

Volume 97, 2023

Corpo Editorial

Sandra Mara Cardoso Malta (Editor Chefe)

Laboratório Nacional de Computação Científica - LNCC
Petrópolis, RJ, Brasil

Eduardo V. O. Teixeira (Editor Executivo)

University of Central Florida - UCF
Orlando, FL, EUA

Lilian Markenzon

Universidade Federal do Rio de Janeiro - UFRJ
Rio de Janeiro, RJ, Brasil

Marcelo Sobottka

Universidade Federal de Santa Catarina - UFSC
Florianópolis, SC, Brasil

Paulo F. de Arruda Mancera

Universidade Estadual Paulista Júlio de Mesquita Filho- UNESP
Botucatu, SP, Brasil

Sandra Augusta Santos

Universidade Estadual de Campinas - UNICAMP
Campinas, SP, Brasil

Tânia Schmitt

Universidade de Brasília - UnB
Brasília, DF, Brasil

A Sociedade Brasileira de Matemática Aplicada e Computacional - SBMAC publica, desde as primeiras edições do evento, monografias dos cursos que são ministrados nos CNMAC.

Para a comemoração dos 25 anos da SBMAC, que ocorreu durante o XXVI CNMAC em 2003, foi criada a série **Notas em Matemática Aplicada** para publicar as monografias dos minicursos ministrados nos CNMAC, o que permaneceu até o XXXIII CNMAC em 2010.

A partir de 2011, a série passa a publicar, também, livros nas áreas de interesse da SBMAC. Os autores que submeterem textos à série Notas em Matemática Aplicada devem estar cientes de que poderão ser convidados a ministrarem minicursos nos eventos patrocinados pela SBMAC, em especial nos CNMAC, sobre assunto a que se refere o texto.

O livro deve ser preparado em **Latex, com as figuras em .eps, .pdf e etc.** e ter entre **80 e 150 páginas**. O texto deve ser redigido de forma clara, acompanhado de uma excelente revisão bibliográfica e de **exercícios de verificação de aprendizagem** ao final de cada capítulo. O idioma pode ser Português ou Espanhol.

Veja todos os títulos publicados nesta série na página
<http://https://proceedings.science/notas-sbmac>

Otimização sob incerteza: aspectos teóricos, computacionais e aplicações

Luiz Leduino de Salles Neto
luiz.leduino@unifesp.br

Departamento de Ciência e Tecnologia
Instituto de Ciência e Tecnologia
Universidade Federal de São Paulo

Weldon Lodwick
wlodwick@gmail.com

Department of Mathematical and Statistical Sciences
University of Colorado Denver



Sociedade Brasileira de Matemática Aplicada e Computacional

São Carlos - SP, Brasil
2023

Coordenação Editorial: Mateus Bernardes

Coordenação Editorial da Série: Sandra M. C. Malta

Editora: SBMAC

Capa: Matheus Botossi Trindade

Patrocínio: SBMAC

Copyright ©2023 by Luiz Leduino de Salles Neto e Weldon Lodwick Direitos reservados, 2023 pela SBMAC. A publicação nesta série não impede o autor de publicar parte ou a totalidade da obra por outra editora, em qualquer meio, desde que faça citação à edição original.

Catálogo elaborado pela Biblioteca do IBILCE/UNESP
Bibliotecária: Maria Luiza Fernandes Jardim Froner

Salles-Neto, Luiz Leduino de
Otimização sob incerteza:
aspectos teóricos, computacionais e aplicações - São Carlos, SP :
SBMAC, 2023, 109 p., 21,5 cm - (Notas em Matemática
Aplicada; v. 97)

ISBN 978-65-86388-23-7 e-ISBN 978-65-86388-22-0

1. Otimização 2. Incerteza 3. Otimização sob incerteza.
4. Análise Intervalar 5. Modelagem matemática
I. Salles-Neto, Luiz Leduino de II. Lodwick, Weldon
IV. Título. V. Série

CDD - 51

Dedicamos às nossas famílias!

Agradecimentos

Os autores agradecem à Fapesp pelo financiamento (processo número 2021/03269-7) e à Universidade Federal de São Paulo e à Universidade do Colorado Denver por oferecerem as condições para que este livro fosse escrito. Os autores também são muito gratos ao revisor pelas relevantes sugestões apresentadas e que tornaram o livro mais didático.

Verdade, mentira, certeza, incerteza...
Aquele cego ali na estrada também conhece estas
palavras.
Estou sentado num degrau alto e tenho as mãos
apertadas
Sobre o mais alto dos joelhos cruzados.
Bem: verdade, mentira, certeza, incerteza o que
são?
O cego pára na estrada,
Desliguei as mãos de cima do joelho
Verdade mentira, certeza, incerteza são as mes-
mas?
Qualquer coisa mudou numa parte da realidade
— os meus joelhos e as minhas mãos.
Qual é a ciência que tem conhecimento para isto?
O cego continua o seu caminho e eu não faço mais
gestos.
Já não é a mesma hora, nem a mesma gente, nem
nada igual.
Ser real é isto.

Alberto Caeiro (Heterónimo de Fernando Pessoa)

Conteúdo

Prefácio	xv
1 Uma introdução à otimização sob incerteza generalizada	1
1.1 Introdução	1
1.2 Possibilidade e Necessidade	6
1.3 Otimização sob Incerteza Generalizada	8
1.3.1 Intervalos	11
1.3.2 Intervalos <i>fuzzy</i>	12
1.3.3 Intervalos de Possibilidade	12
1.4 Uma Taxonomia de Incerteza Generalizada em Otimização	13
1.5 Resumo do Capítulo	15
1.6 Exercícios do Capítulo	15
2 Análise Intervalar	17
2.1 Introdução	17
2.2 Aritmética Intervalar Padrão - SIA	17
2.3 Aritmética Intervalar Restrita - CIA	18
2.3.1 Exemplificando as diferenças entre o SIA e o CIA	18
2.4 Sistemas Lineares Intervalares	19
2.4.1 Quatro Casos	20
2.4.2 Caso 4:	24
2.5 O Algoritmo Pivkina/Kreinovich	25
2.6 Resumo do capítulo	25
2.7 Exercícios	25
3 Otimização sob incerteza: abordagens e medidas	27
3.1 Introdução e exemplos	27
3.2 Minimizar o máximo arrependimento em um problema de otimização inteira	29
3.3 Minimizar o máximo arrependimento em um problema de otimização multiobjetivo	31
3.4 Mensurando a qualidade de uma solução: a dureza (hardness) de uma solução	39
3.4.1 Otimização Robusta	40
3.4.2 Probabilidade de violação do conjunto de restrições	40
3.4.3 Dureza	41
3.5 De volta ao problema do fazendeiro	42
3.6 Resumo do Capítulo	44
3.7 Exercícios	44

4	Aplicações	47
4.1	Introdução	47
4.2	O Problema de Geometria de Distâncias sob Incerteza	47
4.3	O problema de localização de sensores sob incerteza	53
4.4	O problema de otimização linear intervalar	55
4.5	O problema de corte de estoque sob incerteza	57
4.6	Finanças e otimização sob incerteza	62
4.7	Aplicações ao enfrentamento da covid-19	64
4.8	Resumo do capítulo	66
4.9	Exercícios	66
5	Implementação e experimentos computacionais	69
5.1	Introdução	69
5.2	Como usar o Python em otimização	69
5.2.1	PULP	69
5.2.2	GEKKO	73
5.3	Como usar o AMPL	75
5.3.1	Resolvendo o Problema da Mochila	75
5.3.2	Resolvendo o Problema de Geometria de Distâncias	78
5.4	Como usar o Neos	82
5.5	Resumo do capítulo	83
5.6	Exercícios	83

Prefácio

A incerteza é inerente a diversas situações e problemas da humanidade. Assim, entender, definir, quantificar e considerar a incerteza em problemas de otimização é fundamental nos processos de tomada de decisão. Neste livro nós apresentamos uma discussão sobre a semântica da incerteza, sua definição matemática, resultados teóricos decorrentes, bem como abordagens e métodos de otimização para problemas sob incerteza provenientes de projetos que tiveram a nossa coordenação ou participação. Desta forma, também abordamos a implementação computacional, apresentando um tutorial de como usar duas das ferramentas mais utilizadas na área de otimização: o Python e o AMPL.

Assim, por unir aplicações reais e a teoria que fundamenta as abordagens e modelos matemáticos desenvolvidos, este livro objetiva contribuir com o avanço na incorporação da incerteza em modelos de otimização utilizados para auxiliar a tomada de decisão nos mais variados setores da sociedade. Este é um dos objetivos principais do projeto de pesquisa apoiado pela Fapesp que financiou a escrita deste livro e o desenvolvimento de algumas das aplicações aqui apresentadas (processo número 2021/03269-7).

Mais detalhadamente, no capítulo 1 apresentamos nosso entendimento sobre os tipos de problemas de otimização sob incerteza: otimização flexível e otimização sob incerteza generalizada. Destacamos a motivação e a importância da semântica em cada caso.

O capítulo 2 aborda a análise intervalar, já que uma forma eficiente de modelar problemas de otimização sob incerteza generalizada é por meio de intervalos.

O capítulo 3 apresenta diferentes abordagens, do ponto de vista do tomador de decisão, sobre o problema de otimização sob incerteza generalizada. Detalhamos a abordagem que visa minimizar o máximo arrependimento com foco em problemas bi-objetivo. Também apresentamos uma medida para mensurar a qualidade da solução obtida em um problema de otimização sob incerteza generalizada, incluindo o importante conceito de otimização robusta.

O capítulo 4 apresenta algumas aplicações, tendo por base a nossa experiência prática. Abordamos o Problema de Geometria de Distâncias sob Incerteza, o Problema de localização de sensores/alvos sob incerteza, o problema de corte de estoque sob incerteza, o problema de detecção de esclerose múltipla, o clássico problema de otimização de portfólio e algumas aplicações que desenvolvemos para auxiliar o enfrentamento da pandemia de covid-19.

O capítulo 5 traz um pequeno tutorial de como implementar computacionalmente os modelos matemáticos apresentados nos capítulos anteriores. Em especial, abordamos a linguagem Python e o software de modelagem AMPL, duas das ferramentas mais utilizadas em empresas e universidades para resolução de problemas de otimização. Vale ressaltar que este capítulo pode ser consultado/estudado a qualquer momento, não dependendo dos capítulos anteriores. Ou seja, se em al-

gum momento da leitura o leitor desejar implementar um modelo apresentado, ou usar o Python ou AMPL para resolver um exercício, o capítulo 5 pode e deve ser consultado.

Para um acompanhamento e desenvolvimento pleno do conteúdo apresentado espera-se que o leitor tenha conhecimentos básicos (nível de um curso de graduação) de otimização, em especial da modelagem matemática.

Da teoria à prática, esperamos que este livro possa contribuir com todos que atuam na área de modelagem matemática e otimização.

Denver, 1 de fevereiro de 2023.

Luiz Leduino de Salles Neto
Weldon Lodwick

Capítulo 1

Uma introdução à otimização sob incerteza generalizada

1.1 Introdução

A incerteza está presente em muitas situações do nosso dia a dia, em particular em problemas complexos da sociedade, como na área de saúde, produção, meio ambiente, monitoriamento e prevenção de desastres naturais, cidades inteligentes, etc. Neste livro objetivamos definir e mostrar como abordar a incerteza em problemas e modelos matemáticos de otimização. Ou seja, quando buscamos a melhor solução de um problema, como podemos considerar a incerteza dos dados?

Temos como objetivo apresentar teoria e prática de otimização flexível e sob incerteza generalizada. Assumimos que o leitor tem familiaridade com noções básicas sobre otimização.

Permita-nos começar com um exemplo que introduz as ideias e abordagens deste livro.

Exemplo 1.1. *Suponha que nós temos uma informação incompleta sobre o fornecimento e o preço de petróleo proveniente do oriente médio devido a fatores como instabilidade da produção, situações de conflito e guerra. Além disso, suponha que estamos preocupados em abastecer a demanda de vários tipos de gasolina para o Brasil no mês de janeiro. Como nós modelamos a seguinte situação?*

“Haverá mais ou menos 50000 barris de petróleo do Oriente Médio disponíveis por mais ou menos x dólares por barril”?

Como abordar matematicamente a seguinte informação?

“A demanda do mercado no Brasil para o mais qualificado grau de gasolina é cerca de 40000 litros em janeiro”

Por fim, suponha que o CEO diga “A empresa precisa aumentar o máximo possível o lucro obtido em seu melhor ano.”

Como descrever matematicamente “mais ou menos”, “cerca de”, ou “aumentar o máximo possível” em um modelo matemático se não são fornecidas informações ou dados precisos?

Este livro busca definir como ser matematicamente preciso perante imprecisões e incertezas como:

- Nos dados de entrada - quantidade de petróleo disponível;

- No processamento - número de litros produzidos de várias qualidades;
- No objetivo - lucro.

Em cada fase (entrada, produção, objetivo) existe uma variedade de incertezas não necessariamente probabilísticas. Por exemplo, o suprimento de petróleo do Iraque durante uma guerra civil afeta o número de barris e o preço do petróleo de forma não probabilística. Isso não implica que uma abordagem probabilística não pode ser usada. Entretanto, este livro entende como mais apropriado abordar as incertezas na entrada ou saída utilizando intervalos ou conjuntos *fuzzy* (*fuzzy*). Estes conceitos serão definidos como *incerteza generalizada*. É claro que em casos reais uma decisão precisa ser tomada. Por exemplo, no caso supra mencionado, a produção de gasolina para o Brasil ocorrerá, isto é, a gasolina será entregue em uma quantidade específica no mês de janeiro. Nossa filosofia é considerar toda a incerteza (ou o máximo possível) *em sua forma nativa* até uma decisão determinística ser tomada.

Nosso objetivo neste primeiro capítulo é classificar os problemas de otimização sob incerteza em duas distintas categorias: otimização flexível e otimização sob generalizada incerteza. Uma abordagem de otimização flexível é a otimização *fuzzy*. Uma abordagem de otimização sob incerteza generalizada se dá, neste livro, por meio de intervalos. Aqui, nós apresentamos uma breve introdução à teoria de conjuntos *fuzzy* para ajudar a compreensão da eficácia da otimização flexível e sob incerteza generalizada. As seções seguintes deste primeiro capítulo apresentam as seguintes ideias que são desenvolvidas em mais detalhes nos capítulos subsequentes:

1. Qual é o significado de flexibilidade, incerteza generalizada e o potencial e utilidade destes conceitos em otimização.
2. A linguagem matemática da flexibilidade como teoria de conjuntos *fuzzy* e as linguagens matemáticas da incerteza generalizada por meio da análise intervalar.
3. A taxonomia da otimização flexível e da otimização sob incerteza generalizada que leva à distintas teorias e métodos computacionais. .

Definição 1.1. *Um conjunto fuzzy é definido via sua função de pertinência que, neste livro, é uma função concava e contínua:*

$$\mu : X \subseteq \mathbb{R} \rightarrow [0, 1]. \quad (1.1)$$

Observação 1.1. *A condição da função de pertencimento ser contínua pode ser relaxada. Neste livro esta generalização não é utilizada. Conjuntos fuzzy são conjuntos para os quais o pertencimento é gradual, não booleano (clássico), e esta gradualidade é indicada pelo contra-domínio da função de pertencimento $[0, 1]$. Um conjunto clássico tem como possíveis valores $\{0, 1\}$, ou seja, ou um elemento pertence ou não pertence a um conjunto.*

Observação 1.2. *Uma das preocupações que alguns pesquisadores expressam com respeito aos conjuntos fuzzy é o seu significado matemático, ou seja, a semântica matemática de conjuntos fuzzy. A flexibilidade e, muitas vezes, a maneira ad hoc que conjuntos fuzzy são apresentados por meio de sua função de pertinência - uma função de pertinência que pode ser qualquer coisa que o usuário deseje. A preocupação destes pesquisadores é que esta seja puramente uma entidade heurística e, portanto, fora da matemática clássica. Neste livro um **conjunto fuzzy**, definido*

por sua função de pertinência (1.1), é uma generalização de conjunto (clássico). Ao invés de um conjunto ser definido via uma função de pertencimento booleana por meio da função característica:

$$\mu_A(x) \equiv \chi_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases}, \quad (1.2)$$

um conjunto fuzzy é dado por um pertencimento gradual definido por (1.1).

Desta forma, conjuntos fuzzy são uma generalização de conjuntos clássicos. Alguém pode fazer o mesmo nível de crítica pelo fato dos conjuntos clássicos serem definidos como “uma coleção de objetos” como sendo “ad hoc” e “heurística” pois um conjunto pode ser qualquer coisa que o usuário deseja que seja. Entretanto, esta definição é considerada bem definida e a base de boa parte da matemática. Assim, vale reiterar, um conjunto fuzzy ou fuzzy é uma generalização da teoria dos conjuntos. Em aplicações de conjuntos fuzzy em problemas reais, a forma como a gradualidade é escolhida e implementada pode ser heurística. No entanto, como um objeto matemático, um conjunto fuzzy tem sua semântica bem definida no sentido da teoria generalizada dos conjuntos.

Observação 1.3. Uma palavra sobre a nossa notação de entidades fuzzy e sob incerteza generalizada. A notação para uma entidade fuzzy é um til, \sim , sobre ele, e para entidade sob incerteza generalizada é um circunflexo, $\hat{\sim}$, sobre ele. Um conjunto fuzzy genérico é denotado aqui por meio da sua função de pertinência $\mu(x)$. Muitas vezes, escrevemos um subscrito, $\mu_A(x)$, para indicar que a função de pertinência fuzzy é de um conjunto específico, A . Quando não há designação para se um conjunto é um conjunto fuzzy ou uma entidade de incerteza, o contexto determinará sua semântica.

A flexibilidade, como a usaremos aqui, tem como foco o conjunto de restrições pertencimento. Do nosso ponto de vista, tudo o que chamamos de otimização flexível surge do relaxamento do pertencimento ao conjunto de restrições. Assim, otimização flexível como desenvolvido aqui inclui o que tem sido tradicionalmente chamada de otimização fuzzy.

Otimização flexível e otimização sob incerteza generalizada são abordagens relativamente mais recentes para otimização desenvolvidas ao longo das últimas quatro décadas. Eles são dignos de séria consideração na medida em que na própria natureza e na sociedade muitos modelos de otimização têm flexibilidade e generalizada incerteza. Os primeiros pesquisadores a distinguir *fuzzy* (flexível) de otimização possibilística (um tipo de incerteza generalizada) e usar o termo “otimização flexível” em uma publicação foram [1], que usam este termo para descrever “problemas de otimização com imprecisão”. No entanto, o primeiro pesquisador, até onde sabemos, a utilizar o termo “flexibilidade” em relação foi Hans J. Zimmermann (comunicação pessoal com o professor M. Inuiguchi).

A incerteza generalizada, como o termo é usado neste texto, limita-se a incerteza que surge de informações parciais. Começamos formalizando o que entendemos por flexibilidade e incerteza generalizada.

Definição 1.2. *Flexibilidade* no contexto de um problema de otimização é o relaxamento do significado de relações estabelecidas como pertencimento (ao conjunto de restrições) ou otimização (no objetivo). Ou seja, as relações “pertence a”, \in , $=$, \leq , \min , ou \max podem assumir um significado gradual ao invés de um booleano ou absoluto.

Quando o pertencimento assume o significado de “o mais próximo possível, mas não exceda” então há flexibilidade. Por exemplo, pode-se ter uma restrição determinística de um limite superior das horas de trabalho com base no número de funcionários de uma empresa, *pool* de horas extras e/ou trabalhadores temporários que a empresa pode recorrer.

Os conjuntos *fuzzy* têm uma semântica de pertencimento gradual, não-booleano, a conjuntos, de modo que é uma linguagem matemática natural para problemas em que a flexibilidade é uma parte inerente da modelagem e análise. A flexibilidade inclui conjuntos *fuzzy*, mas no sentido em que estamos usando a palavra, inclui coisas como “chegue o mais próximo possível, mas não exceda”.

Definição 1.3. *Teoria da Incerteza generalizada é uma teoria matemática de incompletude ou falta de informação, falta de especificidade ou imprecisão.*

A definição acima de incerteza generalizada usou a palavra “incerteza” e “incompletude da informação” em sua definição. Essas palavras para este livro terão significados restritos que descrevemos a seguir.

Definição 1.4. [2] *Incerteza é o estado de não conhecer o exato valor de uma entidade ou a verdade de uma proposição/declaração. Ou seja, uma parte das informações ou dados são incertos para um agente quando o agente não sabe se a informação é definitivamente verdadeira ou falsa.*

Exemplo 1.2. *Em um programa de computador, suponha que temos as seguintes instruções: SE $x > 0$ ENTÃO $y = 3$, SE $x < 0$, ENTÃO $y = -3$, CASO CONTRÁRIO $y = 0$. Quando o computador ramifica e retorna $y = 0$, sabemos que foi causado por um valor real 0? Como o computador está sujeito a erros de arredondamento, há uma incerteza associada à ramificação.*

Definição 1.5. [2] *Diz-se que uma informação ou dado é incompleto (ou impreciso, não completamente especificado, falta informação) em um determinado contexto se não for suficiente para permitir que o agente responda a uma pergunta relevante definitivamente neste contexto.*

Exemplo 1.3. *Seja $A \equiv \text{Senior Idade} = \{65, \dots, 90, \dots, 120\}$. Quando estamos perguntando a idade de uma pessoa X e tudo o que sabemos é o fato de que $X \in A$, temos uma informação incompleta, pois não podemos responder a pergunta sobre a idade da pessoa X . No entanto, se a pergunta for, “Esta pessoa X tem idade maior que 59?” então a informação $X \in A$ não está incompleta.*

Neste livro iremos tratar a incerteza generalizada com intervalos [4]. Para um estudo mais abrangente veja [4]. Para clarear as ideias vejamos um exemplo.

Exemplo 1.4. (*Dados fuzzy*) *O conjunto de voxels que define um tumor em uma imagem de tomografia computadorizada (CT) é transicional (pelo menos para alguns vértices), pois existem voxels na transição tumor/não-tumor. Ou seja, no conjunto universal de voxels de uma imagem de CT, o conjunto de voxels “tumor” é um conjunto caracterizado por transição não tumoral para tumoral. Alguns voxels podem ser tumorais e não tumorais em um determinado grau simultaneamente. Ou seja, são simultaneamente tumoral e não tumoral.*

Observe que no último exemplo os *elementos* (voxels) são claramente definidos, mas o conjunto é *fuzzy* e gradual. Outra característica que pode ser compreendida desse exemplo é que um conjunto *fuzzy* genuíno como um *tumor*, não é uma questão

de resolução. O limite de tumor e não tumor permanece gradual, independentemente da resolução. A gradualidade de um tumor sempre permanecerá. Isto é, a imprecisão permanece mesmo com mais dados.

- Conjuntos *fuzzy* são representados por uma função de pertinência **única** [5]. Por exemplo, o intervalo $[5, 7]$ é uma codificação de falta de informação e ser expressa como

$$f(x) = \begin{cases} 0 & \text{se } x < 5 \text{ ou } x > 7 \\ 1 & \text{se } 5 \leq x \leq 7 \end{cases} \quad (1.3)$$

- Conjuntos *fuzzy* são conjuntos que modelam o pertencimento gradual ao conjunto. A incerteza é um estado em que a informação não permite a determinação da verdade ou valor como um número real único da entidade em discussão.

Existem conjuntos *fuzzy* tipo 2 que são descritos por um conjunto de funções de pertinência, uma para cada elemento do intervalo (cada elemento do contradomínio é em si um conjunto *fuzzy*). No entanto, este livro não aborda esses tipos de conjuntos *fuzzy*, mas o leitor interessado pode consultar [6].

Problemas de Otimização: Um modelo de otimização de valor real (determinística) é um modelo matemático *normativo* cujo sistema subjacente é na maioria das vezes *restringido* e sua forma geral é:

$$z = \text{opt } f(x, c) \quad (1.4)$$

$$\text{sujeito a } g_i(x, a) \leq b_i \quad i = 1, \dots, M_1, \quad (1.5)$$

$$h_j(x, d) = e_j \quad j = 1, \dots, M_2, \quad (1.6)$$

$$x \in S \subseteq \mathbb{R}^n. \quad (1.7)$$

Dados a, \dots, e e $S \subseteq \mathbb{R}^n$, o conjunto dos elementos $x \in \mathbb{R}^n$ que satisfazem as restrições (1.5), (1.6), (1.7) é definido por

$$\Omega = \{x | g_i(x, a) \leq b_i \quad i = 1, \dots, M_1, h_j(x, d) = e_j \quad j = 1, \dots, M_2, \\ x \in S \subseteq \mathbb{R}^n\}.$$

Supõe-se que $\Omega \neq \emptyset$. Os valores e/ou natureza de $a, b, c, d, e, \leq, =, \in$, e *opt* (min / max) constituem-se na entrada (dados) para o modelo de otimização. Nosso modelo geral pode ser simbolicamente reformulado como

$$z = \min f(x, c) \quad (1.8)$$

$$x \in \Omega(a, b, d, e, S), \quad (1.9)$$

onde denotamos o conjunto de restrições em função dos parâmetros de entrada.

Igualdade é equivalente a “maior ou igual a” e “menor ou igual a” para variáveis reais. Portanto, não há necessidade matemática de listar estes (1.6) separadamente. No entanto, isso não é verdade para intervalos e, portanto, para intervalos *fuzzy*, uma vez que intervalos *fuzzy* são descritos como um conjunto incontável de intervalos chamados α -cuts. Além disso, os intervalos podem ser considerados conjuntos

fuzzy e ao mesmo tempo podem ser considerados como incerteza generalizada, dependendo de sua semântica. Alertamos o leitor para o fato de estarmos lidando com entidades que possuem uma estrutura diferente dos números reais, como o exemplo a seguir ilustra.

Exemplo 1.5. *Suponha que temos a equação intervalar,*

$$[1, 2]x = [2, 4],$$

e queremos resolvê-lo por duas desigualdades, \leq e \geq . Para

$$[1, 2]x \leq [2, 4],$$

a solução, usando a análise intervalar padrão/clássica (voltaremos em detalhe no próximo capítulo), é,

$$x \in (-\infty, 1].$$

Por

$$[1, 2]x \geq [2, 4],$$

a solução, usando análise intervalar é,

$$x \in [4, \infty).$$

Desta forma,

$$[1, 2]x \leq [2, 4] \text{ e } [1, 2]x \geq [2, 4]$$

ou seja

$$x \in (-\infty, 1] \cap x \in [4, \infty) = \emptyset.$$

Por outro lado, é claro que, usando a análise intervalar clássica,

$$[1, 2]x = [2, 4] \rightarrow x = [2, 2].$$

Assim, distinguimos entre igualdade e desigualdade na representação de problemas de programação matemática cujos conjuntos de restrições possuem flexibilidade ou incerteza generalizada.

1.2 Possibilidade e Necessidade

Uma generalização da incerteza é o conceito de possibilidade/necessidade . Seja:

$$Pos(A) : X \subseteq \mathbb{R} \rightarrow [0, 1].$$

Considere

$$g : A \subseteq X \rightarrow [0, 1], A \in \sigma(X), \quad (1.10)$$

e uma estrutura de subconjuntos de X , $\sigma(X)$, uma sigma algebra sobre X . Um conjunto razoável de hipóteses sobre g são as seguintes:

$$g(\emptyset) = 0, \text{ e } g(X) = 1 \quad (1.11)$$

$$A \subseteq B \Rightarrow g(A) \leq g(B). \quad (1.12)$$

Uma consequência de (1.11) (1.12) é:

$$g(A \cap B) \leq \min\{g(A), g(B)\} \quad (1.13)$$

$$g(A \cup B) \geq \max\{g(A), g(B)\}. \quad (1.14)$$

A incerteza generalizada será chamada de medida de possibilidade e medida de necessidade, satisfazendo (1.10), (1.11) and (1.12) onde a possibilidade satisfaz (1.14) na igualdade e a necessidade satisfaz (1.13) na igualdade. Nós também estamos interessados nas distribuições associadas às medidas de possibilidade.

