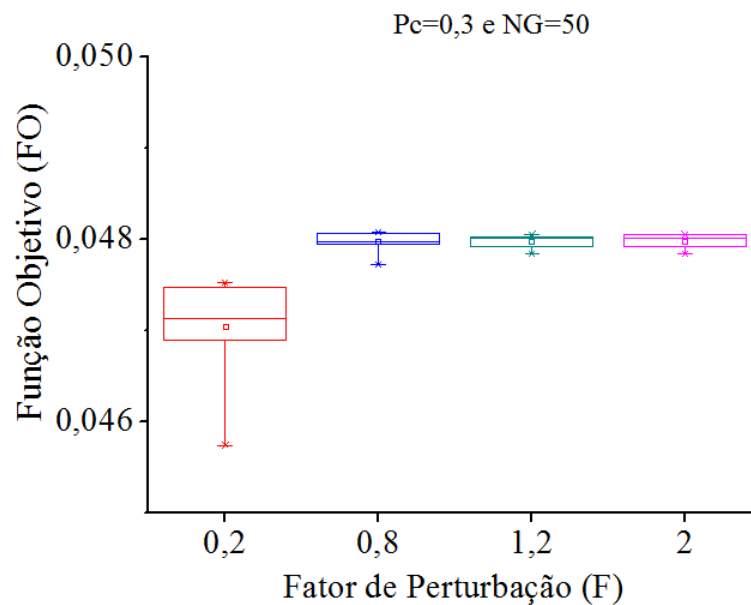


Figura 4.28: Função Objetivo x Fator de Perturbação para $P_c=0,3$; $NG=50$ para o problema de mistura de catalisadores em um reator PFR

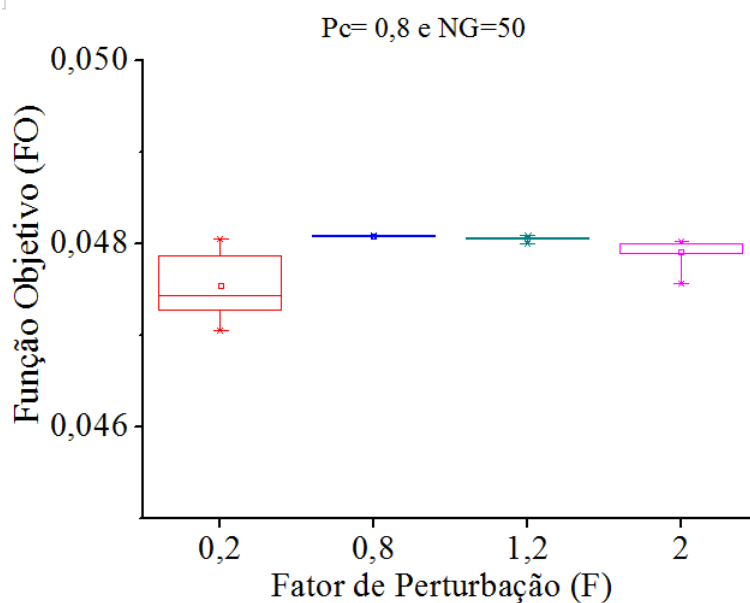


Figura 4.29: Função Objetivo x Fator de Perturbação para $P_c=0,8$; $NG=50$ para o problema de mistura de catalisadores em um reator PFR

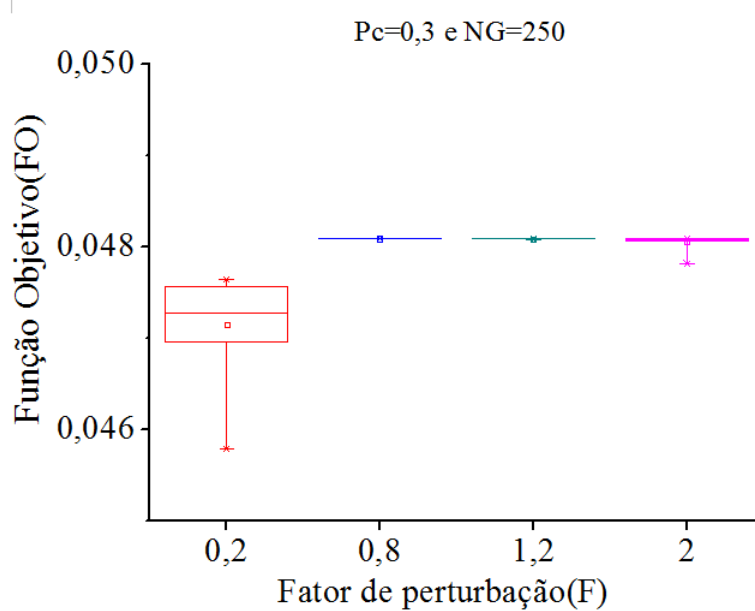


Figura 4.30: Função Objetivo x Fator de Perturbação para $P_c=0,3$; $NG=250$ para o problema de mistura de catalisadores em um reator PFR

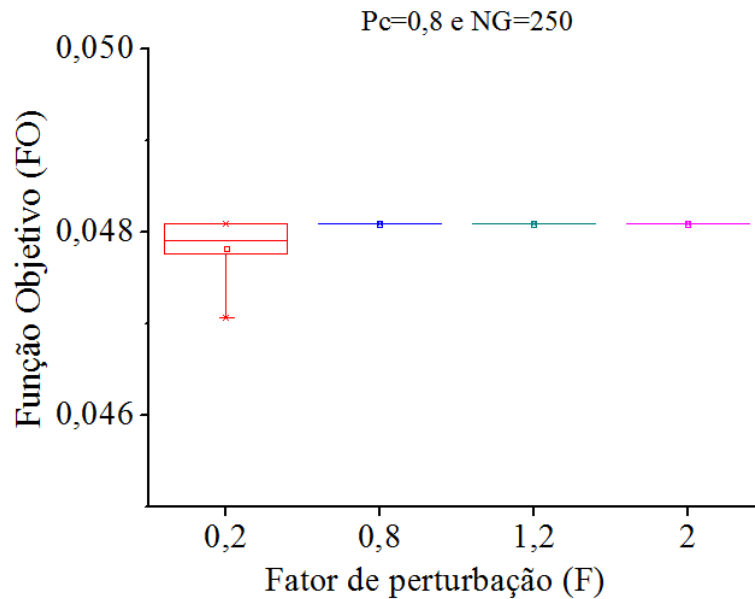


Figura 4.31: Função Objetivo x Fator de Perturbação para $P_c=0,8$; $NG=250$ para o problema de mistura de catalisadores em um reator PFR

- Através das Figuras 4.28 e 4.29 observa-se que tanto para $P_c=0,3$ quanto para $P_c=0,8$, o fator de perturbação $F=0,8$ apresenta o melhor valor médio de Função Objetivo. A dispersão dos valores de Função Objetivo foi em geral menor para $P_c=0,8$ principalmente para $F=0,8$ e $F=1,2$.
- Através das Figuras 4.30 e 4.31, observa-se que o fator de perturbação $F=0,8$ apresentou o melhor valor médio de Função Objetivo para $P_c=0,3$. Para uma probabilidade de cruzamento $P_c=0,8$, os fatores $F=0,8$, $F=1,2$ e $F=2$ apresentaram o mesmo valor médio de Função Objetivo. A dispersão dos valores de Função Objetivo foi menor para $P_c=0,8$ principalmente para $F=0,2$ e $F=2$.

Através da análise geral considerando todas as variações de parâmetros, observa-se que os conjuntos $\{F=0,8, P_c=0,8, NG=250\}$, $\{F=1,2, P_c=0,8, NG=250\}$, $\{F=2, P_c=0,8, NG=250\}$, $\{F=0,8, P_c=0,3, NG=250\}$ e $\{F=1,2, P_c=0,3, NG=250\}$ apresentaram o melhor valor médio de Função Objetivo. Entretanto, considerando diferentes números de gerações $NG=50$ e $NG=250$, os parâmetros $F=0,8$ e $P_c=0,8$ apresentaram uma menor dispersão dos valores de Função Objetivo.

Na Figura 4.32 é apresentado o perfil da variável de controle obtido por este trabalho comparado com o perfil obtido por outros autores. O perfil das variáveis de

estado é apresentado na Figura 4.33. A Tabela 4.19 apresenta o melhor valor encontrado por este trabalho comparando com valores reportados da literatura.

