COMPUTAÇÃO PARALELA APLICADA À ESTIMAÇÃO BI-OBJETIVO DE PARÂMETROS DE MODELOS NARX POLINOMIAIS.

JIM JONES DA SILVEIRA MARCIANO

COMPUTAÇÃO PARALELA APLICADA À ESTIMAÇÃO BI-OBJETIVO DE PARÂMETROS DE MODELOS NARX POLINOMIAIS.

Dissertação apresentada ao Programa de Pós-Graduação em Programa de Pós Graduação em Engenharia Elétrica da Universidade Federal de São João del-Rei em parceria ampla com CEFET-MG como requisito parcial para a obtenção do grau de Mestre no Programa de Pós Graduação em Engenharia Elétrica.

Orientadora: Valceres Vieira Rocha e Silva
Co-Orientador: Márcio Falcão dos Santos Barroso

São João del-Rei

Janeiro de 2011

© 2011, Jim Jones da Silveira Marciano.
Todos os direitos reservados.

Marciano, Jim Jones da Silveira
Computação Paralela aplicada à Estimação Bi-objetivo de Parâmetros de Modelos NARX Polinomiais. / Jim Jones da Silveira Marciano. — São João del-Rei, 2011
xxiv, 76 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de São João del-Rei
Orientadora: Valceres Vieira Rocha e Silva
Co-Orientador: Márcio Falcão dos Santos Barroso

1. Estimador Bi-Objetivo. 2. Computação Paralela. 3. Pareto-Ótimo. 4. Otimização. 5. Identificação de Sistemas. I. Título.

CDU

Este trabalho é dedicado a minha mãe Rosângela Aparecida da Silveira (Linda) (in memorian).

Agradecimentos

Este trabalho é fruto da confiança e do empenho de várias pessoas que participaram diretamente e indiretamente do mesmo, e que sem as mesmas este trabalho não existiria. Portanto fica a minha eterna gratidão a vocês:

Minhas famílias (Silveiras, Marcianos e Basílios), pelo apoio e compreensão, em especial minha mãe Rosângela (*in memorian*) por ter me proporcionado momentos e conhecimentos que jamais esquecerei e minha irmã Layla pela minha ausência.

Minha namorada Indiane e toda sua família pela compreensão por minhas ausências e pelo apoio.

Meus amigos e companheiros de mestrado: Almira, Abdalla, Luciana, Paulinho (P1), Paulinho (P2), Silvan, Sidney e Wellington pelo companheirismo e longas conversas.

Agradeço ao PPGEL e a todos os professores do programa, em especial a minha orientadora Valceres pela confiança e dedicação, e ao meu co-orientador Márcio Barroso pela dedicação, confiança e disponibilidade.

A FAPEMIG pelo suporte financeiro sem o qual impossibilitaria o desenvolvimento deste trabalho.

A todos aqueles não foram citados me perdoem e deixo meu MUITO OBRIGADO a todos vocês.

"Há quem não veja nada além do fim,
e há aqueles que vêem no fim o início de uma nova jornada."
(Autor Desconhecido)

Resumo

Identificar um sistema consiste em modelar matematicamente o sistema a fim de que o mesmo possa ser reproduzido. Neste trabalho para a realização da identificação foi utilizado o estimador bi-objetivo. O estimador bi-objetivo, possui a capacidade de agregar informações auxiliares no processo de estimação. A estrutura do estimador bi-objetivo é composta de 4 partes: escolha de representação, detecção da estrutura, estimação de parâmetros e validação do modelo. Porém o custo computacional de uma solução gerada pelo estimador bi-objetivo é muito alto, o que desperta o interesse pela otimização deste custo. O objetivo deste trabalho é desenvolver meios para que o tempo de solução do estimador possa ser reduzido, e como resultado foram obtidos dois estimadores: bi-objetivo paralelo e bi-objetivo acelerado; ambos realizam a estimação com um tempo menor do que a forma convencional. Para ambos estimadores foram realizadas simulações com sistemas reais para verificação dos resultados dos mesmos.

Palavras-chave: Estimador Bi-Objetivo, Computação Paralela, Pareto-Ótimo, Otimização, Identificação de Sistemas.

Abstract

System identification consists on mathematically modeling a system in order to reproduce its most important features. In this work, a bi-objective estimator was used for process identification. The bi-objective estimator adds useful information to the estimation process. This estimation process implementation occurs in a four steps procedure: representation choice, framework detection, parameters estimation and model validation. The bi-objective estimation solution achievement involves high computational cost, what motivates this cost optimization interest. The objective of this work is to develop a technique that can reduce the estimation solution time. Two estimations technique came out as a result of this study: parallel bi-objective and accelerated bi-objective techniques; both estimate parameters in shorter time than the conventional way. For both, simulations using real systems were performed to verify the efficiency.

Keywords: Bi-Objective Estimation, Parallel Computation, Optimal-Pareto, Optimization, System Identification.

Lista de Figuras

2.1 Dominância do conjunto Pareto-ótimo. 13
3.1 Modelo do fluxo da informação para o paradigma sequencial. 20
3.2 Modelo do fluxo da informação para o paradigma paralelo. 21
3.3 Arquitetura SISD. 23
3.4 Arquitetura SIMD. 23
3.5 Arquitetura MISD. 24
3.6 Arquitetura MIMD. 24
4.1 Dados Não-Complementares - Conjunto de correlações com 12 pontos no Pareto. 29
4.2 Dados Não-Complementares - Conjunto de correlações com 100 pontos no Pareto. 30
4.3 Dados Não-Complementares - Conjunto de correlações com 1000 pontos no Pareto. 31
4.4 Dados Não-Complementares - Ampliação da Figura (4.3) na região do valor mínimo. 31
4.5 Dados Não-Complementares - Conjunto de correlações com 10000 pontos no Pareto. 32
4.6 Dados Não-Complementares - Ampliação da Figura (4.5) na região do valor mínimo. 32
4.7 Dados Não-Complementares - Conjunto de correlações com 100000 pontos no Pareto. 33
4.8 Dados Não-Complementares - Ampliação da Figura (4.7) na região do valor mínimo. 33
4.9 Dados Complementares - Conjunto de correlação para 12 pontos no Pareto-Ótimo. 34
4.10 Dados Complementares - Conjunto de correlação para 100 pontos no Pareto-Ótimo. 34
4.11 Dados Complementares - Conjunto de correlação para 1000 pontos no Pareto-Ótimo. 35

4.12 Dados Complementares - Mínimos local e global da Figura (4.11) 35
4.13 Dados Complementares - Mínimos local da Figura (4.11) 36
4.14 Dados Complementares - Conjunto de correlação para 10000 pontos no Pareto-Ótimo. 36
4.15 Dados Complementares - Mínimos local e global da Figura (4.14) 37
4.16 Dados Complementares - Mínimo local da Figura (4.14) 37
4.17 Dados Complementares - Conjunto de correlação para 100000 pontos no Pareto-Ótimo. 38
4.18 Dados Complementares - Mínimos local e global da Figura(4.17). 38
4.19 Tempos para estimador bi-objetivo de 1.000 à 19.000 pontos no Pareto. . . 40
4.20 Fluxograma do algoritmo do estimador bi-objetivo paralelo. 43
4.21 Primeira iteração bi-objetivo acelerado 44
4.22 Primeira escolha bi-objetivo acelerado 45
4.23 Segunda iteração bi-objetivo acelerado 45
4.24 Segunda escolha bi-objetivo acelerado 46
4.25 Terceira iteração bi-objetivo acelerado 47
4.26 Terceira escolha bi-objetivo acelerado 47
4.27 N-ésima iteração do bi-objetivo acelerado: 48
4.28 N-ésima escolha do bi-objetivo acelerado: 48
4.29 Fluxograma do algoritmo do estimador bi-objetivo acelerado. 49
5.1 Polimerizador - Validação estática do modelo escolhido pelo decisor 53
5.2 Polimerizador - Validação dinâmica do modelo escolhido pelo decisor para 4.000 pontos no Pareto-Ótimo 53
5.3 Polimerizador - Validação estática do modelo escolhido pelo decisor para 4.000 pontos no Pareto-Ótimo 54
5.4 Polimerizador - Validação estática do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado 54
5.5 Polimerizador - Validação dinamica do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado 55
5.6 Polimerizador - Comparação entre os tempos para solução do Estimador Bi-Objetivo Convencional e do Estimador Bi-Objetivo Paralelo 56
5.7 Polimerizador - Correlação dos estimadores para 1.000 a 19.000 pontos . . 57
5.8 Polimerizador - Validação dinâmica do modelo por Mínimos Quadrados Estendidos 59
5.9 Polimerizador - Validação estática do modelo por Mínimos Quadrados Estendidos 59
5.10 Estrutura de um conversor CC-CC Buck 60

5.11 Buck - Validação dinâmica do modelo escolhido pelo decisor para 19.000 pontos no Pareto-Ótimo 61
5.12 Buck - Validação estática do modelo escolhido pelo decisor para 19.000 pontos no Pareto-Ótimo 62
5.13 Buck - Validação estática do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado 62
5.14 Buck - Validação dinâmica do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado 63
5.15 Buck - Tempos para solução do Estimador Bi-Objetivo Convencional e do Estimador Bi-Objetivo Paralelo 64
5.16 Buck - Correlação para 1.000 a 19.000 pontos 65

Lista de Tabelas

4.1 Relação da correlação com a quantidade de pontos para dados complementares 28
5.1 Polimerizador - Tempos de solução para estimação Bi-Objetivo Convencional e Bi-Objetivo Paralelo 56
5.2 Buck - Tempos de solução para estimação Bi-Objetivo Convencional e Bi-Objetivo Paralelo 64

Sumário

Agradecimentos **ix**

Resumo **xiii**

Abstract **xv**

Lista de Figuras **xvii**

Lista de Tabelas **xxi**

1 Introdução **1**

1.1 Identificação de Sistemas 1

1.2 Estimação Bi-Objetivo 1

1.3 Computação Paralela 2

1.4 Objetivos 3

1.5 Organização do texto 3

2 Identificação de sistemas - Estimador Bi-Objetivo **5**

2.1 Introdução 5

2.2 Identificação de Sistemas 6

2.2.1 Escolha de Representação e de Estrutura 6

2.2.2 Agrupamento de Termos e Coeficientes de Agrupamentos 8

2.3 Estimação de Parâmetros 10

2.3.1 Estimador Bi-objetivo 11

2.3.2 Escolha dos Funcionais 12

2.3.3 Polarização no Estimador Bi-Objetivo 14

2.3.4 Etapa de escolha 15

2.4 Validação 16

3 Computação Paralela **19**

3.1 Introdução 19
3.2 Contextualização 21
3.3 Arquiteturas Paralelas 22
3.4 Modelo de Comunicação da Computação Paralela 25
3.4.1 Acesso a Dados Compartilhados 25
3.4.2 Troca de Mensagens 25

4 Estimadores Bi-Objetivo Acelerado e Paralelo. 27
4.1 Motivação 27
4.1.1 Dados Não-Complementares - Processo Térmico 29
4.1.2 Dados Complementares - Conversor Estático CC-Buck 33
4.1.3 Tempo de solução 39
4.2 Implementação do Estimador Bi-Objetivo Paralelo 40
4.2.1 Paralelização do Estimador Bi-Objetivo 40
4.3 Implementação do Estimador Bi-Objetivo Acelerado 44

5 Estudo de Casos 51
5.1 Aplicação em um Polimerizador – Dados Não-complementares 51
5.1.1 Resultados 52
5.1.2 Análise do Tempo de Estimação 54
5.1.3 Validação da Correlação dos Estimadores 57
5.1.4 Resultados Secundários 58
5.2 Conversor estático CC-CC Buck 60
5.2.1 Estrutura do Conversor CC-CC Buck 60
5.2.2 Resultados 61
5.2.3 Análise do Tempo de Estimação 63
5.2.4 Análise da Correlação dos Estimadores 64
5.3 Considerações Finais 66

6 Considerações Finais e Trabalhos Futuros 67
6.1 Estimador Bi-Objetivo Acelerado 67
6.1.1 Trabalhos Futuros 68
6.2 Estimador Bi-Objetivo Paralelo 68
6.2.1 Trabalhos Futuros 68

Referências Bibliográficas 69

Capítulo 1

Introdução

1.1 Identificação de Sistemas

Em diversas áreas da engenharia, principalmente na área de controle de processos, a modelagem matemática é uma etapa fundamental, uma vez que nem sempre é possível se aplicar diretamente ao sistema real as análises pertinentes à aplicação desejada (Weyer & Gábor, 2000; Miyano et al., 2000). Pode-se dizer então de modo muito geral e incerto que, achar um modelo é encontrar uma relação de causa e efeito entre variáveis de interesse, para substituir o sistema real e assim efetuar análises pertinentes para não só entender a dinâmica do sistema, mas também para a sintonia de controladores aplicados ao mesmo (Eykhoff, 1981). Além de sua estrutura, tais modelos possuem parâmetros, que podem ter significado físico, ou não, que devem ser estimados, para que tal estrutura se ajuste adequadamente aos dados disponíveis e assim, reproduza cooforme especificado a priori o sistema sob investigação. A estimação é realizada em virtude da minimização de funções de custo. Neste contexto, existem dois tipos de estimadores: Mono-Objetivo, leva-se em consideração apenas um objetivo, com ou sem restrições; e Multi-objetivo, quando existe mais de um objetivo no processo de estimação dos parâmetros (Aguirre, 1996).

1.2 Estimação Bi-Objetivo

A estimação Bi-Objetivo, que é objeto deste trabalho, consiste na geração de soluções para problemas de otimização em que não há uma única solução que minimize duas funções custo. Porém, o que se obtém como resposta do estimador é um conjunto de soluções, denominado conjunto Pareto-Ótimo[1]. Os Estimadores Bi-Objetivo são

[1]Para facilitar o texto, o Conjunto Pareto-ótimo será referenciado simplesmente como Pareto.

importantes por estarem na classe de estimadores não-polarizados, o que é essencial para aplicação em modelagem (Aguirre, 1996; Barroso, 2006; Nepomuceno, 2002). Por outro lado, como pode ser visto em Wuppalapati et al. (2008), independentemente do método adotado, a geração do Pareto-Ótimo é computacionalmente cara (Lesmere et al., 2007a; Ziviani, 2007). Além disso, ainda há a necessidade de se obter dentre os modelos candidatos do conjunto Pareto aquele que atende às necessidades de projeto ou a uma métrica pré-definida. Esta é a chamada etapa de decisão. Essa escolha pode ser feita por interação humana ou de forma automática, baseada em uma inteligência previamente determinada. Existem vários trabalhos sobre a Estimação Multi-Objetivo, dentre eles, pode-se destacar os trabalhos de Al-Fawzana & Haouarib (2005), Przybylski et al. (2006) e Mavrotas et al. (2009). Em Ebrahim & Razmi (2009) , Eusébio & Figueira (2009) são apresentados métodos de buscas para a escolha do "melhor" modelo entre os disponíveis no Pareto. Em Barroso et al. (2007) é descrito o processo de estimação Bi-Objetivo com critério de escolha da melhor solução pela correlação mínima.

1.3 Computação Paralela

A computação paralela consiste na capacidade de executar dois ou mais processos concomitantemente (Coulouris et al., 2005). A computação paralela tem sido utilizada como um artifício para a otimização de tempo de processamento, pelo fato de distribuir a carga de processamento para vários núcleos de processamento. Preocupados com a questão do custo computacional, trabalhos mais recentes propõem uma solução escrita na forma paralela e/ou distribuída (Lesmere et al., 2007a,b; Mezmaz & Melab, 2006; Abbasi et al., 2006; Siirola & Hauan, 2004; Wuppalapati et al., 2008; Negro et al., 2004; Rao, 2009).

Segundo Miller & Stout (1996) aplicação da mesma consiste em 3 etapas:

1. Memória compartilhada, em que um computador possui mais de um processador, esta técnica é utilizada neste trabalho é denominada computação Paralela;

2. Memória distribuída, em que cada unidade de processamento consiste em um computador em *cluster*;

3. Híbrido, esta técnica consiste utilização das duas técnicas anteriores, ou seja, um cluster com computadores com multiprocessadores.

Por atingir níveis de processamento equivalente a supercomputadores, a computação paralela tem sido objeto de trabalhos relacionados à otimização e processamento de grandes volumes de dados.

1.4 Objetivos

Pode-se verificar por meio das bases de indexação de periódicos que, do ano de 2000 em diante, as publicações na área de estimação bi-objetivo têm aumentado expressivamente, o que indica o aumento do interesse de tais estimadores e também da necessidade de se estudar maneiras de se otimizar o processo de geração do conjunto Pareto-Ótimo e da procura inteligente da solução considerada ótima, dentre as disponíveis. A questão de tempo de solução tem sido um empecilho para a utilização do mesmo.

Portanto, neste trabalho foram propostos dois métodos de otimização do estimador bi-objetivo:

- estimador bi-objetivo acelerado, que utiliza uma métrica baseada em exclusão de semi-espaços na estimação e busca da mínima correlação.
- estimador bi-objetivo paralelo, que realiza o processo de estimação paralelamente.

Os métodos propostos buscam um melhor tempo de solução do estimador bi-objetivo, para que este método possa ser utilizado para estimações de sistemas.

1.5 Organização do texto

Este trabalho está organizado da seguinte forma: No Capítulo 2 será descrito o Estimador Bi-Objetivo e suas funções matemáticas; no Capítulo 3 é realizada uma revisão bibliográfica da estrutura da computação paralela; No Capítulo 4 é apresentada a motivação que deu origem aos estudos apresentados neste trabalhos, juntamente com os métodos propostos para a otimização do estimador bi-objetivo: estimador bi-objetivo acelerado/paralelo. No Capítulo 5 são apresentados os resultados de simulações de identificação realizadas com dois sistemas reais: conversor estático CC-CC Buck e o reator polimerizador, a fim de validar os métodos propostos. No Capítulo 6 são apresentadas as considerações finais deste trabalho.

Capítulo 2

Identificação de sistemas - Estimador Bi-Objetivo

2.1 Introdução

Um estimador Bi-Objetivo permite a utilização de dois tipos de informações para compor sua estrutura. Porém, o que se obtém como solução do estimador é um conjunto de soluções, denominado conjunto Pareto-Ótimo. Portanto, a existência do Pareto é devida à existência de funcionais que são conflitantes, como por exemplo, as características estáticas e dinâmicas de sistemas não-lineares. Quantificar essas relações na forma de funções custo, pode ser uma maneira de agregar, na estimação dos parâmetros de representações não-lineares, características que são complementares e que poderão tornar os modelos mais representativos. A possibilidade da utilização de dois tipos de informações para compor sua estrutura, torna-o uma ferramenta diferenciada, pois nem sempre é possível obter um conjunto de dados com uma quantidade suficiente de informações das características do sistema. Neste caso, o problema bi-objetivo irá, na verdade, se tornar um estimador de parâmetros bi-objetivo, e passa a fazer parte da categoria dos estimadores não polarizados (Aguirre, 1996; Barroso, 2006; Nepomuceno, 2002). Portanto, pode-se observar que uma característica como polarização não deve ser o único fator determinante para escolha do estimador. Deve ser considerado também a capacidade de agregar informações que não estão disponíveis nos dados medidos utilizados (Barroso, 2006). Para ressaltar esta importância, Stephanopoulos & Ng (2000) diz que em aplicações industriais, deseja-se um modelo que seja capaz de recuperar não só as características dinâmicas, mas também as características estáticas.

Por outro lado, como pode ser visto em Wuppalapati et al. (2008), independentemente do método adotado, a geração do Pareto-Ótimo é computacionalmente cara (Lesmere et al., 2007a; Ziviani, 2007). Além disso, ainda há a necessidade de se obter

dentre os modelos candidatos do conjunto Pareto, aquele que atende às necessidades de projeto, ou a uma métrica pré-definida. Esta é a chamada etapa de decisão. Essa escolha pode ser feita por interação humana ou de forma automática, baseada em uma inteligência previamente determinada.

2.2 Identificação de Sistemas

O problema de identificação de sistemas pode ser dividido nas cinco etapas principais listadas a seguir (Ljung, 1987):

- Obtenção de dados de experimentação do sistema que se deseja modelar;
- Aplicação de testes aos dados obtidos para detecção de não-linearidades;
- Escolha da estrutura que será utilizada para representar o sistema;
- Estimação dos parâmetros do modelo,
- Validação do modelo obtido.

Essas etapas são comuns tanto para sistemas lineares quanto para sistemas não-lineares. Neste texto serão destacadas algum aspectos da identificação que são necessárias para o entendimento das técnicas desenvolvidas (Aguirre, 1996).

2.2.1 Escolha de Representação e de Estrutura

Na modelagem de sistemas não-lineares um dos passos importantes é a escolha dos modelos que irão representar o sistema em estudo, uma vez que existe uma grande diversidade de não-linearidades distintas. O modelo escolhido deve ser suficientemente rico para poder representar as não-linearidades mas, ao mesmo tempo, simples.

Neste trabalho, foram utilizados os modelos não-lineares auto-regressivos com entrada exógena NARX (*Non-linear Auto-Regressive with eXogenous inputs*), que são estruturas paramétricas do tipo entrada/saída (I/O), capazes de representar uma grande variedade de sistemas não-lineares (Billings & Chen, 1989).

Um modelo NARX com período de amostragem normalizado pode ser representado como (Leontaritis & Billings, 85 a,b; Billings & Chen, 1989):

$$\begin{aligned} y(k) &= F^l[y(k-1), y(k-2), ..., y(k-n_y), u(k-d-1), ... \\ &\quad ..., u(k-d-n_u)] + e(k), \end{aligned} \tag{2.1}$$

sendo $k = 1, ..., N$. F^l é uma função não-linear qualquer, $y(k)$ e $u(k)$ são, respectivamente, saída e entrada do sistema, que têm seus atrasos representados por n_y e n_u, respectivamente. d representa o tempo de retardo do sistema e $e(k)$ representa qualquer incerteza ou ruído. No contexto de regressão, os termos $y(k-i)$, com $(i = 1, ..., n_y)$ e $u(k-d-j)$, com $(j = 1, ..., n_u)$ são chamados de regressores de processo do modelo.

