



Universidade Federal de Pernambuco  
Centro de Ciências Exatas e da Natureza

Programa de Pós-Graduação em Matemática Computacional

**Regularização Geométrica Para o  
Problema da Análise de Ativação de Raios  
Gama Prontos Através de Nêutrons  
(PGNAA) em Fluidos Multifásicos**

Fabiano Barbosa Mendes da Silva

Tese de Doutorado

RECIFE  
2015

Fabiano Barbosa Mendes da Silva

**Regularização Geométrica Para o Problema da Análise de  
Ativação de Raios Gama Prontos Através de Nêutrons  
(PGNAA) em Fluidos Multifásicos**

*Trabalho apresentado ao Programa de Pós-Graduação em Matemática Computacional do Centro de Ciências Exatas e da Natureza da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Matemática Computacional.*

Orientador: *Prof. Dr. Sílvio de Barros Melo*

RECIFE  
2015

Catálogo na fonte  
Bibliotecário Jefferson Luiz Alves Nazareno CRB4-1758

S586r Silva, Fabiano Barbosa Mendes da.  
Regularização geométrica para o problema da análise de ativação de raios gama prontos através de nêutrons (PGNAA) em fluidos multifásicos./ Fabiano Barbosa Mendes da Silva – Recife: O Autor, 2015.  
105 f.: fig., tab.

Orientador: Silvio de Barros Melo.  
Tese (Doutorado) – Universidade Federal de Pernambuco. CCEN, Matemática computacional, 2015.  
Inclui referências e apêndices.

1. Matemática aplicada. 2. Matemática computacional. I. Melo, Silvio de Barros. (Orientador). II. Título.

004.0151            CDD (22. ed.)            UFPE-MEI 2015-136

FABIANO BARBOSA MENDES DA SILVA

**REGULARIZAÇÃO GEOMÉTRICA PARA O PROBLEMA DA ANÁLISE DE  
ATIVAÇÃO DE RAIOS GAMA PRONTOS ATRAVÉS DE NÊUTRONS  
(PGNA) EM FLUIDOS MULTIFÁSICOS**

Tese apresentada ao Programa de Pós- Graduação em Matemática Computacional da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Matemática Computacional.

Aprovada em: 28/08/2015.

**BANCA EXAMINADORA**

---

Prof<sup>o</sup>. Dr<sup>o</sup>. Silvio Barros de Melo (orientador)  
Universidade Federal de Pernambuco

---

Prof<sup>a</sup>. Dr<sup>a</sup>. Marcília Andrade Campos (Examinadora Interna)  
Universidade Federal de Pernambuco

---

Prof<sup>o</sup>. Dr<sup>o</sup>. Carlos Costa Dantas (Examinador Externo)  
Universidade Federal de Pernambuco

---

Prof<sup>o</sup>. Dr<sup>o</sup>. Cláudio Tadeu Cristino (Examinador Externo)  
Universidade Federal Rural de Pernambuco

---

Prof<sup>o</sup>. Dr<sup>o</sup>. Emerson Alexandre de Oliveira Lima (Examinador Externo)  
Universidade de Pernambuco

**Aos meus Pais**

# Agradecimentos

Agradeço aos meus pais, Reginaldo e Irene, pela compreensão quanto às minhas ausências nos momentos em que todos estavam reunidos, pelo apoio e carinho de sempre, e por acreditar que este seria mais um objetivo possível de ser alcançado. Faço aqui um agradecimento especial ao meu Orientador, o Professor Sílvio Melo, pelos valiosos ensinamentos, constante incentivo, determinação, paciência, compreensão, e sobretudo por acreditar no meu potencial. Agradeço também ao meu aluno Bruno Barros pela valiosíssima contribuição com os programas computacionais e a disponibilidade para aprender e ensinar. Um obrigadíssimo aos meus irmãos, familiares e amigos, por compreenderem que é necessário abdicar de bons momentos para conseguir vencer os obstáculos, mesmo que para isso seja preciso estar distante de todos; aos amigos Rui e Teófilo pelas palavras de perseverança e pelos estudos conjuntos, que não raramente invadiram as madrugadas, e ao amigo Halisson pelos materiais concedidos e discussões produtivas. Meu último agradecimento é dedicado à Universidade Federal Rural de Pernambuco - Unidade Acadêmica de Garanhuns (UFRPE - UAG), por compreender que o investimento na qualificação docente é uma das mais efetivas formas de garantir um ensino de qualidade.

# Resumo

Este estudo baseia-se no problema apresentado em Meric *et al.* [25], que consiste na utilização da Análise de Ativação de Raios Gama Prontos Através de Nêutrons (PGNAA), para a determinação das frações em peso das principais substâncias que compõem um fluido multifásico [3] obtido em processos de extração de petróleo submarino. Esse método envolve um problema mal-posto e tem suscitado o desenvolvimento de diferentes abordagens para a sua resolução. Levando-se em consideração os espectros obtidos da PGNAA para as substâncias desse fluido, e ainda estudos recentes, foi desenvolvido neste trabalho um processo de geometrização cuja ideia central é a representação das fases (substâncias presentes no fluido) através de vértices de triângulos ou de tetraedros, respectivamente para o problema com três e quatro fases. Para tal, foi necessária a utilização de importantes conceitos e ferramentas matemáticas, como o método dos mínimos quadrados e a regularização de Tikhonov, com ênfase em suas características geométricas. Foi ainda desenvolvido um caso especial do método de regularização de Tikhonov: a Regularização Geométrica. Esta nova técnica baseia-se no processo de geometrização proposto e determina um modelo de referência para os casos com três e quatro fases, além de promover mudanças de escalas (normalização de dados) e otimização do condicionamento para todas as escolhas de intervalos de coordenadas. Com o objetivo de realizar alguns testes para verificar a eficiência da regularização geométrica, foi gerada uma nova substância a partir das já existentes, com o auxílio de alguns códigos computacionais. As soluções encontradas foram representadas graficamente (caso com três fases), juntamente com os espectros das substâncias. Por fim, foram determinados os limites do *background* para o caso com três fases, estabelecendo-se analiticamente uma previsão para sua localização.

Palavras-chave: Regularização Geométrica. Geometrização. PGNAA. Fluidos Multifásicos.

# Abstract

This study is based on the problem presented in Meric *et al.* [25], about the use of Prompt Gamma-Ray Neutron Activation Analysis (PGNAA) for determining the weight fractions of a multiphase flow [3] obtained in offshore oil extraction processes. This method involves an ill-posed problem and has enabled the development of different approaches to its resolution. Based both on the spectra of substances in that fluid and recent studies, it was development in this work a geometric process whose central idea is to represent the phases (substances present in the fluid) through the vertices of triangles or tetrahedrons, respectively to the problem with three or four phases. To this end, the use of important mathematical concepts and tools were needed, such as the least squares method and Tikhonov regularization, with emphasis on its geometric characteristics. It was presented a special case of Tikhonov regularization method: the Geometric Regularization. This new technique is based on the proposed geometric process and determines a reference model to the cases of three and four fases, as well as promoting scale changing (data normalization) and condition number optimization for all coordinate ranges of choices. In order to perform some tests to verify the efficiency of geometric regularization, it was generated a new substance from the other existing, aided by some computational codes. The solutions were plotted (three phase case) together with spectra of substances. Finally, the limits of background were calculated to the three phases case, establishing a preview for its location.

Keywords: Geometric Regularization. Geometrization. PGNAA. Multiphase Flow.

# Lista de Figuras

2.1	Configuração padrão de um feixe de raios x em um <i>synchrotron</i> . . . . .	20
2.2	Solução exata e outras soluções usando a Regularização de Tikhonov. Problema do modelo de mapeamento da gravidade 1-D. . . . .	23
2.3	O vetor $g$ é mais próximo de $A\hat{f}$ do que de $Af$ para qualquer outro $f$ . . . . .	24
2.4	Esquema das regiões de modelos equivalentes e de soluções. . . . .	28
2.5	Efeito do ruído em sistemas lineares compatíveis. . . . .	30
2.6	Efeito do ruído em problemas lineares. . . . .	31
2.7	Interpretação geométrica para a regularização. . . . .	33
2.8	Efeito da regularização em problemas lineares. . . . .	34
2.9	Efeito da regularização e ruído. . . . .	35
2.10	Efeito do ruído em soluções regularizadas. . . . .	35
3.1	Esquema de geração das bibliotecas elementares de raios gama prontos para MCLLS. . . . .	41
3.2	Espectros das bibliotecas de raios gama prontos de óleo, sal, água, gás e <i>background</i> . . . . .	42
4.1	Esquema de obtenção de coordenadas - Problema com três fases . . . . .	44
4.2	Lista de dados de contagem referentes aos espectros do fluido multifase . . . . .	44
4.3	Triângulo de vértices $A5$ , $C5$ e $D5$ . . . . .	46
4.4	Triângulo de vértices $A2$ , $C2$ e $D2$ . . . . .	47
4.5	Esquema de obtenção de coordenadas - Problema com quatro fases. . . . .	48
4.6	Triângulo de vértices $A5n$ , $C5n$ e $D5n$ . . . . .	50
4.7	Recorte do código <code>analise_substancia_3.py</code> . . . . .	51
4.8	Triângulo resultante de normalização e otimização de dados. . . . .	52
4.9	Espectros das substâncias A, C, D e P. . . . .	54

4.10	Regularização Geométrica - Triângulos $\times$ Valor de $\lambda$ . . . . .	55
4.11	Plano, triângulo, substância desconhecida e solução de mínimos quadrados. Regularização geométrica com $\lambda = 0.1$ . . . . .	55
4.12	Espectros da substância desconhecida para diferentes escolhas do parâmetro de regularização . . . . .	56
4.13	Esquema de localização do <i>background</i> para o problema com três fases. . . . .	56
4.14	Esquema da região de localização do <i>background</i> para o problema com três fases. . .	58
4.15	Triângulo ótimo e representação da região do <i>background</i> - Problema com três fases.	58
4.16	Recorte do código <code>analise_substancias4.py</code> - Problema com quatro fases. . . . .	60
5.1	Análise do Condicionamento - Problema com três fases. . . . .	63
5.2	Condicionamento antes e depois da normalização - Problema com três fases. . . . .	63
5.3	Análise do Condicionamento - Problema com quatro fases. . . . .	65
5.4	Condicionamento antes e depois da regularização - Problema com quatro fases. . . .	65
5.5	Condicionamento com Regularização Geométrica - Problema com três fases. . . . .	66

# Lista de Tabelas

3.1	Composição elementar de óleo, gás, água e sal, do fluido estudado. . . . .	40
3.2	Características da fonte nêutrons de fissão espontânea $^{252}\text{Cf}$ . . . . .	40
3.3	Frações em peso resultantes dos elementos da mistura de óleo (40%), água salgada (40%) e gás (20%). . . . .	41
A.1	Lista de dados de contagem referentes aos espectros do fluido multifásico . . . . .	72
A.2	Continuação - Planilha de dados . . . . .	73
A.3	Continuação - Planilha de dados . . . . .	74
A.4	Continuação - Planilha de dados . . . . .	75
A.5	Continuação - Planilha de dados . . . . .	76

# Sumário

<b>1</b>	<b>Introdução</b>	<b>14</b>
1.1	Motivação . . . . .	14
1.2	Objetivos . . . . .	15
1.3	Organização . . . . .	15
1.3.1	Capítulo 2 - Problemas Inversos e Regularização de Tikhonov . . . . .	15
1.3.2	Capítulo 3 - A PGNAA em Fluidos Multifásicos . . . . .	16
1.3.3	Capítulo 4 - Geometrização do Problema da PGNAA em Fluidos Multifásicos	16
1.3.4	Capítulo 5 - Discussão dos Resultados . . . . .	16
1.3.5	Capítulo 6 - Conclusões e Trabalhos Futuros . . . . .	16
1.3.6	Apêndice . . . . .	17
<b>2</b>	<b>Problemas Inversos e Regularização de Tikhonov</b>	<b>18</b>
2.1	Problemas Inversos . . . . .	18
2.2	Regularização de Tikhonov . . . . .	21
2.3	A Regularização de Tikhonov numa Perspectiva Geométrica . . . . .	23
2.3.1	Método dos Mínimos Quadrados . . . . .	23
2.3.2	Efeitos do ruído em problemas inversos . . . . .	25
2.3.3	Efeitos da regularização de Tikhonov em problemas lineares inversos . . . . .	31
<b>3</b>	<b>A PGNAA em Fluidos Multifásicos</b>	<b>37</b>
3.1	Pressupostos . . . . .	37
3.2	O Método MCLS . . . . .	38
3.2.1	O mau-condicionamento da matriz de covariância . . . . .	39

3.2.2	O problema da medida em fluidos multifásicos . . . . .	40
<b>4</b>	<b>Geometrização do Problema da PGNAA em Fluidos Multifásicos</b>	<b>43</b>
4.1	Geometrização para o problema do fluido com três fases . . . . .	43
4.2	Geometrização para o problema do fluido com quatro fases . . . . .	47
4.3	Regularização Geométrica . . . . .	48
4.3.1	Regularização geométrica para o problema com três fases . . . . .	49
4.3.2	Estimativa do <i>background</i> . . . . .	56
4.3.3	Regularização geométrica para o problema com quatro fases . . . . .	58
<b>5</b>	<b>Discussão dos Resultados</b>	<b>62</b>
<b>6</b>	<b>Conclusão e Trabalhos Futuros</b>	<b>67</b>
	<b>Referências</b>	<b>69</b>
	<b>Apêndice</b>	<b>71</b>
<b>A</b>	<b>Lista de dados de contagem referentes aos espectros do fluido multifásico</b>	<b>72</b>
<b>B</b>	<b>Análise de Substâncias - Problema com três fases</b>	<b>77</b>
<b>C</b>	<b>Análise de substâncias - Problema com quatro fases</b>	<b>82</b>
<b>D</b>	<b>Gerador da quarta substância</b>	<b>86</b>
<b>E</b>	<b>Lista de dados normalizados - Problema de três fases</b>	<b>92</b>
<b>F</b>	<b>Lista de dados normalizados - Problema com quatro fases</b>	<b>99</b>

# 1

## Introdução

Problemas inversos têm sido amplamente estudados em diversos campos das ciências, tais como sensoriamento remoto, tomografia computadorizada, imagens médicas, estatística e geofísica, por exemplo, pois consistem em uma estrutura que em geral é usada para converter medidas observadas em informações sobre um objeto físico ou algum sistema de interesse.

O problema inverso considerado neste estudo tem origem na Análise de Ativação de Raios Gama Prontos Através de Nêutrons (PGNAA) de um fluido multifásico obtido em processos de extração de petróleo submarino. Tomando como base estudos recentes sobre a determinação das quantidades de substâncias que compõem esse fluido, este trabalho propõe um modelo geométrico para a descrição da PGNAA além de um novo método de regularização, baseado nessa geometrização do problema.

Neste capítulo, serão apresentados uma breve descrição das principais ideias que motivaram a realização deste estudo, os objetivos principais, e ainda um esquema de organização geral dos assuntos abordados nos capítulos subsequentes.

### 1.1 Motivação

Imagine-se que alguém esteja em um aeroporto e que a sua bagagem precise passar por um aparelho de raios x, para que os funcionários façam um "check up" dos objetos presentes em seu interior. De acordo com as informações advindas dos dados ali recolhidos, é possível estabelecer uma rápida comparação entre eles e um banco de dados que possui as características das principais substâncias que compõem os objetos mais comuns, evidenciando instantaneamente aqueles conhecidos e ainda os que possuem uma composição diferente, fato que contribui para o descobrimento de substâncias ilícitas e de outros produtos proibidos no embarque. De certo modo, através da proporção de determinadas substâncias, é possível "prever" a composição de alguns desses objetos e identificá-los, de acordo com os dados.

O problema aqui estudado segue o mesmo princípio: imagine-se que no processo de exploração de um poço de petróleo submarino seja retirado um fluido composto por diversas substâncias, tais como óleo, gás e água salgada, dentre outras. Como é possível estimar as quantidades dessas substâncias numa amostra desse fluido, quer seja em termos percentuais, quer em frações de peso?

Experimentos de diversas naturezas têm sido realizados com o objetivo de responder a esse questionamento, mas os de interesse neste estudo são aqueles em que a ideia principal é submeter os objetos em análise a uma radiação, semelhantemente aos procedimentos realizados pelas máquinas de raios x dos aeroportos. Meric *et al.* [25] propuseram um método computacional para resolver esse problema, tratando também os casos de mau-condicionamento do Método dos Mínimos Quadrados de Análise de Componentes Conjugado com o Monte Carlo (Monte Carlo Library Least Squares) para a Análise de Ativação de Raios Gama Prontos Através de Nêutrons (*Prompt Gamma Ray Neutron Activation Analysis - PGNAA*) para um fluido multifásico obtido na exploração de petróleo submarino.

Considerando-se ainda o estudo citado, observa-se que a determinação de soluções não é trivial, posto que se trata de um problema inverso em que a matriz do modelo apresenta alto grau de mau-condicionamento. Isto motivou o desenvolvimento de um caso especial da regularização de Tikhonov [33] a partir da realização de um estudo sobre as características geométricas do problema e da criação de um processo de geometrização, tanto para o caso do fluido com três fases (substâncias) quanto para o caso com quatro fases.

## 1.2 Objetivos

Os principais objetivos deste estudo são:

1. Propor uma interpretação geométrica do problema de medida do fluido multifásico abordado em Meric *et al.* [25], adaptado inicialmente ao caso com três fases, e em seguida do original, com quatro fases.
2. Desenvolver um novo método de regularização, baseado na geometrização do problema estudado.
3. Identificar os intervalos de coordenadas que representam soluções de melhor condicionamento, após a aplicação do método de regularização proposto.
4. Representar geometricamente o problema com três fases, com base na escolha dos intervalos de coordenadas, relativamente às partições concebidas no processo de geometrização.
5. Determinar os limites (e conseqüentemente uma região) para o *background*, em função das escolhas de intervalos de coordenadas, para o problema com três fases.

## 1.3 Organização

Uma breve descrição dos conteúdos que compõem cada capítulo desta Tese é apresentada a seguir:

### 1.3.1 Capítulo 2 - Problemas Inversos e Regularização de Tikhonov

Neste capítulo, as ideias iniciais e os principais conceitos sobre regularização são trazidos à tona, objetivando a familiarização com os problemas inversos e com o mais conhecido método de regu-

larização: a regularização de Tikhonov. O processo de Regularização Geométrica proposto neste estudo constitui-se num caso particular desse método, o que justifica uma exposição detalhada das suas principais ferramentas, como a determinação de solução de mínimos quadrados, e as suas características geométricas. Destacam-se ainda estudos recentes sobre os efeitos do ruído e da regularização de Tikhonov nos problemas lineares inversos, com foco na abordagem geométrica desses efeitos.

### 1.3.2 Capítulo 3 - A PGNAA em Fluidos Multifásicos

O terceiro capítulo é consagrado à Análise de Ativação de Raios Gama Prontos Através de Nêutrons, realizando-se uma explanação do problema de medida do fluido multifásico obtido da exploração de petróleo submarino, com ênfase nas suas principais características e nas ideias consideradas para a determinação das quantidades das principais substâncias que o compõem, com base nos estudos de Meric *et al.* [25], e de Wang *et al.* [35]. Os espectros das bibliotecas dessas substâncias, juntamente com a lista de dados de contagem dos níveis de energia por substância e canal, servirão de base para o processo de geometrização proposto, apresentado no decorrer do estudo.

### 1.3.3 Capítulo 4 - Geometrização do Problema da PGNAA em Fluidos Multifásicos

Este capítulo descreve o processo de geometrização da PGNAA para o fluido multifásico apresentado no capítulo anterior, em duas partes: a primeira consiste na geometrização para o caso com três fases, em que os vetores são considerados como pontos no espaço  $\mathbb{R}^3$  e possibilitam a representação geométrica do problema em função das substâncias e dos intervalos de coordenadas escolhidos, e a segunda parte refere-se ao caso com quatro fases, utilizando uma abordagem análoga à realizada na primeira parte. É ainda apresentada a Regularização Geométrica e, especificamente para o caso com três substâncias, são determinados analiticamente os limites do *background* em função dos vértices do triângulo escolhido e da substância desconhecida.

### 1.3.4 Capítulo 5 - Discussão dos Resultados

Neste capítulo são apresentadas algumas discussões sobre os resultados obtidos, evidenciando-se as vantagens de utilização da Regularização Geométrica para o problema com três e quatro fases, com o auxílio de códigos computacionais, tabelas e figuras.

### 1.3.5 Capítulo 6 - Conclusões e Trabalhos Futuros

Através de uma breve abordagem, os principais objetivos deste estudo são novamente referidos para que se possa constatar que os métodos e processos propostos foram fundamentais para o seu alcance. Algumas novas ideias que surgiram durante esse processo mas não foram executadas, são aqui evidenciadas, constituindo uma lista de trabalhos que podem ser desenvolvidos futuramente.

### **1.3.6 Apêndice**

Este capítulo é composto pelas tabelas que apresentam as listas de dados, e pelos códigos computacionais desenvolvidos neste estudo.

## 2

# Problemas Inversos e Regularização de Tikhonov

Em diversas áreas do conhecimento, é possível observar a existência de problemas que envolvem a ideia de determinar uma causa (desconhecida) para um efeito observado ou medido, fato que sugere inúmeras aplicações, como por exemplo, em medicina, com as imagens obtidas em exames de tomografia computadorizada, eletrocardiograma, e ultrassom; em geociências, com a detecção de petróleo, gás e de outras riquezas minerais no subsolo; em física, com problemas envolvendo difusão, condução de calor, espalhamento de partículas e ondas, além de aplicações nas engenharias, dentre outras. Este percurso contribuiu para a formulação de um campo de estudos em matemática, denominado de problemas inversos.

## 2.1 Problemas Inversos

Diversos modelos matemáticos produzem resultados muito variados a partir de uma pequena mudança nos dados de entrada. Os problemas inversos, em particular, apresentam frequentemente esse comportamento, exibindo uma alta sensibilidade a pequenas mudanças nesses dados. Em geral muitos problemas físicos (alguns deles citados acima) são sensíveis a essas mudanças, sendo portanto problemas inversos, e podem ser expressos como um sistema de equações, tal como

$$Af = g, \tag{2.1}$$

para um dado efeito  $g$  (dados de saída) e uma causa a ser determinada  $f$  (dados de entrada), que pertence a um certo conjunto de dados admissíveis. Essa equação é gerada a partir de um modelo matematicamente determinado (ou a determinar)  $A$ , que em geral é um operador definido entre espaços de Hilbert  $\mathbb{X}$  e  $\mathbb{Y}$ . Para uma grande quantidade de aplicações, o operador  $A$  é linear e a teoria é amplamente desenvolvida, como por exemplo, para equações integrais da primeira espécie [18]. É possível, entretanto, que o operador  $A$  seja não-linear, e desse modo a formulação e o correspondente tratamento para a obtenção da solução tornam-se muito mais complicados [10].

Um problema inverso relativo a (2.1) consiste em determinar  $f$  a partir do operador inverso  $A^{-1}$ , caso ele exista, tal que  $f = A^{-1}g$ . A título de ilustração, são apresentados alguns exemplos de problemas lineares inversos, a seguir:

**Exemplo 2.1** Determinar um polinômio  $p(x)$  de grau  $n - 1$ , com os valores dados  $y_1, \dots, y_n$ , em  $x_1, \dots, x_n$ .

O problema direto consiste em encontrar os valores  $y_1, \dots, y_n$ , de um dado polinômio  $p(x)$ , em  $x_1, \dots, x_n$ . O problema inverso é chamado de problema da interpolação de Lagrange e o problema direto é o de determinação dos valores do polinômio.

**Exemplo 2.2** Dada uma matriz real e simétrica  $A$  de ordem  $n$ , e  $n$  números reais  $\lambda_1, \dots, \lambda_n$ , encontrar uma matriz diagonal  $D$  tal que  $A + D$  tenha os autovalores  $\lambda_1, \dots, \lambda_n$ .

No Exemplo 2.2, o problema direto correspondente consiste em determinar os autovalores  $\lambda_1, \dots, \lambda_n$  de uma dada matriz real e simétrica,  $A + D$ .

Embora considere-se que a estrutura geral de um problema inverso seja relativamente simples, nem sempre a determinação de soluções também o é. É importante então realizar uma análise sobre existência, unicidade e estabilidade de soluções, pois se a função  $g$  não pertencer ao espaço imagem do operador, então (2.1) não possui solução; além disso, se  $A$  é não-injetor podem existir muitas soluções. Para que se tenha estabilidade no sistema, a solução deve depender continuamente dos dados. Essas três condições constituem a base para a definição de problema bem-posto (bem-condicionado), proposta por Hadamard [19], em que ele considera que um problema é bem-posto se a sua solução existe, é única e é estável, ou seja, para dado  $g$  existe uma solução apropriada  $f$  que é única e depende continuamente dos dados (o operador  $A^{-1}$  é contínuo).

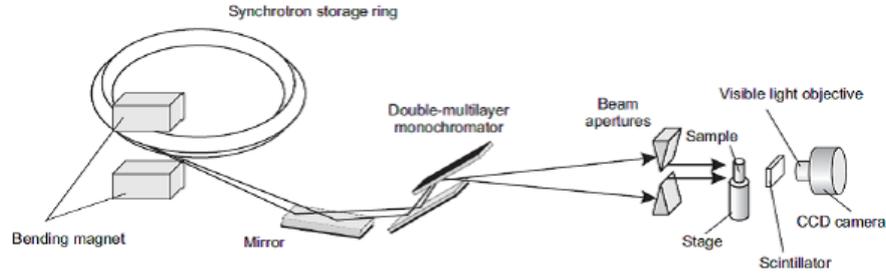
No caso de uma das três condições ser violada, obtém-se um problema mal-posto (mal-condicionado), isto é, um problema em que a solução não é única ou não é uma função contínua dos dados, e assim um pequeno ruído nesses dados pode causar uma grande perturbação da solução. Em geral, problemas inversos são mal-postos.

Para as duas primeiras condições é relativamente simples evitar a sua violação. Por exemplo, se uma solução não existe, pode-se aumentar o espaço solução de modo que seja possível determinar uma existência nesse espaço. No caso da unicidade, sua ausência indica que podem faltar informações no modelo; é possível então adicionar propriedades, impor condições ao operador ou ainda escolher, dentre as várias soluções disponíveis, qual é a que melhor corresponde ao modelo. O caso da estabilidade é mais complicado, pois não é possível utilizar artifícios matemáticos para fazer com que um problema se torne estável.

Esses conceitos foram introduzidos, pela primeira vez, no campo das equações diferenciais parciais. Para Hadamard os problemas mal-postos tinham interesse apenas acadêmico, pois considerava que eles eram artificiais e que não descreviam sistemas físicos, segundo Monteiro [28]. Apenas com o desenvolvimento dos sonares e radares, observou-se que os sistemas físicos associados a esses equipamentos podiam também ser descritos por problemas mal-postos, o que caracterizou um novo campo de aplicação desse tipo de problema, principalmente nas engenharias e na matemática. O exemplo a seguir tem como objetivo ilustrar essas afirmações:

**Exemplo 2.3** Tomografia computadorizada de raios  $x$ .

A Figura 2.1 representa um experimento padrão de radiação em um *synchrotron* [32]. Nele, há uma amostra sendo rotacionada em torno do eixo vertical  $z$ ,  $\theta$  é o ângulo de rotação e  $x$  é a

Figura 2.1: Configuração padrão de um feixe de raios x em um *synchrotron*.

Fonte: Titarenko *et al.* [32].

coordenada horizontal. Suponha que um feixe de raios monocromáticos paralelos incide sobre a amostra, e que o feixe é perpendicular a  $x$  e  $y$  e é escolhido um corte horizontal (fatia) da amostra (por simplicidade,  $z = 0$ ). Cada raio  $x$  do feixe está dentro ou é paralelo ao corte, ou seja, nenhum raio  $x$  intersecta esse corte. As intensidades do feixe de raios  $x$  antes e depois da amostra são denotadas por  $I_0(x)$  e  $I(\theta, x)$ , respectivamente. Portanto,  $I(\theta, x) = I_0(x)e^{-p(\theta, x)}$ , onde  $p(\theta, x)$  é um fator de atenuação. Se  $\xi(x, y)$  é o coeficiente de atenuação do corte, então  $p(\theta, x)$  é a transformada de Radon de  $\xi(x, y)$ :

$$p(\theta, x) = \int_{-\infty}^{+\infty} \xi(x \cos \theta - y \sin \theta, x \sin \theta + y \cos \theta) dy. \quad (2.2)$$

O problema inverso associado à tomografia é encontrar uma aproximação adequada para a distribuição de densidade  $f$  através das medidas dos  $I(\theta, x)$  de várias seções transversais em diferentes ângulos da região. Nesse sentido, encontrar  $\xi(x, y)$  configura-se num problema instável, e por isso mal-posto.