Tabela 4.19: Valores de Função Objetivo obtidos por diferentes técnicas para o problema de mistura de catalisadores em um reator PFR

Função Objetivo	Método de solução	Referência
0,048087	Evolução Diferencial	Este trabalho
0,0480800	Indireto (PVC)	Bell e Sargent (2000)
0,0480557	Abordagem Analítica	Corrêa do Quinto (2010)
0,048054	Híbrido	Lobato (2004)

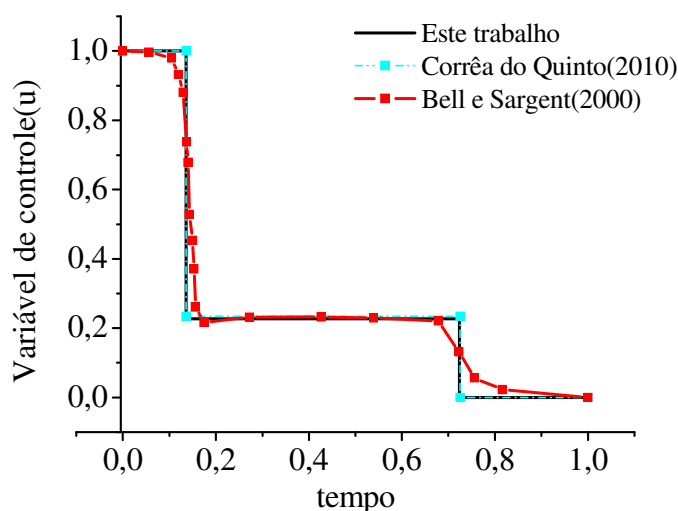


Figura 4.32: Perfil da variável de controle (u) para o problema de mistura de catalisadores em um reator PFR

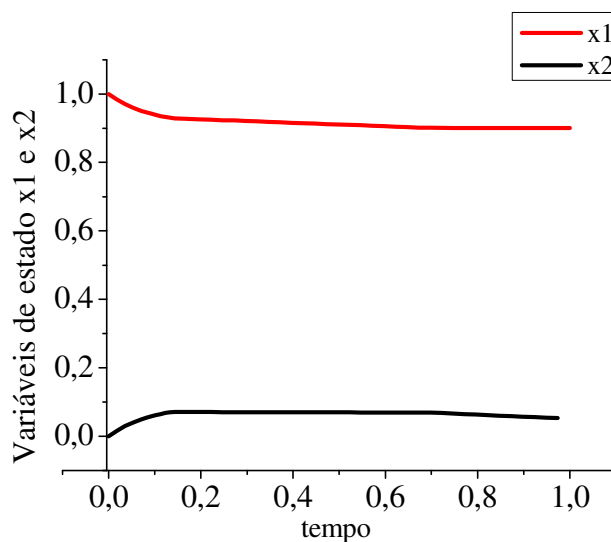


Figura 4.33: Perfis das variáveis de estado x_1 e x_2 obtidas por este trabalho para o problema de mistura de catalisadores em um reator PFR.

4.7.2. Reator Batelada com reação consecutiva

O problema de índice 1 que foi considerado em trabalhos como em Bilous e Amundson (1956) e Luus e Okongwu (1999) consiste em encontrar um perfil de temperatura que maximize a concentração de um produto de interesse B segundo a reação consecutiva $A \rightarrow B \rightarrow C$. As reações tanto de produção quanto de consumo de B são de primeira ordem. Os balanços materiais das espécies A e B são apresentados nas Equações 4.60 e 4.61. O tempo final da batelada é de 30 minutos.

$$\max J = x_2(t_f) \quad (4.59)$$

$$\dot{x}_1 = -k_1 x_1 \quad (4.60)$$

$$\dot{x}_2 = k_1 x_1 - k_2 x_2 \quad (4.61)$$

Onde x_1 e x_2 representam as concentrações em mol/L das espécies A e B respectivamente. As constantes k_1 e k_2 e as condições iniciais são apresentadas nas Equações (4.58), (4.59) e (4.60).

$$k_1 = 5,35 \times 10^{10} \exp\left(-\frac{9000}{T}\right) \quad (4.62)$$

$$k_2 = 4,61 \times 10^{17} \exp\left(-\frac{15000}{T}\right) \quad (4.63)$$

$$x(0)^T = [0,95 \ 0,05] \quad (4.64)$$

Para a análise de sensibilidade, os conjuntos dos parâmetros utilizados para este problema são os mesmos do item 4.6.1.

- *Análise dos valores de Função Objetivo com o aumento do Número de Gerações (NG) considerando o mesmo Fator de Perturbação(F) para diferentes Probabilidades de Cruzamento (Pc).*

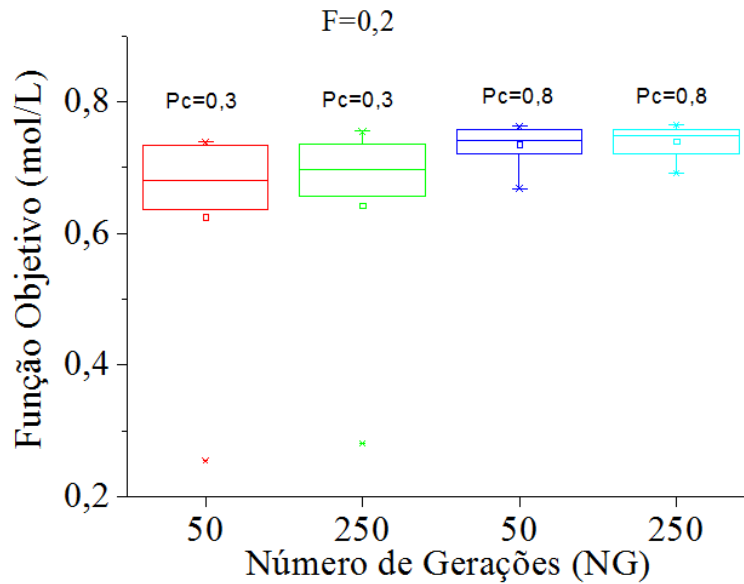


Figura 4.34: Função Objetivo x Número de Gerações para $F=0,2$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva

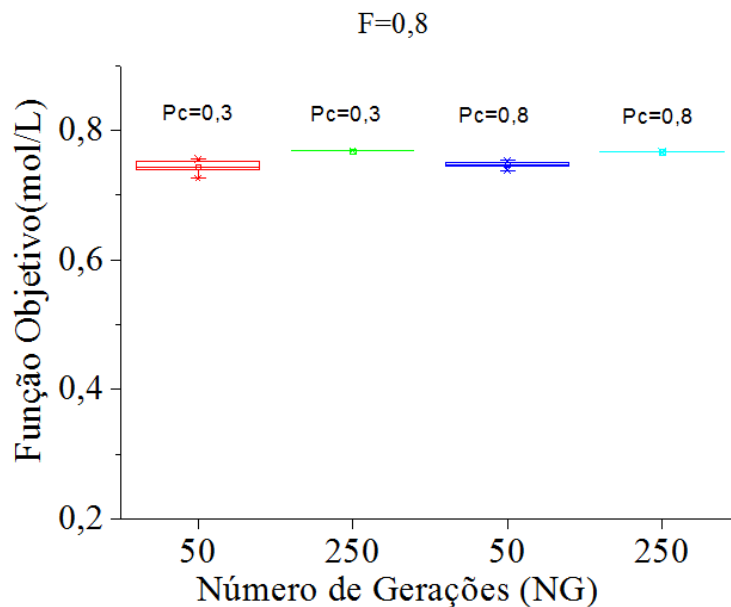


Figura 4.35: Função Objetivo x Número de Gerações para $F=0,8$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva

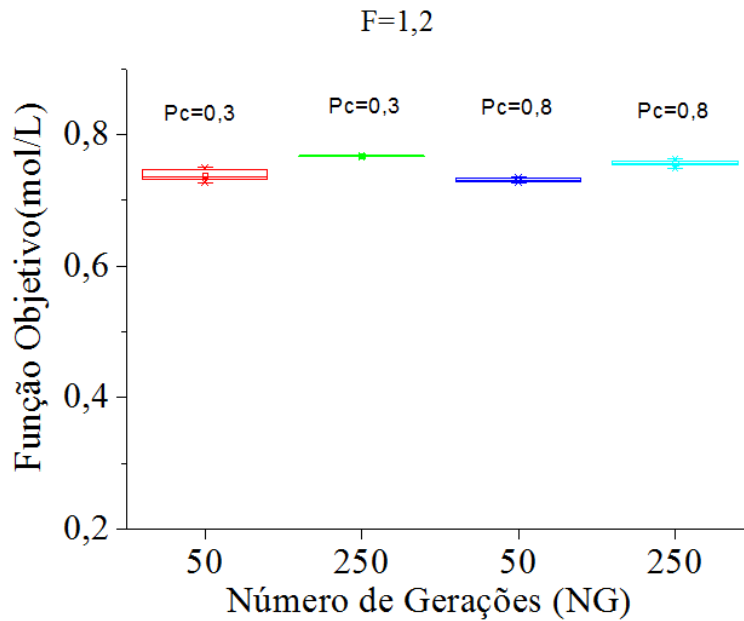


Figura 4.36: Função Objetivo x Número de Gerações para $F=1,2$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva

