

A Tutorial on the CVX System for Modeling and Solving Convex Optimization Problems

D. A. Guimarães, G. H. F. Floriano and L. S. Chaves

Abstract— In many areas of knowledge, situations in which we have to model and solve optimization problems are recurrent. Among the mathematical theories that support the solution of such problems, convex optimization unveiled to be an important tool. This is mainly due to the existence of algorithms whose computational solution has matured a lot in terms of speed of solution and reliability during the past few years. In this context, it deserves attention the CVX, a system for modeling and solving convex optimization problems. CVX uses the disciplined convex programming concept, which is a set of conventions or rules that permits the verification of convexity and automatic conversion of the problem instance into forms directly handled by the embedded solvers. This tutorial presents CVX in a didactical manner whose objective is to facilitate the learning about using the tool along with disciplined convex programming. Besides a number of examples, the tutorial also provides some applications, their corresponding CVX codes, the numerical solutions and discussions.

Keywords— Convex optimization, CVX, disciplined convex programming.

I. INTRODUÇÃO

UM problema no qual se busca maximizar ou minimizar uma função real (função objetivo) de variáveis reais ou inteiras, sujeito a restrições sobre essas variáveis, é um problema de otimização matemática ou programação matemática [1, p. 1]. A formulação geral padrão dessa classe de problema é

$$\begin{aligned} & \text{minimize } f_0(\mathbf{x}) \\ & \text{sujeito a } f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \quad \quad h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p, \end{aligned} \quad (1)$$

sendo $\mathbf{x} = (x_1, x_2, \dots, x_n)$ a variável (ou vetor de variáveis) de otimização do problema, $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ a função objetivo, $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ com $i = 1, 2, \dots, m$ as funções associadas às restrições de desigualdade e $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$ com $i = 1, 2, \dots, p$ as funções associadas às restrições de igualdade. O vetor \mathbf{x}^* é chamado de ponto ótimo ou solução do problema (1) se, dentre os vetores que satisfazem as restrições, é com ele que a função objetivo apresenta menor valor, ou seja, para qualquer \mathbf{z} tal que $f_1(\mathbf{z}) \leq 0, \dots, f_m(\mathbf{z}) \leq 0$ e $h_1(\mathbf{z}) = 0, \dots, h_p(\mathbf{z}) = 0$, tem-se $f_0(\mathbf{z}) \geq f_0(\mathbf{x}^*)$.

Neste tutorial os problemas são tratados sob a ótica da otimização convexa [1] e sua solução através do CVX [2] via programação convexa disciplinada [2][3][4]. Portanto, é recomendável que o leitor tenha, pelo menos, conhecimento básico sobre álgebra linear, funções convexas e conjuntos convexas. É desejável que o leitor também tenha conhecimento sobre as formas padrão das várias instâncias de problemas de otimização convexa [1].

A. Breve histórico.

Formalmente, o início histórico da otimização remonta a antiguidade, especificamente na resolução de problemas geométricos tais como o cálculo da mínima distância entre um ponto e uma reta ou a determinação do retângulo de máxima área dentre aqueles que possuem um perímetro constante (Euclides, 300 a.C.). A partir daí o estudo de problemas de otimização segue sem grande evolução até a idade moderna quando, no Século XVII, Isaac Newton e Gottfried Leibniz apresentam a análise matemática através do cálculo diferencial. Com essa ferramenta o estudo de problemas de otimização ganha um grande impulso, com contribuições significativas dos matemáticos da família Bernoulli, de Leonhard Euler, de Joseph-Louis Lagrange, de Gaspard Monge e de outros mais.

No Século XIX, Adrien-Marie Legendre em 1805 e Carl Friedrich Gauss em 1809 publicaram, de forma independente, o Método dos Mínimos Quadrados (*least squares*, LS) que é uma técnica ainda muito utilizada na solução de problemas de otimização. Em 1815 a ideia de concavidade é apresentada nos trabalhos de Thomas Malthus e outros. Jean-Baptiste Joseph Fourier, em 1826, formulou problemas de otimização lineares para resolver aplicações de mecânica e probabilidade.

Já no Século XX, Gyula Farkas apresenta, em 1902, o Lema de Farkas que, mais tarde, foi utilizado na prova do teorema de Karush-Kuhn-Tucker. Em 1917, Harris Hancock publica o que é considerado o primeiro livro sobre otimização: *Theory of Minima and Maxima*, enquanto D'Arcy Wentworth Thompson publica o livro *On Growth and Form*, no qual aplica otimização matemática para analisar o crescimento de organismos vivos. Leonid Vitaliyevich Kantorovich, em 1939, aplicou a programação linear (*linear programming*, LP) na análise de planejamento econômico. Em 1947 George Bernard Dantzig introduz o algoritmo Simplex para LP e John von Neumann apresenta a teoria da dualidade. Em 1951 Harold William Kuhn e Albert William Tucker reapresentam as condições de otimalidade que já haviam sido apresentadas por William Karush anos antes, denominadas de condições KKT. Também nesse ano Harry Max Markowitz publica a programação quadrática (*quadratic programming*, QP). Dois anos mais tarde, em 1953, Richard Ernest Bellman apresenta a

D. A. Guimarães, Instituto Nacional de Telecomunicações (INATEL), Santa Rita do Sapucaí, Minas Gerais, Brasil, dayan@inatel.br

G. H. F. Floriano, Instituto Nacional de Telecomunicações (INATEL), Santa Rita do Sapucaí, Minas Gerais, Brasil, giovanni@inatel.br

L. S. Chaves, Instituto Nacional de Telecomunicações (INATEL), Santa Rita do Sapucaí, Minas Gerais, Brasil, lucas-srs@hotmail.com

programação dinâmica. Depois, Lester Randolph Ford Junior e Delbert Ray Fulkerson, em 1956, publicam o algoritmo de Ford-Fulkerson para o cálculo de fluxo máximo numa rede de fluxos. Robert Butler Wilson apresenta, em 1963, a Programação Sequencial Quadrática que é ampliada na década de 1970 por Shih-Ping Han e Michael James David Powell. Em 1979, Leonid Genrikhovich Khachiyan lança o primeiro algoritmo em tempo polinomial para problemas lineares e em 1984 Narendra Krishna Karmarkar apresenta o algoritmo de pontos interiores.

Em 2006, Michael Grant, Stephen Boyd e Yinyu Ye apresentam a Programação Convexa Disciplinada (*disciplined convex programming*, DCP) que, implementada no Matlab[®] sob o nome de CVX, tornou a programação convexa uma opção importante para a solução de problemas de otimização, conforme discutiremos a seguir.

B. A otimização convexa.

A programação convexa ou otimização convexa [1] é o ramo da otimização matemática no qual a função objetivo é convexa e o conjunto de pontos factíveis definidos pelas restrições também é convexo. Apesar de suas bases teóricas serem anteriores, experimentou grande desenvolvimento a partir das últimas décadas do Século XX com o desenvolvimento de algoritmos baseados no conceito de pontos interiores. A generalização desses algoritmos propiciou o surgimento de métodos confiáveis e eficientes para a solução de uma grande variedade de problemas de otimização utilizando as interessantes propriedades dos conjuntos convexos. A otimização convexa é uma subclasse da programação não linear (*nonlinear programming*, NLP) que unifica e generaliza o método dos mínimos quadrados (LS), a programação linear (LP) e a programação quadrática (QP) [3].

Atualmente a literatura especializada já contém um número expressivo de livros e artigos científicos relacionados à otimização convexa. Clássicos como [1], [5] e [6], dentre muitos outros, já compõem grande parte dos acervos. Excelentes textos tutoriais podem também ser encontrados, a exemplo de [7], [8] e [9]. É também vasto o cabedal de publicações relacionadas às aplicações da otimização convexa nas mais variadas áreas do conhecimento, com destaque aqui para as áreas de comunicações e processamento de sinais [9][10][11].

A disseminação da otimização convexa por muito tempo foi dificultada por exigir do usuário um grande conhecimento sobre análise convexa e sobre algoritmos numéricos. A partir da metodologia de modelagem proposta por Michael Grant, Stephen Boyd, and Yinyu Ye em [3] e [4], a programação convexa disciplinada (DCP), essa dificuldade foi sensivelmente reduzida. A DCP estabelece um conjunto de convenções que devem ser seguidas na construção e solução de um problema de otimização. O problema assim construído pode ser automaticamente convertido para uma forma solúvel através da adaptação de métodos numéricos comuns da área de otimização. Assim o usuário poderá concentrar todo seu esforço apenas na modelagem do problema, sem dispender

energia com os detalhes matemáticos relacionados à solução do mesmo.

C. Notação e abreviações

A notação matemática adotada neste tutorial é a mesma das referências [1] e [2], exceto no caso da indicação de matrizes, vetores conjuntos numéricos. No caso das grandezas matriciais e vetoriais o nome vem em **bold** e não em *italico*; nos conjuntos numéricos utiliza-se a simbologia padrão comumente adotada na literatura. Assim, por exemplo, a notação que em [1] e [2] é escrita como $A \in \mathbf{R}^{m \times n}$, nesse tutorial é representada por $\mathbf{A} \in \mathbf{R}^{m \times n}$. Transcreve-se a seguir parte da notação de [1, pp. 697-699], já com a adaptação supracitada.

Conjuntos

\mathbb{R}	Números reais.
\mathbb{R}^n	Vetores reais n -dimensionais (matrizes $n \times 1$).
$\mathbb{R}^{m \times n}$	Matrizes $m \times n$ reais.
$\mathbb{R}_+, \mathbb{R}_{++}$	Números reais não negativos, positivos.
$\mathbb{R}_+^n, \mathbb{R}_{++}^n$	Vetores reais não negativos, positivos, elemento a elemento.
\mathbb{C}	Números complexos.
\mathbb{C}^n	Vetores complexos n -dimensionais
$\mathbb{C}^{m \times n}$	Matrizes $m \times n$ complexas.
\mathbb{Z}	Inteiros.
\mathbb{Z}_+	Inteiros não negativos.
\mathbb{S}^n	Matrizes simétricas de ordem $n \times n$.
$\mathbb{S}_+^n, \mathbb{S}_{++}^n$	Matrizes simétricas semidefinidas positivas, definidas positivas, de ordem $n \times n$.

Vetores e matrizes e outras variáveis

$\mathbf{1}$	Vetor com todos os elementos iguais a 1.
d^*	Valor ótimo do problema de otimização dual.
e_i	i -ésimo vetor da base canônica.
\mathbf{I}	Matriz identidade.
p^*	Valor ótimo do problema de otimização primal.
\mathbf{x}^*	Ponto ótimo ou solução do problema de otimização.
\mathbf{X}^T	Transposta da matriz \mathbf{X} .
\mathbf{X}^H	Hermitiano (ou conjugado complexo transposto) da matriz \mathbf{X} .
$\text{tr}(\mathbf{X})$	Traço da matriz \mathbf{X} .
$\lambda_i(\mathbf{X})$	i -ésimo maior autovalor da matriz simétrica \mathbf{X} .
$\lambda_{\min}(\mathbf{X})$	Mínimo autovalor da matriz simétrica \mathbf{X} .
$\lambda_{\max}(\mathbf{X})$	Máximo autovalor da matriz simétrica \mathbf{X} .
$\sigma_i(\mathbf{X})$	i -ésimo maior valor singular da matriz \mathbf{X} .
$\sigma_{\min}(\mathbf{X})$	Mínimo valor singular da matriz \mathbf{X} .
$\sigma_{\max}(\mathbf{X})$	Máximo valor singular da matriz \mathbf{X} .
\mathbf{X}^\dagger	Inversa de Moore-Penrose (ou pseudo-inversa) da matriz \mathbf{X} .

$\text{diag}(\mathbf{x})$	Matriz diagonal com elementos iguais a x_1, \dots, x_n
$\text{rank}(\mathbf{A})$	Ranque (posto) da matriz \mathbf{A} .
$\mathcal{R}(\mathbf{A})$	Espaço das colunas (<i>range, column-space</i>) da matriz \mathbf{A} .
$\mathcal{N}(\mathbf{A})$	Espaço nulo (<i>null-space</i>) da matriz \mathbf{A} .

Normas e distâncias

$\ \cdot\ $	Norma.
$\ \cdot\ _*$	Dual da norma (ou norma dual).
$\ \mathbf{x}\ _2$	Norma \square_2 (ou norma euclidiana) do vetor \mathbf{x} .
$\ \mathbf{x}\ _1$	Norma \square_1 do vetor \mathbf{x} .
$\ \mathbf{x}\ _\infty$	Norma \square_∞ do vetor \mathbf{x} .

Desigualdades generalizadas

$\mathbf{x} \succeq \mathbf{y}$	Desigualdade componente a componente entre os vetores \mathbf{x} e \mathbf{y} . $\mathbf{x} \succeq \mathbf{y} \Rightarrow x_i \leq y_i, i = 1 \dots n$.
$\mathbf{x} \prec \mathbf{y}$	Desigualdade estrita componente a componente entre \mathbf{x} e \mathbf{y} . $\mathbf{x} \prec \mathbf{y} \Rightarrow x_i < y_i, i = 1 \dots n$.
$\mathbf{X} \succeq \mathbf{Y}$	Desigualdade matricial entre as matrizes simétricas \mathbf{X} e \mathbf{Y} . $\mathbf{X} \succeq \mathbf{Y} \Rightarrow (\mathbf{Y} - \mathbf{X}) \in \mathbb{S}_+^n$.
$\mathbf{X} \prec \mathbf{Y}$	Desigualdade matricial estrita entre as matrizes simétricas \mathbf{X} e \mathbf{Y} . $\mathbf{X} \prec \mathbf{Y} \Rightarrow (\mathbf{Y} - \mathbf{X}) \in \mathbb{S}_{++}^n$.
$\mathbf{x} \succeq_K \mathbf{y}$	Desigualdade generalizada induzida pelo cone próprio K . $\mathbf{x} \succeq_K \mathbf{y} \Rightarrow (\mathbf{y} - \mathbf{x}) \in K$.
$\mathbf{x} \prec_K \mathbf{y}$	Desigualdade estrita generalizada induzida pelo cone próprio K . $\mathbf{x} \prec_K \mathbf{y} \Rightarrow (\mathbf{y} - \mathbf{x}) \in \text{int}(K)$.
$\mathbf{x} \succeq_{K^*} \mathbf{y}$	Desigualdade generalizada induzida pelo cone dual K^* .
$\mathbf{x} \prec_{K^*} \mathbf{y}$	Desigualdade estrita generalizada induzida pelo cone dual K^* .

Topologia e análise convexa

$\text{card}(C)$	Cardinalidade do conjunto C .
$\text{int}(C)$	Interior do conjunto C .
$\text{relint}(C)$	Interior relativo do conjunto C .
K^*	Cone dual associado ao cone K .
f^*	Função conjugada de f .

Probabilidade

$E[\mathbf{x}]$	Valor esperado (esperança) do vetor aleatório \mathbf{x} .
$\text{Prob}[S]$	Probabilidade de ocorrência do evento S .
$\text{var}[X]$	Variância da variável escalar aleatória X .
$\mathcal{N}(\mathbf{c}, \Sigma)$	Distribuição gaussiana n -dimensional de média \mathbf{c} e matriz de covariância Σ .
Φ	Função distribuição cumulativa de uma variável aleatória $\mathcal{N}(0, 1)$.

Funções e derivadas

$f : A \rightarrow B$	Significa que f é uma função de domínio em A ($\text{dom}(f) \subseteq A$) sobre o contradomínio (imagem) no conjunto B .
$\text{dom}(f)$	Domínio da função f .
$\text{epi}(f)$	Epígrafo da função f .
∇f	Gradiente da função f .
$\nabla^2 f$	Hessiana da função f .
Df	Matriz derivada (Jacobiano) da função f .

Siglas

DCP	<i>Disciplined convex programming</i> (programação convexa disciplinada).
GP	<i>Geometric programming</i> (programação geométrica).
KKT	Karush-Kuhn-Tucker.
LMI	<i>Linear matrix inequality</i> (desigualdade matricial linear).
LP	<i>Linear programming</i> (programação linear).
LS	<i>Least squares</i> (mínimos quadrados).
MIDCP	<i>Mixed integer disciplined convex programming</i> (programação convexa disciplinada inteira mista)
NLP	<i>Nonlinear programming</i> (programação não linear).
QCQP	<i>Quadratically constrained quadratic programming</i> (programação quadrática com restrição quadrática).
QP	<i>Quadratic programming</i> (programação quadrática).
SDP	<i>Semidefinite programming</i> (programação semidefinida).
SOCP	<i>Second order cone programming</i> (programação cônica de segunda ordem).

D. Estrutura do restante do artigo

A Seção II mostra como obter e instalar o pacote CVX no Matlab, a sintaxe básica dos seus comandos e as principais regras para modelagem de um problema de otimização de forma disciplinada (DCP). A Seção III apresenta os mais importantes formatos padrão de problemas de otimização e como proceder sua modelagem no CVX via DCP. Na Seção IV são descritos os passos necessários na obtenção do problema dual de um problema de otimização (primal) e a aplicação desse procedimento para vários casos e a correspondente implementação no CVX. Já a Seção V discute algumas aplicações de otimização convexa apresentando a correspondente modelagem, o código CVX, os resultados e sua interpretação. É importante citar que embora a otimização encontre aplicações em praticamente todas as áreas de conhecimento, neste tutorial foi dada especial atenção a problemas de otimização relacionados apenas com a Engenharia de Telecomunicações. As considerações finais do tutorial estão na Seção VI.

Os códigos em CVX referentes a todos os exemplos e aplicações citados ao longo do artigo podem ser solicitados aos autores e serão fornecidos gratuitamente.

Grande parte dos exemplos e casos citados ao longo do artigo está presente em [1], de forma que o leitor possa obter mais detalhes de caráter teórico se assim for necessário.

II. O SISTEMA CVX

A. Descrição

O CVX [2] é um sistema de modelagem em *software* desenvolvido na plataforma Matlab para a construção e solução de problemas de otimização através da programação convexa disciplinada [3]. Por ser implementado no Matlab e utilizar funções e comandos comuns do mesmo, o CVX acaba transformando-o numa linguagem de modelagem para problemas de otimização que facilita a manipulação dos resultados obtidos na solução do problema e a integração desse problema a um projeto maior no qual o problema de otimização é um dos componentes.

O CVX suporta e integra a programação linear (LP), a programação quadrática (QP), a programação cônica de segunda ordem (SOCP) e a programação semidefinida (SDP), as quais serão discutidas mais adiante. Também é capaz de resolver problemas complexos de otimização nos quais a modelagem utilize funções não diferenciáveis. A partir da versão 2.0 o CVX passou a suportar problemas de otimização inteira mista na forma convexa disciplinada (MIDCP).

Para duas classes específicas de problemas de otimização o CVX oferece modos especiais de construção da programação. A primeira é a classe dos problemas que utilizam a programação semidefinida (SDP), para a qual o CVX aplica uma interpretação matricial para o operador de desigualdade, permitindo que as desigualdades matriciais lineares (LMI) e o programa SDP em si sejam expressos de forma mais natural. A segunda é a classe dos problemas formulados a partir da programação geométrica (GP), para os quais o CVX aceita todas as funções especiais e regras de combinação de GP, incluindo as funções especiais monomiais, as posinomiais e as posinomiais generalizadas, transformando esses problemas na forma convexa de modo que possam ser resolvidos.

É fundamental citar que a utilização eficaz do CVX requer do usuário o conhecimento razoável dos conceitos principais sobre otimização convexa [1], sobre álgebra linear [12] e sobre as convenções estabelecidas pela programação convexa disciplinada [3]. É importante também, antes de inserir um problema no CVX, saber se o mesmo é um problema de otimização convexa, já que o CVX não deve ser utilizado para verificação de convexidade: há problemas de otimização convexa que não podem ser resolvidos pelo CVX, a menos que sejam traduzidos em uma forma equivalente compatível com as regras DCP. Por outro lado, se o CVX declara um problema como resolvido, tem-se a garantia de que se trata de um problema de otimização convexa.

Deve também ficar claro que é pouco provável que o CVX funcione bem se o número de variáveis do problema a ser resolvido for muito grande (problema de grande porte). Neste caso é aconselhável desenvolver um solucionador (*solver*) específico que permita alcançar a eficiência desejada. Todavia, antes de começar esse desenvolvimento, pode-se aplicar o

CVX para resolver versões em escala reduzida do problema, permitindo que o modelo seja testado.

O CVX será capaz de resolver muitos problemas de médio e grande porte que apresentem determinadas propriedades exploráveis, tais como a esparsidade (pequeno número de elementos não nulos) de vetores ou matrizes de entrada, e que evitem o uso de *loops*. Para esses portes de problemas também deve ser evitado o uso de funções que necessitem de aproximação sucessiva, tais como a função logarítmica e a função exponencial. Esse tipo de funções geralmente torna o processamento em Matlab extremamente lento.

Nos casos em que são encontradas dificuldades para resolver problemas de grande porte com o CVX, é aconselhável considerar a possibilidade de publicar sua modelagem no Fórum CVX: <http://ask.cvxr.com/questions>. A denominada “comunidade CVX” pode ser capaz de sugerir uma formulação equivalente para o problema, formulação esta capaz de permitir que o CVX processe de forma mais rápida e eficiente o problema em questão. O Fórum CVX também permite a troca de informações e experiências entre usuários do CVX no mundo todo.

B. Download e instalação

O CVX é um *software* livre para uso em ambientes acadêmicos e comerciais. As informações completas sobre as plataformas suportadas, licenciamento e instalação podem ser obtidas diretamente em [2].