Alguns problemas discretos de dimensão finita possuem propriedades similares aos mal-postos, como por exemplo, a alta sensibilidade a perturbações de alta frequência, e são por isso chamados de problemas mal-postos discretos [21]. Para ilustrar esse fato, o sistema de equações lineares (2.1)

$$Af = g, \quad A \in \mathbb{R}^{n \times n},$$

e um problema linear de mínimos quadrados

$$\min_f \|Af - g\|_2, \quad A \in \mathbb{R}^{m \times n}, \quad m > n, \quad (2.3)$$

em que

$$\|Af - g\|_2 = \sqrt{\sum_i (Af_i - g_i)^2}, \quad (2.4)$$

refere-se à norma Euclidiana, podem constituir um problema discreto mal-posto, desde que ambos os critérios a seguir sejam satisfeitos:

1. Os valores singulares de  $A$  decaem gradualmente a zero.
2. A razão entre o maior ( $\mu_1$ ) e o menor ( $\mu_n$ ) valor singular não-nulo de  $A$  é bastante grande.

Esta razão é denominada de condicionamento da matriz  $A$ , e define-se como [16], [29], [9]:

$$\kappa(A) = \frac{\mu_1}{\mu_n}, \quad (2.5)$$

em que  $\mu_1 > \mu_2 > \dots > \mu_n$ , para valores crescentes de  $n$ . Quanto mais próximo de 1.0 for o valor de  $\kappa$ , mais bem condicionada será a matriz.

Diante desses exemplos, é possível fazer o seguinte questionamento: O que significa resolver um problema mal-posto? Em resposta a esta questão, Tikhonov [33] definiu um método que consiste na determinação de uma solução aproximada para a matriz de covariância. A busca da solução mais suave (regular) passa pelo acréscimo de uma informação adicional, que tem o papel de melhorar o condicionamento do problema.

Para um problema mal-posto em que a condição violada é a de estabilidade, a determinação da solução numérica por métodos "convencionais" é bastante difícil e produz instabilidade, mesmo que os dados sejam exatos. Assim, os métodos de regularização têm sido usados para obter uma aproximação estável da solução.

Retome-se a equação (2.1). Uma forma natural para determinar uma solução é aplicação do operador inverso  $A^{-1}$  mas, na prática, tal operador é frequentemente não-invertível ou então, se possui uma inversa, ela é limitada, fato que analiticamente não garante a dependência contínua dos dados, o que possibilita a produção de soluções inadequadas e caracteriza um problema mal-posto. Uma regularização define-se como:

**Definição 2.4** *Um método de regularização consiste em uma família de operadores  $R_\alpha : \mathbb{Y} \rightarrow \mathbb{X}$ , contínuos e não necessariamente lineares, em que os conjuntos  $\mathbb{X}$  e  $\mathbb{Y}$  são em geral espaços de Banach.*

Esses operadores aproximam a inversa generalizada de  $A$  [5], denotada por  $A^\dagger$ . O método de regularização depende ainda de uma regra para escolha de parâmetros  $\alpha$ , a qual pode ser feita *a-priori*, isto é, quando  $\alpha$  depende apenas do nível de ruído ( $\epsilon$ ), e *a-posteriori*, em que  $\alpha$  depende do nível de ruído e do dado com ruído ( $g_\epsilon$ ), consoante com o método de regularização utilizado [7]. Escolhendo-se  $\alpha$ , a solução regularizada  $f_\alpha^\epsilon$  converge em norma para a solução de (2.1) quando  $\epsilon \rightarrow 0$ , isto é, quando o erro no pior caso [10] satisfaz

$$\limsup_{\epsilon \rightarrow 0} \{ \|R_{\alpha(\epsilon)} g_\epsilon - A^\dagger g\| : g_\epsilon \in \mathbb{Y}, \|g_\epsilon - g\| \leq \epsilon \} = 0. \quad (2.6)$$

Pode-se então afirmar que um método de regularização destina-se a aproximar uma solução  $f^\dagger = A^\dagger g$  de um problema mal-posto (2.1), por uma família de problemas bem-postos  $R_\alpha(g_\epsilon)$ , de modo a obter uma solução estável e que convirja para a solução do problema inverso considerado [7]. Portanto, o propósito da regularização é incorporar informações adicionais acerca da solução desejada, no intuito de estabilizar o problema e obter uma solução estável.

## 2.2 Regularização de Tikhonov

O método de regularização de Tikhonov caracteriza-se por possuir uma formulação que envolve um processo de regularização, cuja principal estratégia é obter uma solução de norma mínima, admitindo dados com ruídos. Uma das representações gerais desse método é [4]:

$$\min_f \{ \|Af - g\|_2^2 + \lambda^2 \|Lf\|_2^2 \}, \quad (2.7)$$

onde  $\lambda \geq 0$  é o parâmetro de regularização que controla o balanço (peso) entre o primeiro e o segundo termo da expressão e é fundamental para a solução de Tikhonov pois, se ele assumir valores muito altos, muito peso é dado para a minimização da norma da solução e portanto, para  $\lambda \rightarrow \infty$ , ela aproxima o espaço nulo de  $L$ . Observe-se que, se  $\lambda = 0$ , obtém-se a solução exata. Ainda em (2.7), o termo  $\|Af - g\|_2^2$  representa a medida de quanto a solução  $f$  prevê os dados  $g$ , e  $\|Lf\|_2^2$  é o termo de regularização, que tem a função de diminuir as altas frequências dos componentes com grande ruído. O termo estabilizador  $L$  pode ser uma aproximação discreta de um operador derivada (ou da matriz identidade), e possibilita a inclusão de informação *a priori* sobre a solução.

É possível reescrever (2.7) como o problema de mínimos quadrados:

$$\min_f \left\| \begin{pmatrix} A \\ \lambda L \end{pmatrix} f - \begin{pmatrix} g \\ 0 \end{pmatrix} \right\|_2. \quad (2.8)$$

Considerando-se  $L = I$  na equação acima e efetuando-se uma análise através da decomposição em valores singulares (DVS) de  $A$ , pode-se compreender teoricamente o significado da solução de Tikhonov ( $f_\lambda$ ). As equações normais <sup>1</sup> para o problema (2.8) são:

$$f_\lambda = (A^T A + \lambda^2 I)^{-1} A^T g. \quad (2.9)$$

em que  $I$  representa a matriz identidade e  $A^T$  a transposta de  $A$ . Considerando-se a decomposição em valores singulares (DVS) de  $A$  e o fato de que  $I = VV^T$  [21], é possível redefinir  $f_\lambda$  [4], [16]:

$$\begin{aligned} f_\lambda &= (V\Sigma^2 V^T + \lambda^2 VV^T)^{-1} V\Sigma U^T g \\ &= V(\Sigma^2 + \lambda^2 I)^{-1} V^T V\Sigma U^T g \\ &= V(\Sigma^2 + \lambda^2 I)^{-1} \Sigma U^T g. \end{aligned} \quad (2.10)$$

em que  $U = (u_1, \dots, u_n)$  e  $V = (v_1, \dots, v_n)$  são matrizes com colunas ortonormais,  $U^T U = VV^T = I_n$  e  $\Sigma = \text{diag}(\mu_1, \dots, \mu_n)$  é a matriz diagonal contendo os valores singulares de  $A$ , tais que  $\mu_1 \geq \mu_2 \geq \dots \geq \mu_n \geq 0$ . Inserindo-se os valores singulares e os vetores singulares à esquerda e à direita, respectivamente  $u_i$  e  $v_i$ , em (2.10) obtém-se [22]:

$$f_\lambda = \sum_i^n \hat{f}_i \frac{u_i^T g}{\sigma_i} v_i, \quad (2.11)$$

em que  $\hat{f}_i$  são os fatores de filtragem para  $i = 1, 2, \dots, n$ , tal que  $\hat{f}_i = \frac{\mu_i^2}{\mu_i^2 + \lambda^2} \simeq \begin{cases} 1, & \mu_i \gg \lambda \\ \frac{\mu_i^2}{\lambda^2}, & \mu_i \ll \lambda. \end{cases}$

Esses fatores de filtragem amortecem as contribuições da maior parte dos valores singulares quando  $\lambda$  é grande, enquanto que, se  $\lambda$  é pequeno, a maior parte dos componentes da DVS é usada. Definindo-se, para todo  $i$ :

$$\hat{f}_i = 1, \quad (2.12)$$

a solução da regularização de Tikhonov é a solução de mínimos quadrados (a ser detalhada mais adiante).

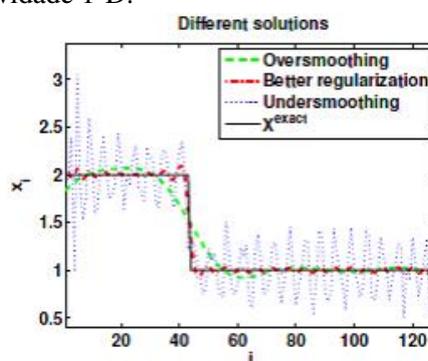
Uma das características mais relevantes dessa regularização é o amortecimento dos menores valores singulares e, portanto, as contribuições dos componentes são dominadas pelo ruído. A

<sup>1</sup>As equações normais para o problema de mínimos quadrados serão apresentadas, com mais detalhes, na próxima seção.

propósito, o parâmetro de regularização determina se a solução regularizada deve ajustar os dados ou remover parte do ruído. Quando  $\lambda$  é muito grande muita regularização é adicionada (*oversmoothing*), o termo  $\|Af_\lambda - g\|_2^2$  torna-se maior, e a solução não combina com os dados  $g$  [22]. Por outro lado, se  $\lambda$  é muito pequeno, uma pequena regularização é aplicada (*undersmoothing*) e a solução é dominada pelo ruído, do mesmo modo que a solução original. Portanto, é necessário estabelecer uma boa estratégia para a determinação de um valor de  $\lambda$  que seja o mais adequado para balancear essas características.

A figura a seguir apresenta um representação gráfica de diferentes soluções encontradas usando a regularização de Tikhonov. O experimento consiste no problema do modelo de mapeamento da gravidade 1-D, e pode ser visto, com detalhes, em [21]:

Figura 2.2: Solução exata e outras soluções usando a Regularização de Tikhonov. Problema do modelo de mapeamento da gravidade 1-D.



Fonte: Hansen [21].

Na Figura 2.2 a linha preta contínua representa a solução exata, e as outras linhas simbolizam soluções determinadas pela regularização de Tikhonov; a linha vermelha refere-se à melhor solução, a verde representa a solução calculada com muita regularização, ou seja, com um grande valor para  $\lambda$  (*oversmoothing*), e a linha pontilhada, caracterizada por muitas oscilações, refere-se à solução obtida com um pequeno valor de  $\lambda$  (*undersmoothing*), sendo portanto dominada pelo ruído.

## 2.3 A Regularização de Tikhonov numa Perspectiva Geométrica

Nesta seção, a ideia central é realizar uma breve abordagem sobre as características do método dos mínimos quadrados, bem como evidenciar, de um ponto de vista analítico e geométrico, a influência do ruído nas soluções de problemas inversos, com destaque para as deformações topográficas da função custo decorrentes dessa influência e da regularização de Tikhonov.

### 2.3.1 Método dos Mínimos Quadrados

Como mencionado, em diversos casos não é possível estabelecer a solução exata de um sistema linear, independentemente dele representar um problema direto ou inverso. É razoável, portanto, pensar em formas de determinar uma solução que melhor se aproxime da solução exata, produzindo o mínimo de erros (ruídos). Recorde-se então o sistema (2.1). A melhor forma de determinar uma

solução aproximada para o sistema  $Af = g$  é calcular  $f$  de modo que  $Af$  seja o mais próximo possível de  $g$ , ou seja, pensar em  $Af$  como uma aproximação de  $g$  e assim encontrar a menor distância entre eles (representada por  $\|g - Af\|_2$ ), aqui chamada de melhor aproximação. De posse deste conceito, define-se então uma solução de mínimos quadrados [23]:

**Definição 2.5** *Seja  $A$  uma matriz  $m \times n$  e  $g \in \mathbb{R}^m$ . Define-se  $\hat{f} \in \mathbb{R}^n$  como uma solução de mínimos quadrados de  $Af = g$ , quando*

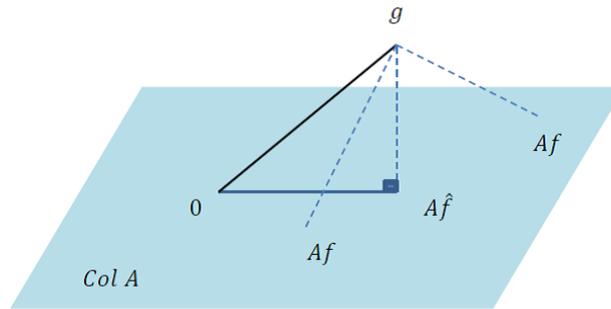
$$\|g - A\hat{f}\|_2 \leq \|g - Af\|_2, \quad (2.13)$$

para todo  $f \in \mathbb{R}^n$ .

O aspecto mais importante do problema dos mínimos quadrados é que, independentemente de qual  $f$  é selecionado, o vetor  $Af$  estará no espaço coluna,  $ColA$ , e portanto busca-se um  $f$  que torne  $Af$  o ponto no espaço coluna mais próximo de  $g$ .

A figura abaixo ilustra esse fato, evidenciando que a solução de mínimos quadrados  $\hat{f}$  é aquela para a qual obtém-se a melhor aproximação de  $g$ .

Figura 2.3: O vetor  $g$  é mais próximo de  $A\hat{f}$  do que de  $Af$  para qualquer outro  $f$ .



Fonte: Adaptado de Lay [23].

Pode-se então afirmar que  $A\hat{f}$  representa a projeção ortogonal de  $g$  no espaço das colunas de  $A$  ( $ColA \subseteq \mathbb{R}^m$ ), e que a distância  $\|g - A\hat{f}\|_2$  é menor do que todas as distâncias  $\|g - Af\|_2$ , para qualquer outro  $f$ . Considerando-se que  $\hat{g} = \text{proj}_{ColA} g$  ( $\text{proj}_{ColA}$  representa a projeção ortogonal de  $g$  sobre o espaço  $ColA$ ), então  $\hat{g}$  está em  $ColA$  e assim a equação  $Af = \hat{g}$  é consistente. Portanto, existe um  $\hat{f} \in \mathbb{R}^n$  tal que

$$A\hat{f} = \hat{g}. \quad (2.14)$$

Assim, um vetor  $\hat{f}$  é solução de mínimos quadrados de  $Af = g$  se satisfaz (2.14). Pelo Teorema da Decomposição Ortogonal [23], [17], é possível afirmar que a projeção  $\hat{g}$  tem a propriedade de que  $g - \hat{g}$  é ortogonal a  $ColA$  e portanto  $g - A\hat{f}$  é ortogonal a cada coluna de  $A$ . Se  $a_j$  é uma coluna qualquer de  $A$ , então  $a_j \cdot (g - A\hat{f}) = 0$  (a operação  $\cdot$  representa o produto interno) e  $a_j^T (g - A\hat{f}) = 0$ . Como  $a_j^T$  é uma linha de  $A^T$ , tem-se que

$$A^T (g - A\hat{f}) = 0 \quad (2.15)$$

e daí,

$$A^T g - A^T A\hat{f} = 0, \quad (2.16)$$

isto é,

$$A^T A \hat{f} = A^T g. \quad (2.17)$$

Isto significa que cada solução de mínimos quadrados de  $Af = g$  satisfaz o sistema de equações  $A^T A f = A^T g$ , denominado de **equações normais** de  $Af = g$ .

Os resultados enunciados a seguir são bastante conhecidos e encontram-se, por exemplo, em [23] e [17]. Tratam da definição de um conjunto de soluções de mínimos quadrados para (2.1), e ainda sobre a unicidade dessas soluções:

**Teorema 2.6** *O conjunto de soluções de mínimos quadrados de  $Af = g$  coincide com o conjunto não-vazio de soluções das equações normais  $A^T A \hat{f} = A^T g$*

**Teorema 2.7** *Seja  $A$  uma matriz  $m \times n$ . As seguintes afirmações são equivalentes:*

1. *A equação  $Af = g$  tem uma única solução de mínimos quadrados para cada  $g \in \mathbb{R}^m$ .*
2. *As colunas de  $A$  são linearmente independentes.*
3. *A matriz  $A^T A$  é invertível.*

Quando as afirmações acima são verdadeiras, a solução de mínimos quadrados  $\hat{f}$  é dada por

$$\hat{f} = (A^T A)^{-1} A^T g. \quad (2.18)$$

Em alguns casos, essas equações normais para o problema dos mínimos quadrados podem ser mal-condicionadas, isto é, pequenos erros nos cálculos das entradas de  $A^T A$  podem causar erros bastante grandes na solução  $\hat{f}$ . Então, se as colunas de  $A$  são linearmente independentes, é possível determinar uma solução de mínimos quadrados através da fatoração de  $A$  por algum método auxiliar (como a fatoração QR, por exemplo)<sup>2</sup>.

### 2.3.2 Efeitos do ruído em problemas inversos

Embora tenha sido bastante estudado devido à sua existência em diversos tipos de problemas, o ruído produzido nos processos de inversão de sistemas lineares e não-lineares constitui um grande desafio, posto que controlá-lo é uma tarefa ainda não resolvida completamente. Além disso, a maior parte dos estudos tenta determinar técnicas de regularização que visem diminuir o impacto causado pelo ruído na solução encontrada por mínimos quadrados ou por métodos de otimização local. Em geral, o efeito do ruído nos dados é estudado através de análises estatísticas e experimentos numéricos e, desse modo, pouca teoria analítica tem sido desenvolvida com esse objetivo. Neste estudo, serão pontuados alguns resultados recentes trazidos por Fernández-Martínez *et al.* [11], [12], [13], com o objetivo de tecer uma base analítica para o papel do ruído e da regularização de Tikhonov, considerando, para tal, uma interessante visão geométrica da sua atuação na deformação topográfica da função custo, no processo de inversão linear. O caso não-linear pode ser encontrado em [14].

Considere-se o problema inverso (2.1), aqui representado por

$$F(m) = d, \quad (2.19)$$

<sup>2</sup>Detalhes sobre este processo de fatoração podem ser encontrados em Malajovich [27].

em que o operador  $F$  representa o problema no sentido direto,  $m$  o conjunto discreto de modelos de parâmetros a ser determinado, e  $d$  denota os dados observados, que são sempre afetados pelo ruído. No caso de problemas inversos lineares, o operador  $F$  corresponde a uma aplicação linear entre dois espaços lineares  $F : \mathbb{R}^n \rightarrow \mathbb{R}^s$ , e a função custo é usualmente definida como a norma euclidiana quadrada do vetor residual [13]:

$$c(m) = \|F(m) - d\|_2^2. \quad (2.20)$$

Para sistemas lineares mal-condicionados, através da pseudo-inversa de Moore-Penrose [1], [18] o ruído nos dados pode determinar diferentes soluções (conhecidas como soluções equivalentes<sup>3</sup>), se nenhum método de regularização for utilizado. Isto depende dos vetores singulares à direita  $v_k \in V$  do operador  $F$ , que correspondem aos valores singulares  $\mu_k \in \Sigma$  (considerando-se a DVS:  $F = U\Sigma V^T$ ), que estão próximos de zero e "quase" contribuem pra expandir o espaço nulo de  $F$ ,  $N(F) = \{m \in \mathbb{R}^n : F(m) = 0\}$  ( $N(F)$  refere-se ao núcleo de  $F$ ) [12], ou seja, o subespaço linear do  $\mathbb{R}^n$  contendo os modelos que não têm influência sobre a pré-determinação dos dados.

A maior parte dos problemas inversos pode ser escrita, na forma discreta, como:

$$F(m) \approx d^{obs}, \quad (2.21)$$

em que  $m = (m_1, m_2, \dots, m_n) \in M \subset \mathbb{R}^n$  é o modelo estimado que pertence ao conjunto de modelos admissíveis  $M$ , definido a partir de algum conhecimento *a priori*,  $d^{obs} \in \mathbb{R}^s$  é o dado observado,  $F(m) = (f_1(m), f_2(m), \dots, f_s(m))$  é o campo vetorial que representa o modelo direto e  $f_j(m)$  é o campo escalar que considera o  $j$ -ésimo dado. O símbolo ( $\approx$ ) significa que a relação pode não ser exata, e então o problema definido em (2.21), que consiste em determinar o modelo ou modelos  $m$ , cujas pré-definições  $F(m)$  combinem precisamente com os dados observados  $d^{obs}$ , podem não ter solução [12]. Esse fato está relacionado ao caráter mal-posto dos problemas inversos, isto é, ou o problema inverso não admite solução, ou ela existe mas não é única ou, finalmente, a solução existe e é única mas é instável, ou seja, a solução não depende continuamente do dado observado.

Os problemas lineares inversos são aqueles em que as funções que compõem o campo vetorial  $F$  dependem linearmente do modelo  $m$ . Neste caso, resolver  $F(m) = d^{obs}$ , com  $F \in M_{(s,n)}(\mathbb{R})$  equivale a determinar a interseção dos  $s$  hiperplanos em  $\mathbb{R}^n$ :  $f_j(m) - d_j = 0$ ,  $j = 1, \dots, s$ . A presença do ruído  $\delta d$  no dado observado  $d^{obs}$  faz com que o problema da interseção não tenha solução. Uma outra causa da incompatibilidade é o fato de que o operador  $F$  é um modelo matemático que se aproxima da realidade, o que pode causar erros de modelagem<sup>4</sup>. Devido a essa incompatibilidade de (2.21), o problema inverso pode ser resolvido através dos mínimos quadrados (2.20), onde a solução generalizada para o problema inverso é determinada pela pseudo-inversa de Moore-Penrose  $m^\dagger$ , encontrando [13]:

$$c(m^\dagger) = \min_{m \in \mathbb{R}^n} \|Fm - d^{obs}\|_2^2. \quad (2.22)$$

Sejam  $d^{obs} = d^{true} + \delta d$ , onde  $d^{true}$  é o dado observado que seria medido na ausência de ruído, e  $m^{true}$  o modelo hipotético que gera esse dado, isto é,  $Fm^{true} = d^{true}$ .

Então,

$$m^\dagger = F^\dagger d^{obs} = F^\dagger (d^{true} + \delta d) = F^\dagger d^{true} + F^\dagger \delta d = m^{true} + \Delta m, \quad (2.23)$$

com

$$\Delta m = F^\dagger \delta d = V \Sigma^\dagger U^T \delta d = \sum_{k=1}^r \frac{\delta d_{Uk}}{\mu_k} v_k, \quad (2.24)$$

<sup>3</sup>Consultar, por exemplo, Aster *et al.* [1] e Menke [24].

<sup>4</sup>Detalhes sobre erros de modelagem podem ser encontrados em Franco (2006).

em que  $V\Sigma^\dagger U^T$  e  $r$  são, respectivamente, a decomposição DVS e o posto de  $F$ , e  $\delta d_U$  é o termo de ruído  $\delta d$  na base ortonormal  $U$ .

Considere-se a região das soluções equivalentes, representada por

$$L_{tol} = \left\{ m \in M : \|Fm - d^{obs}\|_2^2 \leq tol^2 \right\}. \quad (2.25)$$

É possível observar como o ruído atua na região de soluções equivalentes,  $L_{tol}$ . Segundo Fernández-Martínez *et al.* [12], a região que contém os modelos que se ajustam aos dados observados, dentro de uma tolerância  $tol$ ,  $\|Fm - d^{obs}\|_2^2 \leq tol^2$ , é uma hiperquádrica centrada em um certo  $m_0$ , tendo os seus eixos principais orientados segundo os vetores singulares à direita (base  $V$ ) do operador direto  $F$  (considerando-se a decomposição DVS de  $F$ ). Os semi-eixos dessa hiperquádrica linear são os inversos dos valores singulares ( $\mu_k$ ) de  $F$ .

Para provar esse fato, considere-se  $m = m^\dagger + \Delta m$ , onde  $m^\dagger = F^{-1}d$  é a única solução do sistema linear. Então, os modelos equivalentes com pré-determinação do erro de tolerância ( $tol$ ), satisfazem [12]:

$$\frac{\|Fm - d\|_2}{\|d\|_2} \leq tol \Rightarrow (\Delta m)^T F^T F (\Delta m) \leq tol^2 \|d\|_2^2. \quad (2.26)$$

De fato,

$$\frac{\|Fm - d\|_2}{\|d\|_2} \leq tol \Rightarrow \|Fm - d\|_2 \leq tol \|d\|_2. \quad (2.27)$$

Elevando-se a expressão anterior ao quadrado, e substituindo-se o valor de  $m$ , obtém-se:

$$\|F(m^\dagger + \Delta m) - d\|_2^2 \leq tol^2 \|d\|_2^2, \quad (2.28)$$

isto é,

$$\|Fm^\dagger + F\Delta m - d\|_2^2 \leq tol^2 \|d\|_2^2. \quad (2.29)$$

Mas  $Fm^\dagger = d$ , e portanto

$$\|F\Delta m\|_2^2 \leq tol^2 \|d\|_2^2. \quad (2.30)$$

Escrevendo-se o primeiro membro de (2.30) através do produto interno, tem-se que:

$$\langle F\Delta m, F\Delta m \rangle \leq tol^2 \|d\|_2^2, \quad (2.31)$$

ou seja,

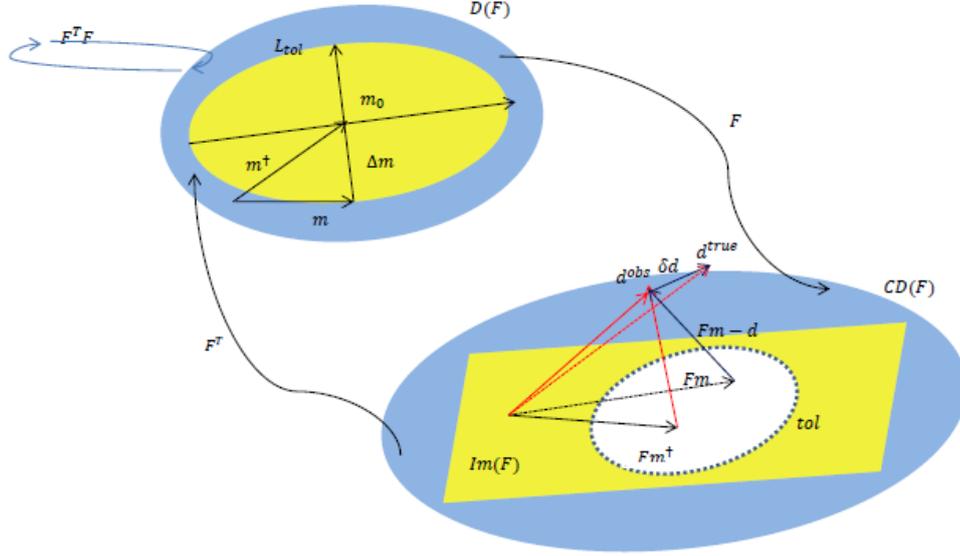
$$(F\Delta m)^T F\Delta m \leq tol^2 \|d\|_2^2. \quad (2.32)$$

Portanto,

$$(\Delta m)^T F^T F \Delta m \leq tol^2 \|d\|_2^2. \quad (2.33)$$

Como conclusão,  $F^T F$  é a matriz simétrica que representa a hiper-quádrica (que neste caso é uma cônica), no domínio do operador  $F$ . Um esquema gráfico desta cônica e da região de modelos equivalentes, bem como das soluções no conjunto imagem de  $F$ , é apresentado na figura a seguir:

Figura 2.4: Esquema das regiões de modelos equivalentes e de soluções.



Na representação acima, a fronteira da região  $L_{tol}$  é a hiper-quádrica contendo os modelos lineares equivalentes, que pré-determinam os dados observados com erro de tolerância  $tol$ , representado por uma região do plano  $Im(F)$ . Os vetores  $Fm$  e  $Fm^\dagger$  representam as soluções na imagem de  $F$ , sendo que o segundo deles indica a solução de mínimos quadrados, determinada pela projeção ortogonal de  $d^{obs}$  em  $Im(F)$ .

Analicamente, esta hiper-quádrica pode ser representada por [11]:

$$\|Fm\|_2^2 - 2(Fm)^T d^{obs} + \|d^{obs}\|_2^2 - tol^2 = 0, \quad (2.34)$$

ou equivalentemente,

$$\|\Sigma m_V\|_2^2 - 2(\Sigma m_V)^T d_U^{obs} + \|d_U^{obs}\|_2^2 - tol^2 = 0 \quad (2.35)$$

com  $d_U^{obs} = U^T d^{obs}$ , e  $m_V = V^T m$ . Em (2.35) a hiperquádrica refere-se às bases ortogonais  $U$  e  $V$  (eixos principais).

Segundo Fernández-Martínez *et al.* [13]:

1. O ruído nos dados não afeta a matriz da hiper-quádrica  $F^T F$ . Assim, a orientação da hiper-quádrica linear (eixo principal  $V$ ) não se altera, mesmo sob efeito do ruído.
2. O ruído perturba o centro da hiper-quádrica, que coincide com a solução de mínimos quadrados  $m^\dagger$  do problema linear inverso definido na equação (2.22).
3. O ruído nos dados deforma homogeneamente a topografia da função de custo não-perturbada na região de modelos equivalentes, limitada por  $L_{tol}$ .