Definição 1.6. *Uma medida de possibilidade, axiomáticamente definida, é uma função Pos sobre σ_X de um conjunto universal $X \subset \mathbb{R}$,*

$$Pos : \sigma_X \rightarrow [0, 1]$$

tal que

$$Pos(\emptyset) = 0, \quad Pos(X) = 1, \quad (1.15)$$

$$Pos(A \cup B) = \max\{Pos(A), Pos(B)\}, \quad A, B \in \sigma(X) \quad (1.16)$$

$$\text{se } A_i \in \sigma_X, \quad \bigcup_{i \in I} A_i \in \sigma_X, \quad \text{então}$$

$$Pos\left(\bigcup_{i \in I} A_i\right) = \sup_{i \in I} Pos(A_i) \quad (1.17)$$

Exemplo 1.6. *Suponha que temos dois candidatos para uma posição. Seja A o conjunto das pessoas que possivelmente votariam no candidato um e B o conjunto das pessoas que possivelmente vote no candidato dois. A possibilidade do resultado eleitoral dos votos de pessoas que votariam no candidato um ou dois $A \cup B$, é dado por,*

$$Pos(A \cup B) = \max\{Pos(A), Pos(B)\}.$$

Exemplo 1.7. Medidas de possibilidade e necessidade: *O departamento de transporte de uma cidade entrevista 100 motoristas de ônibus para saber quantas pessoas cabem em um ônibus durante a hora do rush. Quarenta motoristas dizem até 40 pessoas, trinta dizem até 50 e trinta dizem até 60. Neste caso*

$$U_1 = \{1, \dots, 40\}$$

$$U_2 = U_1 \cup \{41, \dots, 50\}$$

$$U_3 = U_2 \cup \{51, \dots, 60\}$$

tal que $U_1 \subset U_2 \subset U_3$. Portanto, nós podemos definir as seguintes medidas de necessidade e possibilidade:

$$m(U_1) = 0.40$$

$$m(U_2) = 0.30$$

$$m(U_3) = 0.30$$

$$Pos(\{1\}) = Pos(\{2\}) = \dots = Pos(\{40\}) = 1.0$$

$$Pos(\{41\}) = Pos(\{42\}) = \dots = Pos(\{50\}) = 0.6$$

$$Pos(\{51\}) = Pos(\{52\}) = \dots = Pos(\{60\}) = 0.3$$

Para a necessidade nós temos,

$$Nec(U_1) = 1 - Pos(U_1^C) = 0.4$$

$$Nec(U_2) = 1 - Pos(U_2^C) = 0.7$$

$$Nec(U_3) = 1 - Pos(U_3^C) = 1.0$$

Definição 1.7. *Dada uma medida de possibilidade, Pos, a distribuição de possibilidade associada Pos é a função*

$$pos(x) : X \rightarrow [0, 1]$$

tal que

$$pos(x) = Pos(\{x\}). \quad (1.18)$$

No capítulo 3 veremos uma aplicação da medida de necessidade no clássico problema do fazendeiro.

1.3 Otimização sob Incerteza Generalizada

Defendemos a otimização sob incerteza generalizada como um estudo diferente do estudo da otimização flexível (*fuzzy*). Este texto considera os seguintes três princípios cruciais no processo de modelagem de otimização:

1. Modelos de otimização de sistemas reais são muitas vezes *satisfatórios* (veja [7], [8]). Satisfatório conforme definido por Herbert Simon (ver [7], [8]), significa que os tomadores de decisão raramente trabalham deterministicamente com a “melhor” solução ou são mesmo capazes de obter “a melhor” solução para um problema real, mas procuram obter soluções que sejam satisfatórias. Soluções que são satisfatórias são inerentemente flexíveis. Otimização flexível e generalizada são capazes de modelar o *satisfatório* de forma natural e direta. São linguagens matemáticas que podem e codificam problemas satisfatórios.
2. Muitos modelos de otimização são *epistêmicos*. Modelos que são epistêmicos são aqueles que nós, como humanos, construímos a partir do conhecimento sobre um sistema, ao invés de modelos que são construídos a partir do próprio sistema. Por exemplo, um piloto automático de um avião modela a física do sistema. Um chip de lógica fuzzy que controla uma panela de arroz é um modelo do que sabemos sobre cozinhar arroz ao invés da física do cozimento do arroz.
3. Muitas vezes os dados de entrada que compõem o modelo são parâmetros com informações incompletas. Alguns dados, como projeções (por exemplo, o demandas futuras de um produto ou o que acontecerá com os suprimentos em meio a uma pandemia) são inerentemente incompletos e incertos.

Modelos de otimização determinísticos e estocásticos requerem entradas bem definidas de parâmetros (coeficientes, valores do lado direito), relacionamentos (desigualdades, igualdades) e preferências (funções de valor real para maximizar, minimizar). Qualquer modelo em grande escala deste tipo requer esforço de coleta de valores precisos. Se o modelo tiver projeções de valores futuros, é claro que números reais serão representações inadequadas, mesmo assumindo que o modelo captura corretamente o sistema subjacente. Também é conhecido da teoria de programação matemática que apenas algumas das variáveis e restrições são necessárias para descrever uma solução ótima (variáveis básicas e restrições ativas). Assim, apenas poucos parâmetros precisam ser obtidos com precisão, aqueles que estão na base final. Claro que não se conhece *a-priori* quais variáveis serão básicas e quais restrições são ativas.

Herbert Simon (veja [7], [8]) afirma:

“Claro que a decisão que é ótima no modelo simplificado raramente será ótimo no mundo. O tomador de decisão tem uma escolha entre decisões ótimas para um mundo imaginário simplificado ou decisões que são “boas o suficiente”. O que uma pessoa não pode fazer, ela não fará, não importa o quão ele quer fazer isso. A economia normativa mostrou que soluções exatas para a maioria dos problemas de otimização do mundo simplesmente não estão ao alcance. Diante dessa complexidade, uma empresa do mundo real se volta para procedimentos que encontram respostas suficientemente boas para perguntas cujas melhores respostas são inacessíveis. Assim, a microeconomia normativa, ao mostrar a otimização do mundo real ser impossível, demonstra que o ser humano econômico é de fato uma pessoa que aceita alternativas “suficientemente boas”, não porque prefira menos, mas porque ela não tem escolha.”

Esta observação é claramente refletida por Fandel, [9]. Em uma discussão com o Professor Rommelfanger [10] ele relata o seguinte.

“Na verdade, Herbert Simon desenvolve uma abordagem de tomada de decisão, que ele chama de *o conceito de racionalidade limitada*. Ele a formulou na sequência de duas teses. *Tese 1*: Em geral um ser humano não luta por decisões ótimas, mas tende a escolher um curso de ação que atenda aos padrões mínimos de satisfação. *Tese 2*: O curso de ações e consequências alternativas são, em geral, desconhecidas a priori, mas eles devem ser encontrados por meio de um procedimento de busca.”

Um modelo de um problema real é sempre uma visão abreviada do sistema real subjacente. Se um problema pudesse ser manipulado em sua totalidade complexa *in situ* para obter uma solução em uma representação, então não haveria necessidade de modelar o problema matematicamente em primeiro lugar. Intrinsecamente, um modelo matemático é uma representação simbólica do problema e não o problema em si. Assim, um modelo está em processo de conhecimento ou descoberta que chamamos de *processo epistemológico*.

A matemática é a ciência da precisão. O coração da análise matemática é a ordem - a ordem é mais fundamental que a medida (L. Collatz, palestra Oregon State University, palestra do Departamento de Matemática 1976). Medida (quantidade e suas generalizações) e extensão (distância, área, integração) podem ser derivadas da ordem. A otimização requer uma ordem, um critério normativo; a medida que impomos é a função objetivo. E é a função objetivo que nos diz o que é “melhor.” A ordem mais ideal é encontrada no conjunto de números reais (a reta numérica real). Na otimização, muitas vezes mapeamos, através do função objetivo, sobre os números reais. Na otimização flexível, mapeamos conjuntos *fuzzy* (conjuntos de funções de pertinência) no conjunto de números reais usando operadores de agregação. A otimização estocástica geralmente usa o valor esperado para mapear distribuições na reta numérica real. Na otimização sob incerteza generalizada, mapeamos intervalos, por exemplo, em números reais.

Uma abordagem útil para flexibilidade e otimização sob incerteza generalizada também deve aderir ao *Princípio do Mínimo Compromisso* que afirma que, “Apenas decida quando você precisar .”

Um dos autores ouviu pela primeira vez o termo Princípio do Mínimo Compromisso, usado pelo Dr. J. Keller em uma conferência NAFIPS de 17 a 20 de setembro de 1995, College Park, MD. No contexto de otimização, o princípio do

mínimo compromisso é usar entidades determinísticas de valor real somente quando uma decisão determinística deve ser tomada, o mais próximo possível da última etapa do processo de modelagem. Esta é a abordagem que este texto usa - carregar toda a extensão da incerteza e/ou gradualidade do início até que se deva escolher deterministicamente o mais próximo possível do último passo.

Resumimos o que foi articulado citando (e acrescentando algumas observações) Dubois e Prade [11].

“Otimização fuzzy, que chamamos aqui *otimização flexível*, oferece uma ponte entre abordagens numéricas e determinísticas e as linguísticas ou qualitativas. O objetivo dessas abordagens é fornecer ao tomador de decisão qual é a incerteza nos resultados.

Teoria dos conjuntos *fuzzy* e seu ambiente matemático de operadores de agregação (“*e*”, *t-norms*), análise intervalar [12], números graduais [13], e modelagem de preferência, fornecem uma visão geral da estrutura para colocar problemas de decisão de uma forma mais aberta, e fornece uma unificação de técnicas e teorias existentes.

A teoria dos conjuntos *fuzzy* nos dá a capacidade de traduzir variáveis linguísticas em termos quantitativos em uma forma flexível e útil.”

Em resumo, a teoria dos conjuntos *fuzzy* e a teoria da incerteza generalizada oferecem duas abordagens que estão mais próximas da flexibilidade subjacente, processos satisfatórios e de falta de informação que muitas vezes são os ambiente típicos em que os tomadores de decisão se encontram.

Teoria dos Conjuntos *fuzzy* e Teoria da Possibilidade - Uma Breve Nota:

A Teoria de Possibilidade surgiu da teoria dos conjuntos *fuzzy* e, de fato, o título do primeiro artigo sobre a teoria da possibilidade foi, em livre tradução, “Conjuntos *fuzzy* como base para uma teoria de possibilidade,” [16]. No entanto, desde desde o início, mostrou-se diferente da teoria dos conjuntos *fuzzy*. Então, o que distingue conjuntos *fuzzy* e possibilidades? Primeiro, a teoria dos conjuntos *fuzzy* é associada à teoria generalizada dos conjuntos. A teoria da possibilidade está associada com a teoria generalizada da probabilidade. Segundo, a semântica de conjuntos *fuzzy* e possibilidades são distintas. Conjuntos *fuzzy* estão associados a conjuntos graduais de pertencimento e, portanto, sua semântica é a de conjuntos generalizados. As possibilidades são associadas às deficiências de informação e, portanto, sua semântica está ligada a probabilidade generalizada. Terceiro, quando visto desde os primeiros princípios (axiomas), as medidas que eles induzem (medidas fuzzy, medidas de possibilidade) são distintas (veja [14]). Ou seja, os fundamentos subjacentes da teoria dos conjuntos *fuzzy* e da teoria da possibilidade são distintas.

A confusão entre a teoria dos conjuntos *fuzzy* e a teoria da possibilidade surge mais especialmente em otimização, uma vez que tanto a otimização fuzzy (flexível otimização) e otimização possibilísticas (otimização sob incerteza) usam intervalos ou intervalos *fuzzy* (números *fuzzy*) que dão a impressão de que não há distinção entre conjuntos *fuzzy* e possibilidades. Em particular, intervalos *fuzzy* (números *fuzzy*) podem codificar matematicamente conjuntos com pertencimento gradual e intervalos *fuzzy* também podem codificar deficiência de informação, valores parcialmente conhecidos e/ou não especificidade. Para garantir que as distinções sejam mantidas, é preciso entender a semântica dos parâmetros subjacentes e relacionamentos, ou seja, o modelo.

Os dados dos quais surgem os parâmetros e relacionamentos ditarão sua semântica. Se o modelo codifica o pertencimento relaxado ao conjunto, ele representa flexibilidade e sua linguagem é a teoria dos conjuntos *fuzzy*. Se o modelo surgir de dados cujos parâmetros são parcialmente conhecidos, ambíguos ou incompletamente especificados, então é um tipo de incerteza generalizada. A teoria da possibilidade é uma linguagem matemática que pode ser usada para representar matematicamente esse tipo de incerteza.

1.3.1 Intervalos

Um intervalo é um conjunto conexo compacto de números reais que denotamos por $[x]$ onde $[x] = [\underline{x}, \bar{x}] = \{x \mid \underline{x} \leq x \leq \bar{x}\}$. Temos

$$-\infty < \underline{x} \leq \bar{x} < \infty$$

Intervalos são a chave para as operações matemáticas sobre números *fuzzy*, intervalos *fuzzy* e alguns tipos de incerteza generalizada em que o conjuntos de níveis de intervalos *fuzzy* e os conjuntos de níveis de possibilidade superior/inferior dos limites das distribuições são intervalos.

Um intervalo $[a, b]$ pode ser considerado como o seguinte conjunto *fuzzy* cuja função de pertinência é

$$\mu_{[a,b]}(x) = \begin{cases} 1 & x \in [a, b] \\ 0 & \text{caso contrário} \end{cases} \quad (1.19)$$

Um intervalo também pode codificar deficiência de informação e é assim, neste contexto, possibilístico, um intervalo possibilístico. Dado um intervalo $[a, b]$, do ponto de vista da representação baseada em teoria da possibilidade, se tudo o que sabemos é que o suporte de uma probabilidade é $[a, b]$, podemos construir as seguintes distribuições (cumulativos):

$$pos(x) = \begin{cases} 0 & x \in (-\infty, a) \\ 1 & x \in [a, \infty) \end{cases}, \quad (1.20)$$

$$nec(x) = \begin{cases} 0 & x \in (-\infty, b) \\ 1 & x \in [b, \infty) \end{cases}. \quad (1.21)$$

Suponha $[a, b] = [1, 4]$. A primeira função, $pos(x)$, (1.20), é uma distribuição cumulativa de possibilidade superior e a segunda função, $nec(x)$, (1.21), é uma distribuição inferior que é dual para $pos(x)$. Do ponto de vista de um intervalo que representa a possibilidade baseada em probabilidade associada a incompletude da informação, no contexto do intervalo $[1, 4]$, qualquer distribuição cujo suporte esteja em $[1, 4]$ é uma instanciação válida das informações incompletas fornecidas e codificada por $\mu_{[a,b]}$. Ou seja, o intervalo *fuzzy* $[a, b]$, que é o intervalo *fuzzy* $[1, 4]$ neste exemplo, pode codificar infinitas famílias de distribuições cumulativas. Uma dessas é aquela que corresponde à distribuição de probabilidade uniforme, dado por:

$$\psi_{unif}(x) = \begin{cases} 0 & \text{para } x < 1 \\ \frac{1}{3}x - \frac{1}{3} & \text{para } 1 \leq x \leq 4 \\ 1 & \text{para } x > 4 \end{cases}$$

Veremos que essas diferentes representações dependem do contexto do problema, da natureza da incerteza. Intervalos como codificação de informações parciais constituem-se em uma dessas representações de incerteza (generalizada).

1.3.2 Intervalos *fuzzy*

O que geralmente é chamado de número *fuzzy* é um caso particular de um número mais geral, um *intervalo fuzzy*. O que é chamado de número *fuzzy* x^* , é um conjunto *fuzzy* $\mu_{x^*}(x)$ para o qual $\mu_{x^*}(x^*) = 1$ exclusivamente. Um intervalo *fuzzy* é o mesmo, exceto que em vez de um único x^* para o qual $\mu_{x^*}(x^*) = 1$, temos uma associação $\mu(x) = 1$ que pode ser um intervalo $\underline{x} \leq x \leq \bar{x}$. Um intervalo *fuzzy* pode ser traduzido em um par de distribuições de possibilidades e, portanto, pode ser um modelo tanto para a falta de informação, como para a gradualidade. Em problemas de otimização, intervalos *fuzzy* podem modelar tanto a gradualidade (flexibilidade) quanto a incerteza (falta de especificidade). Assim, alguns conjuntos *fuzzy* (intervalos *fuzzy*, por exemplo) *podem* ter ou assumir uma dupla natureza - a de capturar a gradualidade (flexibilidade) de pertencimento e capturar a falta de informação (incerteza). Quando a mesma entidade tem múltiplas representações, como para intervalos *fuzzy*, distinções semânticas claras são essenciais/necessárias. Formalizamos o que queremos dizer por número *fuzzy* e intervalo *fuzzy*.

Definição 1.8. *Um número fuzzy x^* é um conjunto fuzzy sobre o domínio dos números reais cuja função de pertinência é contínua com um e apenas um valor, x^* , tal que $\mu(x^*) = 1$ (x^* é o número que é “fuzzy”), tal que se $x \leq x^*$ então $\mu(x)$ é não-decrescente e se $x \geq x^*$ então $\mu(x)$ é não-crescente, onde seu suporte é um intervalo contendo x^* .*

Definição 1.9. *O conjunto de elementos do domínio (números reais) para os quais os valores de associação são iguais a um é chamado de *core*.*

Por exemplo, um intervalo *fuzzy* $\tilde{2}$ tem exclusivamente $\mu(2) = 1$. Um número *fuzzy*, digamos $\tilde{2}$, muitas vezes surge no contexto de obter um valor para um parâmetro baseado em conhecimento epistêmico como em, “o valor é em torno de 2.”

“A teoria dos conjuntos *fuzzy* e suas extensões parecem se encaixar com três conceitos básicos que diferem visivelmente um do outro, mas podem aparecer conjuntamente em várias circunstâncias:

1. *Gradualidade*: A ideia de que muitas categorias em linguagem natural são uma questão de grau, incluindo a verdade.
2. *Incerteza epistêmica*: A ideia de informação incompleta, na sua forma mais primitiva, é muitas vezes descrita por meio de um conjunto de valores possíveis de alguma quantidade de interesse. Essa visão leva à teoria da possibilidade [16].
3. *Bipolaridade*: A ideia de que a informação pode ser descrita por distinguir entre lados positivos e negativos, possivelmente tratados separadamente.”

Este texto não trata da bipolaridade como entidade, apenas gradualidade (conjuntos *fuzzy*) e incerteza epistêmica (possibilidade), a partir da qual vêm os conceitos de conjunção e disjunção.

1.3.3 Intervalos de Possibilidade

A teoria da possibilidade foi proposta pela primeira vez por L. Zadeh [16] e mais extensivamente articulado em D. Dubois e H. Prade [18]. Necessidade foi desenvolvido pela primeira vez por D. Dubois e H. Prade [18]. Se conhecemos a possibilidade

de um conjunto A , a possibilidade do complemento de A , $Pos(A^C)$ não é definido como $1 - Pos(A)$, em contradição com a probabilidade. Precisamos definir um conceito que seja o “dual” de possibilidade chamada necessidade onde

$$Nec(A) = 1 - Pos(A^C),$$

isso é,

$$Pos(A^C) = 1 - Nec(A).$$

Concretamente, em um modelo de otimização que busca matar tumores cancerígenos poupando a longo prazo todos os outros órgãos e tecidos de um corpo humano, se temos uma restrição que é flexível - (a) “em um pixel tumoral entregar menos de 61 unidades de radiação, mas sob nenhuma circunstância entregar mais de 62 unidades” e (b) “em um pixel tumoral entregar mais de 59 unidades de radiação, mas nunca menos de 58 unidades” - temos uma restrição fuzzy ou flexível. Se nós desejarmos entregar a dose tumorcida mínima representada pelo intervalo *fuzzy* que foi construído a partir de pesquisa e experiência em oncologia de radiação e é o estado de conhecimento que temos sobre “dose tumorcida mínima”, então temos uma possibilidade. Enquanto o conjunto *fuzzy* ou flexível e distribuição possibilística são representadas pela mesma função, essas duas representações levam a métodos de otimização distintos, como descrevemos abaixo. O primeiro (*fuzzy*/flexível) usa um operador de agregação e o segundo (possibilística) usa uma abordagem estocástica semelhante a um recurso.

Uma razão pela qual alguém pode querer usar a possibilidade em vez da probabilidade é precisamente em situações para as quais valores reais ou funções de densidade de probabilidade não estão disponíveis. Por exemplo: (1) não sabemos qual dos conjuntos de probabilidades usar, ou (2) tudo o que sabemos é que a probabilidade é limitada por duas funções, ou (3) não temos a distribuição de probabilidade em todo conjunto, mas em alguns subconjuntos, dos quais podem haver interseções não vazias (sobreposição).

1.4 Uma Taxonomia de Incerteza Generalizada em Otimização

Esta exposição considera problemas de otimização como (1.4), (1.5) e (1.6) na presença de dados $\{a, b, c, d, e\}$ que são todos ou uma mistura de parâmetros números reais, intervalos, possibilidade/necessidade, isto é, incerteza generalizada, com pelo menos um parâmetro sendo um desses tipos ou relacionamentos $\{\leq, =, \in\}$ que são flexíveis. Em seguida, desenvolvemos um taxonomia, uma vez que relacionamos a taxonomia aos métodos de solução nos capítulos que seguem. Discutimos com mais detalhes os itens apresentados a seguir. Otimização na presença de coeficientes de incerteza generalizada, e flexibilidade podem ser considerados como segue:

1. Otimização Flexível

- (a) **Relações de restrições suaves:** As relações matemáticas $\leq, =, \in$ ou \in como “restrições suaves” assumem um significado flexível. Por exemplo, nós pode ter “menor ou igual a” onde seu significado é: “Desejo ser menor que um certo valor, mas eu posso tolerar ir além do valor um pouco além do que é desejado, mas certamente não mais do que este ‘pouco’.” A

extensão da flexibilidade precisa ser especificada. Outro tipo de flexibilidade é uma especificação de ser maior do que uma dada probabilidade, possibilidade ou necessidade de violação de restrição (veja [19]).

- (b) **A função objetivo expressa um conjunto aceitável de metas:** Uma decisão pode ser atingir uma quantidade ideal de lucro que só é alcançado sob uma rara confluência de situações. Uma função-objetivo flexível seria chegar o mais próximo possível do ideal ou exceder esse ideal. Estes modelos incluem o Modelo original de Bellman e Zadeh [21] e modelo de Zimmermann [22]. A meta é declarada como obtenção ou superação, no caso de maximização, ou ficando abaixo, no caso de um mínimo. Essas metas são adicionadas ao conjunto de desigualdades de tratados como um restrições suaves.
- (c) **O valor do lado direito de uma restrição é um intervalo *fuzzy* que é semanticamente um alvo:** O objetivo de uma restrição pode ser, “Não aplique menos de 58 unidades de radiação ao tumor, preferencialmente entre 59 unidades e 61 unidades, mas nunca exceda 62 unidades.” Quando o intervalo *fuzzy* é semanticamente um alvo, é uma restrição flexível resultando em um problema de otimização flexível.

2. **Otimização sob incerteza generalizada** - Com um único tipo de incerteza generalizada na definição do conjunto de restrições, temos o seguinte.

- (a) **Distribuições nos Parâmetros do Conjunto de Restrições:** Consideramos o caso de quando um intervalo *fuzzy* é usado para capturar a incerteza de um conjunto de parâmetros. Suponha que os dados sobre a exata dose mínima de radiação que matará uma célula cancerosa vem de uma informação que é parcial ou não específica. Todos os especialistas, resultados experimentais, dados experimentais e evidências indicam que menos de 58 unidades certamente não matarão a célula cancerosa e o mínimo radiação certamente não deve ser superior a 62 unidades. A maioria dos especialistas e evidências indicam que o melhor intervalo é entre 59 e 61. Um modelo para essa informação incompleta, é o intervalo trapezoidal *fuzzy* 58/59/61/62. No entanto, esta é uma distribuição possibilística e semanticamente distinta da meta do lado direito indicada na classificação 1(c) acima. Enquanto o trapézio é o mesmo e a localização dentro do conjuntos de restrições (lado direito) é o mesmo, como lidamos com a flexibilidade e incerteza generalizada é distinta.
- (b) **Distribuições em Coeficientes de Função Objetivo:** Suponha que o os parâmetros da função objetivo \vec{c} são semanticamente ligadas às informações deficientes, por exemplo, o custo do tempo de viagem é incerto em um problema de otimização que minimiza os custos de combustível. Este caso é diferente do caso em que a função objetivo é considerada uma meta e traduzida em uma restrição suave. Para coeficientes de custo de incerteza generalizada, existe uma família infinita não-enumerável de funções objetivo possíveis. No entanto, esta família é limitada acima e abaixo, ou seja, a família de funções objetivo estão contidas em um envelope, então temos um conjunto multi-objetivo de funções limitadas acima e abaixo. Dados os limites, minimizar o máximo arrendimento pode ser obtido, como explicaremos nos capítulos seguintes.

1.5 Resumo do Capítulo

Este capítulo definiu e delineou os conceitos da otimização flexível e da otimização sob incerteza generalizada. Examinou as diferenças entre estes conceitos e indicou como a semântica é importante para distinguir qual das várias abordagens deve ser usada. A taxonomia dos tipos de otimização que estão vinculadas à otimização flexível e sob incerteza generalizada foi indicada. Otimização flexível e sob incerteza generalizada são relevantes nas aplicações.

1.6 Exercícios do Capítulo

1. Como você classificaria os seguintes problemas de otimização? Justifique sua resposta.
 1. Problema de Planejamento da Produção baseada na demanda futura dos itens a serem produzidos.
 2. Planejamento da produção de uma fazenda baseada na previsão climática para os próximos seis meses.
 3. Previsão da distribuição de guinchos de uma seguradora na cidade de São Paulo considerando que cada cliente que aciona o serviço deve ser atendido em até 30 minutos.
 4. Dimensionamento de salas de cirurgia em uma hospital da cidade de São José dos Campos que atende vítimas de acidentes de trânsito na Rodovia Presidente Dutra e na própria cidade.

Capítulo 2

Análise Intervalar

2.1 Introdução

O problema de encontrar a estrutura tridimensional de uma proteína, ou seja, as posições de todos os seus átomos em \mathbb{R}^3 , tem uma grande importância para a biologia, química e medicina, uma vez que a estrutura 3D está relacionada com a função da proteína. Experimentalmente, o problema de determinar a estrutura 3D das proteínas pode ser abordada através de cristalografia de raios X ou espectroscopia de ressonância magnética nuclear (RMN) [57].

A espectroscopia de RMN pode ser usada quando uma proteína não pode ser cristalizada [61], fornecendo medição de distâncias entre átomos que estão próximos o suficiente. O cálculo da estrutura 3D da proteína usando estas distâncias é uma das aplicações mais importantes do *Problema de Geometria de Proteínas* em \mathbb{R}^3 .

Algumas destas distâncias fornecidas pela RMN são intervalares, ou seja, não há uma precisão em algumas medidas. Podemos ter uma informação como *a distância entre os átomos 3 e 7 pertence ao intervalo* [1.5, 1.8]. Voltaremos a este exemplo no capítulo 3.

Este é um caso real que evidencia a importância de sabermos operar ou calcular com intervalos. O que é uma soma de intervalos? O que é uma multiplicação de intervalos? Como resolver um sistema linear intervalar? Ou antes, o que seria uma solução de um sistema linear intervalar?

Neste capítulo apresentamos duas abordagens: Aritmética Intervalar Padrão, doravante denominada SIA, do inglês *Standard Interval Arithmetic* na seção 2; Aritmética Intervalar Restrita, doravante denominada CIA, do inglês *Constrained Interval Arithmetic*.

2.2 Aritmética Intervalar Padrão - SIA

A aritmética intervalar foi definida por R. C. Young [23] em 1931, M. Warmus [24] em 1956, e depois de forma independente por T. Sunaga [25] em 1958. Moore [26, 27] redescobre e estende a aritmética intervalar para aritmética intervalar com arredondamento, permitindo assim que a aritmética intervalar seja útil em matemática computacional. Além disso, Moore desenvolveu análise intervalar [28].

Seja $[a, b]$ um intervalo, com $a, b \in \mathbb{R}$, $a < b$. Considere as seguintes operações entre intervalos:

1. $[a, b] + [c, d] = [a + c, b + d]$.

2. $[a, b] - [c, d] = [a - d, b - c]$.
3. $[a, b] \times [c, d] = [\min\{ac, ad, bc, bd\}, \max\{ac, ad, bc, bd\}]$.
4. Se $0 \notin [c, d]$, $[a, b] \div [c, d] = [\min\{\frac{a}{c}, \frac{a}{d}, \frac{b}{c}, \frac{b}{d}\}, \max\{\frac{a}{c}, \frac{a}{d}, \frac{b}{c}, \frac{b}{d}\}]$.

Definição 2.1. *A aritmética intervalar associada às operações 1-4 acima é chamada de Aritmética Intervalar Padrão (SIA).*

Existem várias propriedades associadas ao SIA que são diferentes das dos números reais.

Exemplo 2.1. *Temos: $[1, 2] \times (1 - 1) = [1, 2] \times (0) = [0, 0] = 0$. E $[1, 2] \times (-1) + [1, 2] \times (1) = [1, 2] + [-2, -1] = [-1, 1]$*

2.3 Aritmética Intervalar Restrita - CIA

A grande vantagem do SIA é que é simples de aplicar. Sua complexidade é no máximo quatro vezes maior que a aritmética nos números reais. No entanto, o SIA pode acarretar superestimações como no exemplo a seguir:

Exemplo 2.2. *Seja $f(x) = x(x - 1)$ com $x \in [0, 1]$. Usando o SIA, nós temos:*

$$[0, 1]([0, 1] - 1) = [0, 1]([0 - 1, 1 - 1]) = [0, 1][-1.0] = [-1, 0]$$

Entretanto, o intervalo que corresponde a $f([0, 1])$ é $[-0.25, 0]$,

Para definir uma nova aritmética chamada CIA, definimos o intervalo como sendo uma função real de uma variável e dois parâmetros.

Definição 2.2. *[29] Um intervalo $[\underline{x}, \bar{x}]$ é uma função $f_x : [0, 1] \rightarrow [\underline{x}, \bar{x}]$ tal que*

$$f_x(\lambda_x) = \underline{x} + w_x \lambda_x$$

onde $w_x = \bar{x} - \underline{x}$ é o tamanho do intervalo.

Desta forma em uma vez que os números \underline{x} e \bar{x} (consequentemente w) são conhecidos (entradas ou dados), eles são coeficientes, enquanto λ varia, embora restrito, entre 0 e 1, daí o nome “Aritmética Intervalar Restrita”.