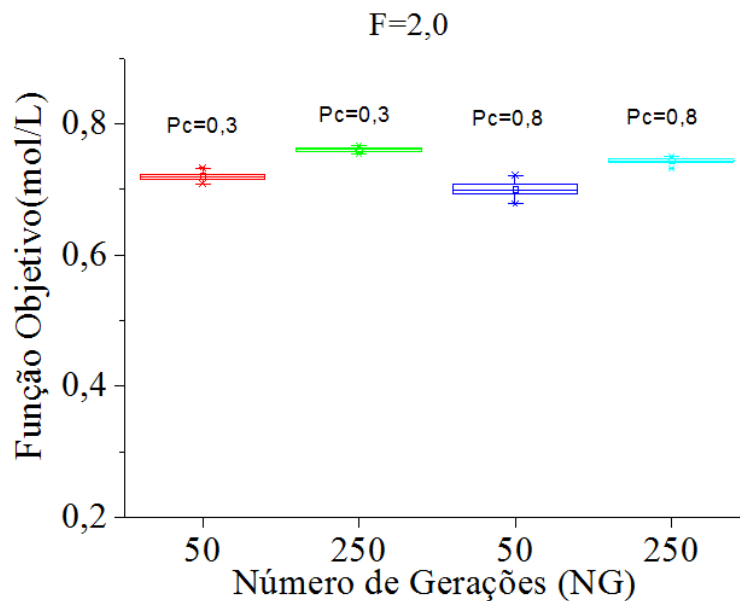


Figura 4.37: Função Objetivo x Número de Gerações para $F=2,0$; $P_c=0,3$ e $0,8$ para o problema do Reator Batelada com reação consecutiva

- Analisando a Figura 4.34 observa-se que independentemente do número de gerações, o valor médio de Função Objetivo obtido para $P_c=0,8$ é melhor que o valor médio para $P_c=0,3$. A dispersão dos valores de Função Objetivo tende a ser menor com o aumento de gerações, principalmente para $P_c=0,8$.

- Analisando a Figura 4.35, independentemente do número de gerações os valores médios da Função Objetivo são praticamente os mesmos para $P_c=0,8$ e $P_c=0,3$. A dispersão dos valores de Função Objetivo foi menor que em $F=0,2$.
- A análise da Figura 4.36 demonstra que independentemente do número de gerações, os valores médios de Função Objetivo para $P_c=0,3$ foram melhores que os valores médios para $P_c=0,8$.
- Analisando a Figura 4.37, observa-se que o valor médio da Função Objetivo é maior para $P_c=0,3$ tanto para $NG=50$ quanto para $NG=250$. A dispersão é maior que $F=0,8$ e $F=1,2$.
- *Análise dos valores de Função Objetivo com o aumento da Taxa de Cruzamento (CR) considerando o mesmo Fator de Perturbação (F) para diferentes Números de Gerações.*

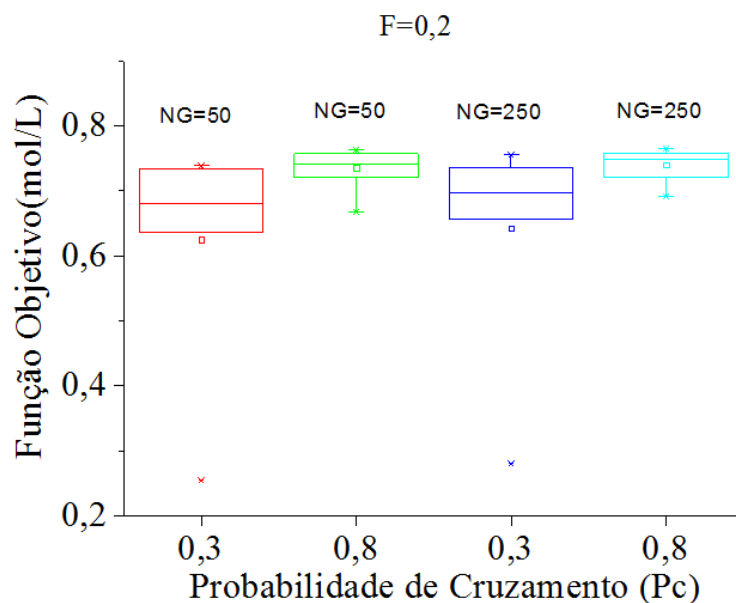


Figura 4.38: Função Objetivo x Probabilidade de Cruzamento para $F=0,2$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva

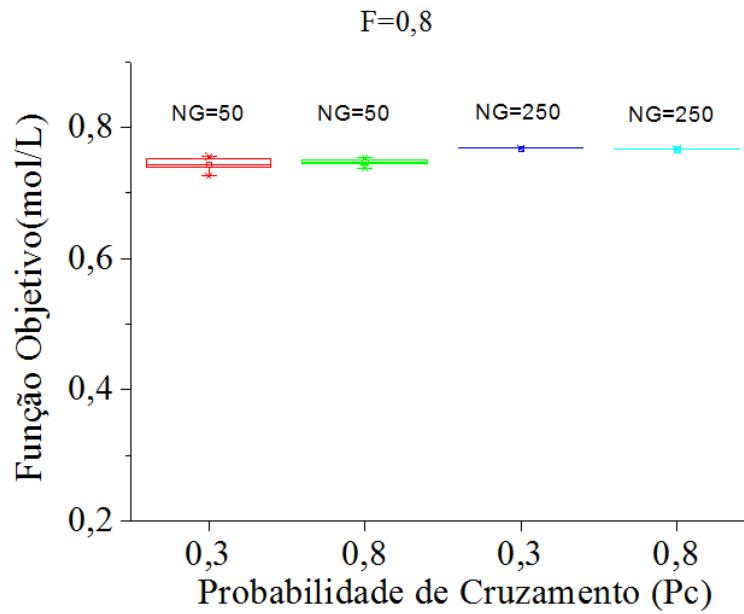


Figura 4.39: Função Objetivo x Probabilidade de Cruzamento para $F=0,8$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva

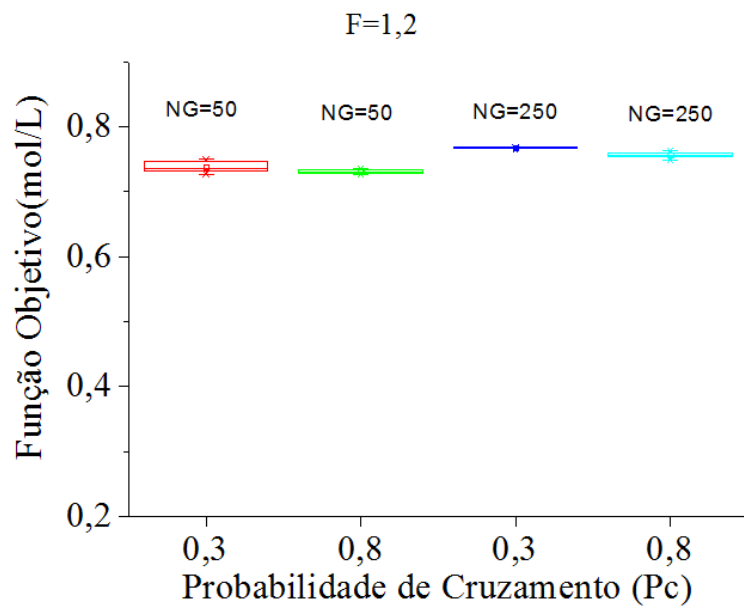


Figura 4.40: Função Objetivo x Probabilidade de Cruzamento para $F=1,2$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva

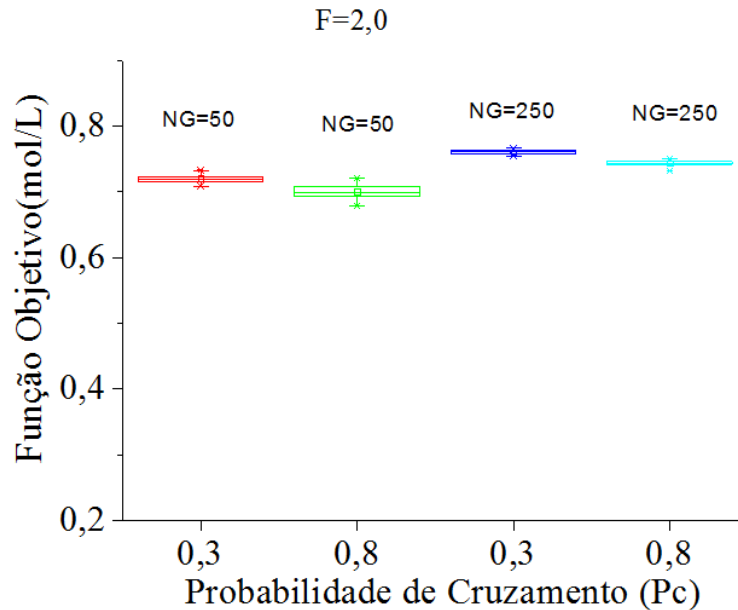


Figura 4.41: Função Objetivo x Probabilidade de Cruzamento para $F=2,0$; $NG=50$ e 250 para o problema do Reator Batelada com reação consecutiva