Para provar analiticamente a afirmação (3), os referidos autores reescreveram a equação (2.35) como:

$$m_V^T \Sigma^T \Sigma m_V - 2m_V^T \Sigma^T d_U^{obs} = tol^2 - \left(d_U^{obs}\right)^T d_U^{obs}, \quad (2.36)$$

que, após algumas manipulações algébricas, resultou em

$$\sum_{k=1}^r \left( \mu_k m_{V_k} d_U^{obs} \right)^2 = tol^2 - \sum_{k=r+1}^s \left( d_U^{obs} \right)^2. \quad (2.37)$$

Estudaram ainda o comportamento da hiper-quádrica a partir de restrições acerca do posto da matriz  $F$ . Essa hiper-quádrica linear é uma hiper-elipsóide para sistemas lineares puramente sobre-determinados (ou de posto completo), em que seu centro  $m_c = F^\dagger d$  coincide com a solução de mínimos quadrados do problema inverso linear  $Fm = d$ . Se o sistema linear é mal-condicionado, os eixos da hiper-quádrica ao longo das direções da base  $V$ , que corresponde aos menores valores singulares  $\mu_k$ , são muito longos, e a determinação do centro da hiper-quádrica torna-se bastante imprecisa devido ao ruído. No caso de sistemas lineares puramente sub-determinados ou de posto deficiente, a hiper-quádrica degenera para um cilindro elíptico com eixos infinitamente longos (representados por  $\infty_j$ ) nas direções que abrangem o núcleo do operador direto [11]. Em resumo, se  $r = \text{posto}(F) < n$ , os componentes  $m_{V_k}$ , com  $k = r + 1, \dots, n$ , não são determinados pelo sistema linear e a hiper-quádrica torna-se um cilindro elíptico nas direções de  $v_k$ , com  $k = r + 1, \dots, n$ . Se  $r = \text{posto}(F) = n$ , os comprimentos dos eixos da hiper-quádrica, definidos por

$$\left[ \frac{\sqrt{tol^2 - \sum_{k=r+1}^s (d_{U_k}^{obs})^2}}{\mu_k}, \infty_j \right]; \quad k = 1, \dots, r; \quad j = r + 1, \dots, n, \quad (2.38)$$

são finitos.

Considerando-se agora que  $d_{U_k}^{obs} = d_{U_k}^{true} + \delta d_{U_k}$  (dados com ruído), a equação da hiper-quádrica (2.37) torna-se:

$$\sum_{k=1}^r \left[ \mu_k m_{V_k} - (d_{U_k}^{true} + \delta d_{U_k}) \right]^2 = tol^2 - \sum_{k=r+1}^s (d_{U_k}^{true} + \delta d_{U_k})^2, \quad (2.39)$$

onde o modelo e os dados referem-se respectivamente às bases  $V$  e  $U$ , e os comprimentos dos eixos são dados por:

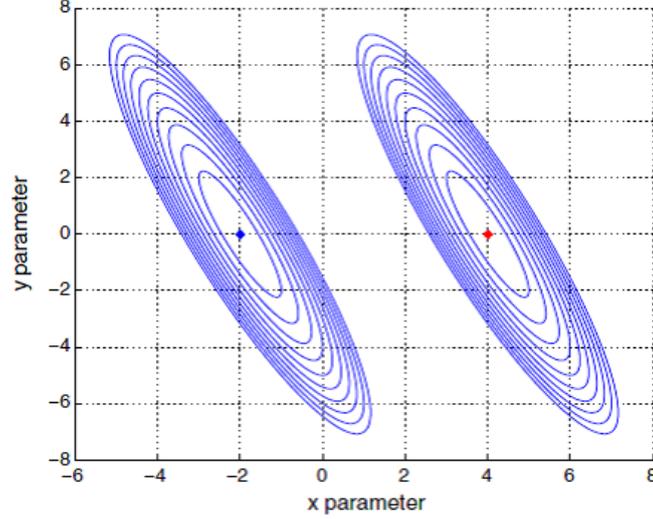
$$\left[ \frac{\sqrt{tol^2 - \sum_{k=r+1}^s (d_{U_k}^{true} + \delta d_{U_k})^2}}{\mu_k}, \infty_j \right]; \quad k = 1, \dots, r; \quad j = r + 1, \dots, n. \quad (2.40)$$

No caso livre de ruídos, a equação da hiper-quádrica é a versão simplificada de (2.39):

$$\sum_{k=1}^r (\mu_k m_{V_k} - d_{U_k}^{true})^2 = tol^2 - \sum_{k=r+1}^s (d_{U_k}^{true})^2. \quad (2.41)$$

O termo  $\sum_{k=r+1}^s (d_{U_k}^{true} + \delta d_{U_k})^2$  na equação (2.39) é não-nulo para sistemas lineares incompatíveis, com posto incompleto. Neste caso, a cônica no plano está definida, mas nas direções que pertencem ao núcleo de  $F$  (não degenerado). Para o caso de sistemas compatíveis o mesmo termo se anula, independentemente das dimensões do sistema. Como resultado, o ruído apenas transfere a solução de lugar, sem alterar a topografia da região de equivalência, como ilustra a Figura (2.5). Nesta representação, pode-se observar que a região de modelos equivalentes do sistema linear sem ruídos (com centro marcado em azul) foi transferida para a direita (região com centro marcado em vermelho), após perturbação pelo ruído, sem, no entanto, ter a sua topografia deformada.

Figura 2.5: Efeito do ruído em sistemas lineares compatíveis.



Fonte: Fernández-Martínez *et al.* [13]

Para que os eixos das hiper-quádricas (2.39) e (2.41) tenham os mesmos comprimentos, a condição a seguir tem que ser satisfeita:

$$tol^* = \sqrt{tol^2 + \sum_{k=r+1}^s [(d_{Uk}^{true} + \delta d_{Uk})^2 - (d_{Uk}^{true})^2]} = \sqrt{tol^2 + \sum_{k=r+1}^s (d_{Uk}^2 + 2\delta d_{Uk} d_{Uk}^{true})}. \quad (2.42)$$

Seja  $\Delta tol = \sum_{k=r+1}^s (\delta d_{Uk}^2 + 2\delta d_{Uk} d_{Uk}^{true})$ . Então [13]:

i. Se  $\Delta tol > 0$ , então  $tol^* > tol$ , isto é, na presença de ruído a região de mesmo tamanho corresponde à de maior desajuste. Assim as regiões de modelos equivalentes decrescerão em tamanho.

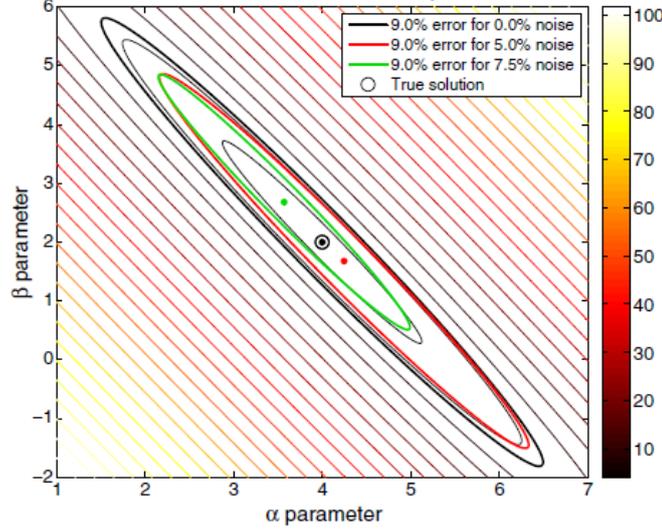
ii. Se  $\Delta tol < 0$ , então  $tol^* < tol$ , e a região com o mesmo tamanho na presença do ruído corresponde à região de menor desajuste. Neste caso, as regiões de modelos equivalentes irão crescer em tamanho.

Para ilustrar as conclusões apontadas nesta seção, Fernández-Martínez [13] conceberam um exemplo sintético, para um problema de regressão linear. Para tal, gerou-se um conjunto de dados  $\{(x_1, y_1), (x_2, y_2), \dots, (x_s, y_s)\}$ , de acordo com o modelo teórico  $y_k = \alpha x_k + \beta + \varepsilon$ , onde  $\alpha = 4$  e  $\beta = 2$ . Para o primeiro teste, foi adicionada uma variável aleatória Gaussiana  $\varepsilon \rightarrow N(0, 0.05)$ , tentando simular o ruído no dado observado durante a medida. Para o segundo teste foi adicionado ao primeiro um termo de ruído extra,  $\delta \rightarrow N(0, 0.025)$ .

A Figura 2.6 mostra um desajuste relativo (região de tolerância) de 9%, que corresponde à topografia da função custo no caso livre de ruído (elipse em preto), centrada em  $(\alpha, \beta)$ . É mostrada ainda a mesma linha de contorno da função de custo no primeiro caso (ruído  $\varepsilon \rightarrow N(0, 0.05)$ ), marcada em vermelho. Esta linha está centrada na vizinhança da solução de mínimos quadrados  $m^\dagger = (4.25, 1.68)$ , e corresponde à hiper-quádrica definida pela equação (2.34), cuja solução perturbada apresenta-se deslocada em relação à da hiper-quádrica livre de ruído. Ainda para este exemplo, é possível identificar a mesma linha de contorno (em verde) com região de tolerância de 9% para o segundo caso (ruído  $\varepsilon \rightarrow N(0, 0.05) + \delta \rightarrow N(0, 0.025)$ ). Neste caso, a solução de mínimos quadrados foi deslocada para  $(3.57, 2.68)$ , mais distante da solução verdadeira, e a região

de equivalência tornou-se ainda menor.

Figura 2.6: Efeito do ruído em problemas lineares.



Fonte: Fernández-Martínez *et al.* [13]

### 2.3.3 Efeitos da regularização de Tikhonov em problemas lineares inversos

A regularização de um problema inverso pressupõe a estabilização da inversão, que pode ser realizada através de truncamento e amortização (*damping*). Como exemplo do uso do truncamento, mencionou-se o processo de inversão generalizada de Moore-Penrose, cujo truncamento tem a função de evitar efeitos indesejados do ruído nos dados, devido aos menores valores singulares de  $F$ . A segunda estratégia consiste na adição de um termo extra na função de custo, que penaliza o tamanho da solução, no sentido em que [12]:

$$\|Fm - d\|_2^2 + \lambda^2 \|m\|_2^2 = (Fm - d)^T (Fm - d) + \lambda^2 m^T m = \|d\|_2^2 m^T (F^T F + \lambda^2 I_n) m - 2m^T F^T d, \quad (2.43)$$

em que  $\lambda$  é o parâmetro de amortização, próximo de zero. No caso de um problema linear inverso, pode-se observar como a expressão (2.43) serve para estabilizar a inversão. As equações normais são:

$$(FF^T + \lambda^2 I_n) m = F^T d, \quad (2.44)$$

onde o operador  $FF^T + \lambda^2 I_n$  torna-se de posto completo, ou seja, bijetivo. Assim o operador inverso  $(FF^T + \lambda^2 I_n)^{-1}$  existe e é contínuo.

A regularização de Tikhonov tem dois efeitos principais:

1. A hiper-quádrica linear torna-se uma hiper-elipsóide, cujos maiores eixos são  $1/\lambda$  nas direções dos vetores  $V$ , abrangendo o espaço nulo de  $F$  ( $\text{posto}(F)$ ).
2. O centro da hiper-quádrica é transferido para  $m_c^\lambda = (FF^T + \lambda^2 I_n)^{-1} F^T d$ , e os seus principais eixos são ligeiramente rotacionados.

Esses dois efeitos podem se tornar praticamente negligíveis quando o parâmetro de regularização tende a zero ( $\lambda \rightarrow 0$ ). O principal efeito da regularização é limitar o comprimento da região do "vale de desajuste", fazendo com que a região linear de equivalência torne-se limitada [11].

A regularização de Tikhonov de ordem zero consiste na minimização de uma combinação linear da previsão de dados desajustados e a distância para um modelo de referência ( $m^{ref}$ ). Conforme Fernández-Martínez *et al.* [13], essa regularização é dada por

$$\min_{m \in \mathbb{R}^n} \left( \|Fm - d^{obs}\|_2^2 + \lambda^2 \|m - m^{ref}\|_2^2 \right), \quad (2.45)$$

e neste caso, a solução regularizada de mínimos quadrados é [1], [13]:

$$m^\varepsilon = m^{ref} + (F^T F + \varepsilon^2 I_n)^{-1} F^T (d^{obs} - Fm^{ref}), \quad (2.46)$$

que pode ser reescrita, considerando-se a decomposição DVS de  $F$ , como

$$m^\varepsilon = m^{ref} + V (\Sigma^T \Sigma + \varepsilon^2 I_n)^{-1} \Sigma^T U^T (d^{obs} - Fm^{ref}), \quad (2.47)$$

ou ainda como,

$$m_V^\varepsilon = m_V^{ref} + (\Sigma^T \Sigma + \varepsilon^2 I_n)^{-1} \Sigma^T \Delta d_U^{ref}. \quad (2.48)$$

Nesta última equação os modelos referem-se à base  $V$  e  $\Delta d_U^{ref}$  é o modelo de referência dos dados com desajuste, na base  $U$ :

$$\Delta d_U^{ref} = U^T (d^{obs} - Fm^{ref}) = d_U^{obs} - \Sigma m_V^{ref}. \quad (2.49)$$

Finalmente, em função das coordenadas na base  $V$ , a solução regularizada  $m^\lambda$  pode ser escrita como

$$m^\lambda = \sum_{k=1}^r \frac{\lambda^2 m_{V_k}^{ref} + \mu_k d_{U_k}^{obs}}{\mu_k^2 + \varepsilon^2} V_k + \sum_{j=r+1}^n m_{V_j}^{ref} V_j. \quad (2.50)$$

Observações:

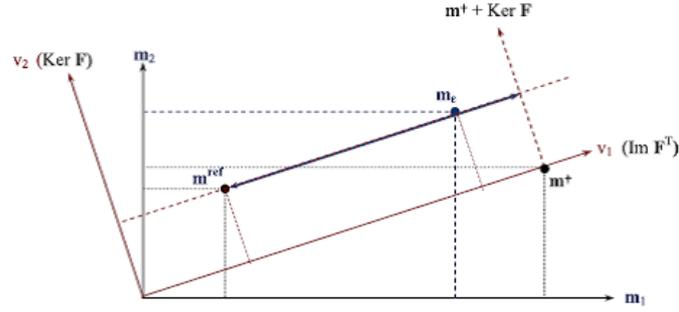
1. As coordenadas dos  $r$  primeiros vetores da base  $V$  são combinações lineares entre as coordenadas do modelo de referência  $m_{V_k}^{ref}$  e os componentes dos dados observados  $d_{U_k}^{obs}$ . Os pesos dessa combinação linear são

$$\frac{\lambda^2}{\mu_k^2 + \lambda^2} \text{ e } \frac{\mu_k}{\mu_k^2 + \lambda^2}. \quad (2.51)$$

2. As coordenadas de  $m^\dagger$ , que estavam originalmente situadas no núcleo de  $F$ , são agora informadas pelas coordenadas de  $m_{V_k}^{ref}$ .

A figura a seguir ilustra essas afirmações, no caso bidimensional:

Figura 2.7: Interpretação geométrica para a regularização.



Fonte: Fernández-Martínez *et al.* [13]

É possível observar que o modelo  $m^\dagger$ , advindo da pseudo-inversa de Moore-Penrose, não possui componentes no núcleo de  $F$ , mas  $m^\lambda$  (na imagem representado por  $m_\epsilon$ ) incorpora as coordenadas do modelo de referência, que está no núcleo.

Analogamente à seção anterior, define-se a equação da hiper-quádrica da região de equivalência, neste caso para a solução regularizada, como [12], [13]:

$$\|Fm - d^{obs}\|_2^2 + \lambda^2 \|m - m^{ref}\|_2^2 = tol^2, \quad (2.52)$$

isto é,

$$\|Fm\|_2^2 - 2(Fm)^T d^{obs} + \|d^{obs}\|_2^2 + \lambda^2 (\|m\|_2^2 - 2m^T m^{ref} + \|m^{ref}\|_2^2 - tol^2) = 0. \quad (2.53)$$

Considerando-se a base  $V$  e levando-se em conta que as transformações ortogonais conservam as normas dos vetores, reescreve-se a equação (2.53), após algumas manipulações algébricas, como a equação analítica da hiper-quádrica no caso regularizado:

$$\sum_{k=1}^r \left( \sqrt{\mu_k^2 + \lambda^2} m_{V_k} - \frac{\lambda^2 m_{V_k}^{ref} + \mu_k d_{U_k}^{obs}}{\sqrt{\mu_k^2 + \lambda^2}} \right)^2 + \lambda^2 \sum_{k=1}^r (m_{V_k} - m_{V_k}^{ref})^2 = tol^2 + \Delta r, \quad (2.54)$$

com

$$\Delta r = \sum_{k=1}^r \frac{(\lambda^2 m_{V_k}^{ref} + \mu_k d_{U_k}^{obs})^2}{\mu_k^2 + \lambda^2} - \|d_U^{obs}\|_2^2 - \lambda^2 \sum_{k=1}^r (m_{V_k}^{ref})^2. \quad (2.55)$$

Adotando-se a mesma metodologia para estudar os efeitos do ruído na resolução de problemas lineares inversos, são definidas as regiões de tolerância com e sem regularização, representadas, respectivamente por  $tol$  e  $tol^*$ . Para tal, é necessário estabelecer a seguinte relação entre essas regiões, assumindo-se que elas possuem o mesmo tamanho (ou eixos):

$$\frac{\sqrt{tol^{*2} + \Delta r}}{\sqrt{\mu_i^2 + \lambda^2}} = \frac{\sqrt{tol^2 - \sum_{k=r+1}^s (d_{U_k}^{obs})^2}}{\mu_i}, \quad i = 1, \dots, r. \quad (2.56)$$

Daí,

$$tol^*(\mu_i) = \sqrt{tol^2 + \frac{\varepsilon^2}{\mu_i^2} \left[ tol^2 - \sum_{k=r+1}^s (d_{Uk}^{obs})^2 \right] + \lambda^2 \sum_{k=1}^r \frac{(d_{Uk}^{obs} - \mu_k m_{Vk}^{ref})^2}{\mu_k^2 + \lambda^2}}. \quad (2.57)$$

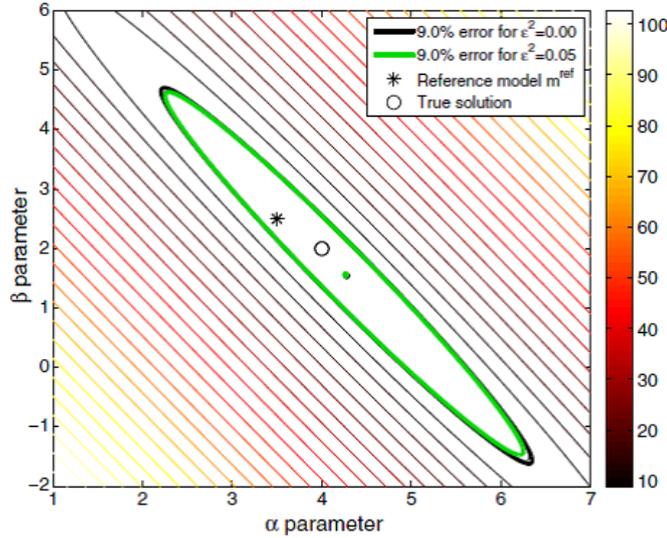
Neste caso,  $tol^*$  depende dos índices  $i$  dos eixos, e as regiões com erro de tolerância  $tol > tol_{ci}(\mu_i)$  têm decréscimo no tamanho devido ao efeito da regularização. Assim, os modelos devem ser encontrados em grandes áreas, deformando diferentemente os eixos na região de equivalência de acordo com (2.57). Reciprocamente,  $tol^*(\mu_i) < tol$  para qualquer região de tolerância  $tol < tol_{ci}(\mu_i)$ . Estas regiões equivalentes decrescem de tamanho e relação ao caso não regularizado [13].

O valor de  $tol_{ci}(\mu_i)$  depende de cada valor singular  $\mu_i$ , de da soma  $\sum_{k=r+1}^s (d_{Uk}^{obs})^2$ . É dado por:

$$tol_{ci}(\mu_i) = \sqrt{\sum_{k=r+1}^s (d_{Uk}^{obs})^2 - \mu_i^2 \sum_{k=1}^r \frac{(d_{Uk}^{obs} - \mu_k m_{Vk}^{ref})^2}{\mu_k^2 + \lambda^2}}. \quad (2.58)$$

Como conclusão, pode-se afirmar que a regularização afeta apenas os eixos da região de equivalência linear (hiper-quádrica) que corresponde aos menores valores singulares, deformando essa região de modo anisotrópico<sup>5</sup>. A figura abaixo utiliza o exemplo da seção anterior. Pode-se observar que a comparação entre as regiões de contorno com tolerância de 9%, nos casos com e sem regularização.

Figura 2.8: Efeito da regularização em problemas lineares.

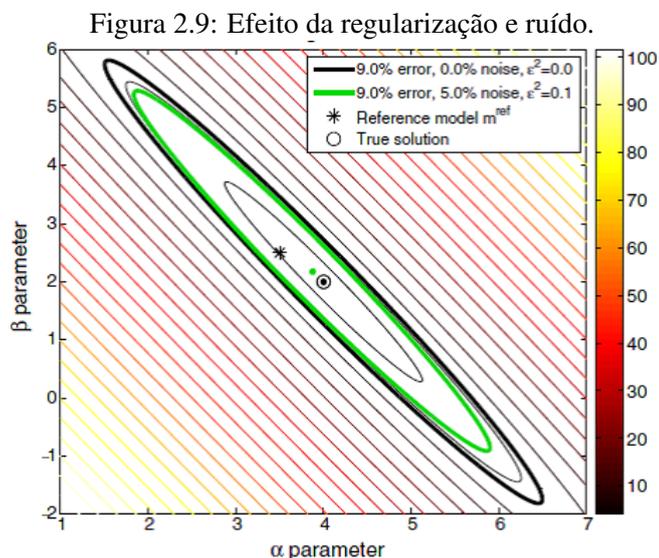


Fonte: Fernández-Martínez *et al.* [13]

Na Figura 2.8, a legenda apresenta, além das regiões de contorno, o parâmetro de regularização  $\lambda$  (os autores utilizam  $\varepsilon$  para este parâmetro), a solução do modelo de referência  $m^{ref}$  (asterisco) e a solução exata (círculo). O tamanho da região regularizada (em verde) apresentou decréscimo em relação à região original. A solução regularizada encontra-se mais próxima da solução verdadeira

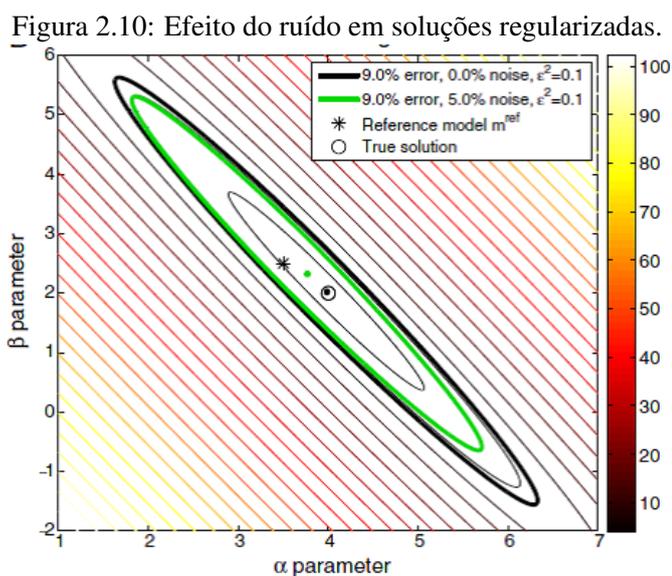
<sup>5</sup>A deformação varia conforme a direção

do que a solução de referência. Analogamente, pode-se comparar o efeito combinado do ruído e da regularização com respeito ao caso sem ruído. As figuras a seguir, usando os mesmos dados do exemplo já abordado, possibilitam a comparação entre os efeitos geométricos da regularização e do ruído em problemas lineares inversos.



Fonte: Fernández-Martínez *et al.* [13]

A figura acima mostra as regiões de 9% de tolerância para o caso livre de ruído e não-regularizado, com modelo de referência  $m^{ref}$  e um parâmetro de amortização  $\lambda^2 = 0.01$ . É possível observar que a região de equivalência, na presença de ruído e com regularização, apresenta tamanho reduzido em relação ao caso original, sem ruído e não-regularizado. A solução regularizada está localizada mais próxima da solução verdadeira do que a do modelo de referência.



Fonte: Fernández-Martínez *et al.* [13]

A Figura 2.10 compara as mesmas regiões de equivalência para o caso regularizado ( $\lambda^2 =$

0.01), com 5% de ruído gaussiano, ao caso livre de ruído. Como no exemplo anterior, a região com a presença do ruído tem tamanho menor em relação ao caso regularizado sem ruído, e sua solução ficou mais distante da verdadeira.

## 3

# A PGNAA em Fluidos Multifásicos

Nas linhas que se seguem, será realizada uma descrição da Análise de Ativação de Raios Gama Prontos Através de Nêutrons (*Prompt gamma-ray neutron activation analysis* - PGNAA) em fluidos multifásicos, baseada nos estudos de Meric *et al.* [25], com o objetivo de determinar as quantidades de substâncias presentes em amostras desses fluidos. Este tipo de análise pressupõe a resolução de um problema inverso, que em geral é bastante impreciso, dentre outros fatores, devido ao mau condicionamento da matriz de covariância, o que normalmente é causado quando duas ou mais substâncias da amostra são altamente relacionadas, isto é, possuem elementos em comum na sua composição, ou ainda quando há vestígios de elementos que possuem números expressivamente pequenos de seções de choques de captura de nêutrons térmicos. No sentido de minimizar esse mau condicionamento para auxiliar a execução da PGNAA, Meric *et al.* [25] propuseram um método iterativo para a utilização do Método dos Mínimos Quadrados de Análise de Componentes conjugado ao Monte Carlo (*Monte Carlo Library Least Squares* - MCLLS), para determinar os percentuais referentes às frações em peso das principais substâncias encontradas nas amostras estudadas.

### 3.1 Pressupostos

Estudos recentes do Centro de Aplicações de Engenharia de Radioisótopos - *Center for Engineering Applications of Radioisotopes* (CEAR) da *North Carolina State University*, envolvendo PGNAA para a análise de volumes de materiais como carvão e cimento (ver por exemplo, CLARK *et al.* [8]; SHYU *et al.* [30]; ZHANG e GARDNER [36]; HAN *et al.* [20]), a partir do desenvolvimento de diversas ferramentas computacionais, tais como o MCLLS, inspiraram a utilização dessas técnicas em aplicações em diferentes áreas, culminando com o interesse na análise de fluidos multifásicos obtidos na exploração de poços de petróleo submarino. Essa tarefa baseia-se no bombardeamento de uma amostra com nêutrons termais ou rápidos e na detecção dos raios gama prontos, produzidos no interior da amostra através de duas reações: a captura de nêutrons térmicos e o espalhamento não-elástico de nêutrons rápidos [25].

Os analisadores de radiação utilizam dados obtidos no processo de produção de raios gama prontos induzidos por nêutrons. A análise das substâncias que compõem uma amostra representa um problema inverso, pois nesse processo deseja-se determinar algumas características químicas ou físicas do objeto observado, a partir dos dados. Para este estudo, os dados que representam

maior interesse referem-se às quantidades de fração em peso dos diferentes elementos da amostra considerada.

No método MCLLS para a PGNAA aplicada a um fluido multifásico, o espectro total é representado pela soma de todas as bibliotecas <sup>1</sup>, e assim a resposta espectral da PGNAA a um dado elemento é função não apenas da quantidade desse elemento, mas também das quantidades de todos os outros elementos presentes em uma amostra, caracterizando-se como uma medida não linear. A abordagem MCLLS consegue então manipular essa não-linearidade através da utilização de um código Monte Carlo, especialmente proposto para a realização dos cálculos dos espectros individual e total das bibliotecas.

Nos estudos realizados por Wang *et al.* [35], foi considerada uma mistura homogênea de óleo, gás, água e sal para a aplicação da PGNAA associada ao MCLLS, objetivando determinar as quantidades dessas substâncias numa amostra (fluido multifásico). Inspirados nesse trabalho e levando em consideração que essa associação de métodos revelou-se não ser suficientemente sensível para a determinação das frações em peso de óleo e gás separadamente, Meric *et al.* [25] estudaram as causas desse problema e propuseram um novo método de utilização do MCLLS em PGNAA, observando que a matriz de covariância, que precisava ser invertida na execução dos mínimos quadrados, podia ser singular ou mal-condicionada. Esse mau-condicionamento motivou a elaboração do modelo proposto por eles.

## 3.2 O Método MCLLS

Devido ao fato de que a PGNAA é inerentemente uma técnica não-linear (como já mencionado), um método de mínimos quadrados (LLS, da sigla em inglês) não pode ser aplicado diretamente à análise quantitativa dos espectros de PGNAA e, portanto, gerar uma biblioteca que seja precisa e que ainda considere essas não-linearidades, é uma tarefa extremamente difícil. O método MCLLS trata essas não-linearidades gerando bibliotecas de espectros pertinentes, através de simulações Monte Carlo (MC) e portanto os efeitos dessas não-linearidades são parte integrante das simulações e das novas bibliotecas geradas pelo MC. Assim, os números de contagens de qualquer espectro podem ser determinados como uma combinação linear entre as escalas de contagem e as correspondentes bibliotecas [25], [26]

$$R_{s,i} = \sum_{j=1}^n \alpha_j R_{j,i} + e_i, \quad (3.1)$$

em que  $R_{s,i}$  é a escala de contagem no  $i$ -ésimo canal do espectro de uma amostra desconhecida,  $R_{j,i}$  é a escala de contagem no  $i$ -ésimo canal da  $j$ -ésima componente,  $\alpha_j$  é o multiplicador na  $j$ -ésima componente e  $e_i$  representa o erro no  $i$ -ésimo canal.