As operações algébricas na “Aritmética Intervalar Restrita” (CIA) são definidas a seguir.

Definição 2.3. *Sejam $[x] = [\underline{x}, \bar{x}]$ e $[y] = [\underline{y}, \bar{y}]$ dois intervalos. Definimos*

$$\begin{aligned} [z] &= [\underline{z}, \bar{z}] = [x] \circ [y] \\ &= \{z | z = x \circ y, \forall x \in f_x(\lambda_x), y \in f_y(\lambda_y), 0 \leq \lambda_x, \lambda_y \leq 1\} \\ &= \{z | z = ((1 - \lambda_x)\underline{x} + \lambda_x\bar{x}) \circ ((1 - \lambda_y)\underline{y} + \lambda_y\bar{y}), 0 \leq \lambda_x \leq 1, 0 \leq \lambda_y \leq 1\} \end{aligned}$$

2.3.1 Exemplificando as diferenças entre o SIA e o CIA

Os autores entendem que o CIA permite abordar melhor a incerteza em alguns modelos matemáticos. Alguns casos são apresentados e explicados no capítulo 4. Além disso, o CIA possui propriedades aritméticas que o SIA não, como detalhado a seguir.

- $[x] - [x]$. Considere o intervalo $[x] = [a, b]$. Usando o SIA nós temos $[x] - [x] = [a - b, b - a]$. Assim, se $a \neq b$, $[x] - [x] \neq 0$. Já usando o CIA nós temos $[x] - [x] = [\min_{0 \leq \lambda_1 \leq 1} (a + \lambda_1 - (a + \lambda_1)), \max_{0 \leq \lambda_1 \leq 1} (a + \lambda_1 - (a + \lambda_1))] = [0, 0]$.
- $[x]/[x]$. Considere o intervalo $[x] = [a, b]$ com $a \neq 0$ e $b \neq 0$. Usando o SIA, nós temos $[x]/[x] = [\min\{\frac{a}{a}, \frac{a}{b}, \frac{b}{a}, \frac{b}{b}\}, \max\{\frac{a}{a}, \frac{a}{b}, \frac{b}{a}, \frac{b}{b}\}]$. Assim, se $a \neq b$ nós temos que $[x]/[x] \neq 1$. Já usando o CIA nós temos que $[x]/[x] = [\min_{0 \leq \lambda_1 \leq 1} (a + \lambda_1)/(a + \lambda_1), \max_{0 \leq \lambda_1 \leq 1} (a + \lambda_1)/(a + \lambda_1)] = [1, 1]$.

Outra diferença entre as abordagens é o fato do SIA ampliar a incerteza nas operações quando comparado com o CIA. Veja o exemplo a seguir.

Exemplo 2.3. Seja $[x] = [1, 2]$ e $y = [-2, 3]$. Vamos calcular $[x][x] + [x][y]$.

1. Usando o SIA temos:

$$[1, 2][1, 2] + [1, 2][-2, 3] = [1, 4] + [-4, 6] = [-3, 10].$$

2. Usando o CIA temos:

$$(1 + \lambda_x)(1 + \lambda_x) + (1 + \lambda_x)(-2 + 5\lambda_y) = (\lambda_x + 1)^2 + 5\lambda_x\lambda_y - 2\lambda_x + 5\lambda_y - 2 = \lambda_x^2 + 5\lambda_x\lambda_y - 1$$

Assim,

$$[1, 2][1, 2] + [1, 2][-2, 3] = [\min_{0 \leq \lambda_x, \lambda_y \leq 1} (\lambda_x^2 + 5\lambda_x\lambda_y - 1), \max_{0 \leq \lambda_x, \lambda_y \leq 1} (\lambda_x^2 + 5\lambda_x\lambda_y - 1)] = [-1, 5]$$

2.4 Sistemas Lineares Intervalares

Sistemas lineares intervalares tem uma longa história, que remonta pelo menos a W. Oettli e W. Prager em 1964 [30] e ao primeiro livro de R. E. Moore em análise intervalar em 1966 [27].

Houve uma variedade de algoritmos que foram propostos ao longo dos anos seguintes, entre eles Hansen [31, 32], Rohn [33, 34, 35, 36], Neumaier [37], [38].

O problema de resolver um sistema linear intervalar pode ser definido como segue. Seja $[b] = [\underline{b}, \bar{b}]$ um vetor intervalar, onde $\underline{b} \leq \bar{b}$, e $[A] = [\underline{A}, \bar{A}]$, $\underline{A} \leq \bar{A}$, uma matriz intervalar $n \times n$. Aqui a ordenação de vetores e matrizes \leq é feita em componentes. Estamos interessados em encontrar as soluções para o sistema intervalar $[A]x = [b]$, que é chamado de *Sistema Linear Intervalar*. Shary em uma série de artigos [40, 41, 42, 43] definiu quatro casos de soluções para sistemas de equações lineares intervalares. Estes são os quatro casos estudados por Lodwick e Dubois [45] do ponto de vista do CIA [46]. Os quatro casos são os seguintes:

- **Caso 1:** x é solução do sistema linear intervalar $[A]x = [b]$ quando para todo $A \in [A]$ existe $b \in [b]$ tal que $Ax = b$, ou seja, $[A]x \subseteq [b]$.
- **Caso 2:** x é solução do sistema linear intervalar $[A]x = [b]$ quando para todo $b \in [b]$ existe $A \in [A]$ tal que $Ax = b$, ou seja, $[A]x \supseteq [b]$.
- **Caso 3:** x é solução do sistema linear intervalar $[A]x = [b]$ quando $[A]x \subseteq [b]$ e $[A]x \supseteq [b]$, ou seja, $[A]x = [b]$ no sentido clássico da teoria dos conjuntos.

- **Caso 4:** x é solução do sistema linear intervalar $[A]x = [b]$ quando existe $A \in [A]$ e existe $b \in [b]$ tal que $Ax = b$, ou seja, $[A]x \cap [b] \neq \emptyset$.

Os quatro casos têm conjuntos de soluções, que podem ser vazios, que denotamos Ω_i , para os casos $i = 1, \dots, 4$. Cada um dos quatro conjuntos de soluções de um sistema linear intervalar possui um *intervalo hull* (envoltório intervalar), a menor caixa intervalar que contem o conjunto de soluções. Os *intervalos hulls*, ou envoltórios intervalares, são denotados como $[\Omega_i], i = 1, \dots, 4$.

Além disso, consideramos válida a propriedade da *inclusão isotônica*. A inclusão isotônica na análise intervalar significa essencialmente que as operações realizadas em intervalos tem a garantia de conter todos os valores possíveis. No contexto de funções temos a seguinte definição, onde Π^k é o espaço de k -tupla $x_1 \times x_2 \times \dots \times x_k$, $x_j = [\underline{x}_j, \bar{x}_j]$ é o intervalo real.

Definição 2.4. (veja [44]) Se $F : \mathbb{V} \subseteq \mathbb{I}^m(\mathbb{R}) \rightarrow \mathbb{I}^n(\mathbb{R})$ é inclusão isotônica e se $x_i \subseteq \hat{x}_i$, então $F(x_1, \dots, x_m) \subseteq F(\hat{x}_1, \dots, \hat{x}_m)$.

Lodwick e Dubois, [45], descrevem o que cada sistema linear intervalar significa no contexto da CIA e apresentam os métodos para encontrar *intervalo hull* para cada caso. Este trabalho apresenta uma abordagem de otimização para cada um dos quatro casos. Percebe-se que [47] desenvolveu um método para calcular o *intervalo hull* para o Caso 1. O que temos como método para Caso 1, é essencialmente a abordagem de Pivkina e Kreinovich.

Usamos um clássico sistema linear intervalar 2×2 [48] para ilustrar o métodos que desenvolvemos:

Exemplo 2.4. Este exemplo foi retirado de [48].

$$\begin{aligned} [2, 4]x_1 + [-2, 1]x_2 &= [-2, 2] \\ [-1, 2]x_1 + [2, 4]x_2 &= [-2, 2] \end{aligned} \quad (2.1)$$

A representação do Caso 4 correspondente ao sistema linear intervalar (2.1) é dado na Figura 2.1 (veja [48])

A partir da ilustração, fica claro que o intervalo hull para (2.1) é $[-4, 4] \times [-4, 4]$.

2.4.1 Quatro Casos

Caso 1:

Este caso, resolvido via programação linear, é bem conhecido (veja [4]).

O caso 1 para Exemplo 2.3 é

$$\begin{aligned} [2, 4]x_1 + [-2, 1]x_2 &\subseteq [-2, 2] \\ [-1, 2]x_1 + [2, 4]x_2 &\subseteq [-2, 2] \end{aligned}$$

Observação 2.1. A delimitação simbólica do problema geral do Caso 1 é

$$\underline{b} \leq [A]x \leq \bar{b}.$$

Nossa abordagem aplicada ao Exemplo 2.3 gera quatro problemas de otimização linear da seguinte forma.

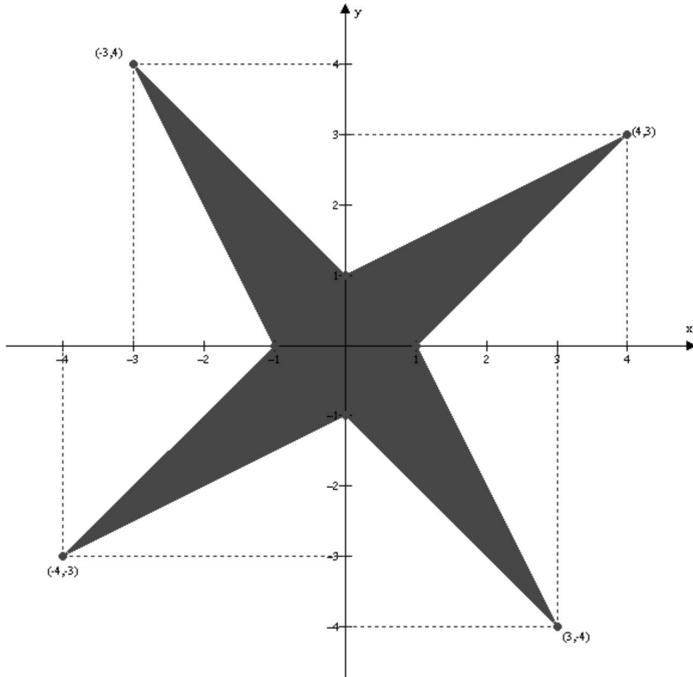


Figura 2.1: Caso 4 Solução Conjunto para Exemplo 2x2

$$\underline{x}_i = \min x_i, i = 1, 2$$

Sujeito a:

$$\begin{aligned} -2 &\leq 2x_1 - 2x_2 \leq 2 \\ -2 &\leq 2x_1 + 1x_2 \leq 2 \\ -2 &\leq 4x_1 - 2x_2 \leq 2 \\ -2 &\leq 4x_1 + 1x_2 \leq 2 \\ -2 &\leq -1x_1 + 2x_2 \leq 2 \\ -2 &\leq -1x_1 + 4x_2 \leq 2 \\ -2 &\leq 2x_1 + 2x_2 \leq 2 \\ -2 &\leq 2x_1 + 4x_2 \leq 2 \end{aligned}$$

$$\bar{x}_i = \max x_i, i = 1, 2$$

Sujeito a:

$$\begin{aligned} 2 &\leq 2x_1 - 2x_2 \leq 2 \\ -2 &\leq 2x_1 + x_2 \leq 2 \\ -2 &\leq 4x_1 - 2x_2 \leq 2 \\ -2 &\leq 4x_1 + 1x_2 \leq 2 \\ -2 &\leq -1x_1 + 2x_2 \leq 2 \\ -2 &\leq -1x_1 + 4x_2 \leq 2 \\ -2 &\leq 2x_1 + 2x_2 \leq 2 \\ -2 &\leq 2x_1 + 4x_2 \leq 2 \end{aligned}$$

O intervalo hull $[\Omega_1] = [\underline{x}_1, \bar{x}_1] \times [\underline{x}_2, \bar{x}_2]$ para este exemplo é $x_1 \in [-0.5, 0.5]$ e $x_2 \in [-0.5, 0.5]$. O problema geral para um sistema linear intervalar $n \times n$ $[A]x = [b]$ requer resolver $2n$ problema de otimização linear, onde cada problema possui $2n^2$ restrições e é dado por

$$m_i = \underline{x}_i = \min x_i$$

Sujeito a:

$$\begin{aligned} \underline{b}_i &\leq \sum_{j=1}^n \text{bound}(a_{ij})x_j \leq \bar{b}_i & i = 1 \dots n \\ \text{bound}(a_{ij}) &= \underline{a}_{ij} \text{ ou } \text{bound}(a_{ij}) = \bar{a}_{ij} & i = 1 \dots n, j = 1, \dots, n. \end{aligned} \quad (2.2)$$

$$M_i = \bar{x}_i = \max x_i$$

Sujeito a:

$$\begin{aligned} \underline{b}_i &\leq \sum_{j=1}^n \text{bound}(a_{ij})x_j \leq \bar{b}_i & i = 1 \dots n \\ \text{bound}(a_{ij}) &= \underline{a}_{ij} \text{ ou } \text{bound}(a_{ij}) = \bar{a}_{ij} & i = 1 \dots n, j = 1, \dots, n. \end{aligned} \quad (2.3)$$

O seguinte teorema é válido.

Teorema 2.1. O conjunto $[\Omega_1] = \prod_{i=1}^n [m_i, M_i]$ é o intervalo hull para o Caso 1.

Demonstração. Observe que os problemas de otimização (2.2) e (2.3) são problemas contínuos sobre conjuntos compactos. \square

Caso 2:

O caso 2 para Exemplo 2.3 é

$$\begin{aligned} [2, 4]x_1 + [-2, 1]x_2 &\supseteq [-2, 2] \\ [-1, 2]x_1 + [2, 4]x_2 &\supseteq [-2, 2] . \end{aligned}$$

Observação 2.2. A delineação simbólica do problema geral do Caso 2 é

$$\min_{A \in [A], x \in \mathbb{R}} Ax \leq \underline{b} \leq \bar{b} \leq \max_{A \in [A], x \in \mathbb{R}} Ax.$$

No caso 2 para Exemplo 2.3, o problema é inviável, ou seja, não existe solução. Dada a estrutura das restrições, parece que a inviabilidade ocorrerá muitas vezes, talvez quase sempre.

Proposição 2.1. Considere o sistema linear intervalar $[A]x = [b]$, com n linhas e n colunas. Se $0 \in [b]$, $[b] \neq [0, 0]$ e não existe uma matriz singular $A \in [A]$, tal que $Ax = 0$, o conjunto solução $\Omega_2 = \emptyset$.

Demonstração. Suponha que exista $x \in \Omega_2$, ou seja, Ω_2 é não vazio. Então, existe $A \in [A]$ e $x \in \Omega_2$ tal que $Ax = 0$. Como $[b] \neq [0, 0]$, existe $b^* \in [b]$, $b^* \neq 0$. Isso implica que $x \neq 0$. Lembre-se que $\min_{A \in [A], x \in \mathbb{R}} Ax \leq \underline{b} \leq \bar{b} \leq \max_{A \in [A], x \in \mathbb{R}} Ax$. Portanto, A é uma matriz singular, o que é uma contradição. Desta forma, $\Omega_2 = \emptyset$ como queríamos demonstrar. \square

Caso 3:

Temos que

$$\Omega_3 = \Omega_1 \cap \Omega_2$$

de modo que

$$[\Omega_3] = [\Omega_1 \cap \Omega_2] \subseteq [\Omega_1] \cap [\Omega_2].$$

Observação 2.3. A delineação simbólica do problema geral do Caso 3 é

$$[A]x \subseteq [b] \text{ e } [b] \subseteq [A]x.$$

O caso 3 para o Exemplo 2.3 é

$$[\Omega_3] = \emptyset.$$

Proposição 2.2. Considere o sistema linear intervalar $[A]x = b$ com n linhas e n colunas. Se $b \neq [0, 0]$ e $[A]$ tem uma ou mais linhas, todas coeficientes são intervalos de largura positivos, então o conjunto de soluções está vazio.

Demonstração. Suponha que exista $x \in \Omega_3$. Seja k^{th} a linha nenhuma de cujos coeficientes são intervalos de largura zero (números reais), ou seja, tal que $[a_{kj}]$ não é um número real para todos os $j = 1, \dots, n$. Como $[b] \neq [0, 0]$, existe $l \in \{1, 2, \dots, n\}$ tal que $x_l \neq 0$. Seja $A^* \in [A]$ tal que $a_{kl}^* = \underline{a}_{kl}$. Como $x \in \Omega_3$ temos $A^*x = b$. Seja $A^{**} \in [A]$ tal que $a_{ij}^{**} = a_{ij}^*$ para todo $i = 1, \dots, n, j = 1, \dots, n$, mas $a_{kl}^{**} = \bar{a}_{kl}$. Portanto, temos $A^{**}x \neq A^*x$, $x_l \neq 0$ e $a_{kl}^{**} \neq a_{kl}^*$. Então, $A^{**}x \neq b$, implica que $x \notin \Omega_3$, o que é uma contradição. \square

2.4.2 Caso 4:

O caso 4 para Exemplo 2.3 é

$$\begin{aligned} &([2, 4]x_1 + [-2, 1]x_2) \cap [-2, 2] \neq \emptyset \\ &([-1, 2]x_1 + [2, 4]x_2) \cap [-2, 2] \neq \emptyset \end{aligned}$$

Observação 2.4. A delimitação simbólica do problema geral do Caso 1 é

$$[A]x \cap [b] \neq \emptyset.$$

O *intervalo hull* para o Caso 4 requer a solução para dos dois problemas de otimização a seguir.

$$\begin{aligned} & m_i = \underline{x}_i = \min x_i \\ \text{Sujeito a:} & \sum_{j=1}^n \alpha_{ij} x_j = \beta_i \quad i = 1 \dots n \\ & \underline{a}_{ij} \leq \alpha_{ij} \leq \bar{a}_{ij} \quad i = 1 \dots n, j = 1 \dots n. \\ & \underline{b}_i \leq \beta_i \leq \bar{b}_{ij} \quad i = 1 \dots n. \end{aligned} \quad (2.4)$$

$$\begin{aligned} & M_i = \bar{x}_i = \max x_i \\ \text{Sujeito a:} & \sum_{j=1}^n \alpha_{ij} x_j = \beta_i \quad i = 1 \dots n \\ & \underline{a}_{ij} \leq \alpha_{ij} \leq \bar{a}_{ij} \quad i = 1 \dots n, j = 1 \dots n. \\ & \underline{b}_i \leq \beta_i \leq \bar{b}_{ij} \quad i = 1 \dots n. \end{aligned} \quad (2.5)$$

O exemplo 2.3 apresentado por [45], necessita da solução dos seguintes quatro problemas de otimização.

$$\begin{aligned} m_i &= \underline{x}_i = \min x_i, i = 1, 2 \\ a_{11}x_1 + a_{12}x_2 &= b_1 \\ a_{21}x_1 + a_{22}x_2 &= b_2 \\ 2 &\leq a_{11} \leq 4 \\ -2 &\leq a_{12} \leq 1 \\ -1 &\leq a_{21} \leq 2 \\ 2 &\leq a_{22} \leq 4 \\ -2 &\leq b_1 \leq 2 \\ -2 &\leq b_2 \leq 2 \end{aligned}$$

$$\begin{aligned} M_i &= \bar{x}_i = \max x_i, i = 1, 2 \\ a_{11}x_1 + a_{12}x_2 &= b_1 \\ a_{21}x_1 + a_{22}x_2 &= b_2 \\ 2 &\leq a_{11} \leq 4 \\ -2 &\leq a_{12} \leq 1 \\ -1 &\leq a_{21} \leq 2 \\ 2 &\leq a_{22} \leq 4 \\ -2 &\leq b_1 \leq 2 \\ -2 &\leq b_2 \leq 2 \end{aligned}$$

O *intervalo hull* para o Caso 4 é $[\Omega_4] = [-4, 4] \times [-4, 4]$, que corresponde ao dado por [45].

O problema geral para um sistema linear intervalar $n \times n$ $[A]x = [b]$ requer $2n$ problema de otimização linear, onde cada problema tem restrições $2n + n^2$ e é dado por Equações(2.4) e (2.5).

O seguinte teorema é válido.

Teorema 2.2. *O conjunto $[\Omega_4] = \prod_{i=1}^n [m_i, M_i]$ é o intervalo hull para o Caso 4.*

Demonstração. Observe que os problemas de otimização, Equações 2.4 e (2.5). são problemas contínuos sobre conjuntos compactos. \square

2.5 O Algoritmo Pivkina/Kreinovich

Pivkina e Kreinovich em 2019 (veja [47]) propuseram um método para calcular a solução para um problema um pouco mais geral do nosso Caso 1. Nosso caso 1 é um subconjunto de seu método *tolerance solution* para sistemas lineares intervalares. Rohn (veja [34, 35, 36]) e Shary (veja [40, 41, 42, 43]) chamam uma solução para o Caso 1 de *solução de tolerância*. Sua abordagem é essencialmente a que apresentamos para o Caso 1. Pivkina e Kreinovich provam que descobrir se o conjunto solução para as Equações 2.2 e 2.3 é polinomial no tempo. Adicionalmente, se o conjunto de soluções é não vazio, encontrar uma solução também é um problema polinomial. Além disso, como pode ser visto das Equações 2.2 e 2.3, o conjunto solução é um poliedro convexo. Quando existe uma solução de tolerância, existem muitos valores no conjunto solução. Eles estão principalmente interessados em encontrar soluções internas. O restante do artigo [47] trata de encontrar “soluções ótimas” dados vários critérios, Nosso interesse é calcular o intervalo hull de forma computacionalmente eficiente.

2.6 Resumo do capítulo

Este capítulo apresentou os conceitos básicos de análise intervalar, tanta a padrão, como a chamada CIA, que será aplicada em problemas reais no capítulo 4. Apresentamos também a definição e alguns resultados sobre sistemas lineares intervalares, que também serão utilizados em aplicações descritas no capítulo 4.

2.7 Exercícios

1. Determine $[x] + [y]$, $[x] - [y]$, $[x][y]$, $[x]/[y]$ e $[x] \cup [y]$, usando CIA e SIA, onde:

a $[x] = [-1, 3]$ e $[y] = [-2, 5]$.

b $[x] = [1, 2]$ e $[y] = [3, 7]$.

c $[x] = [0, 1]$ e $[y] = [-1, 0]$.

2. Considere a função $f(x) = x(x - 1)$. Determine $f([0, 1])$ usando CIA e SIA. Comente os resultados.

3. [Este exercício deve ser resolvido após o estudo das ferramentas computacionais apresentadas no capítulo 5.] Obtenha o hull dos sistemas lineares intervalares abaixo para os casos 1, 2, 3 e 4:

(a)

$$\begin{aligned} [2, 3]x_1 + [0, 1]x_2 &= [0, 120] \\ [1, 2]x_1 + [2, 3]x_2 &= [60, 240] \end{aligned} \tag{2.6}$$

(b)

$$\begin{aligned} [2, 3]x_1 + [1, 2]x_2 + [3, 4]x_3 &= [15, 20] \\ [4, 5]x_1 + [2, 3]x_2 - [1, 2]x_3 &= [5, 10] \end{aligned} \tag{2.7}$$

Capítulo 3

Otimização sob incerteza: abordagens e medidas

3.1 Introdução e exemplos

Este capítulo aborda problemas de otimização sob incerteza generalizada. Para clarear as ideias, considere o seguinte exemplo: existem três possibilidades de investimentos, (I_1, I_2, I_3) , e três cenários diferentes, otimista, médio, pessimista. Cada investimento pode ter o retorno (em porcentagem) em cada cenário conforme mostrado na tabela 3.1:

Tabela 3.1: Retorno

Investimentos	Otimista	Médio	Pessimista
I_1	15	3	-10
I_2	10	5	0
I_3	7	6	3

Vamos agora considerar quatro maneiras de selecionar o melhor investimento a partir de diferentes perfis de um investidor:

1. **MAXIMAX**: Esta opção maximiza o máximo retorno disponível, compatível com um investidor sem aversão ao risco, ou seja, escolhe-se o melhor dos melhores. Então, neste exemplo selecione o retorno máximo considerando todos os cenários: o investimento I_1 na tabela 3.2 é selecionado:

Tabela 3.2: Critério MAXIMAX

Investimentos	Otimista	Médio	Pessimista	Max
I_1	15	3	-10	15
I_2	10	5	0	10
I_3	7	6	3	7

2. **MAXIMIN**: Esta opção maximiza o retorno mínimo para cada cenário, ou seja, o melhor do pior. Esta opção seria escolhida por um investidor com aversão ao risco. Portanto, neste exemplo, selecionamos o máximo retorno mínimo considerando todos os cenários: Portanto, o investimento I_3 na tabela 3.3 é selecionado:

Tabela 3.3: Critério MAXIMIN

Investimentos	Otimista	Médio	Pessimista	Max
I_1	15	3	-10	-10
I_2	10	5	0	0
I_3	7	6	3	3

3. **Laplace**: Esta opção maximiza a média de retorno para cada cenário, ou seja, o melhor da média. Portanto, neste exemplo, o investimento I_3 na tabela 3.4 é selecionado:

Tabela 3.4: Critério de Laplace

Investimentos	Otimista	Médio	Pessimista	Médio
I_1	15	3	-10	8/3
I_2	10	5	0	15/3
I_3	7	6	3	16/3

4. **MINIMAX Arrependimento**: Esta opção minimiza o arrependimento máximo, ou seja, o melhor investimento considerando o máximo de arrependimento. O arrependimento na escolha de um investimento i no cenário j é a diferença do valor ótimo do cenário j e retorno do investimento i neste cenário j . Por exemplo, se o investidor escolheu a opção de investimento 3 e ocorreu o cenário otimista, ele terá um arrependimento igual a 8. Já se ele escolheu o investimento 1 e ocorreu o cenário otimista, o arrependimento é igual a zero. Portanto, neste exemplo, investimento I_2 na tabela 3.5 é selecionado:

Tabela 3.5: MINIMAX critério de arrependimento

Investimentos	Otimista	Médio	Pessimista	Arrependimento
I_1	0	2	13	13
I_2	5	1	3	5
I_3	8	0	0	8

Neste exemplo de escolha de um investimento considerando a incerteza, nós temos um único objetivo: maximizar o retorno. Em muitas situações práticas existem vários objetivos. Nestes casos a abordagem requer outros conceitos, por exemplo do que seria uma solução boa o suficiente. Novamente, para introduzir o tema considere o seguinte exemplo:

Exemplo 3.1. *Uma pessoa precisa viajar da cidade A para a cidade B. Há quatro opções: (1) carro, (2) trem, (3) avião, (4) ônibus. É preciso analisar dois cenários: (1) bom tempo e (2) mau tempo. O custo (em US\$) é $c_{11} = 120$, $c_{12} = 180$, $c_{21} = 150$, $c_{22} = 160$, $c_{31} = 300$, $c_{32} = 310$, $c_{41} = 60$, $c_{42} = 80$ onde c_{ij} é o custo da opção i no cenário j . O tempo de viagem (em horas) é $t_{11} = 3$, $t_{12} = 5$, $t_{21} = 2$, $t_{22} = 4$, $t_{31} = 1$, $t_{32} = 4$, $t_{41} = 4$, $t_{42} = 7$, onde t_{ij} são os tempos de viagem da opção i no cenário j .*

É natural desejar minimizar o tempo e o custo da viagem, porém são dois objetivos conflitantes mesmo em uma situação sem incerteza. Neste capítulo iremos definir os conceitos e apresentar métodos para resolver problemas como este.

3.2 Minimizar o máximo arrependimento em um problema de otimização inteira

Considere um investidor que possui n possibilidades de investimento, I_k , $k = 1, 2, \dots, n$, sendo que cada investimento k pode resultar em um lucro l_k^j , onde j diz respeito a um cenário dentre M possibilidades. Cada investimento k requer uma quantia de investimento pré-determinada e fixa q_k . O investidor possui um capital C para investir e objetiva maximizar o lucro. Como podemos resolver este problema?

Se não houvesse a incerteza, teríamos o clássico Problema da Mochila, que pode ser modelado da seguinte forma:

$$\begin{aligned} \text{Maximizar} \quad & \text{Lucro} = \sum_{k=1}^n l_k * x_k \\ \text{s.a :} \quad & \sum_{k=1}^n q_k * x_k \leq C \\ & x_k \in \{0, 1\}, \quad k = 1, \dots, n \end{aligned} \quad (3.1)$$

onde $x_k = 1$ se o investimento k é escolhido, e é igual a zero caso contrário. Como neste caso teríamos apenas um cenário, poderíamos expressar o lucro do investimento k como sendo l_k .

Para aplicarmos as abordagens apresentadas na seção 3.1, vamos considerar um exemplo com $n = 5$ possibilidades de investimento, $M = 3$ cenários, uma quantia disponível para aplicação de 50, os lucros por investimentos em cada cenário dados na tabela 3.6 e o valor de cada investimento inicial é apresentado na tabela 3.7.

Tabela 3.6: Lucros de cada investimento por cenário

Investimentos	Cenário 1	Cenário 2	Cenário 3
I_1	18	6	8
I_2	20	24	10
I_3	22	21	19
I_4	15	12	7
I_5	25	18	25

Resolvendo O Problema da Mochila para cada cenário obtemos as seguintes soluções ótimas:

- Solução ótima para o cenário 1: $O_1 = (1, 0, 1, 1, 0)$.