- Da Figura 4.38, observa-se que o valor médio de Função Objetivo não apresenta variação de um número de geração $NG=50$ para $NG=250$, considerando uma probabilidade de cruzamento $Pc=0,8$. Para $Pc=0,3$ a variação existe, no entanto é praticamente imperceptível. A dispersão dos valores de Função Objetivo foi praticamente a mesma para os diferentes números de gerações.
- Analisando a Figura 4.39, observa-se que o valor médio da Função Objetivo é um pouco melhor para uma probabilidade de cruzamento $Pc=0,8$ considerando um número de gerações $NG=50$. Considerando um número de gerações $NG=250$, o melhor valor médio da Função Objetivo é observado para $Pc=0,3$. Para ambas as probabilidades de cruzamento, o valor médio de Função Objetivo é melhor para um número de gerações $NG=250$. A dispersão dos valores de Função Objetivo foi maior para um número de gerações $NG=50$.
- Analisando a Figura 4.40, observa-se que o valor médio de Função Objetivo independentemente da probabilidade de cruzamento é melhor para um número de gerações $NG=250$. Considerando a dispersão dos valores de Função Objetivo, o fator de perturbação $F=1,2$ apresentou menor dispersão que $F=0,2$ independentemente do número de gerações e da probabilidade de cruzamento.
- Da Figura 4.41, novamente observa-se que para um número maior de gerações, o valor médio de Função Objetivo é maior.

- *Análise dos valores de Função Objetivo com o aumento do fator de perturbação (F) para diferentes Probabilidades de Cruzamento (P_c) e mesmo Número de Gerações (NG).*

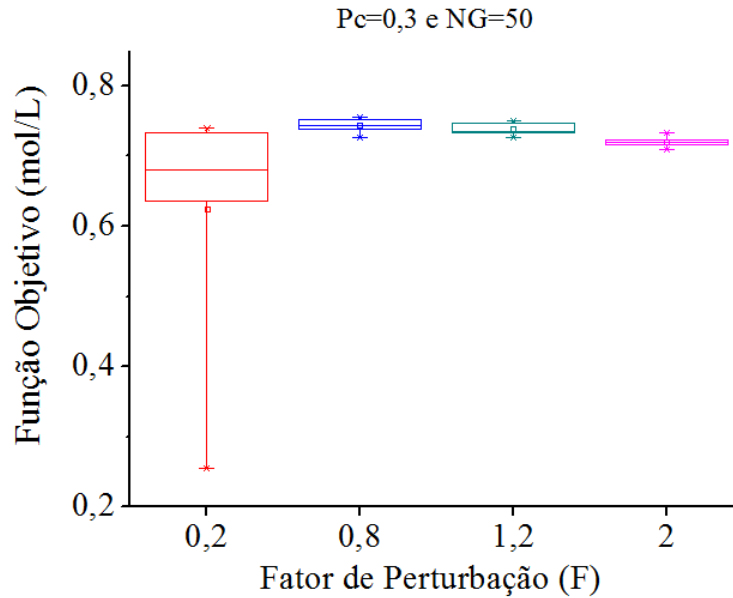


Figura 4.42: Função Objetivo x Fator de Perturbação para $P_c=0,3$ e $NG=50$ para o problema do Reator Batelada com reação consecutiva

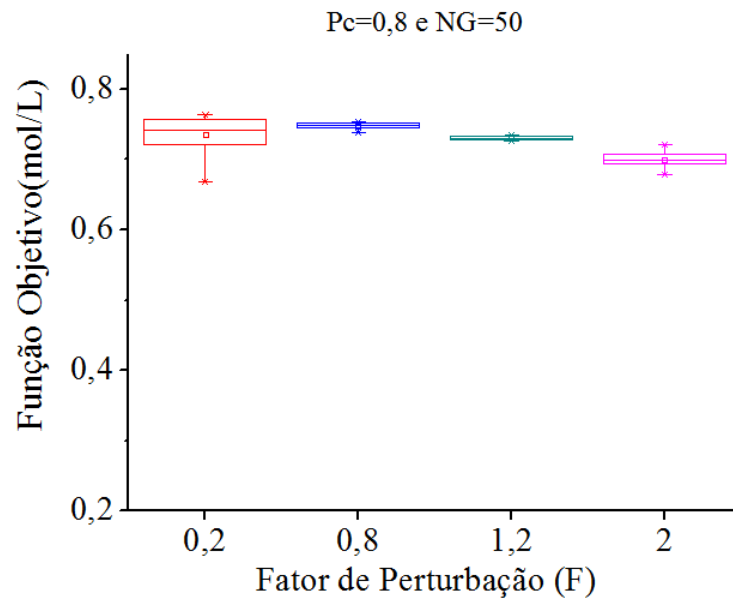


Figura 4.43: Função Objetivo x Fator de Perturbação para $P_c=0,8$ e $NG=50$ para o problema do Reator Batelada com reação consecutiva

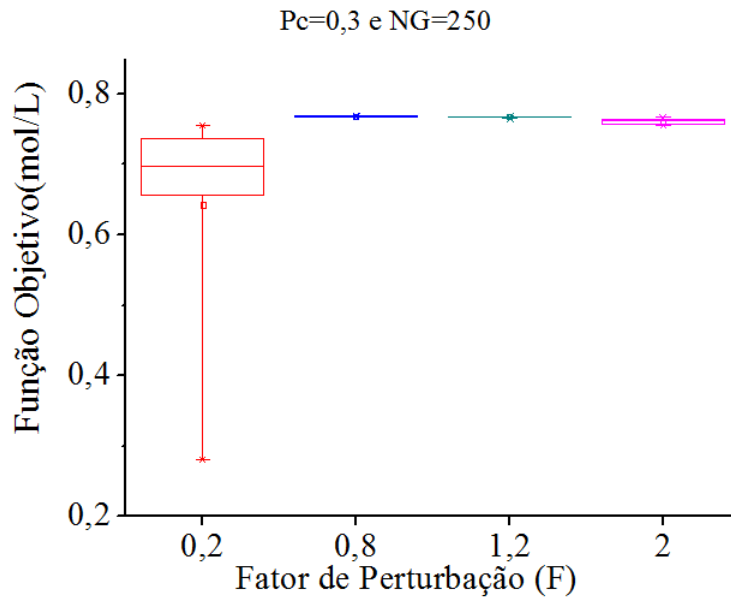


Figura 4.44: Função Objetivo x Fator de Perturbação para $P_c=0,3$ e $NG=250$ para o problema do Reator Batelada com reação consecutiva

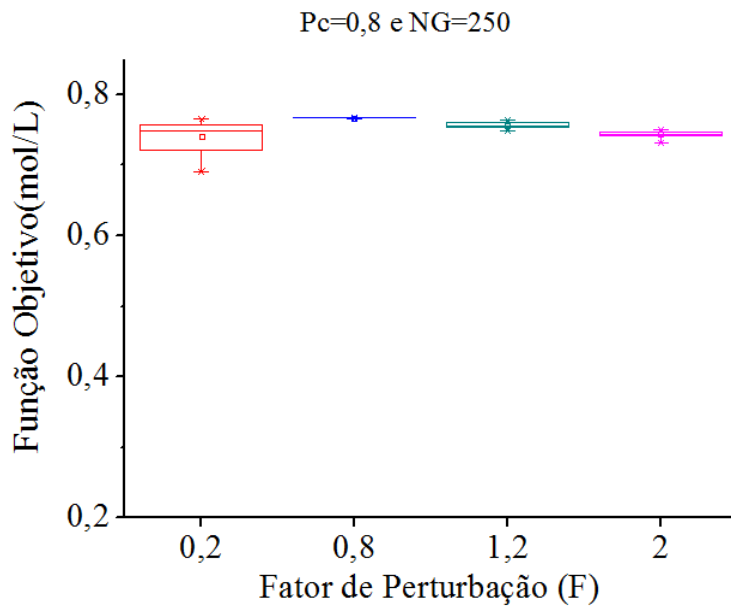


Figura 4.45: Função Objetivo x Fator de Perturbação para $P_c=0,8$ e $NG=250$ para o problema do Reator Batelada com reação consecutiva

- Através das Figuras 4.42 e 4.43 observa-se que tanto para $P_c=0,3$, quanto para $P_c=0,8$, o fator de perturbação $F=0,8$ apresentou o melhor valor médio de Função Objetivo. A dispersão dos valores de Função Objetivo foi em geral menor para $P_c=0,8$ exceto para $F=2,0$.