Neste trabalho F^l será restringida a uma função polinomial com grau de não-linearidade l. O modelo assim estabelecido pode ser representado, tendo em vista o problema de regressão da seguinte maneira:

$$\begin{aligned} y(k) \quad = \quad & \theta_0 + \textstyle\sum_{i_1}^{n} \theta_{i_1} x_{i_1} + \sum_{i_1}^{n} \sum_{i_2}^{n} \theta_{i_1} \theta_{i_2} x_{i_1}(k) x_{i_2}(k) + ... + \\ & \textstyle\sum_{i_1}^{n} ... \sum_{i_1=i_{l-1}}^{n} \theta_{i_1 ... i_l} x(k) ... + x_{i_l}(t) + e(t), \end{aligned} \tag{2.2}$$

sendo

$x_1(k) = y(k-1), x_2(k) = y(k-2), ..., x_{n_y+1}) = u(k-d-1), ..., x_n(k) = u(k-d-n_e)$. $n = n_y + n_u$.

Os θ's são os parâmetros que deverão ser estimados para que a estrutura escolhida para o modelo possa se ajustar do ponto de vista de regressão à janela de dados utilizados na estimação. No contexto deste trabalho é esperado também que o modelo não só se ajuste aos dados, mas principalmente que ele possa reproduzir a dinâmica do sistema original.

Apesar de ser impossível definir a melhor representação em termos gerais, podem-se destacar algumas vantagens dos modelos polinomiais sobre outras representações. É possível obter modelos NARX polinomiais que ajustem aos dados de identificação com boa exatidão, contanto que não haja uma variação muito brusca, evitando uma taxa de variação muito elevada. Além disso o modelo NARX polinomial pode ser transformado em uma representação linear, fixando-se o ponto de operação do sistema; ou seja, obtendo-se uma linearização do modelo. Outra vantagem é a facilidade de se obter informações analíticas sobre a dinâmica e as características em estado estacionário do modelo (Jácome, 1996). As funções não-lineares polinomiais são lineares nos parâmetros, o que permite a utilização de algoritmos de estimação de parâmetros para sistemas lineares (Billings & Voon, 1983; Chen et al., 1989). Esses algoritmos de estimação são fáceis de implementar, convergem rapidamente e já foram estudados em um vasto número de trabalhos (Davis & Vinter, 1985; Korenberg et al., 1988; Chen et al., 1989; Aström & Witternmark, 1990).

2.2.2 Agrupamento de Termos e Coeficientes de Agrupamentos

O modelo NARX (Equação 2.1) definido na Seção 2.2.2 pode ser reescrito como:

$$y(k) = \sum_{m=0}^{l} \sum_{p=0}^{m} \sum_{n_1,n_m}^{n_y,n_u} C_{p,m-p} \prod_{P}^{i=1} y(k-n_i) \prod_{m}^{i=p+1} u(k-n_i), \tag{2.3}$$

sendo

$$\sum_{n_1,n_m}^{n_y,n_u} \equiv \sum_{n_l=1}^{n_y} ... \sum_{n_m}^{n_y}. \tag{2.4}$$

Os monômios da Equação(2.3) são agrupados de acordo com sua ordem $m(0 \leq m \leq l)$, sendo l o grau de não-linearidade do modelo. Cada termo de ordem m contém fatores multiplicativos em $y(k-i)$ e $(m-p)$ fatores multiplicativos em $u(k-j)$. Os parâmetros destes termos são representados por $C_{p,m-p}(n_l, ..., n_m)$, nos quais $(n_l, ..., n_m)$ indicam os atrasos de cada fator constituinte do monômio considerado.

O primeiro somatório da Equação(2.3) se refere aos monômios da Equação(2.1), separando-os de acordo com sua ordem. O segundo somatório se refere ao número de fatores em $y(k-i)$ no termo considerado. Dentro do conjunto de termos de ordem m, um termo qualquer pode ser acessado através do ajuste do valor de p adequado. O último somatório permite que seja feita a distinção entre os termos da Equação(2.1), através do ajuste dos atrasos de cada um dos fatores constituintes do termo.

Analisando o modelo em estado estacionário para entradas constantes tem-se

$$\begin{array}{rcl} \bar{y} = y(k-1) & = & y(k-2) = ... = y(k-n_y) \\ \bar{u} = u(k-1) & = & u(k-2) = ... = u(k-n_u) \end{array}, \tag{2.5}$$

aplicando a Equação(2.5) na Equação(2.3), chega-se a

$$y(k) = \sum_{n_l,n_m}^{n_y,n_u} c_{p,m-p}(n_l, ..., n_m) \sum_{m=0}^{l} \sum_{p=0}^{m} y(k-1)^p u(k-1)^{m-p}. \tag{2.6}$$

o conjunto da forma $y(k-i)^p u(k-j)^{m-p}$ é denominado *agrupamento de termos* (Aguirre & Billings, 2007). Esses serão representados por $\Omega_{y^p u^{m-p}}$. A constante

$$\sum_{n_l,n_m}^{n_y,n_u} c_{p,m-p}(n_l, ..., n_m)$$

é o coeficiente do agrupamento de termos $y(k-i)^p u(k-j)^{m-p}$ e será representado por

$\sum_{y^p u^{m-p}}$. Todos os termos pertencentes a um dado agrupamento de termos explicam o mesmo tipo de não-linearidade no modelo.

2.2.2.1 Pontos fixos em sistemas não autonômos

Um modelo NARX não autônomo analisado em estado estacionário para entrada constante pode ser escrito como segue:

$$\sum_{n_l,n_{ml}}^{n_y,n_u} C_{p,ml-p}(n_l,...,n_{ml}) \sum_{ml=0}^{l} \sum_{p=0}^{ml} y(k-1)^p u(k-1)^{ml-p}, \tag{2.7}$$

sendo que ml corresponde ao grau de não linearidade de cada termo e está na faixa $1 \leq ml \leq l$. Cada termo de grau ml pode conter um fator dos termos $y(k)$ de ordem p, um fator em $u(k)$ de ordem $(ml-p)$ e um coeficiente $C_{p,ml-p}(n_l,...,n_{ml})$.

Para melhor compreensão, a Equação(2.7) pode ser reescrita como:

$$\begin{array}{lllll} \sum_{y^l} y^l & + & \sum^{l}_{ml=l-1}{}^{[\sum_{u^{ml-(l-1)}y^{l-1}}\, u^{ml-(l-1)}]y^{l-1}} & + & ... \\ & + & \sum^{l}_{ml=p}{}^{[\sum_{u^{ml-p}y^p}\, u^{ml-p}]y^p} & + & ... \\ & + & \sum^{l}_{ml=1}{}^{[\sum_{u^{ml-1}y^p}\, u^{ml-1}-1]y} & + & ... \\ & + & \sum^{l}_{ml=1}{}^{\sum_{u^{ml}}\, u^{ml}+\sum_0} = 0 & & \end{array} \tag{2.8}$$

Pode-se notar que as localizações dos pontos fixos dependem dos valores de entrada (constante) do sistema, ou seja, essas variam com o patamar do sinal de excitação. Essas equações passam a ter a função de um mapeamento que leva $\bar{u}$ a um valor determinado para $\bar{y}$. Os valores gerados por estes mapeamentos caracterizam uma curva estática para o sistema. Então, dessa forma, a função estática de sistemas não autônomos pode ser estimada a partir de modelos dinâmicos. nesse caso os conceitos de agrupamentos de termos e seus coeficientes são úteis para representar o procedimento de forma compacta. Então, a Equação(2.7) descreve o comportamento do modelo quando em estado permanente; em outras palavras, descreve a característica estática do sistema.

Os coeficientes de agrupamento podem ser escritos em função dos parâmetros do modelo da seguinte forma:

$$\mathbf{R}\theta = \sum_{\mathbf{x}}, \tag{2.9}$$

sendo que a matriz $\mathbf{R}$ mapeia θ para $\sum_{\mathbf{x}}$. θ é o vetor dos parâmetros do modelo NARX polinomial e $\sum_x$ é o vetor dos coeficientes de termos do modelo.

Ainda pode-se escrever a função estática $\bar{y} = f(\bar{u}, \bar{y})$ (Equação 2.7), de forma

matricial da seguinte maneira:

$$\bar{y} = [\bar{y}^0\bar{u}^0 \ \bar{y}^1\bar{u}^0 \ ... \ \bar{y}^p\bar{u}^{m-p}](\mathbf{R}\theta). \tag{2.10}$$

Chamando a matriz $[\bar{y}^0\bar{u}^0 \ \bar{y}^1\bar{u}^0 \ ... \ \bar{y}^p\bar{u}^{m-p}]$ de (Q), tem-se que

$$\bar{y} = \mathbf{Q}(\mathbf{R}\theta). \tag{2.11}$$

A apresentação matricial apresentada pela Equação(2.11) será muito importante para desenvolvimento do trabalho, principalmente para a sua construção metodológica.

2.3 Estimação de Parâmetros

Uma vez escolhida a estrutura de um modelo, deve-se estimar seus parâmetros para que o modelo possa se aproximar do comportamento dinâmico do sistema original. Isto sempre é feito em modelos polinomiais.

Considere a estrutura polinomial descrita em forma de regressão:

$$y(k) = \sum_{i=l}^{n} p_i(k)\theta_i + e(k), \tag{2.12}$$

onde os regressores do modelo, $p_i(k)$, correspondem aos diferentes termos no polinômio e os θ_i's são os respectivos parâmetros.

Escrevendo a Equação (2.12) na forma de erro de predição, tem-se:

$$y(k) = \sum_{i=l}^{n} p_i(k)\hat{\theta}_i + \xi(k, \hat{\theta}), \tag{2.13}$$

sendo

$$\sum_{i=l}^{n} p_i(k)\hat{\theta}_i = \hat{y}(k, \hat{\theta}), \tag{2.14}$$

sendo que o simbolo $(\wedge)$ sobre as variáveis faz referência a valores estimados. O resíduo de identificação $\xi(k, \hat{\theta})$ é definido como:

$$\xi(k, \hat{\theta}) = y(k) - \hat{y}(k, \hat{\theta}), \tag{2.15}$$

O vetor de resíduos $\{\xi(k), k = 1, ..., N\}$ representa os erros de modelagem, ruído do sistema e/ou qualquer incerteza. A Equação (2.14) é denominada preditor de um

passo à frente e $\hat{y}(k,\hat{\theta})$ é a predição de um passo à frente de $y(k)$.

Os parâmetros são estimados de modo a minimizar um índice de desempenho estabelecido previamente. Considere a função custo (Aström & Witternmark, 1990):

$$J_N(\theta) = \frac{1}{N}\sum_{k=1}^{N} f(\cdot), \tag{2.16}$$

sendo $f(\cdot)$ uma função matemática genérica.

Os parâmetros estimados serão diferentes para cada $f(\cdot)$ considerado. Utilizando-se $f = \xi^2$, tem-se o chamado método de mínimos quadrados. Nesse caso o índice de desempenho (ou função de custo) passa a ser:

$$J_N(\theta) = \frac{1}{N}\sum_{k=1}^{N} \xi(k,\theta)^2, \tag{2.17}$$

que deverá ser minimizado para determinar o conjunto de parâmetros correspondente.

Representando a Equação(2.13) em notação matricial:

$$\mathbf{y} = \mathbf{\Psi}\hat{\theta} + \vec{\xi}, \tag{2.18}$$

sendo

$$\begin{aligned} \mathbf{y} &= [y(1)\ y(2)\ ...\ y(N)]^T \\ \vec{\xi} &= [\xi(1)\ \xi(2)\ ...\ \xi(N)]^T \end{aligned} \tag{2.19}$$

$$\mathbf{\Psi} = \begin{bmatrix} p_1(1) & p_2(1) & p_n(1) \\ p_1(2) & p_2(2) & p_n(2) \\ . & . & . \\ . & . & . \\ . & . & . \\ p_1(N) & p_2(N) & p_n(N) \end{bmatrix}, \tag{2.20}$$

$$\theta = [\theta_1\ \theta_2\ ...\ \theta_n]^T. \tag{2.21}$$

A matriz $\mathbf{\Psi}$ é denominada matriz de regressores do modelo e θ indica o vetor de parâmetros nominal.

2.3.1 Estimador Bi-objetivo

Sejam duas funções custo, que podem ser escritas como sendo lineares nos parâmetros,

$$J_1 = f_1(\hat{\vec{\theta}}) \tag{2.22}$$

e

$$J_2 = f_2(\hat{\vec{\theta}}), \tag{2.23}$$

sendo de modo geral escrita como

$$J_i = \mathbf{\Psi}\vec{\theta}, \tag{2.24}$$

em que J_i é o i-ésimo funcional, $\vec{\theta}$ é o vetor de parâmetros a serem estimados e $\mathbf{\Psi}$ é uma matriz com as medições e/ou informações a respeito das variáveis do problema.

2.3.1.1 Conjunto Pareto-Ótimo

Uma maneira de se escrever este problema bi-objetivo como um problema mono-objetivo equivalente é escrever os funcionais como uma combinação convexa, ou seja (Barroso, 2006; Chankong & Haimes, 1983), ou seja:

$$J_{\hat{\vec{\theta}}} = \lambda_1 J_1(\hat{\theta}) + \lambda_2 J_2(\hat{\vec{\theta}}) \tag{2.25}$$

sendo que $\lambda_1 + \lambda_2 = 1$. Este é o chamado problema P_λ. Variando os valores de λ da seguinte forma: $0 < \lambda_i < 1$, é possível obter todo o conjunto de soluções eficientes ou Pareto-Ótimas, que pode ser escrita matematicamente como:

$$\hat{\theta}^* \in \hat{\Theta}^* \Leftrightarrow \left\{ \nexists\ \hat{\theta} : \mathrm{J}(\hat{\theta}) \leq \mathrm{J}(\hat{\theta}^*) \quad e \quad \mathrm{J}(\hat{\theta}) \neq \mathrm{J}(\hat{\theta}^*) \right\} \tag{2.26}$$

ou seja, existe uma região no espaço de parâmetros em que as soluções não são ordenáveis. Esses pontos podem ser observados na Figura 2.1, nos pontos p_1 e p_2. Pode-se dizer então que não existe, simultaneamente, um valor para $\vec{\theta}$ que minimiza os dois funcionais. A diminuição do valor de um dos funcionais, leva a um aumento do outro, e vice-e-versa. No entanto, o ponto p_3, é dominado pela solução Pareto, ou seja, é capaz de melhorar simultâneamente ambos os funcionais.

2.3.2 Escolha dos Funcionais

De maneira geral, em identificação de sistemas, deseja-se obter as características dinâmicas e em estado estacionário de sistemas. No entanto, é comum que as estruturas dos modelos dinâmicos e estáticos de representação polinomiais não-lineares apresentem soluções conflitantes quanto ao ajuste aos dados dinâmicos e estáticos disponíveis

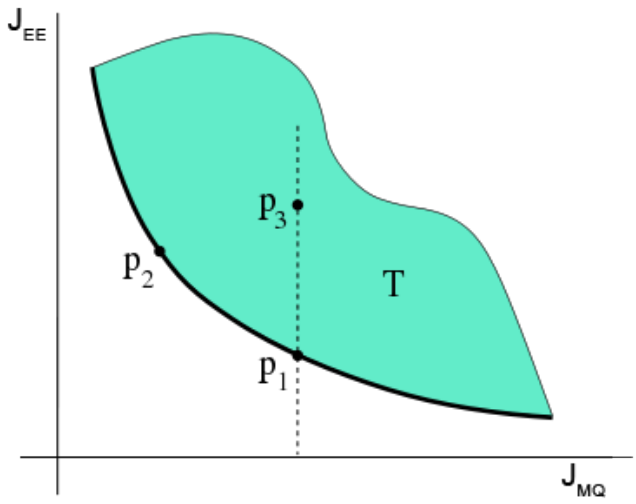

Figura 2.1: Dominância do conjunto Pareto-ótimo.

(Aguirre, 1996). Por isso, naturalmente, os funcionais escolhidos para representar este conflito são a minimização dos resíduos de identificação um passo à frente (dinâmica) e o ajuste de curva da estrutura estática aos dados em estado estacionário (estática), que são representados, respectivamente pelas Equações (2.27 e 2.28), a seguir:

$$J_{\mathrm{MQ}}(\hat{\vec{\theta}}) = \|\xi\|^2 = (\vec{y} - \mathbf{\Psi}\vec{\theta})^2 \tag{2.27}$$

e

$$J_{\mathrm{EE}}(\hat{\vec{\theta}}) = \|\zeta\|^2 = (\bar{\vec{y}} - \mathbf{QR}\vec{\theta})^2 \tag{2.28}$$

em que $\vec{y}$ é o vetor de dados de saída de identificação (medidos ou simulados), $\mathbf{\Psi}$ é a matriz de regressores do processo, $\bar{\vec{y}}$ são os dados estáticos de saída (medidos ou simulados), $\mathbf{Q}$ é a matriz de dados estáticos de entrada e saída (estrutura estática racional, escrita de forma polinomial) e $\mathbf{R}$ é uma matriz de zeros e uns que mapeiam os parâmetros da estrutura dinâmica $\vec{\theta}$ em seus respectivos coeficientes de agrupamentos de termos (Aguirre, 1996).

Uma solução analítica para o problema apresentado na Equação (2.25) é definida em Nepomuceno (2002) como:

$$\begin{aligned} \hat{\theta}^*(\lambda) \quad = \quad & \left[\lambda \Psi^T \Psi + (1-\lambda)(\mathbf{QR})^T \mathbf{QR}\right]^{-1} \ldots \\ & \left[\lambda \Psi^T \Psi + (1-\lambda)(\mathbf{QR})^T \bar{\vec{y}}\right] \end{aligned} \tag{2.29}$$

Esses funcionais foram escolhidos porque em Barroso (2006), foi demonstrado que

existe uma solução não-polarizada no Pareto, caso, a estrutura for coincidente com a do sistema a ser identificado, com estes funcionais.

2.3.3 Polarização no Estimador Bi-Objetivo

O estimador bi-objetivo é do tipo $Ay_1 + By_2$, e a solução é dada da seguinte maneira:

$$\hat{\theta} = Ay_1 + By_2 \tag{2.30}$$

Sendo a Equação(2.30) um estimador bi-objetivo, A e B matrizes que dependem da estrutura dos funcionais. y_1 será o vetor contendo os dados dinâmicos e que pode ser modelado de duas maneiras possíveis: modelos de erro na equação de regressão e modelos de erro na saída. O vetor y_2 contém dados de saída em estado estacionário. Este é modelado por um polinômio em que o erro associado será considerado insignificante.

2.3.3.1 Modelos de erro na equação de regressão

Os vetores y_n são obtidos pela seguinte representação polinomial:

$$y_n = G_n\theta + e_n. \tag{2.31}$$

Portanto, partindo da definição de polarização $b = E[\hat{\theta}] - \theta$, pode-se chegar na seguinte expressão:

$$b = E[AG_1 + BG_2 - I]\theta + E[Ae_1 + Be_2]. \tag{2.32}$$

As condições para não-polarização ($b = 0$) são as seguintes:

$$(AG_1 + BG_2) = I \tag{2.33}$$

e

$$E[Ae_1 + Be_2] = 0 \tag{2.34}$$

Segundo Barroso (2006) para que não haja polarização em modelos de erros na equação de regressão algumas hipóteses devem ser consideradas:

Hipótese 1: O erro $e_1(k)$ somado à equação de regressão do modelo dinâmico apresenta média zero, ou seja, $E[e_1] = 0$. É razoável dizer que $E[e_1] = 0$, uma vez que esse erro apresenta toda incerteza e/ou informação que não pode ser modelada.

Hipótese 2: O erro $e_2(k)$ associado aos dados estáticos, pode, sem perda de generalidade, ser considerado insignificante em relação ao erro dinâmico. De maneira diferente, $||\mathbf{e_2}|| << ||\mathbf{e_1}||$ e $||\mathbf{e_2}|| \approx 0$.

Hipótese 3: A matriz A e $\mathbf{e_1}$ são estatisticamente independentes. Portanto $E[A\mathbf{e_1}] = E[A]E[\mathbf{e_1}]$

2.3.3.2 Modelos de erro na saída

Para os modelos de erro na saída, o estudo apresentado em Barroso (2006) considera que os vetores de y_n são obtidos por meio da seguinte representação polinomial:

$$\begin{aligned} z &= G_n\theta \\ y_n &= z + \nu_n \end{aligned} \tag{2.35}$$

e a polarização é definida como:

$$b = E[AG_1 + BG_2 - I]\theta + E[A\nu_1 + B_2\nu_2] \tag{2.36}$$

As condições para não-polarização ($b = 0$) são as seguintes

$$(AG_1 + BG_2) = I \tag{2.37}$$

$$E[A\nu_1 + B\nu_2] = 0 \tag{2.38}$$

As condições para não-polarização ainda continuam as mesmas:

- O termo $(AG_1 + BG_2) = I$.
- O termo $E[A\nu_1 + B\nu_2] = 0$.

Hipótese 4: A matriz de erro $\nu_1(k)$ somado à saída do modelo dinâmico é ruído branco e tem média zero, ou seja $E[\nu_1] = 0$.

Hipótese 5: O erro $\nu_2(\mathbf{k})$ associado aos dados estáticos, pode, sem perda de generalidade, ser considerado insignificante em relação ao erro dinâmico. De maneira diferente $||\nu_2|| << ||\nu_1||$ e $||\nu_2|| \approx 0$.

2.3.4 Etapa de escolha

A escolha do modelo, entre os disponíveis no Pareto, é feita por meio da comparação entre as simulações livre dos modelos e os respectivos resíduos de simulação, conforme descrito em Barroso (2006).

Na simulação do modelo, infinitos passos a frente (ou simulação livre) $\hat{\vec{y}}$ é realizada pela Equação(2.39):

$$\hat{\vec{y}}(k) = \hat{\mathbf{\Psi}}(k-1)\,\hat{\vec{\theta}}, \tag{2.39}$$

Então, os resíduos de simulação livre podem ser escritos como:

$$\vec{\eta} = \vec{y}_{\mathrm{v}} - \hat{\vec{y}}, \tag{2.40}$$

em que $\vec{y}_{\mathrm{v}}$ é o vetor de dados dinâmicos de validação, que devem ser distintos dos usados para a etapa de identificação.