Tabela 3.7: Valor fixo de cada investimento inicial

Investimentos	Valor do investimento inicial
I_1	8
I_2	12
I_3	12
I_4	10
I_5	18

- Solução ótima para o cenário 2: $O_2 = (0, 1, 1, 0, 0)$.
- Solução ótima para o cenário 3: $O_3 = (0, 0, 1, 0, 1)$.

Para encontrar a solução MAXMIN, podemos usar o seguinte modelo de otimização:

$$\begin{aligned}
 &\text{Maximizar} && \text{Minimo} = t \\
 &s.a : && \sum_{k=1}^n q_k * x_k \leq C \\
 &&& t \leq \sum_{k=1}^n l_k^j * x_k \quad j = 1, \dots, M \\
 &&& x_k \in \{0, 1\}, \quad k = 1, \dots, n \\
 &&& t \in \mathbf{Z}
 \end{aligned} \tag{3.2}$$

l_k^j é o lucro dado pelo investimento k no cenário j .

A solução ótima deste modelo para o exemplo que estamos analisando é $O_{MAXMIN} = (0, 0, 1, 0, 1)$, com valor ótimo $Lucro = 39$.

A tabela 3.8 mostra o Lucro obtido em cada cenário considerando as soluções ótimas O_1 , O_2 e O_3 . Esta tabela facilita a compreensão da solução MAXMIN encontrada pelo modelo acima.

Tabela 3.8: Lucro de cada solução ótima por cenário

Solução Ótima	Cenário 1	Cenário 2	Cenário 3
O_1	55	39	34
O_2	42	45	29
O_3	47	39	44

Dada um cenário j e uma solução viável y o arrependimento é definido como sendo $Arrependimento(y, j) = Lucro(y, j) - O_j$, onde, neste caso, $Lucro(y, j) = \sum_{k=1}^n l_k^j * y_k$ e O_j é a solução ótima no cenário j . O máximo arrependimento da solução y é dado por $AMax(y) = Max_j Lucro(y, j)$, sendo neste exemplo $j = 1, 2, 3$. Queremos encontrar y tal que $AMAX(y)$ seja mínimo, ou seja, queremos resolver o seguinte problema de otimização:

$$\text{Minimizar}_x \quad Amax(x) = MAX_j(Lucro(x, j) - O_j) \tag{3.3}$$

onde x é viável e $j = 1, 2, 3$. Mais especificamente no nosso exemplo, precisamos resolver o seguinte problema de otimização:

$$\begin{aligned}
 &\text{Minimizar} && Amax = t \\
 &s.a : && \sum_{k=1}^n q_k * x_k \leq C \\
 &&& t \geq \sum_{k=1}^n l_k^j * x_k - O_j \quad j = 1, \dots, M \\
 &&& x_k \in \{0, 1\}, \quad k = 1, \dots, n \\
 &&& t \in \mathbb{Z}
 \end{aligned} \tag{3.4}$$

Resolvendo este modelo para o nosso exemplo obtemos o Mínimo Arrependimento Máximo igual a 8, e a solução ótima $O_{Amax} = (0, 0, 1, 0, 1)$.

No final deste capítulo apresentamos um exercício para o caso intervalar, ou seja, quando o lucro possível em cada cenário é dado por um intervalo, o que pode ocorrer em situações reais na área de investimentos. Na próxima seção abordaremos o problema de otimização com múltiplos objetivos considerando a incerteza.

3.3 Minimizar o máximo arrependimento em um problema de otimização multiobjetivo

Há situações reais onde além de incertezas existem múltiplos objetivos, como custo, impacto ambiental, lucro, risco e assim por diante. Dentro o problema de otimização multiobjetivo sob incerteza, a pergunta que surge é: “Qual é o melhor solução?” neste contexto. Se formos capazes de definir “melhor”, a próxima pergunta é “Como podemos obtê-lo?” O que segue trata dessas questões.

Uma formulação clássica de um problema de otimização multiobjetivo (POM) é :

$$\begin{aligned}
 &\text{(POM) Minimizar} && f(x) \\
 &s.a : && g_i(x) \leq b, \quad i = 1, \dots, m, \\
 &&& x \in S,
 \end{aligned} \tag{3.5}$$

onde S é um subconjunto aberto de \mathbb{R}^n , $f = (f_1, \dots, f_p) : S \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^p$ e $g = (g_1, \dots, g_m) : S \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ são diferenciáveis. Para definir qual é a melhor solução para o POM, chamada de *solução eficiente*, no contexto de otimização multiobjetivo, as seguintes convenções de notação para igualdades e desigualdades são as seguintes.

Se $x = (x_1, \dots, x_n)$, $y = (y_1, \dots, y_n) \in \mathbb{R}^n$, então

$$\begin{aligned}
 x = y &\Leftrightarrow x_i = y_i, \quad \forall i = 1, \dots, n, \\
 x < y &\Leftrightarrow x_i < y_i, \quad \forall i = 1, \dots, n, \\
 x \leq y &\Leftrightarrow x_i \leq y_i, \quad \forall i = 1, \dots, n, \\
 x \leq y &\Leftrightarrow x_i \leq y_i, \quad \forall i = 1, \dots, n, \quad \text{e existe } j \text{ tal que } x_j < y_j, \\
 x \leq_j y &\Leftrightarrow x_i \leq y_i \quad \forall i = 1, \dots, n, \quad \text{e } x_j < y_j, \text{ para algum } j \in \{1, \dots, n\}.
 \end{aligned}$$

Agora podemos definir o que queremos dizer com solução eficiente e solução fracamente eficiente para um POM.

Definição 3.1. Um ponto factível, \bar{x} , é dito ser uma *solução eficiente* para o POM caso não exista outro ponto factível, x , tal que $f(x) \leq f(\bar{x})$.

Definição 3.2. Um ponto factível, \bar{x} , é dito ser uma *solução fracamente eficiente* para o POM caso não exista outro ponto factível, x , tal que $f(x) < f(\bar{x})$.

Considere o seguinte problema bi-objetivo:

Definição 3.3.

$$\begin{aligned}
 (BPU) \quad & \text{Minimizar } (f_1(x, \xi), f_2(x, \xi)) \\
 \text{s.t. : } & g_i(x) \leq b_i, \quad i = 1, \dots, m, \\
 & x \in S, \xi \in U
 \end{aligned} \tag{3.6}$$

onde S é um conjunto aberto de \mathbb{R}^n , $U = (\xi_1, \dots, \xi_n)$ é um conjunto finito de cenários, $f_i : S \times U \rightarrow \mathbb{R}$ $i=1,2,\dots,n$. e $g = (g_1, \dots, g_m) : S \times U \rightarrow \mathbb{R}^m$ são diferenciáveis

Denote, por $P(\xi)$, o problema de otimização bi-objetivo para um cenário ξ . Denominamos a solução MiniMax Regret Efficient por *MMRE* Duas definições, que são semelhantes ao que Kuhn et al. [49] apresentaram, são dados no contexto de problemas bio-objetivos robustos.

No entanto, a definição é adaptada ao problema minimax regret bi-objetivo ([1]). A generalização para funções *k-objetivo* é simples, embora leve a métodos de solução mais complexos.

Definição 3.4. Solução $x \in S$ é **fracamente MMRE** se x é uma solução eficiente do $P(\xi)$ para pelo menos um ξ em U .

Definição 3.5. Solução $x \in S$ é **fortemente MMRE** se x é um solução eficiente de $P(\xi)$ para cada ξ em U .

Claramente, se $x \in X$ é uma solução fortemente MMRE, então x é uma solução fracamente MMRE solução. Agora, uma nova definição para solução eficiente de um problema bi-objetivo minimax regret para o problema de otimização bi-objetivo sob incerteza é introduzido. Para isso, as soluções MMRE são tomadas no contexto de mini/máximo arrependimento, que é definido a seguir.

Definição 3.6. Solução $x \in X$ é **estritamente MMRE** se x é uma solução eficiente de:

$$\begin{aligned}
 & (MMR-BPU) \\
 \min_{x \in X} \quad & (\max_{\xi \in U} (f_1(x, \xi) - \min_x f_1(x, \xi)), \max_{\xi \in U} (f_2(x, \xi) - \min_x f_2(x, \xi)))
 \end{aligned}$$

Teorema 3.1. Se $x^* \in X$ é uma solução fortemente MMRE, então x^* é uma solução estritamente Solução MMRE.

Demonstração. Seja $x^* \in X$ uma solução fortemente MMRE do MMR-BPU, $z_1^* = \min_x f_1(x, \xi)$ e $z_2^* = \min_x f_2(x, \xi)$ e suponha que x^* não é uma solução estritamente MMRE. Isso significa que existe \hat{x} tal que:

$$(\max_{\xi \in U} (f_1(\hat{x}, \xi) - z_1^*(\xi)), \max_{\xi \in U} (f_2(\hat{x}, \xi) - z_2^*(\xi))) \leq (\max_{\xi \in U} (f_1(x^*, \xi) - z_1^*(\xi)), \max_{\xi \in U} (f_2(x^*, \xi) - z_2^*(\xi)))$$

Dado este resultado, existem ξ_1, ξ_2, ξ_3 e ξ_4 em U (de acordo com as definições 3.1 e 3.5) tal que:

$$(f_1(\hat{x}, \xi_1) - z_1^*, f_2(\hat{x}, \xi_2) - z_2^*) \leq (f_1(x^*, \xi_3) - z_1^*, f_2(x^*, \xi_4) - z_2^*)$$

Está implícito:

1. $(f_1(\hat{x}, \xi_3) - z_1^*, (f_2(\hat{x}, \xi_4) - z_2^*)) \leq (f_1(\hat{x}, \xi_1) - z_1^*, (f_2(\hat{x}, \xi_2) - z_2^*));$

2. $(f_1(\hat{x}, \xi_3) - z_1^*, (f_2(\hat{x}, \xi_3) - z_4^*) \leq (f_1(\hat{x}, \xi_1) - z_1^*, (f_2(\hat{x}, \xi_2) - z_2^*);$
3. $(f_1(\hat{x}, \xi_4) - z_1^*, (f_2(\hat{x}, \xi_4) - z_2^*) \leq (f_1(\hat{x}, \xi_1) - z_1^*, (f_2(\hat{x}, \xi_2) - z_2^*);$
4. $(f_1(\hat{x}, \xi_4) - z_1^*, (f_2(\hat{x}, \xi_3) - z_2^*) \leq (f_1(\hat{x}, \xi_1) - z_1^*, (f_2(\hat{x}, \xi_2) - z_2^*).$

Assim, temos:

$$(f_1(\hat{x}, \xi_3), f_2(\hat{x}, \xi_3)) \leq (f_1(x^*, \xi_3), f_2(x^*, \xi_3))$$

ou

$$(f_1(\hat{x}, \xi_4), f_2(\hat{x}, \xi_4)) \leq (f_1(x^*, \xi_4), f_2(x^*, \xi_4)).$$

Isso é uma contradição, então x^* é uma solução eficiente para todos os cenários. \square

Agora, o Exemplo 3.1 pode ser resolvido do ponto de vista de nossas definições.

Resolvendo o Exemplo 3.1

É fácil ver que:

- x_2, x_3, x_4 são soluções fracamente MMRE, porque são soluções eficientes para algum cenário.
- x_2, x_4 são soluções fortemente MMRE, porque ambos são eficientes soluções para todos os cenários.

Para encontrar as soluções estritamente MMRE, é necessário analisar a eficiência dos seguintes vetores: $fm(x_1) = (120, 4)$, $fm(x_2) = (100, 3)$, $fm(x_3) = (250, 3)$ e $fm(x_4) = (20, 6)$, onde:

$$fm(x_i) = (\max_{\xi \in U} (f_1(x, \xi) - \min_x f_1(x, \xi)), \max_{\xi \in U} (f_2(x, \xi) - \min_x f_2(x, \xi))).$$

Portanto, concluímos que x_2, x_3 e x_4 são MMRE soluções (x_1 não é uma solução estritamente MMRE porque $fm(x_2) < fm(x_1)$).

O Problema de Minimax Regret Bi-objetivo com incerteza generalizada

Considere o seguinte exemplo clássico de um problema de otimização linear sob incerteza (ver [50, 52]).

Exemplo 3.2. *Um fazendeiro cultiva trigo, milho e beterraba em 500 hectares de terra. O custo de plantio por hectare de trigo, milho e beterraba são R\$150, R\$230, e R\$260, respectivamente. O agricultor sabe que pelo menos 200 toneladas de trigo e São necessárias 240 toneladas de milho para a alimentação do gado. Esses valores podem ser levantados em na fazenda ou comprado no mercado. Qualquer produção em excesso da alimentação requisito será vendido. Os preços de venda são R\$170 e R\$150 por tonelada de trigo e milho, respectivamente. O preço de compra é maior, R\$238 por tonelada de trigo e R\$210 por tonelada de milho. Outra cultura possível é a beterraba, que vende a R\$36 por tonelada. No entanto, a comissão estadual impõe uma cota de beterraba açucareira produção de 6.000 toneladas. Qualquer quantidade que exceda a cota só pode ser vendida a R\$10 a tonelada. O rendimento médio da terra do agricultor é 2,5, 3 e 20 toneladas por hectare para trigo, milho e beterraba, respectivamente. Agora, suponha que:*

- *O agricultor coleta dados meteorológicos dos últimos 360 anos de sua cidade de fontes do governo local, e descobre que houve 120 anos de mau tempo, mas o agricultor não sabia dizer qual era o padrão desse mau tempo. São 180 anos nos quais não acontecem coisas ruins; sem tempestades monstruosas, sem granizo devastador, sem surto de insetos. De dados meteorológicos, o agricultor pode determinar o cronograma de plantio durante esses 180 anos. No entanto, como não há dados sobre rendimentos, o agricultor não sabe com certeza quando nesses 180 anos têm rendimentos acima da média ou média. Além disso, os dados meteorológicos mostram que há 60 anos de condições caóticas; misturado épocas boas e más para a agricultura. O agricultor não sabe dizer qual desses 60 anos fornecem colheitas abaixo da média, na média ou acima da média. Por esta informações, o agricultor tem um conjunto aleatório de rendimentos em relação ao incerteza $\hat{v} = v_1 = [2, 2.4, 16]^T$, $v_2 = [2.5, 3, 20]^T$, $v_3 = [3, 3.6, 24]^T$ tal que: $m(\{v_1\}) = 1/3$, $m(\{v_2, v_3\}) = 1/2$, $m(\{v_1, v_2, v_3\}) = 1/6$ (veja a seção 1.2), onde v_1, v_2 e v_3 referem-se aos vetores abaixo rendimentos médios, médios e acima da média de trigo, milho e beterraba.*

Exemplo 3.3. • *O agricultor também deseja minimizar a quantidade de água utilizada por causa da impacto ambiental. Para cada hectare de trigo/beterraba/milho, o consumo é proporcional a 7,25/6,5/10 m³ de água.*

Antes de resolver este problema de exemplo, mais alguns conceitos serão definidas. Um problema é um problema de otimização linear bi-objetivo sob incerteza generalizada (POBIG), se satisfizer a seguinte definição.

Definição 3.7.

$$(POBIG) \quad \begin{array}{ll} \text{Minimize} & (\max_{p \in M} (z_1(x, y) - \min_{x, y} z_1(x, y)), f_2(x)) \\ \text{s.t. :} & Ax \leq b_i, \quad i = 1, \dots, m \\ & p \in M \end{array}$$

onde $z_1(x, y) = c^t x + q_1 \sum_{k=1}^{K_1} p_1^k y_1^k + q_2 \sum_{k=1}^{K_2} p_2^k y_2^k + \dots + q_m \sum_{k=1}^{K_m} p_m^k y_m^k$, $M = (p_1, \dots, p_n)$ é um conjunto finito de probabilidades de todas as realizações de todos os cenários $U = (\xi_1, \dots, \xi_m)$, $f_1 : S \times U \rightarrow \mathbb{R}$, e $f_2 : S \rightarrow \mathbb{R}$ são funções lineares.

Estamos agora em posição de definir para o caso bi-objetivo o que queremos dizer com uma solução minimax regret eficiente.

Definição 3.8. *Solução $x \in X$ é estritamente MMRE para POBIG se x for uma solução eficiente (de acordo com Definição 3.4, Definição 3.5 ou Definição 3.6) do POBIG.*

Exemplo 3.2 é um POBIG. Assim, o problema pode ser matematicamente formulado da seguinte forma:

$$(MP-Ex2) \quad \begin{array}{ll} \min_x & (\max_{p \in M} ((z_1(x) - \min_x z_1(x, p)), (f_2(x))) \\ \text{s.t. :} & Ax \leq b, \\ & p \in M \end{array}$$

A Heurística H- ϵ Aplicada ao Min/Max Regret sob Incerteza

Esta seção apresenta uma heurística baseada no método ϵ -constraint, que chamaremos do método H- ϵ .

Definição 3.9. *Seja o POBIG dado por:*

$$(POBIG) \quad \begin{array}{ll} \text{Minimizar} & (\max_{p \in M} (z_1(x, y) - \min_{x, y} z_1(x, y)), f_2(x)) \\ \text{s.t. :} & Ax \leq b_i, \quad i = 1, \dots, m, \\ & p \in M \end{array}$$

onde

$$z_1(x, y) = c^t x + q_1 \sum_{k=1}^{K_1} p_1^k y_1^k + q_2 \sum_{k=1}^{K_2} p_2^k y_2^k + \dots + q_m \sum_{k=1}^{K_m} p_m^k y_m^k, M = (p_1, \dots, p_n)$$

é um conjunto finito de probabilidades de todas as possíveis cenários.

$$\begin{aligned} U &= (\xi_1, \dots, \xi_m), \\ z_1 &: S \times U \rightarrow \mathbb{R} \text{ e } f_2 : S \rightarrow \mathbb{R} \end{aligned}$$

são funções lineares. Nós associamos este problema com uma família de problemas de otimização linear P_ϵ , $\epsilon \in \mathbb{R}$:

$$P_\epsilon \quad \begin{array}{ll} \text{Minimizar} & h(x) = (\max_{\xi \in U} ((z_1(x, y) - \min_{x, y} z_1(x, y)))) \\ \text{s.t. :} & Ax \leq b_i, \quad i = 1, \dots, m, \\ & f_2(x) \leq \epsilon \\ & p \in M \end{array}$$

Definição 3.10. x^* é uma possível MMRE solução se é uma solução ótima para o problema P_ϵ , $\epsilon \in \mathbb{R}$.

Teorema 3.2. *Seja x^* uma solução estritamente MMRE solution para o POBIG. Então x^* é uma solução possível MMRE solution para o POBIG, onde $\epsilon = f_2(x^*)$.*

Demonstração. Seja x^* uma solução estritamente MMRE para POBIG. Suponha que existam uma solução viável \hat{x} para POBIG tal que $h(\hat{x}) < h(x^*)$. Portanto,

$$(h(\hat{x}), f_2(\hat{x})) \leq (h(x^*), f_2(x)),$$

o que é uma contradição. \square

Observação 3.1. *A contrapositiva deste teorema não é válida. Por exemplo, considere que no exemplo 3.1 temos incerteza apenas no custo, que chamaremos de $f_1(x)$. Reescreva o problema da seguinte maneira: Uma pessoa precisa viajar de cidade A para a cidade B. Existem quatro opções: (1) carro, (2) trem, (3) avião, (4) ônibus. É necessário analisar dois cenários: (1) bom tempo e (2) mau tempo. O custo (em US\$) é $c_{11} = 120$, $c_{12} = 180$, $c_{21} = 150$, $c_{22} = 160$, $c_{31} = 300$, $c_{32} = 310$, $c_{41} = 60$, $c_{42} = 80$ onde c_{ij} é o custo da opção i no cenário j . O tempo de viagem (em horas) é $t_1 = 3$, $t_2 = 2,5$, $t_3 = 1$, $t_4 = 4$. Seja $f_2(x)$ a função tempo de viagem, e $\epsilon = 3$. Nesse caso, temos que x_1 é a solução ótima. No entanto, x_1 não é um Solução MMRE.*

Observação 3.2. *É possível aumentar a região viável e ter uma solução pior? Por o minimax a resposta é sim. Considere que no exemplo acima temos um nova forma de viajar da cidade A para a cidade B e uma quinta opção (5) trem expresso com $c_{51} = 30$, $c_{52} = 170$ e $t_5 = 4$. Para este caso, a incerteza é apenas no custo, que denotaremos $f_1(x)$. Reescreva o problema em da seguinte forma: Uma pessoa precisa viajar da cidade A para a cidade B. Existem quatro opções: (1) carro, (2) trem, (3) avião, (4) ônibus. É necessário análise de dois cenários: (1) bom tempo e (2) mau tempo. O custo (em US\$) é $c_{11} = 120$, $c_{12} = 180$, $c_{21} = 150$, $c_{22} = 160$, $c_{31} = 300$, $c_{32} = 310$, $c_{41} = 60$, $c_{42} = 80$ onde c_{ij} é o custo do opção i no cenário j . O tempo de viagem (em horas) é $t_1 = 3$, $t_2 = 2,5$, $t_3 = 1$, $t_4 = 4$. Este caso resulta em soluções MMRE x_1, x_2, x_3, x_4 , todas com maior $f_1(x)$ do que antes da introdução da opção x_5 .*

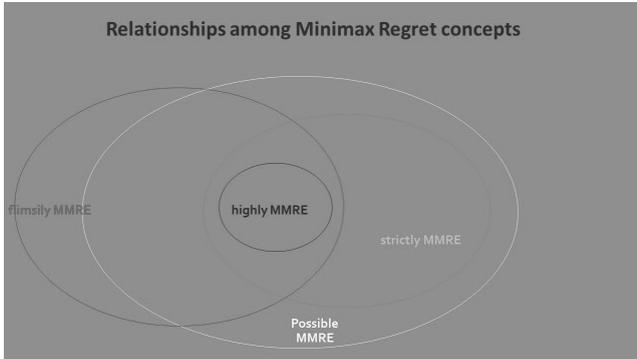


Figura 3.1: Relação entre os conceitos do problema de minimizar o máximo arrependimento

Podemos agora propor o método $H-\epsilon$ para encontrar o arrependimento minimax eficiente solução para o problema de otimização linear bi-objetivo sob incerteza. Este método junta ideias do algoritmo ϵ -constraint [51] e o algoritmo proposto em [52], mas o algoritmo não garantiu que todas as soluções eficientes encontradas são MMRE, apenas que algumas as soluções eficientes são encontradas.

Data: $f_1(x, p)$, $f_2(x)$, A , b , p - o vetor de probabilidades, $\epsilon = (\epsilon_1, \dots, \epsilon_T)$

0. **for** $k=1:T$ **do**

1. Inicialização. Escolha $p^{(1)} = p$. Resolva $\min_x f_1(x)$, e obtenha uma solução ótima x_1 .
2. Seja $j = 1$.
3. Resolva o seguinte problema relaxado corrente para obter uma solução ótima $R(j)$, $x(j)$:

$$\begin{aligned} \min_{R, X} \quad & R \\ \text{s.t. :} \quad & R \geq 0 \\ & R \geq f_{1p^{(i)}}(x) - f_{1p^{(i)}}(x_j) \quad i = 1, \dots, j \\ & f_2(x) \leq \epsilon_k \end{aligned}$$

4. Comece com $p^{(j)}$ e $\bar{p}^{(r)*}$ como em [52]. Calcule $F^{(j)}_{p^{(r)*}}$ e sua solução ótima $x(j+1)$

5. **if** $F^{(j)}_{p^{(r)*}} > R(j)$ **then**

| Seja $p^{r^a} = P^{j+1}$. Seja $j = j + 1$, retorne ao passo 2

end

6. **if** $F^{(j)}_{p^{(r)^a}} \leq R(j)$ **then**

| selecione p que não tenha sido usado nesta iteração, e refaça o sistema como proposto em [52] até encontrar $p^{(r)^a}$ tal que

| $F^{(j)}_{p^{(r)^a}} > R(k)$, então continue para o passo 4

else

| $M = a$ e nós terminamos o procedimento. Uma solução eficiente é X_j

end

end

Nós usamos este método para obter uma aproximação da curva de Pareto para o exemplo 3.2.

Resolvendo o exemplo 3.2: Para obter as probabilidades ρ_1, ρ_2, ρ_3 usamos o mesmo procedimento que [52]. Por exemplo, as funções de probabilidade que geram os menores valores esperados são calculados da seguinte maneira:

$$p_1^1 = Nec(v_1, v_2, v_3) - Nec(v_2, v_3) = 1/2$$

$$p_2^1 = Nec(v_2, v_3) - Nec(v_3) = 1/2$$

$$p_3^1 = Nec(v_3) = 0$$

onde Nec é a medida de de necessidade.

Para gerar a aproximação da curva de Pareto, 21 valores de ϵ foram usados, $\epsilon_i = 2000 + 150 * i$, $i = 0, \dots, 20$, e os valores obtidos são apresentados na tabela 3.9.

Observação 3.3. *É importante lembrar que as soluções apresentadas na tabela 3.9 podem não ser soluções MMRE.*

$\min_x \{\max_{\xi \in U} (f_1(x, \xi) - \min_x f_1(x))\}$	$f_2(x)$
977.289	2000
977.260	2150
977.219	2300
977.014	2450
976.532	2600
976.516	2750
976.453	2900
976.348	3050
976.167	3350
910.456	3500
599.567	3650
211.805	3800

Tabela 3.9: Soluções obtidas usando o método H- ϵ

3.4 Mensurando a qualidade de uma solução: a dureza (hardness) de uma solução

Um problema de otimização sob incerteza (OPUU) é uma família de problemas de otimização que compartilham a mesma estrutura, mas com valores de parâmetros diferentes. Esses parâmetros são *incertos* e pertencem a um *conjunto incerto* [53].

Definição 3.11. *Dado um objetivo (minimização/maximização), uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$, uma função de restrição $g : \mathbb{R}^n \times \mathbb{R}^q \rightarrow \mathbb{R}^m$ e um conjunto não vazio $\mathcal{V} \subset \mathbb{R}^q$, uma OPUU associada a ele pode ser definida como:*

$$\{\min f(\vec{x}) : g(\vec{x}, \vec{v}) \leq 0\}_{\vec{v} \in \mathcal{V}}, \quad (3.7)$$

onde \mathcal{V} é chamado conjunto incerto e \vec{v} é chamado parâmetro incerto.

Considere a situação em que o conjunto incerto é descrito por informações de contorno para cada parâmetro incerto. Pode-se optar por encontrar a solução robusta para o OPUU, ou seja, encontrar a melhor solução viável para todos os problemas de otimização pertencentes à família OPUU. Esta pesquisa apresentará duas medidas de qualidade para comparação de soluções OPUU: a medida de probabilidade de violação de restrição e uma nova, a medida de dureza. A fim de demonstrar a eficácia dessas métricas, uma OPUU será resolvida por uma abordagem de otimização robusta para cinco diferentes níveis de proteção de incerteza.

A principal contribuição desta pesquisa é a definição e verificação da medida de eficácia denominada dureza. Nossa nova medida é útil porque indica como as soluções se comparam (melhor/pior) a todas as soluções possíveis, considerando também as preferências do tomador de decisão. A probabilidade de violação do conjunto de restrições não é uma medida comparativa e não é sensível a preferências. A medida de dureza pode comparar soluções dadas por qualquer método para uma OPUU que tenha pelo menos uma solução possível. Tem 1 como limite superior, o que significa que a solução é robusta. Quanto mais próxima a medida chega de 0, menos robusta é a solução.

3.4.1 Otimização Robusta

O conceito geral de otimização robusta é uma solução que continua sendo uma solução satisfatória, segundo critérios do decisor, em qualquer cenário. Para tratar adequadamente a incerteza associada a este conceito, deve-se definimos o conceito de *otimização sob incerteza* e sua respectiva contraparte robusta.

Definição 3.12. *Um problema de otimização sob incerteza é uma família de problemas de otimização que compartilham a mesma estrutura, mas com diferentes parâmetros pertencentes a um conjunto incerto \mathcal{V} :*

$$\left\{ \min_x f(x) : h(x, v_h) = 0, g(x, v_g) \leq 0 \right\}_{(v_h, v_g) \in \mathcal{V}} \quad (3.8)$$

Onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é a função objetivo, $g : \mathbb{R}^n \times \mathbb{R}^{q_1} \rightarrow \mathbb{R}^{m_1}$ é a função de restrição de desigualdade, $h : \mathbb{R}^n \times \mathbb{R}^{q_2} \rightarrow \mathbb{R}^{m_2}$ é função de restrição de igualdade, $x \in \mathbb{R}^n$ é a variável de decisão, $(v_g, v_h) \in \mathbb{R}^{q_1+q_2}$ são os parâmetros incertos que pertencem a conjuntos não vazios incertos conjunto $\mathcal{V} \subset \mathbb{R}^{q_1+q_2}$.