- Através das Figuras 4.44 e 4.45, novamente o fator de perturbação $F=0,8$ apresentou o melhor valor médio de Função Objetivo para $P_c=0,3$ e $P_c=0,8$. No entanto, o fator de perturbação $F=1,2$ apresentou um valor médio bem próximo a $F=0,8$ para uma taxa de cruzamento $CR=0,3$. A dispersão dos valores de Função Objetivo foi maior em $F=0,2$ e $F=2,0$ para uma taxa de cruzamento $P_c=0,3$ em relação a uma taxa de cruzamento $P_c=0,8$.
- Através da análise geral considerando todas as variações de parâmetros, observa-se que o conjunto $\{F=0,8, P_c=0,3, NG=250\}$ apresenta o melhor valor médio de Função Objetivo. Observa-se também que o mesmo conjunto apresenta uma menor dispersão dos valores de Função Objetivo em relação aos outros conjuntos de parâmetros.

Na Figura 4.46 é apresentado o perfil da variável de controle obtido por este trabalho. Os perfis das variáveis de estado são apresentados na Figura 4.47. A Tabela 4.20 apresenta o melhor valor de Função Objetivo encontrado por este trabalho comparando com valores reportados da literatura.

Tabela 4.20: Resultados obtidos por diferentes técnicas de solução: Reator batelada com reação consecutiva

FO (mol/L)	Método de solução	Referência	Nº de Avaliações de FO
0,768188	Evolução Diferencial	Este trabalho	25100
0,76837	Programação Dinâmica Iterativa	Luus e Okongwu (1999)	9000-45000

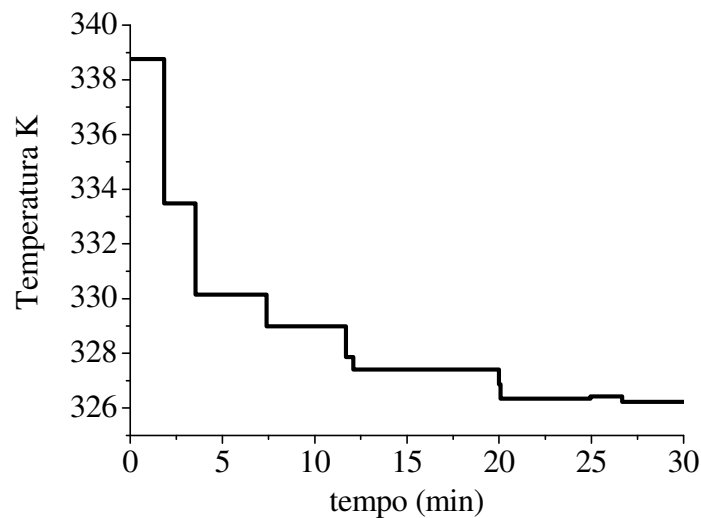


Figura 4.46: Perfil da variável de controle (T) para o problema do Reator Batelada com reação consecutiva

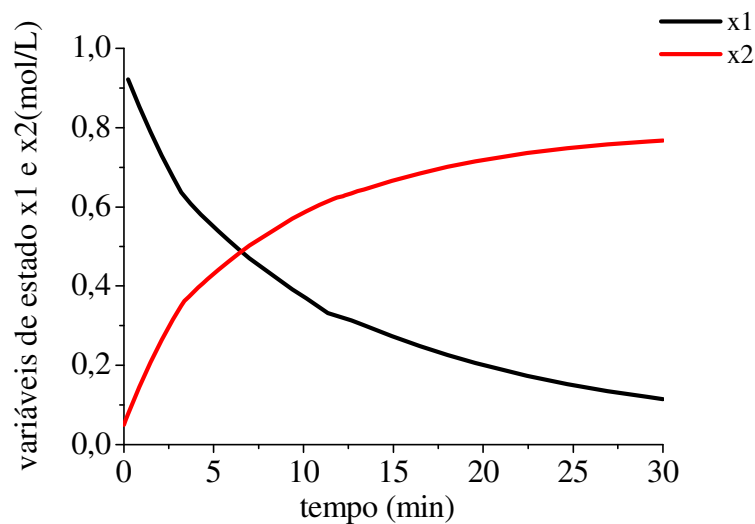


Figura 4.47: Perfis das variáveis de estado (x1, x2) obtidas por este trabalho para o problema do Reator Batelada com reação consecutiva

Neste capítulo foram apresentados alguns estudos de caso clássicos de diferentes índices diferenciais. Os resultados obtidos em cada estudo de caso são válidos apenas para o conjunto de sementes referido em cada problema. Os resultados obtidos foram comparados com resultados obtidos por outros autores utilizando diferentes técnicas de solução. Foi realizada uma análise de sensibilidade paramétrica avaliando a influência de alguns parâmetros do algoritmo de Evolução Diferencial nos valores de Função Objetivo obtidos.

Capítulo 5

CONCLUSÕES E SUGESTÕES

5.1 Conclusões

O algoritmo de Evolução Diferencial se mostrou eficiente na solução de problemas de otimização dinâmica clássicos, largamente reportados na literatura para a avaliação de métodos numéricos de solução. De fácil implementação, o algoritmo requer a definição de um número reduzido de parâmetros e evita a solução de problemas de valor no contorno oriundos da abordagem indireta, manipulações algébricas e diferenciações sucessivas em problemas de índice superior e a aplicação de métodos diretos clássicos, sequenciais ou simultâneos.

No primeiro estudo de caso, reator batelada com reação consecutiva, foi observado que os geradores randômicos (sementes) tiveram pouca influência nos resultados dos valores de Função Objetivo obtidos, principalmente considerando 05(cinco) e 10(dez) elementos de controle. O melhor valor obtido por este trabalho considerando 20 (vinte) elementos de controle foi 0,05% inferior ao valor obtido por Dadebo e Mcauley (1994) considerando 80 elementos. O perfil da variável de controle obtido foi compatível com os perfis obtidos por diferentes autores utilizando diferentes técnicas, tais como, Método Direto Simultâneo e Colônia de Formigas.

Para os dois estudos de caso singulares, problemas de Jacobson et al. (1970), os resultados diferiram do melhor resultado apresentado na literatura apenas na quarta casa decimal para o primeiro problema (Luus, 2001) e na terceira casa decimal para o segundo problema (Luus, 1995). Em ambos os casos, o melhor resultado da literatura foi obtido com um número de avaliações de Função Objetivo de cerca de quatro vezes superior ao número de avaliações necessárias aplicando Evolução Diferencial.

A técnica de Funções Penalidade Estáticas se mostrou eficaz na solução dos problemas com restrições de fim e de trajetória nas variáveis de estado. Para o caso 4 (problema do carro), a convergência para o tempo mínimo aconteceu após 500 gerações

em que todas as sementes apresentaram o mesmo valor de Função Objetivo. Tanto o caso 5 (produção de etanol) quanto o caso 6 (produção de penicilina) apresentaram em todas as gerações e todas as sementes os melhores valores de Função Objetivo sem nenhuma violação pelas restrições de fim e de trajetória. No caso 5 (produção de etanol), os valores da Função Objetivo foram influenciados pelo número de gerações até $N_{\text{gerações}}=200$. Para o caso 6 (produção de penicilina), o aumento do número de gerações influenciou os valores de Função Objetivo até $N_{\text{gerações}}=300$. De uma maneira geral para os casos 5 (produção de etanol) e 6 (produção de penicilina), o melhor valor de Função Objetivo obtido por Evolução Diferencial foi inferior ao melhor valor obtido por outros métodos de solução, exceto ao valor obtido por Upreti (2004) utilizando algoritmos genéticos. Para o caso 5 (produção de etanol), o melhor valor obtido por Evolução Diferencial foi 0,23% e 0,19% respectivamente inferiores aos obtidos por Luus e Henessy (1999) e Bojkov e Luus (1996), para um número de avaliações de Função Objetivo 44 vezes inferior ao número de avaliações em Luus e Henessy (1999). Considerando o caso 6 (produção de penicilina), o melhor valor de Função Objetivo obtido por Evolução Diferencial foi 0,08% e 0,026% inferior ao melhor valor obtido por Luus e Henessy (1999) e Luus (1992), sendo que o número de avaliações de Função Objetivo foi cerca de 5,5 vezes inferior ao número de avaliações realizadas em Luus e Henessy (1999). Upreti (2004) utilizando algoritmos genéticos obteve como melhor valor de Função Objetivo, em ambos problemas (produção de etanol e produção de penicilina), resultados inferiores aos obtidos por Evolução Diferencial com um número de avaliações de Função Objetivo bem superior.