O procedimento para a obtenção e instalação do CVX na sua versão acadêmica é apresentado a seguir:

- i) Faça o download da última versão do CVX a partir do site <http://cvxr.com/cvx/download>.
- ii) Descompacte o arquivo baixado em qualquer pasta que você desejar e uma subpasta denominada *cvx* será criada. Existem duas importantes exceções: a primeira é de não colocar o CVX na pasta que contém a caixa de ferramentas do Matlab e a segunda é não descompactar uma nova versão do CVX sobrepondo uma versão antiga. Nesse último caso é recomendável mover a versão antiga do CVX para fora do caminho, mas não eliminá-la até ter a certeza que a nova versão está funcionando adequadamente.
- iii) Inicialize o Matlab.
- iv) Altere o diretório do topo do Matlab para aquele onde você baixou a nova distribuição do CVX e execute o comando `cvx_setup`. Por exemplo, se você instalou o CVX em um computador com sistema operacional Windows no caminho `C:\Matlab\pessoal\cvx`, digite os seguintes comandos no *prompt* de comandos do Matlab: `cd C:\Matlab\pessoal\cvx, cvx_setup`. Se você instalou o CVX em Linux ou em um Mac, no endereço `~/Matlab/CVX` digite no *prompt* do Matlab os comandos: `cd ~/Matlab / CVX, cvx_setup`.
- v) A função `cvx_setup` irá executar uma variedade de tarefas para verificar se a instalação está correta. Se nenhum erro acontecer faça o seguinte: selecione o caminho de busca do Matlab para que ele possa encontrar todos os seus arquivos de programa CVX e execute um problema teste simples para verificar se realmente sua instalação está correta.

vi) Em alguns casos, geralmente no Linux, o comando `cvx_setup` pode instruí-lo para criar ou modificar um arquivo `startup.m`. O procedimento irá permitir que você use o CVX sem ter que digitar `cvx_setup` cada vez que reiniciar o Matlab.

C. Sintaxe base

A estrutura genérica de um bloco de código CVX usado para modelar um problema de otimização no Matlab é:

```
cvx_begin (comando de inicialização do bloco)
    declaração das variáveis;
    função objetivo;
    restrições;
cvx_end (comando de finalização do bloco)
```

Se o problema for resolvido com sucesso, o valor ótimo e o ponto ótimo do problema estarão disponíveis respectivamente na variável `cvx_optval` e nas variáveis do problema definidas pelo usuário. Caso ocorra algum erro, tal como problema não factível, ilimitado ou falha na solução, mensagens de erros correspondentes irão aparecer na janela de comandos do Matlab.

A seguir é apresentada a sintaxe dos principais comandos que devem ou podem ser utilizados em um bloco de código CVX.

Comandos de inicialização e finalização do bloco

Todo bloco de código CVX deve ser inicializado com o comando `cvx_begin` e terminar com o comando `cvx_end`. Todas as declarações de variáveis, a função objetivo e as restrições que modelam o problema de otimização devem estar entre estes dois comandos.

Variáveis

Todas as variáveis do problema de otimização devem ser declaradas antes de serem utilizadas nas funções. As variáveis podem ser números escalares reais ou complexos, vetores, matrizes ou arranjos multidimensionais. A declaração de uma variável é feita utilizando o comando `variable`, conforme indicado a seguir:

```
Variable nome(d1,...,dn) extensão1...extensão;
```

A identificação da variável deve ser escrita no lugar de `nome`. Exceto para variáveis unidimensionais, deve-se definir a dimensão da variável em questão nos lugares de `d1,...,dn`. Opcionalmente pode-se ainda fornecer informações sobre a estrutura e as condições da variável através de palavras chave que devem ser escritas nos lugares de `extensão1, ..., extensãon`, conforme exemplos a seguir:

```
variable a;
variable v(6) nonnegative;
variable Z(2,2) semidefinite;
variable Q(3,3) complex semidefinite;
```

Nestes exemplos são declaradas respectivamente uma variável escalar `a`, um vetor (matriz coluna) `v` de 6 elementos não negativos, uma matriz `Z` de ordem 2×2 semidefinida

positiva e uma matriz `Q` hermitiana complexa semidefinida positiva de ordem 3×3 . Utilizando o `solver` para MIDCPs é possível declarar variáveis binárias e inteiras através das palavras chaves `binary` e `integer` respectivamente.

Existem diversas palavras chaves possíveis de serem utilizadas na estruturação das variáveis matriciais [2, p. 20]. Quando múltiplas palavras chaves são utilizadas, a estrutura resultante corresponde à interseção de todas as estruturas declaradas. Se não existir nenhuma matriz não nula que se encaixe na interseção dessas estruturas ocorrerá um erro.

É possível declarar mais de uma variável de uma só vez utilizando o comando `variables`, por exemplo:

```
variables a b c x(2) y(2,2) z(3,3);
```

A única limitação para este tipo de declaração é que com ela não é possível declarar variáveis complexas, inteiras ou estruturadas. Isso deve ser feito por meio da repetição da declaração de uma única variável com o comando `variable`, como ilustrado a seguir:

```
variables a x(2) y(2,2,4);
variable v(6) nonnegative;
variable Z(2,2) semidefinite;
```

Função objetivo

A declaração da função objetivo de um problema de otimização deve ser sempre precedida dos comandos `minimize` (`minimise`) ou `maximize` (`maximise`), dependendo da curvatura da função em questão. Se a função objetivo é convexa, trata-se de um problema de minimização e, portanto, deve-se utilizar o comando `minimize`. Se a função objetivo é côncava, trata-se de um problema de maximização e, portanto, deve-se utilizar o comando `maximize`. Se a função objetivo é afim (convexa e côncava), o problema de otimização pode ser tratado tanto como um problema de maximização quanto um problema de minimização, o que implica no uso dos comandos `minimize` e `maximize` respectivamente, a depender do tipo de problema de otimização em questão. A função objetivo deve ser um escalar e não se pode declarar mais do que uma função objetivo por bloco de código CVX. Se nenhuma função objetivo for declarada, o problema passa a ser interpretado como um problema de factibilidade. Neste caso, `cvx_optval` será 0 caso o problema seja factível ou `+Inf` caso o problema seja infactível. Como exemplos de declarações de funções objetivos temos:

```
maximize sqrt(Ax + b);
minimize abs(a'y);
```

Na primeira declaração a função objetivo $f(\mathbf{x}) = \sqrt{\mathbf{Ax} + \mathbf{b}}$ é côncava por se tratar da composição de uma função côncava com uma função afim e, portanto, estará presente em um problema de maximização. Na segunda declaração a função objetivo $f(\mathbf{y}) = |\mathbf{a}^T \mathbf{y}|$ é convexa por se tratar da composição de uma função convexa com uma função afim e, portanto, estará presente em um problema de minimização. As regras de composição serão abordadas na Seção III.

Restrições

O CVX aceita as seguintes restrições:

- restrições de igualdade, $=$, onde ambos os lados são funções afins;
- restrições de desigualdade menor igual, \leq , onde o lado esquerdo é uma função convexa e o lado direito é uma função côncava;
- restrições de desigualdade maior igual, \geq , onde o lado esquerdo é uma função côncava e o lado direito é uma função convexa.

A restrição de não igualdade \neq não é permitida, já que não é convexa. Note que o sinal $=$ é diferente de $==$. O primeiro significa uma atribuição, enquanto o segundo significa uma igualdade. As desigualdades estritas dadas por $<$ ou $>$ são aceitas pelo CVX, porém são tratadas da mesma maneira que as desigualdades não estritas \leq e \geq , respectivamente.

A seguir é apresentado um exemplo de modelagem de um problema de otimização convexa com o CVX, utilizando alguns dos comandos que acabamos de apresentar.

Caso II.1 – Exemplo de problema de otimização convexa.

```
minimize ||Ax - b||2
sujeito a x ≥ 0
```

```
cvx_begin
  variable x(n);
  minimize norm(A*x-b);
  subject to
    x >= 0;
cvx_end
```

O bloco de código CVX representa o problema de otimização convexa sob a variável $\mathbf{x} \in \mathbb{R}^n$, cujo objetivo é minimizar a função convexa $\|\mathbf{Ax} - \mathbf{b}\|_2$ restrita à condição de que $\mathbf{x} \geq 0$, sendo $\mathbf{A} \in \mathbb{R}^{m \times n}$ e $\mathbf{b} \in \mathbb{R}^m$ os dados de entrada.

Por definição, no Matlab as restrições de igualdade e de desigualdade implicam em comparações elemento a elemento. Isto não gera nenhum problema na desigualdade vetorial, como no Caso II.1, mas o faz na desigualdade matricial. Como exemplo, se for feita a declaração $\mathbf{A} \leq \mathbf{B}$ envolvendo as matrizes $\mathbf{A} \in \mathbb{R}^{m \times n}$ e $\mathbf{B} \in \mathbb{R}^{m \times n}$, o CVX irá interpretá-la como $m \times n$ desigualdades escalares do tipo $\mathbf{A}(i, j) \leq \mathbf{B}(i, j)$ e não como $\mathbf{A} \preceq \mathbf{B}$. O tratamento das restrições de desigualdade é alterado de forma adequada dentro do modo de otimização semidefinida, o qual é acionado utilizando-se o comando `cvx_begin sdp` em vez de `cvx_begin` apenas, no início do bloco CVX [2, pp. 39-40]. Exemplos de uso dessa sintaxe serão dados mais adiante.

Funções

O CVX possui uma biblioteca de funções chamada de biblioteca átomo que inclui uma variedade de funções convexas, côncavas e afins adequadas à sintaxe do CVX. Estas funções aceitam variáveis ou expressões como argumentos do bloco de código CVX. Muitas funções dessa biblioteca são comuns no Matlab e foram reimplementadas para serem

suportadas no CVX, tais como `sum`, `trace`, `diag`, `sqrt`, `max`, e `min`. Outras funções são novas e não são encontradas no Matlab. Uma lista completa de funções da biblioteca átomo do CVX pode ser encontrada em [2, pp. 53-62].

Um exemplo de função da biblioteca do CVX é a função `quad_form(x,P)`. Esta função, não encontrada no Matlab, representa a função $\mathbf{x}^T \mathbf{P} \mathbf{x}$ para \mathbf{x} com elementos reais e \mathbf{P} real e simétrica ou $\mathbf{x}^H \mathbf{P} \mathbf{x}$ para \mathbf{x} com elementos complexos e \mathbf{P} Hermitiana. Ela é convexa em \mathbf{x} para \mathbf{P} constante e semidefinida positiva, e côncava em \mathbf{x} para \mathbf{P} constante e semidefinida negativa. Pela regra da composição pode-se verificar que a concavidade ou convexidade da função `quad_form(x,P)` é mantida caso seu argumento seja afim, continuando a ser válida para o CVX.

Um exemplo de composição da função `quad_form(x,P)` com uma função afim é `quad_form(Ax-b, P)`. Esta sintaxe representa a função $(\mathbf{Ax} - \mathbf{b})^T \mathbf{P} (\mathbf{Ax} - \mathbf{b})$, sendo $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$ e $\mathbf{P} \in \mathbb{S}^n$ constantes dadas, e $\mathbf{x} \in \mathbb{R}^n$ a variável. Esta função é a composição de uma função côncava com uma função afim e, portanto, é côncava em \mathbf{x} .

Conjuntos

O CVX suporta a definição e o uso de conjuntos convexas tais como o cone de matrizes $n \times n$ semidefinidas positivas, o cone de segunda ordem (ou cone de Lorentz) e várias normas sob bolas (*norm balls*). Uma lista completa destes conjuntos pode ser encontrada em [2, pp. 60-61].

Para representar um conjunto o CVX possui funções que retornam uma variável anônima que pertence a este conjunto. Para exigir que uma determinada variável \mathbf{x} do problema de otimização pertença a este conjunto basta utilizar a restrição de igualdade, `==`, ou o pseudo-operador, `<In>`, de modo que do lado esquerdo da expressão fique a variável \mathbf{x} e do lado direito fique a função que representa o conjunto. Veja os exemplos:

```
x == nonnegative(n);
Y <In> semidefinite(n);
(w,t) == lorentz(n);
```

O primeiro exemplo restringe a variável \mathbf{x} a pertencer ao ortante não negativo de ordem n , onde todos os elementos (vetores) deste conjunto têm seus componentes não negativos. O segundo exemplo restringe a variável matricial \mathbf{Y} a pertencer ao cone de matrizes semidefinidas positivas de ordem $n \times n$. O terceiro exemplo condiciona a lista ordenada dos elementos \mathbf{w} e t a pertencer ao cone de Lorentz de ordem n , sendo \mathbf{w} uma variável real vetorial e t uma variável escalar.

As representações de conjuntos podem ser inseridas em funções afins e é possível restringir uma função afim para que ela pertença a um conjunto convexo. Seja o exemplo:

```
B*Y*B'-2Y <In> C*semidefinite(n)*C';
```

em que \mathbf{Y} é uma variável matricial simétrica, \mathbf{B} e \mathbf{C} são matrizes constantes de ordem $n \times n$. Esta restrição exige que $\mathbf{B} \mathbf{Y} \mathbf{B}^T - 2 \mathbf{Y} = \mathbf{C} \mathbf{X} \mathbf{C}^T$, para algum $\mathbf{X} \in \mathbb{S}_+^n$.

Variáveis duais

A dualidade, ou mais especificamente a dualidade de Lagrange [1, Cap. 5], tem papel importantíssimo no contexto da modelagem e da solução de problemas de otimização convexa. As principais razões dessa importância são:

- o assunto serve para desmistificar os conceitos sobre os multiplicadores de Lagrange;
- problemas duais são sempre problemas de otimização convexa;
- a solução de problemas duais é muito útil na determinação de limitantes inferiores para a solução de problemas de otimização não convexa de difícil solução;
- a solução de problemas duais pode ser uma alternativa para a solução de problemas de otimização convexa ou como solução de alguns (poucos) problemas de otimização não-convexa de difícil solução;
- a solução de problemas duais sempre serve como algum tipo de certificação sobre a otimalidade da solução de problemas de otimização convexa;
- a solução de problemas duais sempre serve como algum tipo de certificado sobre o grau de proximidade entre a solução do problema dual e a solução de um problema de otimização não convexa ou de um problema original de difícil solução;
- a solução de problemas duais pode servir como critério de parada em algoritmos de otimização.

Todo problema de otimização convexa quando é resolvido pelo CVX tem automaticamente o seu problema dual resolvido. Neste contexto o problema original é chamado também de problema primal. Caso os problemas primal e dual sejam resolvidos com sucesso, é possível acessar os valores ótimos das variáveis primais e duais. Para acessar os valores ótimos das variáveis duais é preciso antes declarar estas variáveis usando a sintaxe `dual variables` e associá-las às correspondentes restrições do problema primal, por meio do separador `;`, conforme mostrado no caso a seguir.

Caso II.2 – Acesso às variáveis duais.

```

minimize cTx
sujeito a Ax = b
          x ≥ 0

```

```

n = size(A,2);
cvx_begin
    variable x(n);
    dual variables y z;
    minimize c'*x;
    subject to
        y : b - A * x == 0;
        z : -x <= 0;
cvx_end

```

O problema do Caso II.2 é um problema de otimização linear na forma padrão sob $x \in \mathbb{R}^n$ com m restrições de igualdade e n restrições de desigualdade. O comando `dual variables` é utilizado para declarar as variáveis duais y e z , as quais são associadas posteriormente a restrição de igualdade $Ax = b$ e de desigualdade $x \geq 0$, respectivamente, utilizando as sintaxes y ; e z : conforme mostra o código. Não é preciso

determinar a dimensão de uma variável dual em sua declaração, pois ela é determinada automaticamente pelo CVX de acordo com a restrição relacionada à variável. Para este caso y terá dimensão m e z terá dimensão n .

Como a variável relacionada à restrição de desigualdade é maior ou igual à zero, pode-se declarar a variável dual do lado direito sem que ocorra nenhuma alteração em seu valor ótimo, tal como em $-x \leq 0 : z$. Quando se trata da restrição de igualdade, declarar a variável dual à direita da restrição tem como consequência a negação do valor ótimo da variável dual.

Se o problema de otimização não for resolvido com sucesso porque o problema primal é infactível ou ilimitado, são possíveis duas combinações de valores para as variáveis primais e duais. Se o problema primal for infactível as variáveis primais são carregadas com `NaN` (*not a number*) e as variáveis duais com valores que indicam a direção da ilimitação, significando que o problema dual associado é ilimitado. Se o problema primal for ilimitado, as variáveis primais são carregadas com valores que indicam sua direção de ilimitação, enquanto as variáveis duais são carregadas com `NaN`, o que indica a não factibilidade do problema dual associado.

Atribuições e fixadores de expressões

Assim como na linguagem C e no Matlab, no CVX o operador de única igualdade, `=`, e o operador de dupla igualdade, `==`, possuem funções de atribuição e de igualdade, respectivamente. Para que os operadores de atribuição e de igualdade não sejam utilizados imprópriamente, o CVX usa procedimentos que garantem o seu uso de forma correta.

Considere o trecho de código a seguir:

```

variable x(n);
x = nonnegative(n);

```

À primeira vista pode parecer que a segunda linha de comando `x = nonnegative(n)` restringe x a pertencer ao ortante não negativo, porém não é isto que acontece. O operador de atribuição foi utilizado imprópriamente no lugar do operador de igualdade. Desta forma x seria carregada com um vetor anônimo pertencente ao ortante não negativo. Para evitar este equívoco o CVX não permite que seja atribuído qualquer valor numérico ou expressão a uma variável declarada, resultando na ocorrência de um erro durante a compilação do código caso tal atribuição incorreta seja feita. A atribuição ocorre internamente no CVX somente após a resolução do problema de otimização, momento em que o valor ótimo é carregado na variável.

É possível atribuir valores numéricos ou expressões para outros objetos chamados de fixadores de expressões, que são diferentes dos objetos variáveis. O trecho de código a seguir exemplifica o uso de um fixador de expressão:

```

variables x y;
z = x - y;
z >= 2;
minimize square_pos(z);

```

Neste caso z é um fixador de expressão que armazena a expressão afim $x - y$. Na sequência z é restrito a ser maior que ou igual a 2, o que implica em $x - y \geq 2$. O objetivo do

problema de otimização é **minimize square_pos(z)** que é o mesmo que **minimize square_pos(x - y)**.

Frequentemente é útil declarar um fixador de expressões, equivalente a um vetor, no qual cada posição do fixador armazena uma expressão. Para fazer isto é preciso declarar o fixador de expressões antes de utilizá-lo, através do comando **expression**, tal como exemplificado no Caso II.3 a seguir.

Caso II.3 – Utilização de fixador de expressões.

$$\text{minimize} \quad \sum_{i=1}^m x_i = \sum_{i=1}^m p_i \|s_i - \mathbf{d}\|_2^2,$$

```
cvx_begin
variable d(2)
expression x(m)
for i = 1 : m
    x(i) = p(i)*square_pos(norm(S(i,:) - d));
end
minimize sum(x)
cvx_end
```

onde $\mathbf{S} = (\mathbf{s}_1^T, \dots, \mathbf{s}_m^T)$, $\mathbf{s}_i \in \mathbb{R}^2$, $i = 1, \dots, m$.

O problema de otimização do Caso II.3 é apresentado na parte superior e sua modelagem no CVX na parte inferior. No bloco de código, a variável de otimização é o vetor \mathbf{d} de duas posições e o fixador de expressões é \mathbf{x} de m posições, onde m é uma constante dada. Dentro da estrutura de repetição do comando **for**, a i -ésima posição do fixador \mathbf{x} é carregada com a expressão convexa **p(i)*square_pos(norm(S(i,:) - d))**, onde \mathbf{p} é um vetor constante dado e \mathbf{S} é uma matriz constante dada. O problema de otimização é dado pelo comando **minimize sum(x)**.

Diferentemente de uma variável, um fixador de expressão ou de expressões pode ser escrito e sobrescrito em qualquer parte do bloco de código.

D. Conjunto de regras de Programação Convexa Disciplinada

Para utilizar o CVX é preciso seguir um conjunto de regras de programação (otimização) convexa disciplinada (DCP). Estas regras baseiam-se nos princípios básicos de análise convexa e são de fácil entendimento uma vez que já se conhece um pouco sobre análise e otimização convexa. O CVX é construído de maneira que o usuário seja forçado a seguir o conjunto de regras DCP de tal forma que um erro ocorrerá caso qualquer dessas regras seja burlada.

O conjunto de regras DCP é um conjunto de condições suficientes, mas não necessárias para estabelecer a convexidade de uma função. Sendo assim, existem funções que são convexas e que não são interpretadas como convexas pelo CVX. Como exemplo considere a função convexa $\mathbf{x}^T \mathbf{x} / y$ para $\mathbf{x} \in \mathbb{R}^n$ e $y > 0$. Se ela for escrita como **(x'*x)/y**, o CVX não a aceitará e emitirá uma mensagem de erro, pois violaram-se as regras de composição e de não-produto, assuntos a serem abordados mais adiante nesta seção. Para representar a função $\mathbf{x}^T \mathbf{x} / y$ no CVX existe uma sintaxe na biblioteca átomo dada por **quad_over_lin(x,y)**. Com esta sintaxe, a função será corretamente interpretada como convexa pelo o CVX.