Definição 3.13. *A contrapartida robusta de um problema de otimização sob incerteza (3.8) é o problema de otimização:*

$$\min_x \left\{ \bar{f}(x) = \sup_{(v_h, v_g) \in \mathcal{V}} f(x) \mid h(x, v_h) = 0, g(x, v_g) \leq 0, \forall (v_h, v_g) \in \mathcal{V} \right\} \quad (3.9)$$

Apresentamos a seguir o conceito de uma solução ϵ -robusta:

Definição 3.14. *Seja $e_l \in \mathbb{R}^l$ o vetor de dimensão l que tem todas as entradas iguais a 1. Dado $\epsilon \in \mathbb{R}$, $\epsilon > 0$, um vetor $x \in \mathbb{R}^n$ é chamado de ϵ -robusto viável para (3.8) quando:*

$$|h(x, v_h)| \leq \epsilon \cdot e_{m_1}, \forall v_h \in \mathcal{V}_h = \{v : \exists x \in \mathbb{R}^{q_2} \mid (v, x) \in \mathcal{V}\} \quad (3.10)$$

$$g(x, v_g) \leq \epsilon \cdot e_{m_2}, \forall v_g \in \mathcal{V}_g = \{v : \exists x \in \mathbb{R}^{q_1} \mid (x, v) \in \mathcal{V}\} \quad (3.11)$$

3.4.2 Probabilidade de violação do conjunto de restrições

A primeira medida é chamada *probabilidade de violação do conjunto de restrições*. Dada a OPUU (3.7) com uma função de restrição $g : \mathbb{R}^n \times \mathbb{R}^q \rightarrow \mathbb{R}^m$ e um conjunto incerto $\mathcal{V} \subset \mathbb{R}^q$, a medida de probabilidade de violação do conjunto de restrições $\zeta : \mathbb{R}^n \rightarrow [0, 1]$ é definida como

$$\zeta(\vec{x}) = Pr \left(\sum_{j=1}^m (g(\vec{x}, \vec{v}))_j^+ > 0 : \vec{v} \in \mathcal{V} \right) \quad (3.12)$$

onde o sinal de adição sobrescrito é a função $(\cdot)^+ : \mathbb{R} \rightarrow [0, \infty)$ definida como $(x)^+ = 0$ se $x < 0$ e $(x)^+ = x$ caso contrário.

A probabilidade de violação do conjunto de restrições surge no contexto da otimização de restrições de chance. Esta abordagem reduz a OPUU a um problema de otimização determinístico, onde a probabilidade de violação é limitada por um limitANTE superior. Essa medida carece de sensibilidade e pode dar a impressão ao tomador de decisão de que não há diferença entre as soluções da solução robusta para qualquer outra com menor proteção de incerteza.

3.4.3 Dureza

A segunda medida é uma nova medida chamada medida *dureza* (apresentado [54]). É uma medida de viabilidade que surge no contexto de otimização robusta e tem a capacidade de classificar o nível de proteção de incerteza de forma comparativa. Essa característica torna a dureza um indicador útil para um processo de tomada de decisão que envolva a solução de uma OPUU.

Considere a OPUU definida por (3.7). Seja $\beta : \mathbb{R}^n \rightarrow \mathbb{R}^m$ uma função, chamada *fragilidade positiva* (FP), definida como

$$\beta^+(\bar{x}) = \{ \bar{w} \in \mathbb{R}^m : \bar{w}_i = \sup \{ g(\bar{x}, \bar{v})_i^+ : \bar{v} \in \mathcal{V} \} \}. \quad (3.13)$$

O FP é uma medida multidimensional não negativa do pior valor de violação de restrição possível de um problema de otimização pertencente à família OPUU. É uma medida que dá proteção à incerteza em questões de viabilidade. Ele considera a intensidade da violação de viabilidade de uma determinada solução OPUU, o que não é feito pela medida de probabilidade de violação do conjunto de restrições. Nesse sentido, o FP é uma medida inovadora. No entanto, não induz uma ordem no conjunto de soluções.

Seja $\eta : \mathbb{R}^n \rightarrow [0, 1]$ a medida chamada *dureza* definida como:

$$\eta(\bar{x}) = \sum_{j=1}^m p_j \cdot \int_0^1 Pr(\alpha \cdot \beta^+(\bar{x})_j \geq g(\bar{x}, \bar{v})_j^+ : \bar{v} \in \mathcal{V}) d\alpha. \quad (3.14)$$

onde $\sum_{j=1}^m p_j = 1, p_j \geq 0$ representa a escala de importância do decisor.

A dureza é uma medida não negativa de qualidade com 1 como limite superior. É uma comparação ponderada de quão distante o valor de restrição do pior cenário está de todos os valores de restrição possíveis. Indica o quão difícil é violar as restrições de uma OPUU em termos do conjunto incerto e o impacto dos parâmetros incertos em cada restrição.

As propriedades matemáticas da função de dureza levaram a alguns resultados interessantes em relação à robustez.

Proposição 3.1. *Considere uma OPUU definida como apresentada em (3.7). Se uma solução $\bar{x} \in \mathbb{R}^n$ é tal que $g(\bar{x}, \bar{v}) \leq 0, \forall \bar{v} \in \mathcal{V}$, então $\eta(\bar{x}) = 1$.*

Proposição 3.2. *Se \bar{x}^* é a solução robusta para (3.7), então $\eta(\bar{x}^*) = 1$.*

Ou seja, a medida de dureza é máxima para a solução robusta de uma OPUU. Mas o que é preciso para definir uma solução robusta dado que a sua robustez é máxima?

Proposição 3.3. *Considere o vetor $\bar{x}^* \in \mathbb{R}^n$, uma função $g : \mathbb{R}^n \times \mathbb{R}^q \rightarrow \mathbb{R}^m$ e $\mathcal{V} \subset \mathbb{R}^q$. Suponha que $g(\bar{x}^*, \bar{v}) \leq 0, \forall \bar{v} \in \mathcal{V}$. Se \bar{x}^* é uma solução ótima para um problema de otimização pertencente a OPUU (3.7), então \bar{x}^* é uma solução robusta de OPUU (3.7).*

Observe que a Proposição 3.3 apresenta uma condição necessária, mas não suficiente para caracterizar uma solução robusta. Considere a seguinte OPUU:

$$\max_{x,y} y \quad (3.15)$$

$$\text{S.t. } y + (x - 1) \cdot \left(\frac{\sin \theta}{1 - \cos \theta} \right) \leq 0 \quad \forall \theta \in \left[\frac{\pi}{4}; \frac{3\pi}{4} \right] \quad (3.16)$$

$$y - (x + 1) \cdot \left(\frac{\sin \theta}{\cos \theta + 1} \right) \leq 0 \quad \forall \theta \in \left[\frac{\pi}{4}; \frac{3\pi}{4} \right] \quad (3.17)$$

Sua solução robusta é $(x^*, y^*) = (0, -1 + \sqrt{2})$ mas todo problema pertencente à família tem uma solução ótima na forma $(x^*, y^*) = (\cos \theta, \sin \theta)$.

Proposição 3.4. *Se o conjunto incerto $|\mathcal{V}| = L > 1$ e:*

1. $\bar{x}^* \in \mathbb{R}^n$ é tal que $g(\bar{x}^*, \vec{v}) \leq 0, \forall \vec{v} \in \mathcal{V}$;
2. Não existe $\vec{w} \in \mathbb{R}^n$ com $f(\vec{w}) < f(\bar{x}^*)$ e $g(\vec{w}, \vec{v}) \leq 0, \forall \vec{v} \in \mathcal{V}$;
3. $\eta(\bar{x}^*) = 1$

Então \bar{x}^* é uma solução robusta de OPUU (3.7).

Agora, ilustraremos ambas as medidas usando uma versão modificada do exemplo de Birge e Louveaux (2011) página 4. A vantagem da medida de dureza para este exemplo sobre a outra medida será apresentada.

3.5 De volta ao problema do fazendeiro

Um fazendeiro cultiva trigo, milho e beterraba em uma área de 500 hectares de terra. O fazendeiro também tem gado para cuidar e precisa de pelo menos 240 toneladas (T) de milho e 200 toneladas de trigo para alimentá-los. Existe um atacadista onde ele pode comprar trigo e milho se não puder produzir essas quantidades por 140% do preço de venda do fazendeiro. Nos últimos dez anos, os preços médios de venda do trigo e do milho foram de \$170/t e \$150/t, respectivamente. O governo sabe que a cultura da beterraba costuma ser a mais lucrativa. Para controlar os preços e a produção, o governo fixa em 6.000 toneladas a quantidade de beterraba que pode ser vendida a \$36/t e o excedente da cota pode ser vendido a \$10/t. O agricultor sabe que sua terra pode produzir cerca de 2, 5 T, 3 T e 20 T por hectare de trigo, milho e beterraba, respectivamente, e quer saber a quantidade de terra que deve ser plantada em trigo, milho e safra de beterraba, o peso total da produção desses produtos, a quantidade de ração que ele precisará comprar do atacadista. O principal objetivo do agricultor é maximizar os lucros.

O fato de o agricultor conhecer “aproximadamente” a eficiência produtiva de sua terra e o fato de planejar todo o seu trabalho com base nisso, indica que, pelos dados, existe um nível de risco. Os dados indicam que há rendimentos bastante diferentes para a mesma safra em anos diferentes, principalmente devido às mudanças nas condições climáticas. Rendimentos variando de 20 a 25% acima ou abaixo do rendimento médio não são incomuns, mostrando que é útil abordar a incerteza associada a essas informações.

$$\begin{aligned} \min \quad & 150x_1 + 230x_2 + 260x_3 + 238y_1 - 170w_1 \\ & + 210y_2 - 150w_2 - 36w_3 - 10w_4 \end{aligned} \quad (3.18)$$

$$\text{S. a } x_1 + x_2 + x_3 - 500 \leq 0 \quad (3.19)$$

$$-v_1 \cdot x_1 - y_1 + w_1 + 200 \leq 0 \quad \forall v_1 \in [2, 3] \quad (3.20)$$

$$-v_2 \cdot x_2 - y_2 + w_2 + 240 \leq 0 \quad \forall v_2 \in [2.4, 3.6] \quad (3.21)$$

$$-v_3 \cdot x_3 + w_3 + w_4 \leq 0 \quad \forall v_3 \in [16, 24] \quad (3.22)$$

$$w_3 - 6.000 \leq 0 \quad (3.23)$$

$$-1 \cdot (x_1, x_2, x_3, y_1, y_2, w_1, w_2, w_3) \leq 0 \quad (3.24)$$

Suponha que o agricultor deseja encontrar uma solução que permaneça viável para certas combinações dos parâmetros do pior caso. É plausível supor que nem tudo pode dar errado ao mesmo tempo. Bertimas e Sim [55] propuseram um modelo de otimização robusto que leva em consideração um orçamento de incerteza. Seja J_i o conjunto de parâmetros incertos na i -ésima restrição. Um parâmetro robusto $\Gamma_i \in [0, |J_i|]$ é definido para cada restrição i que é afetada pela incerteza. Ou seja, $\Gamma_i \in [0, 1]$, $\forall i = 1, \dots, 5$. Para cada valor possível de $\Gamma \in [0, 1]^5 \subset \mathbb{R}^5$, o modelo matemático apresentado a seguir, representa o problema de otimização determinística onde a solução permanecerá viável mesmo que $\lfloor \Gamma_i \rfloor$ assumem seus valores máximos (ou seja, a quantidade de parâmetros incertos que podem assumir seu valor máximo é o maior inteiro menor que Γ_i). No problema do agricultor, se $\Gamma_1 = 1$ então a solução ótima para (3.26) estará protegida de cenários onde a eficiência da produção de trigo é a pior possível. Uma modificação direta do Modelo (3.18)-(3.24) resulta no modelo robusto (3.26)-(3.37).

O problema do agricultor sob incerteza pode ser formulado conforme apresentado em Birge e Louveaux (2011). Deixe cada parâmetro assumir valores em um intervalo simétrico em torno da média com magnitude de 40% do valor nominal. O conjunto incerto $\mathcal{V} \subset \mathbb{R}^3$ pode ser definido como $\mathcal{V} = [2, 3] \times [2.4, 3.6] \times [16, 24]$.

Suponha que o agricultor deseja encontrar uma solução que permaneça viável para certas combinações dos parâmetros do pior caso. É plausível supor que nem tudo pode dar errado ao mesmo tempo. Bertimas e Sim [55] propuseram um modelo de otimização robusto que leva em consideração um orçamento de incerteza. Seja J_i o conjunto de parâmetros incertos na i -ésima restrição. Um parâmetro robusto $\Gamma_i \in [0, |J_i|]$ é definido para cada restrição i que é afetada pela incerteza. Ou seja, para o problema do fazendeiro, $\Gamma_i \in [0, 1]$, $\forall i \in \{1, 2, 3\}$. Para cada valor possível de $\Gamma \in [0, 1]^3 \subset \mathbb{R}^3$, o modelo matemático apresentado a seguir, representa o problema de otimização determinística onde a solução permanecerá viável mesmo que $\lfloor \Gamma_i \rfloor$ assumem seus valores máximos. A quantidade de parâmetros incertos que podem assumir seu valor máximo é o maior inteiro menor que Γ_i .

$$\min \quad 150 \cdot x_1 + 230 \cdot x_2 + 260 \cdot x_3 + 238 \cdot y_1 - 170w_1 \quad (3.25)$$

$$210 \cdot y_2 - 150 \cdot w_2 - 36 \cdot w_3 - 10w_4 \quad (3.26)$$

$$\text{s. t.} \quad x_1 + x_2 + x_3 \leq 500 \quad (3.27)$$

$$2,5 \cdot x_1 + y_1 - w_1 - z_1 \cdot \Gamma_1 - p_{21} \geq 200 \quad (3.28)$$

$$3 \cdot x_2 + y_2 - w_2 - z_2 \cdot \Gamma_2 - p_{32} \geq 240 \quad (3.29)$$

$$w_3 + w_4 + z_3 \cdot \Gamma_3 + p_{43} \leq 20 \cdot x_3 \quad (3.30)$$

$$w_3 \leq 6000 \quad (3.31)$$

$$z_1 + p_{21} \geq 0,5 \cdot r_1 \quad (3.32)$$

$$z_2 + p_{32} \geq 0,6 \cdot r_2 \quad (3.33)$$

$$z_3 + p_{43} \geq 4 \cdot r_3 \quad (3.34)$$

$$-r_j \leq x_j \leq r_j, \forall j \in \{1, 2, 3\} \quad (3.35)$$

$$x_1, x_2, x_3, y_1, y_2, w_1, w_2, w_3, w_4 \geq 0 \quad (3.36)$$

$$z_1, z_2, z_3, r_1, r_2, r_3, p_{21}, p_{32}, p_{43} \geq 0 \quad (3.37)$$

No problema do agricultor, se $\Gamma_1 = 1$ então a solução ótima para (3.26) - (3.37) estará protegida de cenários onde a eficiência da produção de trigo é a pior possível.

3.6 Resumo do Capítulo

Este capítulo apresentou alguns perfis de tomadores de decisão com respeito ao risco considerando a incerteza, que está presente na maioria das situações que estes profissionais enfrentam. Destacamos a abordagem minimax regret, ou seja, de minimizar o máximo arrependimento. Para mensurar a qualidade da solução em um problema de incerteza apresentamos a medida chamada Dureza, ou Hardness.

3.7 Exercícios

Nota: Os exercícios deste capítulo poderão ser resolvidos computacionalmente utilizando as ferramentas apresentadas no capítulo 5. Assim, neste capítulo espera-se a formulação dos modelos matemáticos.

1. [Para resolver computacionalmente este exercício recomenda-se o estudo do capítulo 5. Sem o uso das ferramentas computacionais, o leitor deve formular o modelo matemático.] Considere o Problema de Investimento como abordado no capítulo 2. A tabela 3.10 apresenta o intervalo de lucro para cada investimento em cada cenário.

A tabela 3.11 apresenta o valor fixo de cada investimento inicial.

Encontre as soluções MaxMin e Min Max Regret.

2. Seja $G = (V, A)$ um grafo direcionado, no qual V é um conjunto de vértices e A é um conjunto de arcos, onde cada arco $(i, j) \in A$ está associado a um peso $c_{ij} \in R$. Dados $\{s, t\} \in V$, o Problema do Caminho Mais Curto (SP, do inglês Shortest Path Problem) consiste em encontrar um caminho entre o vértice de origem s e o vértice de destino t em G cujo somatório dos custos associados aos arcos no caminho seja mínimo.

Tabela 3.10: Lucros de cada investimento por cenário

Investimentos	Cenário 1	Cenário 2	Cenário 3	Médio
I_1	[17,19]	[5,7]	[7,9]	
I_2	[19,21]	[23,25]	[9,11]	
I_3	[21,23]	[20,22]	[18,20]	
I_3	[14,16]	[11,13]	[6,8]	
I_3	[24,26]	[17,19]	[24,26]	

Tabela 3.11: Valor fixo de cada investimento inicial

Investimentos	Valor do investimento inicial
I_1	8
I_2	12
I_3	12
I_3	10
I_3	18

SP modela diversos problemas de interesse prático e teórico. Entretanto, neste problema cada arco (i, j) em A está associado a um custo x_0 , fato que impede que vários problemas de roteamento encontrados no mundo real possam ser modelados via SP, pois frequentemente os dados deste tipo de problema estão sujeitos a incertezas.

Considere que cada peso c_{ij} seja um intervalo $c_{ij} = [l_{ij}, u_{ij}]$. Modele o problema como MinMax Regret.

3. Dado o seguinte problema de otimização linear sob incerteza:

$$\begin{aligned}
 & \min -x - y \\
 \text{s. a: } & a_{11} \cdot y + a_{12} \cdot x - 1 \leq 0, & a_{11} \in [2, 4], a_{12} \in [-3, -1] \\
 & -y + a_{22} \cdot x - 2 \leq 0, & a_{22} \in [2, 4] \\
 & -x \leq 0 \\
 & -y \leq 0
 \end{aligned}$$

encontre o *hardness* (dureza) da solução associada considerando estes três cenários:

1. extremo direito do intervalo (the right extreme of the interval) - REI
2. o ponto médio do intervalo (the midpoint of the interval) - MPI
3. o extremo esquerdo do intervalo (left extreme of the interval) - LEI

Seção	a_{11}	a_{12}	a_{22}
REI	4	-1	4
MPI	3	-2	3
LEI	2	-3	2

Capítulo 4

Aplicações

4.1 Introdução

Neste capítulo apresentamos algumas aplicações com base em nossa experiência como pesquisadores e professores. A primeira aplicação trata do problema de geometria de distâncias e como tratar a incerteza decorrente da medida da distância entre átomos pela Ressonância Magnética Nuclear. Em seguida apresentamos a modelagem matemática e testes computacionais do problema de localização de alvos por sensores, também abordado por meio da abordagem CIA apresentado no capítulo 2. A seção 4.4 apresenta o problema de otimização linear intervalar e uma nova abordagem ainda inédita na literatura. Todos estes exemplos são de otimização contínua. Como podemos tratar a incerteza em um problema de otimização inteira? E se tivermos mais de um objetivo? É justamente esse tipo de problema que resolvemos no caso do problema de corte de estoque multiobjetivo com incerteza. Em seguida apresentamos um interessante modelo matemático para detecção de esclerose múltipla que incorpora as incertezas inerentes ao processo com base nos tomadores de decisão, neste caso os médicos. Também apresentamos uma clássica aplicação às finanças: otimização de portfólio, que também se caracteriza pela destacada incerteza do mercado. Por fim apresentamos dois aplicativos que foram desenvolvidos para auxiliar o enfrentamento da covid-19 no início da pandemia em 2020. Em um deles, previsão e otimização do número de leitos de UTI em um hospital ou cidade, existe forte componente incerto. Este é um problema real e importante e que não possui uma única resposta. Que abordagem o leitor desenvolveria? Esta é uma das perguntas deste capítulo.

4.2 O Problema de Geometria de Distâncias sob Incerteza

O problema de encontrar a estrutura tridimensional de uma proteína, isto é, as posições de todos os seus átomos, tem uma grande importância para a biologia, química e medicina, já que a estrutura 3D está relacionada com a função da proteína [56]. Experimentalmente, o problema de determinar a geometria da proteína pode ser abordada através de cristalografia de raios-X ou nuclear espectroscopia de ressonância magnética (NMR) [57]. Teoricamente, o problema pode ser resolvido minimizando o potencial molecular associado função de energia [58, 60].

A espectroscopia de Ressonância Magnética Nuclear pode ser usada quando uma proteína não pode ser cristalizada [61], fornecendo medição de distâncias entre átomos que estão próximos o suficiente. O cálculo da estrutura da proteína 3D usando esta distância informação é uma das aplicações mais importantes do *Distância Problema de Geometria* (DGP) em \mathbb{R}^3 [62], está definido do seguinte modo.

Definição 4.1. *Dado um grafo não direcionados $G = (V, E, d)$ cujas arestas são ponderadas por $d : E \rightarrow (0, \infty)$, encontre uma função $x : V \rightarrow \mathbb{R}^3$ tal que*

$$\forall \{i, j\} \in E, \|x_i - x_j\| = d_{i,j}, \quad (4.1)$$

onde $x_i = x(i)$, $x_j = x(j)$, $d_{i,j} = d(\{i, j\})$, e $\|x_i - x_j\|$ é a distância euclidiana entre x_i e x_j .

Devido às incertezas nos dados de RMN, as equações (4.1) devem ser substituído por

$$\underline{d}_{i,j} \leq \|x_i - x_j\| \leq \bar{d}_{i,j}, \quad (4.2)$$

$\forall \{i, j\} \in E$ com $d_{i,j} \in [\underline{d}_{i,j}, \bar{d}_{i,j}]$, tornando o problema ainda mais difícil de resolver. Nesse caso, o problema é chamado *Interval DGP*, ou *iDGP* para abreviar [63].

Para clarear as ideias, apresentamos na tabela a seguir algumas distâncias entre átomos de hidrogênio da proteína 506f [64], obtida pelo Dr. Ítalo Caruso e demais pesquisadores do Centro Multiusuário de Inovação Biomolecular do Ibilce-Unesp.

3HN	3HA	2.580	0.832	0.832
4HN	4HA1	2.464	0.759	0.759
4HN	4HA2	2.503	0.783	0.783
5HN	5HA	2.600	0.845	0.845
6HN	6HA	2.711	0.918	0.918
...
65HN	65HA	2.594	0.841	0.841
66HN	66HA	2.481	0.770	0.770
Leucina67	sem	informação	-	-

Na primeira linha, por exemplo, temos que a distância entre o átomo de Hidrogênio ligado ao Nitrogênio a o Hidrogênio ligado ao Carbono alpha é de 2.58 Ångstrom, e o erro de medição é de até 0.832 Ångstrom. Ou seja, temos uma incerteza nesta distância que pode ser quantificada por um intervalo, ou por meio de uma função, onde a distância d pode ser dada por $d = 1.748 + 1.664 * \lambda$, onde $0 \leq \lambda \leq 1$. Esta é a abordagem que descrevemos a seguir e que foi por nós proposta em [65].

Os autores em [66] propõem outro tipo de problema de Geometria de Distâncias, chamado em inglês de *Generalized DG Problem* (GDGP), onde a ideia é calcular um conjunto significativo de soluções para o *iDGP*. Em vez de resolver o sistema de desigualdades (4.2), o GDGP determina uma posição de “equilíbrio” e um máximo raio de flutuação possível para cada átomo da molécula, de modo que todas possíveis posições atômicas não violem nenhum dos limites de distância fornecidos. Formalmente, o GDGP pode ser descrito da seguinte forma.

Definição 4.2. *Seja $G = (V, E, d)$ uma instância do $iDGP$, onde $d_{i,j} \in [\underline{d}_{i,j}, \bar{d}_{i,j}]$ e $|V| = n$. O GDGP consiste em encontrar $x \in \mathbb{R}^{3n}$ e $r \in \mathbb{R}^n$ tais que*

$$\begin{aligned} \max_{x,r} \quad & \sum_{i=1}^n r_i \\ \text{s.a.} \quad & \|x_i - x_j\| + r_i + r_j \leq \bar{d}_{i,j}, \\ & \|x_i - x_j\| - r_i - r_j \geq \underline{d}_{i,j}, \\ & r_i \geq 0, \quad i = 1, \dots, n. \end{aligned} \quad (4.3)$$

Como o GDGP é um problema difícil, os autores em [66] apresentam um algoritmo de construção geométrica para resolver uma sequência de subproblemas do GDGP. Basicamente, o algoritmo calcula as posições e raios de flutuação de os átomos, um de cada vez, usando as restrições de distância do determinado átomos aos indeterminados. Em vez de resolver o GDGP exatamente, o algoritmo de construção geométrica fornece apenas uma solução aproximada.

O algoritmo primeiro encontra quatro átomos (não no mesmo plano) cujas distâncias interatômicas são conhecidas. As posições atômicas restantes são então determinadas, uma de cada vez, por resolvendo o seguinte subproblema (nós o chamamos de *Sit-Wu Problem*, SWP (abreviado),

$$\begin{aligned} \max_{x_{l+1}, r_{l+1}} \quad & r_{l+1} \\ \text{s.a.} \quad & \|x_i - x_{l+1}\| + r_i + r_{l+1} \leq \bar{d}_{i,l+1}, \quad i = 1, \dots, l, \\ & \|x_i - x_{l+1}\| - r_i - r_{l+1} \geq \underline{d}_{i,l+1}, \quad i = 1, \dots, l, \\ & r_{l+1} \geq 0, \end{aligned} \quad (4.4)$$

onde $x_{l+1} \in \mathbb{R}^3, r_{l+1} \in \mathbb{R}$ são a posição e o raio de flutuação a ser determinado para o átomo $l+1$, $l \geq 4$, e $x_i \in \mathbb{R}^3, r_i \in \mathbb{R}$, $i = 1, \dots, l$, são as posições e raios de flutuação já determinados, a partir dos quais são dadas distâncias limitadas ao átomo $l+1$. Para uma instância $iDGP$ com $|V| = n$ e $|E| = m$, o GDGP (4.3) possui $4n$ variáveis e $2m+n$ restrições, enquanto o SWP (4.4) tem apenas 4 variáveis e $2l+1$ restrições, $l < m$.

Teorema 4.1. *Se os limites em (4.4) são finitos e a região factível não está vazia, o SWP deve ter uma solução ótima.*

Teorema 4.2. *Se $x_{l+1} \in \mathbb{R}^3, r_{l+1} \in \mathbb{R}$ é uma solução viável para o SWP (4.4), $l \geq 4$ e $\{(x_1, r_1), (x_2, r_2), (x_3, r_3), (x_4, r_4)\}$ é um dado ponto de partida viável, então $\{(x_i, r_i) : i = 1, \dots, n\}$ é uma solução viável para o GDGP.*

Com base nos resultados computacionais fornecidos em [66], os autores afirmam claramente que o algoritmo não está pronto para ser aplicado a dados reais de NMR e é necessário mais desenvolvimento. Os resultados apresentados apenas confirmam que o algoritmo converge razoavelmente para problemas de teste cuidadosamente construídos. O objetivo do algoritmo de construção geométrica para o GDGP é resolver uma sequência de subproblemas definida por (4.4). Embora a solução obtida não é necessariamente ótima para o GDGP original, existem resultados de otimização e viabilidade dados pelos próximos dois teoremas [66].

O teorema 2 implica que se for possível encontrar um ponto de partida viável $(x_1, r_1), (x_2, r_2), (x_3, r_3), (x_4, r_4)$ e uma sequência de factíveis soluções (x_i, r_i) para os subproblemas SWP, $i = 5, \dots, n$, $\{(x_i, r_i) : i = 1, \dots, n\}$ é uma solução viável para o GDGP. Este é um resultado importante, mas observe que quando um SWP está sendo resolvido, *por exemplo* calculando valores para (x_{l+1}, r_{l+1}) , todos os anteriores

valores para r_i , $i = 1, \dots, l$, são fixados em seus valores calculados. Por isso, pode acontecer que, para algum $k < l + 1$,

$$r_k > \bar{d}_{k,l+1},$$

o que implica que

$$\|x_k - x_{l+1}\| < 0.$$

Isso significa que esse SWP específico é inviável.

Em [65] propomos um modelo diferente para superar esta desvantagem, que chamamos de *restrição Interval Problem* (CIP), dado por

$$\max_{x_{l+1}, r, \lambda} \sum_{i=1}^{l+1} r_i \quad (4.5)$$

$$s.a : \|x_i - x_{l+1}\|^2 = (\underline{d}_{i,l+1} + \lambda_i(\bar{d}_{i,l+1} - \underline{d}_{i,l+1}))^2, \quad i = 1, \dots, l, \quad (4.6)$$

$$r_{l+1} \leq (1 - \lambda_i)(\bar{d}_{i,l+1} - \underline{d}_{i,l+1}) - r_i, \quad i = 1, \dots, l, \quad (4.7)$$

$$r_{l+1} \leq \lambda_i(\bar{d}_{i,l+1} - \underline{d}_{i,l+1}) - r_i, \quad i = 1, \dots, l, \quad (4.8)$$

$$r_i \leq \bar{r}_i, \quad i = 1, \dots, l, \quad (4.9)$$

$$r_i \geq 0, \quad i = 1, \dots, l + 1, \quad (4.10)$$

$$0 \leq \lambda_i \leq 1, \quad i = 1, \dots, l, \quad (4.11)$$

onde $x_{l+1} \in \mathbb{R}^3$, $r_{l+1} \in \mathbb{R}$ são a posição e raio de flutuação a ser determinado para o átomo $l + 1$, $l \geq 4$, $r_i \in \mathbb{R}$, $i = 1, \dots, l$, são os raios de flutuação já determinados átomos $x_i \in \mathbb{R}^3$, $i = 1, \dots, l$, dos quais são dados limites de distância ao átomo $l + 1$, e \bar{r}_i , $i = 1, \dots, l$, são os raios encontrados nos subproblemas anteriores.