Demonstrada a eficiência do algoritmo de Evolução Diferencial na solução de Problemas de Otimização Dinâmica complexos, foi realizada uma análise para identificar o melhor conjunto de parâmetros aplicáveis na solução de dois problemas de diferentes índices diferenciais. O problema de mistura de catalisadores (índice 3) apresentou cinco diferentes conjuntos de parâmetros com maior valor médio de Função Objetivo. Entretanto, apenas os parâmetros fator de perturbação $F=0,8$ e probabilidade de cruzamento $P_c=0,8$ apresentaram uma menor dispersão dos valores de Função Objetivo, independentemente do número de gerações. O problema do reator batelada com reação consecutiva (índice 1) apresentou $\{F=0,8, P_c=0,3, NG=250\}$ como melhor conjunto de parâmetros. Assim fica demonstrado que a obtenção de melhores resultados de Função Objetivo é dependente do conjunto de parâmetros do algoritmo de

Evolução Diferencial (para um determinado conjunto de sementes) utilizados para a solução de um determinado problema.

Desta forma, a aplicabilidade do algoritmo de Evolução Diferencial na solução de Problemas de Otimização Dinâmica de diferentes tipos fica demonstrada, tanto em relação à eficiência computacional quanto à capacidade de obtenção de resultados compatíveis aos resultados de Função Objetivo, quando comparados a resultados apresentados na literatura. Além disto, apesar de restrita a dois estudos de casos, mas que apresentam dificuldades de solução intrínsecas decorrentes do índice diferencial superior, a análise de sensibilidade paramétrica demonstrou a robustez do método uma vez que a sensibilidade paramétrica não é significativa. Este fato, aliado à facilidade de aplicação e implementação do Algoritmo de Evolução Diferencial, tornam seu uso ampliado para a solução de demais problemas extremamente promissor.

5.2. Sugestões para trabalhos futuros

- Aplicação da Evolução Diferencial na solução de problemas de otimização dinâmica com variáveis de controle múltiplas
- Desenvolvimento de uma estratégia dinâmica e em linha de atualização dos parâmetros do algoritmo de Evolução Diferencial, que permita a diminuição do esforço computacional e a ampliação da sua aplicabilidade a problemas de maior dimensão.
- Estudo da influência sobre a Função Objetivo da utilização de diferentes técnicas de função penalidade na manipulação de restrições

REFERÊNCIAS BIBLIOGRÁFICAS

- ABO-HAMMOUR, Z. S.; ASASFEH A.G.; AL-SMADI A. M. A novel continuous genetic algorithm for the solution of optimal control problems. *Optimal Control Applications and Methods*, v.32, p. 414-432, 2011.
- ANGIRA, R.; BABU, B. V. Optimization of process synthesis and design problems: A modified differential evolution approach. *Chemical Engineering Science*, v. 61, p. 4707–4721, 2006.
- ANGIRA, R.; SANTOSH, A. Optimization of dynamic systems: A trigonometric differential evolution approach. *Computers and Chemical Engineering*, v.31, p. 1055–1063, 2007.
- ASCHER, U.; SPITERI, R. Collocation software for boundary-value differential algebraic equation. *SIAM J. Scient.Comput.*, v.15, p.938-952, 1994.
- AZHMYAKOV, V. A gradient-type algorithm for a class of optimal control processes governed by hybrid dynamical systems. *IMA Journal of Mathematical Control and Information*, v.28, 291–307, 2011.
- BABU, B. V.; ANGIRA, R. Modified differential evolution (MDE) for optimization of non-linear chemical processes. *Computers and Chemical Engineering*, v.30, p. 989–1002, 2006.
- BABU, B. V.; MUNAWAR, S. A. Differential evolution strategies for optimal design of shell-and-tube heat exchangers. *Chemical Engineering Science*, v.62, p. 3720 – 3739, 2007.
- BACHMANN, R. et al. On methods for reducing the index of differential-algebraic equations. *Computers & Chemical Engineering*, v.14 (11), p. 1271–1273, 1990.
- BASTOS, E. A. *Otimização de Seções Retangulares de Concreto armado Submetidas à Flexo-Compressão oblíqua utilizando Algoritmos Genéticos*. Dissertação (Mestrado). Universidade Federal do Rio de Janeiro, 2004.
- BELL, M. L.; SARGENT, R.W.H. Optimal Control of inequality constrained DAE systems. *Computers and Chemical Engineering*, v.24, p.2385-2404,2000.
- BHOWMIK, P. et. al. A New Differential Evolution with Improved Mutation Strategy. Congress Evolutionary Computation, 18-23 de julho de 2010, p.1-8.
- BIEGLER, L. T. Solution of Dynamic Optimization Problems by Successive Quadratic Programming and Orthogonal Collocation. *Computers and Chemical Engineering*, v.8, p. 243-248, 1984.

BIEGLER, L. T.; CERVANTES, A. M.; WÄTCHER, A. Advances in simultaneous strategies for dynamic process optimization. *Chemical Engineering Chemical*, v.57, p.575-593, 2002.

BIEGLER, L. T.; GROSSMANN, I. E. Retrospective on Optimization. *Computers and Chemical Engineering*, v.28, p. 1169-1192, 2004.

BIEGLER L. T.; An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering and Processing*, v. 46, p. 1043–1053, 2007.

BILOUS, O.; AMUNDSON N. R. Optimum Temperature gradients in tubular reactors II, Numerical Study. *Chemical Engineering Science*, v.5, p.115-126, 1956.

BINDER, T. et al. Dynamic Optimization using a wavelet based adaptive control vector strategy. *Computers and Chemical Engineering*, v. 24, p.1201-1207.2000.

BOJKOV, B.; LUUS, R. Optimal Control of Nonlinear Systems with unspecified final times. *Chemical Engineering Science*, v.51, n°6, p.905-919, 1996.

BORGES, P. C. S. *Otimização Dinâmica da Fermentação Alcoólica no Processo em Batelada Alimentada*. Dissertação (Mestrado), Universidade Federal de Uberlândia, 2008.

BRENAN, K. E.; CAMPBELL, S. L.; PETZOLD, L. R. "Numerical Solution of Initial Value Problems in Differential Algebraic Equations". *Classics in Applied Mathematics*, SIAM Philadelphia, 1996.

BRYSON, A. E.; HO, Y. C. *Applied Optimal Control*. Hemisphere Publishing Washington ,1975.

BULIRSCH, R. et al. Combining Direct and Indirect Methods in Optimal Control: Range Maximization of a Hange Glider. *International Series of Numerical Mathematics*, v.111, 1993.

CANTO, E. B. et al. Restricted second order information for the solution of optimal control problems using control vector parameterization. *Journal of Process Control*, v.12, p. 243–255, 2002.

CERVANTES, A.; BIEGLER, L. T. Large-Scale DAE Optimization Using a Simultaneous NLP Formulation. *American Institute of Chemical Engineers Journal*, v.44, n° 5, 1998.

COELLO, C. A. C. Use of a self-adaptive penalty approach for engineering optimization problems. *Computers in Industry*, v.41, p. 113–127, 2000.

COELLO, C. A. C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer Methods Applied Mechanics Engineering*, v. 191, p. 1245–1287, 2002.

CONWAY, B. A. A Survey of Methods Available for the Numerical Optimization of Continuous Dynamic Systems. *J. Optim. Theory Appl.*, v.152, p.271-306, 2012.

CORRÊA DO QUINTO T. *Abordagem Algébrico-Diferencial da Otimização Dinâmica de Processos com Índice Flutuante*. Dissertação (Mestrado), Universidade Federal do Rio de Janeiro, 2010.

CUTHRELL, J. E.; BIEGLER, L. T. On the optimization of differential–algebraic process systems, *AIChE Journal*, v.33, 1257–1270, 1987.

CUTHRELL, J. E.; BIEGLER, L. T. Simultaneous Optimization and Solution Methods for Batch Reactor Control Profiles. *Computers and Chemical Engineering*, v.13, p. 49-62, 1989.

DADEBO, S. A.; MCAULEY, K. B. Dynamic Optimization of Constrained Chemical Engineering Problems Using Dynamic Programming. *Computers and Chemical Engineering*, v. 19, n°5, p.513-525, 1995.

DADEBO, S. A.; MCAULEY, K. B. “On the Computation of Optimal Singular Controls”. in *Proc. IEEE Conf. on Control Applications*, p. 150-155, 1995.

DANG, C.; LI, M. A floating-point genetic algorithm for solving the unit commitment problem. *Applied Mathematics and Computation*, v. 188, p. 895–911, 2007.

DEB, K.; AGRAWAL R. B. Simulated binary crossover for continuous search space, *Complex Systems* 9, p. 115–148, 1995.