Seja ainda a correlação cruzada entre $\hat{\vec{y}}(k)$ e $\vec{\eta}$, definida na Equação (2.41):

$$J_{corr} = \frac{1}{N}\sum_{k=1}^{N} \eta(k)\hat{y}(k) \tag{2.41}$$

Para escolha do melhor modelo, é utilizado o critério da mínima correlação dos dados, representado matematicamente como:

$$\min_{\hat{\vec{\theta}} \in \hat{\mathbf{\Theta}}^*} J_{corr}\left(\hat{\vec{\theta}}\right). \tag{2.42}$$

Em Barroso (2006) é mostrado que a convergência da Equação 2.42 para zero leva os parâmetros a convergirem para o seus valores nominais, ou seja, a escolha é aquela que apresenta os parâmetros não polarizados.

2.4 Validação

A ausência de algum termo importante no modelo pode provocar polarização dos parâmetros e o aparecimento de dinâmicas espúrias na simulação do modelo. Por essa razão, faz-se necessária a utilização de um critério para determinar se o modelo responde às características que lhe são exigidas. No contexto deste trabalho, são considerados dois critérios:

1. predição infinitos passos à frente (simulação livre);
2. capacidade de representar o comportamento estático do sistema.

Outra maneira de verificar a validade dinâmica do modelo é o uso de funções de correlação. Essas funções podem ser aplicadas aos resíduos para a verificação de correlação linear e não-linear. Se os resíduos possuírem alguma correlação, o modelo será

considerado polarizado, uma vez que existem evidências de dinâmicas não modeladas contidas nos resíduos. Outros testes são as correlações cruzadas, lineares e não-lineares, entre os resíduos e os dados de entrada e saída do sistema (Box, 1976; Ljung, 1987; Billings & Voon, 1983, 1984).

Capítulo 3

Computação Paralela

Neste capítulo são apresentadas as bases teóricas fundamentais para o entendimento de técnicas de computação paralela que foi fundamental para o densenvolvimento desta dissertação.

3.1 Introdução

Desde o início da era computacional a velocidade de processamento tem crescido exponencialmente, e este crescimento é devido a evolução dos problemas a serem solucionados (NP-difícil, NP-completos entre outros), exigindo cada vez mais das máquinas utilizadas (Shonkwiler & Lefton, 2006). Cálculos complexos de engenharias, o intenso trabalho na área de genética necessitavam cada vez mais um grande poder de processamento para seus problemas. Portanto foi necessário buscar uma alternativa para suprir esta necessidade de processamento e uma solução encontrada foi a computação paralela.

A computação paralela é uma técnica na qual vários processos podem ser simulados (executados) concomitantemente. O interesse pela computação paralela é crescente devido à necessidade de solucionar problemas que exigem um maior poder de processamento e também uma redução no tempo de solução, o que somente seria encontrado em um supercomputador (Miller & Stout, 1996). Porém, o custo de um supercomputador é muito alto e também seu elevado porte físico não possibilita sua instalação em pequenos ambientes, o que inviabiliza em muitos casos a sua implantação. Há décadas a computação paralela vem sendo utilizada como alternativa para problemas que exigem grande processamento e uma redução no tempo de solução (Greenlaw et al., 1995).

Na busca pelo maior poder de processamento e também redução de tempo de execução de cada processo, foi implantada a idéia de computação paralela distribuída que utiliza uma rede de computadores como meio de comunicação de cada processo

(Coulouris et al., 2005). Com o advendo da evolução das redes, cada vez mais é possível encontrar a utilização da computação paralela local (implementada em máquinas que possuem mais de um núcleo de processamento) conjuntamente com a computação paralela distribuída, afim de obter um melhor poder de processamento e melhoria no tempo de solução.

A computação paralela além de agregar os modelos dos paradigmas sequenciais (Figura 3.1) de programação, a computação paralela agrega também os seguintes modelos (Abrantes, 2008):

- Comunicação entre tarefas: A comunicação entre os dados é devida à necessidade da troca de dados entre as tarefas;
- Concorrência: A concorrência acontece quando existe o acesso simultâneo por duas ou mais tarefas a uma determinada região da memória(dado);
- Escalabilidade: A escalabilidade é a eficácia do processo em aumentar o número de processadores de acordo com a necessidade e/ou disponibilidade/aquisição de novos recursos;
- Localidade: A localidade é a definição de quais dados são locais a uma tarefa.

A Figura (3.1) apresenta o fluxo da informação do paradigma sequencial, este que em comparação com o paradigma paralelo (Figura 3.2) apresenta pode ser considerado um fluxo simples de informação.

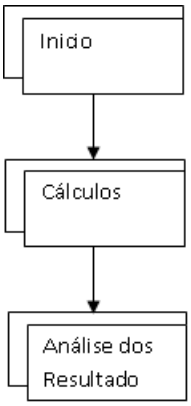

Figura 3.1: Modelo do fluxo da informação para o paradigma sequencial.

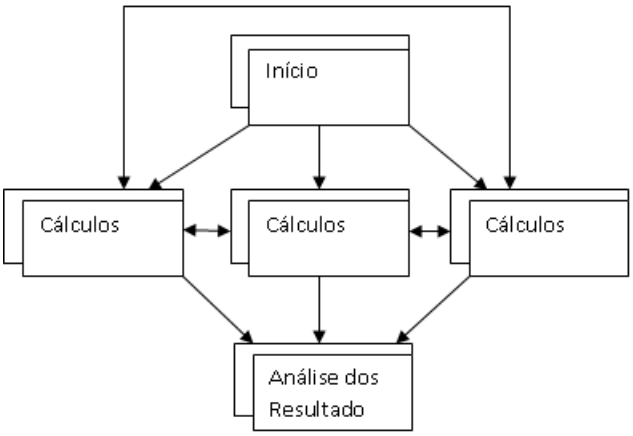

Figura 3.2: Modelo do fluxo da informação para o paradigma paralelo.

3.2 Contextualização

Uma vez que a computação paralela é uma técnica na qual, vários processos são executados concomitantemente isto gera uma otimização no tempo de solução. Devido a esta otimização de tempo, esta técnica tem sido aplicada em diversas áreas do saber. Existem vários trabalhos de computação paralela aplicadas na área de otimização, como exemplo em Migdalas et al. (2003) em que os autores discorrem sobre a aplicação do paralelismo em otimização não-linear. Considerando ainda paralelismo em otimização não-linear em Manke (2001), os autores apresentam necessidade do paralelismo na área aeroespacial devido à necessidade de computação de alto desempenho e fidelidade com os modelos de fluxo na análise e projeto de uma aeronave.

Gatu & Kontoghiorghes (2003) apresentam um estudo sobre a eficiência de algoritmos paralelos na computação de todos as possibilidades de modelos de regressão em um subconjunto, buscando um balanceamento de carga e nenhuma comunicação entre os processos; isto é possível pelo compartilhamento de memória entre eles. Brozolo & Robert (1989) propôs em seu trabalho uma solução paralela para o método do gradiente conjugado em vários processadores. Ainda dentro da área de otimização, a computação é amplamente utilizada para processamento de imagens, principalmente em tês dimensoes (3D) que exige um maior poder de processamento, como é apresentado por Kim & Im (2003). A computação paralela foi aplicada em otimização multi-objetivo para avaliações de ambientes dinâmicos (Cámara et al., 2009). Ainda sobre otimização multi-objetivo Figueira et al. (2010) utiliza computação paralela para melhorar o desempenho em problemas discretos e de natureza combinatória.

A computação paralela é também utilizada fora da aréa de otimização/computação. No estudo de Berthoz & Baker (1978) o processamento paralelo

é utilizado para o controle dos movimentos oculares. Em Guo et al. (2004) a computação paralela é aplicada na construção de espaços de soluções (conjuntos dominantes) por meio de fitas de DNA. Em Zhang et al. (2003) é apresentado um estudo sobre a modelagem de fluidos em zonas não saturadas. Na área de bio-computação/bio-engenharia e neurociência há uma crescente utilização da computação paralela, como pode ser observado pelos seguintes trabalhos Sorin et al. (2003), Andrianov (2004), Gröger & Zimmermann (2010), Bhandarkar et al. (1998), Araújo et al. (2010). Este fato dá-se pelo alto nível de complexidade e processamento necessários para solução de problemas genéticos, simulações neurais, analises de DNA e etc.

Portanto, como uma alternativa a supercomputadores, a computação paralela é aplicada a vários problemas complexos (Gustafson (1999)). Porém, nem sempre é possível e/ou se tem disponível a execução do paralelismo em um único computador. Preocupados com o desempenho dos *clusters* Kosar & Livny (2005),Richard et al. (2005),Liang & Shi (2010) e Liang & Shi (2010) propõem novas metodologias para a melhoria no desempenho dos clusters. Há também aqueles que propoem uma solução híbrida, que consiste no paralelismo local com o paralelismo distribuído (Clematis & Corana (1999), Rus et al. (2003), Tang et al. (2010), Agha & Kim (1999), Sunderam & Geist (1999), Lastovetsky & Reddy (2006), Cap & Strumpen (1993)).

3.3 Arquiteturas Paralelas

Existem diferentes formas (arquiteturas e modelos) para a aplicação de técnicas de computação paralela. Estas arquiteturas e modelos são responsáveis pelas estruturas que serão utilizadas para utilização do paralelismo para solução de um determinado problema. A escolha da arquitetura e do modelo que melhor se ajusta ao problema pode ser pela estrutura da máquina que será utilizada (qual arquitetura é suportada pela máquina) e também pode ser definida pelo estudo do problema e verificação de qual é a melhor solução para o mesmo.

Máquinas paralelas possuem características que determinam sua forma de paralelismo, portanto foi necessário desenvolver arquiteturas para que uma vez conhecida a forma de processamento da máquina, fosse possível utilizar desta arquitetura sobre qual se baseia o desenvolvimento da solução no âmbito de programação.Existem várias formas de se aplicar o paralelismo a um determinado problema, segundo a taxonomia de Flynn (Flynn, 1972; Grama et al., 2003):

- SISD (*Single Instruction Single Data*): conhecido como fluxo único de instruções sobre um determinado conjunto de dados. É o caso de execução de tarefas em

máquinas convencionais, também conhecida como Von Neumann (Figura 3.3);

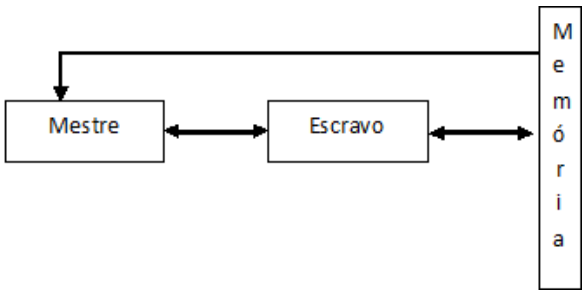

Figura 3.3: Arquitetura SISD.

- SIMD (*Single Instruction Strem Multiple Data Stream*): neste caso a mesma operação (função) é executada sobre vários dados (Figura 3.4);

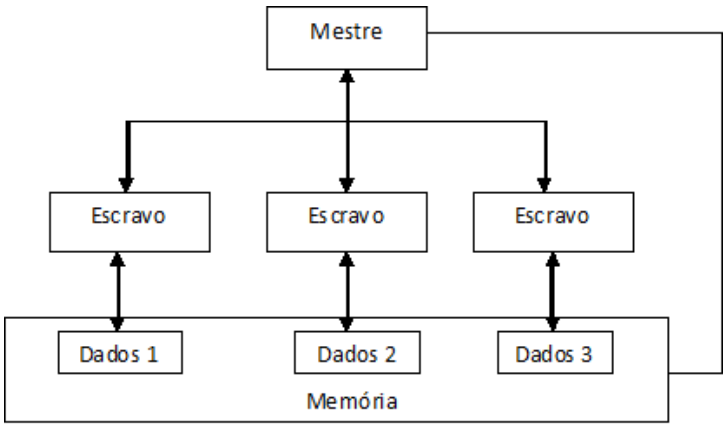

Figura 3.4: Arquitetura SIMD.

- MISD (*Multiple Instruction Stream Single Data Stream*): os dados vão sendo modificados por cada unidade de processamento e passados para a unidade de processamento subsequente, pois cada uma executa uma operação diferente da outra (Figura 3.5);

- MIMD (*Multiple Instruction Stream Multiple Data Stream*): o procedimento similar ao MISD. A diferença está no fato de não existir seqüência de processamento, então os dados processados são devolvidos diretamente para memória e não para o próximo processo (Figura 3.6);

Atualmente as máquinas paralelas, são desenvolvidas baseadas na arquitetura MIMD. Porém a estrutura de programação ainda mais utilizada para paralelizar um determinado problema baseia-se no modelo SPMD (*Single Program Multiple Data*),

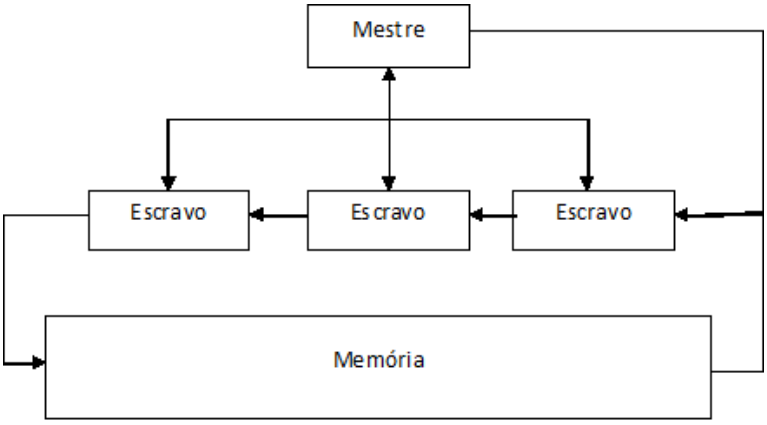

Figura 3.5: Arquitetura MISD.

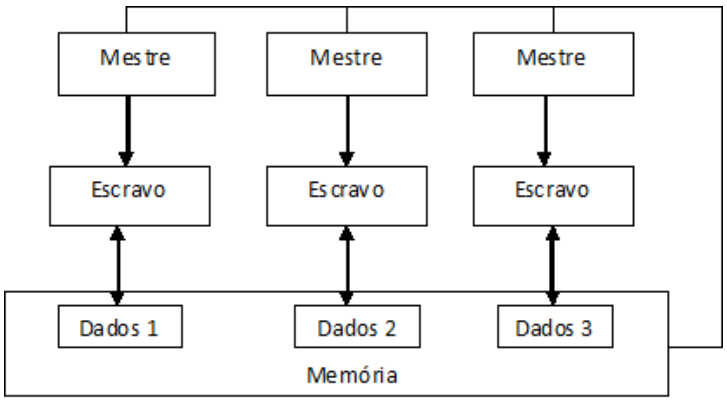

Figura 3.6: Arquitetura MIMD.

que é baseado na estrutura de mestre/escravo e também representa a forma de funcionamento da arquitetura SIMD. O modelo SPMD executa o mesmo trecho de código ou programas completos em cada processo. Por exemplo o código apresentado a seguir seria executado igualmente em cada núcleo de processamento (Grama et al., 2003).

```
1  for (i = 0; i < 10; i++)
2          c[i] = a[i] + b[i];
```

No código apresentado cada processador irá calcular $c[0] = a[0] + b[0]$ até $c[10] = a[10] + b[10]$, isto é possivel por não haver dependência dos dados que estão sendo utilizados, ou seja, os dados de $\vec{a}$ e $\vec{b}$ não estão sendo manipulados por outro processo qualquer, pois estes dados estão sendo utilizados somente para leitura. Portanto é possível definir no algorítmo a subdivisão dos dados que serão utilizados por cada processador.

A arquitetura paralela a ser utilizada deverá ser escolhida baseada nos recursos de *hardware* disponível para implementação da mesma.

3.4 Modelo de Comunicação da Computação Paralela

A comunicação entre as tarefas podem se dar basicamente de duas formas: acesso a dados compartilhados e troca de mensagens entre as tarefas.

3.4.1 Acesso a Dados Compartilhados

O modelo de acesso a dados compartilhados possibilita que todos os processos que estão em execução possam utilizar todos os dados que estão na memória (Grama et al., 2003). Este modelo suporta o modelo de programação SPMD e modelo distribuído (*cluster*).

Mas o acesso aos dados compartilhados pode ser dividido em dois tipos:

- UMA(*Uniform memory access*): todos os processos acessam a memória de forma igual e com o mesmo tempo de acesso.
- NUMA(*Non-uniform memory access*): não existe uma forma definida de acesso a memória, o que torna mais lento o acesso à memória.

A presença de uma memória global facilita a programação de sistemas paralelos, porém é necessário que haja uma atenção especial na parte de escrita na memória devido a concorrência dos processos (Ni, 1991). Se um processo tentar escrever em uma determinada região em que algum processo estiver utilizando o determinado dado que estiver alocado neste espaço certamente ocorrerá um erro de execução. Outro fator preocupante é o fato de processadores possuírem memórias *caches*, o que possibilita a duplicação de dados e isso pode permitir que processos trabalhem com dados desatualizados. Portanto precauções devem ser tomadas quando se trabalha com memória global compartilhada (Dally, 1987).

3.4.2 Troca de Mensagens

Neste modelo não é utilizada a idéia de uma memória global compartilhada, o que se tem são vários nós[1] com suas memórias locais. Estes nós podem ser processadores

[1]Um nó é um núcleo de processamento, este pode ser um processador físico ou então um computador presente em um *cluster*.

simples ou multiprocessadores que compartilham uma memória local (Feng, 1981). Neste modelo as iterações entre os processos em execuções em diferentes nós passam a serem feitas através de trocas de mensagens entre os nós. Estas mensagens são utilizadas para transferência de um dado, sincronização de processadores, etc (Grama et al., 2003).

Este modelo de troca de mensagem baseia-se em duas funções principais que são *send* (enviar) e *receive* (receber), as sintaxes destas funções podem variar de acordo com as API's[2], mas em geral são muito parecidas). Estas mensagens devem ser endereçadas para que a mesma realmente chegue ao seu destino, e para que isto aconteça torna-se necessário a identificação de cada nó. Para auxiliar na etapa de identificação, geralmente é possivel utilizar funções do tipo *whoami* (quem sou eu) cujo nó retornará seu identificador. Outra função importante que é padrão neste modelo é a função *numprocs*, que retorna a quantidade de processos envolvidos no trabalho (Ni, 1991). Com estas quatro funções básicas, é possível realizar a troca de mensagens entre os nós. As duas APIs de troca de mensagens mais difundidas são: *Message Passing Interface* (MPI) e *Parallel Virtual Machine* (PVM), estas que incorporam as quatros funções base para troca de mensagem, e possuem funções de alto nível para minimizar os esforços dos programadores.

[2]Interface de Programação de Aplicações: é um conjunto de funções que facilita o desenvolvimento de softwares. Ex: API para redes, API para ambientes gráficos e etc.

Capítulo 4

Estimadores Bi-Objetivo Acelerado e Paralelo.

4.1 Motivação

Este trabalho teve como motivação a investigação dos efeitos da quantidade de informação presente nos dados dinâmicos em comparação aos dados estáticos, no que tange à estimação bi-objetivo de parâmetros. Para facilitar, sejam as seguintes definições: (i) dados não-complementares, quando os dados dinâmicos e estáticos são redundantes, ou seja, os dados dinâmicos excursionam todos os pontos de operação de interesse; (ii) dados complementares, quando os dados dinâmicos estão limitados à uma região dos pontos de operação de interesse.

Para averiguar o comportamento destes tipos de dados foram realizados testes preliminares em sistemas reais cujos dados podem ser caracterizados como acima. Como esperado, surgiram algumas evidências a respeito do aparecimento de mais de um mínimo, ou seja, mínimos locais, na superfície de decisão do Pareto, que se baseia na procura pela mínima correlação entre os dados dinâmicos simulados e os resíduos de identificação infinitos passos à frente. Estes testes foram realizados de forma que o número de pontos no Pareto foi aumentado gradativamente para analisar o comportamento da escolha do "melhor" ponto e sua correlação (Tabela 4.1).

Toda a conjectura deste trabalho se baseou primordialmente nos dados que se encontram na Tabela 4.1. Algumas análises tais como a verificação do número de mínimos na superfície foi avaliada e motivou o desenvolvimento deste trabalho.

Tabela 4.1: Relação da correlação com a quantidade de pontos para dados complementares

	Pontos no Pareto	Ponto Escolhido	Correlação
1	3	3	0,313421
2	4	4	0,313421
3	5	5	0,313421
4	6	6	0,313421
5	7	7	0,313421
6	8	8	0,313421
7	9	9	0,313421
8	10	10	0,313421
9	11	11	0,313421
10	12	12	0,313421
11	13	13	0,313421
12	14	14	0,313421
13	15	2	0,313148
14	16	2	0,312325
15	100	3	0,304150
16	1000	2	0,098330
17	10.000	2	-0,843598
18	20.000	2	-0,874065
19	30.000	2	-0,879672
20	40.000	2	-0,881825

A Tabela 4.1 apresenta os resultados da simulação para o Conversor Estático CC-Buck Seção 4.1.2, apresentando a quantidade de pontos no pareto, este que foi apresentado na Seção 2.3.1.1, o ponto escolhido (Seção 2.3.4) e a correlação do ponto escolhido(Seção 2.3.4). É possível observar pela Tabela 4.1 que da linha 1 à 12 os valores de correlação escolhidos estão em uma extremidade do Pareto, no caso, aquela extremidade relacionada à minimização dos resíduos de identificação. No entanto, a escolha se altera a partir da linha 13. Com 15 pontos no Pareto, a partir do 15° ponto começa a haver variações no valor da correlação e há uma tendência de serem escolheridos pontos na outra extremidade do Pareto, no caso, o que se refere ao ajuste de curva estática. Este mostra que há uma maior penalização dos dados dinâmicos em detrimento dos dados estáticos. Porém, ainda não há convergência nos valores de

correlação, que só começam a convergir a partir de 10.000 pontos no Pareto, em que o intervalo entre a quantidade de pontos fica cada vez maior e o valor da correlação possui pequenas variações. Portanto, como o λ penaliza mais os dados dinâmicos e como os dados estáticos contém informações complementares, a partir de um determinado número de pontos no Pareto, o mínimo passa para uma região próxima ao funcional estático. No caso de dados não-complementares, há um comportamento assintótico para um valor de mínimo global. Os testes realizados apontam para a necessidade de utilizar de uma grande quantidade de pontos no Pareto para sistemas que possuem dados complementares.