Para a análise quantitativa, os seguintes passos são considerados em MCLLS [25], [35]:

1. Obter ou assumir uma composição inicial da amostra de interesse.
2. Executar as simulações MC <sup>2</sup> com um código apropriado, para obter as bibliotecas dos espectros relevantes.

<sup>1</sup>Segundo [35], uma biblioteca é o espectro de um componente separado, como gás, óleo, ou água.

<sup>2</sup>Neste caso, o código apropriado para o uso do MC foi o CEARCPG [20].

3. Executar uma busca de bibliotecas de mínimos quadrados (LLS). Neste passo, os coeficientes mais adequados retornarão as quantidades de elementos relevantes na amostra.
4. Se as quantidades calculadas em LLS estiverem ainda distantes da estimativa inicial da composição da amostra, de modo que não ainda não se possa afirmar linearidade, deve-se então retornar para o passo 2 e reiterar MC utilizando os resultados recém calculados em LLS, considerando-os como a nova composição inicial.

### 3.2.1 O mau-condicionamento da matriz de covariância

A busca por soluções em MCLLS produz um conjunto de equações lineares (recordar 2.1)

$$Af = g,$$

em que  $A$  é a matriz de coeficientes,  $f$  é a matriz coluna contendo os multiplicadores das bibliotecas (desconhecidos), e  $g$  é o vetor cujos elementos são determinados pelas contagens no espectro de raios gama prontos. As dimensões das matrizes em (2.1) dependerão da quantidade de bibliotecas da amostra.

Para resolver o sistema linear acima, é necessário realizar a pseudo-inversão de  $A$ , obtendo então uma solução de mínimos quadrados  $\hat{f}$ , como em (2.18):

$$\hat{f} = (A^T A)^{-1} A^T g.$$

Além da matriz  $A$  não ser quadrada, e devido à não-singularidade da matriz de covariância  $A^T A$ , o conjunto de soluções do sistema linear pode ainda ser bastante instável, ou seja, a estimativa realizada pelos mínimos quadrados pode produzir multiplicadores imprecisos ao considerarmos grandes quantidades, o que caracteriza o mau-condicionamento de  $A$ . Isto ocorre quando:

1. O espectro de uma biblioteca é semelhante ao de qualquer uma das outras bibliotecas, ou à combinação linear entre elas.
2. Existem algumas bibliotecas com contribuição desprezível para o espectro total de raios gama prontos.

O mau-condicionamento de  $A$  produzirá soluções altamente imprecisas para o sistema (2.1), reduzindo a sensibilidade do método MCLLS, nesses casos, e portanto a solução de mínimos quadrados  $\hat{f}$  será bastante sensível a pequenas mudanças nos argumentos das matrizes. Assim, Meric *et al.* [25] propuseram um tratamento para esse mau-condicionamento através de um método iterativo para MCLLS, levando em consideração o número de condicionamento da matriz  $A$ . Se esse número for próximo de 1.0, pode-se concluir que o sistema linear é bem-posto e que uma solução precisa pode ser obtida através de uma simples aplicação do método dos mínimos quadrados; caso contrário, quanto maior o número de condicionamento, mais mal-posto será o sistema linear.

### 3.2.2 O problema da medida em fluidos multifásicos

O fluido multifásico escolhido representa um problema bastante difícil, pois as bibliotecas da maioria de suas fases têm comportamento espectral semelhante, devido à presença de elementos como carbono e hidrogênio. Esta relação intrínseca produz um sistema altamente mal-condicionado. Explanaremos brevemente o experimento realizado nos estudos de Meric *et al.* [25] sem, no entanto, nos aprofundarmos sobre o método que eles propuseram.

Considere-se o fluido formado pela mistura homogênea de óleo, gás, água e sal. A composição das fases desse fluido é dominada pelos elementos carbono e hidrogênio, em quantidades variáveis. Segundo Speight [31], os percentuais desses elementos por fração de peso são: carbono, de 83 a 87 %, e hidrogênio, de 10 a 14 %. Além destes, o óleo cru pode conter quantidades não-significativas de metais, enxofre, nitrogênio e oxigênio. No sentido de reproduzir computacionalmente o meio físico, para efeito dos cálculos do MC, foram escolhidas as fórmulas químicas  $C_{18}H_{38}$ , para o óleo,  $CH_4$ , para o gás (metano),  $NaCl$  para o sal e  $H_2O$  para a água (pura). A tabela a seguir apresenta as quantidades elementares assumidas para cada uma das fases:

Tabela 3.1: Composição elementar de óleo, gás, água e sal, do fluido estudado.

Fase	Elemento	Fração em peso
Óleo	H	0.15
	C	0.85
Gás	H	0.25
	C	0.75
Água	H	0.1111
	O	0.8889
Sal	Na	0.3936
	Cl	0.6064

Fonte: Adaptada de Meric *et al.* [25] e de Wang *et al.* [35].

A amostra do fluido foi submetida a um detector de cintilação de iodeto de sódio (NaI), com bombardeamento de nêutrons de fissão espontânea de Califórnio ( $^{252}Cf$ ). As principais características dessa fonte, estão presentes na tabela a seguir:

Tabela 3.2: Características da fonte nêutrons de fissão espontânea  $^{252}Cf$ .

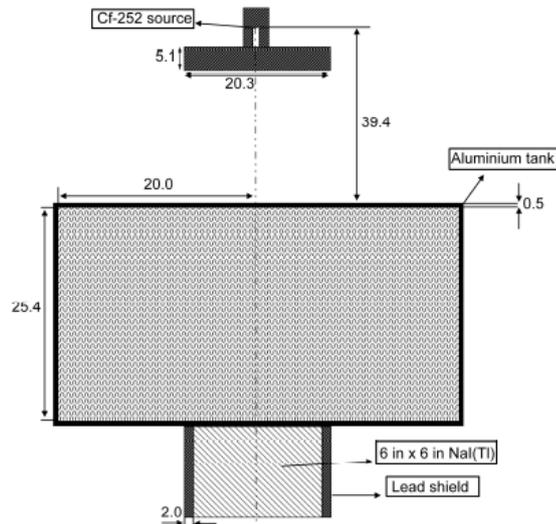
Modo de decaimento	Decaimento $\alpha$	96.6 %
	Fissão espontânea	3.1 %
Meia vida	Decaimento $\alpha$	$2.731 \pm 0.07$ anos
	Fissão espontânea	$85.5 \pm 0.5$ anos
	Média	$2.646 \pm 0.004$ anos
Rendimento de nêutrons de fissão espontânea ( $\#/\mu gs$ )	$2.34 \times 10^6$	
Energia média do nêutron (MeV)	$2.158 \pm 0.0017$	
$\bar{\nu}$ ( $\#$ de nêutrons/fissão espontânea)	$3.731 \pm 0.008$	
Rendimento dos raios gama de fissão ( $\#/\mu gs$ )	$1.3 \times 10^7$	

Fonte: Adaptado de Wang *et al.* [35].

O esquema a seguir ilustra o experimento realizado com a mistura citada. A amostra possui as frações de 40% de óleo, 40% de água salgada e 20% de gás, com fração em peso do sal na água

da ordem de 3.34%. Neste esquema, a fonte (*CF-252 source*) produz raios gama prontos de fissão ao emitir nêutrons de fissão espontânea, a amostra encontra-se no recipiente de alumínio (*Aluminium tank*), e o detector de cintilação, protegido por um "escudo de chumbo" (*lead shield*), está situado do lado oposto à fonte; as medidas apresentadas são dadas em centímetros.

Figura 3.1: Esquema de geração das bibliotecas elementares de raios gama prontos para MCLLS.



Fonte: Meric *et al.* [25].

A função do escudo é atenuar uma fração dos raios gama de fissão e portanto produzir uma correspondente redução do *background*<sup>3</sup> total, enquanto permite a entrada de nêutrons praticamente sem atenuações. Essa característica leva a crer que o uso de camadas mais espessas de chumbo poderia maximizar tal atenuação; porém, grandes quantidades de chumbo na fonte primária de nêutrons podem prover raios gama prontos extras, que devido à sua grande quantidade de energia são difíceis de bloquear, ocasionando o aparecimento de componentes não desejados no *background* do espectro final. Assim, na concepção de analisadores de PGNA, é comum a utilização de "escudos" tão finos quanto possível, evitando portanto a produção excessiva de "raios gama prontos parasitas" [25] (tradução nossa).

A tabela a seguir apresenta os dados obtidos do experimento (frações em peso dos elementos):

Tabela 3.3: Frações em peso resultantes dos elementos da mistura de óleo (40%), água salgada (40%) e gás (20%).

Elemento	Fração em peso	Densidade da mistura ( $g/cm^3$ )
H	0.153	0.7362
C	0.49	
O	0.3437	
Na	0.0052	
Cl	0.0081	

Fonte: Adaptada de Meric *et al.* [25].

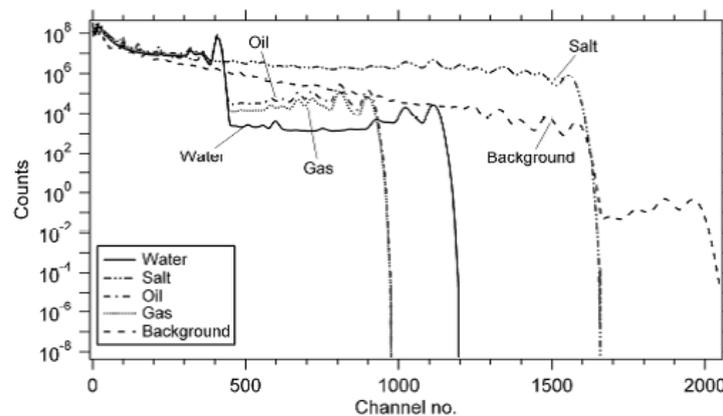
<sup>3</sup>O termo background refere-se ao espectro da soma de todas as substâncias desconhecidas e do ruído.

O background obtido consiste nas seguintes contribuições:

1. Dos raios gama produzidos dos materiais isolados (cercados).
2. Dos raios gama de fissão emitidos pela fonte de nêutrons  $^{252}\text{Cf}$ .

Embora existam outras contribuições, estas acima mencionadas são as únicas que interessam ao método utilizado. Para realizar a análise quantitativa, os espectros das bibliotecas individuais dos elementos pertinentes foram pesados e apropriadamente combinados para produzir a biblioteca da fase de interesse. Os espectros resultantes de óleo, água, gás, sal e *background* estão evidenciados a seguir:

Figura 3.2: Espectros das bibliotecas de raios gama prontos de óleo, sal, água, gás e *background*.



Fonte: Meric *et al.* [25].

Na figura acima os termos *Water*, *Salt*, *Oil*, *Gas* e *Background*, que aparecem na legenda, traduzem-se respectivamente como Água, Sal, Óleo, Gás e Segundo Plano (ou Plano de Fundo). O eixo horizontal (*Channel n<sup>o</sup>*) refere-se aos Canais com a respectiva numeração, e o eixo vertical (*Counts*) apresenta as Contagens dos níveis de energia das substâncias no espectro.

A partir deste ponto, Meric *et al.* [25] realizaram outros passos para completar a análise quantitativa através de MCLLS, combinando fases e verificando quais dessas associações produziu menores condicionamentos, utilizando todos os canais ou apenas aqueles correspondentes aos fotopicos.

Para o estudo que se segue, com base nos espectros mostrados na Figura 3.2, foi proposto um método de geometrização para esse problema, de modo a fazer uma seleção de intervalos de canais e a otimizar o condicionamento. Em particular, esta geometrização possibilita a regularização do problema e permite identificar a razão para o aparecimento de coeficientes negativos nas soluções de mínimos quadrados, conforme será posteriormente abordado.

## 4

# Geometrização do Problema da PGNAA em Fluidos Multifásicos

Neste capítulo, com base no problema do mau-condicionamento em MCLS para a PGNAA, descrito no capítulo anterior, será apresentada uma abordagem geométrica para os processos envolvidos na determinação de soluções. O método de geometrização desenvolvido neste estudo será descrito nas próximas seções através de duas abordagens: a primeira referir-se-á ao fluido mostrado na Figura (3.2), restrito ao caso com apenas três substâncias (água, gás e óleo), e a segunda é similar à primeira, mas, neste caso, com ênfase no fluido original com quatro fases (água, gás, óleo e sal).

## 4.1 Geometrização para o problema do fluido com três fases

O objetivo principal do processo de geometrização descrito nesta e na próxima seção, é estabelecer as condições para a criação de um método de regularização baseado nessa geometria, que além de diminuir o número de condicionamento da matriz de covariância, possa "prever" o comportamento do *background*. A ideia, para tal, é estabelecer pontos a partir da interseção entre conjuntos de canais e os gráficos dos espectros, e com isso construir vetores que representem cada uma das substâncias evidenciadas.

Inicialmente, foram escolhidas as fases água, gás e óleo, pois o espectro dessas três substâncias apresenta-se na mesma região, até aproximadamente o canal 1200 (de um total de 2048 canais). Em seguida esses 1200 canais foram divididos em três partições de dimensão de canais: 1 - 400, 401 - 800 e 801 - 1200, representadas respectivamente por X, Y e Z, em que cada uma é composta por 40 intervalos de coordenadas. Esses intervalos, por sua vez, são formados por conjuntos de 10 canais e ocupam papel central no processo de geometrização, pois é a partir deles que serão obtidos os valores das coordenadas  $x$ ,  $y$  e  $z$  dos vetores que representam as substâncias.

A partir dos dados gentilmente cedidos pelo professor Ilker Meric (autor principal do artigo em estudo), foram calculadas as integrais<sup>1</sup> dos níveis de energia (valores de contagem) referentes ao intervalo de coordenadas escolhido. Por exemplo, para o primeiro intervalo (canais 1 - 10), foram somados todos os valores de contagem de cada substância. No caso da água, por exemplo, nesse

---

<sup>1</sup>A integração mencionada refere-se à soma dos valores de contagem de cada substância, no intervalo considerado.



correspondentes (água, sal, óleo e gás). As somas desses valores para o primeiro conjunto de canais (aqui representado pelas células A2 - A11), encontram-se nas células HI11, JK11 e LM11 (destacados em fundo amarelo), e determinam as coordenadas  $x_1$ ,  $x_2$  e  $x_3$ , para os vetores  $A_{1jk}$ ,  $C_{1jk}$  e  $D_{1jk}$  ( $j, k = 1, \dots, 40$ )<sup>2</sup>, respectivamente. Portanto, a primeira coordenada do vetor  $A_{1jk}$  é dada por  $x_1 = 859800000$ , e analogamente para os vetores os valores  $C_{1jk}$  e  $D_{1jk}$  as primeiras coordenadas são, respectivamente,  $x_2 = 1168800000$  e  $x_3 = 1939000000$ .

Com o objetivo de simplificar a notação para os vetores, serão considerados a partir deste ponto os índices  $i = j = k$  (salvo em algumas exceções), e um vetor  $A_{ijk}$  será representado por  $Ai$ . Assim, por exemplo, o vetor  $A_{111}$  será representado por  $A1$ .

A determinação de vetores (constituídos como pontos do  $\mathbb{R}^3$ ) através desta abordagem, permite construir triângulos cujos vértices representam as três substâncias consideradas e possibilita determinar o plano gerado por uma escolha qualquer de vetores. Desse modo, qualquer ponto do plano pode ser representado como combinação baricêntrica única [34] dos vértices do triângulo. Considere-se então os vetores genéricos (representam quaisquer substâncias do fluido)  $A = (x_1, y_1, z_1)$ ,  $B = (x_2, y_2, z_2)$  e  $C = (x_3, y_3, z_3)$ , com  $A, B, C \in \mathbb{R}^3$ , não-nulos e não-colineares, e seja  $P(x, y, z)$  um ponto qualquer do plano gerado por eles. Assim  $P$  pode ser representado através de coordenadas baricênticas, ou seja, como combinação baricêntrica de  $A, B$  e  $C$ <sup>3</sup>:

$$P = \alpha A + \beta B + \gamma C; \quad \alpha + \beta + \gamma = 1, \quad (4.1)$$

em que  $\alpha, \beta, \gamma \in \mathbb{R}$ . Substituindo-se os valores de  $A, B$  e  $C$  na equação (4.1), obtém-se o sistema:

$$\begin{cases} \alpha x_1 + \beta x_2 + \gamma x_3 = x \\ \alpha y_1 + \beta y_2 + \gamma y_3 = y \\ \alpha z_1 + \beta z_2 + \gamma z_3 = z \\ \alpha + \beta + \gamma = 1. \end{cases} \quad (4.2)$$

Considerando-se os vértices  $A, B$  e  $C$ , a determinação dos vetores  $u = B - A$  e  $v = C - A$  produz, após alguns cálculos, o vetor normal e a equação do plano gerado pelo respectivo triângulo. O vetor normal é dado por:

$$n = \begin{bmatrix} (y_1 - y_2)(z_1 - z_3) - (y_1 - y_3)(z_1 - z_2), \\ (x_1 - x_3)(z_1 - z_2) - (x_1 - x_2)(z_1 - z_3) \\ (x_1 - x_2)(y_1 - y_3) - (x_1 - x_3)(y_1 - y_2) \end{bmatrix}, \quad (4.3)$$

e o plano gerado pelo triângulo de vértices  $A, B$  e  $C$ , que representam substâncias do fluido, tem equação:

$$\begin{aligned} & x(z_1 y_3 - z_3 y_1 + z_2 y_1 - z_1 y_2 + z_3 y_2 - z_2 y_3) + \\ & y(-x_1 z_3 - x_3 z_1 + x_1 z_2 + x_2 z_1 + x_3 z_2 - x_2 z_3) + \\ & z(x_3 y_1 - x_1 y_3 + x_2 y_3 - x_3 y_2 + x_1 y_2 - x_1 y_1) + \\ & ((x_1 - x_3)y_1 + (x_2 - x_1)y_2) z_1 = 0. \end{aligned} \quad (4.4)$$

Para exemplificar o comportamento dos triângulos obtidos nesse processo de geometrização com três fases, serão considerados os vetores formados a partir do quinto intervalo de coordenadas

<sup>2</sup>Os índices  $i, j, k = 1, \dots, 40$ , indicam o intervalo de coordenadas considerado nas partições X, Y e Z, respectivamente.

<sup>3</sup>A escolha deste tipo especial de combinação linear, em que os coeficientes somam 1, deve-se ao fato de que isto ocorre com o fluido multifásico em estudo, que é formado como uma espécie de combinação linear entre os compostos. Esta característica tem origem no problema principal, em que se deseja determinar a fração em peso das componentes.

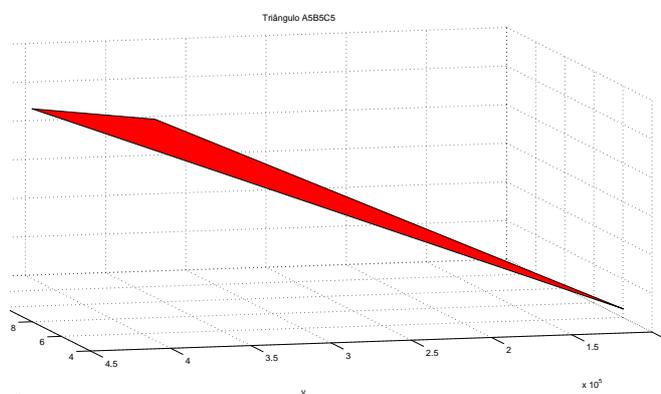
em X, Y e Z, que representam os conjuntos de canais 41 - 50, 441 - 450 e 841 - 850, formando o triângulo  $A5C5D5^4$ . Uma característica observada neste triângulo e em outros obtidos através do mesmo processo, é que eles são acutângulos e possuem pelo menos um dos ângulos bastante agudo, o que, em muitos casos, faz com que sua área seja quase degenerada.

Os valores obtidos da lista de dados são:

$$\begin{aligned} A5 &= (552100000, 104030, 15850), \\ C5 &= (749200000, 378100, 261900), \text{ e} \\ D5 &= (1244000000, 410000, 231000). \end{aligned}$$

Com o auxílio do MatLab<sup>5</sup>, foram determinadas a equação do plano (4.4) para os valores especificados acima, e a imagem:

Figura 4.3: Triângulo de vértices A5, C5 e D5.



O exemplo a seguir refere-se ao triângulo  $A2C2D2$  e ao plano gerado por ele. Os vetores  $u = C - A$ ,  $v = D - A$ , e o vetor normal ( $n$ ), encontram-se abaixo:

$$u = (408000000, 123800000, 2332230),$$

$$v = (1.4220 \times 10^9, 0.4411 \times 10^9, 0.0021 \times 10^9),$$

$$n = (-0.7741 \times 10^9, 2.4771 \times 10^9, 3.9252 \times 10^9).$$

O plano gerado a partir desses valores, é determinado pela equação:

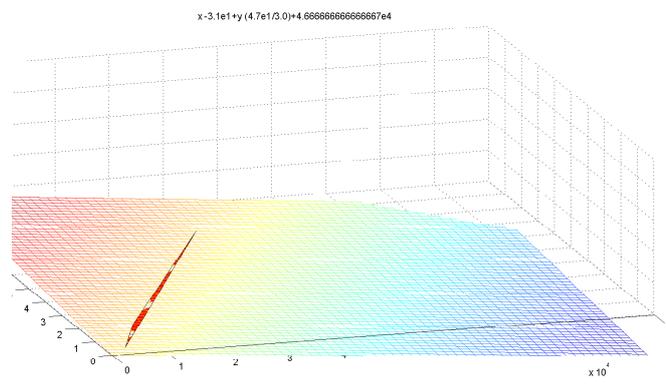
$$plano = 2477081220000000y - 7740615790000000x + 39252 \times 10^{11}z - 238615425 \times 10^{12}. \quad (4.5)$$

Similarmente ao triângulo da Figura 4.3, o triângulo  $A2C2D2$  apresenta-se bastante agudo, conforme pode ser visto na figura a seguir. Em geral, a escolha aleatória dos intervalos de coordenadas nas partições X, Y e Z produz o mesmo tipo de comportamento.

<sup>4</sup>Ao se considerar exemplos práticos com três fases, serão sempre usadas as substâncias representadas pelas colunas A, C e D da lista de dados, respectivamente água, óleo e gás.

<sup>5</sup>MatLab & Simulink Student Version Release 2013a é um programa MathWorks.

Figura 4.4: Triângulo de vértices A2, C2 e D2.



Triângulos muito agudos podem indicar que existem grandes diferenças de magnitude entre as coordenadas dos vetores de substâncias, ou ainda que alguns desses vetores possuem direções muito próximas entre si. Como será evidenciado mais adiante, tornar o triângulo o mais próximo possível de ser equilátero é parte da regularização proposta. Outra providência seria tentar fazer com que as arestas do triângulo tivessem magnitudes próximas dos tamanhos dos vetores extraídos dos vértices.

## 4.2 Geometrização para o problema do fluido com quatro fases

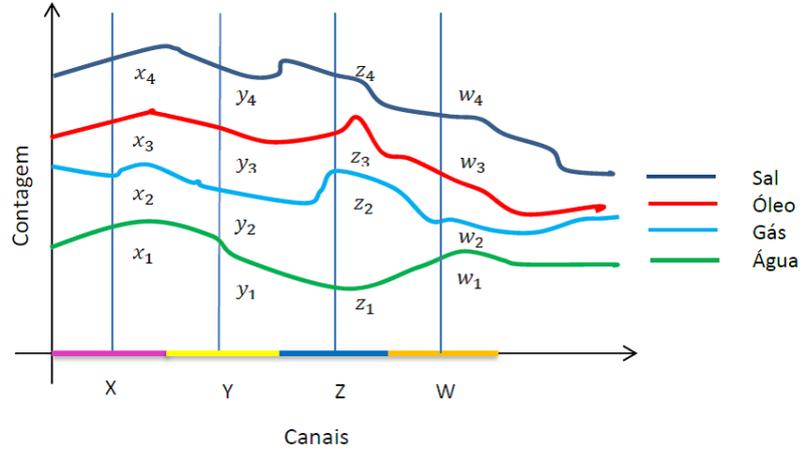
De modo análogo ao que foi desenvolvido na seção anterior, será considerado o problema do fluido multifásico apresentado em Meric *et al.* [25], agora com as quatro bibliotecas: água, sal, óleo e gás. Neste caso, os 1200 primeiros canais<sup>6</sup> foram divididos em quatro partições, X, Y, Z e W, sendo cada uma delas composta por 30 intervalos de 10 canais, de onde serão obtidas as coordenadas  $x_i, y_i, z_i$  e  $w_i$  ( $i = 1, 2, 3, 4$ ), dos vetores que representam as substâncias do espectro. Essa divisão, baseada na percepção humana dos gráficos dos espectros, não é única: existem outras formas, inclusive automatizadas, para realizá-la. Há ainda muitas escolhas de canais extraídos a partir dessas quatro divisões, que produzem conjuntos de vetores linearmente independentes, a depender dos gráficos dos espectros.

É possível então imaginar que com quatro vetores, sendo aqui interpretados como pontos do  $\mathbb{R}^4$ , obtenha-se um tetraedro em algum subconjunto desse espaço. Esses pontos, sendo não-nulos e não-coplanares, irão gerar, através de combinações baricêntricas, uma variedade  $\mathfrak{W}$  isomorfa ao  $\mathbb{R}^3$ .

A Figura (4.5), analogamente à Figura 4.1, apresenta um esquema ilustrativo para a obtenção das coordenadas dos vetores de substâncias, que utiliza o mesmo modelo de integração do problema com três fases (a soma dos valores de contagem por substância, em cada intervalo), considerando a mesma lista de dados do caso anterior (ver figura 4.2).

<sup>6</sup>Os demais canais pois foram desconsiderados pois, a partir deste marco, apenas uma das substâncias (o sal) apresenta contagem dos níveis de energia.

Figura 4.5: Esquema de obtenção de coordenadas - Problema com quatro fases.



No esquema acima, o eixo das abscissas apresenta as quatro partições X, Y, Z e W, e as linhas verticais representam os intervalos de coordenadas, compostos por 10 canais. Os valores de  $x_i, y_i, z_i$  e  $w_i$  ( $i = 1, 2, 3, 4$ ), que representam as coordenadas dos vetores, por substância, localizam-se nas interseções entre os intervalos e os gráficos do espectros.

Seja  $P(x, y, z, w)$  um ponto qualquer de  $\mathcal{W}$ . É possível então representar  $P$  em coordenadas baricêntricas:

$$P = \alpha A + \beta B + \gamma C + \delta D; \quad \alpha + \beta + \gamma + \delta = 1, \quad (4.6)$$

com  $\alpha, \beta, \gamma, \delta \in \mathbb{R}$ . Substituindo-se os valores genéricos de  $A, B, C$  e  $D$  em (4.6), obtém-se o sistema:

$$\begin{cases} \alpha x_1 + \beta x_2 + \gamma x_3 + \delta x_4 = x \\ \alpha y_1 + \beta y_2 + \gamma y_3 + \delta y_4 = y \\ \alpha z_1 + \beta z_2 + \gamma z_3 + \delta z_4 = z \\ \alpha w_1 + \beta w_2 + \gamma w_3 + \delta w_4 = w \\ \alpha + \beta + \gamma + \delta = 1. \end{cases} \quad (4.7)$$

Na seção a seguir será discutido um novo modelo de regularização, proposto a partir das informações obtidas do processo de geometrização acima descrito.

### 4.3 Regularização Geométrica

A ideia de regularização pressupõe a resolução de um problema inverso mal-posto, tentando inverter sistemas não-invertíveis ou próximos da invertibilidade ou da singularidade, determinando parâmetros para que ele se torne o mais próximo possível de um problema bem-posto. Com base no processo de geometrização desenvolvido para o problema da medida do fluido multifásico na PGNAA, foi elaborado um método de regularização (denominado de Regularização Geométrica), que leva em consideração as medidas das arestas do triângulo ou do tetraedro, a depender do caso, o condicionamento da matriz de coeficientes do sistema linear obtido do processo de geometrização, e ainda estabelece um modelo de referência para a solução aproximada, caracterizando-se como um

caso particular da regularização de Tikhonov. Diferentemente do método usado por Meric *et al.* [25] e de outros métodos que utilizam todos os canais ou apenas aqueles correspondentes aos fotopicos, a regularização geométrica procura, semelhantemente ao que foi proposto por Cardoso [6], realizar uma seleção dos intervalos de coordenadas, otimizando o condicionamento.

Conforme mostrado na seção anterior, os triângulos produzidos a partir dos vetores obtidos do processo de geometrização do problema estudado, em geral, possuem pelo menos um ângulo muito agudo, o que significa que as distâncias entre os vértices de um mesmo triângulo devem ser bastante diferentes. Do mesmo modo, para o caso com quatro substâncias, é possível supor que os vértices dos tetraedros apresentem o mesmo comportamento em relação à distância entre seus vértices. Para facilitar a descrição do método será realizada, inicialmente, uma análise do problema com três fases e em seguida o mesmo será feito para o caso com quatro fases, tomando-se alguns vetores como exemplos e observando-se o condicionamento das matrizes de coeficientes nos sistemas (4.2) e (4.7), para os respectivos casos.