DEB, K. “*Multi-objective Optimization using Evolutionary Algorithms*”. First edition. New York, John Wiley and Sons, 2001.

DEEP, K.; THAKUR M. A new crossover operator for real coded genetic algorithms, *Applied Mathematics and Computation*. v.188, p.895-911, 2007.

DORIGO, M.; MANIEZZO, V.; COLORNI, A. The Ant System: Optimization by a Colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics–Part B*, v.26, No.1, p.1-13, 1996.

EFFATI, S.; ROOHPARVAR H. Iterative dynamic programming for solving linear and nonlinear differential equations. *Applied Mathematics and Computation*, v. 175, p. 247–257, 2006.

ENGELBRECHT, A. P.; *Computational Intelligence: An Introduction*. England, Second Edition, 2007.

ENGLAND, R.; GÓMEZ, S.; LAMOUR, R. Expressing optimal control problems as differential algebraic equations. *Computers and Chemical Engineering*, v. 29, p. 1720–1730, 2005.

FABER, R.; JOCKENHÖVEL, T.; TSATSARONIS, G. Dynamic optimization with simulated annealing. *Computers and Chemical Engineering*, v.29 ,p. 273–290, 2005.

- FAN, H. Y.; LAMPINEN, J.A Trigonometric Mutation Operation to Differential Evolution. *Journal of Global Optimization*, v.27, p.105–129, 2003.
- FEEHERRY, W. F. *Dynamic optimization with path constraints*. Tese (Doutorado) – Massachusetts Institute of Technology, 1998.
- FEEHERRY, W. F.; BARTON P. I. Dynamic Optimization with Equality Path Constraints. *Industrial & Chemical Engineering Chemistry Research*, v.38, n° 6, p.2350–2363,1999.
- FERRARI, A.; GUTIERREZ, S.; BISCAIA JR., E. C. Development of an optimal operation strategy in a sequential batch reactor (SBR) through mixed-integer particle swarm dynamic optimization (PSO). *Computers and Chemical Engineering*, v. 34, p.1994–1998, 2010.
- FIKAR, M. et al. Control Vector Parametrisation versus Iterative Dynamic Programming in Dynamic Optimisation of a Distillation Column. *Computers chem. Engng* v. 22,p. S625-S628, 1998.
- FU, P. C.; BARFORD, J. P. Non-singular optimal control for fed-batch fermentation processes with a differential-algebraic system model. *J. Proc. Cont.*, v.3, n° 4, 1993.
- GOMES, E. O. *Abordagem algébrico-diferencial na solução de problemas de controle ótimo*. Dissertação(Mestrado)-Universidade Federal de Uberlândia,2000.
- GOUDOS, S.K. et al. A comparative study of common and self-adaptive differential evolution strategies on numerical benchmark problems. *Procedia Computer Science* ,v.3,p. 83–88, 2011.
- HA, KS.; RHEE, HK. Optimal Reaction Conditions for the Minimization of Energy Consumption and By-Product Formation in a Poly(ethylene terephthalate). *Process, Journal of Applied Polymer Science*, v. 86, p. 993–1008, 2002.
- HADIYANTO, H. et al. Control vector parameterization with sensitivity based refinement applied to baking optimization. *Food and Bioproducts Processing*, v.86, p. 130–141, 2008.
- HAIRER, E.; WANNER G. Stiff differential equations solved by radau methods. *Journal Computation and Applied Mathematics*, v.111, p.93-111,1999.
- HANKE, M.; LI PU; Simulated annealing for the optimization of batch distillation Processes. *Computers and Chemical Engineering*, v. 24, p. 1–8, 2000.
- JACOBSON, D. H.; GERSHWIN, W. D.; LELE, M. M.“Computation of Optimal Singular Controls”, *IEEE Trans.On Auto.Control*, v. 15, p. 67-73, 1970.
- JOINES, J. A.; HOUCK, C. R. On the Use of Non-Stationary Penalty Functions to Solve Nonlinear Constrained Optimization Problems with GA's, *D. Fogel (Ed.), Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE Press, Orlando, FL*, p. 579–584, 1994.

- KAPADI, M.; GUDI, R. D. Optimal control of fed-batch fermentation involving multiple feeds using Differential Evolution. *Process Biochemistry*, v.39, p. 1709–1721, 2004.
- KAZARLIS, S.; PETRIDIS, D. Varying Fitness Functions in Genetic Algorithms : Studying the Rate of Increase of the Dynamic Penalty Terms, A.E. Eiben, T. Bäck, M. Schoenauer, H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature V – PPSN V*, Springer, Amsterdam, Netherlands, 1998.
- KIM, N.; CHA, S.; PENG, H.; Optimal Control of Hybrid Electric Vehicles Based on Pontryagin's Minimum Principle. *IEEE Transactions on Control Systems Technology*, v. 19, n°5, 2011.
- KIRKPATRICK, S.; GELATT, C. D. JR.; VECCHI, M. P. Optimization by Simulated Annealing. *Science*, v. 220, n° 4598, p.671-680, 1983.
- LAMNABHI-LAGARRIGUE, F.; "Singular optimal control problems: On the order of a singular arc", *Systems and control letters* 9, p. 173-182, North Holland.1987.
- LEE , M. H.; HAN, C.; CHANG, K. S. Hierarchical time-optimal control of a continuous copolymerization reactor during start-up or grade change operation using genetic algorithms. *Computers and Chemical Engineering*, v. 21, p.1037-1042, 1997.
- LI, W. C.; BIEGLER, L.T. Process Control Strategies for Constrained Nonlinear Systems. *Industrial & Engineering Chemistry Research*, v. 27, n° 8, 1988.
- LINDEN, R. *Algoritmos Genéticos- Uma Importante ferramenta da Inteligência Computacional*, Brasport Livros e Multimídia, Rio de Janeiro, 2006.
- LOBATO, F. S. *Abordagem Mista Para Problemas de Otimização Dinâmica*. Dissertação (Mestrado), Universidade Federal de Uberlândia, 2004.
- LOBATO, F. S. et al. "Solution of Multi-Objective Optimal Control Problems with Fluctuation Index Using Differential Evolution". 6th DINCON – Brazilian Conference on Dynamics, Control and Their Applications, São José do Rio Preto-Brasil, , 2007c.
- LOPEZ CRUZ, I. L.; VAN WILLIGENBURG, L. G.; VAN STRATEN, G. Efficient Differential Evolution algorithms for multimodal optimal control problems. *Applied Soft Computing* 3, p. 97–122, 2003.
- LUUS, R.; ROSEN, O. Application of Dynamic Programming to Final State Constrained Optimal Control Problems. *Industrial & Engineering Chemistry Research*, v.30, p.1525-1530, 1991.
- LUUS, R. Optimization of Fed-Batch Fermentors by Iterative Dynamic Programming. *Biotechnology and Bioengineering*, v. 41, p. 599-602 , 1992.
- LUUS, R. Optimal control of batch reactors by iterative dynamic programming. *Journal of Process Control*, v 4, p. 218-226, 1994.

- LUUS, R. "Use of Iterative Programming for Optimal Singular Control Problems", *In Proc. of IASTED International Conference Control*, p. 286-289, 1995.
- LUUS, R.; OKONGWU, O. N. Towards practical optimal control of batch reactors. *Chemical Engineering Journal*, v. 75, p.1-9, 1999.
- LUUS, R.; HENESSY, D. Optimization of Fed-Batch Reactors by the Luus-Jaakola Optimization Procedure. *Ind. Eng. Chem. Res.*, v. 38, p. 1948-1955, 1999.
- LUUS, R. *Iterative Dynamic Programming*, Chapman and Hall/CRC, 2000.
- LUUS, R. "Use of Luus-Jaakola Optimization Procedure for Singular Optimal Control Problems". *Nonlinear Analysis*, v.47, p. 5647-5658, 2001.
- LUUS, R. Handling Inequality Constraints in Optimal Control by Problem Reformulation. *Industrial & Engineering Chemistry Research*, v. 48, p.9622-9630, 2009.
- MANDAL, K.K.; CHAKRABORTY N. Parameter study of differential evolution based optimal scheduling of hydrothermal systems. *Journal of Hydro-environment Research*, p.1-9, 2012.
- MAURER, H.; PESCH H. J. Direct optimization methods for solving a complex state-constrained optimal control problem in microeconomics. *Applied Mathematics and Computation*, v.204, p.568-579, 2008.
- MICHALEWICZ, Z, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, New York, 1992.
- MICHALEWIC, Z.; N.F. ATTIA, Evolutionary optimization of constrained problems, in: *Proceedings of the 3rd Annual Conference on Evolutionary Programming, World Scientific, Singapore*, p. 98-108, 1994.
- MODARES, H; SISTANI, M.B.N. Solving nonlinear optimal control problems using a hybrid IPSO-SQP algorithm. *Engineering Applications of Artificial Intelligence*, v.24, p.476-484, 2011.
- NANAKORN, P.; MEESOMKLIN K. An adaptive penalty function in genetic algorithms for structural design optimization. *Computers and Structures*, v. 79, p. 2527- 2539, 2001.
- OBERLE, H. J.; SOTHMANN, B. Numerical Computation of Optimal Feed Rates for a Fed-Batch Fermentation Model. *Journal of Optimization Theory and applications*: v. 100, n°. 1, p. 1-13, 1999.
- OH, S. H.; LUUS, R. Use of Orthogonal method in optimal control problems, *Int. J. Control*, v. 26, n° 5 , p. 657-673, 1977.