Dois testes com sistemas reais foram realizados para averiguação do comportamenteo do estimador para dados complementares e não-complementares.

4.1.1 Dados Não-Complementares - Processo Térmico

Este primeiro estudo é baseado na estrutura e dados disponíveis em Cassini (1999). Trata-se de um processo térmico com dados reais, cujos dados dinâmicos e estáticos são não-complementares. A aplicação da metodologia a um sistema real, permite verificar a convergência, na presença de incertezas nos dados.

Os resultados foram obtidos para valores crescentes de numeros de pontos no Pareto: 12; 100; 1000; 10.000; 100.000 pontos.

- Simulação com 12 pontos no Pareto.

A Figura(4.1) apresenta o conjunto de correlação para a estimação com 12 pontos no Pareto-Ótimo.

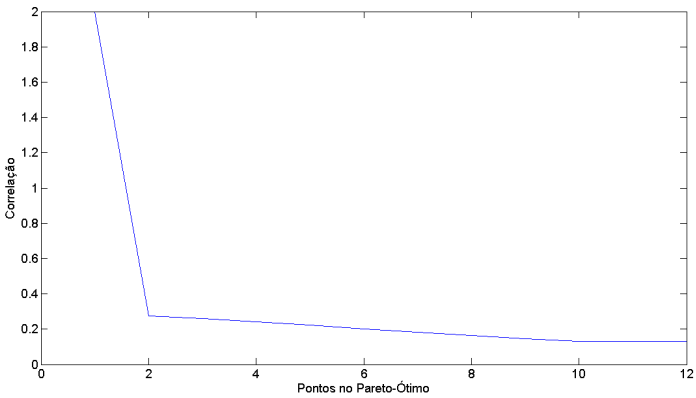

Figura 4.1: Dados Não-Complementares - Conjunto de correlações com 12 pontos no Pareto.

Pode-se observar que ela apresenta um único valor de mínimo no conjunto, sendo ele $0,13056$. O que a Figura (4.1) não mostra, é que o valor de mínima correlação está na região próxima ao mínimo dinâmico, ou seja, próximos aos obtidos pelos mínimos quadrados convencional. Nos trabalhos de Cassini (1999) é possível verificar que este resultado é coerente. Então, espera-se que a mínima correlação seja convergente para o valor utópico no Pareto. Este resultado ficará mais evidente para as simulações seguintes.

- Simulação com 100 pontos no Pareto.

A Figura (4.2) apresenta o conjunto de correlação para a estimação com 100 pontos no Pareto-Ótimo. É possível observar, que a correlação apresenta um único valor de mínimo no conjunto. Porém o valor do mínimo atual é $0,13056$. Já é possível, neste momento, verificar graficamente, que a mínima correlação converge para o valor utópico.

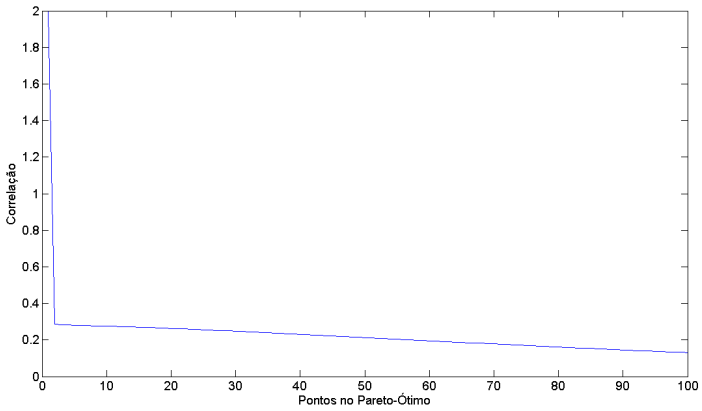

Figura 4.2: Dados Não-Complementares - Conjunto de correlações com 100 pontos no Pareto.

- Simulação com 1.000 pontos no Pareto.

Na Figura (4.3) foi realizada a estimação com 1.000 pontos no conjunto Pareto, observa-se que o ponto de mínimo nao pode ser determinado, mas é possível observar que existe um único mínimo na curva.

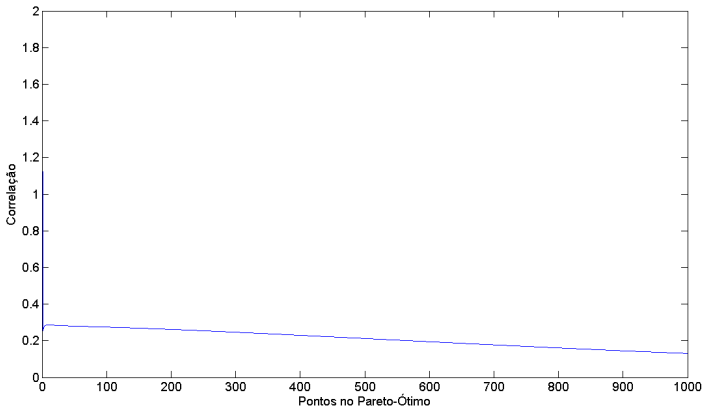

Figura 4.3: Dados Não-Complementares - Conjunto de correlações com 1000 pontos no Pareto.

Portanto para explicitar o mínimo foi feita uma ampliação na Figura (4.3) como mostrado na Figura (4.4).

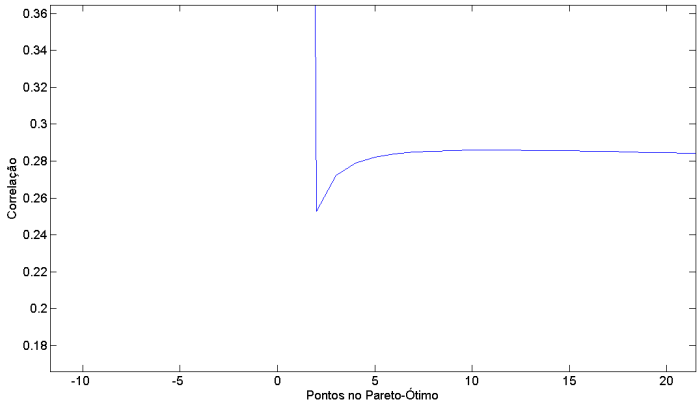

Figura 4.4: Dados Não-Complementares - Ampliação da Figura (4.3) na região do valor mínimo.

A Figura (4.4) é uma ampliação da Figura (4.3) para mostrar o valor do mínimo para estimação com 1.000 pontos. Na Figura (4.4) é possível observar que o valor de mínimo continua sendo o mesmo para simulações com número inferior de pontos no conjunto Pareto, ou seja, $0,13056$.

- Simulação com 10.000 pontos no Pareto.

A Figura (4.5) apresenta o conjunto de correlações para a estimação de parâmetros com 10.000 pontos no Pareto. Uma ampliação da região deste ponto de mínimo pode ser vista na Figura (4.6). O valor de correlação é $0,13056$.

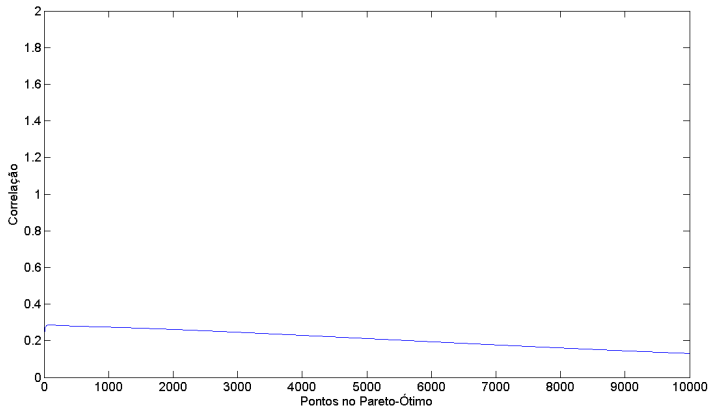

Figura 4.5: Dados Não-Complementares - Conjunto de correlações com 10000 pontos no Pareto.

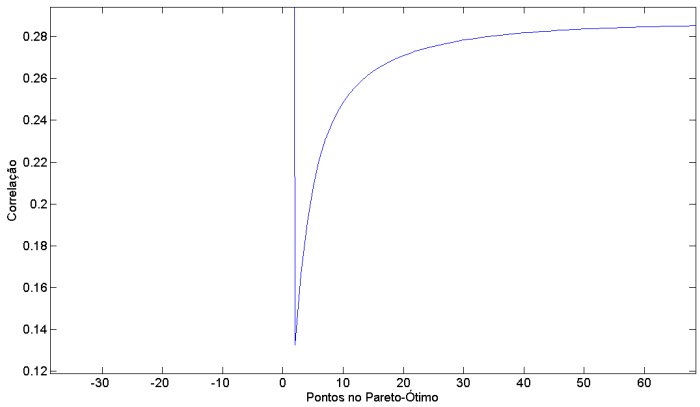

Figura 4.6: Dados Não-Complementares - Ampliação da Figura (4.5) na região do valor mínimo.

- Simulação com 100.000 pontos no Pareto.

Assim como nas Figuras (4.3 e 4.5), não é possível visualizar o ponto de mínimo na Figura (4.7), somente é possível afirmar que a mesma apresenta um único valor de mínimo. Portanto a Figura (4.8) apresenta uma ampliação na região do valor de mínimo da mesma.

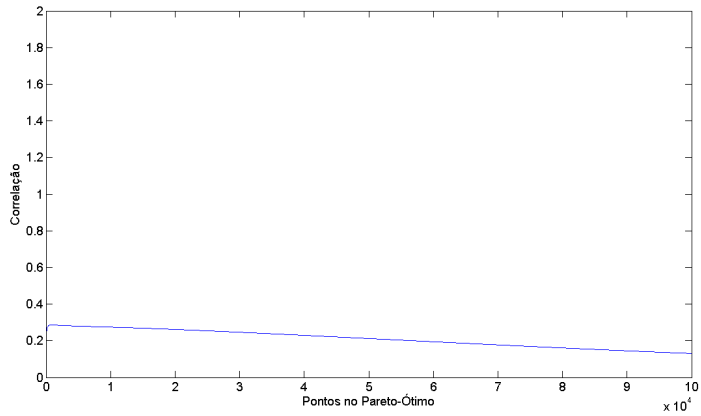

Figura 4.7: Dados Não-Complementares - Conjunto de correlações com 100000 pontos no Pareto.

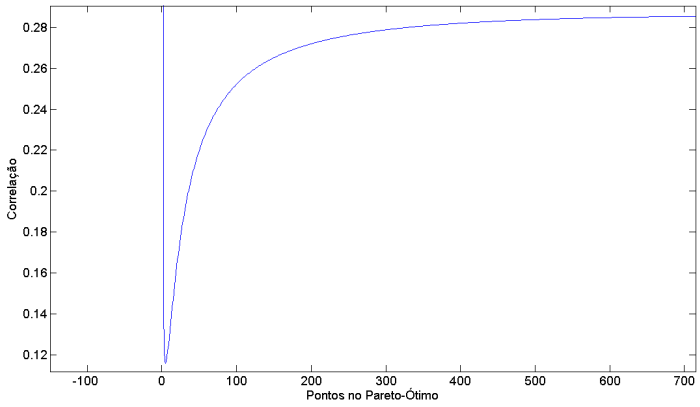

Figura 4.8: Dados Não-Complementares - Ampliação da Figura (4.7) na região do valor mínimo.

A Figura (4.8) mostra uma tendência do valor de mínimo $(0,11576)$, para um mínimo utópico, ou seja para o valor de erro zero (funcionais tendem para zero). Portanto é possível observar em todas as simulações que quanto maior a quantidade de pontos no conjunto Pareto-Ótimo melhor será a correlação.

4.1.2 Dados Complementares - Conversor Estático CC-Buck

Nesta seção foram realizados testes com dados de um conversor CC-CC Buck (Figura 5.10). Maiores informações sobre este conversor podem ser obtidas em Barroso (2006). As análises da aplicação da metodologia a este sistema segue a mesma metodologia usada na Seção 4.1.1.

- Simulação com 12 pontos no Pareto.

Na estimação para 12 pontos no Pareto-Ótimo, é possível observar pela Figura(4.9) que o conjunto apresenta um único valor de mínimo $0,31342$.

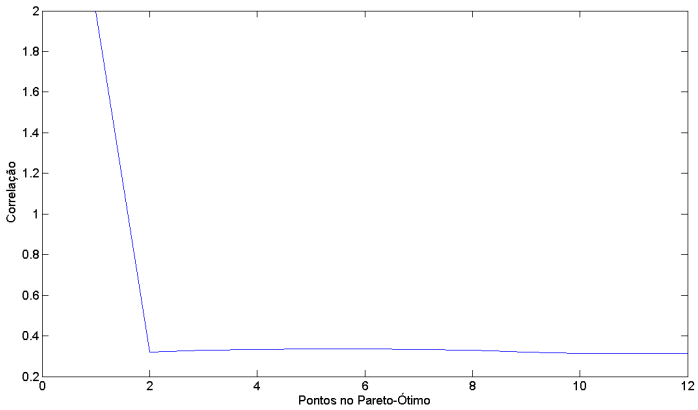

Figura 4.9: Dados Complementares - Conjunto de correlação para 12 pontos no Pareto-Ótimo.

- Simulação com 100 pontos no Pareto.

A Figura (4.10) também apresenta um único valor de correlação mínima $0,30414$ no conjunto de correlações para a estimação de parâmetros com 100 pontos.

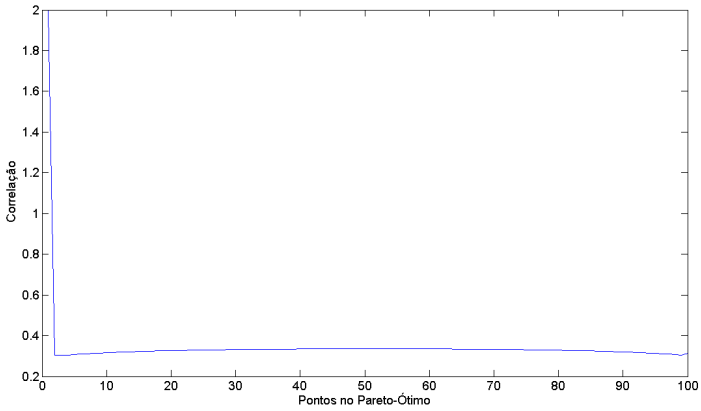

Figura 4.10: Dados Complementares - Conjunto de correlação para 100 pontos no Pareto-Ótimo.

- Simulação com 1.000 pontos no Pareto.

Pela Figura(4.11) é possível observar que existe mais de um valor de mínimo.

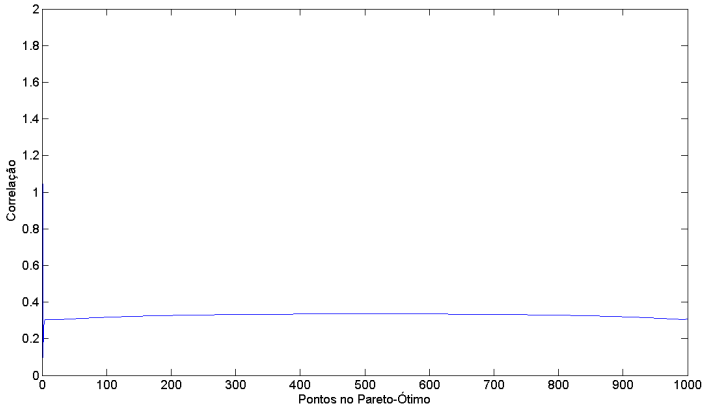

Figura 4.11: Dados Complementares - Conjunto de correlação para 1000 pontos no Pareto-Ótimo.

Portanto serão ampliados os trechos da Figura (4.11) que apresentam tendências para mínimos, a fim de observar a quantidade de valores mínimos no conjunto de correlações.

A Figura (4.13) apresenta 2 gráficos com trechos onde os valores de mínimos aparecem, e como pode ser observado existem três valores de mínimos no conjunto, porém a correlação mínima escolhida é $0,09833$.

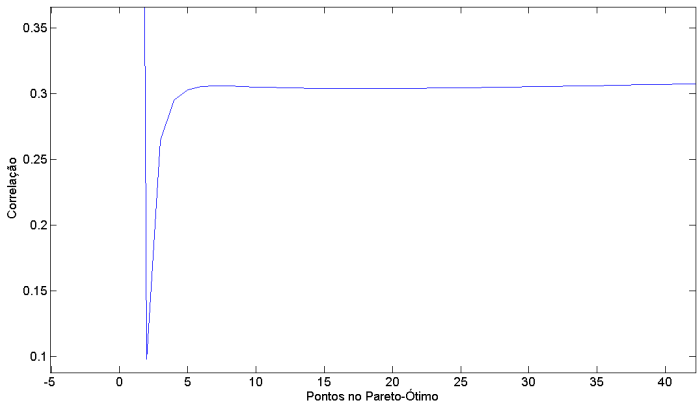

Figura 4.12: Dados Complementares - Mínimos local e global da Figura (4.11)

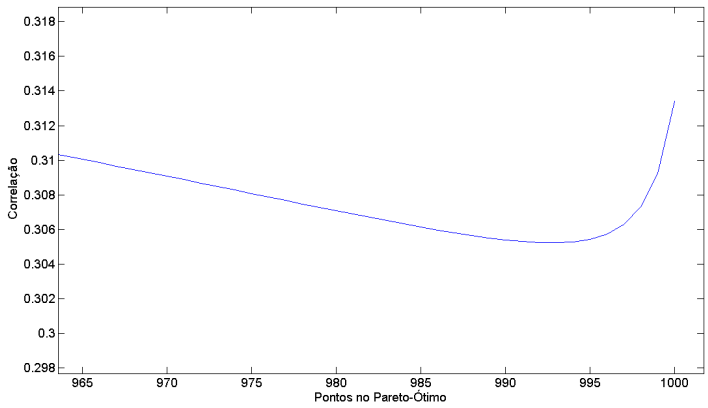

Figura 4.13: Dados Complementares - Mínimos local da Figura (4.11)

- Simulação com 10.000 pontos no Pareto.

É possível observar pelas Figuras(4.11 e 4.14), que ambas têm características semelhantes, portanto novamente será necessário ampliar a região dos mínimos para verificar se existe alguma divergência entre estes pontos de mínimos.

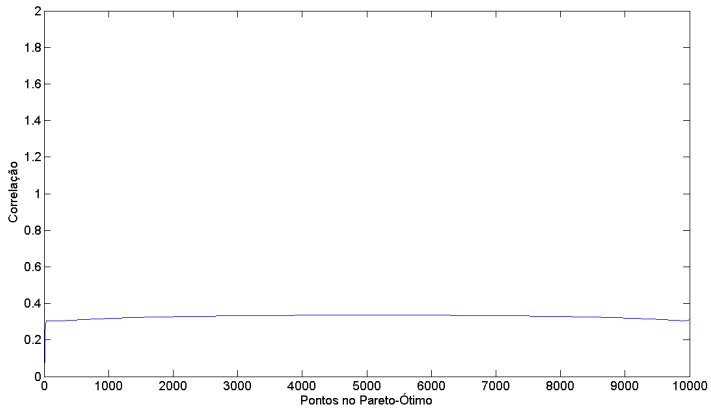

Figura 4.14: Dados Complementares - Conjunto de correlação para 10000 pontos no Pareto-Ótimo.

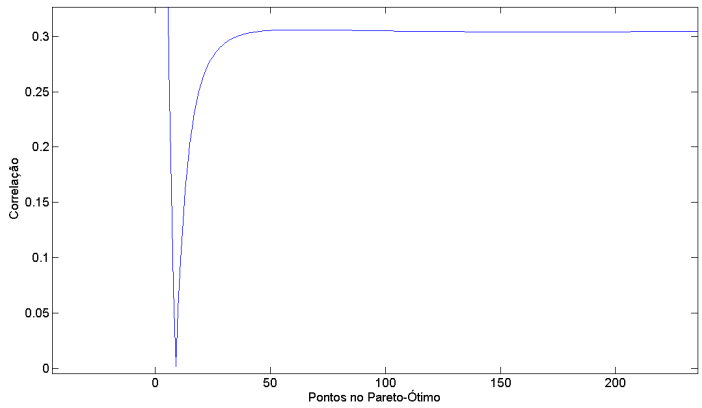

Figura 4.15: Dados Complementares - Mínimos local e global da Figura (4.14)

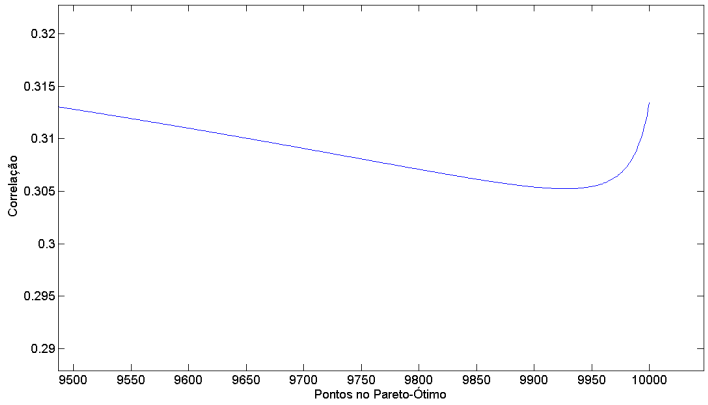

Figura 4.16: Dados Complementares - Mínimo local da Figura (4.14)

Pela Figura (4.16) é possivel observar que o acréscimo de pontos no Pareto-Ótimo possibilitou que a correlação mínima global $0,00179$ convirja para um determinado valor.

- Simulação com 100.000 pontos no Pareto.

Ainda com as características do conjunto mantidas, a Figura (4.17) não apresenta diferenças perceptíveis em relação as Figuras(4.11 e 4.14). Portanto a ampliação dos mínimos novamente será utilizada para averiguação da convergência dos mínimos.