Como outro exemplo, considere a função convexa $\sqrt{x^2 + y^2}$ com variáveis $x \in \mathbb{R}$ e $y \in \mathbb{R}$. Se ela for expressa no CVX como **sqr(x^2 + y^2)** ocorrerá um erro, pois esta sintaxe não segue o conjunto de regras DCP e, portanto, a função não é aceita como convexa pelo CVX. Para escrever esta função de maneira a ser aceita como convexa pelo CVX, utiliza-se a sintaxe **norm(.)** da biblioteca átomo, na qual o argumento é uma função afim, da seguinte forma: **norm([x y])**.

Curvatura de uma função escalar

Na programação convexa disciplinada uma função escalar é classificada pela sua curvatura, podendo ser constante, afim, côncava ou convexa. Para $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, a função será:

constante se $f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) = f(\mathbf{x})$, $\alpha \in \mathbb{R}$;

afim se $f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) = \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y})$, $\alpha \in \mathbb{R}$;

convexa se $f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y})$, $\alpha \in [0, 1]$;

côncava se $f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) \geq \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y})$, $\alpha \in [0, 1]$.

Funções convexas e côncavas são reais por definição. Funções afins e constantes podem ser complexas desde que não sejam utilizadas em restrições de desigualdade.

Tipos de problema de programação convexa disciplinada

O CVX suporta três tipos de programação convexa disciplinada:

- um problema de minimização, com uma função objetivo convexa e zero ou mais restrições;
- um problema de maximização, com uma função objetivo côncava e zero ou mais restrições;
- um problema de factibilidade, sem função objetivo e com restrições.

Regras de expressões

O que distingue a programação convexa disciplinada da otimização convexa mais geral são as regras que governam a construção das expressões utilizadas na função objetivo e nas restrições. A programação convexa disciplinada determina a curvatura de funções escalares aplicando recursivamente as regras apresentadas a seguir.

Uma função constante válida é:

- qualquer expressão do Matlab corretamente formulada e que resulte em um valor finito.

Uma função afim válida é:

- uma função constante válida;
- uma variável declarada;
- uma chamada válida de uma função da biblioteca átomo resultando em uma função afim;
- a soma ou diferença de funções afins;
- o produto entre uma função afim e uma constante.

Uma função convexa válida é:

- uma função afim ou constante válida;
- uma chamada válida de uma função da biblioteca átomo resultando em uma função convexa;
- um escalar afim elevado a uma potência constante $p \geq 1$, $p \neq 3, 5, 7, 9, \dots$;

- uma forma escalar quadrática convexa, (ver *Formas quadráticas escalares*);
- a soma de duas ou mais funções convexas;
- a diferença entre uma função convexa e uma função côncava;
- o produto de uma função convexa e uma constante não negativa;
- o produto de uma função côncava e uma constante não positiva;
- a negação de uma função côncava.

Uma função côncava válida é:

- uma função afim ou constante válida;
- uma chamada válida de uma função da biblioteca átomo resultando em uma função côncava;
- um escalar afim elevado a uma potência constante $p \in (0,1)$;
- uma forma escalar quadrática côncava, (ver *Formas quadráticas escalares*);
- a soma de duas ou mais funções côncavas;
- a diferença entre uma função côncava e uma função convexa;
- o produto de uma função côncava e uma constante não negativa;
- o produto de uma função convexa e uma constante não positiva;
- a negação de uma função convexa.

Se uma determinada expressão não segue estas regras ela não será aceita pelo CVX. Para vetores e matrizes de expressões (fixadores de expressões) estas regras são aplicadas elemento a elemento.

Com exceção das formas quadráticas escalares, as regras de expressões proíbem o produto de duas ou mais funções não constantes. Esta regra é chamada de regra do *não-produto*. Como exemplo, a função entropia, $-x \cdot \log(x)$, é côncava em x , mas sua concavidade não pode ser verificada utilizando o conjunto de regras DCP, sendo então rejeitada pelo CVX. Para solucionar este problema existe uma função côncava da biblioteca átomo que representa esta função, que é **entr(x)**.

Funções

No CVX, uma função pode ser classificada quanto a sua curvatura (constante, afim, côncava ou convexa) e quanto a sua monotonicidade (não decrescente, não crescente ou não monotônica). A curvatura de uma função determina em quais condições ela pode aparecer dentro de uma expressão, respeitando as regras de expressões dadas no item anterior. A monotonicidade de uma função determina como ela pode ser utilizada dentro de composições de funções, assunto que será abordado no próximo item desta seção. Uma lista de funções do CVX e suas classificações quanto a curvatura e monotonicidade pode ser encontrada em [2, pp. 53-62].

As funções convexas e côncavas no CVX são interpretadas como a *extensão de seus valores estendidos*. Isto quer dizer que os valores de funções convexas são interpretados como $+\infty$ quando os seus argumentos estão fora de seus domínios e os valores de funções côncavas são interpretados como $-\infty$ quando os seus argumentos estão fora de seus domínios.

A monotonicidade de uma função é determinada em seu sentido estendido, isto é, incluindo os valores dos argumentos fora de seu domínio. Considere por exemplo a função $\log(x)$. Esta função é não decrescente, pois para valores não positivos de x o valor da função assume o valor $-\infty$.

O CVX não considera que uma função seja convexa ou côncava somente em uma parcela de seu domínio, mesmo que o argumento desta função seja confinado nesta parcela por efeito das restrições. Como exemplo, considere a função x^3 . Esta função é convexa para $x \geq 0$ e côncava para $x \leq 0$. Assim, a sintaxe x^3 não pode ser escrita no CVX mesmo que a restrição $x \geq 0$ seja declarada em um problema de minimização ou que a restrição $x \leq 0$ seja declarada em um problema de maximização. Para expressar a parte convexa desta função no CVX utiliza-se a função **pow_pos(x,p)**, a qual representa a função $\max\{x,0\}^p$ para $x \in \mathbb{R}$ e $p \geq 1$, que é convexa e não decrescente. Escrevendo x^3 como **pow_pos(x,3)** tem-se uma função convexa e não decrescente. Escrevendo **pow_pos(-x,3)** tem-se uma função côncava não decrescente que representa a parte côncava de x^3 .

Funções com múltiplos argumentos têm sua curvatura considerada de forma conjunta, mas a monotonicidade é classificada argumento a argumento. Um exemplo é a função $x^T x / y$, que na biblioteca átomo é representada por **quad_over_lin(x,y)**. Esta função é conjuntamente convexa em x e y mas é monotonicamente não crescente somente em y .

Composições

Considere uma função de múltiplos argumentos, com curvatura e monotonicidade conhecidas. Para que se mantenha a convexidade, as regras de composição são:

- se a função é não decrescente em um argumento, aquele argumento tem que ser uma função convexa;
- se a função é não crescente em um argumento, aquele argumento tem que ser uma função côncava;
- se uma função é não monotônica em um argumento, aquele argumento tem que ser uma função afim.

Para que se mantenha a concavidade, as regras de composição são:

- se a função é não decrescente em um argumento, aquele argumento tem que ser uma função côncava;
- se a função é não crescente em um argumento, aquele argumento tem que ser uma função convexa;
- se uma função é não monotônica em um argumento, aquele argumento tem que ser uma função afim.

A partir destas regras de composição gerais, segue-se que qualquer função pode ter como argumento uma função constante ou afim (que é côncava e convexa), mantendo-se sua curvatura. Segue-se também que:

- um múltiplo não negativo de uma função convexa (côncava) é uma função convexa (côncava);
- um múltiplo não positivo de uma função convexa (côncava) é uma função côncava (convexa).

Como exemplo de cumprimento das regras de composição, considere a função a seguir sob a variável vetorial x e

constantes de dimensões apropriadas \mathbf{A} , \mathbf{F} e \mathbf{b} : $\text{norm}(\mathbf{x}) - 2 * (\text{sum}(\text{sqrt}(\mathbf{A} * \mathbf{x} + \mathbf{b})) + \text{pos}(\mathbf{F} * \mathbf{x} - \mathbf{1}) - \mathbf{1})$. Esta função é convexa para o CVX. Considere o termo $-2 * (\text{sum}(\text{sqrt}(\mathbf{A} * \mathbf{x} + \mathbf{b})))$. O CVX interpreta $\text{sqrt}(\mathbf{A} * \mathbf{x} + \mathbf{b})$ como sendo uma função côncava, pois é a composição de uma função côncava com uma função afim. Sendo assim $\text{sum}(\text{sqrt}(\mathbf{A} * \mathbf{x} + \mathbf{b}))$ também é côncava por ser a composição de uma função afim (que é côncava) e não decrescente com um argumento que é uma função côncava. Finalmente $\text{pos}(\mathbf{F} * \mathbf{x} - \mathbf{1})$ tem-se que $-2 * (\text{sum}(\text{sqrt}(\mathbf{A} * \mathbf{x} + \mathbf{b})))$ é uma função convexa por ser um múltiplo não negativo de uma função côncava. Considere agora o termo $\text{pos}(\mathbf{F} * \mathbf{x} - \mathbf{1})$. Este termo é uma função convexa, por ser a composição de uma função convexa e crescente com uma função afim. Como o termo $\text{norm}(\mathbf{x})$ é uma função convexa e $-\mathbf{1}$ é afim e, portanto, uma função convexa, tem-se que a função dada é convexa por ser a soma de quatro termos que são funções convexas.

As regras de composição são suficientes, mas não necessárias para que a classificação de uma função quanto a sua curvatura seja de convexidade. Existem funções que são convexas ou côncavas, mas que não satisfazem as regras das composições e, assim, não são aceitas pelo CVX. Um claro exemplo é a função $\text{log}(\text{sum}(\text{exp}(\mathbf{x})))$. Esta função não é aceita pelo CVX, pois não existe regra de composição para uma função côncava e não decrescente com uma função convexa. Para representar esta função de modo que ela seja aceita pelo CVX existe a função $\text{log_sum_exp}(\mathbf{x})$ da biblioteca átomo, que é convexa e não decrescente.

Monotonicidade em composições não lineares

A monotonicidade em composições não lineares é um aspecto crítico que tem como consequência a não aceitação pelo CVX de funções que de fato são côncavas ou convexas. Considere a função $\text{square}(\text{square}(\mathbf{x}) + 2)$, onde x é uma variável escalar. Esta função é convexa, porém não é aceita pelo CVX, uma vez que não existe nenhuma regra de composição de uma função convexa não monotônica com um argumento que seja uma função convexa. Uma opção para escrever corretamente esta função é $x^4 + 4 * x + 4$, que é convexa e aceita pelo CVX por ser a soma de duas funções convexas, ou seja, a soma de um escalar afim elevado a uma potência constante positiva e par, x^4 , e a função afim $4x + 4$.

Para permitir composições não lineares, existem funções na biblioteca átomo em sua forma natural e em uma forma modificada que as tornem monotônicas. Como exemplo, considere a função $\text{square_pos}(\text{square}(\mathbf{x}) + 2)$. A função $\text{square_pos}(\mathbf{x})$ representa a função $\max(x, 0)^2$ para $x \in \mathbb{R}$ e é convexa e crescente. Então, a função $\text{square_pos}(\text{square}(\mathbf{x}) + 2)$ é convexa e, portanto, é aceita pelo CVX. Esta função é equivalente à função $\text{square}(\text{square}(\mathbf{x}) + 2)$ para valores positivos do argumento do square externo.

Outras funções da biblioteca átomo se encontram na versão modificada e na versão natural. Por exemplo, $\text{sum_square_pos}(\mathbf{x})$ é convexa e crescente em x , e $\text{quad_pos_over_lin}(\mathbf{x}, \mathbf{y})$ é convexa, crescente em \mathbf{x} e decrescente em \mathbf{y} .

Formas quadráticas escalares

O conjunto de regras DCP proíbe o produto de duas funções não constantes, mesmo que seja uma simples função quadrática dada por $\mathbf{x} * \mathbf{x}$ onde x é uma variável escalar. Por razões práticas optou-se por criar exceções a estas regras de tal forma que certas funções quadráticas sejam permitidas pelo CVX. Tais funções são mapeadas diretamente em suas formas convexas ou côncavas. O quadro a seguir mostra exemplos de mapeamento de funções quadráticas do lado esquerdo para suas respectivas funções da biblioteca átomo do lado direito:

$\mathbf{y} * \mathbf{y}$	$\text{square}(\mathbf{y})$, (\mathbf{y} real)
$\text{conj}(\mathbf{w}) * \mathbf{w}$	$\text{square_abs}(\mathbf{w})$
$\mathbf{x} * \mathbf{x}$	$\text{sum_square_abs}(\mathbf{x})$
$(\mathbf{A} * \mathbf{x} - \mathbf{b}) * \mathbf{P} * (\mathbf{A} * \mathbf{x} - \mathbf{b})$	$\text{quad_form}(\mathbf{A} * \mathbf{x} - \mathbf{b}, \mathbf{P})$.

Dentro de uma expressão, o CVX examina cada produto e cada termo quadrático de funções afins, verificando a convexidade ou não destes termos. Considere a função com variáveis escalares x e y dada por $x^2 + 2 * x * y + y^2$. Esta função não é aceita pelo CVX, pois o termo $2 * x * y$ não é uma função convexa nem côncava. Esta função pode ser reescrita como $(x + y)^2$ ou como $(x + y) * (x + y)$, passando a ser aceita pelo CVX como uma função convexa.

As formas quadráticas escalares não precisam ser simétricas. Por exemplo, $(\mathbf{x} + \mathbf{a}) * \mathbf{P} * (\mathbf{x} + \mathbf{b})$ é válida para o CVX. Nela \mathbf{x} é uma variável vetorial, \mathbf{a} e \mathbf{b} são constantes e \mathbf{P} é uma matriz definida positiva.

Sempre que possível deve-se optar por utilizar formas não quadráticas no lugar de formas quadráticas. Por exemplo, utilize $\text{norm}(\mathbf{A} * \mathbf{x} - \mathbf{b})$ em vez de $\text{sum}((\mathbf{A} * \mathbf{x} - \mathbf{b}) . ^2)$. A razão para isto é que as formas não quadráticas são tratadas com mais precisão no CVX do que as formas quadráticas.

III. PROBLEMAS DE OTIMIZAÇÃO CONVEXA

Nesta seção são apresentados alguns dos principais problemas de otimização convexa e a correspondente implementação no CVX. São exemplificadas as seguintes instâncias:

- problemas com uma única restrição de igualdade e problemas sem restrições;
- problemas de factibilidade;
- problemas no formato epígrafo;
- otimização quase-convexa;
- otimização linear (LP);
- otimização quadrática (QP) e otimização quadrática restrita quadraticamente (QCQP);
- otimização cônica de segunda ordem (SOCP);
- otimização geométrica (GP);
- otimização cônica;
- otimização semidefinida (SDP);
- escalarização de problemas de otimização vetorial e
- escalarização de problemas de otimização multicritério.

A. Problemas de otimização convexa

Um problema de otimização convexa na forma padrão é

$$\begin{aligned} & \text{minimize } f_0(\mathbf{x}) \\ & \text{sujeito a } f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \quad h_i(\mathbf{x}) = \mathbf{a}_i^T \mathbf{x} - b_i = 0, \quad i = 1, \dots, p \end{aligned} \quad (2)$$

onde f_0, \dots, f_m são funções convexas, h_i é afim e $\mathcal{D} = \bigcap_{i=0}^m \text{dom}(f_i)$.

O conjunto factível de um problema de otimização convexa é convexo, uma vez que ele é a interseção do domínio do problema \mathcal{D} , que é um conjunto convexo, com m conjuntos sub-níveis $\{\mathbf{x} \mid f_i(\mathbf{x}) \leq 0\}$ e p hiperplanos $\{\mathbf{x} \mid \mathbf{a}_i^T \mathbf{x} = b_i\}$. Se $f_0(\mathbf{x})$ em (2) for quase-convexa, o problema passa a ser um problema de otimização quase-convexa.

Outra forma de problema de otimização convexa é o problema de maximização

$$\begin{aligned} & \text{maximize } f_0(x) \\ & \text{sujeito a } f_i(x) \leq 0, \quad i = 1, \dots, m \\ & \quad h_i(x) = \mathbf{a}_i^T \mathbf{x} - b_i = 0, \quad i = 1, \dots, p \end{aligned} \quad (3)$$

onde $f_0(\mathbf{x})$ é uma função côncava, f_1, \dots, f_m são convexas e h_i são afins. Este problema de maximização pode ser resolvido diretamente por um problema de minimização com função objetivo $-f_0(\mathbf{x})$. Se $f_0(\mathbf{x})$ em (3) for uma função quase-côncava, passa-se a ter um problema de otimização quase-côncava.

Outros problemas comumente encontrados são os problemas de otimização convexa sem restrições ou somente com restrições de desigualdade. Um problema de otimização convexa sem restrições possui somente a função objetivo, ou seja, equivale ao problema em (2) com m e p iguais à zero. O Caso III.1 exemplifica este problema.

Caso III.1 – Otimização quadrática sem restrições.

$$\text{minimize } \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r$$

```
n = size(P,2);
cvx_begin
    variable x(n)
    minimize 0.5*x'*P*x + q'*x + r
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{P} \in \mathbb{S}_+^n$ é uma matriz dada, $\mathbf{q} \in \mathbb{R}^n$ é um vetor dado e $r \in \mathbb{R}$ é um escalar dado. Neste problema algumas situações podem ocorrer:

- Se $\mathbf{q} \notin \mathcal{R}(\mathbf{P})$ [12, p. 114], não existirá solução para o problema e ele será ilimitado para baixo;
- Se $\mathbf{P} \in \mathbb{S}_+^n$, existirá um único minimizador para o problema, $\mathbf{x}_{\text{ótimo}} = -\mathbf{P}^{-1} \mathbf{q}$.
- Se \mathbf{P} é singular e $\mathbf{q} \in \mathcal{R}(\mathbf{P})$, o conjunto de pontos ótimos é o conjunto afim $\mathbf{x}_{\text{ótimo}} = -\mathbf{P}^\dagger \mathbf{q} + \mathcal{N}(\mathbf{P})$, onde \mathbf{P}^\dagger é a pseudo-inversa de \mathbf{P} [1, p.649] e $\mathcal{N}(\mathbf{P})$ é o espaço nulo de \mathbf{P} [12, p. 122].

Um problema de otimização convexa somente com restrições de igualdade equivale ao problema em (2) com $m = 0$, como mostra o Caso III.2.

Caso III.2 – Problema de otimização convexa somente com restrições de igualdade.

$$\begin{aligned} & \text{minimize } \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a } \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned}$$

```
n = size(A,2);
cvx_begin
    variable x(n)
    minimize c'*x
    subject to
        A*x == b;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^m$ e $\mathbf{c} \in \mathbb{R}^n$ são vetores dados. Neste problema, se $\mathbf{b} \in \mathcal{R}(\mathbf{A})$ o problema é factível e se $\mathbf{b} \notin \mathcal{R}(\mathbf{A})$ o problema é infactível.

B. Problemas de factibilidade

Um problema de factibilidade corresponde ao problema (2) com $f_0(\mathbf{x}) = 0$. O valor ótimo será zero se o problema for factível, ou infinito se o problema for infactível. O Caso III.3 ilustra a modelagem de um problema de factibilidade.

Caso III.3 – Problema de factibilidade.

$$\begin{aligned} & \text{minimize } \mathbf{0} \\ & \text{sujeito a } \mathbf{A} \mathbf{x} \preceq \mathbf{b} \\ & \quad \mathbf{x} \succeq \mathbf{0} \end{aligned}$$

```
n = size(A,2);
cvx_begin
    variable x(n)
    minimize 0
    subject to
        A*x <= b;
        x >= 0;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^m$ é um vetor dado.

O objetivo de um problema de factibilidade como este é determinar se as restrições do problema, retornando em seguida um ponto factível (caso seja factível).

C. Problemas no formato epígrafo

O formato epígrafo do problema de otimização (2) é

$$\begin{aligned} & \text{minimize } t \\ & \text{sujeito a } f_0(\mathbf{x}) - t \leq 0 \\ & \quad f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \quad \mathbf{a}_i^T \mathbf{x} - b_i = 0, \quad i = 1, \dots, p. \end{aligned} \quad (4)$$

A representação no formato epígrafo está por trás de uma das mais usuais regras de “linearização” de um problema de otimização convexa, na qual uma função objetivo qualquer é convertida em uma função objetivo linear. O problema no

formato epígrafo é um problema de otimização convexa, pois sua função objetivo, t , é linear e, portanto, convexa, e a nova função de restrição $f_0(\mathbf{x}) - t$ é também convexa em (\mathbf{x}, t) . Para entender que a função $t \in \mathbb{R}$ é linear em relação às variáveis $\mathbf{x} \in \mathbb{R}^n$ e t , veja que se pode escrever $t = \mathbf{e}^T(\mathbf{x}, t)$, sendo $\mathbf{e} \in \mathbb{R}^{n+1}$ um vetor com os primeiros n elementos iguais a 0 e o último igual a 1. Como exemplo, o Caso III.4 corresponde ao problema no formato epígrafo do problema do Caso II.1.

Caso III.4 – Problema no formato epígrafo.

$$\begin{aligned} & \text{minimize } t \\ & \text{sujeito a } \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 - t \leq 0 \\ & \mathbf{x} \succeq \mathbf{0} \end{aligned}$$

```
cvx_begin
variables x(n) t;
minimize t
subject to
norm(A*x-b) - t <= 0;
x >= 0;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ e $t \in \mathbb{R}$ são as variáveis do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^m$ é um vetor dado. As variáveis \mathbf{x} e t são ótimas para este problema se e somente se a variável \mathbf{x} for ótima para o problema do Caso II.1 e vice-versa.

D. Problemas de otimização quase-convexa

Um problema de otimização quase-convexa na forma padrão corresponde ao problema (2) com uma função objetivo quase-convexa [2, pp. 67-78].