### 4.3.1 Regularização geométrica para o problema com três fases

Inicialmente, serão utilizados alguns vetores específicos para que seja possível avaliar o condicionamento das matrizes de coeficientes nos sistemas lineares formados a partir desses vetores. Considere-se então o exemplo apresentado na Figura 4.3, em que os vértices do triângulo correspondem aos vetores  $A_5$ ,  $C_5$  e  $D_5$ . Substituindo-os em (4.2), a matriz é obtida:

$$M_{555} = \begin{pmatrix} 552100000 & 749200000 & 1244000000 \\ 104030 & 378100 & 410000 \\ 15850 & 261900 & 231000 \\ 1.0 & 1.0 & 1.0 \end{pmatrix}. \quad (4.8)$$

Os índices em  $M_{ijk}$  representam, respectivamente, os intervalos de coordenadas escolhidos em X, Y e Z. Após fazer a decomposição em valores singulares (DVS) de  $M_{555}$ , a matriz dos valores singulares encontrada é:

$$S_{555} = \begin{pmatrix} 1.4532 \times 10^9 & 0 & 0 \\ 0 & 0.0030 \times 10^9 & 0 \\ 0 & 0 & 0.0001 \times 10^9 \end{pmatrix}. \quad (4.9)$$

Calculando-se o condicionamento de (4.8) através de (2.5), obtém-se  $\kappa = 14532$ . Esse número pode ser considerado como um modo de medir o quanto o sistema é mal-posto, e em geral pode denunciar uma "patologia" no formato do triângulo. Assim, quanto mais distante de 1.0 for o condicionamento, mais mal-posto será o problema e, conseqüentemente, mais agudo o triângulo pode se tornar. Cálculos similares com diversos outros intervalos de coordenadas corroboram com a ideia de que nesses dados da aplicação em pauta o mau-condicionamento pode ser revelado pela agudeza do triângulo.

Com o objetivo de estabelecer uma regularização para esse tipo de problema, foi proposta uma mudança de escala na tentativa de equilibrar as medidas dos lados dos triângulos e, conseqüentemente, buscar que tanto lados quanto ângulos tenham os respectivos valores o mais próximo possível entre si. Desse modo, concebendo-se a regularização geométrica como um caso especial da

regularização de Tikhonov, que estabelece uma solução (modelo) de referência para as aproximações, o modelo de referência definido no método proposto para o problema com três fases foi o triângulo equilátero. Assim, as soluções regularizadas devem tender a esse modelo.

A mudança de escala mencionada consiste no processo de normalização de dados e otimização de escolhas de intervalos, com a finalidade de determinar aqueles cujo triângulo representante é o mais próximo de ser equilátero. Este processo representa uma das etapas fundamentais da regularização geométrica, cuja expressão geral será apresentada mais adiante. Essa normalização consiste em dividir todas as somas<sup>7</sup> dos valores de contagem por substâncias, em um intervalo de coordenadas, pela maior das somas obtidas, produzindo assim intervalos normalizados.

Por exemplo, no primeiro intervalo de coordenadas em X, a maior das somas (Figura 4.2) é 1939000000, referente à substância C. Este é o quociente de normalização, ou seja, é o valor pelo qual as somas de A, C e D foram divididas. Analogamente, foram calculados os quocientes de normalização para cada um dos intervalos de coordenadas em X, Y e Z. Estes cálculos foram efetuados a partir de um código computacional (escrito em Python 2.7.9)<sup>8</sup>, que produziu uma lista de dados normalizados (Apêndice E). Com esses novos valores, é possível reescrever as matrizes e recalculá-las o seu condicionamento, além de reconstruir os respectivos triângulos. Para efeito de comparação, analise-se o triângulo  $A5B5C5$ , agora os valores normalizados. A notação utilizada para representar os novos vetores, referentes às fases A, C e D é  $A5n$ ,  $C5n$  e  $D5n$ , cujos valores:

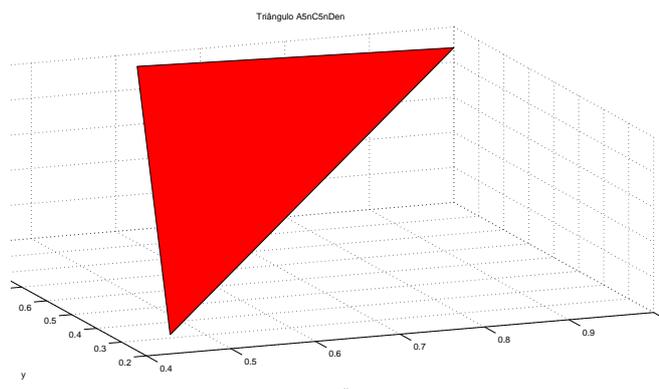
$$A5n = (0.4438102893890675, 0.2537317073170732, 0.0605192821687667),$$

$$C5n = (0.6022508038585209, 0.9221951219512196, 1.0), \text{ e}$$

$$D5n = (1.0, 1.0, 0.8820160366552119),$$

produzem o triângulo:

Figura 4.6: Triângulo de vértices  $A5n$ ,  $C5n$  e  $D5n$ .



Ao substituir as coordenadas desse triângulo no sistema (4.2), e efetuar a decomposição DVS da matriz de coeficientes, obtém-se a matriz de valores singulares:

<sup>7</sup>As referidas somas representam as integrais ao longo de um intervalo de coordenadas.

<sup>8</sup>Este código pode ser visto no Apêndice B.

$$S_{555n} = \begin{pmatrix} 2.7858 & 0 & 0 \\ 0 & 0.6490 & 0 \\ 0 & 0 & 0.2719 \\ 0 & 0 & 0 \end{pmatrix}. \quad (4.10)$$

Assim, o condicionamento do sistema referente à escolha considerada torna-se  $\kappa = 10,2456$ , ou seja, verifica-se uma enorme diminuição no condicionamento, que antes desta etapa da regularização geométrica era  $\kappa = 14532$ . Este processo, como já mencionado, não torna os triângulos equiláteros, mas proporciona uma melhora no seu comportamento, como pode ser visto na Figura (4.6).

A realização de todas as combinações possíveis<sup>9</sup> de intervalos, possibilita determinar qual dos triângulos tem maior equilíbrio nas medidas das arestas, ou seja, serve para identificar qual combinação formada por um intervalo de cada partição representa o triângulo mais próximo de ser equilátero.

A figura abaixo apresenta um recorte do código *analise\_substancia\_3.py* (Apêndice B), que foi desenvolvido para efetuar o mencionado processo de escolha de intervalos.

Figura 4.7: Recorte do código *analise\_substancia\_3.py*.

```

analise_substancia_3.py - C:\Users\Fabiano\Desktop\Fabiano\analise_substancia_3.py (2.7.9)
File Edit Format Run Options Windows Help
# -*- coding: UTF-8 -*-

#dados normalizados no codigos

import xlrd

arq = xlrd.open_workbook('4fases_dados.xls')

plan = arq.sheets()[0]

#=====
#=====
#=====
#Função da distancia no R3
def distr3(a,b):
    d = ((a[0]-b[0])**2 + (a[1]-b[1])**2 + (a[2] - b[2])**2)**(0.5)
    return d

def calculaEpsilon1():
    #FORMULA DO EPSILON
    for i in range(n):
        e = (max(distancias[i][1]) - min(distancias[i][1]))/sum(distancias[i][1])
        ep.append([e,distancias[i][0]])

def calculaEpsilon2():
    #FORMULA DO EPSILON
    for i in range(n):
        a1 = distancias[i][1][0]#VALOR DOS LADOS
        a2 = distancias[i][1][1]
        a3 = distancias[i][1][2]

        f = ((a1-a2)**2 + (a1-a3)**2 + (a2-a3)**2)/sum(distancias[i][1])
        ep.append([f,distancias[i][0]])

```

As duas métricas ( $\epsilon$ ) evidenciadas no código (na execução do código, apenas uma delas deve ser escolhida), apesar de apresentarem formulações diferentes, determinam a mesma solução

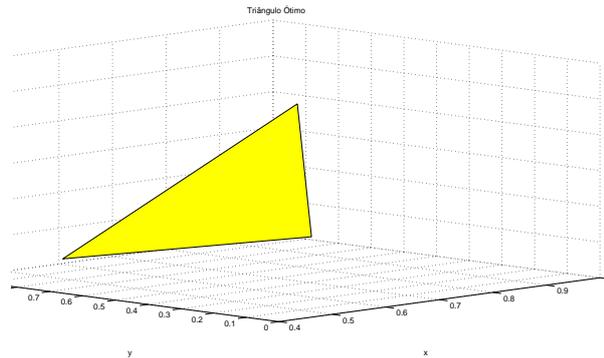
<sup>9</sup>A determinação do melhor triângulo considera todas as escolhas de intervalos em cada partição e efetua uma busca, determinando quais desses intervalos representam o triângulo mais próximo do modelo de referência. Há outras possibilidades para a realização dessa busca, como por exemplo, fazer com que os intervalos variem com sobreposição de canais. Neste caso, o número de combinações seria significativamente maior.

e servem para indicar grau de proximidade entre as medidas das arestas dos triângulos obtidos em cada combinação. A primeira formulação considera a razão entre a diferença da maior e da menor aresta de cada triângulo, pela soma de todas as arestas; a segunda efetua a razão entre as somas dos quadrados dos comprimentos das arestas de cada triângulo, e a soma de todas as arestas, como pode ser visto na Figura (4.7) em # *Fórmula do Epsilon*. Assim, quanto mais próximo  $\epsilon$  se tornar de zero, mais equilibradas serão as medidas dos triângulos.

Após a execução do código *analise\_substancia\_3.py*, os resultados obtidos foram:

1. Valor mínimo  $\epsilon = 0,223576302692$ , encontrado na combinação número 38372, que corresponde aos intervalos de coordenadas 23, 39 e 12, respectivamente nas partições X, Y e Z da lista de dados normalizados.
2. A matriz resultante dos dados indicados no processo de otimização possui condicionamento  $\kappa = 9,8145132$ .
3. O triângulo da figura a seguir é formado pelos vértices obtidos a partir dos intervalos 23, 39 e 12, indicados após a execução do código.

Figura 4.8: Triângulo resultante de normalização e otimização de dados.



O interesse na escolha do modelo de referência é que, por si, ele já resolve boa parte do mau-condicionamento. Como já referido, o modelo de regularização proposto busca o equilíbrio entre as arestas dos triângulos e gera uma enorme quantidade de combinações entre os intervalos de coordenadas. Cada combinação representa um problema mal-posto, pois as matrizes de coeficientes de todos os sistemas obtidos são retangulares, e os dados utilizados são contaminados por ruídos. Isto indica a necessidade de aplicação do método dos mínimos quadrados para a obtenção de cada solução. Analiticamente, os problemas advindos das  $k$  ( $k \in \mathbb{N}$ ) escolhas de intervalos de coordenadas, são representados, em notação matricial, como:

$$B_k \cdot \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \\ 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \quad (4.11)$$

em que  $B_k$  é a matriz diagonal

$$B_k = \begin{pmatrix} \frac{1}{\max_k(x_1, x_2, x_3)} & 0 & 0 \\ 0 & \frac{1}{\max_k(y_1, y_2, y_3)} & 0 \\ 0 & 0 & \frac{1}{\max_k(z_1, z_2, z_3)} \end{pmatrix}. \quad (4.12)$$

Utilizando-se (2.1) e levando-se em conta cada escolha  $k$ , obtém-se a seguinte reescrita para (4.11):

$$A_k f_k = g_k. \quad (4.13)$$

Todos os  $k$  problemas representados por (4.13) são mal-postos e, portanto, é necessário determinar, em cada caso, uma solução de mínimos quadrados. Considerando-se (2.18), essas soluções são dadas por:

$$\hat{f}_k = (A_k^T A_k)^{-1} A_k^T g_k. \quad (4.14)$$

Efetuar a regularização tomando como base as medidas das arestas dos triângulos, significa determinar aquele que mais se aproxima do modelo de referência. Considerando-se (2.45), e utilizando-se  $m^{ref}$  para representar um triângulo equilátero e  $m_k$  para um triângulo qualquer, dado pela escolha  $k$ , obtém-se:

$$\|m_k - m^{ref}\|_2^2. \quad (4.15)$$

A regularização geométrica define-se então como o processo de regularização que considera a solução de mínimos quadrados e a aproximação ao modelo de referência, para cada escolha de intervalos (4.15). Sua expressão geral, a seguir, representa um caso especial da regularização de Tikhonov (2.45):

$$\min_k \left\{ \left\| A_k (A_k^T A_k)^{-1} A_k^T g_k - g_k \right\|_2^2 + \lambda^2 \|m_k - m^{ref}\|_2^2 \right\} \quad (4.16)$$

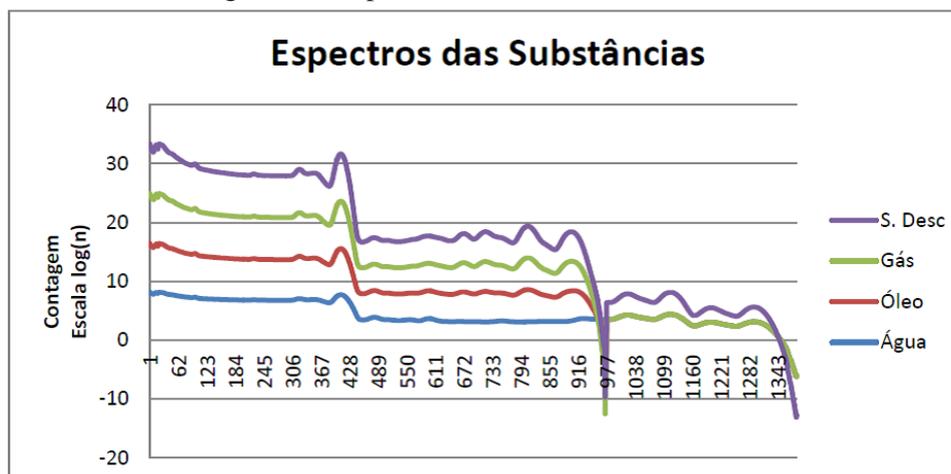
A primeira parte da expressão (4.16) refere-se principalmente aos dados, com a utilização das soluções de mínimos quadrados, a segunda determina aproximações ao modelo de referência, e o termo  $\lambda$  é o parâmetro de regularização. Esta regularização está definida para os problemas estudados com três e quatro fases, e pode naturalmente ser generalizada para uma quantidade maior de substâncias, além de possibilitar o uso da teoria apresentada por Fernández-Martínez [13], com as devidas adaptações, para interpretar os efeitos da regularização geométrica em problemas lineares inversos, o que se constitui numa proposta de trabalho futuro.

No sentido de ilustrar o funcionamento de (4.16), uma nova substância ( $P$ ) foi gerada a partir das já existentes, através do código `geradorSubstancia4.py` (Apêndice D). O espectro dessa nova substância consiste na soma entre percentuais das três fases consideradas, adicionando-se ainda um ruído aleatório a essa soma em cada um dos 2048 canais, produzindo assim uma lista de dados perturbados. Os percentuais utilizados para a construção de  $P$  foram de 20%, 30% e 50% para as substâncias  $A$ ,  $C$  e  $D$ , respectivamente, e em cada canal o ruído aleatório variou entre 0 a 20% do valor da respectiva soma.

A figura a seguir representa graficamente os espectros das substâncias em escala logarítmica, considerando a nova fase. Nesta imagem os valores situados no eixo horizontal representam os canais (1600, de um total de 2048), no eixo vertical aparecem os valores de contagem dos níveis de energia

para as substâncias, e a legenda evidencia as substâncias *A*, *C*, *D* e *P*. Devido à escala logarítmica utilizada para a construção do gráfico e aos valores das coordenadas (que variam de quantidades positivas muito próximas de zero a grandezas da ordem de  $10^9$ ), alguns canais apresentaram números de contagem negativos nesta representação gráfica, fato que não ocorre em escala original.

Figura 4.9: Espectros das substâncias A, C, D e P.



O processo de normalização dos dados, aplicado aos valores de *P* em todos os intervalos, produziu uma lista de dados normalizados para essa substância desconhecida, utilizados em seguida na execução da regularização geométrica. Para tal, os valores de entrada são as *k* escolhas de intervalos de coordenadas, o valor do parâmetro  $\lambda$ , que pode variar segundo se deseje mais influência da busca pelo triângulo mais próximo de ser equilátero ou se deseje mais peso para as soluções de mínimos quadrados, e ainda as coordenadas da substância desconhecida, em cada intervalo.

A norma presente na primeira parte de (4.16) refere-se à norma euclidiana em  $\mathbb{R}^3$  para cada escolha *k* de intervalos, posto que a distância entre os pontos ( $P'_k$ ) que representam as soluções de mínimos quadrados  $A_k(A_k^T A_k)^{-1} A_k^T g_k$  e o valor da respectiva substância desconhecida  $P_k$ <sup>10</sup>, determina a projeção ortogonal de  $P_k$  sobre o plano formado pelos vértices *A*, *C* e *D*.

Para o cálculo da regularização foi escolhido, inicialmente, o parâmetro  $\lambda = 0.1$ . Assim, após a execução do programa, os resultados obtidos foram:

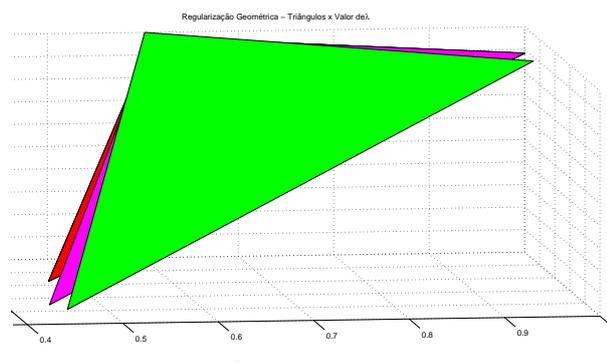
1. Valor Mínimo: 0.251898168195.
2. Combinação número: 27361.
3. Intervalos de coordenadas: 17 , 4 , 1.

Este resultado indica que a minimização desejada foi obtida na combinação 27361, para o parâmetro de regularização considerado. Utilizando-se maiores valores para  $\lambda$ , o peso da regularização geométrica recai gradualmente sobre a segunda parte do processo, dando mais importância à aproximação ao modelo de referência do que às soluções de mínimos quadrados.

<sup>10</sup>Nas formulações utilizadas neste estudo para calcular a regularização geométrica e a solução de mínimos quadrados, a substância desconhecida  $P_k$  é representada em cada escolha como  $g_k$ .

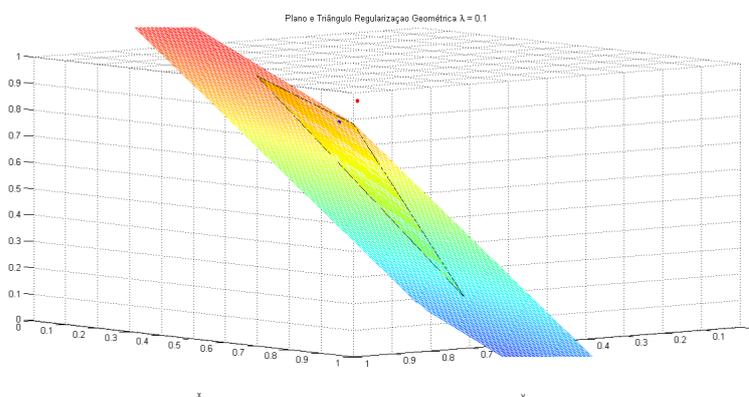
A título de ilustração são apresentadas duas figuras, a seguir: A primeira exhibe o triângulo verde, obtido da regularização geométrica com  $\lambda = 0.1$ , e os triângulos vermelho e magenta, produzidos a partir dos parâmetros  $\lambda = 0.03$  e  $\lambda = 0.3$ , respectivamente. A segunda figura apresenta o mesmo triângulo, com  $\lambda = 0.1$ , o plano gerado por ele, e os pontos que representam a substância desconhecida e a solução de mínimos quadrados.

Figura 4.10: Regularização Geométrica - Triângulos  $\times$  Valor de  $\lambda$ .



Os triângulos obtidos com os diferentes parâmetros de regularização possuem comportamento bastante semelhante e, devido às pequenas diferenças existentes entre as coordenadas dos vértices correspondentes, localizam-se bastante próximos.

Figura 4.11: Plano, triângulo, substância desconhecida e solução de mínimos quadrados. Regularização geométrica com  $\lambda = 0.1$ .



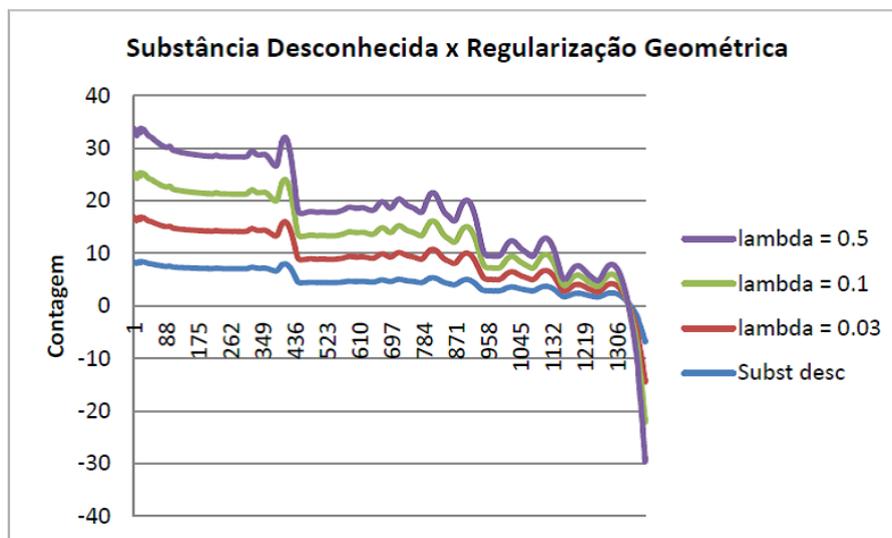
Na figura acima, o ponto referente à substância desconhecida  $P$  está representado em vermelho e o da solução de mínimos quadrados  $P'$  está em azul. Os seus respectivos valores são:

$$P = (0.8939632212885154, 0.9496938048780489, 0.8685829914061073) \text{ e}$$

$$P' = (0.04600301, 0.20346108, 0.75053051).$$

É possível ainda produzir os espectros da substância desconhecida regularizada, em função de alguns valores de  $\lambda$ . Foram escolhidos, para tal,  $\lambda = 0.03$ ,  $\lambda = 0.1$  e  $\lambda = 0.5$ . Os resultados são mostrados na figura a seguir:

Figura 4.12: Espectros da substância desconhecida para diferentes escolhas do parâmetro de regularização

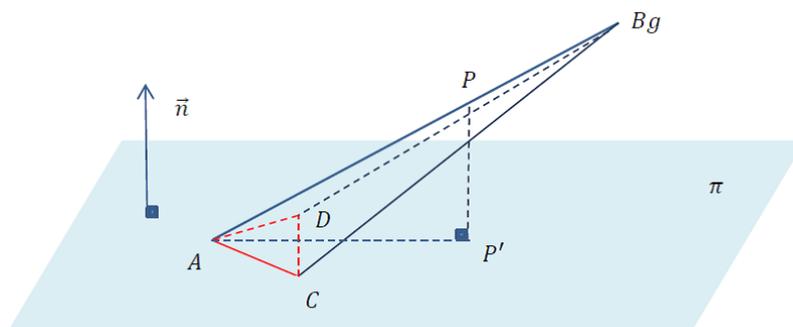


Analogamente à Figura 4.9, os valores apresentados no gráfico acima encontram-se em escala logarítmica, posto que as grandes diferenças de magnitude dos valores de contagem dificultam a representação dos espectros com os valores originais. A escala utilizada produziu, neste caso, resultados não desejados (com valores negativos a partir de um dado canal).

### 4.3.2 Estimativa do *background*

Um dos principais resultados do método de regularização geométrica para o problema com três fases é a estimativa do *background*. Este é definido como o conjunto formado pelas substâncias que se apresentam no fluido em quantidades quase desprezíveis, juntamente com substâncias desconhecidas e com o ruído nos dados. Nesta seção, o objetivo é estabelecer analiticamente os limites para a presença do *background* no espectro. Considere-se primeiramente o esquema:

Figura 4.13: Esquema de localização do *background* para o problema com três fases.



Na figura acima, o triângulo  $ACD$  está evidenciado em vermelho, tendo como vértices  $A = (x_1, y_1, z_1)$ ,  $C = (x_2, y_2, z_2)$  e  $D = (x_3, y_3, z_3)$ ,  $P = (x, y, z)$  representa uma substância desconhe-

cida,  $P' = (x', y', z')$  é a solução de mínimos quadrados (que corresponde à projeção ortogonal de  $\overline{AP}$  sobre o plano  $\pi$ ),  $\vec{n}$  é o vetor normal a  $\pi$ , e  $Bg$  representa o *background*. Neste esquema, a substância desconhecida  $P$  não pertence ao plano, pois, se isto acontecesse, ela poderia ser escrita como uma combinação baricêntrica dos vértices do triângulo  $ACD$ , como já visto. Então a solução de mínimos quadrados  $P' \in \pi$  e pode ser escrita como  $P' = \alpha A + \beta C + \gamma D$ , com  $\alpha + \beta + \gamma = 1$ .

É necessário ainda considerar que as coordenadas de  $P'$  nem sempre são positivas pois, se isto ocorresse,  $P'$  pertenceria ao interior do triângulo. Na Figura 4.13 a solução de mínimos quadrados encontra-se fora do triângulo, o que nos leva a afirmar que alguma das coordenadas é negativa. Isto representa uma vantagem, pois serve como o indicativo da existência de uma substância desconhecida. Obviamente o ruído pode promover o mesmo efeito, deixando  $P'$  fora do triângulo; neste caso, espera-se que a distância ente essa solução e o triângulo seja muito pequena.

Considerando-se a relação entre os vértices do triângulo e o *background*, e ainda supondo-se que  $P \in \overline{ABg}$ , é possível imaginar que existe uma certa liberdade para a localização de  $Bg$  pois, se sua posição variar,  $P$  poderá ser localizado, no máximo, sobre as arestas ou faces do tetraedro (pois a distância entre  $P$  e  $P'$ , além de ser a menor possível, é invariável). Por exemplo, se o vértice  $Bg$  fosse deslocado para cima, as arestas e faces laterais também se moveriam nesse sentido. Como consequência,  $P$  estaria localizado no interior ou sobre a face inferior do tetraedro. O deslocamento similar para os lados, determina comportamento semelhante.

Analicamente, os limites do *background* define-se através da relação de colinearidade entre pontos. Assumindo-se que  $A$ ,  $P$  e  $Bg$  são colineares, tem-se:

$$\overline{AP} = \delta \overline{ABg}, \quad (4.17)$$

com  $\delta \in \mathbb{R}$ ;  $0 < \delta < 1$ . Assim,

$$P - A = \delta(Bg - A), \quad (4.18)$$

e portanto,

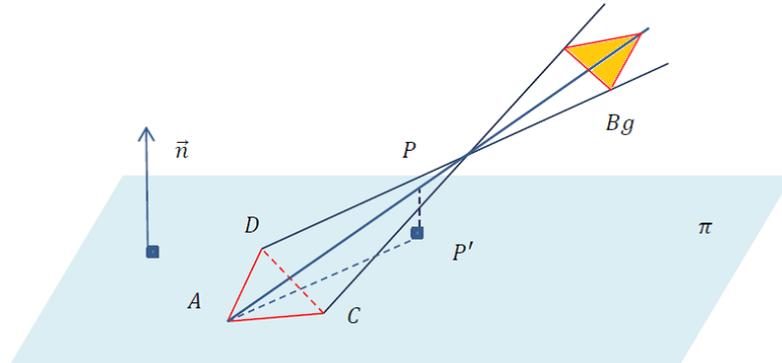
$$Bg = \frac{1}{\delta}P + \left(1 - \frac{1}{\delta}\right)A \quad (4.19)$$

representa os valores obtidos para o *background* em função de  $P$  e de  $A$ . Analogamente, os valores de  $Bg$  são determinados em função dos vértices  $C$  e  $D$ , a seguir:

$$Bg = \frac{1}{\delta}P + \left(1 - \frac{1}{\delta}\right)C, \quad (4.20)$$

$$Bg = \frac{1}{\delta}P + \left(1 - \frac{1}{\delta}\right)D. \quad (4.21)$$

Como consequência dessas relações, pode-se escolher  $\delta$  de modo a delimitar uma região no gráfico dos espectros das substâncias (com tolerância de 10%, por exemplo), determinando uma região triangular plana em que o *background* está contido, conforme o esquema:

Figura 4.14: Esquema da região de localização do *background* para o problema com três fases.