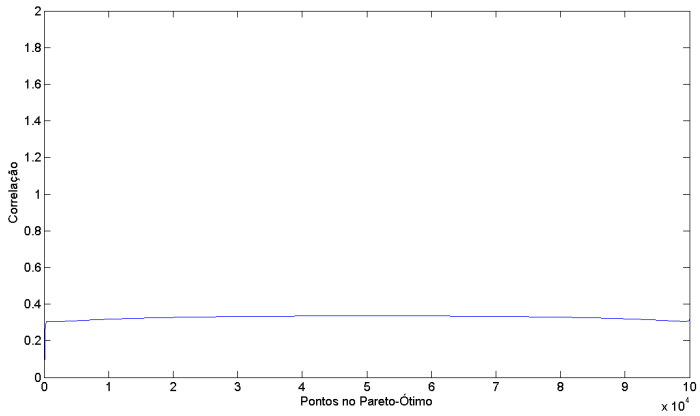

Figura 4.17: Dados Complementares - Conjunto de correlação para 100000 pontos no Pareto-Ótimo.

A Figura (4.18) apresenta a correlação mínima com o valor de $0,00184$.

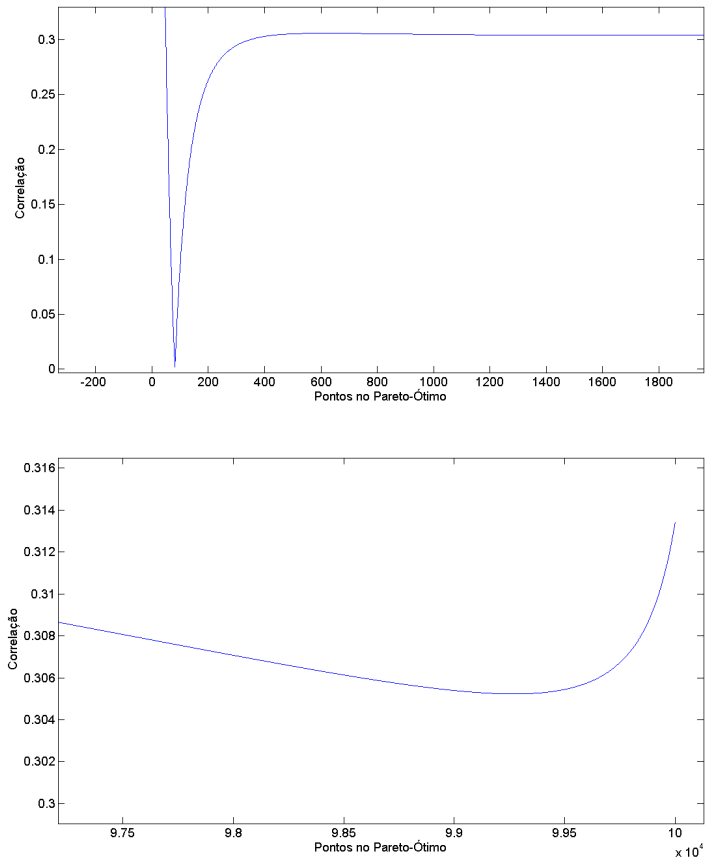

Figura 4.18: Dados Complementares - Mínimos local e global da Figura(4.17).

Este experimento mostra que no caso complementar, faz-se necessário gerar "todo" o Pareto. É necessário entender que gerar "todo" o Pareto não significa gerar infinitos pontos, mas sim o suficiente para que se conheça a região do mínimo global da superfície de escolha da mínima correlação.

Sendo assim, duas conjecturas surgem:

Conjectura 1. *Existe um único valor mínimo de correlação, quando os dados dinâmicos e estáticos não são complementares. Devido os dados não serem complementares ou seja são dados "bons", o processo de estimação não necessita de uma quantidade exorbitante de pontos no Pareto-Ótimo.*

Conjectura 2. *Existe mais de um valor mínimo de correlação, quando os dados dinâmicos e estáticos são complementares. Portanto é necessário que sejam utilizados muitos pontos na geração do Pareto-Ótimo*

4.1.3 Tempo de solução

Uma vez que a quantidade de pontos no Pareto é um fator determinante para a qualidade dos resultados estimados, é necessário avaliar o tempo da solução. Dado que a geração do Pareto é computacionalmente cara (Lesmere et al., 2007a; Wuppalapati et al., 2008) e que há casos em que a quantidade de pontos no Pareto necessita ser elevada, o fator tempo torna-se um empecilho para utilização deste estimador.

Durante as simulações para dados não-complementares e complementares no estimador bi-objetivo, foi possível observar que o acréscimo de pontos no Pareto ocasionava um significativo aumento no tempo de estimação, como pode ser visto na Figura (4.19). Este fato se traduz em um alto custo computacional.

Ao se verificar o formato da curva mostrada na Figura (4.19), o seguinte polinômio (Equação 4.1) de 2ª ordem pode ser ajustado:

$$f(x) = ax^2 + bx + c \tag{4.1}$$

cujo parâmetros a, b e c variam de acordo com a quantidade de dados dinâmicos e estáticos. A Equação (4.1) apresenta uma relação quadrática entre o tempo e a quantidade de pontos (x) no Pareto. Isto faz com que o tempo de espera do estimador bi-objetivo muito alto para elevados números de pontos no Pareto, fato que é muito importante para uma boa identificação.

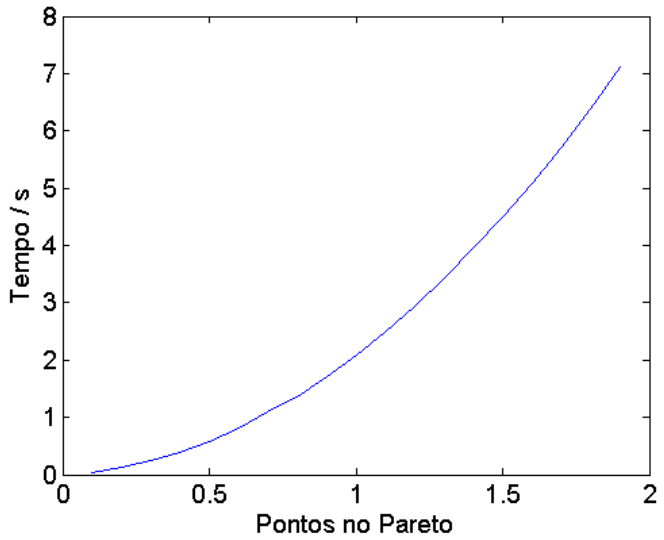

Figura 4.19: Tempos para estimador bi-objetivo de 1.000 à 19.000 pontos no Pareto.

Este foi o fator preocupante que motivou a busca por soluções no processo de estimação que utilizassem números elevados de pontos no Pareto, porém com um tempo de solução menor e mais atraente.

4.2 Implementação do Estimador Bi-Objetivo Paralelo

Para a paralelização do estimador bi-objetivo neste trabalho foi implementada a técnica SPMD (Seção 3.3), em que o mestre é o gerenciador e tem controle sobre os dados na memória e também sobre as tarefas que estão sendo executadas em cada escravo. Os escravos executam o processo enviado pelo mestre e retornam o resultado obtido.

4.2.1 Paralelização do Estimador Bi-Objetivo

Para aplicação da técnica SPMD no Estimador de Parâmetros Bi-Objetivo foram necessárias algumas alterações nas etapas do estimador, entretanto o conjunto Pareto continua com a definição do Capítulo 2. Os dados passam pelas seguintes alterações no processo de estimação: seja p a quantidade de processos a serem executados, e $\mathbf{\Gamma}$ um vetor de valores de λ, dividido em p partes. Vale ressaltar que não são os valores de λ que são divididos, mas o número de valores de λ processados em cada núcleo, por processo disparado. Seja $\vec{\zeta}$ um vetor de soluções retornado de cada processo.

Com estas considerações, a estimação dos parâmetros será calculada pela equação(4.2):

$$\hat{\vec{\zeta}}^{*}{}_{i}\left(\Gamma\right) = \begin{bmatrix}\Gamma\Psi^{T}\Psi + \left(1-\Gamma\right)\left(\mathbf{Q}R\right)^{T}\mathbf{Q}R\end{bmatrix}^{-1} \begin{bmatrix}\Gamma\Psi^{T}\Psi + \left(1-\Gamma\right)\left(\mathbf{Q}R\right)^{T}\bar{\vec{y}}\end{bmatrix} \tag{4.2}$$

A equação (4.2) mostra o resultado da estimação, em que $\hat{\vec{\theta}}^{*}$ é o vetor contendo o conjunto dos p vetores $\vec{\zeta}$ e i é o identificador de cada processo $\vec{\zeta}_i$:

$$\hat{\vec{\theta}}^{*} = \begin{bmatrix} \vec{\zeta}_1 \\ \vec{\zeta}_2 \\ . \\ . \\ . \\ \vec{\zeta}_{\mathrm{p}} \end{bmatrix} \tag{4.3}$$

Uma vez obtido $\hat{\vec{\theta}}^{*}$, os parâmetros deste conjunto são utilizados na simulação livre do modelo $\hat{\vec{\mathrm{y}}}$. Os dados são também repartidos em iguais quantidades para cada processo. Foi definido um vetor $\vec{\epsilon}$ para armazenar os dados de cada processo. O conjunto $\hat{\vec{\theta}}^{*}$ é então subdivido em p conjuntos $\vec{\eta}$, da seguinte forma:

$$\hat{\vec{\epsilon}}_i\left(k\right) = \hat{\Psi}\left(k-1\right)\hat{\vec{\eta}}_i. \tag{4.4}$$

O vetor das p simulações livre $\hat{\vec{y}}$ é então organizado conforme a equação (4.5):

$$\hat{\vec{\mathrm{y}}} = \begin{bmatrix} \vec{\epsilon}_1 \\ \vec{\epsilon}_2 \\ . \\ . \\ . \\ \vec{\epsilon}_{\mathrm{p}} \end{bmatrix} \tag{4.5}$$

Seja ainda $\mathbf{C}_{\mathrm{i}}$ um vetor contendo os p conjuntos de valores de correlações cruzadas, dadas pela seguinte representação:

$$C_i = \frac{1}{N}\sum_{k=1}^{\frac{N}{P}} n(k)\hat{y}(k) \tag{4.6}$$

A etapa de busca da mínima correlação é dada pela Equação (4.7-4.9).

$$\vec{M}_i = \min_{\hat{\vec{\theta}} \in \hat{\mathbf{\Theta}}^*} \mathbf{C}_i \left(\hat{\vec{\eta}} \right). \tag{4.7}$$

$$\hat{\vec{\theta}} = \begin{bmatrix} \vec{M}_1 \\ \vec{M}_2 \\ . \\ . \\ . \\ \vec{M}_\mathrm{p} \end{bmatrix} \tag{4.8}$$

$$\min_{\hat{\vec{\theta}} \in \hat{\Theta}^*} \mathbf{J} \left(\hat{\vec{\theta}} \right). \tag{4.9}$$

Portanto pode-se dizer que na estimação Paralela, não foram feitas alterações nas estruturas matemáticas e o fluxograma do algoritmo se organiza da seguinte forma (Figura 4.20):

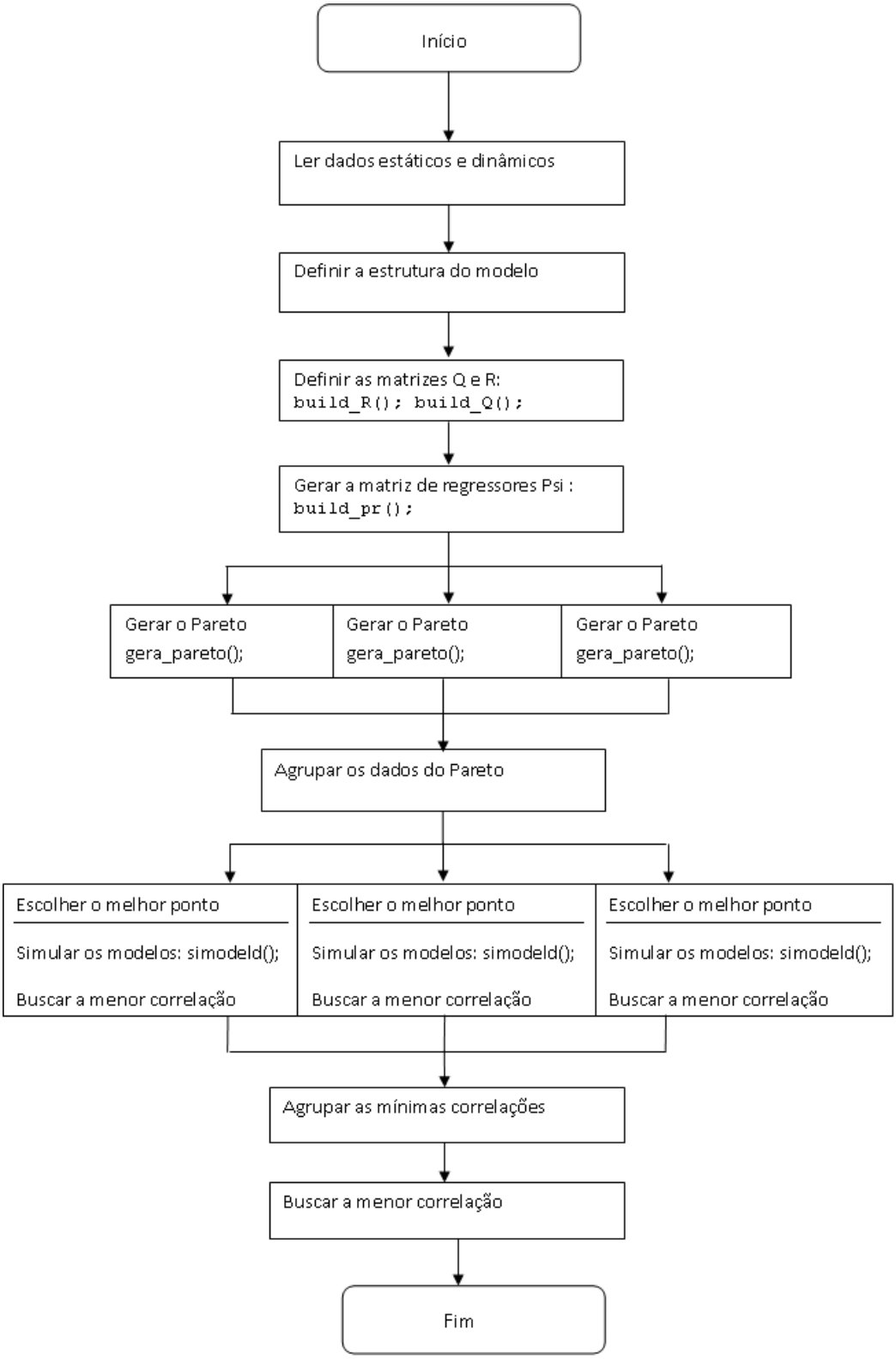

Figura 4.20: Fluxograma do algoritmo do estimador bi-objetivo paralelo.

4.3 Implementação do Estimador Bi-Objetivo Acelerado

Esta Seção é dedicada a apresentação de um método para acelerar o processo de solução do estimador bi-objetivo para casos cujos dados são não-complementares . O processo de solução é o mesmo apresentado no Capítulo(2). Nas soluções apresentadas anterirormente neste trabalho o conjunto λ éra composto por n valores no intervalo $0 \leq \lambda \leq 1$, no estimador bi-objetivo acelerado o λ passa a ter somente 3 valores a cada iteração, em que 2 valores de λ são os delimitadores das extremidades do conjunto e um λ intermediário subdivindo o conjunto em dois sub-conjuntos. Como a métrica de solução é baseada em uma busca de semi-espaços, neste trabalho será chamada de semi-Paretos.

Inicialmente trabalha-se com λ's extremos ou seja $\lambda_1 = 0 \quad \lambda_3 = 1$ e $\lambda_2 = 0,5$, portanto tem-se λ_1 o extremo inferior, λ_3 o extremo superior e λ_2 a fronteira entre λ_1 e λ_3, assim então delimitando 2 semi-Paretos (Figura 4.21)

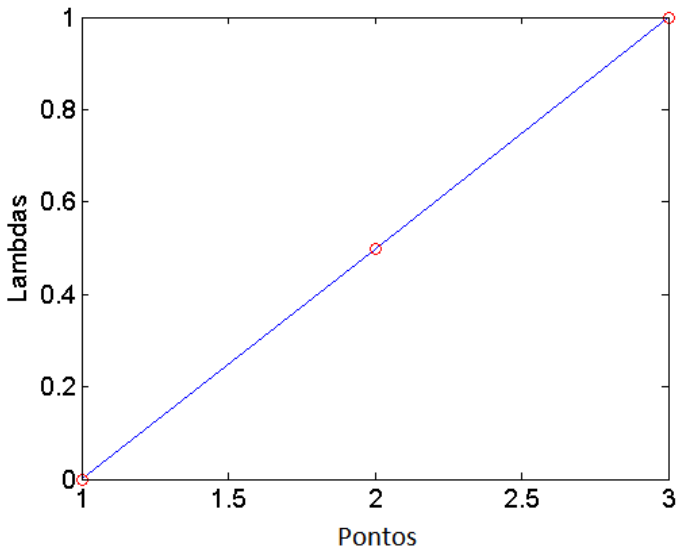

Figura 4.21: Primeira iteração bi-objetivo acelerado:(–): Conjunto dos λ's; (o): valores iniciais para λ.

Dentro de cada um semi-Pareto sorteia-se dois novos valores de λ que serão utilizados para a estimação.

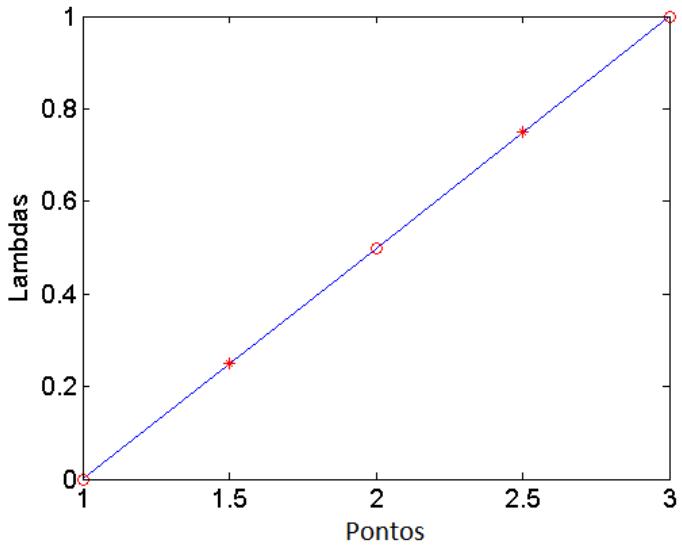

Figura 4.22: (–): Conjunto dos λ's; (o): limites dos semi-Paretos; (*): λ's para estimação.

Uma vez realizada a estimação com estes dois valores de λ, um dos semi-Paretos é eliminado. Conforme vem sendo utilizado neste trabalho o critério de escolha do melhor modelo é realizado pela menor correlação, e portanto, para a escolha do melhor semi-Pareto será também utilizado o critério da menor correlação. Então os novos semi-Paretos a serem trabalhados serão conforme Figura (4.23):

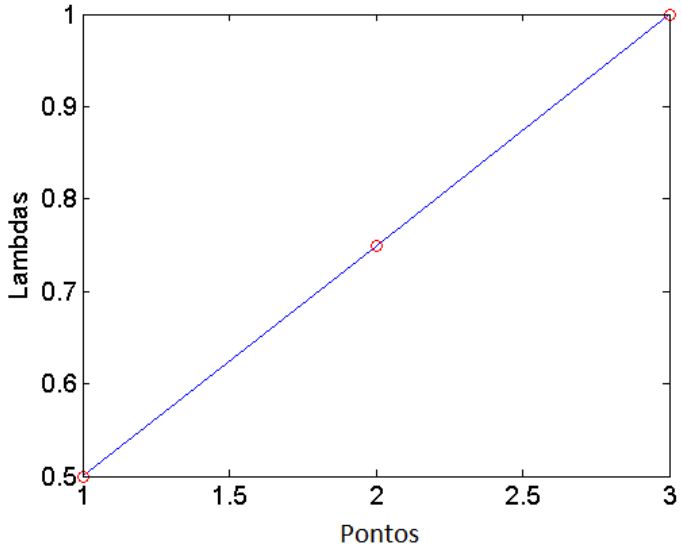

Figura 4.23: Segunda iteração bi-objetivo acelerado: (–): Conjunto dos λ's; (o): limites dos semi-Paretos.

Como é possível observar na Figura(4.23), o semi-Pareto inferior que estava no intervalo de $0 \leq \lambda \leq 0,5$ foi descartado, passando agora para o novo intervalo $0,5 \leq \lambda \leq 1$, o qual define a faixa para os novos valores para λ's que serão utilizados na próxima estimação (Figura 4.24):

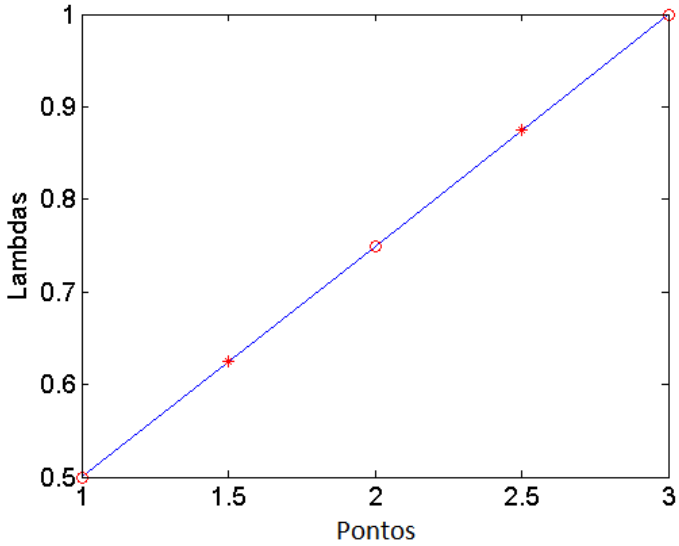

Figura 4.24: Segunda escolha bi-objetivo acelerado: (–): Conjunto dos λ's; (o): limites dos semi-Paretos; (*): λ's para estimação.