Os conjuntos subníveis de uma função $f_0(\mathbf{x})$ quase-convexa (que são convexos) podem ser representados por desigualdades de funções convexas. Estas funções convexas constituem uma família de funções $\phi_t : \mathbb{R}^n \rightarrow \mathbb{R}$ indexadas por $t \in \mathbb{R}$, tal que $f_0(\mathbf{x}) \leq t \Leftrightarrow \phi_t(\mathbf{x}) \leq 0$, ou seja, o conjunto t -subnível de uma função quase-convexa $f_0(\mathbf{x})$ é o conjunto 0-subnível de uma função $\phi_t(\mathbf{x})$. Para valores maiores de t têm-se valores menores de $\phi_t(\mathbf{x})$, ou seja, para cada \mathbf{x} , $\phi_t(\mathbf{x})$ é uma função não crescente de t . Utilizando a função $\phi_t(\mathbf{x})$, pode-se escrever o problema de factibilidade

$$\begin{aligned} & \text{minimize } 0 \\ & \text{sujeito a } \phi_t(\mathbf{x}) \leq 0 \\ & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \mathbf{A}\mathbf{x} = \mathbf{b}, \end{aligned} \quad (5)$$

que se for factível implica que t está acima do valor ótimo do problema de otimização quase-convexa e se for infactível implica que t está abaixo do valor ótimo do problema de otimização quase-convexa.

Através de um algoritmo conhecido como método da bissecção [1, p. 145] é possível resolver um problema de otimização quase-convexa resolvendo um problema de factibilidade como em (4) a cada passo, de tal forma que depois de um número de passos, tem-se a solução do problema

de otimização quase-convexa alcançada dentro de uma margem de erro pré-definida.

O Caso III.5 a seguir corresponde a um problema de factibilidade que pode ser utilizado pelo método da bissecção para resolver um problema de otimização quase-convexa sem restrições, para o qual a função objetivo é $f_0(x) = (x^2 + 12) / (10 - x)^{1/2}$ com $\mathcal{D} = \{x \in \mathbb{R} \mid x < 10\}$.

Caso III.5 – Problema de factibilidade utilizado pelo método da bissecção para resolver um dado problema de otimização quase-convexa.

$$\begin{aligned} & \text{minimize } 0 \\ & \text{sujeito a } \phi_t(x) = (x^2 + 12) - t(10 - x)^{1/2} \leq 0 \end{aligned}$$

```
cvx_begin
variables x;
minimize 0
subject to
x^2 + 12 - t*sqrt(10-x) <= 0;
cvx_end
```

onde $x \in \mathbb{R}$ é a variável de otimização.

E. Problemas de otimização linear

Um problema de otimização é chamado de problema de otimização linear (LP) quando a função objetivo e as funções restrições são todas afins. A forma geral de um problema de otimização linear é

$$\begin{aligned} & \text{minimize } \mathbf{c}^T \mathbf{x} + d \\ & \text{sujeito a } \mathbf{G}\mathbf{x} \preceq \mathbf{h} \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned} \quad (6)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{G} \in \mathbb{R}^{m \times n}$ e $\mathbf{A} \in \mathbb{R}^{p \times n}$ são matrizes dadas, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{h} \in \mathbb{R}^m$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados e $d \in \mathbb{R}$ é um escalar dado.

É comum omitir a constante d da função objetivo uma vez que ela não afeta o conjunto de pontos factíveis ou de pontos ótimos do problema, embora afete o valor ótimo do mesmo.

Uma vez que maximizar a função objetivo $\mathbf{c}^T \mathbf{x} + d$ corresponde a minimizar a função $-\mathbf{c}^T \mathbf{x} - d$, que também é afim, o problema de maximização de uma função afim também é chamado de LP.

Dois casos particulares do LP (6) são muito encontrados na prática: o LP na forma padrão e o LP na forma de desigualdade. Para o LP na forma padrão as restrições de desigualdade são restrições de não negatividade componente a componente da variável de otimização \mathbf{x} , ou seja, $\mathbf{x} \succeq \mathbf{0}$, como mostrado a seguir:

$$\begin{aligned} & \text{minimize } \mathbf{c}^T \mathbf{x} + d \\ & \text{sujeito a } \mathbf{x} \succeq \mathbf{0} \\ & \mathbf{A}\mathbf{x} = \mathbf{b}. \end{aligned} \quad (7)$$

Na forma de desigualdade, o LP não tem restrições de igualdade e é geralmente escrito como:

$$\begin{aligned} & \text{minimize } \mathbf{c}^T \mathbf{x} + d \\ & \text{sujeito a } \mathbf{A}\mathbf{x} \preceq \mathbf{b}. \end{aligned} \quad (8)$$

Os Casos III.6, III.7 e III.8 a seguir exemplificam modelagens no CVX de problemas LP na forma geral, padrão e de desigualdade, respectivamente.

Caso III.6 – Problema LP na forma geral.

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && \mathbf{G} \mathbf{x} \preceq \mathbf{h} \\ & && \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned}$$

```
cvx_begin
variables x(n);
minimize c'*x
subject to
G*x <= h;
A*x == b;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{G} \in \mathbb{R}^{m \times n}$ e $\mathbf{A} \in \mathbb{R}^{p \times n}$ são matrizes dadas, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{h} \in \mathbb{R}^m$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados.

Caso III.7 – Problema LP na forma padrão.

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && \mathbf{A} \mathbf{x} = \mathbf{b} \\ & && \mathbf{x} \succeq \mathbf{0} \end{aligned}$$

```
cvx_begin
variables x(n);
minimize c'*x
subject to
A*x == b;
x >= 0;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{p \times n}$ é uma matriz dada, $\mathbf{c} \in \mathbb{R}^n$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados.

Caso III.8 – Problema LP na forma de desigualdade.

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && \mathbf{A} \mathbf{x} \preceq \mathbf{b} \end{aligned}$$

```
cvx_begin
variables x(n);
minimize c'*x
subject to
A*x <= b;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{p \times n}$ é uma matriz dada, $\mathbf{c} \in \mathbb{R}^n$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados.

F. Problemas de otimização quadrática

O problema de otimização convexa (2) é chamado de problema de otimização quadrática (QP) quando a função objetivo é quadrática e, portanto, convexa, e as restrições são funções afins, como exemplo:

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r \\ & \text{sujeito a} && \mathbf{G} \mathbf{x} \preceq \mathbf{h} \\ & && \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned} \quad (9)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{P} \in \mathbb{S}_+^n$, $\mathbf{G} \in \mathbb{R}^{m \times n}$ e $\mathbf{A} \in \mathbb{R}^{p \times n}$ são matrizes dadas, $\mathbf{q} \in \mathbb{R}^n$, $\mathbf{h} \in \mathbb{R}^m$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados e $r \in \mathbb{R}$ é um escalar dado.

Se a função objetivo e as funções associadas às restrições de desigualdade do problema (2) são quadráticas e, portanto convexas, tem-se um problema de otimização quadrática restrito de forma quadrática (QCQP). Um QCQP tem a seguinte forma:

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{x}^T \mathbf{P}_0 \mathbf{x} + \mathbf{q}_0^T \mathbf{x} + r_0 \\ & \text{sujeito a} && \frac{1}{2} \mathbf{x}^T \mathbf{P}_i \mathbf{x} + \mathbf{q}_i^T \mathbf{x} + r_i \leq 0, \quad i = 1, \dots, m \\ & && \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned} \quad (10)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{p \times n}$ e $\mathbf{P}_i \in \mathbb{S}_+^n$, $i = 1, 2, \dots, m$ são matrizes dadas, $\mathbf{q}_i \in \mathbb{R}^n$, $\mathbf{h} \in \mathbb{R}^m$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados e $r_i \in \mathbb{R}$ são escalares dados.

O QP (9) torna-se um LP se $\mathbf{P} = \mathbf{0}$, o que permite concluir que o QP tem o LP como caso especial. O QCQP em (10) torna-se um QP se $\mathbf{P}_i = \mathbf{0}$ ($i = 1, 2, \dots, m$), concluindo-se que o QCQP tem como caso especial o QP e o LP.

Os Casos III.9 e III.10 a seguir são modelagens no CVX de um QP sem restrições e de um QCQP, respectivamente.

Caso III.9 – Problema QP sem restrições.

$$\text{minimize} \quad \|\mathbf{A} \mathbf{x} - \mathbf{b}\|_2^2$$

```
cvx_begin
variable x(n);
minimize square_pos(norm(A*x - b))
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{p \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^p$ é um vetor dado. Este problema é também conhecido como aproximação dos mínimos quadrados (LS). O LS é um problema de otimização simples e possui solução analítica dada por $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}$, onde \mathbf{A}^\dagger é a pseudo-inversa [1, p. 649] de \mathbf{A} .

Caso III.10 – Problema QCQP.

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} \\ & \text{sujeito a} && \frac{1}{2} \mathbf{x}^T \mathbf{W} \mathbf{x} \leq 0 \\ & && \mathbf{A} \mathbf{x} = \mathbf{b} \end{aligned}$$

```
cvx_begin
variable x(n);
minimize 0.5*norm(Psqrt*x) + q'*x
subject to
0.5*norm(Wsqrt*x) <= 0;
```

```
A*x == b;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{P} \in \mathbb{S}_+^n$, $\mathbf{W} \in \mathbb{S}_+^n$ e $\mathbf{A} \in \mathbb{R}^{p \times n}$ são matrizes dadas, $\mathbf{q} \in \mathbb{R}^n$ e $\mathbf{b} \in \mathbb{R}^p$ são vetores dados. No código em CVX, `Psqrt` é a raiz quadrada matricial de \mathbf{P} realizada de forma adequada ao CVX. A mesma ideia se aplica a `Wsqr`.

Observe que, para representar as funções quadráticas do problema QCQP no CVX foi utilizada a função `norm`. Isto foi feito seguindo as diretrizes de [2, pp. 67], que recomenda que, quando possível, se construam modelos de problemas equivalentes utilizando a função `norm` ao invés de se utilizar as formas quadráticas, tal como a função `quad_form`. Esta recomendação é dada pois as formas quadráticas representam, na maioria das vezes, um desafio numérico para os solvers utilizados pelo CVX, tendo como consequência soluções menos precisas do que as soluções de problemas equivalentes utilizando a função `norm`.

G. Problemas de otimização cônica de segunda ordem

Um problema de otimização cônica de segunda ordem (SOCP) tem a seguinte forma:

$$\begin{aligned} & \text{minimize} && \mathbf{f}^T \mathbf{x} \\ & \text{sujeito a} && \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T \mathbf{x} + d_i, \quad i = 1, \dots, m \\ & && \mathbf{F} \mathbf{x} = \mathbf{g} \end{aligned} \quad (11)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A}_i \in \mathbb{R}^{n_i \times n}$ e $\mathbf{F} \in \mathbb{R}^{p \times n}$ são matrizes dadas, $\mathbf{f}, \mathbf{b}_i, \mathbf{c}_i \in \mathbb{R}^n$ e $\mathbf{g} \in \mathbb{R}^p$ são vetores dados e $d_i \in \mathbb{R}$ são escalares dados.

Restrições de desigualdade tais como as do problema (11) são chamadas de restrições cônicas de segunda ordem, uma vez que impô-las é mesmo que exigir que os pontos $(\mathbf{A}_i \mathbf{x} + \mathbf{b}_i, \mathbf{c}_i^T \mathbf{x} + d_i)$ pertençam à interseção dos respectivos cones de segunda ordem no espaço \mathbb{R}^{n+1} [1, p. 31].

Fazendo $\mathbf{c}_i = 0, i = 1, \dots, m$ e elevando a função objetivo e as restrições ao quadrado, o SOCP em (11) passa a equivaler a um QCQP. Analogamente, se $\mathbf{A}_i = 0, i = 1, \dots, m$, o SOCP em (11) reduz-se a um LP na forma geral. Conclui-se então que problemas SOCPs são mais gerais que QCQPs e LPs.

O Caso III.11 exemplifica um problema SOCP e sua modelagem no CVX.

Caso III.11 – Problema SOCP.

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && \bar{\mathbf{a}}^T \mathbf{x} + \|\mathbf{P}^T \mathbf{x}\|_2 \leq b \end{aligned}$$

```
cvx_begin
variable x(n);
minimize c'*x
subject to
a_med'*x + norm(P'*x) <= b;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{P} \in \mathbb{R}^n$ é uma matriz dada, $\mathbf{c}, \bar{\mathbf{a}} \in \mathbb{R}^n$ são vetores dados e $b \in \mathbb{R}$ é um escalar dado.

H. Problemas de otimização geométrica

Os problemas de otimização geométrica (GP) não são problemas de otimização convexa quando estão em sua forma natural. Para entender essa forma natural é preciso antes entender o que é uma função monomial (ou monômio) e uma função posinomial (ou posinômio). Uma função monomial é uma função $f: \mathbb{R}^n \rightarrow \mathbb{R}$ com $\text{dom}(f) = \mathbb{R}_{++}^n$ definida como

$$f(\mathbf{x}) = c x_1^{a_1} x_2^{a_2} \dots x_n^{a_n} \quad (12)$$

com $c > 0$ e $a_i \in \mathbb{R}$. A soma de funções monomiais resulta em uma função posinomial que tem a forma

$$f(\mathbf{x}) = \sum_{k=1}^K c_k x_1^{a_{1k}} x_2^{a_{2k}} \dots x_n^{a_{nk}} \quad (13)$$

em que $c_k > 0$ e $a_{ik} \in \mathbb{R}$.

Posinômios são fechados por soma, multiplicação, adição e multiplicação por escalares não negativos. Monômios são fechados para multiplicação e divisão. A multiplicação de dois monômios resulta em um posinômio e a divisão de um posinômio por um monômio resulta em um posinômio.

Um GP na forma natural, também chamado de GP na forma posinomial, é assim representado:

$$\begin{aligned} & \text{minimize} && f_0(\mathbf{x}) \\ & \text{sujeito a} && f_i(\mathbf{x}) \leq 1, \quad i = 1, \dots, m \\ & && h_i(\mathbf{x}) \leq 1 \quad i = 1, \dots, p \end{aligned} \quad (14)$$

em que f_0, \dots, f_m são posinômios e h_1, \dots, h_p são monômios. O domínio deste problema é $\mathcal{D} = \mathbb{R}_{++}^n$ e a restrição $\mathbf{x} \succ 0$ é implícita.

Fazendo $x_i = e^{y_i}$, $c = e^b$ para os monômios e $c_k = e^{b_k}$ para os posinômios, pode-se expressar o problema em (14) sob a variável de otimização \mathbf{y} da seguinte forma:

$$\begin{aligned} & \text{minimize} && \sum_{k=1}^{K_0} e^{\mathbf{a}_{0k}^T \mathbf{y} + b_{0k}} \\ & \text{sujeito a} && \sum_{k=1}^{K_i} e^{\mathbf{a}_{ik}^T \mathbf{y} + b_{ik}} \leq 1, \quad i = 1, \dots, m \\ & && e^{\mathbf{g}_i^T \mathbf{y} + h_i} = 1, \quad i = 1, \dots, p \end{aligned} \quad (15)$$

com $\mathbf{a}_{ik} \in \mathbb{R}^n, i = 0, \dots, m$ contendo os expoentes da função objetivo e das restrições de desigualdade posinomiais e $\mathbf{g}_{ik} \in \mathbb{R}^n, i = 0, \dots, p$ contendo os expoentes das restrições de igualdade monomiais do problema em (15).

Extraíndo o logaritmo da função objetivo e das restrições do problema (15) obtém-se o novo problema:

$$\begin{aligned} \text{minimize } & \tilde{f}_0(\mathbf{y}) = \log \left(\sum_{k=1}^{K_0} e^{a_{0k}^T \mathbf{y} + b_{0k}} \right) \\ \text{sujeito a } & \tilde{f}_i(\mathbf{y}) = \log \left(\sum_{k=1}^{K_i} e^{a_{ik}^T \mathbf{y} + b_{ik}} \right) \leq 0, \quad i = 1, \dots, m \quad (16) \\ & \tilde{h}_i(\mathbf{y}) = \mathbf{g}_i^T \mathbf{y} + h_i = 0, \quad i = 1, \dots, p. \end{aligned}$$

Como as funções \tilde{f}_0 e $\{\tilde{f}_i\}$ são convexas e $\{\tilde{h}_i\}$ são afins, o problema (16) trata-se um problema de otimização convexa e é chamado de problema de otimização geométrico na forma convexa. Utilizando o valor e o ponto ótimos do problema (16) chega-se ao valor ótimo e ao ponto ótimo do problema (15) e vice-versa, pois $x_i = e^{y_i}$ e $\tilde{f}_0(\mathbf{y}) = \log f_0(\mathbf{y})$.

O Caso III.12 modela um GP em sua forma posinomial utilizando o modo de programação geométrica do CVX.

Caso III.12 – Problema GP na forma posinomial.

$$\begin{aligned} \text{minimize } & l \\ \text{sujeito a } & \sum_{j=1}^2 \mathbf{A}_{ij} \mathbf{v}_j / (h_{vi}) \leq 1, \quad i = 1, 2 \end{aligned}$$

```
cvx_begin gp
variables v(2) l
minimize l
subject to
(A(1,1)*v(1) + A(1,2)*v(2))/(l*v(1)) <= 1;
(A(2,1)*v(1) + A(2,2)*v(2))/(l*v(2)) <= 1;
cvx_end
```

onde $l \in \mathbb{R}$ e $\mathbf{v} \in \mathbb{R}^2$ são as variáveis de otimização e $\mathbf{A} \in \mathbb{R}^{2 \times 2}$ é uma matriz dada.

A sintaxe **gp** utilizada no código tem a função de configurar o CVX para trabalhar no modo de otimização geométrica [2, pp. 41-42]. Neste modo, problemas de otimização geométrica na sua forma natural tal como (14) são adequadamente interpretados pelo CVX, que automaticamente realiza a conversão necessária para a forma convexa, computa a solução e converte o resultado para o problema original.

I. Problemas com restrições de desigualdade generalizadas

Uma generalização do problema de otimização convexa na forma padrão (2) é obtida permitindo que as restrições de desigualdade estejam associadas a desigualdades generalizadas em relação a cones próprios quaisquer:

$$\begin{aligned} \text{minimize } & f_0(\mathbf{x}) \\ \text{sujeito a } & f_i(\mathbf{x}) \preceq_{K_i} 0, \quad i = 1, \dots, m \quad (17) \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned}$$

em que $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$, $K_i \subseteq \mathbb{R}^{k_i}$ são cones próprios [1, pp. 43-46], $f_i : \mathbb{R}^n \rightarrow \mathbb{R}^{k_i}$ são funções K_i -convexas [1, pp. 109, 167]. Este problema é chamado de problema de otimização convexa com desigualdades generalizadas na forma padrão. O problema em (2) é um caso especial do problema em (17) que ocorre quando $K_i = \mathbb{R}_+, i = 1, \dots, m$.

J. Problemas de otimização cônica

Dentre os mais simples problemas de otimização convexa com desigualdades generalizadas estão os problemas de otimização cônica, os quais têm uma função objetivo afim e uma função de restrição de desigualdade também afim e, portanto, K -convexas. Um problema de otimização cônica na forma geral é assim escrito:

$$\begin{aligned} \text{minimize } & \mathbf{c}^T \mathbf{x} \\ \text{sujeito a } & \mathbf{F}\mathbf{x} + \mathbf{g} \preceq_K \mathbf{0} \quad (18) \\ & \mathbf{A}\mathbf{x} = \mathbf{b}, \end{aligned}$$

e se K é o ortante não negativo, o problema de otimização cônica (18) se reduz ao LP na forma geral (6).

A forma padrão de um problema de otimização cônica é:

$$\begin{aligned} \text{minimize } & \mathbf{c}^T \mathbf{x} \\ \text{sujeito a } & \mathbf{x} \succeq_K \mathbf{0} \quad (19) \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned}$$

Já um problema de otimização cônica em sua forma de desigualdade é:

$$\begin{aligned} \text{minimize } & \mathbf{c}^T \mathbf{x} \\ \text{sujeito a } & \mathbf{F}\mathbf{x} + \mathbf{g} \preceq_K \mathbf{0}. \quad (20) \end{aligned}$$

Os Casos III.6, III.7 e III.8 já apresentados podem ser entendidos como modelagens no CVX de problemas de otimização cônica com $K = \mathbb{R}_+$ nas formas geral, padrão e de desigualdade, respectivamente.

K. Problemas de otimização semidefinida

Se K é \mathbb{S}_+^k , o cone de matrizes semidefinidas de ordem $k \times k$, designa-se o correspondente problema de otimização cônica por problema de otimização (ou programação) semidefinida (SDP), cuja forma geral é:

$$\begin{aligned} \text{minimize } & \mathbf{c}^T \mathbf{x} \\ \text{sujeito a } & x_1 \mathbf{F}_1 + \dots + x_n \mathbf{F}_n + \mathbf{G} \preceq 0 \quad (21) \\ & \mathbf{A}\mathbf{x} = \mathbf{b}, \end{aligned}$$

sendo $\mathbf{x} \in \mathbb{R}^n$ a variável do problema, $\mathbf{G}, \mathbf{F}_1, \dots, \mathbf{F}_n \in \mathbb{S}^k$, e $\mathbf{A} \in \mathbb{R}^{p \times n}$. A restrição de desigualdade do problema é uma restrição de desigualdade matricial linear (LMI) [1, p. 38]. Se as matrizes $\mathbf{G}, \mathbf{F}_1, \dots, \mathbf{F}_n$ são diagonais, então a LMI em (21) passa a ser um conjunto de n desigualdades lineares e o SDP (21) se reduz a um LP.