OLIVEIRA, G.T.S. *Estudo e Aplicações da Evolução Diferencial*, Dissertação (Mestrado), Universidade Federal de Uberlândia, 2006.

OLIVEIRA, F. C. G. et al. Three-Dimensional Vehicular Structure Design Combining Finite Element Modeling and Numerical Optimization". *19th COBEM – International Congress of Mechanical Engineering*, Brasília-Brasil, 2007.

ONO, I.; KOBAYASHI, S. A real-coded genetic algorithm for function optimization using unimodal normal distribution crossover, in: T. Back (Ed.), *Proceedings of the Seventh International Conference on Genetic Algorithms, Morgan Kaufmann, San Mateo, CA*, p. 246–253, 1997.

OUELLET, R.; BUI, R. T. A new tool for solving industrial continuous optimization problems, *Applied Mathematical Modelling*, v.17, Issue 6, p. 298–310, 1993.

PANTELIDES, C.C. et al. "The Mathematical Modeling of Transient Systems using Differential-Algebraic Equations". *Comput. chem. Engng*, v.12, n° 5, p.449-454, 1988.

PFEIFER A. A. *Controle Ótimo de Sistemas Algébrico- Diferenciais com Flutuação do Índice Diferencial*, Dissertação (Mestrado), Universidade Federal de Uberlândia, 2007.

POURTAKEDOUST, S.H.; JALALI, M. A. Thrust-Limited Optimal Three-dimensional Spacecraft Trajectories. *International Journal of Engineering*, v. 14, n°. 1, 2001.

RAHIMPOUR, M. R.; PARVASI, P.; SETOODEH P. Dynamic optimization of a novel radial-flow, spherical-bed methanol synthesis reactor in the presence of catalyst deactivation using Differential Evolution (DE) algorithm. *International Journal of Hydrogen Energy*, 2009.

RAJESH, J. et al. Dynamic optimization of chemical processes using ant colony framework. *Computers and Chemistry*, v.25, p. 583–595, 2001.

RAY, W. H. *Advanced Process Control*. McGraw-Hill, New York, 1989.

RENFRO, J. G.; MORSHEDI, A. M.; ASBJORNSEN, O. A. Simultaneous Optimization and Solution of Systems Described By Differential Algebraic Equations. *Computers and Chemical Engineering*, v.11, n°5, p.503-517, 1987.

ROHMAN, F. S.; AZIZ, N. Optimization of batch electrodialysis for hydrochloric acid recovery using orthogonal collocation method. *Desalination*, v. 275, p.37-49, 2011.

RUSNÁK, A. et al. Receding horizon iterative dynamic programming with discrete time models. *Computers and Chemical Engineering*, v. 25, p.161-167, 2001.

SANTOS, K. G.; LOBATO, F. S.; MALAGONI, R. A.; Resolução de Problemas de Controle Ótimo Usando Técnicas de Otimização Bio-Inspiradas na Natureza. *XIX Congresso Brasileiro de Engenharia Química*, 09-12 de Setembro de 2012, Búzios-RJ.

SARGENT, R.W.H. Optimal Control, *Journal of Computational and Applied Mathematics*, v. 124, p.361-371, 2000.

SARKAR, D.; MODAK, J. M. ANNSA: a hybrid artificial neural network/simulated annealing algorithm for optimal control problems. *Chemical Engineering Science*, v.58, p.3131-3142, 2003.

SARKAR, D.; MODAK, J. M. Optimization of fed-batch bioreactors using genetic algorithm: multiple control variables. *Computers and Chemical Engineering*, v. 28, p. 789-798, 2004.

SCHLEGEL, M.; STOCKMANN, K.; BINDER, T.; MARQUARDT W. Dynamic optimization using adaptive control vector parameterization, *Computers and Chemical Engineering*, v.29 , p.1731–1751, 2005.

SHOPOVA, E. G.; VANKLIEVA-BANCHEVA, N. G. BASIC—A genetic algorithm for engineering problems solution. *Computers and Chemical Engineering*, v.30, p.1293–1309, 2006.

SIMEONI, F. et al. Indirect Optimization of Satellite Deployment into a Highly Elliptic Orbit. *International Journal of Aerospace Engineering* , 2012.

SKOLPAP W. et al.; Fed-batch optimization of α -amylase and protease-producing *Bacillus subtilis* using genetic algorithm and particle swarm optimization. *Chemical Engineering Science* , v.63, p. 4090-4099, 2008.

SØRENSEN, K.; HOUBAK, N.; CONDRA, T. Solving differential–algebraic equation systems by means of index reduction methodology. *Simulation Modelling Practice and Theory*, v.14 , 224–236, 2006.

SRINIVASAN, B.; PALANKI, S.; BONVIN, D. Dynamic optimization of batch processes I. Characterization of the nominal solution. *Computers and Chemical Engineering* , v.27, p.1- 26, 2003.

STORN, R.; PRICE, K. Differential Evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012. *International Computer Science Institute*, Berkeley, 1995.

STORN, R.; PRICE, K. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, v. 11, p.341–359, 1997.

STORN, R.; PRICE, K. Home page of Differential Evolution for Continuous Function Optimization. Disponível em: <http://www1.icsi.berkeley.edu/~storn/code.html>. Acessado em 15 de junho de 2012.

STRYK, O.; BULIRSCH, R. Direct and indirect methods for trajectory optimization. *Annals of Operations Research*, v. 37, p.357-373, 1992.

STRYK, O. *User's Guide for DIRCOL- a Direct Collocation method for the numerical solution of optimal control problems*. [S.l.], 1999. Technische Universität Darmstadt, Fachgebiet Simulation und Systemoptimierung (SIM).

SUMAN, B. Study of simulated annealing based algorithms for multi-objective optimization of a constrained problem, *Computers and Chemical Engineering*, v.28, p. 1849–1871, 2004.

TARCA, L. A.; GRANDJEAN, B.P.A.; LARACHI, F. Embedding monotonicity and concavity in the training of neural networks by means of genetic algorithms .Application to multiphase flow. *Computers and Chemical Engineering*, v. 28, p. 1701-1713, 2004.

UPRETI, S. R. A new robust technique for optimal control of chemical engineering processes. *Computers and Chemical Engineering*, v.28 , p.1325–1336, 2004.

VANDERPLAATS, G. N. *Numerical Optimization Techniques for Engineering Design*. Colorado Springs, CO, Third Edition, 1999.

VASSILIADIS, V. S.; SARGENT, R.W. H.; PANTELIDES C.C. Solution of a Class of Multistage Dynamic Optimization Problems.1.Problems without Path Constraints. *Industrial & Engineering Chemistry Research*, v.33, p.2111-2122, 1994.

ZAHARIE, D. Influence of crossover on the behavior of Differential Evolution Algorithms. *Applied Soft Computing*, v.9, p. 1126–1138, 2009.

ZAVALA, V. M.; ANTONIO FLORES-TLACUAHUAC, EDUARDO VIVALDO-LIMA, Dynamic optimization of a semi-batch reactor for polyurethane production. *Chemical Engineering Science* ,v.60 ,p. 3061 – 3079, 2005.

ZHANG, B.; CHEN, D.; ZHAO, W. Iterative ant-colony algorithm and its application to dynamic optimization of chemical process, *Computers and Chemical Engineering*, v.29, p.2078-2086, 2005.

ZHANG, J.; XIE L.; WANG S. Particle Swarm for dynamic optimization of biochemical Processes, *16 th European Symposium Computer Aided Processes and 9th International Symposium on Processes Systems Enginnering*, 2006.

WESCHENFELDER, T. A.; CASTILHOS, F.; SCHEER, A. P.; Comparação do Desempenho de Algoritmos Estocásticos na Otimização Dinâmica do Processo de Mosturação de Cerveja, XIX, *Congresso Brasileiro de Engenharia Química*, 09-12 de Setembro de 2012, Búzios-RJ.

WRIGHT A. Genetic Algorithms for Real Parameter Optimization. In G. J. E. Rawlins , editor, *Foundations of Genetic Algorithms*, San Mateo, 1991, p. 205-220.