Após o término da estimação, os valores das correlações correspondentes a cada λ são novamente comparados, a fim de eliminar novamente o semi-Pareto que possui uma maior correlação. E como resultado os novos semi-Paretos serão (Figura 4.25):

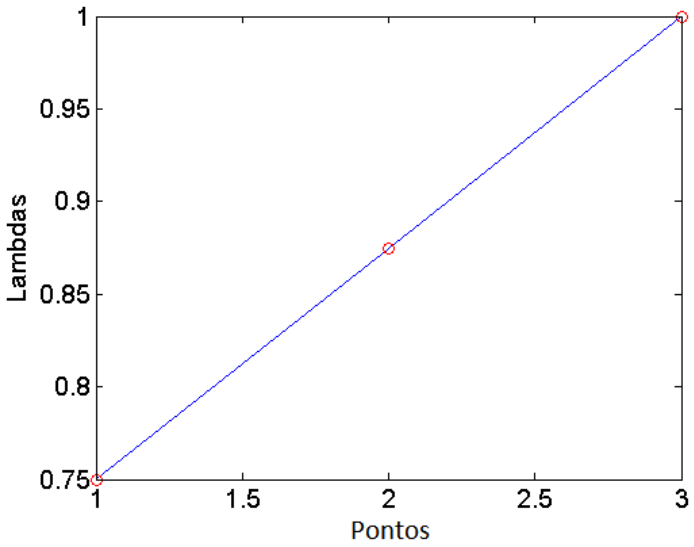

Figura 4.25: Terceira iteração bi-objetivo acelerado:(–): Conjunto dos λ 's; (o) Limites dos semi-Paretos.

E consequentemente os novos valores para λ serão conforme a Figura (4.26):

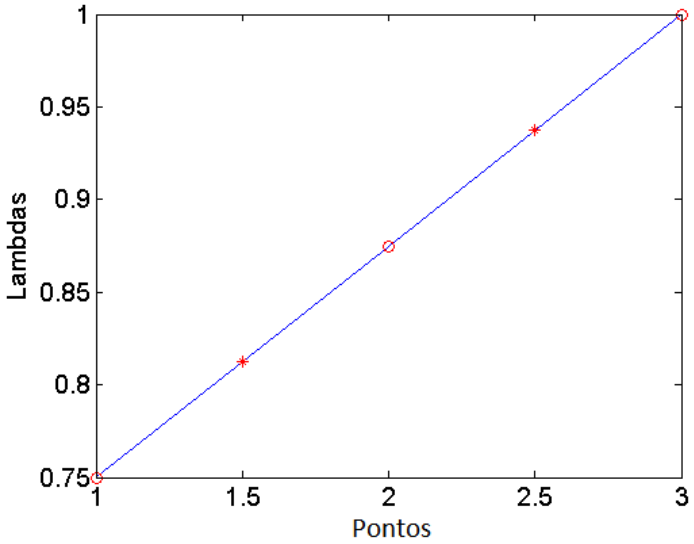

Figura 4.26: Terceira escolha bi-objetivo acelerado: (–): Conjunto dos λ's; (o): limites dos semi-Paretos; (*): λ's para estimação.

Estas iterações ocorrerão até que um critério de parada seja satisfeito. No caso deste trabalho, foi definida uma tolerância de erro. Ao final do processo os semi-Paretos serão praticamente iguais como mostra a Figura (4.27),

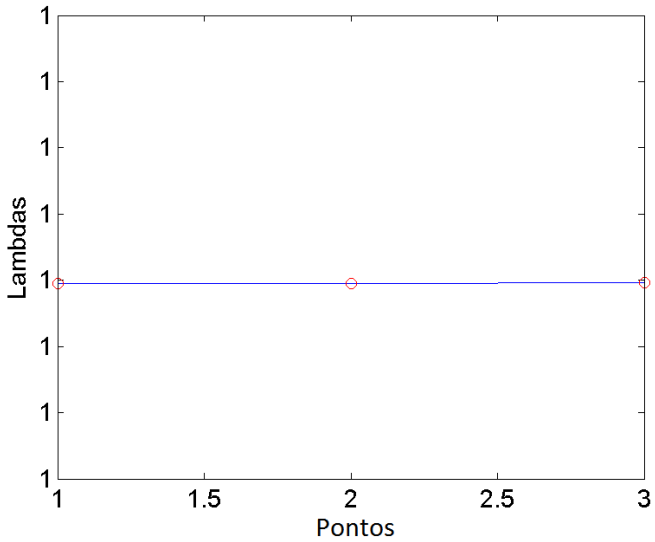

Figura 4.27: N-ésima iteração do bi-objetivo acelerado: (–): Conjunto dos λ's; (o): limites dos semi-paretos.

A Figura (4.28) trás os valores dos λ's iniciais. Note que os valores dos λ's para estimação são iguais.

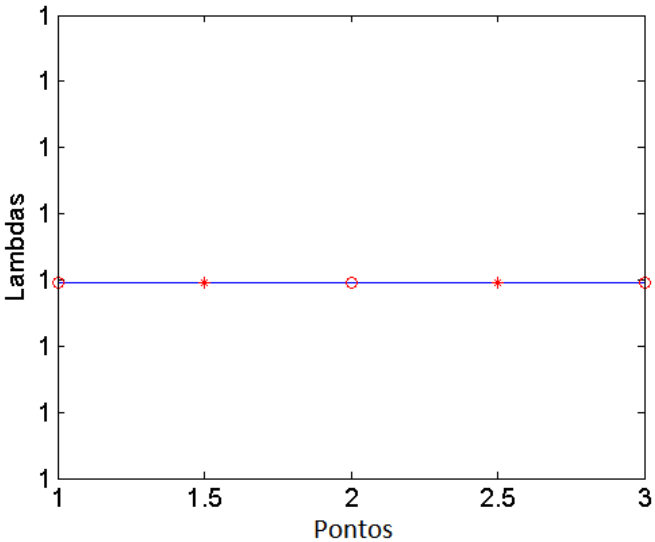

Figura 4.28: N-ésima escolha do bi-objetivo acelerado: (–): Conjunto dos λ's; (o): Limites dos semi-Paretos; (*): λ 's para estimação.

A Figura (4.29) mostra a estrutura do algoritmo de funcionamento do acelerador, mostrado pelas Figuras (4.21 - 4.28).

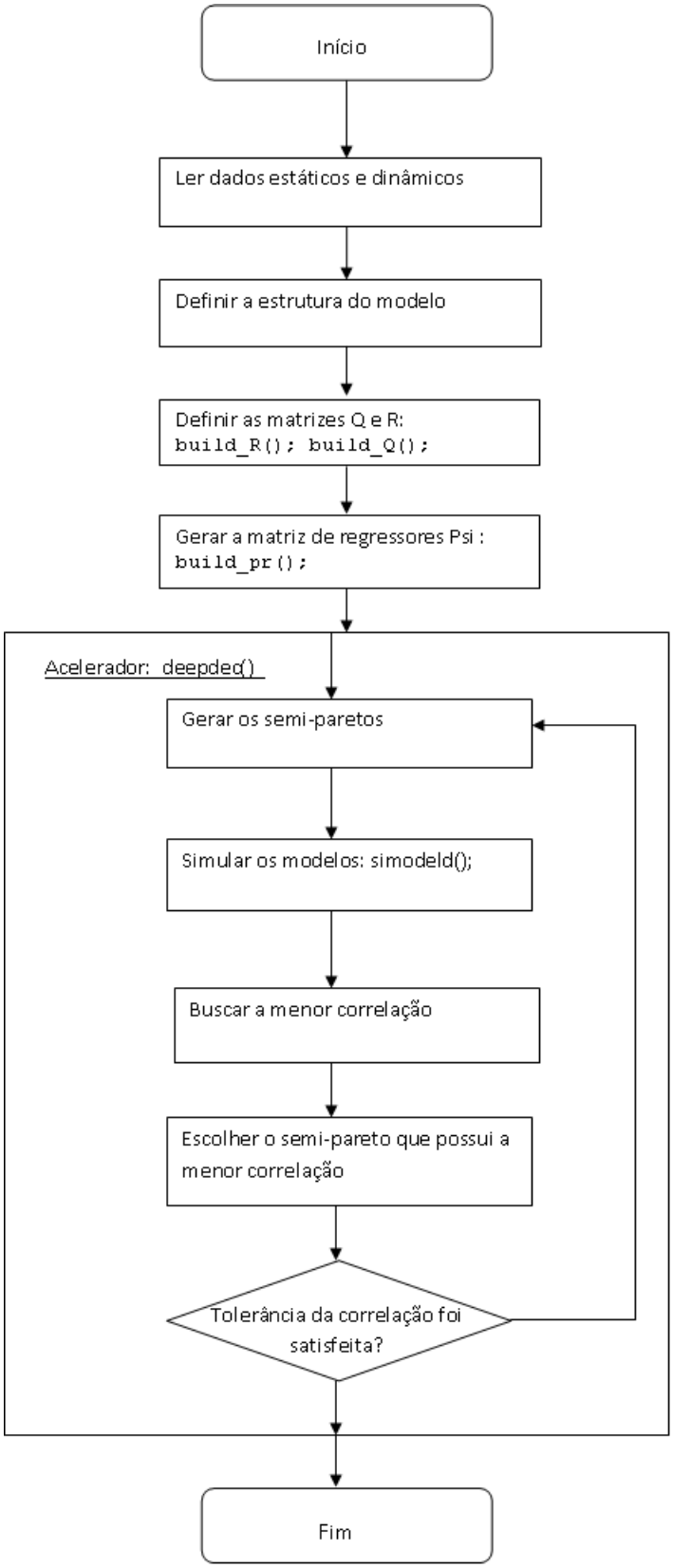

Figura 4.29: Fluxograma do algoritmo do estimador bi-objetivo acelerado.

Capítulo 5

Estudo de Casos

Este Capítulo tem como objetivo apresentar os resultados das aplicações dos métodos propostos no Capítulo 4, e também apresentar dados comparativos entre os estimadores bi-objetivo, bi-objetivo paralelo e bi-objetivo acelerado. Serão realizados dois tipos de testes, ambos com sistemas reais: o primeiro teste será realizado com dados não-complementares e o segundo teste será realizado com dados complementares.

5.1 Aplicação em um Polimerizador – Dados Não-complementares

Para teste e validação, as técnicas propostas no Capítulo 4, foram aplicadas ao sistema descrito em Doyle III et al. (1995) e Ray (1972), e os resultados obtidos foram mostrados em Marciano et al. (2010a,b). Trata-se de um processo de polimerização por radicais livres de metacrilato de metila (MMA), com azo-bisisobutironitrila (AIBN) como catalizador e tolueno como solvente. O dados foram simulados em um conjunto de quatro equações diferenciais não lineares de primeira ordem. A entrada é a taxa do fluxo volumétrico do iniciador e a saída é o número médio do peso molecular (variável controlada). A estrutura do modelo NARX polinomial utilizado nesta Seção foi determinada em Barroso et al. (2007) e é apresentada na Equação (5.1) a seguir:

$$\begin{array}{rcl} y(k) & = & \theta_1 y(k-1) + \theta_2 u(k-1)u(k-2) \\ & & \theta_3 u(k-1)u(k-1) + \theta_4 y(k-2) \\ & & \theta_5 u(k-2)y(k-1) + \theta_6 u(k-2)y(k-2) \\ & & \theta_7 u(k-2)u(k-2) \end{array}, \tag{5.1}$$

em que os θ_i são os parâmetros a serem estimados.

A estrutura estática é dada por:

$$\bar{y} = \Sigma_y \bar{y} + \Sigma_{u^2} \bar{u}^2 + \Sigma_{uy} \bar{u}\bar{y}, \tag{5.2}$$

em que Σ são os coeficientes de agrupamentos de termos, mapeados pela seguinte matriz de mapeamento:

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}. \tag{5.3}$$

Para as análises pertinentes deste trabalho, foram utilizados dois núcleos de processamento, totalizando um número $p = 4$ processos, 2 por núcleo. Os núcleos de processamento, bem como o *hardware* utilizado apresenta a seguinte configuração: processador Intel Core 2 Duo T5670 1.8 GHz, memória RAM de 4GB e unidade de armazenamento de 500GB.

5.1.1 Resultados

Foram realizados inicialmente 3 simulações para os estimadores bi-objetivo e bi-objetivo paralelo, para avaliar a relação entre diferentes quantidades de pontos no conjunto Pareto e os modelos obtidos para cada caso, sendo 10, 100 e 4000 pontos no Pareto.

Pela da Figura 5.1, pode-se observar que o número de elementos no Pareto é determinante para a qualidade da estimação, para ambos os estimadores bi-objetivo convencional e bi-objetivo paralelo. Sendo assim, para 10 elementos no Pareto, não foi possível obter um conjunto de parâmetros que adequasse a estrutura do modelo aos dados. Para 100 pontos, observa-se uma melhoria do resultado. Com 4000 pontos é nítido, a partir de uma inspeção visual, que o resultado obtido representa um ganho na qualidade da simulação do modelo, tanto no que diz respeito aos dados estáticos, quanto aos dados dinâmicos (Figuras 5.1 e 5.2). Para esta quantidade de dados não há desvios significativos da curva traçada com os dados reais.

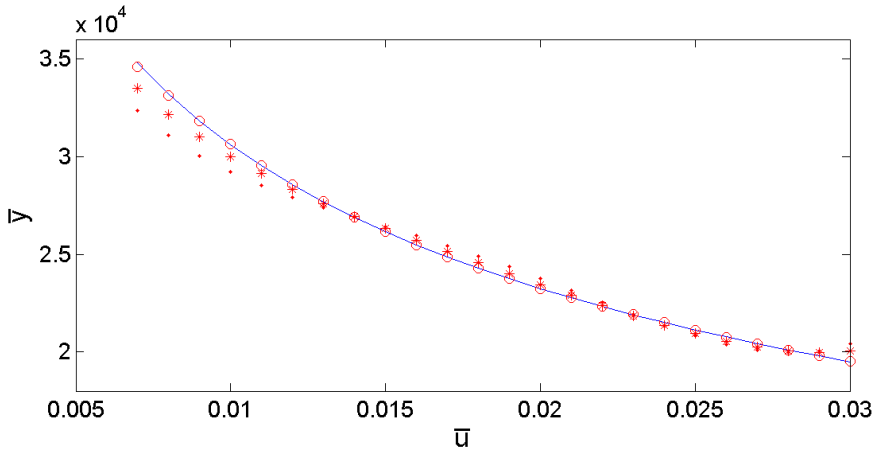

Figura 5.1: Polimerizador - Validação estática do modelo escolhido pelo decisor: (–) dados reais; (...) simulados pelo modelo com 10 pontos no Pareto; (*) dados simulados para o modelo com 100 pontos no Pareto; (o) dados simulado para o modelo com 4000 pontos no Pareto.

Portanto, aumentando a quantidade de pontos, ocorre uma melhoria significativa do modelo estimado. Uma conjectura que se pode levantar, é que o número de termos presentes no Pareto é determinante para a qualidade dos resultados. Sendo assim, é também availiada questão do tempo computacional, uma vez que quanto mais pontos no Pareto, maior será o tempo de processamento.

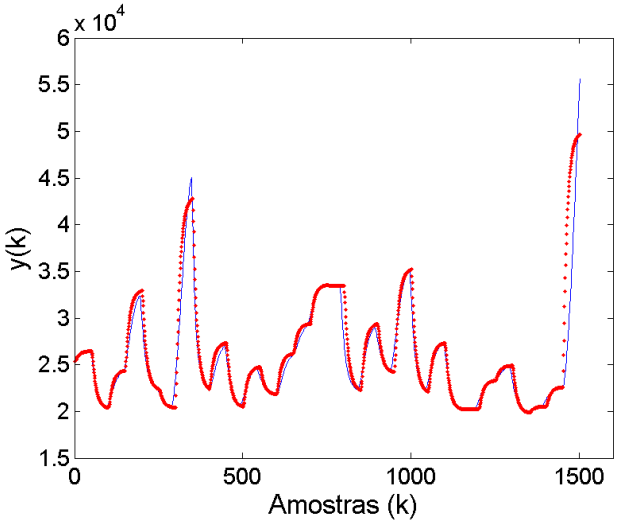

Figura 5.2: Polimerizador - Validação dinâmica do modelo escolhido pelo decisor para 4.000 pontos no Pareto-Ótimo: (–) dados reais; (...) dados estimados.

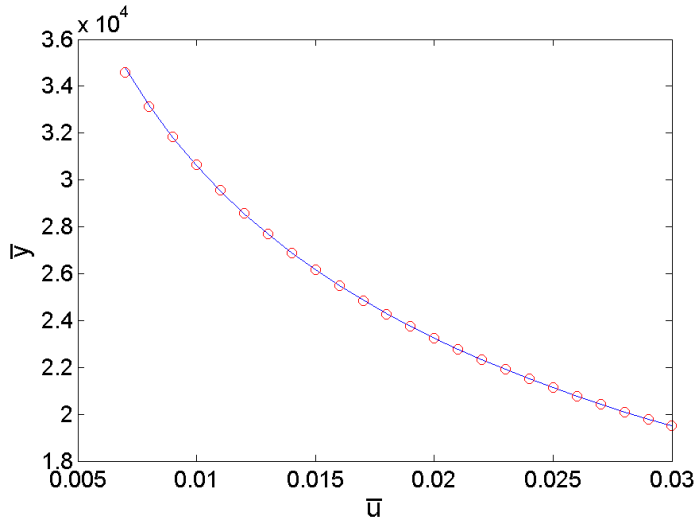

Figura 5.3: Polimerizador - Validação estática do modelo escolhido pelo decisor para 4.000 pontos no Pareto-Ótimo:(–) dados reais; (o) dados estimados.

O estimador bi-objetivo acelerado, por não gerar todo o conjunto de modelos candidatos no Pareto, apresenta um único resultado. As respostas estática e dinâmica do modelo identificado pode ser visto nas Figuras (5.4 e 5.5).

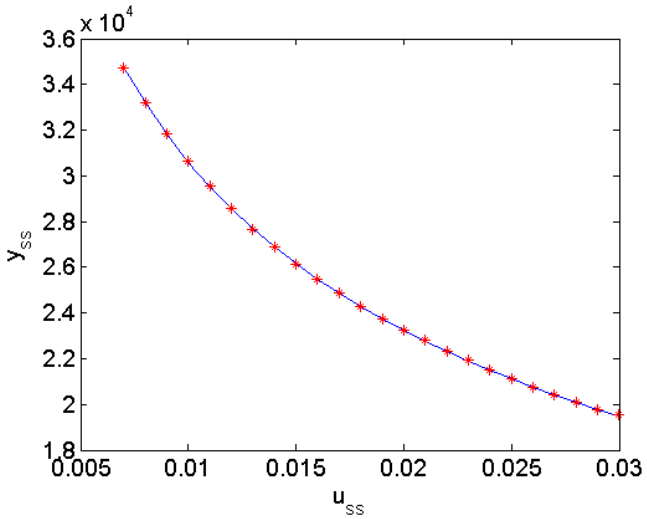

Figura 5.4: Polimerizador - Validação estática do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado: (–) dados reais; (...) dados estimados.

5.1.2 Análise do Tempo de Estimação

Para a análise dos tempos de processamento dos estimadores bi-objetivo convencional e bi-objetivo paralelo, foram realizados 19 simulações com o modelo (Equação 5.1),

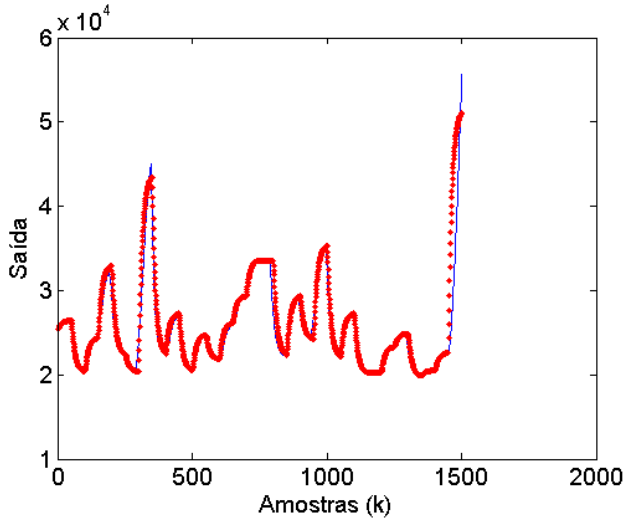

Figura 5.5: Polimerizador - Validação dinamica do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado: (–) dados reais; (o) dados estimados.

para uma quantidade de pontos variando de 1.000 a 19.000 pontos no Pareto. Para uma análise mais objetiva, serão utilizados ajustes de curva entre o número de pontos no Pareto e o seu respectivo tempo de processamento, para as duas implementações do estimador. Desta maneira, será possível, quantitativamente, obter valores para os tempos de processamento, bem como comparar de forma numérica tais dados.

As curvas obtidas para o tempo de processamento da Estimação Convencional e para a Estimação Paralela podem ser representadas respectivamente pelas funções mostrada pelas Equações (5.4 e 5.5):

$$f(x) = (1,839 \times 10^{-5})x^2 + 0,02383x + 8,347 \tag{5.4}$$

e

$$h(x) = 0,01316x - 0,1416 \tag{5.5}$$

em que x é o número de pontos no Pareto e $f(x)$ e $h(x)$ os tempo de processamento da estimação convencional e da estimação paralela.

Pelas funções $f(x)$ e $h(x)$, pode-se observar que o Estimador Paralelo estima os parâmetros em tempo afim e o Estimador Convencional estima os parâmetros em tempo quadrático. A Figura 5.6 mostra graficamente os ajustes obtidos. Este resultado é importante, pois revela de maneira não só qualitativa, mas também quantitativa, a melhoria no tempo de execução de uma estimação por meio da programação paralela do estimador bi-objetivo.