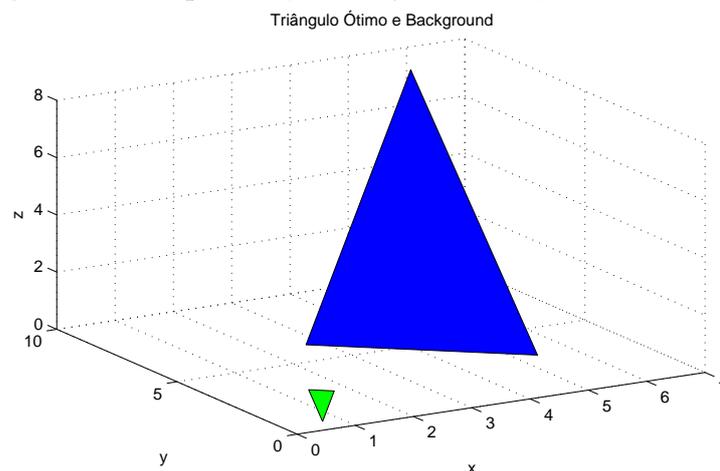
As equações (4.19), (4.20) e (4.21) foram utilizadas no código `analise_substancia_3.py` para calcular os limites do *background* para a escolha de intervalos indicada. Por exemplo, para o triângulo resultante da otimização (busca pela combinação de intervalos que gera o triângulo mais próximo de ser equilátero), determinado pelos canais 23, 39 e 12, respectivamente nas partições X, Y e Z, supondo-se que a substância desconhecida  $P = (p_0, p_1, p_2) = (1, 1, 1)$  e que  $\delta = 0.1$ , os limites do *background* obtidos foram:

BgA = (6.041741534, 9.924506579, 7.234196267),

BgC = (4.536558397, 1, 1),

BgD = (1, 2.057249494, 2.065021072).

O triângulo ótimo (em verde) e a região do seu respectivo *background* (triângulo azul), são mostrados na figura a seguir:

Figura 4.15: Triângulo ótimo e representação da região do *background* - Problema com três fases.

### 4.3.3 Regularização geométrica para o problema com quatro fases

A regularização geométrica (4.16) para o problema com quatro fases é uma generalização natural do caso com três fases, diferenciando-se principalmente por considerar vetores com quatro coordenadas,

devido ao número de substâncias e de partições de intervalos, e por estabelecer um tetraedro regular como modelo de referência. Deste modo, esta seção deter-se-á apenas ao processo de busca pelo tetraedro que mais se aproxima desse modelo.

Os princípios utilizados para a realização do processo citado são os mesmos apresentados na Seção 4.3.1. Para tal, além das substâncias estudadas anteriormente (*A*, *C* e *D*), foi acrescentada a substância *B* (sal), considerando-se as partições *X*, *Y*, *Z* e *W*. Para efeito de comparação, antes de ser efetuada a normalização dos dados, considere-se os intervalos *A3B3C3D3*, por exemplo, cujos correspondentes valores são:

$$\begin{aligned} A3 &= (1262000000, 102210000, 16420, 45940), \\ B3 &= (2536000000, 52640000, 24100000, 23590000), \\ C3 &= (1716000000, 138300000, 514600, 149490), \text{ e} \\ D3 &= (2844000000, 230000000, 453900, 131800). \end{aligned}$$

Substituindo-os no sistema (4.7), obtém-se a matriz

$$M_{3333} = \begin{pmatrix} 1262000000 & 2536000000 & 1716000000 & 2844000000 \\ 102210000 & 52640000 & 138300000 & 230000000 \\ 16420 & 24100000 & 514600 & 453900 \\ 45940 & 23590000 & 149490 & 131800 \\ 1.0 & 1.0 & 1.0 & 1.0 \end{pmatrix}, \quad (4.22)$$

cujo condicionamento, calculado a partir de (2.5), é dado por  $\kappa(M_{3333}) = 41884$ , e indica um sistema muito mau-condicionado.

Com o objetivo de balancear as medidas das arestas do tetraedro, e ainda diminuir o condicionamento das matrizes de coeficientes dos sistemas produzidos pelas escolhas aleatórias de conjuntos de canais, efetuou-se a normalização dos dados de contagem, analogamente ao que foi feito no caso com três fases, dividindo-se as somas ao longo de cada intervalo pelo maior valor delas, através do código *analise\_substancias4.py* (Apêndice C), cujo recorte é apresentado a seguir:

A execução desse código gerou uma lista com os dados normalizados. Para os intervalos utilizados no exemplo acima, foram construídos os vetores:

$$\begin{aligned} A3n &= (0.4437412095639944, 0.4443913043478261, 0.0006813278008298755, \\ &\quad 0.0019474353539635438), \\ B3n &= (0.8284106891701828, 0.2288695652173913, 1.0, 1.0), \\ C3n &= (0.6033755274261603, 0.601304347826087, 0.02133609958506224, \\ &\quad 0.006337007206443408), \text{ e} \\ D3n &= (1.0, 1.0, 0.01883402489626556, 0.0055871131835523525). \end{aligned}$$

Ao substituir esses vetores no sistema (4.7) e calcular o condicionamento da matriz de coeficientes do sistema obtido, o resultado, embora ainda distante de 1.0, revelou-se significativamente menor do que o condicionamento da matriz de dados originais. O novo condicionamento é:  $\kappa(M_{3333}) = 406, 2571$ .

A execução do código realizou todas as combinações possíveis para a escolha dos intervalos de coordenadas em cada partição, determinando, através de uma métrica similar à apresentada na seção anterior, a melhor entre todas as escolhas, ou seja, identificando quais são os intervalos que, combinados, representam um tetraedro com a menor diferença entre as medidas de suas arestas

Figura 4.16: Recorte do código `analise_substancias4.py` - Problema com quatro fases.

```

*analise_substancias4.py - C:\Users\Fabiano\Desktop\Fabiano\analise_substancias4.py (2.7.9)
File Edit Format Run Options Windows Help
# -*- coding: UTF-8 -*-

#dados normalizados no codigos

import xlrd

arq = xlrd.open_workbook('4fases_dados.xls')

plan = arq.sheets()[0]

dados=[]

#####
#####
#####
def dist4(a,b):
    d = ((a[0]-b[0])**2 + (a[1]-b[1])**2 + (a[2] - b[2])**2 + (a[3] -b[3])**2)**(0.5)
    return d

def calculaEpsilon1():
    for i in range(n):
        e = (max(distancias[i][1]) - min(distancias[i][1]))/sum(distancias[i][1])
        ep.append([e,distancias[i][0]])

def calculaEpsilon2():
    for i in range(n):
        a1 = distancias[i][1][0]
        a2 = distancias[i][1][1]
        a3 = distancias[i][1][2]
        a4 = distancias[i][1][3]
        a5 = distancias[i][1][4]
        a6 = distancias[i][1][5]

        f = ((a1-a2)**2 + (a1-a3)**2 + (a1-a4)**2 + (a1-a5)**2 + (a1-a6)**2 +
              (a2-a3)**2 + (a2-a4)**2 + (a2-a5)**2 + (a2-a6)**2 + (a3-a4)**2 +
              (a3-a5)**2 + (a3-a6)**2 + (a4-a5)**2 + (a4-a6)**2 + (a5-a6)**2)/sum(distancias[i][1])
        ep.append([f,distancias[i][0]])

#####
#####
#####

```

(tetraedro ótimo), sendo o mais próximo possível do tetraedro regular (modelo de referência). Como resultado, a combinação de número 567621 produziu o valor mínimo  $\varepsilon = 1,08297775167$  referente aos intervalos 21, 0, 20 e 21, que representam, respectivamente, os intervalos <sup>11</sup> 22, 1, 21 e 22, em X, Y, Z e W, da lista original de dados. Essas escolhas produzem o tetraedro ótimo.

Com os intervalos indicados após o processo de regularização, obtém-se a matriz<sup>12</sup>

$$M_{ot} = \begin{pmatrix} 0.324732 & 1.0 & 0.440262 & 0.730282 \\ 0.444670 & 0.496266 & 0.602036 & 1.0 \\ 0.000645 & 1.0 & 0.128708 & 0.113490 \\ 0.004614 & 1.0 & 0.0 & 0.0 \\ 1.0 & 1.0 & 1.0 & 1.0 \end{pmatrix}, \quad (4.23)$$

cuja decomposição DVS produz a matriz de valores singulares:

$$S_{ot} = \begin{pmatrix} 2.8580 & 0.0 & 0 & 0 \\ 0 & 1.1633 & 0 & 0 \\ 0 & 0 & 0.3783 & 0 \\ 0 & 0 & 0 & 0.0604 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (4.24)$$

Portanto, ao se efetuar a divisão entre o maior e o menor dos valores singulares de  $S_{ot}$ ,

<sup>11</sup>No programa, a contagem do primeiro intervalo inicia-se por 0.

<sup>12</sup>Os valores foram escritos com seis casas decimais, por truncamento.

o condicionamento é dado por  $\kappa(M_{ot}) = 47,317880$ . Utilizando-se os mesmos intervalos e os dados originais, o condicionamento da matriz correspondente seria  $\kappa = 461,684931$ . Isto revela um comportamento similar ao que foi observado para o problema com três fases, obtendo-se um considerável decréscimo no condicionamento.

## 5

# Discussão dos Resultados

O processo de regularização geométrica proposto neste estudo trouxe à tona algumas novas ideias e questionamentos sobre a resolução do problema inverso da PGNAA em fluidos multifase. Para alguns desses questionamentos, foi possível apresentar respostas ao se propor uma abordagem alternativa ao que tem sido feito, em geral, para resolver o mesmo tipo de problema. Contudo, explorar novos conhecimentos suscita o desenvolvimento de novas teorias e técnicas, demandando a produção de outros conhecimentos que devem ser incorporados aos já existentes, visando à evolução e à consolidação da teoria inicial. Tendo em mente que os resultados aqui obtidos representam o início de um novo ciclo de produção do conhecimento, será feita uma breve discussão acerca desses resultados, tentando evidenciar a sua importância para a criação de outras ferramentas que possam ser aplicadas em problemas da mesma natureza.

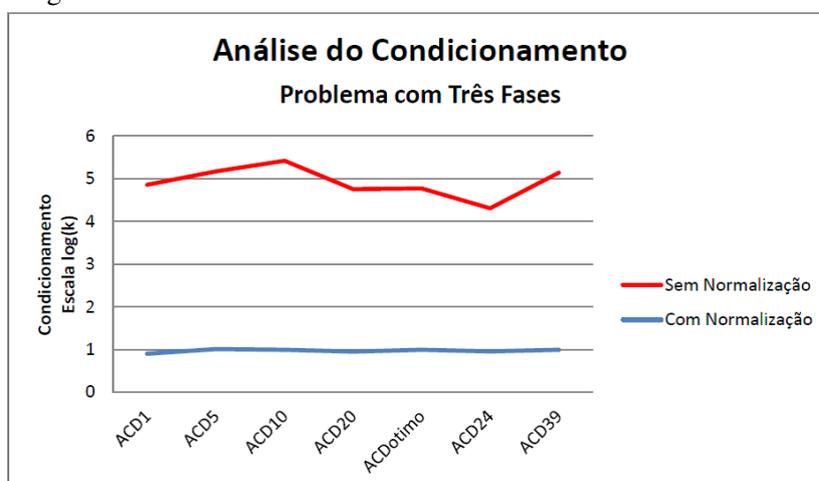
Partindo de um diferente olhar sobre o problema inverso da PGNAA do fluido multifase e do gráfico dos espectros das substâncias desse fluido, trazidos nos estudos de Meric *et al.* (2012), foi apresentada uma nova técnica de representação das principais substâncias presentes no fluido estudado, baseada na construção de vetores (geometrização). Nesse processo de geometrização observou-se que, ao se restringir o problema original, com quatro substâncias, a um problema com apenas três substâncias, foi possível utilizar as representações geométricas dos vetores dessas fases como pontos em  $\mathbb{R}^3$ , possibilitando a representação do problema através de triângulos, e com isso estudar as relações entre o seu comportamento e o condicionamento de cada matriz de coeficientes dos sistemas lineares, originados da representação de pontos como combinações baricêntricas dos vértices desses triângulos. O processo de geometrização do problema possibilitou ainda o desenvolvimento de um caso especial da regularização de Tikhonov: a regularização geométrica.

Verificou-se que o comportamento dos triângulos pode revelar o mau-condicionamento das matrizes relacionadas. Em geral, triângulos muito agudos podem indicar mau-condicionamento. Entretanto, após ser executada a regularização geométrica do problema com três fases, observou-se uma drástica melhora no condicionamento das matrizes para todas as escolhas de intervalos (conforme pode ser visto nas Figuras 5.1, 5.2 e 5.5), sendo possível ainda perceber que nem sempre o triângulo que representa a matriz com melhor condicionamento é aquele mais próximo do modelo de referência (triângulo equilátero). Uma das explicações para esse comportamento deve-se ao fato de que, no método proposto (4.16), o valor do parâmetro de regularização  $\lambda$  controla o quanto se terá um triângulo mais equilátero ou o quanto será considerada a solução de mínimos quadrados relativa ao ponto que representa a substância desconhecida. Portanto, tornar  $\lambda$  cada vez mais próximo de

zero indica que o resultado será provido cada vez mais pela solução de mínimos quadrados, para cada escolha  $k$  de intervalos; caso contrário, a solução dependerá majoritariamente da aproximação ao modelo de referência. Uma outra explicação é que triângulos muito agudos, se estiverem posicionados próximo à origem em  $\mathbb{R}^3$ , podem ter dois vértices que representem vetores (partindo da origem) quase ortogonais. Assim, esse tipo de triângulo, embora muito agudo, pode ter baixo condicionamento.

Na Figura 5.1 alguns intervalos de coordenadas foram escolhidos para ilustrar os comportamentos evidenciados. Por simplicidade, foram utilizados intervalos de mesma ordem em cada partição<sup>1</sup>, com exceção do triângulo ótimo, resultante dos cálculos realizados para determinar o triângulo mais próximo do modelo de referência. Seus vértices foram obtidos a partir dos intervalos 23, 39 e 12, respectivamente para X, Y e Z.

Figura 5.1: Análise do Condicionamento - Problema com três fases.



A imagem acima traz uma ideia das diferenças entre os condicionamentos das matrizes, antes e depois do processo de normalização dos intervalos de coordenadas. No eixo horizontal estão relacionados os triângulos  $ACD$  formados pelos intervalos 1, 5, 10, 20, 24, 39, e o triângulo  $ACD$  ótimo; no eixo vertical estão indicados os valores do condicionamento  $\kappa$  em escala logarítmica. Os gráficos em vermelho e azul ilustram, respectivamente, o comportamento geral dos condicionamentos, com destaque para as escolhas relacionadas. O quadro a seguir oferece mais detalhes sobre esses valores:

Figura 5.2: Condicionamento antes e depois da normalização - Problema com três fases.

Triângulo	Condicionamento		%
	Sem Normalização	Com Normalização	
$ACD1$	9086.666666666666	7.998821796	0.088
$ACD5$	14532	10.24567855	0.070
$ACD10$	26552	9.90003608	0.037
$ACD20$	6342.666666666666	8.96845540	0.141
$ACD$ ótimo	6028.333333333333	9.81451322	0.162
$ACD24$	2260.5	9.01904064	0.398
$ACD39$	14286	8.92597968	0.062

O quadro da Figura 5.2 lista os intervalos de coordenadas escolhidos e os condicionamentos

<sup>1</sup>Por exemplo, o triângulo  $ACD5$  é formado pelo quinto intervalo de coordenadas em cada partição X, Y e Z.

das matrizes representadas pelos triângulos  $ACD$ . A segunda coluna relaciona os condicionamentos originais e a terceira destaca os valores encontrados após a normalização. A última coluna indica a que percentual do respectivo valor original o condicionamento foi reduzido.

Com base nesse conjunto de informações, é razoável considerar as seguintes observações:

1. O decréscimo do condicionamento não acontece na mesma proporção entre cada uma das escolhas de intervalos. Por exemplo, em  $ACD1$  o resultado representa apenas 0,088% do valor original, enquanto que em  $ACD24$  o resultado representa 0,398% do primeiro valor. Para o triângulo ótimo, o condicionamento sem normalização é  $\kappa = 6028,3333$ , enquanto que o condicionamento resultante do processo de regularização representa 0,162% desse valor.
2. Conforme mencionado, determinar o triângulo que mais se aproxima do modelo de referência não significa necessariamente encontrar a escolha de intervalos de coordenadas que produz o melhor condicionamento após normalização. No exemplo apresentado, esse valor foi  $\kappa = 9,81451322$ , que embora esteja bem próximo dos demais, não é o menor.
3. O processo de normalização dos dados em cada intervalo de coordenadas, por si já pode ser considerado como um tipo de regularização, pois consegue tratar o mau-condicionamento. Nos códigos presentes dos Apêndices B e C, os dados normalizados são utilizados para a busca do triângulo ou o tetraedro que mais se aproximam dos modelos de referência.
4. Todos os condicionamentos apresentados foram reduzidos, após regularização, a um percentual que representa menos de 0.5% de seu valor original, evidenciando a eficiência do método.

Para finalizar a discussão dos resultados para o problema com três fases, é necessário destacar um dos resultados mais significativos deste estudo: a previsão dos limites do *background* em função de substância desconhecida  $P$  e dos vértices de triângulo  $ACD$ , calculados através das Equações (4.19), (4.20), e (4.21). Devido à simplicidade dessa formulação, o cálculo dos limites, e consequentemente a indicação de uma região que contém o *background*, torna-se uma tarefa relativamente simples, que depende das coordenadas dos vértices do triângulo e da escolha <sup>2</sup> do parâmetro  $\delta$ , que estabelece uma certa tolerância para a localização do *background* (que pode ser influenciada pelo ruído). O desenvolvimento de diferentes formas para a determinação do parâmetro  $\delta$  e do *background* constitui-se como temática para trabalhos futuros.

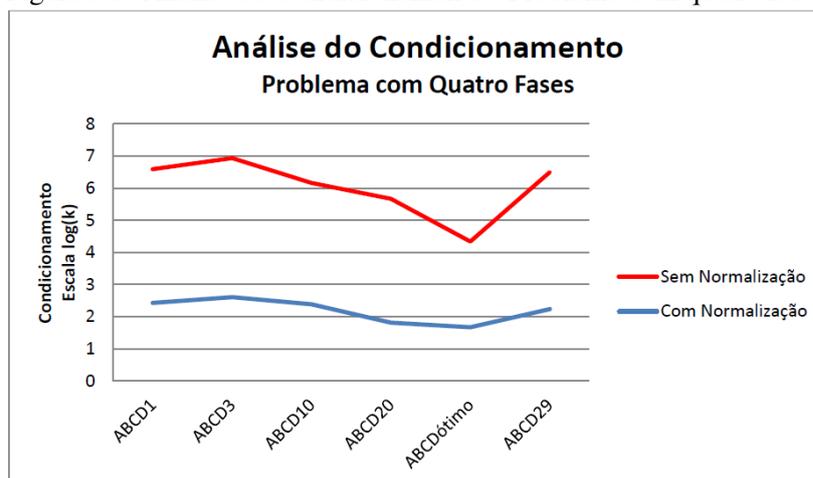
Ao analisar os resultados obtidos no caso do problema com quatro fases, percebe-se inicialmente que o processo de regularização geométrica, aplicado ao caso anterior, sofreu pequenas modificações para ser aplicado ao fluido original. O acréscimo da substância  $B$  (sal) modificou a estrutura dos vetores de substâncias, dotando-os de quatro coordenadas. Como consequência, a normalização foi fundamentada na mudança de escala relativamente a quatro partições  $X$ ,  $Y$ ,  $Z$  e  $W$ , e não mais a três, além de considerar tetraedros, ao invés de triângulos.

A normalização realizada em cada intervalo de coordenadas produziu importantes resultados, reduzindo significativamente, a exemplo do caso anterior, os condicionamentos das matrizes de coeficientes formadas pelos vértices dos tetraedros, conforme o sistema (4.7).

<sup>2</sup>Pode-se estimar o parâmetro  $\delta$  consoante se deseje determinar uma maior ou menor região que contenha o *background*. Neste estudo, os cálculos foram realizados utilizando-se uma taxa de 10% ( $\delta = 0.1$ ) de variação para a localização dos limites ( $Bg$ ), sobre as retas  $AP$ ,  $CP$  e  $DP$  (Figura 4.14).

Na figura a seguir, os condicionamentos estão dispostos no eixo vertical, em escala logarítmica. No eixo horizontal estão representados os tetraedros escolhidos e ainda, nas cores vermelha e azul, encontram-se os gráficos que ilustram o comportamento geral do condicionamento, antes e após a normalização, respectivamente.

Figura 5.3: Análise do Condicionamento - Problema com quatro fases.



Em geral, percebe-se que ocorre um decréscimo significativo nos condicionamentos, o que indica a ação da regularização geométrica com eficiência no problema com quatro fases. A figura a seguir apresenta mais detalhadamente os valores encontrados nesse processo:

Figura 5.4: Condicionamento antes e depois da regularização - Problema com quatro fases.

Tetraedro	Condicionamento		%
	Sem Normalização	Com Normalização	
<i>ABCD1</i>	14682	268.238532	1.826
<i>ABCD3</i>	21359.5	406.257142	1.901
<i>ABCD10</i>	5967.071428	242.955752	4.071
<i>ABCD20</i>	7177	65.389162	0.911
<i>ABCD ótimo</i>	461.684931	47.317880	10.249
<i>ABCD29</i>	18406	171.569536	0.932

Embora o condicionamento tenha decrescido drasticamente no problema com quatro fases, os valores encontrados situam-se numa faixa mais ampla, isto é, estão mais distantes entre si do que os encontrados no caso anterior. Além disso, esse decréscimo não ocorre de modo uniforme através de alguma razão, o que não interfere na eficiência do método, posto que foram encontrados valores que representam pequenos percentuais das quantidades originais, embora ainda um pouco maiores comparativamente ao problema com três fases.

O tetraedro ótimo (*ABCDótimo*) foi concebido a partir dos intervalos de coordenadas 21, 0, 20, 21, da lista de dados normalizados, e possui o menor condicionamento entre os exemplos das figuras (5) e (5), enquanto que a maior redução se deu no tetraedro *ABCD20*, cujo valor encontrado representa apenas 0.911% do seu condicionamento sem normalização.

A regularização geométrica (4.16) para o problema com três fases foi efetuada escolhendo-se os seguintes valores de  $\lambda$ : 0.01, 0.03, 0.05, 0.1, 0.5 e 0.8. O código *geradorSusbtancia4.py* (Apêndice D) determinou a substância desconhecida e os intervalos de coordenadas que representam

os triângulos para o problema com três fases regularizado. O quadro presente na figura abaixo resume os resultados obtidos:

Figura 5.5: Condicionamento com Regularização Geométrica - Problema com três fases.

Intervalos	$\lambda$	Condicionamento		%
		Sem Regularização	Com Regularização	
10, 4, 9	0.01 e 0.03	39056	1.5276	0.039
24, 4, 1	0.05, 0.1, 0.5	9606	10.330978	0.107
23, 39, 12	0.8	6028.333333	9.814513	0.162

Para os parâmetros 0.01 e 0.03, o programa indicou os intervalos 10, 4 e 9, respectivamente em X, Y e Z. Realizando-se a comparação entre os valores do condicionamento antes e depois da regularização geométrica para essa escolha de intervalos, observa-se que o valor obtido representa apenas 0,039% do valor original. Os demais casos também produziram um decréscimo significativo do condicionamento, reduzindo-o sempre a menos de 0,5% do valor sem regularização, para todas as escolhas.

## 6

# Conclusão e Trabalhos Futuros

O desenvolvimento de métodos para encontrar soluções para o problema inverso da determinação de frações em peso de substâncias em fluidos multifásicos, tem sido objeto de diversos estudos, pois a eficiência de tais métodos tem impacto direto no desenvolvimento de novas tecnologias. Neste estudo, a abordagem realizada possibilitou a criação de um processo matemático que visa contribuir com o método MCLLS aplicado à análise PGNAA, em um fluido multifásico obtido da extração de petróleo submarino. Espera-se, deste modo, que as contribuições aqui apresentadas sirvam de subsídios para o surgimento de métodos ainda mais eficientes e precisos para a resolução deste e de outros problemas de mesma natureza.

Com base nos objetivos enunciados no início deste estudo, as principais contribuições desta tese são apresentadas, a seguir:

1. Desenvolvimento de um processo de geometrização para o problema da PGNAA em fluidos com três e quatro fases.
2. A partir do processo de geometrização proposto, foi elaborado um novo método de regularização: a Regularização Geométrica. Este método constitui-se em um caso especial da regularização de Tikhonov, e consiste primordialmente no cálculo de soluções para o problema da determinação das frações em peso das substâncias presentes em um fluido multifásico obtido na nos processos de exploração de petróleo submarino, considerando-se dois casos: um fluido com três ou quatro fases.
3. Para o fluido com três fases, a representação geométrica do problema consiste na determinação de triângulos, e para o caso com quatro fases, tetraedros. Neste último, os vetores atribuídos às respectivas substâncias pertencem a  $\mathbb{R}^4$ , e portanto a representação geométrica reduz-se à abordagem analítica, posto que não é possível produzir imagens neste espaço. Nos triângulos e tetraedros obtidos, os vértices representam as substâncias exibidas nos gráficos dos espectros, relativamente a uma escolha de intervalos de coordenadas, e o grau de agudeza dos triângulos serve como um indicativo de mau-condicionamento das matrizes de coeficientes dos sistemas lineares a eles relacionados.
4. A regularização geométrica, diferentemente de outros métodos já conhecidos de resolução do problema em estudo, caracteriza-se por percorrer todos os intervalos de coordenadas, em cada partição, combinando-os até estabelecer quais deles produzem o triângulo ou o tetraedro que mais se aproxima do modelo de referência (o triângulo equilátero ou o tetraedro regular, dependendo do

caso), através de um processo de normalização dos dados, realizado a partir da integral ao longo desses intervalos, otimizando o condicionamento.

5. Determinou-se a formulação (4.16) para a regularização geométrica, que é equivalente às expressões (2.7) e (2.45).

6. A estimativa de uma região para a localização do *background*, em função da substância desconhecida e da escolha de intervalos de coordenadas, para o fluido com três fases, tornou-se possível graças à determinação das equações (4.19), (4.20) e (4.21).

7. Foram elaborados alguns códigos computacionais para a execução de diversas tarefas: Normalização de dados, busca pelo intervalo de coordenadas mais próximo do modelo de referência, e, especificamente para o fluido com três fases, determinar a solução regularizada, a solução de mínimos quadrados e calcular os limites do *background*, para cada escolha  $k$  de intervalos.

8. Após a criação de uma nova substância, baseada nas já existentes, alguns testes foram realizados no sentido de comparar os condicionamentos das matrizes de coeficientes dos sistemas originados de escolhas de intervalos, antes e depois da aplicação da regularização geométrica, confirmando a eficiência do método.

O desenvolvimento desta nova abordagem, por sua característica de ineditismo, impulsiona o surgimento de uma variedade de questionamentos e revela-se como ponto de partida na criação de novas teorias e técnicas baseadas nessa formulação geométrica. É razoável então acreditar que a realização de outras experiências com o modelo proposto neste estudo, inclusive em outros problemas, possa contribuir para explorar ainda mais o seu potencial. Nesse sentido, são apresentadas a seguir algumas propostas de trabalhos futuros:

1. Executar a regularização geométrica fazendo com que a busca pelos melhores intervalos de coordenadas considere também variações por sobreposição de canais. Isto é, ao invés de considerar cada intervalo com 10 canais diferentes, esses intervalos poderiam ter interseções entre os canais, como por exemplo, o primeiro intervalo na partição  $X$  poderia ser constituído pelos canais (1 - 10) e o segundo intervalo, na mesma partição, poderia ser constituído pelos canais (5 - 15).

2. Encontrar, através da geometria, formas distintas de determinar o *background*, inclusive para o problema com quatro fases.

3. Realizar experimentos com dados reais.

4. A partir da introdução do método proposto, determinar o quanto será reduzido em etapas de mínimos quadrados em MCLS.

5. Fornecer subsídios sobre o *background* para a fase Monte Carlo, ou seja, a partir do espectro gama do *background*, determinar se existe uma substância a mais no fluido ou se o *background* é produzido por ruído externo.

6. Generalizar a regularização geométrica para um número arbitrário de fases.

7. Adaptar a teoria apresentada em [13], que trata dos efeitos da regularização de Tikhonov na topografia de problemas lineares inversos, para a regularização geométrica, ou seja, estudar os efeitos da regularização geométrica na topografia dos problemas lineares inversos.

# Referências

- [1] ASTER, Richard C.; BORCHERS, Brian; THURBER, Clifford H. **Parameter Estimation and Inverse Problems**, 2nd ed. New York: Academic Press, 2012.
- [2] BORGES, Leonardo S. **Métodos Para Problemas Discretos Mal-Postos de Grande Porte**. Campinas: UNICAMP, 2013. 204f. Tese (Doutorado em Matemática Aplicada). Programa de Pós-Graduação em Matemática Aplicada, Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, Campinas, 2013.
- [3] BRENNEN, Christopher E. **Fundamentals of Multiphase Flows**. California: Cambridge University Press, 2005.
- [4] BRUUN, Christian E., NIELSEN, Trine B. **Algorithms and Software for Large-Scale Geophysical Reconstructions**. Denmark: Technical University of Denmark, 2008. 114f. Dissertação (Master in Informatics and Mathematical Modelling), Technical University of Denmark. Denmark, 2008.
- [5] BURGER, Martin. **Inverse Problems**. Lecture Notes. Münster: WestfälischeWilhelms-Universität Münster, 2007. 78f. Notas de aula. Institute für Numerische und Angewandte Mathematik, WestfälischeWilhelms-Universität Münster. Münster, 2007.
- [6] CARDOSO, Halisson A. C. **Minimizando o mau Condicionamento na Análise por Radiação Gama**. Recife: UFPE, 2015. 99f. Dissertação (Mestrado em Ciência da Computação). Programa de Pós-Graduação em Ciência da Computação, Centro de Informática, Universidade Federal de Pernambuco, Recife, 2015.
- [7] CEZARO, Adriano de. **Métodos de Regularização Tipo Level Set para Problemas Inversos**. Florianópolis: UFSC, 2006. 52f. Dissertação (Mestrado em Matemática). Programa de Pós-Graduação em Matemática e Computação Científica, Centro de Ciências Físicas e Matemáticas, Universidade Federal de Santa Catarina, Florianópolis, 2006.
- [8] CLARK, T. Chris; GARDNER, Robin C.; VERGHESE, Kuruvilla. A monte Carlo model for in situ prompt gamma-ray analysis probes. **Nucl. Instrum. Methods**, 193, p. 365-370, 1982.
- [9] CLINE, Alan K.; MOLER, Cleve B.; STEWART, G. W.; WILKINSON, James H. An estimate for the condition number of a matrix. **SIAM J. Numer. Anal.**, 16, p. 368-375, 1979.
- [10] ENGL, Heinz W.; HANKE, Martin; NEUBAUER, Andreas. **Regularization of Inverse Problems**. USA: Kluwer Academic Press, 1996.