Um SDP na forma padrão possui restrições de igualdade lineares e uma restrição de não negatividade sob a variável matricial $\mathbf{X} \in \mathbb{S}^n$, tal como em

$$\begin{aligned} \text{minimize } & \text{tr}(\mathbf{C}\mathbf{X}) \\ \text{sujeito a } & \text{tr}(\mathbf{A}_i \mathbf{X}) = b_i, \quad i = 1, \dots, p \quad (22) \\ & \mathbf{X} \succeq 0 \end{aligned}$$

com $\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p \in \mathbb{S}^n$, lembrando que

$\text{tr}(\mathbf{C}\mathbf{X}) = \sum_{i,j=1}^n C_{ij} X_{ij}$ é a forma geral de uma função linear de valor real em \mathbb{S}^n .

Um SDP na forma de desigualdade possui uma LMI e nenhuma restrição de igualdade, tal como em

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n \preceq \mathbf{B}, \end{aligned} \quad (23)$$

com variável $\mathbf{x} \in \mathbb{R}^n$, matrizes dadas $\mathbf{B}, \mathbf{A}_1, \dots, \mathbf{A}_p \in \mathbb{S}^k$ e vetor dado $\mathbf{c} \in \mathbb{R}^n$.

Os Casos III.13 e III.14 em seguida correspondem a exemplos de modelagens no CVX de problemas de otimização cônica. O problema do Caso III.13 possui K correspondente ao cone de segunda ordem e o problema do Caso III.14 é um SDP na forma de desigualdade.

Caso III.13 – Problema de otimização cônica referente ao cone de segunda ordem.

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && -(\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{d}^T \mathbf{x} + f) \preceq_K \mathbf{0} \\ & && \mathbf{F}\mathbf{x} = \mathbf{g} \end{aligned}$$

```
n = size(A,2);
m = size(A,1);
cvx_begin
    variable x(n)
    minimize c'*x
    subject to
        {A*x+b, d'*x+f} <In> lorentz(m);
        F*x == g;
cvx_end
```

onde $K = \{(\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{d}^T \mathbf{x} + f) \in \mathbb{R}^{m+1} \mid \|\mathbf{A}\mathbf{x} + \mathbf{b}\|_2 \leq \mathbf{d}^T \mathbf{x} + f\}$, $\mathbf{A} \in \mathbb{R}^{m \times n}$ e $\mathbf{F} \in \mathbb{R}^{p \times n}$ são matrizes dadas, $\mathbf{c}, \mathbf{d} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ e $\mathbf{g} \in \mathbb{R}^p$ são vetores dados e $f \in \mathbb{R}$ é um escalar dado. Note que o termo à esquerda na restrição de desigualdade é um vetor em \mathbb{R}^{m+1} e pode ser interpretado como o termo à esquerda da restrição de desigualdade em (18).

Caso III.14 – Problema SDP na forma de desigualdade.

$$\begin{aligned} & \text{minimize} && t \\ & \text{sujeito a} && \begin{bmatrix} t\mathbf{I} & \mathbf{A} \\ \mathbf{A}^T & t\mathbf{I} \end{bmatrix} \succeq \mathbf{0} \end{aligned}$$

```
n = size(A,2)
cvx_begin sdp
    variable t
    minimize t
    subject to
        [t*eye(4,4) A; A' t*eye(4,4)] >= 0
cvx_end
```

onde $t \in \mathbb{R}$ é a variável de otimização e $\mathbf{A} \in \mathbb{R}^{n \times n}$ é uma matriz dada.

Como já mencionado, a sintaxe `cvx_begin sdp` utilizada no código tem a função de configurar o CVX para trabalhar no modo de otimização semidefinida ou modo SDP [2, pp. 39-40]. Nesse modo, operadores de desigualdades padrão do Matlab

como \succeq e \preceq são interpretados com desigualdades matriciais. Como exemplo, se em uma determinada modelagem no CVX no modo SDP existir, a restrição $\mathbf{X} \succeq \mathbf{0}$ implica que $\mathbf{X} \in \mathbb{S}_+^n$.

L. Problemas de otimização convexa vetorial

Um problema de otimização vetorial tem a seguinte forma

$$\begin{aligned} & \text{minimize} && f_0(\mathbf{x}) \text{ (em relação a } K) \\ & \text{sujeito a} && f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & && h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p \end{aligned} \quad (24)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável de otimização, $K \subseteq \mathbb{R}^q$ é um cone próprio, $f_0: \mathbb{R}^n \rightarrow \mathbb{R}^q$ é a função objetivo K -convexa, $f_i: \mathbb{R}^n \rightarrow \mathbb{R}$ são as funções correspondentes às restrições de desigualdade convexas e $h_i: \mathbb{R}^n \rightarrow \mathbb{R}$ são funções correspondentes às restrições de igualdade afins, que, como em (2) podem ser representadas por $\mathbf{A}\mathbf{x} = \mathbf{b}$, com $\mathbf{A} \in \mathbb{R}^{p \times n}$.

Para entender o significado de um problema de otimização vetorial, considere que os pontos \mathbf{x} e \mathbf{y} sejam dois pontos factíveis para o problema (24). Os correspondentes valores da função objetivo, $f_0(\mathbf{x})$ e $f_0(\mathbf{y})$, agora estão no espaço \mathbb{R}^q e são comparáveis utilizando a desigualdade generalizada \preceq_K e não a desigualdade ordinária em \mathbb{R} . Se $f_0(\mathbf{x}) \preceq_K f_0(\mathbf{y})$, significa que \mathbf{x} é igual ou melhor do que \mathbf{y} do ponto de vista do problema de otimização. Entretanto, na otimização vetorial dois valores da função objetivo não precisam ser comparáveis, ou seja, pode-se não ter $f_0(\mathbf{x}) \preceq_K f_0(\mathbf{y})$ nem $f_0(\mathbf{y}) \preceq_K f_0(\mathbf{x})$. É usual representar esta situação por $f_0(\mathbf{x}) \not\preceq_K f_0(\mathbf{y})$. Isto não acontece em um problema de otimização cujo valor da função objetivo é escalar, como em todos os problemas de otimização anteriormente tratados neste tutorial, por exemplo.

A maioria dos problemas de otimização vetorial não possui somente um ponto ótimo, embora isso possa acontecer em alguns casos especiais. Para esta maioria existe um conjunto de pontos factíveis que levam a função objetivo aos seus melhores valores. Aos elementos desse conjunto dá-se o nome de minimals [1, pp. 177-178]. Tais elementos são conhecidos como pontos eficientes ou pontos ótimos de Pareto [1, pp. 177-178]. Os valores da função objetivo para os pontos ótimos de Pareto são chamados de valores ótimos de Pareto. Um ponto \mathbf{x} é um ponto ótimo de Pareto se ele e um ponto factível do problema e, para qualquer \mathbf{y} factível, $f_0(\mathbf{y}) \preceq_K f_0(\mathbf{x})$ implica que $f_0(\mathbf{y}) = f_0(\mathbf{x})$. Todo valor ótimo de Pareto é um valor alcançável que se localiza na fronteira do conjunto de valores alcançáveis do problema de otimização vetorial.

M. Escalarização de problemas de otimização vetorial

Para encontrar os pontos ótimos de Pareto de um problema de otimização convexa vetorial utiliza-se uma técnica chamada de escalarização, baseada na caracterização de elementos mínimos e minimals via desigualdades generalizadas duais [1, pp. 54-58]. A técnica consiste em encontrar a solução do problema de otimização convexa escalar:

$$\begin{aligned} & \text{minimize} \quad \lambda^T f_0(\mathbf{x}) \\ & \text{sujeito a} \quad f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & \quad \quad \quad h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p \end{aligned} \quad (25)$$

em que $\lambda \succ_{K^*} 0$, ou seja, λ é qualquer vetor positivo sob a desigualdade generalizada dual. O ponto ótimo do problema (25) também é ótimo para o problema (24). Desta forma é possível encontrar pontos ótimos de Pareto em (24) resolvendo problemas tal como (25) para diferentes λ . A dimensão de λ é a dimensão do cone dual K^* de referência para a desigualdade generalizada.

O método de escalarização resulta em pontos ótimos de Pareto à medida que λ varre o interior do cone dual K^* , mas não necessariamente resulta em todos os pontos ótimos do problema (24). Há um sutil potencial problema nos limites da fronteira ótima de Pareto, embora tecnicamente esta sutileza possa produzir pouco ou quase nenhum efeito negativo na prática [1, p. 3].

O Caso III.15 logo adiante é um exemplo de uso da técnica de escalarização para encontrar os pontos ótimos de Pareto para o problema de otimização convexa vetorial dado por

$$\begin{aligned} & \text{minimize} \quad \mathbf{X} \text{ (em relação a } \mathbb{S}_{++}^n \text{)} \\ & \text{sujeito a} \quad \mathbf{X} \succeq \mathbf{A}_1 \\ & \quad \quad \quad \mathbf{X} \succeq \mathbf{A}_2 \\ & \quad \quad \quad \mathbf{X} \succeq \mathbf{A}_3 \end{aligned} \quad (26)$$

em que $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3 \in \mathbb{S}^n$ são matrizes dadas. Como a função objetivo do problema (26) é matricial, para a escalarização do problema deve-se utilizar a operação $\text{tr}(\Lambda \mathbf{X}) = \sum_{i,j=1}^n \Lambda_{ij} \mathbf{X}_{ij}$, sendo $\Lambda \in \mathbb{S}_{++}^n$. Esta operação corresponde ao produto interno entre duas matrizes simétricas reais. Se $\mathbf{X} = \mathbf{x} \in \mathbb{R}^n$ e $\Lambda = \lambda \in \mathbb{R}_{++}^n$, logo $\text{tr}(\Lambda \mathbf{X}) = \lambda^T \mathbf{x}$ e a escalarização seria tal como aquela adotada em (25).

Caso III.15 – Exemplo de uso da técnica de escalarização.

$$\begin{aligned} & \text{minimize} \quad \text{tr}(\Lambda \mathbf{X}) \\ & \text{sujeito a} \quad \mathbf{X} \succeq \mathbf{A}_1 \\ & \quad \quad \quad \mathbf{X} \succeq \mathbf{A}_2 \\ & \quad \quad \quad \mathbf{X} \succeq \mathbf{A}_3 \end{aligned}$$

```
cvx_begin sdp
variable X(n,n) semidefinite
minimize trace(LAMBDA*X);
subject to
X >= A1;
X >= A2;
X >= A3;
cvx_end
```

onde $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3 \in \mathbb{S}^n$ e $\Lambda \in \mathbb{S}_{++}^n$.

N. Problemas de otimização multicritério

Um problema de otimização vetorial (24) é chamado de problema de otimização multicritério ou multiobjetivo se o cone de referência for $K = \mathbb{R}_+^q$. As funções componentes de

f_0 , digamos F_1, \dots, F_q , podem ser interpretadas como q diferentes funções objetivo escalares. Um problema de otimização multicritério é um problema de otimização convexa desde que f_1, \dots, f_m sejam convexas, h_1, \dots, h_p sejam afins e F_1, \dots, F_q sejam convexas.

Em um problema de otimização multicritério, um ponto ótimo \mathbf{x}^* satisfaz $F_i(\mathbf{x}^*) \leq F_i(\mathbf{y})$, $i=1, \dots, q$ para todo ponto factível \mathbf{y} . Quando existe um ponto ótimo único diz-se que as funções objetivo não competem, pois não existem relações de compromisso entre os objetivos. Quando duas ou mais funções objetivo competem entre si, haverá mais de um ponto ótimo, os já conhecidos minimalis ou pontos ótimos de Pareto. Um ponto é ótimo de Pareto se e somente se ele é factível e não existe ponto melhor do que ele em relação ao cone de referência do problema. O conjunto de pontos ótimos de Pareto para um problema multicritério é chamado de superfície de compromisso (ou *trade-off*) quando $q > 2$ e curva de *trade-off* quando $q = 2$.

O. Escalarização de problemas de otimização multicritério

Aplicando a técnica de escalarização em um problema de otimização multicritério obtém-se uma função objetivo escalar correspondente a uma soma ponderada das componentes da função objetivo multicritério original:

$$\lambda^T f_0(\mathbf{x}) = \sum_{i=1}^q \lambda_i F_i(\mathbf{x}), \quad (27)$$

com $\lambda \succ 0$. Pode-se interpretar λ_i como o peso atribuído à i -ésima componente da função objetivo original. Se for desejada uma solução com um valor pequeno (grande) para F_i , maior (menor) deve ser o valor de λ_i . A relação λ_i / λ_j pode ser interpretada como uma importância relativa entre os objetivos F_i e F_j , uma vez que um decréscimo em F_i por um fator α implica em um acréscimo de $\alpha(\lambda_i / \lambda_j)$ em F_j [1, p. 184].

O Caso III.16 em seguida é um exemplo de uso da técnica de escalarização para encontrar os pontos ótimos de Pareto para o problema de otimização bicritério dado por

$$\text{minimize} \quad f_0(\mathbf{x}) = (\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2, \|\mathbf{x}\|_2^2), \quad (K = \mathbb{R}_+^2), \quad (28)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^m$ é um vetor dado.

Caso III.16 – Exemplo de uso da técnica de escalarização em um problema bicritério

$$\text{minimize} \quad \lambda^T (\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2, \|\mathbf{x}\|_2^2)$$

```
cvx_begin quiet
variable x(n)
expression f(2)
f(1) = square_pos(norm(A*x - b));
f(2) = square_pos(norm(x));
minimize lambda'*f;
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada, $\mathbf{b} \in \mathbb{R}^m$ é um vetor dado e $\lambda \succ 0$. Se o código em questão for inserido em um laço de programa no qual se varia o vetor λ de forma que varra o interior do cone dual de referência, serão encontrados os pontos ótimos de Pareto que estabelecem a solução de compromisso entre a minimização de $\|\mathbf{Ax} - \mathbf{b}\|_2^2$ e $\|\mathbf{x}\|_2^2$.

IV. PROBLEMAS DUAIS DE OTIMIZAÇÃO CONVEXA NO CVX

Os assuntos abordados nesta seção baseiam-se no conceito de dualidade [1, pp. 215-272]. Estes assuntos abrangem:

- a função dual de Lagrange;
- a obtenção de problemas duais;
- a dualidade fraca, forte e a condição de Slater;
- a folga complementar e as condições KKT;
- a análise de sensibilidade e perturbação;
- exemplos de simples transformações em problemas primais resultando em distintos problemas duais.
- exemplos de modelagens de problemas primais e seus respectivos problemas duais no CVX.

A. Função dual de Lagrange

O chamado Lagrangiano $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ associado ao problema (1) é definido por [1, p. 215]:

$$L(\mathbf{x}, \lambda, \mathbf{v}) = f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^p v_i h_i(\mathbf{x}), \quad (29)$$

em que λ_i é o multiplicador de Lagrange associado à restrição de desigualdade $f_i(\mathbf{x}) \leq 0$ e v_i é o multiplicador de Lagrange associado à restrição de igualdade $h_i(\mathbf{x}) = 0$. Os vetores λ e \mathbf{v} são chamados de variáveis duais ou vetores multiplicadores de Lagrange associados ao problema (1). Por fim, $\text{dom}(L(\mathbf{x}, \lambda, \mathbf{v})) = \mathcal{D} \times \mathbb{R}^m \times \mathbb{R}^p$, $\mathcal{D} = \bigcap_{i=1}^m \text{dom} f_i(\mathbf{x})$.

A função dual de Lagrange $g : \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ corresponde ao mínimo valor do Lagrangiano em \mathbf{x} para $\lambda \in \mathbb{R}^m$ e $\mathbf{v} \in \mathbb{R}^p$, ou seja,

$$g(\lambda, \mathbf{v}) = \inf_{\mathbf{x} \in \mathcal{D}} L(\mathbf{x}, \lambda, \mathbf{v}). \quad (30)$$

Se o Lagrangiano (29) for ilimitado para baixo, a função dual de Lagrange (30) assumirá o valor $-\infty$. Por ser o mínimo de um conjunto de funções afins de (λ, \mathbf{v}) , a função dual de Lagrange é sempre côncava, mesmo que o problema (1) não seja um problema de otimização convexa.

Uma importante propriedade da função dual de Lagrange é que ela produz um limitante inferior p^* para o valor ótimo do problema (1), isto é, para quaisquer $\lambda \succeq 0$ e \mathbf{v} tem-se que

$$g(\lambda, \mathbf{v}) \leq p^*. \quad (31)$$

B. Obtenção de problemas duais

Como a função dual de Lagrange (30) estabelece um limitante inferior para o valor ótimo p^* do problema primal para cada (λ, \mathbf{v}) e $\lambda \succeq 0$, é natural surgir a seguinte pergunta: qual é o melhor limitante inferior para o valor ótimo p^* que pode ser obtido a partir da função dual de Lagrange? A resposta para esta pergunta decorre da solução do seguinte problema de otimização

$$\begin{aligned} & \text{maximize} && g(\lambda, \mathbf{v}) \\ & \text{sujeito a} && \lambda \succeq 0. \end{aligned} \quad (32)$$

Este problema é conhecido como problema dual de Lagrange e está relacionado com o problema (1), que neste contexto é também chamado de problema primal. Um par (λ, \mathbf{v}) é factível para o problema dual se $\lambda \succeq 0$ e $g(\lambda, \mathbf{v}) \geq -\infty$. Quando o par (λ, \mathbf{v}) é ótimo para o problema dual (32) ele é chamado de ótimo dual e é identificado por $(\lambda^*, \mathbf{v}^*)$. O valor ótimo do problema dual (32) é $d^* = g(\lambda^*, \mathbf{v}^*)$, o melhor limitante inferior para p^* .

O problema de otimização dual de Lagrange (32) é sempre um problemas de otimização convexa, uma vez que o objetivo é maximizar a função $g(\lambda, \mathbf{v})$, que é côncava, e a função de restrição é afim e, portanto, convexa. Isto ocorre mesmo que o problema primal não seja um problema de otimização convexa.

Como mencionado no tópico *variáveis duais* do item B da Seção II, todo problema de otimização convexa quando é resolvido pelo CVX tem automaticamente o seu problema dual resolvido, possibilitando a obtenção dos valores ótimos das variáveis duais relacionadas às restrições do problema primal.

Os Casos IV.1 e IV.2 a seguir são modelagens no CVX dos problemas duais referentes aos problemas LP na forma padrão e na forma de desigualdade.

Caso IV.1 – Problema dual de Lagrange de um LP na forma padrão.

$$\begin{aligned} & \text{maximize} && -\mathbf{b}^T \mathbf{v} \\ & \text{sujeito a} && \mathbf{A}^T \mathbf{v} + \mathbf{c} \succeq 0 \end{aligned}$$

```
cvx_begin
variable v(n)
maximize -b'*v
subject to
A'*v + c >= 0
cvx_end
```

onde $\mathbf{v} \in \mathbb{R}^m$ (representada por \mathbf{v} no código) é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada, $\mathbf{b} \in \mathbb{R}^m$ e $\mathbf{c} \in \mathbb{R}^n$ são vetores dados. Este é o problema dual de Lagrange do problema LP na forma padrão do Caso III.7.

Caso IV.2 – Problema dual de Lagrange de um LP na forma de desigualdade.

$$\begin{aligned} & \text{maximize} && -\mathbf{b}^T \boldsymbol{\lambda} \\ & \text{sujeito a} && \mathbf{A}^T \boldsymbol{\lambda} + \mathbf{c} = 0 \\ & && \boldsymbol{\lambda} \succeq 0 \end{aligned}$$

```
cvx_begin
  variable l(n)
  maximize -b'*l
  subject to
    A'*l + c == 0
    l >= 0
cvx_end
```

onde $\boldsymbol{\lambda} \in \mathbb{R}^m$ (representada por \mathbf{l} no código) é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada, $\mathbf{c} \in \mathbb{R}^n$ são vetores dados e $\boldsymbol{\lambda} \succ 0$. Este é o problema dual de Lagrange do problema LP na forma de desigualdade do Caso III.8.

C. Dualidade fraca, forte e condição de Slater

Dualidade fraca

O valor ótimo do problema dual é sempre menor que ou igual ao valor ótimo do problema primal, ou seja,

$$d^* \leq p^* \quad (33)$$

é uma desigualdade válida mesmo que o problema primal não seja um problema de otimização convexa. Esta importante desigualdade é chamada de dualidade fraca, a qual se mantém mesmo para valores infinitos de d^* e p^* : se um problema primal é ilimitado para baixo, ou seja $p^* = -\infty$, o problema dual relacionado é infactível e $d^* = -\infty$. Reciprocamente, se o problema dual é ilimitado para cima, $d^* = \infty$, e o problema primal relacionado é infactível, ou seja, $p^* = \infty$.

A diferença $p^* - d^*$ é chamada de intervalo de dualidade ótimo (*optimal duality gap*), uma vez que é a diferença entre o valor ótimo do problema primal e o melhor limitante inferior de p^* , correspondente ao valor ótimo do problema dual, d^* . O intervalo de dualidade ótimo é sempre não negativo.

Nos casos em que o problema primal é de difícil solução, tem-se como alternativa encontrar o limitante inferior d^* para seu valor ótimo p^* . O Caso IV.3 logo adiante modela um problema dual cuja solução é o melhor limitante inferior para o problema primal conhecido como problema de partição bidirecional (*two-way partitioning*) [1, pp 219-221], que não é um problema de otimização convexa e é dado por:

$$\begin{aligned} & \text{minimize} && \mathbf{x}^T \mathbf{W} \mathbf{x} \\ & \text{sujeito a} && x_i^2 = 1, \quad i = 1, \dots, n, \end{aligned} \quad (34)$$

em que $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema e $\mathbf{W} \in \mathbb{S}^n$ é uma matriz dada.

Caso IV.3 – Problema dual de um problema primal de otimização não convexa.

$$\begin{aligned} & \text{maximize} && -\mathbf{1}^T \mathbf{v} \\ & \text{sujeito a} && \mathbf{W} + \text{diag}(\mathbf{v}) \succeq 0 \end{aligned}$$