Observe que (7) e (8) são usados para obter uma solução max-min para r_{l+1} . Além disso, para evitar a inviabilidade de alguns dos novos subproblemas, os raios anteriores não são mais fixos, mas apenas limitados. Além disso, as restrições com funções de raiz quadrada são substituídas pela igualdade restrições.

$$\|x_i - x_{l+1}\|^2 = (\underline{d}_{i,l+1} + \lambda_i(\bar{d}_{i,l+1} - \underline{d}_{i,l+1}))^2, \quad i = 1, \dots, l,$$

onde $\lambda_i \in [0, 1]$ são novas variáveis no modelo que são usadas para representar as distâncias intervalares. Esta ideia é baseada no CIA apresentado no capítulo 2, que modela o dependência ou independência das operações aritméticas através da parâmetros λ_i . Neste problema os parâmetros λ_i são usados para codificar ambas as incertezas relacionado às distâncias entre x_i e x_{l+1} e também ao valores r_{l+1} . Uma instância i DGP com $|V| = n$ e $|E| = m$ gera um CIP (4.5-4.11) com $2l + 4$ variáveis e $7l + 1$ restrições, $l < m$.

Um resultado trivial, mas importante, relacionado ao novo modelo é o seguinte.

Teorema 4.3. *Seja $G = (V, E, d)$, $|V| = n$, uma instância do GDGP. Se $\{(x_i^{SWP}, r_i^{SWP}) : i = 1, \dots, n\}$ e $\{(x_i^{CIP}, r_i^{CIP}) : i = 1, \dots, n\}$ são as soluções obtidas para a sequências de SWP's e CIP's, então*

$$\sum_{i=1}^n r_i^{SWP} \leq \sum_{i=1}^n r_i^{CIP}.$$

Para os testes computacionais a estrutura 3D da nossa instância de molécula será definida em termos de suas *coordenadas internas* dadas pelos comprimentos

da ligações covalentes $d_{1,2}, \dots, d_{n-1,n}$, pelo ângulos covalentes $\theta_{1,3}, \dots, \theta_{n-2,n}$ (formado por três átomos consecutivos), e pelos ângulos de torção $\omega_{1,4}, \dots, \omega_{n-3,n}$ (formado por quatro átomos consecutivos). Cada ângulo de torção $\omega_{i-3,i}$ é o ângulo entre os planos definidos pelos átomos $i-3, i-2, i-1$ e $i-2, i-1, i$, respectivamente.

Agora fixamos os comprimentos das ligações covalentes (por exemplo, $d_{i-1,i} = 1.0$) e os valores dos ângulos covalentes (por exemplo, $\theta_{i-2,i} = 2.0$ radianos). Desta forma, exceto para rotações, translações e reflexões, uma estrutura molecular 3D é determinada pelos ângulos de torção $\omega_{1,4}, \dots, \omega_{n-3,n} \in [0, 2\pi]$. Uma maneira de proceder é escolher aleatoriamente valores para $\omega_{i-3,i} \in \{\frac{\pi}{3}, \pi, \frac{5\pi}{3}\}$.

Para determinar tais pares de pontos $\{i, j\}$, precisamos primeiro obter as coordenadas cartesianas a partir das coordenadas internas, como segue:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} = B_1 B_2 B_3 \cdots B_i \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

Onde

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1 & 0 & 0 & -d_{1,2} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$B_3 = \begin{bmatrix} -\cos \theta_{1,3} & -\sin \theta_{1,3} & 0 & -d_{2,3} \cos \theta_{1,3} \\ \sin \theta_{1,3} & -\cos \theta_{1,3} & 0 & d_{2,3} \sin \theta_{1,3} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

e

$$B_i = \begin{bmatrix} -\cos \theta_{i-2,i} & -\sin \theta_{i-2,i} & 0 & -d_{i-1,i} \cos \theta_{i-2,i} \\ \sin \theta_{i-2,i} \cos \omega_{i-3,i} & -\cos \theta_{i-2,i} \cos \omega_{i-3,i} & -\sin \omega_{i-3,i} & d_{i-1,i} \sin \theta_{i-2,i} \cos \omega_{i-3,i} \\ \sin \theta_{i-2,i} \sin \omega_{i-3,i} & -\cos \theta_{i-2,i} \sin \omega_{i-3,i} & \cos \omega_{i-3,i} & d_{i-1,i} \sin \theta_{i-2,i} \sin \omega_{i-3,i} \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

para $i = 4, \dots, n$. Para mais detalhes, veja [59].

Utilizamos o software AMPL com o solver Baron 17.4.1 em um notebook Lenovo, com 6 GB Ram e intel celeron 1.6 GHz, para resolver os três modelos apresentados neste artigo: o GDGP (4.3), o SWP (4.4) e o CIP (4.5-4.11).

Para cada $n = 10, 20, 30, 40, 50, 60$, geramos 10 instâncias aleatórias de acordo com o procedimento descrito acima. Como todos os três modelos requerem distâncias intervalares para todo $\{i, j\} \in E$, além de distâncias associadas aos pares $\{i, j\}$, onde $|j - i| \geq 3$, dado por

$$\underline{d}_{i,j} = d_{i,j} - \min\{0.3, 0.3d_{i,j}\},$$

$$\bar{d}_{i,j} = d_{i,j} + \min\{0.3, 0.3d_{i,j}\},$$

também definimos distâncias intervalares para $\{i, j\}$, onde $|j - i| < 3$, dadas por

$$\underline{d}_{i,j} = d_{i,j} - 0,15,$$

$$\bar{d}_{i,j} = d_{i,j} + 0,15.$$

As distâncias intervalares para o último caso são menores porque estão relacionadas a comprimentos de ligação e ângulos de ligação, considerados valores precisos em proteínas [56].

Paramos com $n = 60$ porque não foi encontrada solução para o GDGP nem para o SWP, e impusemos um limite de tempo de processamento de 1.800 segundos.

A tabela 4.1 mostra o número de problemas resolvidos para cada modelo. Vale notar que o modelo GDGP fornece mais soluções do que o modelo CIP somente quando $n = 20$ e $n = 40$. Em todos os casos, o modelo CIP fornece mais soluções do que o modelo SWP.

A tabela 4.2 dá a média da soma dos raios (o valor de a função objetivo do modelo GDGP). Como era de se esperar,

$$\sum_{i=1}^n r_i^{SWP} \leq \sum_{i=1}^n r_i^{CIP} \leq \sum_{i=1}^n r_i^{GDGP}$$

A tabela 4.3 mostra a média do tempo computacional, em segundos, considerando apenas as instâncias resolvidas por cada modelo. Também como esperado, SWP e os modelos CIP geram soluções muito mais rapidamente que o modelo GDGP.

O principal objetivo desses experimentos computacionais é ilustrar a vantagem do modelo CIP, em comparação com o modelo SWP. Além de fornecer mais soluções, o tempo computacional necessário para encontrar uma solução não aumenta exponencialmente como o modelo GDGP.

Tabela 4.1: Número de soluções encontradas

número de atomoss	Modelo SWP	Modelo CIP	Modelo GDGP
10	9	10	10
20	2	8	9
30	0	5	5
40	0	5	8
50	0	7	5
60	0	6	0

Tabela 4.2: Valor médio de $\sum_{i=1}^n r_i$

número de atomos	Modelo SWP	Modelo CIP	Modelo GDGP
10	0.2829	0.2940	0.4932
20	1.1076	1.1247	1.4700
30	-	1.578	2.0742
40	-	2.47586	2.922875
50	-	3.0793	3.6540
60	-	3.7069	-

Tabela 4.3: Tempo computacional médio em segundos

número de átomos	Modelo SWP	Modelo CIP	Modelo GDGP
10	0.4947	0.8185	30.7557
20	1.1601	2.7343	103.2811
30	-	6.1300	406.9412
40	-	6.755	537.3872
50	-	6.4374	1095.33386
60	-	9.6070	-

4.3 O problema de localização de sensores sob incerteza

Esta seção apresenta a abordagem que propusemos em [67]. A localização de sensores e/ou alvos tem grande importância em diversos setores da sociedade. Por exemplo, para monitoramento de desastres naturais, como alagamentos e deslizamentos. Também possui um papel destacado na indústria 4.0. De acordo com [69] uma das principais técnicas de medidas de localização de alvos utilizando uma rede de sensores sem fio é por meio de medidas *relativas a distâncias* (ou em inglês: *distance related measurements*).

As medidas relativas a distâncias consideram o tempo de propagação do sinal para determinar a distância entre os sensores sem fio e o alvo que deve ser localizado. As duas principais estratégias de medida de distância por tempo de propagação são:

- Propagação *uma-direção*: Considera o intervalo de tempo entre a emissão do sinal e o recebimento do sinal pelo alvo;
 - Necessária sincronização refinada entre transmissor e receptor (alto custo → baixa popularidade)
- Propagação *ida e volta*: Considera o tempo entre o envio do sinal pelo transmissor e a recepção do sinal proveniente de um segundo sensor;
 - Erro de tempo de propagação proveniente do atraso necessário para que o segundo sensor faça a recepção e transmissão do sinal (erro conhecido *a priori* por calibração)

Todas as metodologias para determinação da distância entre alvos e sensores sem fio baseadas no tempo de propagação do sinal são associadas a medidas de incerteza que devem ser tratadas para encontrar de forma efetiva as coordenadas de um determinado alvo.

Definição 4.3. *Sejam p_1, p_2, p_3 sensores sem fio instalados de forma não colinear, $d_{ij}, i, j \in \{1, 2, 3\}$ as distâncias conhecidas entre eles. Sejam $\bar{d}_{0j}, j \in \{1, 2, 3\}$ as distâncias aferidas entre os sensores e o alvo, com erros associados ϵ_j . O problema de localização consiste em determinar as coordenadas $x_0, y_0 \in \mathbb{R}$ do alvo de forma que $d_{0j}, j \in \{1, 2, 3\}$ seja **consistente** com as leituras realizadas.*

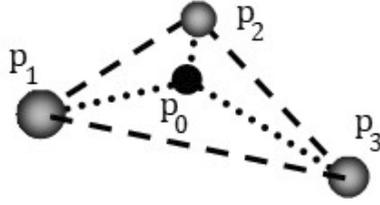


Figura 4.1: Exemplo de instalação não colinear dos sensores p_1, p_2, p_3 e do posicionamento do alvo p_0 em \mathbb{R}^2 .

[68] considera que a distância real do alvo aos sensores é da forma $d_{0j}^2 = \bar{d}_{0j}^2 + \epsilon_j$. Essa hipótese distancia o problema da realidade pois não é sensível à qualidade de leitura e transmissão de informação dos sensores, dificultando que haja aplicação real.

A vantagem teórica da hipótese de [68], caso ela seja satisfeita, é que existem A, b, c , dependentes de \bar{d}_{0j}, d_{ij} com $i, j \in \{1, 2, 3\}$ tais que $\epsilon^T A \epsilon + \epsilon^T b + c = 0$. Esse resultado é obtido através de manipulações algébricas aplicadas ao determinante de Cayley-Menger da sequência de pontos $p_0; p_1; p_2; p_3$. Esse resultado garante que é possível resolver (4.12) e é possível encontrar as distâncias reais entre os sensores e o alvo a partir dos valores ótimos de ϵ_i .

$$\min_{\epsilon_1, \epsilon_2, \epsilon_3} \epsilon_1^2 + \epsilon_2^2 + \epsilon_3^2 \quad (4.12)$$

$$S. a \quad \epsilon^T A \epsilon + \epsilon^T b + c = 0 \quad (4.13)$$

Com as distâncias reais obtidas, [68] utiliza a estratégia de triangulação para encontrar efetivamente as coordenadas do alvo.

Nós usamos o determinante de Cayley-Menger para obter as distâncias faltantes.

Definição 4.4. A *matriz de Cayley-Menger* de duas sequências de n pontos $\{p_1, \dots, p_n\}$ e $\{q_1, \dots, q_n\} \in \mathbb{R}^m$ é definida como:

$$M(p_1, \dots, p_n; q_1, \dots, q_n) = \begin{pmatrix} (d^2(p_i, q_j))_{n \times n} & 1 \\ 1 & 0 \end{pmatrix} \quad (4.14)$$

e o *bi-determinante de Cayley-Menger* destas duas sequências de pontos é definida como o determinante de M .

Consideramos a hipótese de que a distância real $j \in \{1, 2, 3\}$ é um número intervalar da forma:

$$d_{0j} = \bar{d}_{0j} \cdot (1 + (2\lambda_j - 1) \cdot \alpha), \lambda_j \in [0, 1]$$

, onde α é dado pelo método de aquisição da distância j .

Para que as distâncias desconhecidas $d_{0j}, j \in \{1, 2, 3\}$ sejam distâncias entre pontos em \mathbb{R}^2 é necessário e suficiente que $CM(p_0; p_1; p_2; p_3) = 0$ e que as desigualdades triangulares sejam respeitadas, pois as distâncias conhecidas foram tomadas de pontos que pertencem a \mathbb{R}^2 (ou seja, s ao conhecidas as localizações exatas de cada sensor âncora).

Desta forma, é necessário encontrar os intervalos máximos viáveis de cada distância real d_{0j} , $j \in \{1, 2, 3\}$ de forma que elas sejam consistentes com distâncias euclidianas entre pontos presentes em \mathbb{R}^2 . Para isso, é proposto o modelo (4.15), com solução ótima denotada por λ_i , $i \in \{1, 2, 3\}$.

$$\max_{\lambda_1, \lambda_2, \lambda_3} \quad (\lambda_1)^2 + (\lambda_2)^2 + (\lambda_3)^2 \quad (4.15)$$

$$S. a \quad CM(p_0, p_1, p_2, p_3) = 0 \quad (4.16)$$

$$\lambda_1, \lambda_2, \lambda_3 \in [0, 1] \quad (4.17)$$

Uma vez encontrados os intervalos máximos que as distâncias reais d_{0j} podem pertencer, para todo $j \in \{1, 2, 3\}$, resta encontrar as coordenadas $x_0, y_0 \in \mathbb{R}$ do alvo. Com esse intuito, o PGD pode ser formulado como o problema (4.18).

$$\min_{x, y, \lambda_1, \lambda_2, \lambda_3} \quad \sum_{i=1}^3 \left(\sqrt{(p_{ix} - x)^2 + (p_{iy} - y)^2} - (\bar{d}_{0i}(1 + (2\lambda_i - 1) \cdot \alpha)) \right)^2 \quad (4.18)$$

$$S. a \quad \lambda_i \in [0, \bar{\lambda}_i] \quad (4.19)$$

Se a função-objetivo for 0 na solução ótima, o alvo está localizado de forma efetiva. Caso contrário, os parâmetros relativo a localização do alvo não estão corretos.

Para determinar a eficiência da metodologia foram geradas 1.000 coordenadas de alvos aleatórios em que o alvo i tem coordenadas tais que:

$$(x_i, y_i) \in \{(x, y) \in \mathbb{R}^2 \mid x \in [-25, 70], y \in [-20, 20]\} \quad (4.20)$$

Foram gerados arquivos de dados de entrada utilizando MATLAB (*R2016a* Versão: 9.0.0.341360) para o modelo implementado em ambiente AMPL (foi utilizado *AMPL IDE* Versão: 3.1.0.201510231950). Os modelos matemáticos foram resolvidos utilizando o BARON (Versão 17.4.1 (2017.04.01)) e as saídas geradas pelo AMPL foram exportadas em arquivos .csv e carregados no MATLAB para processamento.

Os 1.000 alvos sintéticos gerados aleatoriamente.

4.4 O problema de otimização linear intervalar

O problema de otimização linear intervalar é o seguinte. Seja $[c] = [\underline{c}, \bar{c}]$ e $[b] = [\underline{b}, \bar{b}]$ vetores intervalares, e $[A] = [\underline{A}, \bar{A}]$, $\underline{A} \leq \bar{A}$, uma matriz intervalar $n \times m$. Aqui, a ordenação do vetor e da matriz \leq é feita por componentes. Estamos interessados em encontrar as soluções para o seguinte problema de otimização linear :

$$\begin{aligned} \min \quad & [c]x \\ \text{s.a.} \quad & [A][x] \leq [b] \end{aligned} \quad (4.21)$$

Uma pergunta que surge após esta definição é o que seria uma solução para este problema. Usando os conceitos definidos no capítulo 2, em especial o caso 4 de um sistema linear intervalar, [70] define o seguinte conjunto solução:

- O conjunto de soluções ótimas

$$f(A, b, c) := \min c^T x \text{ sujeito a } Ax \leq b$$

onde $A \in [A]$, $b \in [b]$, $c \in [c]$.

- $\underline{f} = \min f(A, b, c)$.
- $\bar{f} = \max f(A, b, c)$.

Hladik (2014) apresentou métodos exatos e de aproximação para o valor ótimo de melhor caso \underline{f} de programas lineares intervalares com restrição de desigualdade.

É possível encontrar o valor exato de \underline{f} na resolução de um problema de otimização quadrática:

Modelo 1

$$\begin{aligned} \underline{f} &= \min \sum_{j=1}^n \gamma_j x_j \\ \text{s. a:} \quad & \sum_{j=1}^n \alpha_{ij} x_j \leq \bar{b}_i \quad i = 1 \dots n \\ & \underline{a}_{ij} \leq \alpha_{ij} \leq \bar{a}_{ij} \quad i = 1 \dots n, j = 1 \dots n. \\ & \underline{c}_j \leq \gamma_j \leq \bar{c}_j \quad j = 1 \dots n. \end{aligned} \quad (4.22)$$

where $[A] = [\underline{a}_{ij}, \bar{a}_{ij}]$, $[b] = [\underline{b}_i, \bar{b}_i]$ e $[c] = [\underline{c}_j, \bar{c}_j]$.

Exemplo 4.1. Considere o exemplo apresentado em [70]:

$$\begin{aligned} \min \quad & [c]x \\ \text{s.a:} \quad & [A]x \leq [b] \end{aligned} \quad (4.23)$$

onde

$$A = \begin{pmatrix} -[4, 5] & -[2, 3] \\ [4, 5] & -[1, 2] \\ [4, 5] & -[1, 2] \\ [2, 3] & [5, 6] \end{pmatrix}.$$

$$b = \begin{pmatrix} -[11, 12] \\ [26, 28] \\ [43, 45] \end{pmatrix}$$

$$c = \begin{pmatrix} [2, 3] \\ [6, 7] \end{pmatrix}$$

Hladik [70] propôs uma heurística para encontrar o limite superior de \underline{f} , encontrando $\underline{f}_L = -58.39$, e o limite inferior de \underline{f} , encontrando $\underline{f}_U = -16.61$. Ou seja, neste exemplo $-58.39 \leq \underline{f} \leq -16.615$

Resolvemos o seguinte problema de otimização quadrática:

$$\begin{aligned}
\min \quad & \gamma_1 x_1 + \gamma_2 x_2 \\
\text{s.a :} \quad & \alpha_{11} x_1 + \alpha_{12} x_2 \leq -11 \\
& \alpha_{21} x_1 + \alpha_{22} x_2 \leq 28 \\
& \alpha_{31} x_1 + \alpha_{32} x_2 \leq 45 \\
& -5 \leq \alpha_{11} \leq -4 \\
& -3 \leq \alpha_{12} \leq -2 \\
& 4 \leq \alpha_{21} \leq 5 \\
& -2 \leq \alpha_{22} \leq -1 \\
& 2 \leq \alpha_{31} \leq 3 \\
& 5 \leq \alpha_{32} \leq 6 \\
& 2 \leq \alpha_{31} \leq 3 \\
& 2 \leq \gamma_1 \leq 3 \\
& 6 \leq \gamma_2 \leq 7
\end{aligned} \tag{4.24}$$

Obtivemos a solução exata $f = -41.38461538$ em 0.078 segundos usando o solver *Knitro* em um Laptop com 16GB de RAM.

Fica em aberto como obter o valor de \bar{f} - veja os exercícios no final deste capítulo.

4.5 O problema de corte de estoque sob incerteza

O Brasil enfrenta um período de desindustrialização, ou seja, uma redução da sua capacidade industrial. Entre os pesquisadores e entidades da área há diferentes avaliações das causas e do que deve ser feito. Contudo, é possível afirmar que sem uma modernização do modo de produção o país dificilmente conseguirá reverter a situação. Esse processo de modernização deve acompanhar o que boa parte do mundo tem chamado de quarta revolução industrial, ou indústria 4.0 (nomenclatura de origem alemã) ou manufatura avançada (nomenclatura de origem americana). Nesse sentido propomos uma atuação em prol do desenvolvimento de novos modelos matemáticos e métodos computacionais no contexto da indústria 4.0 para os problemas de corte.

O problema de corte (clássico) objetiva encontrar a melhor forma de se obter itens demandados a partir do corte de peças maiores, minimizando algum tipo de custo ou maximizando o lucro. Assim, um problema de corte pode ser decompostos em dois sub-problemas distintos: gerar os padrões de corte e combiná-los da melhor forma possível, a fim de atingir a demanda exigida.

Existem vários trabalhos que abordam o problema de redução do número de padrões de corte na solução ótima, objetivo importante em diversas indústrias, principalmente quando há uma grande demanda para ser atendida em pouco tempo: [71], [72],[73], [74], [75].

Liu et al. [76] apresentaram, recentemente e de forma pioneira, o problema de corte de estoque no contexto da Indústria/Internet das Coisas 4.0. Contudo esse trabalho, único publicado até o momento segundo o nosso conhecimento, apresenta apenas uma heurística, sem considerar todos os aspectos envolvidos no processo descrito pelos próprios autores.

O problema de corte e estoque (clássico) consiste em cortar peças maiores W em menores w_i com objetivo de atender uma demanda d_i onde $i = 1..n$.

Cada combinação diferente de itens que serão cortados de uma bobina específica é chamado de padrão de corte e a cada troca de padrão de corte temos um custo para reconfiguração da máquina chamado *setup*.

Quando temos que atender uma grande demanda de itens em um curto espaço de tempo setup passa a ter um grande custo, visto que a máquina fica parada durante esse período de tempo. Assim, devemos encontrar um equilíbrio entre número de *setups* e a quantidade de objetos processados, já que são objetivos conflitantes.

Apresentamos a seguir o modelo matemático para o problema de corte de estoque:

$$\begin{aligned}
 \text{Minimize} \quad & c_1 \sum_{j=1}^n x_j + c_2 \sum_{j=1}^n y_j \\
 \text{s.a:} \quad & \sum_{j=1}^n a_{ij}x_j \geq d_i, \quad i = 1, \dots, m. \\
 & x_j \leq M y_j, \quad j = 1, \dots, n. \\
 & x_j \in \mathbb{N}, \quad j = 1, \dots, n. \\
 & y_j \in \{0, 1\} \quad j = 1, \dots, n.
 \end{aligned} \tag{4.25}$$

onde c_1 é o custo de cada bobina e c_2 o custo de setup.

O problema de corte de estoque (Cutting Stock Problem - CSP) consiste em encontrar a melhor forma de produzir m itens de tamanhos $w_1 < w_2 < \dots < w_m$ e demandas d_1, d_2, \dots, d_m , respectivamente, a partir de cortes de objetos (matéria-prima) de tamanho $W > w_m$, por hipótese em quantidades ilimitadas em estoque. Este problema é classificado como Problema de Corte de Estoque com Único Tamanho em Estoque (Single Stock Size Cutting Stock Problem (1D-SSCSP)).

O problema de corte de estoque, considerando o objetivo de minimizar o número de objetos processados e o atendimento exato da demanda, sem excesso de produção, pode ser modelado da seguinte forma:

$$\left\{ \begin{array}{l} \text{Minimizar}_{a,x} \quad \sum_{j=1}^n x_j \\ \text{sujeito a} \quad \sum_{j=1}^n a_{ij}x_j = d_i, \quad i = 1, \dots, m. \\ \\ a_{ij}, x_j \in \mathbb{N}, \quad i = 1, \dots, m, \quad j = 1, \dots, n. \end{array} \right.$$

onde

- $a_j \in \mathbb{N}^m$, $j = 1, \dots, n$ são variáveis que representam os padrões de corte, ou seja, a forma de cortar os objetos em estoque;
- n é o número de padrões de corte viáveis;
- x_j são variáveis que representam o número de objetos que serão cortados com o padrão j .

Como este modelo é não linear (pois temos produtos de variáveis nas restrições), o CSP também pode ser formulado da seguinte forma:

$$\left\{ \begin{array}{l} \text{Minimizar}_x \quad \sum_{j=1}^n x_j \\ \text{sujeito a} \quad \sum_{j=1}^n a_{ij}x_j = d_i, \quad i = 1, \dots, m. \\ \\ a_{ij}, x_j \in \mathbb{N}, \quad i = 1, \dots, m, \quad j = 1, \dots, n. \end{array} \right.$$

onde

- $a_j \in N^m$, $j = 1, \dots, n$ são os padrões de corte, ou seja, a forma de cortar os objetos em estoque (parâmetros definidos anteriormente/pré processamento e/ou por geração de colunas);
- n o número de padrões de corte gerados;
- x_j são variáveis que representam o número de objetos que serão cortados com o padrão j .

Temos agora um problema de otimização linear inteiro, com um grande número de variáveis em geral, dado o grande número de padrões de cortes viáveis em problemas reais.

Vale salientar que, como estamos trabalhando com restrições de igualdade, ou seja, a demanda deve ser atendida exatamente, é equivalente minimizar o número de objetos processados e minimizar o desperdício total. Contudo, quando se trabalha com restrições de desigualdade, como é comum na literatura da área, ou seja, quando é permitido produzir mais do que a demanda, minimizar o número de objetos processados não é equivalente à minimizar o desperdício total. Goulamins (2020) [77] chama especial atenção a este fato, mostrando como soluções apresentadas na literatura, em revistas de alto impacto, não são as melhores para uma situação real da indústria.

Essa abordagem enfrenta um grande problema para ser útil, de fato, na prática: na maioria dos casos não é possível atribuir um custo monetário para o tempo de preparação da máquina. A importância do setup depende sobretudo da velocidade de produção necessária para atender a demanda dentro do prazo acordado com os clientes. Dessa forma, o tempo de preparação da máquina, chamado na área de setup, é variável e pode ser melhor gerenciada com base na experiência do gerente de produção. Nesse sentido, a abordagem por meio de um problema multiobjetivo é, na nossa visão, mais apropriado para a indústria. A literatura contém alguns trabalhos que usam métodos heurísticos para gerar boas soluções para o problema de corte com múltiplos objetivos [75, 72]. Alguns trabalhos recentes trazem análises teóricas e métodos exatos [78, 81, 73], mas nenhum deles consideram a incerteza na demanda.

Em algumas indústrias, como a de móveis, é importante minimizar o número de ciclos de serra. Seja CC o número máximo de objetos que podem ser cortados ao mesmo tempo. Considere o seguinte modelo matemático:

$$\begin{aligned}
 \text{Min} \quad & c_1 \sum_{j=1}^n x_j + c_2 \sum_{j=1}^n z_j \\
 \text{s.a:} \quad & \sum_{j=1}^n a_{ij} x_j \geq d_i, \quad i = 1, \dots, m. \\
 & x_j \leq CC z_j, \quad j = 1, \dots, n. \\
 & x_j \in N, \quad j = 1, \dots, n. \\
 & z_j \in \{0, 1\}, \quad j = 1, \dots, n.
 \end{aligned} \tag{4.26}$$

Considerando a abordagem multiobjetiva mais adequada para a indústria, nós estudamos nesse trabalho os seguintes Problemas de Corte com Múltiplos Objetivos (PCMo):

$$(PCMo1) \left\{ \begin{array}{l} x^* = \min_z f(x) = (f_1(z), f_2(z)) \\ \text{s. a: } z \in A \end{array} \right. \tag{4.27}$$

onde

- $f_1(x) = \sum_{j=1}^n x_j \equiv$ número de objetos processados.
- $f_2(y) = \sum_{j=1}^n y_j \equiv$ setup.
- $A = \{z \in Z^n, x \geq 0 : \sum_{j=1}^n a_{ij}x_j \geq d_i, i = 1, \dots, m\}, x_j \leq My_j$ para $j = 1, \dots, n$,
 $x_j \in Z$ para $j = 1, \dots, n, y_j \in \{0, 1\}, \mathbb{Z}$ é o conjunto dos números inteiros}.

Salles-Neto et al. [82] fazem uma análise teórica sobre este problema, também sem considerar a incerteza na demanda.

Como já comentado, existem poucos trabalhos na literatura que trabalham com o problema de corte multiobjetivo (PCMo). Recentemente, Arana and Salles Neto [81] apresentaram condições de otimalidade para esse problema. Aliano et al. [78] apresentam alguns métodos para obtenção de soluções eficientes para o problema considerando como funções-objetivo o número de objetos e o setup. Na nossa opinião trata-se do estado da arte nesse problema. Contudo, esses trabalhos não tratam da incerteza da demanda, inerente ao processo e que, com o processo de revolução industrial em curso, será cada vez mais importante. Esses trabalhos também não abordam a função-objetivo do número de ciclos de serra.