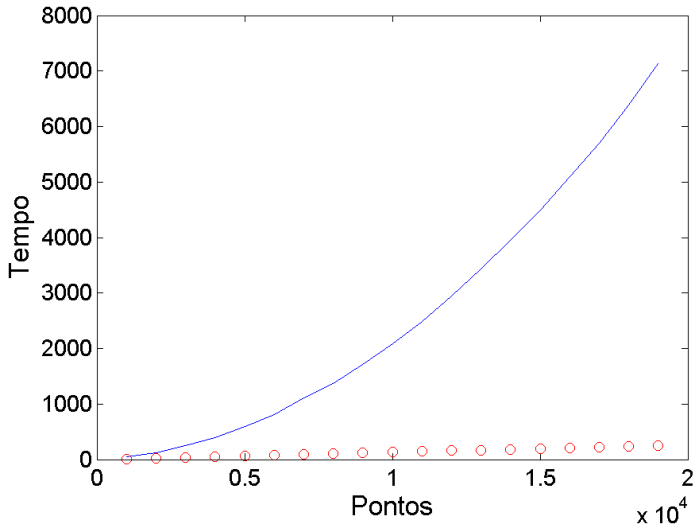

Figura 5.6: Polimerizador - Comparação entre os tempos para solução do Estimador Bi-Objetivo Convencional e do Estimador Bi-Objetivo Paralelo para a estimação do intervalo de 1.000 à 19.000 pontos: (o) Estimador Bi-Objetivo Paralelo; (–) Estimador Bi-Objetivo Convencional.

Calculando a diferença dos tempos gastos na solução dos estimadores (Tabela 5.1)(Figura 5.6), foi obtido como resultado um ganho percentual partindo de $332,8\%$ com uma taxa decrescente em torno de 150%, chegando a $2.830,3\%$ de economia em tempo. É um ganho considerável e que permite ao operador aumentar o número de pontos no Pareto para que a estimação possa convergir para um resultado confiável. Lembrando que, este caso em especial, mostra que para 100 pontos no Pareto, o modelo não foi capaz de atender de forma satisfatória os critérios de qualidade necessários, ou seja, nao encontrou um modelo que represente dinamicamente e estaticamente o sistema em estudo.

Tabela 5.1: Polimerizador - Tempos de solução para estimação Bi-Objetivo Convencional e Bi-Objetivo Paralelo

Pontos no Pareto-Ótimo	Tempo Bi-Objetivo Convencional(s)	Tempo Bi-Objetivo Paralelo(s)
1.000	467,04	14,03
10.000	2.082,45	131,28
19.000	7.126,72	251,80

Entretanto, o estimador bi-objetivo acelerado possui um mesmo tempo de solução, que no caso do polimerizador foi de: 5,294 s. Como se trata de um caso em que os dados

são não-complementares, a existência de um mínimo global faz com que a convergência do acelerador seja muito mais eficiente, pois não precisa gerar todo[1] o conjunto Pareto.

5.1.3 Validação da Correlação dos Estimadores

A forma de análise utilizada aqui é similar à utilizada na Seção 5.1.2. Foram armazenadas as correlações para cada ponto dentro do intervalo de 1.000 a 19.000 para cada estimador. O que se pretende com esta análise é mostrar que não há perda na paralelização do processo. Em outras palavras, não há perda de qualidade na estimação, mesmo com algum ganho computacional. Caso seja possível mostrar que os valores das correlações são idênticos para os dois estimadores e que as escolhas se repetiram, então, tais estimadores apresentam resultados iguais. Esta é uma informação importante, uma vez que em Barroso et al. (2007) foi mostrado que o estimador Bi-Objetivo é não polarizado (Capitulo 2).

Pode-se verificar pela Figura (5.7) que os valores das correlações para ambos estimadores são idênticos o que garante a igualdade dos resultados.

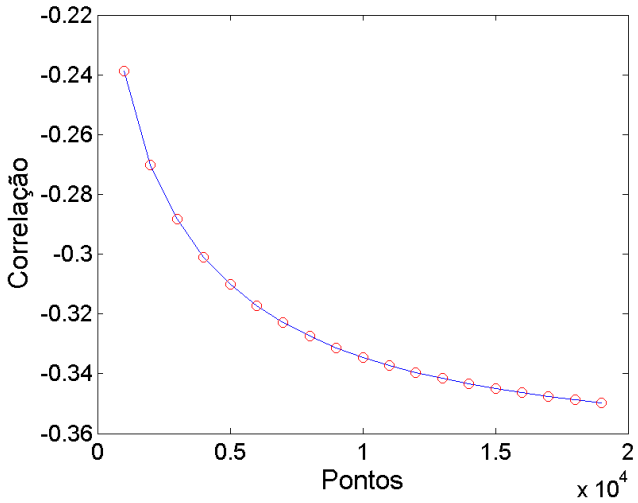

Figura 5.7: Polimerizador - Correlação dos estimadores para 1.000 a 19.000 pontos: (–) Estimador Bi-Objetivo Convencional; (o) Estimador Bi-Objetivo Paralelo.

Para uma comparação quantitativa, a mesma estratégia utilizada na Seção 5.1.2 foi utilizada. A função para as correlações dos estimadores são idênticas e é mostrada a seguir (Equação 5.6):

$$g(x) = -0,3222e^{((4,485\times10^{-6})x)} + 0,122e^{(-0,0003783x)}, \tag{5.6}$$

[1] O termo "todo" presente no texto se refere a uma quantidade significativa de pontos no Pareto.

sendo x o número de pontos no Pareto e $g(x)$ a estimativa para a correlação do modelo escolhido pelo decisor.

A função $g(x)$ revela que a correlação decresce de forma exponencial com o aumento de pontos. Porém, observa-se que existe um certo ponto de acomodação, isto é, ao aumentar a quantidade de pontos do Pareto, o valor da correlação converge para um valor constante e é exatamente este fator que faz com que o acelerador seja atraente para o caso não-complementar.

Já o estimador bi-objetivo acelerado gerou como resultado de correlação mínima o valor de $-0,42$, que é o melhor resultado comparado com a estimação bi-objetivo convencional/paralela para 50.000 pontos que gerou como resultado um valor igual a $-0,359$ (convergindo para o valor de $-0,42$).

5.1.4 Resultados Secundários

Ao comparar os estimadores bi-objetivo com o de Mínimos Quadrados Estendidos (MQE), que é também um estimador da classe dos não-polarizados, é possível verificar a importância de tais estimadores e que o ganho computacional pode torná-lo atraente para aplicações mais complexas, como por exemplo, aquelas em que os dados dinâmicos não contemplam informações suficientes para que os estimadores clássicos (como é o caso do estimador de mínimos quadrados). A Figura (5.8) mostra que o estimador MQE não foi capaz de retornar parâmetros compatíveis com as expectativas do problema.

Seguindo a tendência do modelo dinâmico, o modelo estático cujos parâmetros foram estimados por meio do MQE também não se ajusta tão bem aos dados. Conjectura-se que este fato acontece porque o MQE é um estimador que não permite a utilização de informação auxiliar não contida nos dados dinâmicos, como por exemplo, os dados estáticos. Em Aguirre (1996),Barroso (2006) e Barroso et al. (2007), esse fato foi estudado e o presente trabalho é um indicativo no mesmo sentido das conclusões daqueles trabalhos.

Esses resultados são importantes, pois revelam, não só qualitativamente, mas também quantitativamente, que no exemplo utilizado neste trabalho: (i) o número de elementos no Pareto é determinante para a melhoria da precisão do estimador; (ii) a informação contida nos dados estáticos é essencial para o equilíbrio entre polarização e variância.

A Figura (5.7) revela também que existe uma tendência para acomodação em um valor constante para correlação. Isto implica que existe um limiar de elementos no Pareto que a partir de então, não agregará melhoria significante ao estimador com inclusão de mais elementos ao mesmo. Uma vez identificada a função que melhor

ajusta a curva da correlação com a quantidade de pontos, tona-se possível saber qual é a quantidade mínima de pontos para se obter o melhor modelo. Este resultado secundário é determinante, uma vez que pode ser uma informação para se disparar ou não o processo de paralelização do estimador. Dependendo do número de elementos do Pareto, pode não ser compensador o uso de tal ferramenta, como mostra a Figura (5.6).

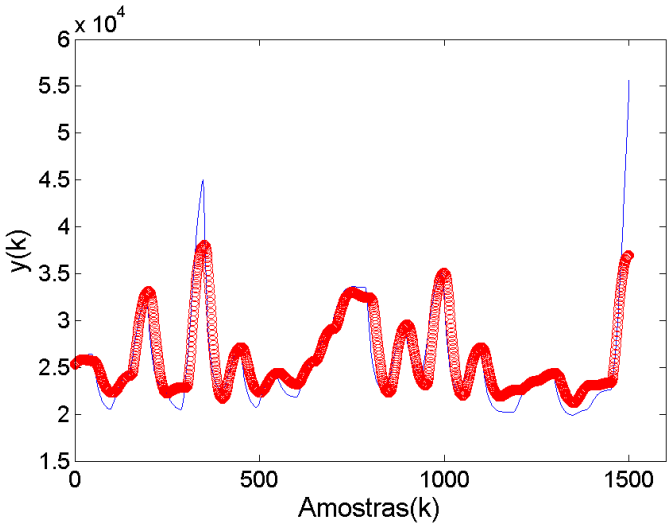

Figura 5.8: Polimerizador - Validação dinâmica do modelo por Mínimos Quadrados Estendidos: (–) dados reais; (o) dados estimados.

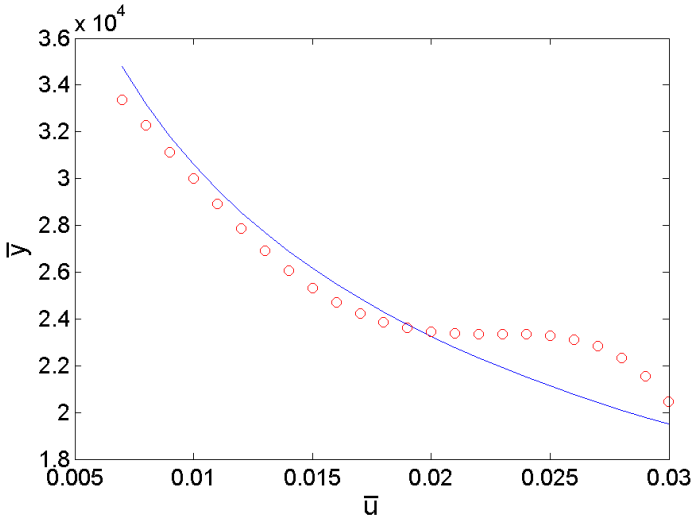

Figura 5.9: Polimerizador - Validação estática do modelo por Mínimos Quadrados Estendidos.(–) dados reais; (o) dados estimados.

5.2 Conversor estático CC-CC Buck

Esse exemplo é muito importante pois abrange uma série de dificuldades que tornam o problema de identificação ainda mais complexo. Em primeiro lugar, os dados dinâmicos são limitados a uma estreita faixa de operação por vários motivos: questões operacionais, impossibilidade de ensaios controlados, etc.. Tais dados não apresentam informação suficiente a respeito de uma larga faixa de pontos de operação de interesse do sistema. Devido a isso, a detecção de estrutura e estimação dos parâmetros, utilizando de técnicas caixa-preta, pode se tornar ineficientes. No entanto, é possível obter analiticamente a curva característica do sistema. E assim pode-se saber *a priori* como as relações em estado estacionário são descritas, sejam matematicamente, ou por ensaios específicos em estado estacionário. A possibilidade de utilizar informação *a priori* (ou auxiliar) nas etapas de detecção de estrutura e estimação dos parâmetros será considerada.

5.2.1 Estrutura do Conversor CC-CC Buck

A Figura (5.10) mostra um esquema representativo de um conversor CC-CC Buck.

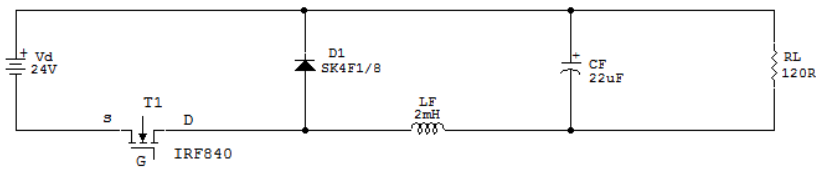

Figura 5.10: Estrutura de um conversor CC-CC Buck

Durante os testes para aquisição dos dados de identificação e validação, a fonte de tensão contínua (V_d) foi mantida constante e igual a $24V$; o sistema de regulação de carga não é mostrado na Figura (5.10). A razão cíclica é definida como a proporção de tempo em que a chave está ligada em relação ao período total de operação, T, ou, $D = T_{\text{ligado}}/T$.

Para variar essa razão, utilizaram-se técnicas de modulação por largura de pulso (PWM) a uma taxa de $1/T = 33kHz$, utilizando para tal um circuito integrado LM3524. A taxa de 33 kHz resultou em um modo de operação contínua, ou seja, a corrente através do indutor não se anulava.

Quando a razão cíclica está perto do valor unitário, a corrente através do indutor (LF) e a tensão na carga (RL), v_0, aumentam, pois a fonte v_d energiza a malha formada por ela, o capacitor (CF) e indutor (LF). Quando $D \to 0$, a tensão v_0 diminui com

um regime dinâmico diferente. Esse fato caracteriza um regime dinâmico não-linear do sistema.

A estrutura do modelo NARX polinomial utilizado nesta seção foi detectada em Barroso (2006)[2] e é apresentada na equação 5.7 a seguir:

$$y(k) = \theta_1 y(k-1) + \theta_2 y(k-1) + \theta_3 u(k-5) + \theta_4 y(k-1) + \theta_5 + \theta_6 y(k-3) \quad (5.7)$$

5.2.2 Resultados

Para o modelo (Equação 5.7) apresentado na Seção 5.2.1, foi realizada uma simulação com os estimadores bi-objetivo convencional e paralelo com 19.000 pontos no Pareto e o resultado da dinâmica pode ser vista pela Figura (5.11) e a estática pela Figura (5.12):

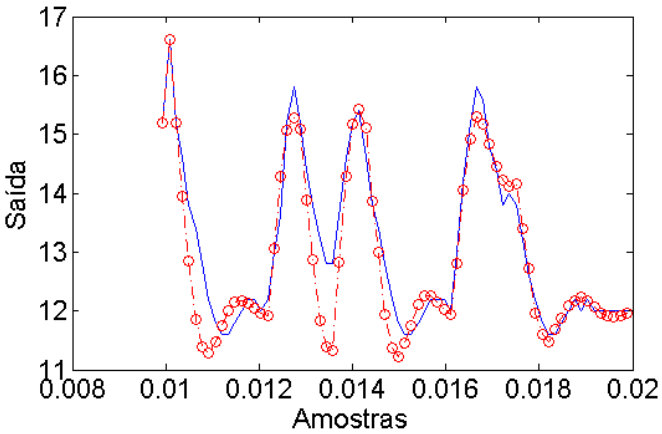

Figura 5.11: Buck - Validação dinâmica do modelo escolhido pelo decisor para 19.000 pontos no Pareto-Ótimo: (–) dados reais; (-o-) dados estimados.

[2]Este sistema é largamente usado como padrão para a verificação de problemas na estimação de parâmetros e até mesmo para o caso de detecção de estruturas. Maiores informações podem ser otidas em bibliografia contida em Barroso (2006). Não serão comparados aqui os resultados deste trabalho com aqueles apresentados naquele trabalho. O que se deseja é mostrar o caso não complementar como maneira de validar este trabalho.

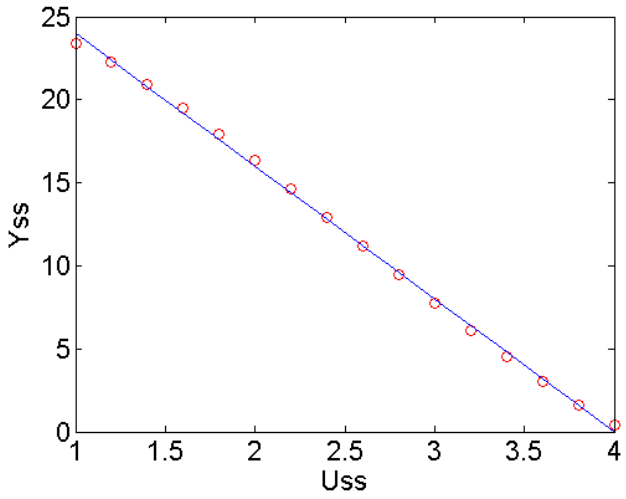

Figura 5.12: Buck - Validação estática do modelo escolhido pelo decisor para 19.000 pontos no Pareto-Ótimo: (–) dados reais; (o) dados estimados.

O estimador bi-objetivo acelerado, como foi apresentado na Seção 5.1.1 deste item não realiza o "preenchimento" do Pareto para estimação, sempre apresenta a mesma curva estática e dinâmica, conforme as Figuras (5.13, 5.14).

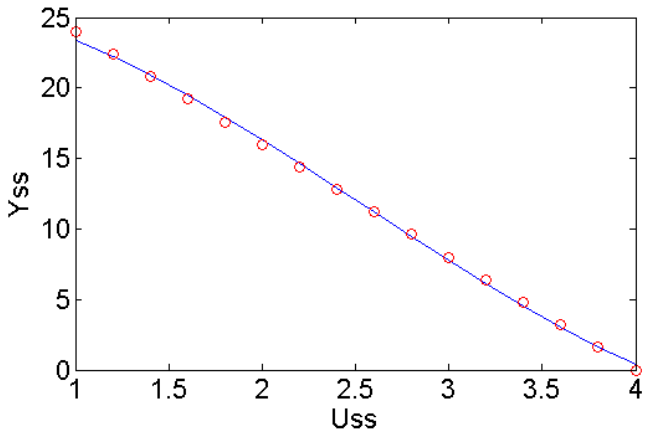

Figura 5.13: Buck - Validação estática do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado: (–) dados reais; (-o-) dados estimados.

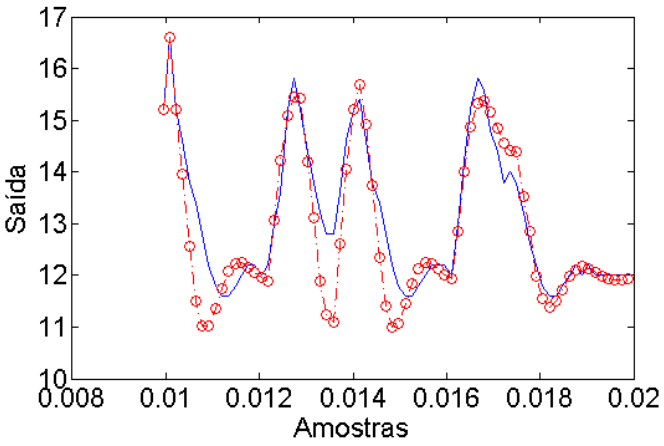

Figura 5.14: Buck - Validação dinâmica do modelo escolhido pelo decisor para o estimador bi-objetivo acelerado: (–) dados reais; (o) dados estimados.

5.2.3 Análise do Tempo de Estimação

Para a análise do tempo de processamento dos estimadores bi-objetivo convencional e bi-objetivo paralelo, foram realizados 19 testes com o modelo representado pela Equação (5.1), igualmente feito na Seção 5.1.2, com uma quantidade de pontos variando de 1.000 a 19.000 pontos no Pareto. Para uma análise mais objetiva, nesta Seção também serão utilizados ajustes de curva entre o número de pontos no Pareto e o seu respectivo tempo de processamento, para as duas implementações do estimador. Desta maneira, será possível, quantitativamente, obter valores para os tempos de processamento, bem como comparar de forma numérica tais dados.

As curvas obtidas para o tempo com a Estimação Convencional e para a Estimação Paralela podem ser representadas respectivamente pelas funções nas Equações (5.8 e 5.9), mostradas a seguir:

$$f(x) = (4,2826 \times 10^{-7})x^2 + 0,0016131x + 2,0437 \tag{5.8}$$

e

$$h(x) = 0,0021452x - 4,4384 \tag{5.9}$$

em que x é o número de pontos no Pareto e $f(x)$ e $h(x)$ os tempo de processamento das estimações convencional e paralela, respectivamente.

Como no caso anterior, pelas funções $f(x)$ e $h(x)$, pode-se observar que o Estimador Paralelo estima os parâmetros em tempo afim e o Estimador Convencional estima os parâmetros em tempo quadrático. A Figura 5.15 mostra graficamente os ajustes ob-

tidos. Este resultado mostra de maneira não só qualitativa, mas também quantitativa, a melhoria no tempo de execução de uma estimação por meio da programação paralela do estimador bi-objetivo.

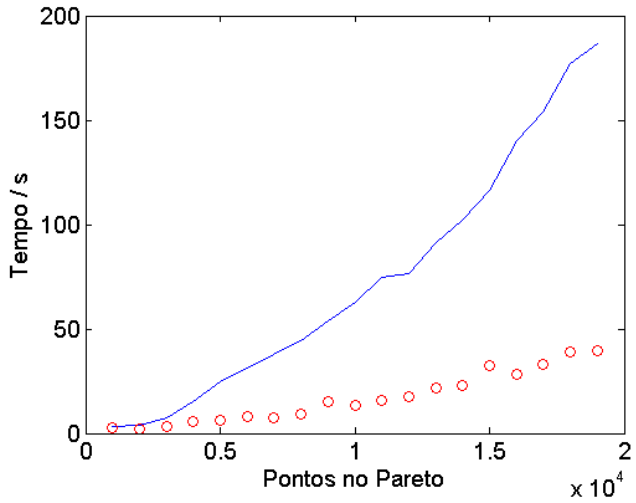

Figura 5.15: Buck - Tempos para solução do Estimador Bi-Objetivo Convencional e do Estimador Bi-Objetivo Paralelo para a estimação do intervalo de 1.000 à 19.000 pontos: (o) Estimador Bi-Objetivo Paralelo; (–) Estimador Bi-Objetivo Convencional.