```
cvx_begin sdp
  variable v(n)
  maximize -ones(n,1)*v
  subject to
    W + diag(v) >= 0
cvx_end
```

onde $\mathbf{v} \in \mathbb{R}^n$ é a variável do problema e $\mathbf{W} \in \mathbb{S}^n$ é uma matriz dada.

A solução deste problema de otimização convexa é o melhor limitante inferior para o problema de otimização não convexa (34).

Dualidade forte e condição de Slater

Se o intervalo de dualidade ótimo é zero ($d^* - p^* = 0$), tem-se que $d^* = p^*$ e então se diz que existe dualidade forte entre o problema primal e o problema dual correspondente.

Quando o problema primal é um problema de otimização convexa, geralmente tem-se dualidade forte, porém apenas a convexidade não é suficiente para garantir esta situação. Existem condições chamadas de qualificações de restrições (*constraint qualifications*) que garantem a dualidade forte. Uma simples qualificação de restrição é chamada de *condição de Slater*, a qual estabelece que, para o problema de otimização convexa (2), existe um ponto $\mathbf{x} \in \text{relint}(\mathcal{D})$ tal que

$$f_i(\mathbf{x}) < 0, \quad i = 1, \dots, m, \quad h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p. \quad (35)$$

Este ponto é algumas vezes chamado de estritamente factível, uma vez que ele satisfaz as restrições de desigualdade estritamente. O teorema de Slater diz que se a condição de Slater se manifesta e se trata de um problema de otimização convexa, $d^* = p^*$.

A condição de Slater pode ser refinada quando existem restrições de desigualdade como funções afins. Segundo a condição de Slater refinada, as restrições de desigualdade com funções afins podem ser satisfeitas também com igualdades (não estritamente) [1, p. 226].

A condição de Slater é reduzida a um problema de factibilidade quando todas as restrições são afins e o domínio de f_0 é aberto. Em outras palavras, se as restrições são afins, basta que o problema seja factível que a dualidade forte se aplicará.

Além de estabelecer dualidade forte para problemas de otimização convexa, a condição de Slater também garante que existe um ponto dual ótimo alcançável quando $d > -\infty$, ou seja, existe um ponto dual factível $(\boldsymbol{\lambda}^*, \mathbf{v}^*)$, sendo $g(\boldsymbol{\lambda}^*, \mathbf{v}^*) = d^* = p^*$.

As qualificações de restrição estabelecem condições (além da análise de convexidade) sob as quais a dualidade forte se aplica. São, *per se*, um extenso ramo no estudo de otimização. A qualificação de Slater é apenas uma das possíveis formas de qualificação, embora muito útil no contexto de problemas de otimização convexa.

Os casos apresentados na sequência são modelagens de problemas primais e duais que possuem dualidade forte quando a condição de Slater para cada caso é satisfeita.

Caso IV.4 – Solução de sistemas de equações lineares pelo método dos mínimos quadrados.

Problema primal

$$\begin{aligned} & \text{minimize} && \mathbf{x}^T \mathbf{x} \\ & \text{sujeito a} && \mathbf{Ax} = \mathbf{b} \end{aligned}$$

```
n = size(A,2)
cvx_begin
    variable x(n)
    minimize x'*x
    subject to
        A*x == b
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{m \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^m$ é um vetor dado.

Problema dual

$$\text{maximize} \quad -\frac{1}{4} \mathbf{v}^T \mathbf{A} \mathbf{A}^T \mathbf{v} - \mathbf{b}^T \mathbf{v}$$

```
m = size(A,1)
cvx_begin
    variable v(m)
    maximize (-1/4)*(v'*A)*(A'*v)-b'*v
cvx_end
```

onde $\mathbf{v} \in \mathbb{R}^m$ é a variável do problema.

A condição de Slater para que exista a dualidade forte nesse caso é que o problema primal simplesmente seja factível.

Caso IV.5 – Problema dual de Lagrange de um QCQP

Problema primal

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r \\ & \text{sujeito a} && \frac{1}{2} \mathbf{x}^T \mathbf{W} \mathbf{x} \leq 0 \end{aligned}$$

```
n = length(a1);
cvx_begin
    variable x(n)
    minimize (1/2)*x'*P*x + q'*x + r
    subject to
        (1/2)*x'*W*x <= 0
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{P}, \mathbf{S} \in \mathbb{S}^n$ são matrizes dadas e $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{q} \in \mathbb{R}^n$ são vetores dados.

Problema dual

$$\text{maximize} \quad -\frac{1}{2} \mathbf{q}^T (\mathbf{P} + \lambda \mathbf{W})^{-1} \mathbf{q} + r$$

```
n = length(a1);
m = 2;
cvx_begin
    variable lam
    maximize -(1/2)*matrix_frac(q,(P+lam*W))+r
    subject to
        lam >= 0
```

cvx_end

onde $\lambda \in \mathbb{R}$ é a variável do problema. No código, a variável λ está identificada como **lam**.

Para que exista a dualidade forte neste caso, segundo a condição de Slater, a desigualdade do problema primal deve ser satisfeita estritamente, ou seja, basta que exista um \mathbf{x} qualquer tal que $\frac{1}{2} \mathbf{x}^T \mathbf{W} \mathbf{x} < 0$.

Caso IV.6 – Maximização da entropia (ou minimização da entropia negativa).

Problema primal

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^n x_i \log x_i \\ & \text{sujeito a} && \mathbf{Ax} \preceq \mathbf{b} \\ & && \mathbf{1}^T \mathbf{x} = 1 \end{aligned}$$

```
n = size(A,2);
cvx_begin
    variable x(n)
    expression aux(n)
    for i = 1:n
        aux(i) = -entr(x(i));
    end
    minimize sum(aux)
    subject to
        A*x <= b;
        ones(n)'*x == 1
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}_+^n$ é a variável do problema, $\mathbf{A} \in \mathbb{R}^{n \times n}$ é uma matriz dada e $\mathbf{b} \in \mathbb{R}^n$ é um vetor dado.

Problema dual

$$\begin{aligned} & \text{maximize} && -\mathbf{b}^T \boldsymbol{\lambda} - v - e^{-v-1} \sum_{i=1}^n e^{-\mathbf{a}_i^T \boldsymbol{\lambda}} \\ & \text{sujeito a} && \boldsymbol{\lambda} \succeq 0 \end{aligned}$$

```
cvx_begin
    variables lam(m) v
    expression aux(m)
    for i=1:m
        aux(i) = exp(-(A(:,i))*lam-v-1);
    end
    maximize -b'*lam-v-sum(aux)
    subject to
        lam >= 0
cvx_end
```

onde $\boldsymbol{\lambda} \in \mathbb{R}^n$ e $v \in \mathbb{R}$ são variáveis do problema.

A condição de Slater refinada para o problema primal deste caso diz que a dualidade forte se aplicará se existir um $\mathbf{x} \succ 0$ com $\mathbf{Ax} \preceq \mathbf{b}$ e $\mathbf{1}^T \mathbf{x} = 1$.

Caso IV.7 – Elipsoide de cobertura com mínimo volume.

Problema primal

$$\begin{aligned} & \text{minimize} && \log \det \mathbf{X}^{-1} \\ & \text{sujeito a} && \mathbf{a}_i^T \mathbf{X} \mathbf{a}_i \leq 1, \quad i = 1, \dots, m. \end{aligned}$$

```
n = length(a1);
m = 2;
cvx_begin
```

```

variable X(n,n)
minimize det_inv(X)
subject to
  (a1)'*X*a1 <= 1
  (a2)'*X*a2 <= 1
cvx_end
log(cvx_optval)

```

onde $\mathbf{X} \in \mathbb{S}_{++}^n$ é a variável do problema, $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{R}^n$ são vetores dados.

Problema dual

```

maximize log det( sum_{i=1}^m lambda_i a_i a_i^T ) - 1^T lambda + n
sujeito a lambda >= 0

```

```

cvx_begin
variable lam(n)
expressions aux1(n,n) aux2(n,n)
aux1 = lam(1)*(a1)*(a1)';
aux2 = lam(2)*(a2)*(a2)';
aux = log_det(aux1 + aux2)
maximize aux - ones(1,2)*lam + n
subject to
  lam >= 0
cvx_end

```

onde $\lambda \in \mathbb{R}_+^n$ é a variável do problema.

A condição de Slater refinada para o problema primal deste caso diz que a dualidade forte ocorrerá se existir um $\mathbf{X} \succ 0$ com $\mathbf{a}_i^T \mathbf{X} \mathbf{a}_i \leq 1, i = 1, \dots, m$.

D. Folga complementar e condições KKT

Folga complementar

Quando a dualidade forte entre um problema primal e seu problema dual se aplica, pode-se concluir a partir de [1, pp. 242-243] que

$$\sum_{i=1}^m \lambda_i^* f_i(\mathbf{x}^*) = 0. \quad (36)$$

Como $\lambda_i^* \geq 0$ e $f_i(\mathbf{x}^*) \leq 0, i = 1, \dots, m$, conclui-se que

$$\lambda_i^* f_i(\mathbf{x}^*) = 0, i = 1, \dots, m. \quad (37)$$

Esta condição é conhecida como folga complementar (*complementary slackness*) e é mantida para qualquer ponto ótimo primal \mathbf{x}^* e ponto dual $(\lambda^*, \mathbf{v}^*)$ quando a dualidade forte se aplica. A condição de folga complementar pode ser assim expressa:

$$\begin{aligned} \lambda_i^* > 0 &\Rightarrow f_i(\mathbf{x}^*) = 0 \\ \lambda_i^* = 0 &\Rightarrow f_i(\mathbf{x}^*) < 0. \end{aligned} \quad (38)$$

Condições KKT

Considere um problema de otimização (não necessariamente convexa) com sua função objetivo e suas funções de restrições diferenciáveis. Considere também que o par \mathbf{x}^* e $(\lambda^*, \mathbf{v}^*)$ seja qualquer par de pontos ótimos primal e

dual, respectivamente, e que a dualidade forte se aplica. Uma vez que \mathbf{x}^* é um minimizador de $L(\mathbf{x}, \lambda^*, \mathbf{v}^*)$ [1, pp. 242-243], segue que o gradiente do Lagrangiano deve ser nulo:

$$\nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(\mathbf{x}^*) + \sum_{i=1}^p v_i^* \nabla h_i(\mathbf{x}^*) = 0. \quad (39)$$

Agrupando a essa condição a folga complementar e as demais condições impostas ao problema, tem-se que:

$$\begin{aligned} f_i(\mathbf{x}^*) &\leq 0, i = 1, \dots, m \\ h_i(\mathbf{x}^*) &= 0, i = 1, \dots, p \\ \lambda_i^* &\geq 0, i = 1, \dots, m \\ \lambda_i^* f_i(\mathbf{x}^*) &= 0, i = 1, \dots, m \end{aligned} \quad (40)$$

$$\nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(\mathbf{x}^*) + \sum_{i=1}^p v_i^* \nabla h_i(\mathbf{x}^*) = 0$$

Estas são as condições de Karush-Kuhn-Tucker ou, simplesmente, condições KKT. Qualquer par de pontos ótimos primal e dual com intervalo de dualidade zero, para qualquer problema de otimização com a função objetivo e as funções de restrições diferenciáveis, deve satisfazer as condições KKT.

Quando se trata de um problema de otimização convexa, as condições KKT são suficientes para que os pontos sejam ótimos primal e dual e com intervalo de dualidade zero.

Se um problema de otimização convexa com a função objetivo e as funções de restrições diferenciáveis satisfaz a condição de Slater, logo as condições KKT são necessárias e suficientes para a otimalidade, ou seja, os pontos primal e dual são alcançáveis, possuem intervalo de dualidade igual à zero e atendem às condições KKT.

As condições KKT são muito importantes para a solução de problemas de otimização convexa. Em alguns casos especiais é possível resolver as correspondentes equações analiticamente e, portanto o problema de otimização relacionado. De forma mais geral, muitos algoritmos para otimização convexa são baseados em, ou podem ser interpretados como algoritmos de solução das condições KKT.

O Caso IV.8 modela um problema de otimização convexa no CVX, referente a um sistema de equações obtido através das condições KKT. A solução deste sistema de equações, que é obtida através de comandos do Matlab, corresponde à solução do problema de otimização convexa dado.

Caso IV.8 – Solução de um problema de otimização convexa através das condições KKT.

Problema de otimização convexa

```

minimize 1/2 x^T P x + q^T x
sujeito a Ax = b

```

```

cvx_begin
variable x(n)
dual variable v
minimize 0.5*x'*P*x + q'*x
subject to
  v: b - A*x == 0
cvx_end

```

onde $\mathbf{x} \in \mathbb{R}^n$ é a variável do problema, $\mathbf{P} \in \mathbb{S}_+^n$ e $\mathbf{A} \in \mathbb{R}^{n \times n}$ são matrizes dadas e $\mathbf{b}, \mathbf{q} \in \mathbb{R}^n$ são vetores dados.

No código, a variável de otimização \mathbf{x} e a variável dual \mathbf{v} devem ter os mesmos valores que as variáveis a serem obtidas pela solução do sistema de equações a seguir.

Sistema de equações obtido através das condições KKT

$$\begin{bmatrix} \mathbf{x}^* \\ \mathbf{v}^* \end{bmatrix} = \begin{bmatrix} \mathbf{P} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{q} \\ \mathbf{b} \end{bmatrix}$$

```
sol_KKT = inv([P A'; A zeros(3)])*[-q; b];
x_KKT = sol_KKT(1:3)
v_KKT = sol_KKT(4:6)
```

onde \mathbf{x}^* é a variável ótima primal e \mathbf{v}^* é a variável ótima dual. Os valores de \mathbf{x}_KKT e \mathbf{v}_KKT encontrados na solução do sistema no código possuem os mesmos valores de \mathbf{x} e \mathbf{v} encontrados anteriormente na solução do problema de otimização por meio do CVX.

E. Análises de sensibilidade e perturbação

Problema com perturbação

Considere a seguinte versão do problema original (1), a qual estamos chamando de problema com perturbação:

$$\begin{aligned} &\text{minimize} && f_0(\mathbf{x}) \\ &\text{sujeito a} && f_i(\mathbf{x}) \leq u_i, \quad i = 1, \dots, m \\ &&& h_i(\mathbf{x}) = v_i, \quad i = 1, \dots, p, \end{aligned} \quad (41)$$

com variável $\mathbf{x} \in \mathbb{R}^n$.

Este problema coincide com o problema original quando $u_i = 0$ e $v_i = 0$. Quando u_i é positivo significa que a i -ésima restrição de desigualdade foi afrouxada e quando u_i é negativo diz-se que ela foi apertada. Quando v_i for diferente de zero, entende-se tal perturbação como uma alteração do valor alvo (*target* ou *set-point*) do problema, este determinado pelas restrições de igualdade.

O valor ótimo do problema com perturbação (41) em função das perturbações u_i e v_i é dado pela função $p^* : \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$:

$$\begin{aligned} p^*(\mathbf{u}, \mathbf{v}) = \inf \{ & f_0(\mathbf{x}) \mid \exists \mathbf{x} \in \mathcal{D}, f_i(\mathbf{x}) \leq u_i, i = 1, \dots, m, \\ & h_i(\mathbf{x}) = v_i, i = 1, \dots, p \} \end{aligned} \quad (42)$$

Quando $p^*(\mathbf{u}, \mathbf{v}) = \infty$ tem-se um problema infactível para as perturbações dadas. Se as perturbações são nulas, o valor ótimo do problema com perturbação é igual ao valor ótimo do problema original, ou seja, $p^*(0, 0) = p^*$. Quando o problema original é um problema de otimização convexa, a função $p^*(\mathbf{u}, \mathbf{v})$ também é convexa.

Desigualdade global

Considere que o problema original (1) é um problema de otimização convexa e que a condição de Slater se aplica. Considere também que o ponto $(\lambda^*, \mathbf{v}^*)$ seja ótimo para o

problema dual (32) do problema sem perturbação. Desta forma é estabelecida a seguinte desigualdade [1, p. 250]:

$$p^*(\mathbf{u}, \mathbf{v}) \geq p^*(0, 0) - \lambda^{*\top} \mathbf{u} - \mathbf{v}^{*\top} \mathbf{v} \quad (43)$$

A partir dessa desigualdade é possível concluir que:

- se λ_i^* é grande e $u_i < 0$, é esperado que o valor ótimo $p^*(\mathbf{u}, \mathbf{v})$ cresça muito;
- se ν_i^* é grande e positivo e $v_i < 0$ ou se ν_i^* é grande e negativo e $v_i > 0$, é esperado que $p^*(\mathbf{u}, \mathbf{v})$ cresça muito;
- se λ_i^* é pequeno e $u_i > 0$, o valor ótimo $p^*(\mathbf{u}, \mathbf{v})$ não decrescerá muito;
- se ν_i^* é pequeno e positivo e $v_i > 0$ ou se ν_i^* é pequeno e negativo e $v_i < 0$, $p^*(\mathbf{u}, \mathbf{v})$ não decrescerá muito.

As conclusões anteriores e a desigualdade (43) estabelecem um limitante inferior sobre o valor ótimo do problema com perturbação, mas não um limitante superior. Em outras palavras, os resultados não são simetricamente modificados a partir do afrouxamento ou aperto de uma restrição.

Análise de sensibilidade local

Considere que $p^*(\mathbf{u}, \mathbf{v})$ é diferenciável em $\mathbf{u} = 0$ e $\mathbf{v} = 0$, e também que a dualidade forte se aplica. Desta forma as variáveis ótimas duais λ^* e \mathbf{v}^* são relacionadas ao gradiente de p^* em $\mathbf{u} = 0$ e $\mathbf{v} = 0$, ou seja,

$$\lambda_i^* = -\frac{\partial p^*(0, 0)}{\partial u_i}, \quad \nu_i^* = -\frac{\partial p^*(0, 0)}{\partial v_i}. \quad (44)$$

Pode-se concluir que os multiplicadores de Lagrange podem ser interpretados como a taxa de variação em p^* devido a variações nas restrições. Quando a i -ésima restrição de desigualdade é apertada um pouco (fazendo u_i pequeno e negativo) tem-se um acréscimo em p^* de aproximadamente $-\lambda_i^* u_i$. Quando a i -ésima restrição de desigualdade é afrouxada um pouco (fazendo u_i pequeno e positivo) tem-se um decréscimo em p^* de, aproximadamente, $\lambda_i^* u_i$.

O resultado da sensibilidade local dá uma medida quantitativa de quanto ativa é uma restrição no ponto ótimo \mathbf{x}^* . Se $f_i(\mathbf{x}^*) < 0$, tem-se uma restrição inativa e desta forma é possível apertá-la ou afrouxá-la um pouco sem provocar alterações em p^* . Isto ocorre porque, de acordo com a folga complementar, o multiplicador de Lagrange correspondente a esta restrição deve ser zero. Se $f_i(\mathbf{x}^*) = 0$ tem-se uma restrição ativa e, portanto, o multiplicador de Lagrange relacionado deverá dizer o quão ativa é esta restrição. Caso λ_i^* seja pequeno, um pequeno afrouxamento ou aperto na restrição correspondente não deverá alterar o valor ótimo do problema. Caso λ_i^* for grande, um pequeno afrouxamento ou aperto na restrição correspondente deverá causar uma grande alteração no valor ótimo do problema.

O Caso IV.9 apresenta modelagens de um problema de otimização convexa com dualidade forte nas versões não perturbada e perturbada e também um esboço da função $p^*(u)$ e de seu limitante inferior.

Caso IV.9 – Problema de otimização convexa com perturbação.

Problema sem perturbação

$$\begin{aligned} & \text{minimize} && \frac{1}{(x-2)} \\ & \text{sujeito a} && (x-3) \leq 0 \end{aligned}$$

```
cvx_begin
variable x
dual variable lamb
minimize inv_pos(x-2)
subject to
lamb: (x-3) <= 0
cvx_end
val_otimo = cvx_optval;
```

onde $\{x \in \mathbb{R} \mid x > 2\}$ é a variável do problema.

No código, **lamb** armazena o valor ótimo da variável dual do problema e **val_otimo** armazena o valor ótimo do problema. Estas variáveis serão utilizadas para plotar o limitante inferior de $p^*(u)$.

Problema com perturbação

$$\begin{aligned} & \text{minimize} && \frac{1}{(x-2)} \\ & \text{sujeito a} && (x-3) \leq u \end{aligned}$$

```
u = -1:0.1:2;
for i=1:length(u)
cvx_begin
variable x
minimize inv_pos(x-2)
subject to
(x-3) <= u(i)
cvx_end
p(i) = cvx_optval;
end
```

onde $\{x \in \mathbb{R} \mid x > 2\}$ é a variável do problema e $u \in \mathbb{R}$ é a perturbação.