- [11] FERNÁNDEZ-MARTÍNEZ, Juan L. *et al.* On the topography of the coast functional in linear and nonlinear inverse problems. **Geophysics**, v. 1, n. 77, p. w1-w15, 2012.
- [12] FERNÁNDEZ-MARTÍNEZ, Juan L. *et al.* From Bayes to Tarantola: New insights to understand uncertainty in inverse problems. **Journal of Applied Geophysics**, 98, p. 62-72, 2013.
- [13] FERNÁNDEZ-MARTÍNEZ, Juan L. *et al.* The effect of noise and Tikhonov's regularization in inverse problems. Part I: The linear case. **Journal of Applied Geophysics**, 108, p. 176-185, 2014a.
- [14] FERNÁNDEZ-MARTÍNEZ, Juan L. *et al.* The effect of noise and Tikhonov's regularization in inverse problems. Part II: The nonlinear case. **Journal of Applied Geophysics**, 108, p. 186-193, 2014b.
- [15] FRANCO, Neide B. **Cálculo Numérico**. São Paulo: Pearson Prentice Hall, 2006.
- [16] FUHRY, Martin. **A New Tikhonov Regularization Method**. Kent: Kent State University, 2011. 31f. Tese (Requisito para recebimento de Honra Universitária), Department of Mathematics, Kent State University, 2011.
- [17] GOLUB, Gene H.; VAN LOAN, Charles F. **Matrix Computations**. 3ª ed. USA: The Johns Hopkins University Press, 1996.
- [18] GROETSCH, Charles W. **The theory of Tikhonov regularization for Fredholm equations of the first kind**. London: Pitman Publishing Limited, 1984.
- [19] HADAMARD, Jacques S. Sur les problèmes aux dérivées partielles et leur signification physique. **Princeton University Bulletin**, v. 13, n. 4, p. 49-52, 1902.
- [20] HAN, Xiaogang; GARDNER, Robin P.; METWALLY, Walid A. CEARCPG: a Monte Carlo simulation code for normal and coincidence prompt gamma-ray neutron activation analysis (PGNAA). **Nucl. Sci. Eng.**, 155, p. 143-153, 2007.
- [21] HANSEN, Per Christian. **Regularization Tools**. A Matlab Package for Analysis of Discrete Ill-Posed Problems. Cambridge University Press. 2008, 126f. Manual da versão 4.1 para Matlab 7.3. Informatics and Mathematical Modelling, Technical University of Denmark. Denmark, 2008.
- [22] KJOLLER, Steffen Brun. **Parameter Estimation for Large-Scale Reconstruction**. Denmark: Technical University of Denmark, 2007. 86f. Tese (PhD in Informatics and Mathematical Modelling), Technical University of Denmark. Denmark, 2007.
- [23] LAY, David C. **Linear Algebra and Its Applications**. 4ª ed. Boston: Addison-Wesley, 2012.
- [24] MENKE, William. **Geophysical data analysis: discrete inverse theory**. No. 40, International Geophysical Series. New York: Academic Press, 1989.
- [25] MERIC, Ilker *et al.* On the treatment of ill-conditioning cases in the Monte Carlo Library Least Squares (MCLLS) approach for inverse radiation analysers. **Meas. Sci. Technol**, 23, 055603, 2012.

- [26] MERIC, Ilker *et al.* A Library Least-Squares Approach for Scatter Correction in Gamma-Ray Tomography. **Radiation Physics and Chemistry**, 108, p. 39-47, 2015.
- [27] MALAJOVICH, Gregorio. **Álgebra Linear**. Rio de Janeiro: DMA-UFRJ, 2010.
- [28] MONTEIRO, Roberto P. G. **Análise Linear em Problemas Inversos - Aplicação ao espectro de aniquilação de pósitrons e ao estudo de sistemas líquidos**. UFMG, Minas Gerais, 2005. 106f. Tese (Doutorado em Ciências - Química). Instituto de Ciências Exatas, Universidade Federal de Minas Gerais. Minas Gerais, 2005.
- [29] O'LEARY, Dianne P. Estimating Matrix Condition Numbers. **SIAM J. Stat. Comput.**, v. 1, n. 2, p. 205-209, 1980.
- [30] SHYU, C M.; GARDNER, Robin P.; VERGHESE, Kuruvilla. Development of the Monte Carlo-library least-squares method of analysis for neutron capture prompt gamma-ray analyzers. **Nucl. Geophys**, 7, p. 241-267, 1993.
- [31] SPEIGHT, James G. **The chemistry and technology of petroleum**. New York: Marcel Dekker, 1999.
- [32] TITARENKO, Valeriy *et al.* Regularization methods for inverse problems in X-ray tomography. **Proceedings of SPIE - Developments in X-ray Tomography VII**, v. 7804, p. 78040z-1 - 78040z-10, 2010.
- [33] TIKHONOV, Andrey N. Solution of incorrectly formulated problems and the regularization method, **Dokl. Akad. Nauk. SSSR**, v. 151, pp. 501-504, 1963.
- [34] UNGAR, Abraham A. **Barycentric Calculus in Euclidean and Hyperbolic Geometry: A comparative Introduction**. Singapore: World Scientific Publishing Co., 2010.
- [35] WANG, Jiaxin; LI, Fusheng; GARDNER, Robin P. On the use of prompt gamma-ray neutron activation analysis for determining phase amounts im multiphase flow. **Meas. Sci. Technol**, 19, 094005, 2008.
- [36] ZHANG, Wenchao; GARDNER, Robin P. The analog linear interpolation approach for Monte Carlo simulation of PGNA: the CEARPGA code, **Nucl. Instum. Methods B**, 213, p. 116-123, 2003.

## Apêndice A

# Lista de dados de contagem referentes aos espectros do fluido multifásico

Lista de dados de contagens dos níveis de energia referentes à partição  $X$ . Os dados referentes às partições  $Y$ ,  $Z$  e  $W$  foram omitidos.

Table generated by Excel2LaTeX from sheet 'Planilha1'

Tabela A.1: Lista de dados de contagem referentes aos espectros do fluido multifásico

Agua	Sal	Óleo	Gás	Background
139000000	267000000	189000000	313000000	100000000
115000000	221000000	156000000	259000000	82900000
99200000	191000000	135000000	224000000	71700000
87700000	168000000	119000000	198000000	63300000
79200000	152000000	108000000	179000000	57000000
72700000	139000000	98900000	164000000	52400000
66700000	128000000	90700000	150000000	48300000
62800000	120000000	85400000	142000000	45000000
64200000	124000000	87200000	145000000	46200000
73300000	141000000	99600000	165000000	52700000
86300000	167000000	117000000	195000000	62400000
100000000	193000000	136000000	226000000	68900000
112000000	216000000	152000000	252000000	83300000
121000000	234000000	164000000	273000000	97700000
129000000	248000000	175000000	290000000	106000000
132000000	254000000	179000000	297000000	97300000
113000000	218000000	154000000	256000000	73800000
83700000	161000000	114000000	189000000	53000000
116000000	224000000	159000000	263000000	73200000

Tabela A.2: Continuação - Planilha de dados

---

138000000	266000000	189000000	312000000	887000000
139000000	264000000	190000000	315000000	959000000
137000000	259000000	186000000	309000000	1060000000
134000000	254000000	182000000	301000000	1080000000
130000000	249000000	178000000	295000000	1080000000
128000000	243000000	174000000	288000000	1030000000
126000000	236000000	170000000	283000000	926000000
124000000	228000000	168000000	278000000	854000000
120000000	218000000	163000000	270000000	818000000
115000000	208000000	156000000	259000000	759000000
109000000	197000000	149000000	246000000	681000000
103000000	188000000	141000000	234000000	622000000
97500000	179000000	133000000	220000000	588000000
91900000	171000000	126000000	208000000	556000000
86800000	164000000	119000000	197000000	516000000
82200000	157000000	113000000	186000000	480000000
77600000	150000000	106000000	175000000	452000000
73100000	142000000	99500000	165000000	429000000
69200000	135000000	94100000	156000000	405000000
66100000	129000000	89800000	149000000	383000000
63400000	124000000	86200000	143000000	373000000
61000000	120000000	82900000	138000000	375000000
58900000	115000000	80000000	133000000	383000000
57300000	111000000	77800000	129000000	390000000
56300000	107000000	76300000	127000000	388000000
55700000	104000000	75500000	125000000	373000000
55100000	100000000	74700000	124000000	350000000
54200000	97200000	73500000	122000000	324000000
52900000	94200000	71700000	119000000	301000000
51200000	91300000	69500000	115000000	284000000
49500000	88400000	67300000	112000000	271000000
47700000	85500000	65000000	108000000	263000000
45900000	82800000	62700000	104000000	256000000
44200000	80300000	60400000	100000000	252000000
42600000	78000000	58200000	96400000	248000000
41100000	75900000	56000000	92900000	243000000
39700000	73800000	54000000	89600000	237000000
38300000	71700000	52100000	86400000	231000000
37000000	69700000	50300000	83500000	226000000
35800000	67800000	48600000	80700000	224000000
34700000	65900000	47100000	78100000	223000000
33600000	64200000	45600000	75700000	225000000
32500000	62600000	44300000	73400000	227000000
31500000	61100000	43100000	71300000	229000000
30600000	59500000	41900000	69200000	230000000

---

Tabela A.3: Continuação - Planilha de dados

29700000	57900000	40700000	67200000	22900000
28800000	56200000	39600000	65200000	22700000
28000000	54600000	38400000	63300000	22200000
27200000	53100000	37200000	61400000	21700000
26400000	51700000	36100000	59600000	21100000
25700000	50400000	35000000	57900000	20600000
24900000	49200000	33900000	56100000	20200000
24100000	47900000	32900000	54500000	20000000
23400000	46600000	31900000	52800000	19700000
22700000	45400000	31000000	51300000	19500000
22100000	44400000	30100000	50000000	19200000
21600000	43600000	29500000	48800000	18900000
21200000	42900000	28900000	47800000	18600000
20800000	42400000	28300000	46900000	18400000
20400000	42000000	27800000	46100000	18200000
20000000	41800000	27300000	45200000	18000000
19700000	41500000	26800000	44400000	17900000
19300000	41300000	26200000	43500000	17900000
18900000	41000000	25700000	42700000	17900000
18600000	40800000	25300000	41900000	18000000
18300000	40400000	24800000	41200000	18000000
18000000	40000000	24400000	40500000	18100000
17700000	39600000	24000000	39800000	18200000
17400000	39200000	23600000	39200000	18300000
17200000	39000000	23300000	38800000	18400000
17100000	39200000	23200000	38500000	18500000
17100000	40100000	23200000	38600000	18600000
17300000	42100000	23600000	39100000	18900000
17800000	45500000	24200000	40000000	19300000
18400000	50300000	25000000	41400000	19700000
19000000	56100000	25900000	42800000	20100000
19500000	62200000	26600000	44100000	20300000
19800000	67500000	27000000	44700000	20300000
19700000	70900000	26900000	44500000	20000000
19200000	71600000	26200000	43400000	19300000
18300000	69000000	25000000	41400000	18500000
17200000	63600000	23500000	38900000	17500000
16000000	56200000	21800000	36200000	16500000
15000000	48200000	20400000	33800000	15700000
14100000	40700000	19200000	31900000	15100000
13500000	34500000	18400000	30400000	14600000
13000000	30000000	17700000	29400000	14300000
12700000	26900000	17300000	28700000	14100000
12500000	24900000	17000000	28200000	14000000
12300000	23800000	16800000	27800000	13900000

Tabela A.4: Continuação - Planilha de dados

12200000	23000000	16500000	27400000	13800000
12000000	22500000	16300000	27100000	13700000
11900000	22100000	16100000	26800000	13600000
11700000	21800000	15900000	26400000	13400000
11600000	21500000	15800000	26100000	13300000
11500000	21300000	15600000	25900000	13100000
11400000	21100000	15400000	25600000	12900000
11200000	20900000	15200000	25300000	12800000
11100000	20700000	15100000	25100000	12600000
11000000	20500000	14900000	24800000	12300000
10900000	20300000	14800000	24600000	12100000
10800000	20100000	14600000	24300000	11900000
10700000	19900000	14500000	24100000	11700000
10600000	19700000	14400000	23800000	11500000
10500000	19400000	14200000	23600000	11300000
10400000	19200000	14100000	23400000	11200000
10300000	18900000	13900000	23100000	11100000
10200000	18700000	13800000	22900000	11100000
10100000	18500000	13700000	22700000	11000000
10000000	18300000	13600000	22500000	10900000
9910000	18100000	13500000	22400000	10900000
9830000	18000000	13400000	22200000	10800000
9750000	17800000	13300000	22000000	10700000
9670000	17600000	13200000	21800000	10600000
9590000	17500000	13100000	21700000	10500000
9510000	17400000	13000000	21500000	10400000
9440000	17300000	12900000	21300000	10300000
9370000	17300000	12800000	21200000	10300000
9300000	17400000	12700000	21000000	10200000
9230000	17700000	12600000	20800000	10100000
9160000	18300000	12500000	20700000	10000000
9090000	19200000	12400000	20500000	9960000
9020000	20600000	12300000	20400000	9900000
8960000	22400000	12200000	20200000	9860000
8890000	24600000	12100000	20000000	9850000
8820000	26900000	12000000	19900000	9860000
8760000	29200000	11900000	19700000	9900000
8690000	31000000	11800000	19600000	9970000
8630000	32200000	11700000	19400000	10000000
8570000	32400000	11600000	19300000	10100000
8510000	31600000	11500000	19200000	10200000
8450000	29900000	11500000	19000000	10200000
8400000	27500000	11400000	18900000	10200000
8350000	24800000	11300000	18800000	10100000
8300000	22200000	11300000	18700000	10100000

Tabela A.5: Continuação - Planilha de dados

---

8250000	19700000	11200000	18600000	9950000
8200000	17700000	11100000	18500000	9820000
8160000	16200000	11100000	18400000	9690000
8120000	15100000	11000000	18300000	9550000
8080000	14400000	10900000	18200000	9420000
8040000	13900000	10900000	18100000	9280000
8000000	13700000	10800000	18000000	9140000
7960000	13500000	10800000	17900000	9000000
7910000	13500000	10700000	17800000	8860000
7870000	13400000	10700000	17700000	8710000
7820000	13400000	10600000	17600000	8570000
7170000	11300000	9780000	16200000	
7770000	13300000	10500000	17500000	8430000
7720000	13300000	10500000	17400000	8290000
7670000	13200000	10400000	17300000	8150000
7620000	13100000	10400000	17200000	8000000
7570000	13000000	10300000	17100000	7860000
7530000	12900000	10200000	17000000	7710000
7480000	12800000	10200000	16900000	7570000
7440000	12700000	10100000	16800000	7440000
7400000	12500000	10100000	16700000	7320000
7360000	12300000	10000000	16600000	7220000
7320000	12100000	9980000	16500000	7130000
7280000	11900000	9930000	16500000	7050000
7240000	11700000	9880000	16400000	7000000
7210000	11500000	9830000	16300000	6960000
7130000	11100000	9730000	16100000	
7100000	11000000	9670000	16000000	
7060000	10800000	9620000	16000000	
7030000	10700000	9580000	15900000	
7000000	10600000	9530000	15800000	
6970000	10500000	9490000	15700000	
6950000	10400000	9450000	15700000	
6920000	10300000	9410000	15600000	
6900000	10200000	9380000	15600000	
6870000	10100000	9350000	15500000	
6850000	10100000	9310000	15400000	
6820000	10000000	9280000	15400000	
6800000	9930000	9260000	15300000	
6780000	9860000	9230000	15300000	
6760000	9780000	9200000	15300000	
6740000	9700000	9180000	15200000	
6720000	9620000	9150000	15200000	
6710000	9550000	9130000	15100000	
6690000	9490000	9100000	15100000	

---

## Apêndice B

# Análise de Substâncias - Problema com três fases

Este código determina os valores normalizados para os dados em cada intervalo de coordenadas, e calcula os limites do *background* para cada escolha de intervalos.

```
# -*- coding: UTF-8 -*-

# analise_substancia_3.py

#dados normalizados no codigo
import xlrd
%arq = xlrd.open_workbook('4fasesdados.xls')
plan = arq.sheets()[0]
#=====
#=====
#=====

#Função da distancia no R3

def distr3(a,b):

d = ((a[0]-b[0])**2 + (a[1]-b[1])**2 + (a[2]-b[2])**2)**(0.5)

return d
def calculaEpsilon1():
#FORMULA DO EPSILON

for i in range(n):
e = (max(distancias[i][1]) - min(distancias[i][1]))/sum(distancias[i][1])
ep.append([e,distancias[i][0]])

def calculaEpsilon2():
```

```
#FORMULA DO EPSILON

for i in range(n):
a1 = distancias[i][1][0]#VALOR DOS LADOS
a2 = distancias[i][1][1]
a3 = distancias[i][1][2]

f = ((a1-a2)**2 + (a1-a3)**2 + (a2-a3)**2)/sum(distancias[i][1])
ep.append([f,distancias[i][0]])

#Calcula o produto por escalar
def produtoPescalar(e, vetor):
res = []
res.append(vetor[0] * e)
res.append(vetor[1] * e)
res.append(vetor[2] * e)

return res

def somaVetor(v1, v2):
res = [v1[0]+v2[0], v1[1]+v2[1] ,v1[2]+v2[2]]

return res

#=====
#=====
#=====

dados=[]

for i in xrange(plan.nrows):
dados.append(plan.row_values(i))

dados = dados[1:]

soma=[]
normal = []

i=0
while i < 1200:
aux = dados[i:i+10][:]
somax = somay = somaz = 0
for j in range(10):
somax += dados[i+j][0]
somay += dados[i+j][2]
somaz += dados[i+j][3]
#somaw += dados[i+j][3]
```

```
m = max(somax, somay, somaz)

soma.append([somax/m, somay/m, somaz/m])
i+=10

X=soma[0:40][:]

Y=soma[40:80][:]

Z=soma[80:120][:]

"""
maximos = [[], [], []]
for i in range(40):
maximos[0].append(max(X[i]))
maximos[1].append(max(Y[i]))
maximos[2].append(max(Z[i]))

m1 = max(maximos[0])
m2 = max(maximos[1])
m3 = max(maximos[2])

print Valores da nomalização ,m1,m2,m3,

for i in range(40):
X[i] = [X[i][0]/m1,X[i][1]/m1,X[i][2]/m1]
Y[i] = [Y[i][0]/m2,Y[i][1]/m2,Y[i][2]/m2]
Z[i] = [Z[i][0]/m3,Z[i][1]/m3,Z[i][2]/m3]

"""
print >>>> ,len(X),len(Y),len(Z)#,len(W)

distancias = []

for i in range(40):
for j in range(40):
for k in range(40):
aux = []
aux.append(distr3(X[i],Y[j]))
aux.append(distr3(X[i],Z[k]))
aux.append(distr3(Y[j],Z[k]))

distancias.append([i,j,k],aux])

n = len(distancias)

ep=[]
```

```
#Calculo dos Epsilons
calculaEpsilon1()

i= 1
j= 0
while i < n:
if ep[i][0] < ep[j][0]:
j = i
i+=1

m = j #indice do valor minimo

print -----COORDENADAS - X-----
for i in range(40):
print i, = ,X[i],

print -----COORDENADAS - Y-----
for i in range(40):
print i,' = ',Y[i],

print -----COORDENADAS - Z-----
for i in range(40):
print i,' = ',Z[i],

print Valor Minimo - , m,' - ',ep[m][0], Coordenadas dos pontos,
ep[m][1][0],',',ep[m][1][1],',',ep[m][1][2]

print X[ep[m][1][0]],Y[ep[m][1][1]], Z[ep[m][1][2]],

while True:
bg = raw_input("Estimar o background s/n: ")

while bg not in ['s','n']:
bg = raw_input("Estimar o background s/n: ")

if bg == 's':
d = input("Digite o valor de Delta: ")
p=[]
q = input("Digite o valor de p0: ")
p.append(q)
q = input("Digite o valor de p1: ")
p.append(q)
q = input("Digite o valor de p2: ")
p.append(q)

#coordenadas = raw_input("Digite o conjunto de canais separados por espaço: ")
```

---

```
c = [] #Vetor de coordenadas
a = input("Digite o canal X: ")
c.append(a)
a = input("Digite o canal Y: ")
c.append(a)
a = input("Digite o canal Z: ")
c.append(a)

A = [X[c[0]][0], Y[c[1]][0], Z[c[2]][0]]
B = [X[c[0]][1], Y[c[1]][1], Z[c[2]][1]]
C = [X[c[0]][2], Y[c[1]][2], Z[c[2]][2]]

bga = somaVetor(produtoPescalar(1/d, p), produtoPescalar(1-(1/d), A))
bgb = somaVetor(produtoPescalar(1/d, p), produtoPescalar(1-(1/d), B))
bgc = somaVetor(produtoPescalar(1/d, p), produtoPescalar(1-(1/d), C))

print Limites do BackGround: BGa,bga,BGb,bgb,BGc,bgc,

else:
break
```

## Apêndice C

# Análise de substâncias - Problema com quatro fases

```
# -*- coding: UTF-8 -*-

# analise_substancias4.py

#dados normalizados no codigos

import xlrd

arq = xlrd.open_workbook('4fases_dados.xls')

plan = arq.sheets()[0]

dados=[]

#=====
#=====
#=====

def distr4(a,b):
d = ((a[0]-b[0])**2 + (a[1]-b[1])**2 + (a[2]-b[2])**2 + (a[3]-b[3])**2)
**(0.5)
return d

def calculaEpsilon1():
for i in range(n):
e = (max(distancias[i][1]) - min(distancias[i][1]))/sum(distancias[i][1])
ep.append([e,distancias[i][0]])

def calculaEpsilon2():
for i in range(n):
a1 = distancias[i][1][0]
```

---

```

a2 = distancias[i][1][1]
a3 = distancias[i][1][2]
a4 = distancias[i][1][3]
a5 = distancias[i][1][4]
a6 = distancias[i][1][5]

f = ((a1-a2)**2 + (a1-a3)**2 + (a1-a4)**2 + (a1-a5)**2 + (a1-a6)**2 +
      (a2-a3)**2 + (a2-a4)**2 + (a2-a5)**2 + (a2-a6)**2 + (a3-a4)**2 +
      (a3-a5)**2 + (a3-a6)**2 + (a4-a5)**2 + (a4-a6)**2 + (a5-a6)**2)
sum(distancias[i][1])
ep.append([f,distancias[i][0]])

#=====
#=====
#=====

for i in xrange(plan.nrows):
    dados.append(plan.row_values(i))

dados = dados[1:]

soma=[]
normal = []

i=0
while i < 1200:
    aux = dados[i:i+10][:]
    somax = somay = somaz = somaw = 0
    for j in range(10):
        somax += dados[i+j][0]
        somay += dados[i+j][1]
        somaz += dados[i+j][2]
        somaw += dados[i+j][3]
    m = max(somax,somay,somaz,somaw)

    soma.append([somax/m,somay/m,somaz/m,somaw/m])
    i+=10

X=soma[0:30][:]

Y=soma[30:60][:]

Z=soma[60:90][:]

W=soma[90:120][:]

"""

```

```
maximos = [[], [], [], []]
for i in range(30):
    maximos[0].append(max(X[i]))
    maximos[1].append(max(Y[i]))
    maximos[2].append(max(Z[i]))
    maximos[3].append(max(W[i]))

m1 = max(maximos[0])
m2 = max(maximos[1])
m3 = max(maximos[2])
m4 = max(maximos[3])

print 'Valores da nomalização - ', m1, m2, m3, m4,

for i in range(30):
    X[i] = [X[i][0]/m1, X[i][1]/m1, X[i][2]/m1, X[i][3]/m1]
    Y[i] = [Y[i][0]/m2, Y[i][1]/m2, Y[i][2]/m2, Y[i][3]/m2]
    Z[i] = [Z[i][0]/m3, Z[i][1]/m3, Z[i][2]/m3, Z[i][3]/m3]
    W[i] = [W[i][0]/m4, W[i][1]/m4, W[i][2]/m4, W[i][3]/m4]
    ""
print '>>>> ', len(X), len(Y), len(Z), len(W)

distancias = []

for i in range(30):
    for j in range(30):
        for k in range(30):
            for l in range(30):
                aux = []
                aux.append(distr4(X[i], Y[j]))
                aux.append(distr4(X[i], Z[k]))
                aux.append(distr4(X[i], W[l]))
                aux.append(distr4(Y[j], Z[k]))
                aux.append(distr4(Y[j], W[l]))
                aux.append(distr4(Z[k], W[l]))

            distancias.append([[i, j, k, l], aux])

n = len(distancias)

ep=[]

#Calculo dos Epsilons
calculaEpsilon2()

i= 1
j= 0
```

---

```
while i < n:
if ep[i][0] < ep[j][0]:
j = i
i+=1

m = j#indice do valor minimo

print '-----COORDENADAS - X-----'
for i in range(30):
print i,' = ',X[i],

print '-----COORDENADAS - Y-----'
for i in range(30):
print i,' = ',Y[i],

print '-----COORDENADAS - Z-----'
for i in range(30):
print i,' = ',Z[i],

print '-----COORDENADAS - W-----'
for i in range(30):
print i,' = ',W[i],

print 'Valor Minimo - ', m,' - ',ep[m][0],
Coordenadas dos pontos,ep[m][1][0],',',ep[m][1][1],',',ep[m][1][2],',',
ep[m][1][3]
print 'X[ep[m][1][0]],Y[ep[m][1][1]],Z[ep[m][1][2]],W[ep[m][1][3]]
```

## Apêndice D

# Gerador da quarta substância

```
# -*- coding: UTF-8 -*-

# geradorSubstancia4.py

#dados normalizados no codigos

import xlrd
import random
from numpy import*
from numpy.linalg import inv
import csv

arq = xlrd.open_workbook('4fases_dados.xls')

plan = arq.sheets()[0]

dados=[]

escreve = csv.writer(open("resultadoQuartaSubstancia.csv", "wb"))
#Objeto que escreve no arquivo .CSV
escreve.writerow(['A','C','D','NOVA'])

#=====
#=====
#=====

def distr4(a,b):
d = ((a[0]-b[0])**2 + (a[1]-b[1])**2 + (a[2]-b[2])**2 + (a[3]-b[3])**2)**(0.5)
return d

#Função da distancia no R3
def distr3(a,b):
d = ((a[0]-b[0])**2 + (a[1]-b[1])**2 + (a[2]-b[2])**2)**(0.5)
```

```
return d

def calculaEpsilon1():
#FORMULA DO EPSILON
for i in range(n):
e = (max(distancias[i][1]) - min(distancias[i][1]))/sum(distancias[i][1])
ep.append([e,distancias[i][0]])

def calculaEpsilon2():
#FORMULA DO EPSILON
for i in range(n):
a1 = distancias[i][1][0]#VALOR DOS LADOS
a2 = distancias[i][1][1]
a3 = distancias[i][1][2]

f = ((a1-a2)**2 + (a1-a3)**2 + (a2-a3)**2)/sum(distancias[i][1])
ep.append([f,distancias[i][0]])

#Calcula o produto por escalar
def produtoPescalar(e, vetor):
res = []
res.append(vetor[0] * e)
res.append(vetor[1] * e)
res.append(vetor[2] * e)

return res

def somaVetor(v1, v2):
res = [v1[0] + v2[0], v1[1]+v2[1] ,v1[2]+v2[2]]

return res

def transposta(m):
m0 = m[0]
m1 = m[1]
m2 = m[2]
m3 = m[3]

n0 = [m0[0], m1[0], m2[0], m3[0]]
n1 = [m0[1], m1[1], m2[1], m3[1]]
n2 = [m0[2], m1[2], m2[2], m3[2]]

t = [n0,n1,n2]

return t

#Metodo para fazer a diferenca de matrizes 4x1
```