Tabela 5.2: Buck - Tempos de solução para estimação Bi-Objetivo Convencional e Bi-Objetivo Paralelo

Pontos no Pareto-Ótimo	Tempo Bi-Objetivo Convencional (s)	Tempo Bi-Objetivo Paralelo (s)
1.000	2,985	2,583
10.000	62,522	13,365
19.000	186,882	39,414

Pode-se ver que o estimador bi-objetivo acelerado possui um tempo constante de solução, que no caso estudado foi de $1,969$ s.

5.2.4 Análise da Correlação dos Estimadores

A forma de análise utilizada aqui é similar à utilizada na Seção 5.1.2. Foram armazenadas as correlações para cada ponto dentro do intervalo de 1.000 à 19.000 pontos no Pareto para cada estimador. O que se pretende com esta análise é mostrar que não há perda na paralelização do processo. Em outras palavras, não há perda de qualidade

na estimação, com o ganho computacional. Caso seja possível mostrar que os valores das correlações são idênticos para os dois estimadores e que as escolhas se repetiram, então, tais estimadores apresentam resultados iguais. Esta é uma informação importante, uma vez que em Barroso et al. (2007) mostrou que o estimador Bi-Objetivo é não polarizado.

Pode-se verificar pela Figura (5.16) que os valores das correlações para ambos estimadores são idênticos. Sendo assim, pode-se conjecturar que as mudanças propostas, ou seja, o paralelismo, não afeta os cálculos efetuados.

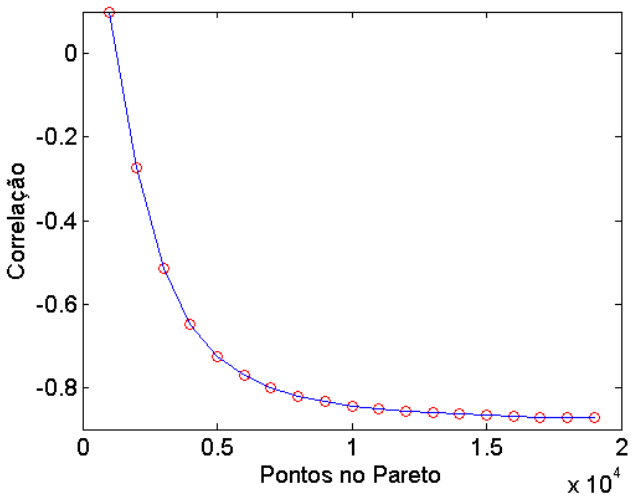

Figura 5.16: Buck - Correlação para 1.000 a 19.000 pontos: (–) Estimador Bi-Objetivo Convencional; (o) Estimador Bi-Objetivo Paralelo.

Para uma comparação quantitativa, a mesma estratégia utilizada na Seção 5.2.3 foi utilizada. A função para as correlações dos estimadores são idênticas e são mostradas a seguir (Equação 5.10):

$$g(x) = -0,8259e^{((3,097\times10^{-6})x)} + 1,573e^{(-0,0005265x)}, \tag{5.10}$$

sendo x o número de pontos no Pareto e $g(x)$ a estimativa para a correlação do modelo escolhido pelo decisor.

A função $g(x)$ revela que a correlação decresce de forma exponencial com o aumento de pontos. Porém observa-se que existe um certo ponto de acomodação, isto é, ao aumentar a quantidade de pontos do Pareto, o valor da correlação converge para um valor constante.

O estimador bi-objetivo acelerado gerou como resultado de correlação mínima o valor de $0,303998$, que é um pior resultado comparado com a estimação bi-objetivo con-

vencional/paralela para 100.000 pontos com um valor igual a $-0,885053$. Comparando a correlação obtida pelo estimador bi-objetivo acelerado com a Tabela (4.1), a correlação obtida é próxima à estimação bi-objetivo convencional/paralelo com 100 pontos no Pareto. O resultado apresentado pelo estimador bi-objetivo acelerado pode ser explicado pela presença de mínimos locais, como pode ser observado pela Tabela(4.1).

5.3 Considerações Finais

É possível afirmar que a precisão do resultado do estimador bi-objetivo acelerado é garantida somente para dados não-complementares. Pelas duas simulações realizadas (Seções 5.1, 5.2), no estudo do polimierizador o estimador bi-objetivo acelerado obteve um melhor resultado tanto em tempo quanto em correlação se comparado aos estimadores bi-objetivo convencional/paralelo. Porém no caso do conversor Buck, houve uma inversão nos resultados, os estimadores bi-objetivo convencional/paralelo apresentaram um melhor resultado.

Portanto, em casos, cujos dados utilizados na estimação são complementares, é interessante que seja utilizado o estimador bi-objetivo paralelo, pois o mesmo gera todo o Pareto para realizar a estimação, o que não ocorre com o estimador bi-objetivo acelerado. Embora este apresente um tempo de solução maior que o estimador bi-objetivo acelerado e menor que o estimador bi-objetivo convencional, a utilização de uma grande quantidade de pontos no Pareto, garantirá a qualidade do resultado.

Já nos casos, em que os dados utilizados na estimação são não-complementares, é interessante que seja utilizado o estimador bi-objetivo acelerado, pois como foi apresentado neste Capítulo, o resultado obtido para este tipo de dados com o estimador bi-objetivo acelerado foi melhor tanto na correlação e no tempo de solução em comparação com os estimadores bi-objetivo convencional/paralelo.

Capítulo 6

Considerações Finais e Trabalhos Futuros

Este trabalho foi desenvolvido com o objetivo de reduzir o tempo de processamento do estimador bi-objetivo. Para isto foram apresentadas dois métodos: estimador bi-objetivo acelerado e estimador bi-objetivo paralelo.

Também foi observado durante o desenvolvimento que o tipo de dados que está sendo utilizado, dados complementares e não-complementares, interfere diretamente no resultado, fazendo com que este seja um fator determinante na escolha do método a ser utilizado no processo de identificação.

6.1 Estimador Bi-Objetivo Acelerado

O estimador bi-objetivo acelerado, apresenta uma forma de solução sem a necessidade do preenchimento do Pareto. Por utilizar uma métrica baseada na exclusão de semi-espaços, este método apresenta uma convergência para o valor de mínimo com uma pequena quantidade de iterações, por isto, seu tempo de solução é extremamente menor que qualquer outro método apresentado neste trabalho, conforme pode ser visto no Capítulo 5.

Entretanto, foi observado no Capítulo 4 que em casos que os dados do sistema são complementar, este estimador não apresenta um bom resultado. Isto acontece devido a presença de mínimos locais dentro do conjunto de correlações, fazendo com que a convergência da mínima correlação tedencie para um mínimo local. Portanto, os resultados apresentados no Capítulo 5 comprovam de certa forma as conjecturas apresentadas no Capítulo 4.

6.1.1 Trabalhos Futuros

Como foi apresentado no Capítulo 4.1, em casos que os dados são complementares, o estimador bi-objetivo acelerado apresenta uma convergência para um mínimo local. Propõe-se como trabalho futuro desenvolver uma metodologia que não permita a convergência para uma região de atração de um mínimo local, possibilitando que o estimador possa pesquisar e convergir para o mínimo global do conjunto Pareto.

6.2 Estimador Bi-Objetivo Paralelo

O estimador bi-objetivo paralelo, foi desenvolvido sob os modelos matemáticos do estimador bi-objetivo convencional. Portanto, o mesmo trabalha com a necessidade de definição da quantidade de pontos. Estes são utilizados na geração do Pareto para realização da identificação, garantindo assim a qualidade dos resultados obtidos. Este fato, como pode ser observado no Capítulo 5, torna o estimador mais lento em relação ao tempo de processamento que o estimador bi-objetivo acelerado. Porém em comparação com o estimador bi-objetivo convencional, o mesmo apresentou um ganho de tempo de $2.830,3\%$ no estudo do polimerizador (Capítulo 5) .

A vantagem do estimador bi-objetivo paralelo sobre o estimador bi-objetivo acelerado, é que este estimador gera resultados satisfatórios tanto para dados complementares como para dados não-complementares. O fator qualitativo a favor do estimador bi-objetivo paralelo será a quantidade de pontos a ser utilizada no processo de estimação.

6.2.1 Trabalhos Futuros

Embora o estimador bi-objetivo paralelo possibilite um excelente ganho em tempo de solução em comparação ao estimador bi-objetivo convencional, o mesmo ainda possui um tempo maior em comparação com o estimador bi-objetivo acelerado. Este fato desperta o interesse de aproximar os tempos de solução. Para que isto aconteça, propõe-se para trabalhos futuros o desenvolvimento do estimador bi-objetivo paralelo com a memória distribuída, ou seja, utilizar técnicas de computação distribuída na implementação do estimador bi-objetivo paralelo.

Referências Bibliográficas

Abbasi, B.; Shadrokh, S. & Arkat, J. (2006). Bi-objective resource-constrained project scheduling with robustness and makespan criteria. *Applied Mathematics and Computation*, 80:146–152.

Abrantes, R. (2008). Aplicações de computação paralela em otimização contínua. Master's thesis, Instituto de Matemática e Estatística (IME), Universidade de São Paulo, São Paulo, SP, Brasil.

Agha, G. A. & Kim, W. (1999). Actors: A unifying model for parallel and distributed computing. *Journal of Systems Architecture*, 45(15):1263 – 1277.

Aguirre, L. A. (1996). *Introdução à Identificação de Sistemas: técnicas lineares e não-lineares aplicadas a sistemas reais.* Editora da UFMG,Belo Horizonte. 3ª edição.

Aguirre, L. A. & Billings, S. A. (2007). Improved structure selection for non-linear models based on term clustering. *International Journal of Control*, 67:772–782.

Al-Fawzana, M. & Haouarib, M. (2005). A bi-objective model for robust resource-constrained project scheduling. *International Journal Production Economics*, 96:175–187.

Andrianov, S. (2004). Role of parallel and distributed computing in beam physics. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 519(1-2):37 – 41. Proceedings of the Sixth International Conference on Charged Part icle Optics.

Araújo, F.; d'Azevedo, E. & Gray, L. (2010). Boundary-element parallel-computing algorithm for the microstructural analysis of general composites. *Computers & Structures*, 88(11-12):773 – 784.

Aström, K. J. & Witternmark, B. (1990). *Computer-controlled systems- theory and desing.* Prentice-Hall International, New York, USA.

Barroso, M. (2006). *Otimização Bi-Objetivo aplicada à estimação de parâmetros de modelos Narx Polinomiais: Caracterização e tomada de decisão*. PhD thesis, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.

Barroso, M.; Takahashi, R. & Aguirre, L. (2007). Multi-objective parameter estimation via minimal correlation criterion. *Journal of Process Control*, 17:321–332.

Berthoz, A. & Baker, R. (1978). Parallel processing in the vestibular control of eye movements. *Neuroscience*, 3(1):119 – 123.

Bhandarkar, S. M.; Machaka, S.; Chirravuri, S. & Arnold, J. (1998). Parallel computing for chromosome reconstruction via ordering of dna sequences. *Parallel Computing*, 24(8):1177 – 1204.

Billings, S. A. & Chen, S. (1989). Identification of non-linear rational systems using a prediction error estimation algorithm. *Int. J. Systems Sci*, 20(3):467–494.

Billings, S. A. & Voon, W. (1983). Structure detection and model validity tests in the identification of nonlinear systems. *IEE Proceedings, Part D*, 130(4):601–615.

Billings, S. A. & Voon, W. (1984). Least square parameter estimation algorithms for non-linear systems. *International Journal of Systems and Science*, 15(6):601–615.

Box, G. E. P. e Jenkins, G. M. (1976). *Time Series Analysics-Forecasting and Control.* Holden Day, San Francisco.

Brozolo, G. R. D. & Robert, Y. (1989). Parallel conjugate gradient-like algorithms for solving sparse nonsymmetric linear systems on a vector multiprocessor. *Parallel Computing*, 11(2):223 – 239.

Cap, C. H. & Strumpen, V. (1993). Efficient parallel computing in distributed workstation environments. *Parallel Computing*, 19(11):1221 – 1234.

Cassini, C. (1999). Estimação recursiva de características estaticas não lineares utilizando modelos polinomiais narmax. Master's thesis, Programa de Pós Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, Minas Gerais, Brasil.

Chankong, V. & Haimes, Y. (1983). Multiobjective decision making: theory and methodology. *North-Holland (Elsevier).*

Chen, S.; Billings, S. & Luo, W. (1989). Orthogonal least-squares methods and their applications to non-linear system identification. *Int. J. Control*, 50(5):1873–1896.

Clematis, A. & Corana, A. (1999). Modeling performance of heterogeneous parallel computing systems. *Parallel Computing*, 25(9):1131 – 1145.

Cámara, M.; Ortega, J. & de Toro, F. (2009). A single front genetic algorithm for parallel multi-objective optimization in dynamic environments. *Neurocomputing*, 72(16-18):3570 – 3579. Financial Engineering; Computational and Ambient Intelligence (IWANN 2007).

Coulouris, G.; Dollimore, J. & Kindberg, T. (2005). *Distributed Systems: Concepts and Design.* Harlow, England: Addison Wesley. 4th ed.

Dally, W. J. (1987). *A VLSI Architecture for Concurrent Data Structures.* Kluwer Academic Publishers, Boston, MA.

Davis, M. & Vinter, R. (1985). *Stochastic modeling and control.* Chapman and Hall, London, UK.

Doyle III, F.; Ogunnaike, B. & Pearson, R. (1995). Nonlinear modelbased control using second-order volterra models. *Automatica*, 31(5):697--714.

Ebrahim, R. M. & Razmi, J. (2009). A hybrid meta heuristic algorithm for bi-objective minimum cost flow (bmcf) problem. *Advances in Engineering Software*, 40:1056–1062.

Eusébio, A. & Figueira, J. (2009). Finding non-dominated solutions in bi-objective integer network flow problems. *Computers and Operations Research*, 36:2554–2564.

Eykhoff, P. (1981). *Trends and Progress in System Identification.* Pergamon, Oxford. In INFAC series.

Feng, T. Y. (1981). A survey of interconnection networks. *IEEE Computer*, p. 1227.

Figueira, J.; Liefooghe, A.; Talbi, E.-G. & Wierzbicki, A. (2010). A parallel multiple reference point approach for multi-objective optimization. *European Journal of Operational Research*, 205(2):390 – 400.

Flynn, M. J. (1972). Some computers organizations and their effectiveness. *IEEE Transactions on Computers*, C-21 (9):948–960.

Gatu, C. & Kontoghiorghes, E. J. (2003). Parallel algorithms for computing all possible subset regression models using the qr decomposition. *Parallel Computing*, 29(4):505 – 521. Parallel computing in numerical optimization.

Grama, A.; Gupta, A.; Karypis, G. & Kumar, V. (2003). *Introduction to Parallel Computing*. Essex, England: Harlow. 2th ed.

Greenlaw, R.; Hoover, H. J. & Ruzzo, W. L. (1995). *Limits to Parallel Computation: P-Completeness Theory*. New York, USA: Oxford University Press. 4th ed.

Gröger, T. & Zimmermann, R. (2010). Application of parallel computing to speed up chemometrics for gc × gc-tofms based metabolic fingerprinting. *Talanta*, In Press, Corrected Proof.

Guo, M.; Ho, M. S.-H. & Chang, W.-L. (2004). Fast parallel molecular solution to the dominating-set problem on massively parallel bio-computing. *Parallel Computing*, 30(9-10):1109 – 1125. High-performance parallel bio-computing.

Gustafson, K. (1999). Parallel computing forty years ago. *Mathematics and Computers in Simulation*, 51(1-2):47 – 62.

Jácome, C. (1996). Uso de conhecimento prévio na identificação de modelos polinomiais narmax. Master's thesis, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.

Kim, S. Y. & Im, Y. T. (2003). Parallel processing of 3d rigid-viscoplastic finite element analysis using domain decomposition and modified block jacobi preconditioning technique. *Journal of Materials Processing Technology*, 134(2):254 – 264.

Korenberg, M.; Billings, S.; Liu, Y. & Mcilroy, P. (1988). Orthogonal parameter estimation algorithm for non-linear stochastic systems. *Int. J. Control*, 48(1):193–210.

Kosar, T. & Livny, M. (2005). A framework for reliable and efficient data placement in distributed computing systems. *Journal of Parallel and Distributed Computing*, 65(10):1146 – 1157. Design and Performance of Networks for Super-, Cluster-, and Grid-Computing Part I.

Lastovetsky, A. & Reddy, R. (2006). Heterompi: Towards a message-passing library for heterogeneous networks of computers. *Journal of Parallel and Distributed Computing*, 66(2):197 – 220.

Leontaritis, I. J. & Billings, S. A. (1985 a). Input-output parametric models for non-linear systems - part i: deterministic non-linear system. *Int. J. Control*, 41(2):303–328.

Leontaritis, I. J. & Billings, S. A. (1985 b). Input-output parametric models for non-linear systems - part ii: sthocastic non-linear systems. *Int. J. Control*, 41(2):329–344.

Lesmere, J.; Dhaenens, C. & Talbi, E. (2007a). An exact parallel method for a bi-objective permutation flowshop problem. *Journal of Operational Research*, 177:1641–1655.

Lesmere, J.; Dhaenens, C. & Talbi, E. (2007b). Parallel partitioning method (ppm): A new exact method to solve bi-objective problems. *Computers & Operations Research*, 34:2450–2462.

Liang, Z. & Shi, W. (2010). A reputation-driven scheduler for autonomic and sustainable resource sharing in grid computing. *Journal of Parallel and Distributed Computing*, 70(2):111 – 125.

Ljung, L. (1987). *System Identification: Theory for the User*. Pretice Hall, London.

Manke, J. W. (2001). Parallel computing in aerospace. *Parallel Computing*, 27(4):329 – 336.

Marciano, J.; Silva, V. & Barroso, M. (2010a). Solução paralela para otimização do estimador bi-objetivo. In *Nono Simpósio de Mecânica Computacional*, São João del-Rei,MG.

Marciano, J.; Silva, V. & Barroso, M. (2010b). Técnica de computação paralela na estimação de parâmetros bi-objetivo. In *XVIII Congresso Brasileiro de Automática.*, Bonito, MS.

Mavrotas, G.; Figueira, J. R. & Florios, K. (2009). Solving the bi-objective multi-dimensional knapsack problem exploiting the concept of core. *Applied Mathematics and Computation*, 215:2502–2514.

Mezmaz, M.and Talbi, E. & Melab, N. (2006). Parallel cooperative meta-heuristics on the computational grid. a case study: the bi-objective flow-shop problem. *Parallel Computing*, 32:643–659.

Migdalas, A.; Toraldo, G. & Kumar, V. (2003). Nonlinear optimization and parallel computing. *Parallel Computing*, 29(4):375 – 391. Parallel computing in numerical optimization.

Miller, R. & Stout, Q. F. (1996). *Parallel Algorithms for Regular Architetures: Meshes and Pyramids.* London, England: MIT Press.

Miyano, T.; Kimoto, S.; Shibta, H.; Nakashima, K.; Ikenaga, Y. & Ailara, K. (2000). Time series analysis and prediction on complex dynamical behavior observed in a blast furnace. *Physica D*, 135:305–330.

Negro, F. T.; Ortega, J.; Ros, E.; Mota, S.; Paechter, B. & Martín, J. (2004). Psfga: Parallel processing and evolutionary computation for multiobjective optimization. *Parallel Computing*, 30:721–739.

Nepomuceno, E. G. (2002). Identificação multiobjetivo de sistemas não-lineares. Master's thesis, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.

Ni, L. M. (1991). A layered classification of parallel computers. *In Proceedings of 1991 International Conference for Young Computer Scientists.*, p. 2833.

Przybylski, A.; Gandibleux, X. & Ehrgott, M. (2006). Two phase algorithms for the bi-objective assignment problem. *European Journal of Operational Research*, 185:509–533.

Rao, A. R. M. (2009). Distributed evolutionary multi-objective mesh-partitioning algorithm for parallel finite element computations. *Computers and Structures*, 87:1461–1473.

Ray, W. (1972). On the mathematical modeling of polymerization reactors. *J.Macromol. Sci. Rev. Macromol. Chem.*, 8:1–56.

Richard, B.; Maillard, N.; Rose, C. A. D. & Novaes, R. (2005). The i-cluster cloud: distributed management of idle resources for intense computing. *Parallel Computing*, 31(8-9):813 – 838.

Rus, P.; Stok, B. & Mole, N. (2003). Parallel computing with load balancing on heterogeneous distributed systems. *Advances in Engineering Software*, 34(4):185 – 201.

Shonkwiler, R. W. & Lefton, L. (2006). *An Introduction to Parallel and Vector Scientific Computation.* Cambridge University Press, New York - USA.

Siirola, J. D. & Hauan, S. W. A. W. (2004). Computing pareto fronts using distributed agents. *Computers and Chemical Engineering*, 29:113–126.

Sorin, E. J.; Rhee, Y. M.; Nakatani, B. J. & Pande, V. S. (2003). Insights into nucleic acid conformational dynamics from massively parallel stochastic simulations. *Biophysical Journal*, 85(2):790 – 803.

Stephanopoulos, G. & Ng, C. (2000). Perspectives on the synthesis of plantwide control structudes. *Journal of rocess Control*, 10:97–111.

Sunderam, V. S. & Geist, G. A. (1999). Heterogeneous parallel and distributed computing. *Parallel Computing*, 25(13-14):1699 – 1721.

Tang, X.; Li, K.; Li, R. & Veeravalli, B. (2010). Reliability-aware scheduling strategy for heterogeneous distributed computing systems. *Journal of Parallel and Distributed Computing*, 70(9):941 – 952.

Weyer, E. & Gábor, S.and katalin, H. (2000). Grey-box modeling and identification using physical knowledge and bayesian techniques. *Automatica*, 29(2):285–308.

Wuppalapati, S.; Belegundu, A.; Aziz, A. & Agarwala, V. (2008). Multicriteria decision making with parallel clusters in structural topology optimization. *Advances in Engineering Software*, 39:416–421.

Zhang, K.; Wu, Y.-S. & Bodvarsson, G. S. (2003). Parallel computing simulation of fluid flow in the unsaturated zone of yucca mountain, nevada. *Journal of Contaminant Hydrology*, 62-63:381 – 399. JCH SI. Yucca Mountain.

Ziviani, N. (2007). *Projeto de algoritmos: com implementações em Java e C++*. São Paulo: Thomson.