Como pode ser visto no código, o problema de otimização está localizado dentro da estrutura de repetição **for** condicionada se a repetir para valores de u entre -1 e 2 em passos de $0,1$. Cada elemento de u corresponde a uma perturbação diferente para o problema de otimização, uma vez que $u(i)$ está localizado no lado direito da restrição do problema. A cada novo ciclo de repetição tem-se um novo problema com perturbação a ser resolvido. A cada problema resolvido, um novo valor ótimo é encontrado e armazenado na i -ésima posição do vetor **p**. Quando a estrutura de repetição é encerrada, todos os valores ótimos correspondentes aos problemas com perturbação ficam armazenados no vetor **p** de tal forma que o i -ésimo elemento de **p** corresponda ao i -ésimo elemento de **u**.

De posse dos vetores **p** e **u** e dos valores ótimos da variável dual **lamb** e do problema sem perturbação **val_otimo** é possível plotar a função p^* e seu limitante inferior $p^*(0) - \lambda^* u$, como ilustrado na Fig. 1.

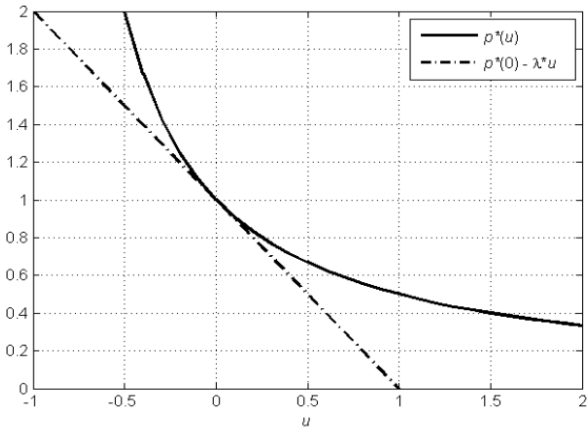


Figura 1. Valor ótimo do problema com perturbação (p^*) em função da perturbação (u) e limitante inferior do valor ótimo ($p^*(0) - \lambda^* u$)

Na Fig. 1, $p^*(u)$ é o valor ótimo do problema com perturbação em função da perturbação u e $p^*(0) - \lambda^* u$ é o limitante inferior de $p^*(u)$. Para $u = 0$ tem-se o valor ótimo do problema sem perturbação. Para $u > 0$ tem-se um problema com perturbação com a restrição afrouxada e para $u < 0$ tem-se um problema com perturbação com a restrição apertada.

F. Problemas duais de problemas de otimização com desigualdades generalizadas

Considere o problema de otimização (não necessariamente convexa) com desigualdade generalizada:

$$\begin{aligned} & \text{minimize} && f_0(\mathbf{x}) \\ & \text{sujeito a} && f_i(\mathbf{x}) \preceq_{K_i} 0, \quad i = 1, \dots, m \\ & && h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p, \end{aligned} \quad (45)$$

em que $K_i \subseteq \mathbb{R}^{k_i}$ são cones próprios.

A cada $f_i(\mathbf{x}) \preceq_{K_i} 0$ associa-se um vetor multiplicador de Lagrange $\lambda_i \in \mathbb{R}^{k_i}$ e assim define-se o Lagrangiano:

$$L(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{v}) = f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i^T f_i(\mathbf{x}) + \sum_{i=1}^p v_i h_i(\mathbf{x}) \quad (46)$$

em que $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)$ e $\mathbf{v} = (v_1, \dots, v_p)$.

A função dual de Lagrange é definida exatamente como no caso escalar em (30). Como em um problema com desigualdades escalares, a função dual de Lagrange estabelece limitantes inferiores sobre o valor ótimo p^* do problema primal (45). Para um problema com desigualdades escalares exige-se que $\lambda_i \geq 0$. Para o caso de problemas com desigualdades generalizadas a exigência é que

$$\lambda_i \succeq_{K_i^*} 0, \quad i = 1, \dots, m, \quad (47)$$

em que K_i^* é o cone dual [1, pp. 51-58] de K_i .

O problema de otimização dual correspondente a um problema de otimização com desigualdades generalizadas é dado por

$$\begin{aligned} & \text{maximize} && g(\boldsymbol{\lambda}, \mathbf{v}) \\ & \text{sujeito a} && \lambda_i \succeq_{K_i^*} 0, \quad i = 1, \dots, m. \end{aligned} \quad (48)$$

Uma vez que $g(\boldsymbol{\lambda}, \mathbf{v})$ é sempre uma função côncava, tem-se que o problema dual é sempre um problema de otimização convexa. A dualidade fraca, ou seja, $d^* \leq p^*$, onde d^* é o valor ótimo do problema dual, se aplica independente da convexidade do problema primal (45).

A dualidade forte, $d^* = p^*$, também se aplica para este caso quando o problema original é um problema de otimização convexa e satisfaz uma qualificação de restrição apropriada. Por exemplo, uma versão generalizada da condição de Slater para o problema (17) é que exista um $\mathbf{x} \in \text{relint}(\mathcal{D})$ com $\mathbf{Ax} = \mathbf{b}$ e $f_i(\mathbf{x}) \prec_{K_i^*} 0$, $i = 1, \dots, m$. Esta condição implica que a dualidade forte se aplica e que o dual ótimo é alcançável.

O Caso IV.10 modela um problema de otimização semidefinida e seu correspondente problema dual.

Caso IV.10 – Problema de otimização semidefinida.

Problema primal

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{sujeito a} && x_1 \mathbf{F}_1 + x_2 \mathbf{F}_2 + \mathbf{G} \preceq 0 \end{aligned}$$

```
cvx_begin sdp
variable x(2)
minimize c'*x
subject to
x(1)*F_1 + x(2)*F_2 + G <= 0
cvx_end
```

onde $\mathbf{x} \in \mathbb{R}^2$ é a variável de otimização, $\mathbf{F}_1, \mathbf{F}_2, \mathbf{G} \in \mathbb{S}^k$ são matrizes dadas e $\mathbf{c} \in \mathbb{R}^2$ é um vetor dado.

Problema dual

$$\begin{aligned} & \text{maximize} && \text{tr}(\mathbf{GZ}) \\ & \text{sujeito a} && \text{tr}(\mathbf{F}_1 \mathbf{Z}) + c_1 = 0 \\ & && \text{tr}(\mathbf{F}_2 \mathbf{Z}) + c_2 = 0 \\ & && \mathbf{Z} \succeq 0 \end{aligned}$$

```
cvx_begin sdp
variable Z(2,2) symmetric
maximize trace(G*Z)
subject to
trace(F_1*Z) + c(1) == 0
trace(F_2*Z) + c(2) == 0
Z >= 0
cvx_end
```

onde $\mathbf{Z} \in \mathbb{S}^2$ é a variável do problema, $c_1, c_2 \in \mathbb{R}$ são os elementos de $\mathbf{c} \in \mathbb{R}^2$ e \mathbb{S}_+^2 é o cone dual.

A dualidade forte para este caso se aplica se o problema primal é estritamente factível, ou seja, se existe um \mathbf{x} tal que $x_1 \mathbf{F}_1 + x_2 \mathbf{F}_2 + \mathbf{G} \prec 0$.

V. APLICAÇÕES

Nesta seção são apresentadas algumas aplicações de otimização convexa em situações concretas. Em cada aplicação a modelagem do problema, sua implementação no CVX e os resultados obtidos são fornecidos. Dentre as aplicações escolhidas algumas podem ser modeladas por problemas de otimização que apresentam solução analítica. Para essas aplicações pode-se comparar o esforço demandado na solução analítica com aquele despendido pelo CVX.

Aplicação V.1 – Constelações de energia mínima em sistemas de transmissão digital.

Seja um sistema de transmissão digital no qual o conjunto dos M símbolos da sua constelação é representado por $\{\mathbf{s}_i\}$, $i = 1, 2, \dots, M$. Seja p_i a probabilidade do envio do símbolo i de energia E_i . A energia média por símbolo da constelação é dada por $E = \sum_{i=1}^M p_i E_i$. Pode-se alterar o valor da energia média trasladando a constelação dentro do espaço Euclidiano. Isso é feito subtraindo de cada vetor-sinal um vetor \mathbf{d} . Assim, a energia de cada símbolo trasladado passa ser a norma euclidiana de $(\mathbf{s}_i - \mathbf{d})$ ao quadrado, o que leva a uma energia média em função de \mathbf{d} dada por

$$E(\mathbf{d}) = \sum_{i=1}^M p_i \|\mathbf{s}_i - \mathbf{d}\|_2^2. \quad (49)$$

Para otimizar o desempenho do sistema pode-se desejar minimizar a potência média de transmissão, que é diretamente proporcional à energia média por símbolo. Tem-se então um problema de otimização, tal como o descrito no Caso II.1 desse tutorial, no qual a variável é o vetor \mathbf{d} .

Considere como exemplo uma constelação bidimensional de 4 símbolos ($M = 4$) representados pelos vetores

$$\mathbf{s}_1 = \begin{bmatrix} 3 \\ 3 \end{bmatrix}, \quad \mathbf{s}_2 = \begin{bmatrix} -1 \\ 3 \end{bmatrix}, \quad \mathbf{s}_3 = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \quad \text{e} \quad \mathbf{s}_4 = \begin{bmatrix} 3 \\ -1 \end{bmatrix},$$

e que as probabilidades *a priori* desses símbolos são os elementos do vetor $\mathbf{p} = [0.2 \ 0.3 \ 0.1 \ 0.4]^T$.

A formulação do problema e a correspondente modelagem no CVX são:

$$\text{minimize} \quad \sum_{i=1}^M p_i \|\mathbf{s}_i - \mathbf{d}\|_2^2,$$

```
cvx_begin
variable d(2)
expression x(4)
for i = 1 : 4
x(i) = p(i) * square_pos(norm(s(i,:) - d));
end
minimize sum(x)
cvx_end
Eoti = cvx_optval
```

onde $\mathbf{s} = (\mathbf{s}_1^T, \dots, \mathbf{s}_4^T)$, $\mathbf{s}_i \in \mathbb{R}^2$, $i = 1, \dots, 4$.

As soluções obtidas pelo CVX para os valores de \mathbf{d}^* e $E[\mathbf{d}^*]$ são, respectivamente, $\mathbf{d} = \begin{bmatrix} 1,4 \\ 1,0 \end{bmatrix}$, $\mathbf{Eoti} = +7.84$

Assim, a constelação ótima (de energia mínima) será:

$$\mathbf{s}_1^* = \begin{bmatrix} 1,6 \\ 2,0 \end{bmatrix}, \mathbf{s}_2^* = \begin{bmatrix} -2,4 \\ 2,0 \end{bmatrix}, \mathbf{s}_3^* = \begin{bmatrix} -2,4 \\ -2,0 \end{bmatrix} \text{ e } \mathbf{s}_4^* = \begin{bmatrix} 1,6 \\ -2,0 \end{bmatrix}.$$

A solução analítica desse problema é dada por:

$$\begin{aligned} \mathbf{d}^* &= \sum_{i=1}^M p_i \mathbf{s}_i = \\ &= 0,2 \begin{bmatrix} 3 \\ 3 \end{bmatrix} + 0,3 \begin{bmatrix} -1 \\ 3 \end{bmatrix} + 0,1 \begin{bmatrix} -1 \\ -1 \end{bmatrix} + 0,4 \begin{bmatrix} 3 \\ -1 \end{bmatrix} = \begin{bmatrix} 1,4 \\ 1,0 \end{bmatrix}, \end{aligned}$$

e o correspondente valor de energia mínima será:

$$\begin{aligned} E(\mathbf{d}^*) &= \sum_{i=1}^M p_i \|\mathbf{s}_i - \mathbf{d}^*\|_2^2 = \sum_{i=1}^M p_i [(\mathbf{s}_i - \mathbf{d}^*)^T (\mathbf{s}_i - \mathbf{d}^*)] \\ &= E(\mathbf{0}) - \|\mathbf{d}^*\|_2^2 = \sum_{i=1}^4 p_i (\mathbf{s}_i^T \mathbf{s}_i) - \|\mathbf{d}^*\|_2^2 \\ &= 10,8 - 2,96 = 7,84 \text{ joules.} \end{aligned}$$

Aplicação V.2 – Maximização da capacidade de canais com ruído colorido Gaussiano.

Esta aplicação trata da maximização da capacidade de um canal com ruído colorido Gaussiano conforme discutido em [13, pp. 719-724]. Na formulação do problema é considerado um transmissor de um sistema de comunicação que dispõe de uma potência fixa P_T que deve ser adequadamente distribuída em U subcanais, sendo que a potência média alocada para cada canal é P_u , $u = 1, \dots, U$. Cada um desses canais está sujeito a um nível aleatório de ruído colorido Gaussiano com potência média σ_u^2 , $u = 1, \dots, U$.

De acordo com o teorema de Hartley-Shannon, a capacidade do u -ésimo subcanal (C_u) é dada por

$$C_u = B \log_2 \left(1 + \frac{P_u}{\sigma_u^2} \right) \text{ bit/s,} \quad (50)$$

onde B é a banda do subcanal em hertz. A capacidade do conjunto de subcanais pode ser obtida pela soma das capacidades individuais, ou seja,

$$C = B \sum_{u=1}^U \log_2 \left(1 + \frac{P_u}{\sigma_u^2} \right) \text{ bit/s.} \quad (51)$$

Assim, considerando a banda B fixa, maximizar a capacidade total C corresponde a maximizar o somatório

$$\frac{C}{B} = \sum_{u=1}^U \log_2 \left(1 + \frac{P_u}{\sigma_u^2} \right) \text{ bit/s/Hz} \quad (52)$$

que, por sua vez, é o mesmo que minimizar $-C/B$. Nessas condições, seja então o caso de um transmissor com $P_T = 1$ watt e $U = 8$ subcanais. Seja $\mathbf{p} \in \mathbb{R}^U$ o vetor de potências médias (P_u) e $\mathbf{n} \in \mathbb{R}^U$ o vetor que contém as potências médias do ruído (σ_u^2). A formulação do problema e a correspondente modelagem no CVX são dadas a seguir:

$$\begin{aligned} &\text{minimize} && - \sum_{u=1}^U \log_2 \left(1 + \frac{P_u}{\sigma_u^2} \right) \\ &\text{sujeito a} && \mathbf{p} \succeq \mathbf{0}, \mathbf{1}^T \mathbf{p} = P_T \end{aligned}$$

```
U = 8;
n = randn(U,1);
PT = 1;
um = ones(U,1)*0.5;
cvx_begin
    variable p(U);
    minimize -sum(log(1 + p./n)./log(2))
    subject to
        -p <= 0;
        PT-(um')*p == 0;
cvx_end
CAP_max_CVX = -cvx_optval
```

Para o vetor aleatório de intensidades de ruído \mathbf{n} dado a seguir, o CVX produziu o vetor ótimo de alocação de potências \mathbf{p}^* e a correspondente capacidade ótima $C(\mathbf{p}^*)/B$ apresentados a seguir:

$$\mathbf{n} = \begin{bmatrix} 0,4090 \\ 0,3301 \\ 0,1710 \\ 0,1449 \\ 0,1706 \\ 0,2670 \\ 0,3636 \\ 0,1546 \end{bmatrix}, \quad \mathbf{p}^* = \begin{bmatrix} 0,0000 \\ 0,0416 \\ 0,2007 \\ 0,2268 \\ 0,2011 \\ 0,1047 \\ 0,0081 \\ 0,2170 \end{bmatrix}, \quad C(\mathbf{p}^*)/B = 5,5483 \text{ bit/s/Hz}$$

Este tipo de problema pode ser resolvido analiticamente aplicando os multiplicadores de Lagrange em (51), usando a restrição $\sum_{u=1}^U P_u = P_T$ e obtendo-se, de acordo com:

$$P_u = \begin{cases} \mu - \sigma_u^2, & \text{se } \mu > \sigma_u^2 \\ 0, & \text{se } \mu \leq \sigma_u^2 \end{cases}, \quad u = 1, \dots, U \quad (53)$$

$$\sum_{u=1}^U P_u = P_T$$

Essa regra para a distribuição de potência entre os subcanais é chamada de *water-filling* e a variável μ está associada à potência total de transmissão [13, p. 721].

Para uma determinada distribuição de ruído pode-se determinar o valor de μ (mi) através da implementação do algoritmo *water-filling*. Um exemplo dessa implementação em Matlab é dado a seguir:

```
K=U;
NT=(um')*n;
pwf=ones(U,1)/U;
Para=0;
while Para==0
    mi=(NT+PT)/K;
    K
    mi
    Para=1;
    for i=1:U
        if n(i)>mi
            if pwf(i)>0
                NT=NT-n(i);
```

```

K=K-1;
pwf(i)=0;
Para=0;
end
else
pwf(i) = mi-n(i);
end
end
end
mi
pwf
Cap_max_wf = sum(log(1 + pwf./n)./log(2))

```

A partir de μ pode-se, via (54), calcular os valores de P_u ótimos (pwf) e, via (52), calcular a capacidade do sistema por unidade de frequência (Cap_max_wf).

Os resultados obtidos por esse algoritmo, para as mesmas entradas, são idênticos àqueles obtidos com o CVX e já apresentados. A partir de resultados como esses foi construído o diagrama da Fig. 2, que ilustra princípio da técnica *water-filling*: perceba que se a variável μ for associada a um determinado nível de água depositando em um recipiente cujo formato é dado pela distribuição dos valores de σ_u^2 , a potência ótima de transmissão será proporcional à diferença entre esse nível e cada patamar de ruído. Em outras palavras, atribui-se maior potência de transmissão ao canal com menor contaminação de ruído.

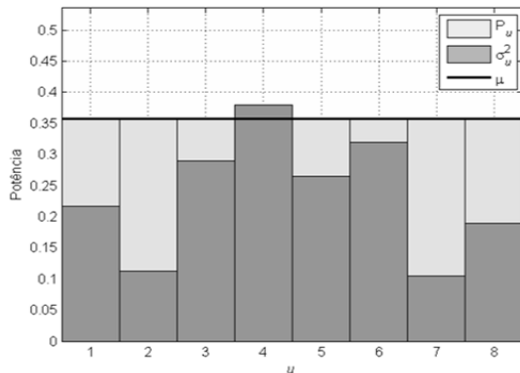


Figura 2. Diagrama que representa a solução da aplicação V.2, por *water-filling*, para alocação de potência a um conjunto de 8 subcanais sujeitos ao ruído colorido.

Aplicação V.3 – Cancelando ruído impulsivo pela combinação de suavização quadrática com suavização por variação total.

O problema proposto aqui consiste na identificação e cancelamento de ruído impulsivo na recepção de um sistema de comunicação. Para isto são formulados e resolvidos dois problemas de otimização bicritérios, parecidos com aquele descrito no Caso III.16, e depois combinadas as soluções para estimar o ruído impulsivo no sinal recebido.

A Fig. 3 é um diagrama em blocos que representa este problema e a solução proposta.

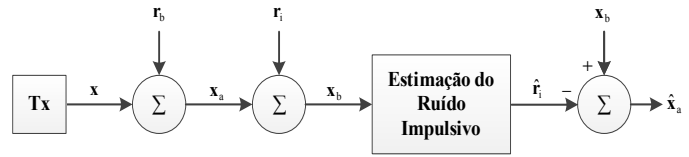


Figura 3. Sistema de comunicação com sinal transmitido (x) corrompido por ruído aditivo Gaussiano branco (r_b) e ruído aditivo impulsivo (r_i), com estimativas do ruído impulsivo (\hat{r}_i) e do sinal sem ruído impulsivo (\hat{x}_a).

O sistema de comunicação da Fig. 3 é composto pelo transmissor (Tx), por uma etapa de adição de ruídos e pela etapa de recepção composta pelo bloco de estimação do ruído impulsivo e pela estimação do sinal corrompido sem ruído impulsivo.

O Tx envia um sinal x (do tipo QPSK [13, pp. 422-432], por exemplo) que em seguida é corrompido por ruído aditivo Gaussiano branco r_b , formando o sinal x_a . O sinal x_a por sua vez é corrompido por ruído aditivo impulsivo r_i , resultando no sinal x_b que chega ao receptor. O bloco de estimação do ruído impulsivo, como o próprio nome sugere, tem como objetivo estimar o ruído impulsivo presente no sinal que está em sua entrada. Para realizar esta estimação são formulados dois problemas de otimização convexa bicritérios em suas formas escalarizadas, tal como no Caso III.16. Um dos problemas é conhecido como problema de otimização convexa regularizado por suavização quadrática [1, p. 312], cuja formulação é

$$\text{minimize } \|x_b - \hat{x}\|_2^2 + \delta \phi_{\text{quad}}(\hat{x})$$

onde $\delta > 0$ é a constante de escalarização do problema,

$$\phi_{\text{quad}}(\hat{x}) = \sum_{i=1}^{n-1} (x_{i+1} - x_i)^2 = \|Dx\|_2^2 \text{ e}$$

$$D \in \mathbb{R}^{(n-1) \times n} = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

Este problema de otimização tem como característica a penalização de qualquer tipo de variação abrupta que esteja acompanhando o sinal recebido x_b . Sendo assim, para um valor de δ adequado, tem-se como resultado praticamente todas as variações rápidas de x_b totalmente eliminadas, restando apenas um ponto ótimo muito parecido com o sinal transmitido x , designado por $\hat{x}_{\text{quad}} = \hat{x}$.