Como comentado anteriormente, na vida real a demanda (vetor d) não é conhecida com precisão pelas indústrias que trabalham com cortes em sua linha de produção (papel, alumínio...). Existem muitos poucos artigos publicados que tratam o problema de corte com incerteza [79, 80]. Entretanto, nenhum usa otimização multiobjetivo.

Podemos considerar os seguintes modelos matemáticos de PCMo sob incerteza (PCMoI):

$$(PCMoI) \begin{cases} w^* = \min_x f(w) = (f_1(w), f_2(w), f_3(w), f_4(w)) \\ \text{s. a: } w \in W \end{cases} \quad (4.28)$$

onde

- $f_1(x) = \sum_{j=1}^n x_j \equiv$ número de objetos processados.
- $f_2(x) = \sum_{j=1}^n (y_j) \equiv$ setup.
- $f_3(x) = \sum_{s=1}^K \sum_{i=1}^m p_s(co_i o_{is}) \equiv$ o valor esperado do custo de sobreprodução.
- $f_4(x) = \sum_{s=1}^K \sum_{i=1}^m p_s(cu_i u_{is}) \equiv$ o valor do custo esperado se subprodução.
- $W = \{(x, y, o, u) \text{ tal que } x \geq 0 : \sum_{j=1}^n a_{ij}x_j - o_{is} + u_{is} = d_{is}, i = 1, \dots, m, s = 1, \dots, K\}$ (ou seja, temos K cenários), $x_j \leq My_j$ para $j = 1, \dots, n, x_j \in Z$ para $j = 1, \dots, n, y_j \in \{0, 1\}, \mathbb{Z}$ é o conjunto dos números inteiros}.

$$(PCMoI2) \begin{cases} w^* = \min_x f(w) = (f_1(w), f_2(w), f_3(w), f_4(w)) \\ \text{s. t. } w \in W \end{cases} \quad (4.29)$$

onde X , $f_1(x)$, $f_3(x)$ e $f_4(x)$ são como acima definido e $f_2(x) = \sum_{j=1}^n (z_j)$ é o número de ciclos de serra.

Proposição 4.1. *Se $w^* = (x^*, y^*, o^*, u^*)$ é uma solução eficiente para o PCMoI1 ou para o PCMoI2 então $u_{is}^* \times o_{is}^* = 0$ para todo $i = 1, \dots, n$, $s = 1, \dots, k$.*

Demonstração:

Seja $w^* = (x^*, y^*, o^*, u^*)$ uma solução eficiente para o PCMoI1 ou PCMoI2. Suponha que exista um $l \in \{1, \dots, n\}$ e um $p \in \{1, \dots, K\}$ tal que $u_{lp}^* o_{lp}^* > 0$. Então $u_{lp}^* > 0$ e $o_{lp}^* > 0$. Suponha, sem perda de generalidade, que $u_{lp}^* - o_{lp}^* > 0$. Seja $u \in R^{n+K}$ e $v \in R^{n+K}$ tal que $u_{is} = u_{is}^*$ e $v_{is} = v_{is}^*$ para todo $i \neq p$ e para todo $s \neq p$, $u_{lp} = u_{lp}^* - o_{lp}^*$ e $o_{lp} = 0$. Seja $w = (x^*, y^*, u, o)$. Logo, $w \in W$ e $f(w) < f(w^*)$, o que é uma contradição.

Proposição 4.2. *Seja $D = \max_{i,s} d_{is}$, $i = 1, \dots, n$ e $j = 1, \dots, n$. Se $CC \geq D$ então se $w^* = (x^*, y^*, o^*, u^*)$ é uma solução eficiente para o PCMoI2 então w^* é uma solução eficiente para o PCMoI1.*

Demonstração: Seja $D = \max_{i,s} d_{is}$, $i = 1, \dots, n$ e $j = 1, \dots, n$. Seja $w^* = (x^*, y^*, o^*, u^*)$ uma solução eficiente para o PCMoI2. Assim, $x_j \leq D$ para todo $j = 1, \dots, n$. De fato, se existe um l tal que $x_l^* > D$ então $w = (x, y^*, o^*, u^*)$, onde $x_i = x_i^*$ para todo $i \neq l$ e $x_l = d$ é tal que $f(w) < f(w^*)$, o que é uma contradição. Por hipótese, $CC \geq D \geq x_j$ para todo j . Então, $z_j = 0$ se $x_j = 0$ e $z_j = 1$ se $x_j > 0$. Isso implica que $\hat{w} = (x^*, y^*, o^*, u^*)$, onde $y^* = w^*$, é uma solução eficiente para o PCMoI1.

Exemplo 3.1.

Considere o seguinte exemplo:

- $W = 20$ - tamanho dos rolos em estoque (objetos); $m = 4$ - número de itens; $p = 3$ - número de cenários; $p_1 = p_2 = p_3 = 1/3$ as probabilidades de ocorrência de cada cenário; $w = (w_1, w_2, w_3, w_4) = (10, 6, 5, 4)$, tamanho dos itens demandados;
- $d_1 = (d_{11}, d_{12}, d_{13}, d_{14}) = (600, 153, 300, 15)$ vetor demanda no cenário 1;
 $d_2 = (d_{21}, d_{22}, d_{23}, d_{24}) = (400, 200, 300, 30)$ vetor demanda no cenário 2;
 $d_3 = (d_{31}, d_{32}, d_{33}, d_{34}) = (600, 200, 400, 30)$ vetor demanda no cenário 3.

Para os testes computacionais todos os padrões de corte viáveis foram gerados. Para obter as soluções eficientes, nós usamos o método ϵ -restrito [51]. Assim, o seguinte problema de otimização foi resolvido para cada $\epsilon = (\epsilon_1, \epsilon_2, \epsilon_3)$:

$$(CSP1_\epsilon) \begin{cases} \operatorname{argmin}_x f_1(w) \\ \text{s. t. } w \in W \\ f_i(w) \leq \epsilon_i \quad \forall i = 1, 2, 3, 4. \end{cases} \quad (4.30)$$

Os seguintes valores para ϵ foram usados: $\epsilon = (4, 100, 100)$, $\epsilon = (3, 100, 100)$, $\epsilon = (2, 100, 100)$, $\epsilon = (4, 100, 0)$, $\epsilon = (3, 100, 0)$, $\epsilon = (2, 100, 0)$, $\epsilon = (4, 0, 100)$, $\epsilon =$

$(3, 0, 100), \epsilon = (2, 0, 100), \epsilon = (4, 1000, 100), \epsilon = (3, 1000, 100), \epsilon = (2, 1000, 100),$
 $\epsilon = (4, 50, 50), \epsilon = (3, 50, 50), \epsilon = (2, 50, 50).$

O problema foi modelado no aplicativo *AMPL* (apresentado detalhadamente no capítulo 5). Para resolver cada problema de otimização foi usado o solver *Gurobi* (também apresentado detalhadamente no capítulo 5). Cada problema de otimização foi resolvido em um tempo computacional menor que um segundo em um Laptop Lenovo, 6 MB Ram, intel celeron processor 1.6 GHz, 64 bit, Windows Home 10. Para cada vetor ϵ *solver* encontrou a solução ótima do respectivo problema de otimização.

As soluções eficientes são mostradas na tabela 4.4.

Tabela 4.4: Soluções eficientes

f1	f2	f3	f4
422	4	1	100
432	3	0	100
447	4	1	50
450	2	170	100
457	3	1	50
472	4	1	0
481	4	0	0
500	3	70	0

Vale observar que essas soluções permitiriam um gerente de produção decidir, com base em sua experiência, qual o melhor plano de produção para aquele momento.

4.6 Finanças e otimização sob incerteza

Os modelos financeiros geralmente são modelos de otimização. Um dos desafios é quantificar o risco.

Risco e Retorno da Carteira Suponha que temos estimativas para:

- Retorno esperado para investimentos de N

$$\mu_n, \quad n \in \{1, \dots, N\} = \Pi \text{ portfólio}$$

- Variação nos N investimentos

$$\sigma_n^2, \quad n \in \{1, \dots, N\}$$

- Correlação nos N investimentos

$$\rho_{i,j} \quad i, j \in \{1, \dots, N\}$$

- Desvio padrão nos N investimentos

$$\sigma_n \quad n \in \{1, \dots, N\}$$

- A média de uma carteira é

$$\mu_P = \sum_{i \in P \subseteq \Pi} w_i \mu_i$$

$$w_i \geq 0, \sum_{i=1}^N w_i = 1$$

Onde

$$w_i > 0 \text{ se } i \in P,$$

$$w_i = 0 \text{ se } i \notin P.$$

- A variância da carteira é

$$\sigma_P^2 = \sum_{i \in P \subseteq \Pi} \sum_{j \in P \subseteq \Pi} w_i w_j \sigma_i \sigma_j \rho_{i,j}$$

$$= \sum_{i \in P \subseteq \Pi} \sum_{j \in P \subseteq \Pi} w_i w_j \sigma_{ij}$$

Onde

$$\sigma_{ij} = \sigma_i \sigma_j \rho_{i,j}$$

Markowitz em seu modelo inicial usou o desvio padrão, σ , como medida de risco, uma vez que é expresso nas mesmas unidades que a média, de modo que os eixos terão as mesmas unidades quando plotamos o que é chamado de *fronteira de eficiência*.

Definindo a eficiência de Markowitz

- Uma carteira P^* é *MV eficiente* se tiver o menor risco para um determinado nível de retorno da carteira. Isso é,

$$\mu_P = \mu_{P^*} \Rightarrow \sigma_P^2 \geq \sigma_{P^*}^2$$

- Uma carteira P^* é *MV eficiente* se tiver máximo retorno esperado para um determinado risco da carteira. Isso é,

$$\sigma_P^2 = \sigma_{P^*}^2 \Rightarrow \mu_P \leq \mu_{P^*}$$

Ou seja, uma carteira é eficiente se é solução eficiente para o problema de otimização bi-objetivo, onde os objetivos, conflitantes, são o retorno esperado e o risco, conforme definição apresentada no capítulo 3.

Restrições de otimização

- Investir uma unidade de ativos (dinheiro) e nenhum investimento negativo (empréstimo). Isso é,

$$w_i \geq 0, \sum_{i=1}^N w_i = 1$$

- A condição de orçamento é uma desigualdade linear. Ou seja, a quantidade máxima que há para comprar, $\$B$, é a restrição. Tomamos $\$B$ como uma unidade. A restrição acima nos obriga a gastar o valor orçado.
- Nenhuma venda a descoberto ou gasto deficitário que se reflita na restrição de não negatividade.

Algoritmos Computacionais O que o modelo de Markowitz acaba sendo a partir da nomenclatura da otimização matemática é um problema de *otimização quadrática*. As restrições são as dadas acima e são lineares.

O modelo 1 é :

$$\max z = -\sigma_P^2 + 2\tau\mu_P = - \sum_{i \in P \subseteq \Pi} \sum_{j \in P \subseteq \Pi} w_i w_j \sigma_{ij} + 2\tau \sum_{i \in P \subseteq \Pi} w_i \mu_{ei}$$

sujeito a:

$$\sum_{i=1}^N w_i = 1$$

$$w_i \geq 0, \tau \geq 0 \text{ (um peso que influencia como a média afeta o ideal)}$$

O modelo 2 é:

$$\min z = \sigma_P^2 + \mu_P = \sum_{i \in P \subseteq \Pi} \sum_{j \in P \subseteq \Pi} w_i w_j \sigma_{ij}$$

sujeito a:

$$w_P^T \mu \geq \mu^*$$

$$1^T w = \sum_{i=1}^N w_i = 1$$

$$w_i \geq 0$$

4.7 Aplicações ao enfrentamento da covid-19

O grupo de pesquisa operacional da Unifesp desenvolveu em 2020 dois aplicativos para auxiliar o enfrentamento à covid-19. Um deles foi o software sala planejada, que a partir das dimensões de uma sala de aula e do distanciamento mínimo entre os alunos retornava o layout ótimo da sala. Este aplicativo foi utilizado por diversas escolas e universidades do país, além de ter sido usado pelo INEP no planejamento de salas do ENEM e do ENAD em 2021 e 2022. Para conhecer a modelagem matemática desenvolvida veja [83]. Neste aplicativo não temos incerteza, pois as dimensões de salas de aula e do distanciamento social são dados de entrada informados pelas autoridades .

O outro aplicativo visava auxiliar hospitais e secretarias de saúde na previsão de leitos de UTI em um futuro próximo. Este foi o foco do desenvolvimento da solução apresentada em [84] e intitulada Forecast UTI, que foi utilizada por cerca de 30 hospitais no país. Este problema tem uma incerteza inerente, pois não é possível prever com precisão o número de leitos de UTI ao longo de uma pandemia. Por outro lado, é gravíssima a situação de não haver leito de UTI para um paciente que precisa. Infelizmente vimos casos como este no Brasil e em outros países. O que temos à disposição são dados passados. A partir destes dados, como podemos fazer uma boa previsão, otimizando os recursos?

O software Forecast UTI trabalha com os seguintes dados históricos de pacientes: data de nascimento, leito de UTI, data de internação, data de saída, CID de entrada, CID de saída, situação do paciente (alta, óbito, internado). Como você modelaria este problema? - veja o exercício 3 deste capítulo.



Figura 4.2: Interface do aplicativo web sala planejada

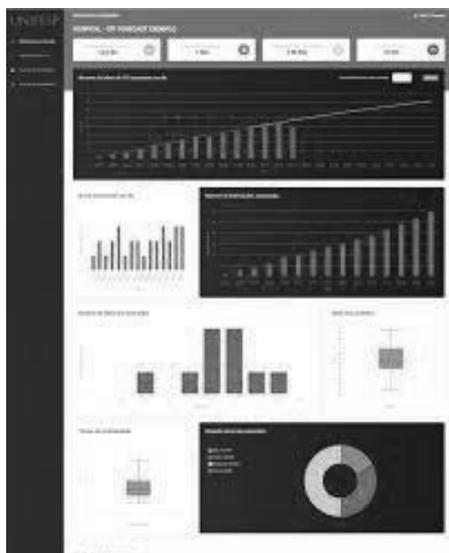


Figura 4.3: Interface do aplicativo web sala planejada

4.8 Resumo do capítulo

Neste capítulo apresentamos abordagens e modelos matemáticos para seis problemas reais e importantes de diferentes áreas, da medicina ao planejamento da produção. Acreditamos que estas abordagens podem ser aplicadas em outros problemas reais, sejam de otimização contínua ou inteira, ou mesmo problemas de otimização mista.

4.9 Exercícios

Nota: Os exercícios deste capítulo poderão ser resolvidos computacionalmente utilizando as ferramentas apresentadas no capítulo 5. Assim, neste capítulo espera-se a formulação dos modelos matemáticos.

- (1) Encontre uma estimativa para \bar{f} no exemplo da seção 4.4.
- (2) Considere uma empresa que possui 402 empregados, distribuídos em 30 pontos na cidade, e precisa organizar o transporte de ida e volta destes empregados e forma a minimizar seu custo mensal. A empresa pode alugar até 10 ônibus com 48 lugares, doravante ônibus 1, e até 30 micro-ônibus com 16 lugares, doravante ônibus 2. O ônibus 1 tem custo fixo mensal que varia de R\$15000,00 a R\$18000,00 e o 2 de R\$4000,00 a R\$8000,00. O custo por km rodado do ônibus 1 varia de R\$3,20 a R\$4,00 e do ônibus 2 é R\$2,90 a R\$3,90. Os demais dados estão disponíveis em <https://bit.ly/busproblem>. Os empregados vão e voltam no mesmo ônibus e do mesmo ponto. Modele o problema.
- (3) Considere uma cidade que possui dois hospitais com leitos de UTI, hospitais A e B, que possuem os dados mostrados na tabela 4.5 (estes dados são fictícios). Modele matematicamente o problema de forma a otimizar a disponibilização do número de leitos de UTI nesta cidade. Mais especificamente preveja quantos leitos de UTI seriam necessários ao longo do tempo - 15, 30, 60 dias.
- (3) Um fabricante sabe que ela/ele deve fornecer 5, 6, 8 e 6 itens respectivamente nos próximos quatro meses. O custo de produção de um item é 1, 4, 2 e 4 unidades (reais), respectivamente durante esses meses. Pode ser preferível não produzir os itens durante o primeiro mês quando a produção é mais barata devido à condição adicional de que há um custo de armazenamento de 1 unidade (reais) por mês por item, e também devido ao fato de que a produção em cada mês deve ser igual ou superior à do mês anterior. Um aumento na produção custa 2 unidades (reais) por item. Como deve ser a produção em cada mês? Modele o problema para otimizar a produção. Considere que pode haver variação na demanda e em cada um dos custos de $\pm 1\%$. Modele o problema com esta condição.
- (4) Uma companhia petrolífera deseja canalizar petróleo de uma fonte, S , para um "tanque" T , através de uma rede de tubos conforme a figura 4.4. Cada ligação entre duas junções, digamos (i, j) , tem capacidade c_{ij} . O que é o fluxo máximo de S para T ? Modele o problema. Considere que pode haver variações de $\pm 1\%$ em cada capacidade. Como ficaria o modelo?

Tabela 4.5: Dados de pacientes na UTI com covid-19 nos hospitais A e B

Data	Internados - A	Altas - A	Óbitos - A	Internados B	Altas B	Óbitos - B
11/04/2020	47	3	0	1	0	0
12/04/2020	61	3	1	13	0	0
13/04/2020	61	5	0	54	1	0
14/04/2020	76	1	0	54	1	0
15/04/2020	86	6	0	50	1	0
16/04/2020	64	11	0	54	22	0
17/04/2020	73	0	0	74	7	0
18/04/2020	79	0	0	92	28	0
19/04/2020	75	21	0	122	13	1
20/04/2020	76	5	0	144	23	0
21/04/2020	81	3	0	152	28	0
22/04/2020	91	1	0	164	20	1
23/04/2020	101	11	0	162	27	0
24/04/2020	112	5	0	210	20	0
25/04/2020	116	15	0	267	27	0
26/04/2020	123	8	0	335	35	0
27/04/2020	116	19	0	352	51	0
28/04/2020	138	18	0	372	42	2
29/04/2020	139	20	0	397	54	2

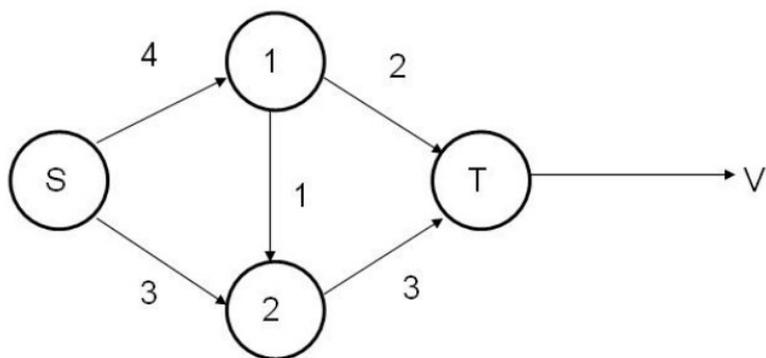


Figura 4.4: Entrada dos arquivos

Capítulo 5

Implementação e experimentos computacionais

5.1 Introdução

Apresentamos neste capítulo um pequeno tutorial de como usar algumas ferramentas computacionais para obter soluções de problemas de otimização linear, inteira, não-linear e também mista. Optamos por apresentar a linguagem Python, por sua praticidade e também pelo fato de ser uma das linguagens mais utilizadas por empresas na área de *analytics* e ciência de dados. Também apresentamos o software *AMPL*, muito utilizado também por quem trabalha com modelos de otimização.

5.2 Como usar o Python em otimização

Para resolver problemas de otimização linear, otimização inteira ou otimização não-linear temos diversas ferramentas gratuitas e eficientes disponíveis em Python. Apresentamos algumas com exemplos nas subseções seguintes.

5.2.1 PULP

PULP é um solver capaz de resolver problemas de otimização linear e otimização linear inteira. A definição das variáveis, função-objetivo e restrições são bem intuitivas e simples de serem declaradas.

Para clarear as ideias vamos considerar o clássico problema da mochila: considere um alpinista que possui uma mochila com capacidade K , n itens, cada um com utilidade u_i e peso m_i tal que $\sum_{i=1}^n m_i > K$. Ele deseja maximizar a utilidade da mochila.

Podemos modelar o problema da seguinte forma:

$$\begin{aligned} \text{(PM)} \quad & \text{Min} && \sum_{i=1}^n u_i x_i \\ & \text{s. a:} && \sum_{i=1}^n m_i x_i \leq K \\ & && x_i \in \{0, 1\} \quad i = 1, \dots, n. \end{aligned}$$

Considere que temos um problema da mochila com $n = 10$, $K = 15$ e os seguintes pares de utilidade e massa:



Figura 5.1: Janela do colab google

$(10, 2), (12, 3), (5, 1), (14, 4), (18, 8), (7, 2), (11, 4), (19, 9), (2, 1.8), (1, 0.5)$

Recomendamos usar o ambiente colab da google, que é gratuito, não precisa fazer download e é bem simples de usar:

<https://colab.research.google.com/> .

A figura 5.1 mostra o ambiente colab e inicialização do PULP descrita a seguir. Inicialmente precisamos fazer a instalação e inicialização do PULP:

```
%pip install pulp
import pulp
from pulp import *
import time
```

Podemos agora entrar com os dados do problema:

```
K=15
n=10
itens=(10,2),(12,3),(5,1),(14,4),(18,8),
(7,2),(11,4),(19,9),(2,1.8),(1,0.5)
```

Vamos agora definir as variáveis de decisão:

```
x = pulp.LpVariable.dicts('item', range(n), lowBound = 0, upBound = 1, cat = 'Integer')
```

Também poderíamos ter definido a variável como binária colocando *cat='binary'* e retirando os limites inferiores e superiores.

Podemos agora definir o problema:

```
problem = LpProblem("Problema da Mochila", LpMaximize)
```

Assim, nomeamos o problema de otimização linear e o definimos como de otimização.

Vamos agora definir a função-objetivo:

```
problem += lpSum([ x[i] * (itens[i])[0] for i in range(n) ]),
"Objetivo: Maximizar a utilidade"
```

Aqui é importante observar que o Python começa a contar do zero, e não do um. Assim, é preciso ter atenção a este fato quando transcrevemos o modelo matemático no PuLP-Python.

Agora adicionamos ao problema a restrição de capacidade da mochila:

```
problem += lpSum([ x[i] * (itens[i])[1] for i in range(n) ]) <=
K, "Restrição: capacidade da mochila"
```

Agora basta solicitar a resolução do problema:

```
start_time = time.time() problem.solve()
```

Problema resolvido vamos solicitar a impressão do resultado:

```
print("Status:", LpStatus[problem.status])
for v in problem.variables():
    print(v.name, "-", v.varValue)
print("Utilidade total = ", value(problem.objective))
```

Note que para imprimir o valor ótimo de cada variável foi usada uma estrutura `for` e indentação. No Python a indentação é fundamental e define, no caso do `for` neste caso, os comandos que serão executados neste *loop*.

Vamos considerar agora um exemplo de problema de otimização linear contínuo.

Exemplo 5.1. *João possui uma pequena fazenda de 45 hectares e deseja plantar milho e feijão. Cada hectare de milho gera um lucro de R\$200,00 e cada hectare de feijão retorna R\$300,00 de lucro. O número de empregados e fertilizantes necessários para cada hectare são descritos na tabela abaixo. Considerando que João pode contar com 100 empregados e 120 toneladas de fertilizantes, como ele pode maximizar seus lucros?*

Tabela 5.1: Requisitos do sítio

	Milho	Feijão
Empregados	3	2
Fertilizantes	2 ton	4 ton

Este problema pode ser modelado da seguinte forma:

$$\begin{aligned}
 \text{(PM)} \quad & \text{Max} \quad 200x_1 + 300x_2 \\
 \text{s. a:} \quad & 3x_1 + 2x_2 \leq 100 \\
 & 2x_1 + 4x_2 \leq 120 \\
 & x_1 + x_2 \leq 45 \\
 & x_i \geq 0 \quad i = 1, 2
 \end{aligned}$$

onde x_1 é a quantidade em hectares plantados com milho e x_2 é a quantidade em hectares plantados com feijão.

Seguindo os mesmos procedimentos adotados no problema da mochila:

1. Instalação/importação do PuLP:

```
%pip install pulp
import pulp
```

2. Definindo o problema como sendo de maximização:

```
prob = pulp.LpProblem('Problema do Sitio',
pulp.LpMaximize)
```

3. Definindo as variáveis de decisão

```
x1 = pulp.LpVariable('Milho', lowBound=0,
cat='Continuous')

x2 = pulp.LpVariable('Feijão', lowBound=0,
cat='Continuous')
```

4. Definindo a função Objetivo

```
Lucro=200*x1+300*x2
prob += Lucro
```

5. Adicionando as restrições:

```
Empregados=3*x1+2*x2
prob += (Empregados<=100)
Fertilizantes=2*x1+4*x2
prob += (Fertilizantes<=120)
Area=x1+x2
prob += (Area<=45)
```

6. Escrevendo o problema

```
print (prob)
```

7. Resolvendo o problema

```
optimization_result = prob.solve()
```

8. Verificando se a solução ótima foi encontrada

```
assert optimization_result == pulp.LpStatusOptimal
```

9. Mostrando o resultado

```
for var in (x1, x2): print('A produção ótima deve ser
{:1.0f}'.format(var.name, var.value()))
```

Esta forma ou sintaxe não é a única possível no PuLP, mas ela é simples e funciona, podendo ser adaptada para qualquer outro problema de otimização linear contínua ou inteira. A documentação do PuLP está disponível em <https://coin-or.github.io/pulp/> ,

5.2.2 GEKKO

E se tivermos que resolver um problema de otimização não-linear? O Python possui diversos solvers disponíveis e gratuitos. Vamos apresentar aqui um deles, o GEKKO. A sintaxe é muito similar ao PuLP.

Vamos considerar novamente o Problema da Mochila, mas vamos modelá-lo como um problema de otimização quadrático, binário e sem restrições, conhecido como QUBO, do inglês *Quadratic unconstrained binary optimization*. Este formato permite resolver problemas em computadores quânticos utilizando o Algoritmo de Grover (ver [85], por exemplo).

Podemos modelar o problema da mochila como um QUBO penalizando na função-objetivo os pontos inviáveis por meio de uma função quadrática da seguinte forma:

$$(PM) \quad \text{Min} \quad \sum_{i=1}^n u_i x_i + (\sum_{i=1}^n m_i + s_1 + 2s_2 + 4s_3 + 8s_4 + 16s_5 - 15)^2 \\ x_i, s_j \in \{0, 1\} \quad i = 1, \dots, n, \quad j = 1, \dots, 5.$$

Aqui adicionamos variáveis de folga para termos uma restrição de igualdade. Desta forma, a função-objetivo será penalizada se a soma das massas de uma solução candidata ultrapasse a capacidade da mochila. É importante notar que todas as variáveis são binárias.

Podemos resolver este problema no Python usando o Gekko no ambiente Colab seguindo os passos abaixo:

- Instalando e Inicializando o GEKKO.

```
!pip install -q pyomo
!pip install gekko
from gekko import GEKKO m = GEKKO()
m.options.SOLVER=1
```

A opção `solver = 1` opta pelo método APOPT, apto a resolver problemas de otimização não-linear inteira mista.

- Definindo os parâmetros do GEKKO.

```
m.solver_options = ['minlp_maximum.iterations 1000',
# número máximo de iterações
'minlp_max_iter_with_int_sol 1000',
# número máximo de iterações com valores inteiros
'nlp_maximum.iterations 1000',
# número máximo de iterações do problema não-linear
'minlp_integer_tol 0.05',
# máximo erro admitido em relação a uma solução inteira
'minlp_gap_tol 0.01'
# critério de convergência]
```

- Definindo as variáveis.

```

x1 = m.Var(lb=0,ub=1,integer=True)
x2 = m.Var(lb=0,ub=1,integer=True)
x3 = m.Var(lb=0,ub=1,integer=True)
x4 = m.Var(lb=0,ub=1,integer=True)
x5 = m.Var(lb=0,ub=1,integer=True)
x6 = m.Var(lb=0,ub=1,integer=True)
x7 = m.Var(lb=0,ub=1,integer=True)
x8 = m.Var(lb=0,ub=1,integer=True)
x9 = m.Var(lb=0,ub=1,integer=True)
x10 = m.Var(lb=0,ub=1,integer=True)
s1 = m.Var(lb=0,ub=1,integer=True)
s2 = m.Var(lb=0,ub=1,integer=True)
s3 = m.Var(lb=0,ub=1,integer=True)
s4 = m.Var(lb=0,ub=1,integer=True)
s5 = m.Var(lb=0,ub=1,integer=True)

```

- Definindo a função-objetivo

```

m.Maximize(10*(x1)+12*(x2)+5*(x3)+
14*(x4)+18*(x5)+7*(x6)+
11*(x7)+19*(x8)+2*(x9)+
(x10)-1000*(2*(x1)+3*(x2)+
(x3)+4*(x4)+8*(x5)+
2*(x6)+4*(x7)+9*(x8)+
1.8*(x9)+0.5*(x10)+s1+
2*s2+4*s3+8*s4-15)**2)

```

Vale destacar aqui que foi escolhido, arbitrariamente, o parâmetro de penalização igual a 1000.