---

```
def subm(m1, m2):
m = [m1[0]-m2[0], m1[1]-m2[1], m1[2]-m2[2], m1[3]-m2[3]]
return m

def tikhonov(A, g, lam, m):
at = transposta(A)

mc = dot(at, A)
i = inv(mc)

tk = dot(A,i)
tk = dot(tk, at)
tk = dot(tk, g)
tk = list(tk)
tk = distr4(tk,g)

tk = tk + (lam**2) * m

return tk

#=====
#=====
#=====

for i in xrange(plan.nrows):
dados.append(plan.row_values(i))

dados = dados[1:]

g1 = 0.2
g2 = 0.3
g3 = 0.5

for i in range(2048):
dados[i] = [dados[i][0],dados[i][2],dados[i][3]]

for i in range(2048):
ruido = random.randint(0,101)
ruido = ruido/1000.0

s = dados[i][0] * g1 + dados[i][1] * g2 + dados[i][2] * g3
s = s + s*ruido
dados[i].append(s)

soma = []
normal = []
```

```
i=0
while i < 2048:
    aux = dados[i:i+10][:]
    somax = somay = somaz = somaw = 0
    if i == 2040:
        for j in range(8):
            somax += dados[i+j][0]
            somay += dados[i+j][1]
            somaz += dados[i+j][2]
            somaw += dados[i+j][3]
        if j < 7:
            escreve.writerow([str(dados[i+j][0]),str(dados[i+j][1]),str(dados[i+j][2]),
            str(dados[i+j][3]))])
        else:
            for j in range(10):
                somax += dados[i+j][0]
                somay += dados[i+j][1]
                somaz += dados[i+j][2]
                somaw += dados[i+j][3]
            if j < 9:
                escreve.writerow([str(dados[i+j][0]),str(dados[i+j][1]),str(dados[i+j][2]),
                str(dados[i+j][3]))])

    m = max(somax,somay,somaz)
    if m == 0:
        soma.append([0,0,0,0])
        escreve.writerow(['0','0','0','0',' ','0','0','0','0'])
    else:
        soma.append([somax/m,somay/m,somaz/m,somaw/m])
        escreve.writerow([str(dados[i+9][0]),str(dados[i+9][1]),str(dados[i+9][2]),
        str(dados[i+9][3]),str(soma[-1][0]),str(soma[-1][1]),str(soma[-1][2]),
        str(soma[-1][3]))])
    i+=10

X=soma[0:40][:]

Y=soma[40:80][:]

Z=soma[80:120][:]

#W=soma[90:120][:]

print '>>>> ',len(X),len(Y),len(Z)#,len(W)

distancias = []

for i in range(40):
```

```
for j in range(40):
for k in range(40):
aux = []
aux.append(distr3(X[i],Y[j]))
aux.append(distr3(X[i],Z[k]))
aux.append(distr3(Y[j],Z[k]))
#aux.append(distr3(Y[j],Z[k]))
#aux.append(distr3(Y[j],W[1]))
#aux.append(distr3(Z[k],W[1]))

distancias.append([i,j,k],aux)

n = len(distancias)

ep=[]

#Calculo dos Epsilons
calculaEpsilon1()

i= 1
j= 0
while i < n:
if ep[i][0] < ep[j][0]:
j = i
i+=1

m = j #indice do valor minimo

print '-----COORDENADAS - X-----'
for i in range(40):
print i,' = ',X[i],

print '-----COORDENADAS - Y-----'
for i in range(40):
print i,' = ',Y[i],

print '-----COORDENADAS - Z-----'
for i in range(40):
print i,' = ',Z[i],

#print '-----COORDENADAS - W-----'
#for i in range(30):
#print i,' = ',W[i],

print 'Valor Minimo - ', m,' - ',ep[m][0],
'Coordenadas dos pontos',ep[m][1][0],',',ep[m][1][1],',',ep[m][1][2]
print 'X[ep[m][1][0]],Y[ep[m][1][1]],Z[ep[m][1][2]],'
```

---

```
while True:
    gerar = raw_input('Gerar valores de Tikhnov s/n? ')
    while not (gerar in ['s','n']):
        gerar = raw_input('Gerar valores de Tikhnov s/n? ')

    if gerar == 'n':
        break

    lam = input('Digite o valor de Lambda: ')

    ep2 = []
    i=0
    while i < n:
        ix = ep[i][1][0]
        iy = ep[i][1][1]
        iz = ep[i][1][2]
        A = [X[ix][0:3],Y[iy][0:3],Z[iz][0:3],[1,1,1]]
        g = [X[ix][3],Y[iy][3],Z[iz][3],1]
        res = tikhonov(A, g, lam, ep[i][0])

        cop = ep[i][:]
        cop.append(res)
        ep2.append(cop)

    i+=1

    i= 1
    j= 0
    while i < n:
        if ep2[i][2] < ep2[j][2]:
            j = i
            i+=1

    m = j#indice do valor minimo

    print 'Valor Minimo Tikhonov - ', m, ' - ', ep2[m][0],
    'Coordenadas dos pontos', ep2[m][1][0], ', ', ep2[m][1][1], ', ', ep2[m][1][2]
    print X[ep2[m][1][0]], Y[ep2[m][1][1]], ', ', Z[ep2[m][1][2]],
```

## Apêndice E

# Lista de dados normalizados - Problema de três fases

Neste apêndice apresentamos a lista de dados normalizados para o caso com três fases, com os intervalos de canais em cada partição, e em seguida os resultados da normalização e os canais indicados pelo programa são destacados. Por fim, apresentamos um exemplo de resultados obtidos ao realizarmos a regularização geométrica, escolhendo o valor  $\lambda = 0.1$ .

```
Python 2.7.9 (default, Dec 10 2014, 12:28:03) [MSC v.1500 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
```

```
>>>> 40 40 40
```

```
-----COORDENADAS - X-----
```

```
0 = [0.4434244455905106, 0.6027849406910779, 1.0, 0.7933246725116039]
```

```
1 = [0.44300822561692127, 0.6028202115158637, 1.0, 0.8107451938895417]
```

```
2 = [0.4437412095639944, 0.6033755274261603, 1.0, 0.8135637834036569]
```

```
3 = [0.44233496999454447, 0.6042553191489362, 1.0, 0.8106968739770868]
```

```
4 = [0.4438102893890675, 0.6022508038585209, 1.0, 0.8008938987138263]
```

```
5 = [0.44258373205741625, 0.6028708133971292, 1.0, 0.8015150608960417]
```

```
6 = [0.44263775971093045, 0.6050888286660644, 1.0, 0.7993680216802168]
```

---

7 = [0.44284284284284287, 0.6038038038038038, 1.0, 0.8076226826826827]  
8 = [0.4438489646772229, 0.6024360535931791, 1.0, 0.8133492082825822]  
9 = [0.4430952380952381, 0.6038095238095238, 1.0, 0.8095056428571429]  
10 = [0.4429165334186121, 0.6031339942436841, 1.0, 0.8005083466581387]  
11 = [0.4435389988358556, 0.6018626309662398, 1.0, 0.8199550252231277]  
12 = [0.44463058419243984, 0.6026632302405498, 1.0, 0.799203823024055]  
13 = [0.44281045751633985, 0.6045751633986928, 1.0, 0.8120115219421101]  
14 = [0.44369323915237135, 0.6029263370332997, 1.0, 0.8194123208879919]  
15 = [0.4439353099730458, 0.6021563342318059, 1.0, 0.8131779299191375]  
16 = [0.44393162393162394, 0.6022792022792023, 1.0, 0.8191511452991453]  
17 = [0.44261603375527425, 0.6027727546714888, 1.0, 0.8006736407474382]  
18 = [0.4428752374920836, 0.6029765674477517, 1.0, 0.8116911019632679]  
19 = [0.4434668417596848, 0.6034799737360472, 1.0, 0.7985957386736704]  
20 = [0.44440677966101694, 0.6024406779661017, 1.0, 0.8167519525423729]  
21 = [0.44466666666666665, 0.6028666666666667, 1.0, 0.7996822333333333]  
22 = [0.443142144638404, 0.6031795511221946, 1.0, 0.8000689089775561]  
23 = [0.4398064961990325, 0.6070490670352453, 1.0, 0.8074279060124395]  
24 = [0.44242209631728047, 0.6038243626062323, 1.0, 0.8154832790368272]  
25 = [0.4432664756446991, 0.602650429799427, 1.0, 0.8094405873925501]  
26 = [0.4432276657060519, 0.6020172910662824, 1.0, 0.8057684077809798]  
27 = [0.4441304347826087, 0.6016666666666667, 1.0, 0.8089454637681159]  
28 = [0.4436999271667881, 0.601456664238893, 1.0, 0.7960939621267298]  
29 = [0.4445995591476855, 0.6030124908155767, 1.0, 0.8163125716385011]  
30 = [0.44467073998642226, 0.6020366598778004, 1.0, 0.8212215614392396]

---

31 = [0.4445982142857143, 0.6008928571428571, 1.0, 0.8057832767857143]  
32 = [0.4443913043478261, 0.601304347826087, 1.0, 0.8002621652173914]  
33 = [0.44427083333333334, 0.6006944444444444, 1.0, 0.8093369907407407]  
34 = [0.44445076835515085, 0.6010244735344337, 1.0, 0.8004461923733637]  
35 = [0.4448315248429469, 0.6013706453455169, 1.0, 0.8072178183894917]  
36 = [0.44409410623332524, 0.601018675721562, 1.0, 0.8099054733608214]  
37 = [0.44387391242333474, 0.6023391812865497, 1.0, 0.8123828412494651]  
38 = [0.44360902255639095, 0.6025528938625634, 1.0, 0.8060332051057877]  
39 = [0.44523505232979865, 0.6011882145694449, 1.0, 0.817763389865635]

-----COORDENADAS - Y-----

0 = [0.4446439257929384, 0.6001396369439458, 1.0, 0.8161082186315579]  
1 = [0.4448779260005034, 0.6006795872136924, 1.0, 0.8048382708280896]  
2 = [0.4446463335496431, 0.6007787151200519, 1.0, 0.7937488059701493]  
3 = [0.43722986247544204, 0.6147129447718839, 1.0, 0.802690506439642]  
4 = [0.2537317073170732, 0.9221951219512196, 1.0, 0.8673797414634147]  
5 = [0.09878133704735376, 1.0, 0.883008356545961, 0.8053093036211699]  
6 = [0.11370942325874567, 1.0, 0.8821304758903246, 0.803434724235739]  
7 = [0.19072404768828147, 1.0, 0.8822332073277116, 0.8211601163128815]  
8 = [0.20005884083553988, 1.0, 0.881729920564872, 0.8140292497793469]  
9 = [0.12078599938593798, 1.0, 0.882407123119435, 0.8122219649984649]  
10 = [0.09168411606341609, 1.0, 0.8827400538438528, 0.7893795692491775]  
11 = [0.08101227912548667, 1.0, 0.8820005989817311, 0.8061797604073077]  
12 = [0.07206703910614524, 1.0, 0.88268156424581, 0.8045180509000622]

---

13 = [0.06993827160493828, 1.0, 0.8827160493827161, 0.7930973765432099]  
14 = [0.07511339582703357, 1.0, 0.8823707287571817, 0.7926854248563653]  
15 = [0.07825242718446603, 1.0, 0.8821081830790569, 0.785616882108183]  
16 = [0.05957854406130268, 1.0, 0.8826628352490421, 0.7955145593869732]  
17 = [0.04006217213911016, 1.0, 0.8824557994948514, 0.7835382824946572]  
18 = [0.05229925716307039, 1.0, 0.8825610187477891, 0.7851985956844713]  
19 = [0.082480794453813, 1.0, 0.8823308975079632, 0.8112099531572043]  
20 = [0.0711760184473482, 1.0, 0.8823981552651806, 0.7861163989239047]  
21 = [0.040831024930747924, 1.0, 0.8821791320406279, 0.7955346408125576]  
22 = [0.031933099961104626, 1.0, 0.882730455075846, 0.7992907156748348]  
23 = [0.03472764881636405, 1.0, 0.8823259020914732, 0.7976221006665135]  
24 = [0.035465116279069765, 1.0, 0.8820251937984496, 0.7848460949612402]  
25 = [0.026607301869991096, 1.0, 0.8824577025823687, 0.7832799216384684]  
26 = [0.016708749588319246, 1.0, 0.8816555055439675, 0.7782531210890329]  
27 = [0.01466896132820409, 1.0, 0.8831747317270703, 0.7804971613686982]  
28 = [0.02041146216017634, 1.0, 0.8822924320352682, 0.7856220073475387]  
29 = [0.024773468648060893, 1.0, 0.882384922073215, 0.7737024175425878]  
30 = [0.014941892639734366, 1.0, 0.8821250691754289, 0.7782013325954621]  
31 = [0.00991778774289985, 1.0, 0.8834080717488789, 0.7806208953662183]  
32 = [0.011288147024110606, 1.0, 0.881554543921767, 0.7727930079244646]  
33 = [0.018259682224428996, 1.0, 0.8820754716981132, 0.7733631256206555]  
34 = [0.027587851804738432, 1.0, 0.8821752265861027, 0.7821525552552074]  
35 = [0.03456369487033882, 1.0, 0.8824531516183987, 0.7735429642248722]

---

36 = [0.03930494371023006, 1.0, 0.882525697503671, 0.7795597454723445]

37 = [0.041452480573819485, 1.0, 0.8822474596533174, 0.785151380753138]

38 = [0.022753872633390706, 1.0, 0.8824440619621342, 0.7832236351118761]

39 = [0.008388157894736842, 1.0, 0.8825278340080972, 0.7761491864878541]

-----COORDENADAS - Z-----

0 = [0.0050130548302872065, 1.0, 0.8817605371130175, 0.7966982506527415]

1 = [0.005869565217391304, 1.0, 0.8827791986359761, 0.788655905370844]

2 = [0.012617065556711759, 1.0, 0.8837148803329865, 0.7932902757544226]

3 = [0.03345548609614141, 1.0, 0.8825177840051736, 0.7982596421642594]

4 = [0.0605192821687667, 1.0, 0.8820160366552119, 0.7831676365024819]

5 = [0.0877044557247603, 1.0, 0.8809926677946982, 0.7896391765369432]

6 = [0.1175890826383624, 1.0, 0.8824867323730099, 0.7937886884003031]

7 = [0.06614473127380449, 1.0, 0.8823529411764706, 0.7924170122725347]

8 = [0.024057971014492755, 1.0, 0.8822273073989321, 0.7804773760488177]

9 = [0.014770546152503927, 1.0, 0.8809108358052696, 0.7734843255976269]

10 = [0.02121071660331881, 1.0, 0.881894873458793, 0.7788994437749143]

11 = [0.06556957396233315, 1.0, 0.882062534284147, 0.8019857121960139]

12 = [0.3073115258545722, 1.0, 0.8816643253729346, 0.8398351394742124]

13 = [1.0, 0.5011179557380789, 0.44184348619666897, 0.6054307164042894]

14 = [1.0, 0.04577344376168849, 0.0404034197168047, 0.2432746780657227]

15 = [1.0, 0.002006400231682595, 0.0017693310165073847, 0.21334572041702868]

16 = [1.0, 1.5414555097230407e-05, 1.360822038892163e-05, 0.2098926577438185]

17 = [1.0, 1.2609770808202652e-08, 1.113235343787696e-08, 0.20921701829186776]

---

18 = [1.0, 0.0, 0.0, 0.20924977777777778]

19 = [1.0, 0.0, 0.0, 0.21171609455292908]

20 = [1.0, 0.0, 0.0, 0.20612328209304692]

21 = [1.0, 0.0, 0.0, 0.20879876543209877]

22 = [1.0, 0.0, 0.0, 0.20898763005780344]

23 = [1.0, 0.0, 0.0, 0.20889881235154392]

24 = [1.0, 0.0, 0.0, 0.20713759980346394]

25 = [1.0, 0.0, 0.0, 0.20874819256467358]

26 = [1.0, 0.0, 0.0, 0.2075060732498841]

27 = [1.0, 0.0, 0.0, 0.20866068148148148]

28 = [1.0, 0.0, 0.0, 0.20954559606718562]

29 = [1.0, 0.0, 0.0, 0.20988818606290194]

30 = [1.0, 0.0, 0.0, 0.20976320156173744]

31 = [1.0, 0.0, 0.0, 0.21277595207531025]

32 = [1.0, 0.0, 0.0, 0.21150494807574832]

33 = [1.0, 0.0, 0.0, 0.20961197689164435]

34 = [1.0, 0.0, 0.0, 0.21089310137972406]

35 = [1.0, 0.0, 0.0, 0.2089686290486861]

36 = [1.0, 0.0, 0.0, 0.20726575443786982]

37 = [1.0, 0.0, 0.0, 0.21297888198757767]

38 = [1.0, 0.0, 0.0, 0.21207022256331545]

39 = [1.0, 0.0, 0.0, 0.2102003600360036]

Valor Minimo - 38372 - 0.223576302692

---

Coordenadas dos pontos

23 , 39 , 12

[0.4398064961990325, 0.6070490670352453, 1.0, 0.8074279060124395]

[0.008388157894736842, 1.0, 0.8825278340080972, 0.7761491864878541]

[0.3073115258545722, 1.0, 0.8816643253729346, 0.8398351394742124]

Gerar valores de Tikhnov s/n? s

Digite o valor de Lambda: 0.1

Valor Minimo Tikhonov - 27361 - 0.251898168195

Coordenadas dos pontos:

17 , 4 , 1

[0.44261603375527425, 0.6027727546714888, 1.0, 0.8006736407474382]

[0.2537317073170732, 0.9221951219512196, 1.0, 0.8673797414634147]

[0.005869565217391304, 1.0, 0.8827791986359761, 0.788655905370844]

Gerar valores de Tikhnov s/n?

## Apêndice F

# Lista de dados normalizados - Problema com quatro fases

```
Python 2.7.9 (default, Dec 10 2014, 12:28:03) [MSC v.1500 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>

>>>> 30 30 30 30

-----COORDENADAS - X-----

0 = [0.4434244455905106, 0.85146982980918, 0.6027849406910779, 1.0]

1 = [0.44300822561692127, 0.854289071680376, 0.6028202115158637, 1.0]

2 = [0.4437412095639944, 0.8284106891701828, 0.6033755274261603, 1.0]

3 = [0.44233496999454447, 0.839607201309329, 0.6042553191489362, 1.0]

4 = [0.4438102893890675, 0.8264469453376205, 0.6022508038585209, 1.0]

5 = [0.44258373205741625, 0.8170943888647237, 0.6028708133971292, 1.0]

6 = [0.44263775971093045, 0.860132490213791, 0.6050888286660644, 1.0]

7 = [0.44284284284284287, 0.8932932932932933, 0.6038038038038038, 1.0]

8 = [0.4438489646772229, 0.9792935444579781, 0.6024360535931791, 1.0]

9 = [0.3234833999652355, 1.0, 0.4408134886146358, 0.730053884929602]
```

---

10 = [0.37251210328133405, 1.0, 0.5072619688004303, 0.8410435718128025]  
11 = [0.4435389988358556, 0.8253783469150174, 0.6018626309662398, 1.0]  
12 = [0.44463058419243984, 0.8195876288659794, 0.6026632302405498, 1.0]  
13 = [0.44281045751633985, 0.823062558356676, 0.6045751633986928, 1.0]  
14 = [0.3255831173639393, 1.0, 0.442428730099963, 0.7338022954461311]  
15 = [0.4088877855014896, 1.0, 0.554617676266137, 0.9210526315789473]  
16 = [0.44393162393162394, 0.7601139601139602, 0.6022792022792023, 1.0]  
17 = [0.44261603375527425, 0.733574442435202, 0.6027727546714888, 1.0]  
18 = [0.4428752374920836, 0.6694110196326789, 0.6029765674477517, 1.0]  
19 = [0.4434668417596848, 0.6400525279054498, 0.6034799737360472, 1.0]  
20 = [0.44440677966101694, 0.7410847457627119, 0.6024406779661017, 1.0]  
21 = [0.3247322297955209, 1.0, 0.44026290165530674, 0.7302823758519961]  
22 = [0.443142144638404, 0.8115960099750623, 0.6031795511221946, 1.0]  
23 = [0.4398064961990325, 0.568279198341396, 0.6070490670352453, 1.0]  
24 = [0.44242209631728047, 0.5743626062322946, 0.6038243626062323, 1.0]  
25 = [0.4432664756446991, 0.5082378223495702, 0.602650429799427, 1.0]  
26 = [0.4432276657060519, 0.4892651296829971, 0.6020172910662824, 1.0]  
27 = [0.4441304347826087, 0.4921014492753623, 0.6016666666666667, 1.0]  
28 = [0.4436999271667881, 0.4764748725418791, 0.601456664238893, 1.0]  
29 = [0.4445995591476855, 0.5417340191036003, 0.6030124908155767, 1.0]  
  
-----COORDENADAS - Y-----  
0 = [0.44467073998642226, 0.49626612355736593, 0.6020366598778004, 1.0]  
1 = [0.4445982142857143, 0.276875, 0.6008928571428571, 1.0]

---

2 = [0.4443913043478261, 0.2288695652173913, 0.601304347826087, 1.0]  
3 = [0.44427083333333334, 0.2818287037037037, 0.6006944444444444, 1.0]  
4 = [0.44445076835515085, 0.3300512236767217, 0.6010244735344337, 1.0]  
5 = [0.4448315248429469, 0.6826956025128498, 0.6013706453455169, 1.0]  
6 = [0.44409410623332524, 0.974937343358396, 0.601018675721562, 1.0]  
7 = [0.44387391242333474, 0.9453715589787477, 0.6023391812865497, 1.0]  
8 = [0.44360902255639095, 0.8788249694002448, 0.6025528938625634, 1.0]  
9 = [0.44523505232979865, 0.15861282857241546, 0.6011882145694449, 1.0]  
10 = [0.4446439257929384, 0.0438460003989627, 0.6001396369439458, 1.0]  
11 = [0.4448779260005034, 0.054241127611376797, 0.6006795872136924, 1.0]  
12 = [0.4446463335496431, 0.2765736534717716, 0.6007787151200519, 1.0]  
13 = [0.09650445675740785, 1.0, 0.13567814984341123, 0.22071789930137317]  
14 = [0.0024053179190751447, 1.0, 0.008742196531791908, 0.009479768786127168]  
15 = [0.0006004232804232804, 1.0, 0.006078306878306878, 0.005367195767195767]  
16 = [0.0008104222821203953, 1.0, 0.007127133872416891, 0.006287061994609164]  
17 = [0.0016575688653019964, 1.0, 0.008690927470305787, 0.007667424816780389]  
18 = [0.001831896551724138, 1.0, 0.009156788793103448, 0.008073814655172413]  
19 = [0.0011540041067761807, 1.0, 0.009554121443238486, 0.008430624816661778]  
20 = [0.000896985659935616, 1.0, 0.009783435762364647, 0.008636230611647644]  
21 = [0.0006442010002381519, 1.0, 0.00795189330793046, 0.007013574660633484]  
22 = [0.0005136031851360319, 1.0, 0.007126741871267419, 0.006290643662906437]  
23 = [0.0005989955062120011, 1.0, 0.008564631245043617, 0.007560137457044674]  
24 = [0.0006831683168316831, 1.0, 0.009095159515951596, 0.008025302530253025]  
25 = [0.0007251928020565552, 1.0, 0.009267352185089974, 0.008174807197943446]

---

26 = [0.0006913031397610447, 1.0, 0.011603223117532648, 0.010241733814948598]  
27 = [0.0007174669450243563, 1.0, 0.01790883785664579, 0.015803757828810022]  
28 = [0.0012035002035002036, 1.0, 0.02301180301180301, 0.020309320309320308]  
29 = [0.0017872513195290297, 1.0, 0.02166869671132765, 0.019118960617133578]

-----COORDENADAS - Z-----

0 = [0.0014290123456790124, 1.0, 0.02007716049382716, 0.01771604938271605]  
1 = [0.0008708152816069318, 1.0, 0.021327294210319023, 0.018814493895234343]  
2 = [0.0006813278008298755, 1.0, 0.02133609958506224, 0.01883402489626556]  
3 = [0.0006205338809034907, 1.0, 0.01786858316221766, 0.015765913757700204]  
4 = [0.000559205500381971, 1.0, 0.015767761650114592, 0.013907563025210084]  
5 = [0.0005578790141896937, 1.0, 0.020967139656460045, 0.018502613890963405]  
6 = [0.0006297062474141498, 1.0, 0.037687215556474966, 0.033227141083988415]  
7 = [0.0006643741403026135, 1.0, 0.045291150848234756, 0.04]  
8 = [0.000635116598079561, 1.0, 0.03111568358481939, 0.027453132144490168]  
9 = [0.0006102678571428572, 1.0, 0.02463392857142857, 0.021736607142857144]  
10 = [0.0006279069767441861, 1.0, 0.04202325581395349, 0.03706976744186047]  
11 = [0.0006253534401508012, 1.0, 0.06305372290292177, 0.05570216776625825]  
12 = [0.0006007178106774338, 1.0, 0.053216689098250336, 0.04691341408703455]  
13 = [0.0006404005224205485, 1.0, 0.03507183282542447, 0.03093600348280366]  
14 = [0.0007801258992805756, 1.0, 0.028277877697841725, 0.024946043165467627]  
15 = [0.000902619871478003, 1.0, 0.026114681166584282, 0.02304498269896194]  
16 = [0.000867170626349892, 1.0, 0.022062634989200865, 0.01947084233261339]  
17 = [0.0007831733483907397, 1.0, 0.018893280632411066, 0.016668548842461887]

---

18 = [0.0007584624211130235, 1.0, 0.03333333333333333, 0.029414802065404474]  
19 = [0.0007202607278652906, 1.0, 0.08586637696903857, 0.07577946768060836]  
20 = [0.0006452232357177149, 1.0, 0.12870859337493998, 0.11349015842534806]  
21 = [0.0006150066994193836, 1.0, 0.10477891916033943, 0.09249665029030818]  
22 = [0.0006631722880583409, 1.0, 0.05256153144940748, 0.046449407474931634]  
23 = [0.0007418738049713193, 1.0, 0.022174952198852772, 0.0195697896749522]  
24 = [0.0007909181636726547, 1.0, 0.013068862275449102, 0.01152694610778443]  
25 = [0.0008036175710594315, 1.0, 0.009162790697674419, 0.008072351421188631]  
26 = [0.0008406504065040651, 1.0, 0.007149051490514905, 0.006308943089430894]  
27 = [0.0008921232876712328, 1.0, 0.013487442922374429, 0.011900684931506849]  
28 = [0.0008650575973669775, 1.0, 0.03595721338453099, 0.03172243554580362]  
29 = [0.0007889095992544269, 1.0, 0.053410997204100655, 0.047050326188257226]

-----COORDENADAS - W-----

0 = [0.000927066450567261, 1.0, 0.04370745542949757, 0.038545380875202595]  
1 = [0.0014349739895958384, 1.0, 0.021884753901560625, 0.019303721488595438]  
2 = [0.0019474353539635438, 1.0, 0.006337007206443408, 0.0055871131835523525]  
3 = [0.0018770877944325481, 1.0, 0.0009406423982869379, 0.0008293790149892933]  
4 = [0.0015589337775926697, 1.0, 7.135776759683466e-05, 6.29862557267805e-05]  
5 = [0.001509838216003498, 1.0, 3.0293397463926543e-06, 2.6714035854831655e-06]  
6 = [0.0016596577017114914, 1.0, 2.5582885085574572e-08, 2.2584987775061133e-08]  
7 = [0.0017397691500524658, 1.0, 2.1938090241343124e-11, 1.9367725078698844e-11]  
8 = [0.0017423851316468767, 1.0, 0.0, 0.0]  
9 = [0.0021384615384615383, 1.0, 0.0, 0.0]

---

10 = [0.0032376400791034937, 1.0, 0.0, 0.0]  
11 = [0.004298222340143274, 1.0, 0.0, 0.0]  
12 = [0.004402035623409669, 1.0, 0.0, 0.0]  
13 = [0.00344141689373297, 1.0, 0.0, 0.0]  
14 = [0.0025126543209876544, 1.0, 0.0, 0.0]  
15 = [0.0021483253588516745, 1.0, 0.0, 0.0]  
16 = [0.0018723958333333333, 1.0, 0.0, 0.0]  
17 = [0.0015837634913186298, 1.0, 0.0, 0.0]  
18 = [0.001997545008183306, 1.0, 0.0, 0.0]  
19 = [0.0031644828547532757, 1.0, 0.0, 0.0]  
20 = [0.004161251015434606, 1.0, 0.0, 0.0]  
21 = [0.004614017769002962, 1.0, 0.0, 0.0]  
22 = [0.004128625472887768, 1.0, 0.0, 0.0]  
23 = [0.002484943977591036, 1.0, 0.0, 0.0]  
24 = [0.0008735371179039302, 1.0, 0.0, 0.0]  
25 = [0.00023911349245007307, 1.0, 0.0, 0.0]  
26 = [0.00014017625712804563, 1.0, 0.0, 0.0]  
27 = [0.0002140081799591002, 1.0, 0.0, 0.0]  
28 = [0.00033844155844155844, 1.0, 0.0, 0.0]  
29 = [0.00039037245256500353, 1.0, 0.0, 0.0]

Valor Minimo - 567621 - 1.08297775167

Coordenadas dos pontos: 21 , 0 , 20 , 21

[0.3247322297955209, 1.0, 0.44026290165530674, 0.7302823758519961]

[0.44467073998642226, 0.49626612355736593, 0.6020366598778004, 1.0]

[0.0006452232357177149, 1.0, 0.12870859337493998, 0.11349015842534806]

[0.004614017769002962, 1.0, 0.0, 0.0]

>>>