Para modelar este problema optou-se por utilizar a sua solução analítica. Esta solução e sua modelagem no Matlab são dadas respectivamente por:

$$\hat{x}_{\text{quad}} = (I + \delta D^T D)^{-1} x_b$$

```
x_quad=(delta*speye(n+1,n+1)+(D'*D))/(delta*x_b);
```

O outro problema formulado é chamado de problema de otimização convexa regularizado com suavização por variação total [1, p. 312]. Este problema e sua correspondente modelagem no CVX são dados por:

```
minimize  $\|x_b - \hat{x}\|_2^2 + \beta\phi_{TV}(\hat{x})$ 
```

```
n = size(x,1)-1;
TV=220;
cvx_begin quiet
variable x_TV(n+1)
minimize norm(x_TV-x_b)
subject to
norm(x_TV(2:(n+1))-x_TV(1:n),1) <= TV;
cvx_end
```

A diferença deste problema em relação ao problema de otimização por suavização quadrática é que, enquanto a última penaliza fortemente qualquer tipo de variação rápida, a suavização por variação total penaliza mais levemente variações de maior amplitude. Isto faz com que variações de menor amplitude que estejam acompanhando o sinal x_b , tais como o ruído Gaussiano r_b , sejam eliminadas quase que totalmente na solução deste problema, enquanto variações de maior amplitude tais como aquelas presentes em r_i são quase que totalmente preservadas. Como resultado da solução deste problema tem-se então um sinal parecido com a soma do sinal transmitido x com o ruído impulsivo r_i , ou seja, $\hat{x}_{TV} = \hat{x} + \hat{r}_i$. A Fig. 4 mostra as formas de onda correspondentes aos sinais x , x_b , \hat{x}_{TV} e \hat{x}_{quad} , respectivamente.

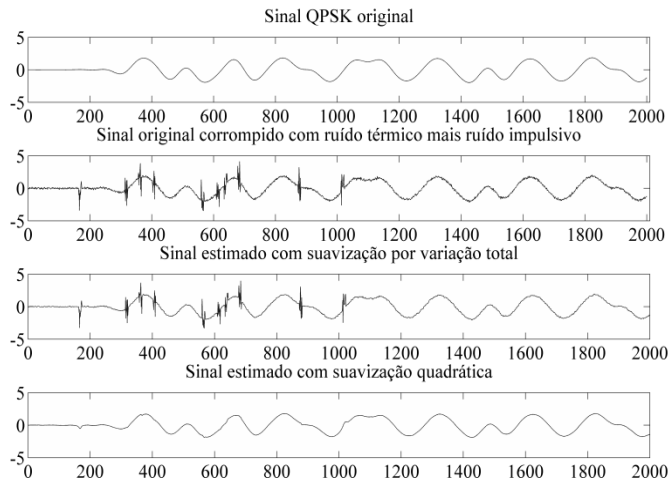


Figura 4. Formas de ondas correspondentes aos sinais x , x_b , \hat{x}_{TV} e \hat{x}_{quad} , respectivamente.

Subtraindo a solução do problema de otimização com suavização quadrática da solução do problema de otimização por suavização por variação total tem-se como resultado o

ruído impulsivo estimado $\hat{r}_i = \hat{x}_{TV} - \hat{x}_{quad}$. A Fig. 5 representa o conteúdo do bloco de estimação do ruído impulsivo. Subtraindo o ruído impulsivo estimado \hat{r}_i do sinal recebido x_b tem-se um sinal parecido com x_a , ou seja, $\hat{x}_a = x_b - \hat{r}_i$, o qual corresponde ao sinal recebido com o ruído impulsivo cancelado.

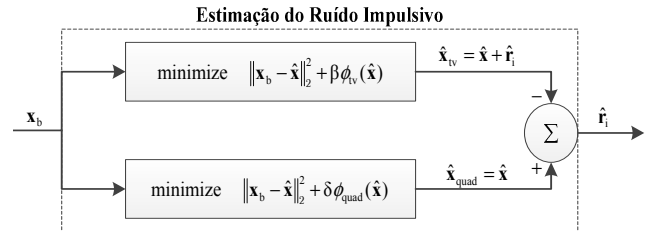


Figura 5. Conteúdo do bloco de estimação de ruído impulsivo.

A Fig. 6 mostra as formas de ondas r_i , \hat{r}_i , x_a e \hat{x}_a , respectivamente.

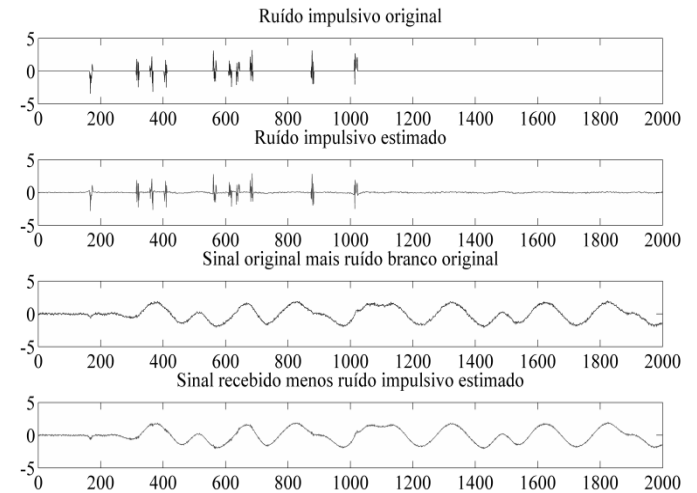


Figura 6. Formas de ondas correspondentes a r_i , \hat{r}_i , x_a e \hat{x}_a , respectivamente.

Calculando o erro quadrático médio entre \hat{x}_a e x_a ,

$$\text{MSE} = \|x_a - \hat{x}_a\|_2^2, \quad (54)$$

verifica-se que o processo de estimação e cancelamento de ruído impulsivo do sinal corrompido x_b é eficiente, pois o valor de MSE será bem próximo de zero, o que significa que a diferença entre \hat{x}_a e x_a é pequena.

Aplicação V.4 – Estimação de um sinal corrompido utilizando técnicas de aproximação robusta.

Deseja-se estimar um sinal transmitido x (com modulação BPSK, por exemplo) a partir de um sinal recebido y

corrompido pelo canal e por adição de ruído. A relação entre \mathbf{x} e \mathbf{y} é dada por

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v} \quad (55)$$

em que $\mathbf{A} \in \mathbb{R}^{m \times n}$ é a matriz que representa o ganho do canal e $\mathbf{v} \in \mathbb{R}^m$ é o ruído.

O sistema de comunicação que representa este problema é mostrado na Fig. 7.

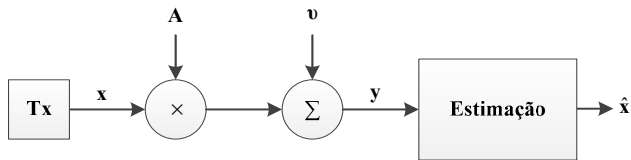


Figura 7. Sistema de comunicação com sinal transmitido \mathbf{x} afetado pelo ganho do canal \mathbf{A} e pelo ruído \mathbf{v} , resultando no sinal recebido $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v}$. A partir de \mathbf{y} , o bloco de estimação encontra o vetor $\hat{\mathbf{x}}$ que mais se aproxima de \mathbf{x} .

O sinal \mathbf{y} chega ao receptor pelo bloco de estimação. Este bloco tem a função de estimar o sinal transmitido \mathbf{x} a partir do sinal recebido \mathbf{y} . O sinal estimado é designado por $\hat{\mathbf{x}}$. Considere que se saiba *a priori* que \mathbf{v} tem pequena intensidade em comparação com a intensidade do sinal e que $\mathbf{A}(u) = \mathbf{A} + u\mathbf{B}$ é o resultado da estimação da matriz do canal onde $\|\mathbf{A}\| = 10$, $\|\mathbf{B}\| = 1$ e $u \in \mathbb{R}$ é uniformemente distribuído em $[-1, 1]$. A matriz aleatória $u\mathbf{B}$ então simula um erro de estimação do canal.

Para encontrar o sinal $\hat{\mathbf{x}}$ é possível modelar problemas de otimização em que o objetivo seja minimizar o resíduo em função de u , o qual é dado por $r(u) = \|\mathbf{A}(u)\hat{\mathbf{x}} - \mathbf{y}\|$. Estes problemas são chamados de problemas de aproximação robusta [1, pp. 318-324].

Nesta aplicação são formulados dois problemas de aproximação robusta conhecidos por aproximação robusta estocástica [1, pp. 318-319] e aproximação robusta do pior caso [1, pp. 319-320]. Também é formulado o problema de otimização nominal, no qual se admite que $u = 0$, o que implica em $\mathbf{A}(u) = \mathbf{A}$.

Para o caso nominal, utilizando a norma Euclidiana ao quadrado, o problema se reduz a um problema de mínimos quadrados (*least-squares*), para o qual se tem a seguinte formulação e modelagem no CVX:

$$\text{minimize} \quad \|\mathbf{A}\hat{\mathbf{x}}_{\text{nom}} - \mathbf{y}\|_2^2$$

```
cvx_begin quiet
    variable x_nom(n)
    minimize norm(A*x_nom - b)
cvx_end
```

onde $\hat{\mathbf{x}}_{\text{nom}} \in \mathbb{R}^n$ é a variável do problema.

Para o caso da aproximação robusta estocástica pode-se utilizar uma variação chamada de problema de aproximação robusta estatística tal como

$$\text{minimize} \quad E \|\mathbf{A}(u)\hat{\mathbf{x}}_{\text{est}} - \mathbf{y}\|_2^2,$$

onde $\hat{\mathbf{x}}_{\text{est}}$ é a variável do problema. Resolvendo a função objetivo tem-se que

$$E \|\mathbf{A}(u)\hat{\mathbf{x}}_{\text{est}} - \mathbf{y}\|_2^2 = \|\mathbf{A}\hat{\mathbf{x}}_{\text{est}} - \mathbf{y}\|_2^2 + \hat{\mathbf{x}}_{\text{est}}^T \mathbf{P} \hat{\mathbf{x}}_{\text{est}},$$

onde $\mathbf{P} = E[u^2] \mathbf{B}^T \mathbf{B} = (1/3) \mathbf{B}^T \mathbf{B}$.

Com a função objetivo resolvida, a formulação e a modelagem deste problema no CVX são dadas por:

$$E \|\mathbf{A}(u)\hat{\mathbf{x}}_{\text{est}} - \mathbf{y}\|_2^2 = \|\mathbf{A}\hat{\mathbf{x}}_{\text{est}} - \mathbf{y}\|_2^2 + \hat{\mathbf{x}}_{\text{est}}^T \mathbf{P} \hat{\mathbf{x}}_{\text{est}}$$

$$\mathbf{P} = E[u^2] \mathbf{B}^T \mathbf{B} = (1/3) \mathbf{B}^T \mathbf{B}$$

```
P = (1/3)*B'*B;
cvx_begin quiet
    variable x_est(n)
    minimize square_pos(norm(A*x_est-b(1:m)))+
    quad_form(x_est,P)
cvx_end
```

A formulação e a modelagem do problema de aproximação robusta por pior caso são dados por:

$$\text{minimize} \quad \max \{ \|\mathbf{A} - \mathbf{B}\hat{\mathbf{x}}_{\text{pc}} - \mathbf{y}\|_2, \|\mathbf{A} + \mathbf{B}\hat{\mathbf{x}}_{\text{pc}} - \mathbf{y}\|_2 \},$$

```
cvx_begin quiet
    variable x_pc(n)
    minimize max( norm((A-B)*x_pc - b,
    norm((A+B)*x_pc - b) ) )
cvx_end
```

onde $\hat{\mathbf{x}}_{\text{pc}} \in \mathbb{R}^n$ é a variável do problema.

Resolvendo os três problemas dados, determinam-se os valores estimados $\hat{\mathbf{x}}_{\text{nom}}$, $\hat{\mathbf{x}}_{\text{est}}$ e $\hat{\mathbf{x}}_{\text{pc}}$, respectivamente. Assim é possível encontrar as curvas de análise de robustez de cada técnica de aproximação para todas as possíveis variações de u . Isto é feito determinando os valores da função resíduo

$$r(u) = \|\mathbf{A}(u)\hat{\mathbf{x}} - \mathbf{y}\|_2^2 \quad (56)$$

para o valor estimado de \mathbf{x} de cada técnica.

A Fig. 8 apresenta curvas de robustez para as técnicas de aproximação aqui analisadas.

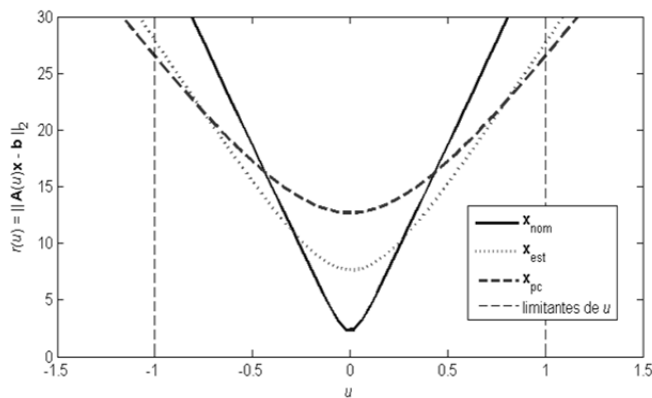


Figura 8. Curvas de análise de robustez da aproximação nominal, estatística e do pior caso.

Pode-se notar que para $u=0$ tem-se a melhor estimativa encontrada pela aproximação nominal, ou seja, é a que resulta no menor valor de resíduo. Entretanto, a aproximação nominal apresenta grande aumento no resíduo para desvios de u em direção a 1 ou -1 . A aproximação por pior caso apresenta grande resíduo para $u=0$, mas seu resíduo não varia muito com a variação de u no intervalo de $[-1, 1]$. A aproximação estocástica se situa entre as outras em termos da variação do resíduo com a variação de u .

A Fig. 9 mostra um conjunto de formas de onda referentes a um sinal transmitido \mathbf{x} do tipo BPSK e os respectivos sinais estimados $\hat{\mathbf{x}}_{\text{nom}}$, $\hat{\mathbf{x}}_{\text{est}}$ e $\hat{\mathbf{x}}_{\text{pc}}$ para uma situação em que $u=0$, ou seja, a matriz \mathbf{A} foi estimada sem erro.

Os erros quadráticos médios calculados para a aproximação nominal, estocástica e de pior caso são, respectivamente: $\text{MSE}_{\text{nom}} = 0,0968$, $\text{MSE}_{\text{est}} = 1,8149$ e $\text{MSE}_{\text{pc}} = 2,7448$. Como era de se esperar, a melhor estimativa realizada dentre as três técnicas para $u=0$ corresponde à aproximação nominal, pois foi a que apresentou menor MSE.

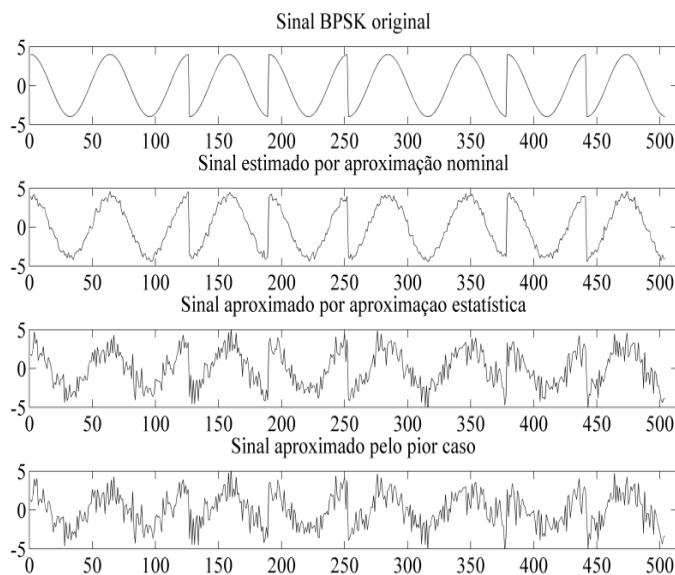


Figura 9. Formas de ondas correspondentes aos sinais \mathbf{x} , \mathbf{x}_{nom} , \mathbf{x}_{est} e \mathbf{x}_{pio} respectivamente.

VI. COMENTÁRIOS FINAIS

O desenvolvimento de soluções que empregam a otimização baseada em programação convexa exige do programador um arcabouço teórico amplo e profundo, especialmente nas áreas de álgebra linear, análise convexa e algoritmos. A aquisição desse arcabouço demanda muito esforço e tempo. No entanto, principalmente na engenharia, há aqueles que necessitam utilizar a otimização convexa em uma aplicação específica e não têm condições de dedicar ao seu estudo e nem podem recorrer à ajuda de um especialista para realizar essa empreitada. Esse tutorial foi desenvolvido objetivando atender esse público, em especial aqueles que trabalham na área de Engenharia de Telecomunicações. Nesse sentido, procurou-se apresentar neste trabalho os principais conceitos da otimização convexa e da sua utilização por meio de programação convexa disciplinada via CVX, de forma objetiva e acompanhada sempre de exemplos. As aplicações selecionadas procuraram complementar essa apresentação e servir de referência para que o leitor possa desenvolver suas próprias soluções. Enfim, espera-se que esse artigo possa contribuir com todos aqueles que necessitem utilizar o CVX, uma ferramenta poderosa para a solução de problemas de otimização convexa, e possa também servir de guia para a aprendizagem e consulta rápida.

REFERÊNCIAS

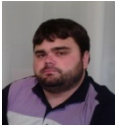
- [1] S. Boyd and L. Vandenberghe, *Convex Optimization*, New York, USA: Cambridge University Press, 2010.
- [2] M. Grant and S. Boyd. *CVX: Matlab software for disciplined convex programming*, version 1.22. Disponível: <http://cvxr.com/cvx/download/>, Feb. 2012.
- [3] M. Grant. *Disciplined Convex Programming*. Ph.D. thesis, Information Systems Laboratory, Department of Electrical Engineering, Stanford University, 2004. Disponível: http://stanford.edu/~boyd/disc_cvx_prog.html.
- [4] M. Grant, S. Boyd and Y. Ye. Disciplined convex programming. In *Global Optimization: from Theory to Implementation, Nonconvex Optimization and its Applications*, L. Liberti and N. Maculan, eds., Springer, 2006. Disponível: http://stanford.edu/~boyd/disc_cvx_prog.html.
- [5] J. Dattorro, *Convex Optimization and Euclidean Distance Geometry*, Palo Alto, CA, USA: MeBoo Publishing, 2010.
- [6] D. P. Bertsekas, *Convex Analysis and Optimization*, Athena Scientific, Belmont, MA, 2003.
- [7] H. Hindi, A Tutorial on Convex Optimization. *American Control Conference*, 2004 June 30-July 2, Boston, MA, USA, 2004.
- [8] H. Hindi, A Tutorial on Convex Optimization II: Duality and Interior Point Methods. *American Control Conference*, 2006 June 14-16, Minneapolis, MN, USA, 2006.
- [9] Z-Q. Luo, and W. Yu, "An introduction to convex optimization for communications and signal processing," *Selected Areas in Communications, IEEE Journal on*, vol.24, no.8, pp.1426-1438, Aug. 2006.
- [10] D. P. Palomar, and Y. C. Eldar, Eds. *Convex Optimization in Signal Processing and Communications*. 1st Ed. Cambridge: Cambridge University Press, 2009.
- [11] IEEE, Special Issue: Advances in Convex Optimization, *IEEE Signal Processing Magazine*, vol.27, no.3, May 2010.
- [12] G. Strang, *Introduction to linear algebra*. 4th Ed., New York, USA: Wellesley-Cambridge Press, 2009.
- [13] D. A. Guimarães, *Digital Transmission: A Simulation-Aided Introduction with VisSim/Comm*, Berlin-Heidelberg, Germany: Springer Verlag, 2009.



Dayan Adionel Guimarães é natural de Carrancas, MG, Brasil. Técnico em Eletrônica pela Escola Técnica de Eletrônica FMC de Santa Rita do Sapucaí, MG, em 1987. Engenheiro Eletricista pelo Instituto Nacional de Telecomunicações de Santa Rita do Sapucaí – INATEL em 1994. Mestre em Engenharia Elétrica nas áreas de Telecomunicações e Telemática pela Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, SP – UNICAMP, em 1996. Doutor em Engenharia Elétrica nas áreas de Telecomunicações e Telemática pela UNICAMP, em 2003. Participou de programa de pós-doutorado na área de Otimização Convexa pela Universidade Federal de Santa Catarina, SC – UFSC, em 2010. Atualmente é professor Titular do INATEL.



Giovanni Henrique Faria Floriano é natural de Itajubá, MG. Graduado em Engenharia Elétrica pela Universidade Federal de Engenharia de Itajubá-UNIFEI em 1988. Mestrando em Telecomunicações no Instituto Nacional de Telecomunicações de Santa Rita do Sapucaí-INATEL. Suas principais áreas de pesquisa são Otimização Convexa, Transmissão Digital, Probabilidade e Processamento Digital de Sinais. Atualmente é professor adjunto do INATEL.



Lucas Chaves Silvestre concluiu o curso Técnico de Eletrônica com ênfase em Controle e Automação pela Escola Técnica de Eletrônica - Francisco Moreira da Costa, Santa Rita do Sapucaí, Minas Gerais, em 2004. Graduiu-se em Engenharia Elétrica pelo Instituto Nacional de Telecomunicações (INATEL), Santa Rita do Sapucaí, Minas Gerais, em 2010. De 2007 a 2012 trabalhou no setor de Pesquisa e Desenvolvimento de Sistemas Embarcados. Mestre em Telecomunicações pelo INATEL desde dezembro de 2014, onde suas pesquisas concentraram nas áreas de Otimização Convexa, Transmissão Digital, Probabilidade e Processamento Digital de Sinais. Atualmente é Professor Assistente do Instituto Politécnico do Centro Universitário Una em Contagem, Minas Gerais.