- Resolvendo o problema.

```

m.options.IMODE = 3
m.solve()

```

- Imprimindo o resultado

```

print('Resultado')
print('x1: ' + str(x1.value))
print('x2: ' + str(x2.value))
print('x3: ' + str(x3.value))
print('x4: ' + str(x4.value))
print('x5: ' + str(x5.value))
print('x6: ' + str(x6.value))
print('x7: ' + str(x7.value))
print('x8: ' + str(x8.value))
print('x9: ' + str(x9.value))
print('x10: ' + str(x10.value))
print('x11: ' + str(x11.value))
print('x12: ' + str(x12.value))
print('x13: ' + str(x13.value))

```

Note-se que também simples usar o Gekko. Contudo, para usá-lo de forma eficiente é preciso entender bem os papéis dos parâmetros a partir do método de otimização escolhido (veja [86]). Isso acontece em qualquer bom *solver*.

5.3 Como usar o AMPL

O AMPL (A Mathematical Programming Language) é uma linguagem de modelagem algébrica para problemas de otimização lineares e não lineares, em variáveis discretas ou contínuas ([87]). O AMPL permite a utilização de diversos solvers, entre eles, o CPLEX, GUROBI, BARON, KNITRO. Os dois primeiros são muito eficientes para encontrar soluções ótimas de problemas de otimização com variáveis inteiras. O BARON e o KNITRO são solvers de otimização não-linear, sendo o BARON de otimização global e o KNITRO um veloz solver de otimização local. O AMPL foi desenvolvido nos Laboratórios Bell e é um dos softwares mais utilizados no mundo na área de otimização. Ele possui uma grande vantagem: é simples e intuitivo.

O AMPL é um software pago, mas é possível usar uma versão gratuita, com limitações no número de variáveis e restrições dos problemas. Também é possível solicitar versões gratuitas para uso acadêmico. Veja na página web

<https://portal.ampl.com/account/ampl/login>

No AMPL é possível utilizar o prompt, mas acreditamos ser mais eficiente trabalhar com três tipos diferentes de arquivos: modelo (.mod), dados (.dat), e o executável (.run).

5.3.1 Resolvendo o Problema da Mochila

Vamos começar mostrando como modelar o problema da mochila.

1. Construindo o modelo Este arquivo deve ser criado como .mod logo na abertura do mesmo, conforme mostrado na figura 5.3. Vamos chamá-lo de modelo.mod. Uma vez aberto, basta digitar os comandos abaixo.

- Declarando os parâmetros do modelo

```

param n;
param K;
param u {i in 1..n};
param m {i in 1..n};

```



- ✔ **Step 1: Download AMPL and Solvers**
Get the latest binaries for your machine and for the cloud.
- ✔ **Step 2: Activate your License**
Activate your Community Edition or other license.
- ✔ **Step 3: Manage your Licenses**
Start commercial solver trials. Move licenses between machines.

Figura 5.2: Janela de criação de arquivo no AMPL

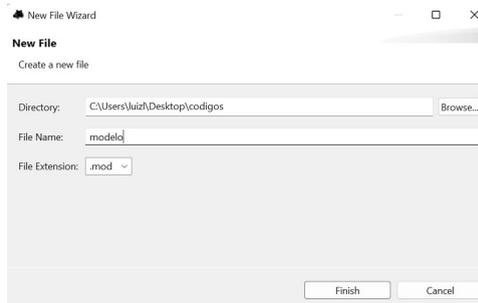


Figura 5.3: Webpage do AMPL para o download do aplicativo e dos solvers

- Declarando as variáveis do modelo

```
var x {i in 1..n} binary;
```

- Declarando a função-objetivo

```
maximize fobj: sum {i in 1..n}u[i] * x[i]
```

- Declarando a restrição de capacidade da mochila

```
subject to restricao1: sum {i in 1..n}m[i] * x[i] <= K;
```

2. Construindo o arquivo de dados

- Abra um arquivo .dat da mesma forma que foi criado o arquivo .mod. Vamos nomeá-lo dados.dat.
- declare os dados digitando as linhas a seguir:

```
param n=10;
param K=15;
param u:=
1 10
2 12
3 5
4 14
5 18
6 7
7 11
8 19
9 2
10 1;
param m:=
1 2
2 3
3 1
4 4
5 8
6 2
7 4
8 9
9 1.8
10 0.5;
```

3. Construindo o arquivo executável

- Crie um arquivo escolhendo a extensão .run (veja figura 5.4
- Indicando o modelo e os dados que serão resolvidos:

```
reset;
model modelo.mod;
data dados.dat;
```

- Declaramos qual solver vamos utilizar (aqui vamos usar o CPLEX), seus parâmetros e o comando de solução:

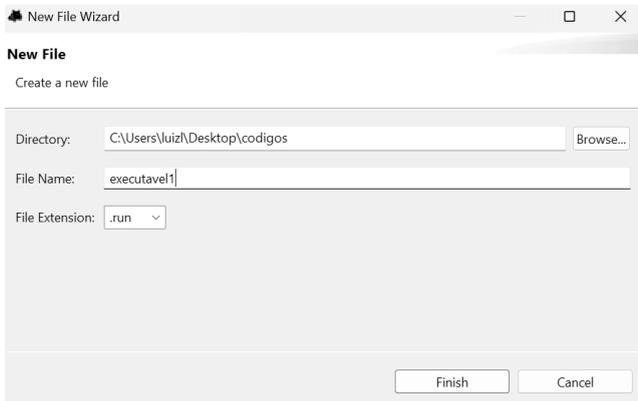


Figura 5.4: Janela do AMPL para criar um arquivo executável

```
option solver cplex;  
option cplex_options "timelimit = 3600";  
solve;
```

Aqui usamos apenas o parâmetro de critério de parada pelo tempo, que neste caso foi definido em 3600 segundos.

- Mostramos a solução obtida e o tempo computacional.

```
display x;  
display _solve_elapsed_time;
```

- Salve o arquivo `executavel.run`. Em seguida digite “include executavel.run” no console e tecele enter. O AMPL irá rodar o modelo no solver usando os dados informados e mostrará a resposta no próprio console.

5.3.2 Resolvendo o Problema de Geometria de Distâncias

Vamos agora mostrar o modelo, uma instância e o executável do problema de geometria de distâncias generalizado abordado por meio do CIA, conforme apresentado na seção 4.2

- Arquivo `.mod`. Aqui a principal diferença em relação ao problema da mochila é que temos variáveis contínuas.

```

param n;

param di {j in 1..n, i in 1..n};

param ds {j in 1..n, i in 1..n};

param k;

param y {j in 1..n, i in 1..3};

param rb {j in 1..n};

var x {i in 1..3};

var lambda {i in 1..k} >=0;

var r {i in 1..k} >=0;

maximize raio: sum {j in 1..k} r[j];

subject to lam1 {i in 1..k - 1}: lambda[i] <=1;

subject to rr1 {i in 1..k - 1}: r[i] <=rb[i];

subject to rr2 {i in 1..k - 1}:
r[k] <= ((1 - lambda[i]) * (ds[i, k] - di[i, k]) - r[i]);

subject to rr3 {i in 1..k - 1}:
r[k] <= lambda[i] * (ds[i, k] - di[i, k]) - r[i];

subject to distancia {i in 1..k - 1}:
((x[1] - y[i, 1])2 + (x[2] - y[i, 2])2 + (x[3] - y[i, 3])2 -
(di[i, k] + lambda[i] * (ds[i, k] - di[i, k]))2) = 0;

```

- Arquivo .dat. A diferença neste caso em relação ao problema da mochila é que temos matrizes para alguns dos parâmetros do problema. É importante notar a forma de declará-la.

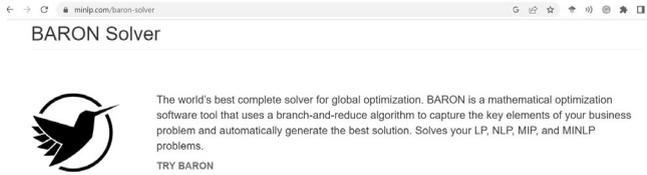


Figura 5.5: Página do solver de otimização global BARON

```

param n=6;
param y: 1 2 3 :=
1 0.000000 0.000000 0.000000
2 -1.000000 0.000000 0.000000
3 -1.676794 0.736172 0.000000;

param di: 1 2 3 4 5 6 :=
1 0.000000 0.850000 1.681281 2.247945 2.824124 3.679591
2 0.850000 0.000000 0.850000 1.734722 2.181280 3.121914
3 1.681281 0.850000 0.000000 0.850000 1.635761 2.367456
4 2.247945 1.734722 0.850000 0.000000 0.850000 1.566655
5 2.824124 2.181280 1.635761 0.850000 0.000000 0.850000
6 3.679591 3.121914 2.367456 1.566655 0.850000 0.000000;

param ds: 1 2 3 4 5 6 := 1 0.000000 1.150000 1.981281 2.847945
3.424124 4.279591 2 1.150000 0.000000 1.150000 2.034722
2.781280 3.721914 3 1.981281 1.150000 0.000000 1.150000
1.935761 2.967456 4 2.847945 2.034722 1.150000 0.000000
1.150000 1.866655 5 3.424124 2.781280 1.935761 1.150000
0.000000 1.150000 6 4.279591 3.721914 2.967456 1.866655
1.150000 0.000000;

```

- Arquivo .run. Neste caso vamos mostrar como rodar várias instâncias e arquivar os resultados em uma planilha csv, que pode ser manipulada, por exemplo, no Excel. Também optamos por usar o Solver Baron, que é um solver de otimização global.

```

reset;
model novodgpg.mod;
param contatempo;
param converge;
option solver baron, solution_round 6;
param repete = 10;
param numero = 6;
param ii default 4;
for {irepete in 1..repete}{
  reset data;
  data (irepete&"auto"&numero&"ran.dat");  data irepete.dat;
  let contatempo := 0;
  let converge := 0;
  let {i1 in 1..3} rb[i1] :=0;
  repeat while ii <= n {
    let k :=ii;
  solve;
  let contatempo := contatempo + _solve_time;
  if solve_result_num >= 200 then{
    let converge := solve_result_num;
    let ii := n;
    {
    let {i1 in 1..3} y[k,i1] := x[i1];
    let {i1 in 1..k} rb[i1] := r[i1];
    let ii:=ii+1; }
    if converge > 0 then
      printf "%d," ,converge >> ("output"&n&"sw.csv");
    else
      printf "0," >> ("output"&n&"sw.csv");
      printf "%f," , contatempo >> ("output"&n&"sw.csv");
      printf "%f" , raiof2 >> ("output"&n&"sw.csv");
      for {imprime in 1..n}{
        printf "%f" , rb[imprime] >> ("output"&n&"sw.csv");
      }
      printf "n" >> ("output"&n&"sw.csv");
      close("output"&n&"sw.csv");
    }
  }
}

```

Note que neste caso vamos resolver 10 instâncias com 6 vértices. Acima apresentamos uma das destas instâncias, que foram geradas conforme descrito no artigo [65]. O comando *solve_result_num* verifica se o solver encontrou o ótimo global. Também podemos verificar a sintaxe dos comandos **for** e **if**, que também são simples e intuitivas, bem como escrever os resultados em uma planilha csv.

O parâmetro *k* diz respeito ao *k*-ésimo vértice que é encontrado no processo iterativo de obter a posição de um vértice por iteração, conforme explicado na seção 4.2.

Web Submission Form

Model File
Enter the location of the AMPL model file (local file)
Escolher arquivo Nenhum arquivo escolhido

Data File
Enter the location of the AMPL data file (local file)
Escolher arquivo Nenhum arquivo escolhido

Commands File
Enter the location of the AMPL commands file (local file)
Escolher arquivo Nenhum arquivo escolhido

Figura 5.6: Entrada dos arquivos

Additional Settings

Dry run: generate job XML instead of submitting it to NEOS

Short Priority: submit to higher priority queue with maximum CPU time of 5 minutes

E-Mail address:

Please do not click the 'Submit to NEOS' button more than once.

Submit to NEOS Clear this Form

Figura 5.7: Entrada dos arquivos

5.4 Como usar o Neos

O NEOS Server é um serviço gratuito baseado na Internet para resolver problemas de otimização. Hospedado pelo Wisconsin Institute for Discovery na University of Wisconsin em Madison, o NEOS Server fornece acesso a diversos *solvers*, como CPLEX, GUROBI, BARON e KNITRO no formato AMPL. Assim, todos os arquivos `.mod` e `.dat` construídos acima podem ser usados no NEOS. Precisamos apenas fazer uma pequena adaptação, não declarando o arquivo `.dat` e `.mod` e o solver utilizado, pois o próprio NEOS faz isso.

Voltamos ao exemplo do Problema da Mochila. Mantemos os arquivos `.mod` e `.dat` e usamos o seguinte arquivo `.run`:

```
option cplex_options "timelimit = 3600";
solve;
display x;
display _solve_elapsed_time;
```

Para indicar os arquivos basta fazer os respectivos uploads, conforme a imagem 5.6 na página web

<https://neos-server.org/neos/solvers/lp:CPLEX/AMPL.html>

Em seguida informe seu email e clique em "submit to NEOS" (veja a figura 5.7. Após resolver o problema o NEOS enviará a solução para o email informado.

5.5 Resumo do capítulo

Neste capítulo apresentamos um pequeno tutorial de como implementar os modelos matemáticos desenvolvidos nos capítulos anteriores.

5.6 Exercícios

1. Resolva os exercícios dos capítulos 2, 3 e 4 utilizando o Python e o AMPL. Compare os resultados e tempos computacionais obtidos.

Bibliografia

- [1] Inuiguchi, M., Ichihashi, H., and Tanaka, H. "Fuzzy programming: A survey of recent developments," in R. Slowinski and J. Teghem, Editors, *Stochastic versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*, Kluwer Academic Publishers, Dordrecht (1990), pp.45-68.
- [2] Dubois, D. and Prade, H. (2009), "Formal representations of uncertainty," Chapter 3 in. D. Bouyssou, D. Dubois, H. Prade, Editors, *Decision-Making Process*, ISTE, London, UK and Wiley, Hoboken, N.J., USA.
- [3] Dubois, D. and Prade, H. (1987), "Fuzzy numbers: An overview," In J. C. Bezdek (Editor). *Analysis of Fuzzy Information, Volume I: Mathematics and Logic*, pp. 3-39, CRC Press.
- [4] Lodwick, Weldon A., and Luiz L. Salles-Neto. Flexible and Generalized Uncertainty Optimization: Theory and Approaches. Vol. 696. Springer Nature, 2021.
- [5] Zadeh, L. A. (1965), "Fuzzy Sets," *Information and Control*, **8**, pp. 338-353.
- [6] Mendel, J. and John, R. I., (2002), "Type 2 fuzzy sets made simple," *IEEE Transactions on Fuzzy Systems*, **10**:2, pp. 117-127.
- [7] Simon, H. A. (1960). *A New Science of Management Decision*. Harper, New York.
- [8] Simon, H. A. (1997). *Models of Bounded Rationality*. MIT Press, Cambridge, Mass.
- [9] Fandel, G. (1994). *PPS-Systeme: Grundlagen, Methoden, Software, Marktanalyse*. Heidelberg: Springer-Verlag.
- [10] Rommelfanger, H. J. (April , 2009), personal communication.
- [11] Dubois, D. (2010), "The role of fuzzy sets in decision sciences: old techniques and new directions," *Fuzzy Sets and Systems*, **184**:1, pp. 3–28.
- [12] Lodwick, W. A. (2008), "Fundamentals of interval analysis and linkages to fuzzy set theory," in Handbook of Granular Computing, Pedrycz, W., Skowron, A., Kreinovich, V., John Wiley, West Sussex, England, pp. 55-79.
- [13] Fortin, J., Dubois, D., and Fargier, H. (2008), "Gradual numbers and their application to fuzzy interval analysis," *IEEE Transactions on Fuzzy Systems*, **16**, pp. 388-402.

-
- [14] Puri, M. L., and Ralescu, D. (1982), "Fuzzy measures are not possibility measures," *Fuzzy Sets and Systems*, **7**, pp. 311-313.
- [15] Dubois, D. and Prade, H. (1991), "Random sets and fuzzy interval analysis," *Fuzzy Sets and Systems* **42**:2,1987, pp. 87-101.
- [16] Zadeh, L. A. (1978), "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems*, **1**, pp. 3-28.
- [17] Puri, M. L., and Ralescu, D. (1986), "Fuzzy random variables," *Journal of Mathematical Analysis and Applications*, **114**, pp. 409-422.
- [18] Dubois, D. and Prade, H. (1988). *Possibility Theory*, Plenum Press, New York.
- [19] Inuiguchi, M., Ichihashi, H., and Kume, Y. (1992), "Relationships between modality constrained programming problems and various fuzzy mathematical programming problems," *Fuzzy Sets and Systems* **49**, pp. 243-259.
- [20] Inuiguchi, M. (1992), "Stochastic programming problems versus fuzzy mathematical programming problems," *Japanese Journal of Fuzzy Theory and Systems* **4**:1, pp. 97-109.
- [21] Bellman, R. E., Zadeh, L. A. (1970), "Decision-making in a fuzzy environment," *Management Science, Serial B* **17**, pp. 141-164.
- [22] Zimmermann, H. (1976), "Description and optimization of fuzzy systems," *International J. of General Systems* **2**, pp. 209-215.
- [23] R.C. Young The algebra of many-valued quantities *Math. Ann.*, 104 (1931), pp. 260-290
- [24] M. Warmus Calculus of approximations *Bull. Acad. Pol. Sci., Cl. III*, 4 (1956), pp. 253-259
- [25] T. Sunaga Theory of an interval algebra and its application to numerical analysis *RAAG Memoirs*, vol. 2 (1958), pp. 547-564
- [26] Moore, R. E. (1965), "The automatic analysis and control of error in digital computing based on the use of interval numbers," In L. B. Rall, (editor), *Error in Digital Computation, Vol. I*, Chapter 2, pages 61-130. John Wiley and Sons, Inc., New York.
- [27] Moore, R. E. (1966). *Interval Analysis*. Prentice-Hall, Englewood Cliffs N. J.
- [28] Moore, R. E. (1979). *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, PA.
- [29] W. Lodwick, Constrained interval arithmetic, University of Colorado at Denver, CCM Report, 138, 1999.
- [30] Oettli, W. and Prager, W. "Computability of approximate solution of linear equations with given error bounds for coefficients and right-hand sides," *Numerische Mathematik*, **6**, pp. 405-409, 1964.
- [31] Hansen, E. "On solving systems of equations using interval arithmetic," *Mathematics of Computation*, **22**, pp. 374-384, 1968.

- [32] Hansen, E. "On linear algebraic equations with interval coefficients," E. Hansen (editor), *Topics in Interval Analysis*, Oxford University Press, pp. 35-46, 1969.
- [33] Rohn, J. (1982). "An algorithm for solving interval linear systems and inverting interval matrices", *Freiburger Intervall-Berichte*, 82/5, pp. 23-36.
- [34] Rohn, J. "A note on solving equations of the type $A^I x = b^I$," *Freiburger Intervall-Berichte*, 84/6, pp. 29-31, 1986.
- [35] Rohn, J. "Solving interval linear systems" *Freiburger Intervall-Berichte* 84/7, pp. 1-14, 1986.
- [36] Rohn, J. "Solvability of systems of linear interval equations," *SIAM Journal on Matrix Analysis and Applications*, **25**:1, pp. 237-245, 2003.
- [37] Neumaier, A. "New techniques for the analysis of linear interval equations," *Linear Algebra*, **58**, pp. 273-325, April 1984.
- [38] Kreinovich, V. Lakeyev, A. V. , Rohn, J. and Kahl, P. "Computational complexity and feasibility of data processing and interval computations," *Applied Optimization*, Chapter 10, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.
- [39] Neumaier, A. (1990). *Interval Methods for Systems of Equations*. Cambridge, UK: Cambridge University Press.
- [40] Shary, S. P. "Optimal solution of interval linear algebraic systems," *Interval Computations*, **2**, pp. 7-30, 1991.
- [41] Shary, S. P. "On controlled solution set of interval algebraic systems, *Interval Computations*, **4**:6, pp. 66-75, 1992.
- [42] Shary, S. P. "Solving the tolerance problem for interval linear equations," *Interval Computations*, **2**, pp. 4-22, 1994.
- [43] Shary, S. P. "A new technique in systems analysis under interval uncertainty and ambiguity," *Reliable Computing*, **8**, pp. 321-418, 2002.
- [44] Moore, R. E., Kearfott R. B. , and Cloud, M. J. (2009). *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.
- [45] Lodwick, W. A. and Dubois, D. (2015), "Interval linear systems as a necessary step in fuzzy linear systems," *Fuzzy Sets and Systems*, **274**, pp. 227-251.
- [46] Lodwick, W. A. (1999), "Constrained Interval Arithmetic." *CCM Report* **138**, Feb.1999.
- [47] Pivkina, I., and Kreinovich, V. (2019). "Finding least expensive tolerance solutions and least expensive tolerance revisions: Algorithms and computational complexity", *International Journal of Intelligent Technologies and Applied Statistics*, **12**:2, pp. 151-168.
- [48] Barth, W., and Nuding, E. (1974). "Optimale losung von intervallgleichungssystemen," *Computing*, **2**, pp. 117-125.

-
- [49] Kuhn, Kenneth, "Bi-objective robust optimisation." *European Journal of Operational Research* 252.2 (2016): 418-431.
- [50] Birge J. R. and Louveaux F. (1997) *Introduction to Stochastic Programming*. Springer.
- [51] Hwang, C-L., and Abu Syed Masud. "Multiple objective decision making methods and applications: a state-of-the-art survey", Vol. 164. *Springer Science & Business Media*, 2012.
- [52] Thipwiwatpotjana, Phantipa, and Weldon A. Lodwick. "Pessimistic, optimistic, and minimax regret approaches for linear programs under uncertainty." *Fuzzy Optimization and Decision Making* 13.2 (2014): 151-171.
- [53] Ben-Tal, Aharon, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*. Vol. 28. Princeton university press, 2009.
- [54] Renan Brito Cano Butkeraites; Luiz Leduino Salles-Neto; Weldon A. Lodwick. A new feasibility quality measure for optimization under uncertainty. In: ANAIS DO LIV SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 2022, Juiz de Fora.
- [55] Bertsimas, Dimitris, Melvyn Sim, and Meilin Zhang. "Adaptive distributionally robust optimization." *Management Science* 65.2 (2019): 604-618.
- [56] B. Donald, *Algorithms in Structural Molecular Biology*, MIT Press, Boston, 2011.
- [57] A. Brünger and M. Nilges, Computational challenges for macromolecular structure determination by X-ray crystallography and solution NMR-spectroscopy, *Quarterly Reviews of Biophysics*, 26:49-125, 1993.
- [58] C. Lavor and N. Maculan, A function to test methods applied to global minimization of potential energy of molecules, *Numerical Algorithms*, 35:287-300, 2004.
- [59] C. Lavor, On generating instances for the molecular distance geometry problem, in *Global Optimization: From Theory to Implementation*, L. Liberti and N. Maculan (eds.), Springer, Berlin, 405-414, 2006.
- [60] C. Lavor, Analytic evaluation of the gradient and Hessian of molecular potential energy functions, *Physica D*, 227:135-141, 2007.
- [61] K. Wütrich, Protein structure determination in solution by nuclear magnetic resonance spectroscopy, *Science*, 243:45-50, 1989.
- [62] G. Crippen and T. Havel, *Distance Geometry and Molecular Conformation*, Wiley, New York, 1988.
- [63] C. Lavor, L. Liberti, and A. Mucherino, The interval BP algorithm for the discretizable molecular distance geometry problem with interval data, *Journal of Global Optimization*, 56:855-871, 2013.
- [64] I. P. Caruso, V. Panwalkar, M. A. Coronado, A. J. Dingley, W. L. Cornélio, D. Willbold, ... and R. J. Eberle. Structure and interaction of *Corynebacterium pseudotuberculosis* cold shock protein A with Y-box single-stranded DNA fragment. *The FEBS journal*, 285(2), 372-390, 2018.

- [65] de Salles Neto, Luiz Leduino, Carlile Lavor, and Weldon Lodwick. "A constrained interval approach to the generalized distance geometry problem." *Optimization Letters* 14 (2020): 483-492.
- [66] A. Sit and Z. Wu, Solving a generalized distance geometry problem for protein structure determination, *Bulletin of Mathematical Biology*, 73:1932-1951, 2011.
- [67] Butkeraites, Renan B., Luiz L. Salles-Neto, and Weldon A. Lodwick. Localização de alvos por uma rede de sensores sem fio a partir de distâncias intervalares. *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics* 6.2 (2018).
- [68] Cao, Ming, Brian DO Anderson, and A. Stephen Morse. Sensor network localization with imprecise distances. *Systems and control letters* 55.11 (2006): 887-893.
- [69] Mao, Guoqiang, Barış Fidan, and Brian DO Anderson. "Wireless sensor network localization techniques." *Computer networks* 51.10 (2007): 2529-2553.
- [70] HLADÍK, Milan. On approximation of the best case optimal value in interval linear programming. *Optimization Letters*, v. 8, p. 1985-1997, 2014.
- [71] Haessler, R.W., (1975). Controlling cutting pattern changes in one-dimensional trim problems. *Operations Research*, 23, 483-493.
- [72] Umetani, S., Yagiura, M., Ibaraki, T., (2003). One-dimensional cutting stock problem to minimize the number of different patterns. *European Journal of Operational Research*, 146, 388-402.
- [73] Moretti, A.C., Salles Neto, L.L., 2008. Nonlinear cutting stock problem model to minimize the number of different patterns and objects. *Computational and Applied Mathematics*, 27, 61-78.
- [74] Golfeto, R.R., Moretti, A.C., Salles Neto, L.L., (2009a). A genetic symbiotic algorithm applied to the one-dimensional cutting stock problem. *Pesquisa Operacional*, 29(3), 365-382.
- [75] Golfeto, R.R., Moretti, A.C., Salles Neto, L.L., (2009b). A genetic symbiotic algorithm applied to the cutting stock problem with multiple objectives. *Advanced Modeling and Optimization*, 11, 473-501.
- [76] X. Liu, J. Pei, L. Liu, H. Cheng, M. Zhou, and P. Pardalos, Cutting Stock Problem with the IoT. In *Optimization and Management in Manufacturing Engineering*. Springer, 2017. ISSN: 1931-6828.
- [77] Goulimins, C. Counterexamples in the CSP. arXiv preprint arXiv:2004.01937, 2020.
- [78] A. Aliano-Filho, A. C. Moretti e M. Pato, A comparative study of exact methods for the bi-objective integer one-dimensional cutting stock problem. *Journal of the Operational Research Society*, 69:91-107, 2018.
- [79] D. Alem, P. Munari, M. Arenales e P. Ferreira, On the cutting stock problem under stochastic demand. *Annals of Operations Research*, 179:169-186, 2010.
- [80] P. Beraldi, M. Bruni e D. Conforti, The stochastic trim-loss problem. *European Journal of Operational Research*, 197:42-49, 2009.

-
- [81] M. Arana-Jimenez e L. L. Salles-Neto, Sufficient condition for partial efficiency in a bicriteria nonlinear cutting stock problem. *RAIRO-Operations Research*, 51:709–716, 2017.
- [82] L. L. Salles-Neto, S. Araujo, and R. R. Golfeto, Um problema de corte de estoque multiobjetivo. *Pesquisa Operacional para o Desenvolvimento*, 6:183–201, 2014.
- [83] Bortolete, J. C., Bueno, L. F., Butkeraites, R., Chaves, A. A., Collaço, G., Magueta, M., ... and Yanasse, H. H. (2022). A support tool for planning classrooms considering social distancing between students. *Computational and Applied Mathematics*, 41, 1-23.
- [84] Salles Neto, L. L. D., Martins, C. B., Chaves, A. A., Konstantyner, T. C. R. D. O., Yanasse, H. H., Campos, C. B. L. D., ... and Soares, F. D. S. (2020). Forecast UTI: Application for predicting intensive care unit beds in the context of the COVID-19 pandemic. *Epidemiologia e Serviços de Saúde*, 29, e2020391.
- [85] GILLIAM, A. WOERNER, Stefan, G. Constantin. Grover adaptive search for constrained polynomial binary optimization. *Quantum*, v. 5, 2021.
- [86] BEAL, L. et al. Gekko optimization suite. *Processes*, v. 6, n. 8, 106, 2018.
- [87] GAY, D. M. The AMPL modeling language: An aid to formulating and solving optimization problems. In *Numerical Analysis and Optimization: NAO-III*, Muscat, Oman, January 2014. Springer international publishing, 95-116, 2015.

Índice

AMPL, 75

covid-19, 64

Dureza, 41

flexibilidade, 2

fuzzy, 2, 3

GEKKO, 73

Hardness, 41

incerteza, 4

Incerteza generalizada, 4

interval hull, 20

intervalo, 5, 11

Markowitz, 63

Minimizar o máximo arrependimento, 32

Minmax regret, 32

Necessidade, 6

NEOS, 82

Otimização flexível, 13

otimização multiobjetivo, 31

Otimização robusta, 40

otimização sob incerteza generalizada, 8

Possibilidade, 6

possibilidade, 12

Problema de corte de estoque, 58

Problema de Geometria de Distâncias,
48

Problema de Localização de sensores, 53

problema de otimização, 5

Problema de Otimização de Portfólio, 64

Problema de otimização linear interva-
lar, 55

Problema do fazendeiro, 33

PULP, 69

Python, 69

sistema linear intervalar, 20

solução eficiente, 31