



Universidade do Estado do Rio de Janeiro
Centro de Tecnologia e Ciências
Instituto Politécnico

Janaína Imbiriba da Costa

Inicialização Global Topográfica aplicada a problemas de
autovalores e de otimização com restrições mistas

Nova Friburgo

2022

Janaína Imbiriba da Costa

Inicialização Global Topográfica aplicada a problemas de autovalores e de otimização com restrições mistas

Tese apresentada como requisito parcial para obtenção do título de Doutora, ao Programa de Pós-Graduação em Modelagem Computacional, da Universidade do Estado do Rio de Janeiro.

Orientador: Prof. Dr. Luiz Nélio Henderson Guedes de Oliveira

Nova Friburgo

2022

CATALOGAÇÃO NA FONTE
UERJ / REDE SIRIUS / BIBLIOTECA CTC/E

C837 Costa, Janaína Imbiriba da.
Inicialização Global Topográfica aplicada a problemas de autovalores e de otimização com restrições mistas / Janaína Imbiriba da Costa. - 2022.
119 f. : il.

Orientadores: Luiz Nélio Henderson Guedes de Oliveira.
Tese (doutorado) – Universidade do Estado do Rio de Janeiro, Instituto Politécnico.

1. Otimização matemática– Teses. 2. Autovalores – Teses. I. Oliveira, Luiz Nélio Henderson Guedes de. II.. Universidade do Estado do Rio de Janeiro. Instituto Politécnico. III. Título.

CDU 512.64

Bibliotecária Pâmela Lisboa CRB7/ 5965

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta Tese, desde que citada a fonte.

Janaína Imbiriba da Costa
Assinatura

25 de abril de 2023
Data

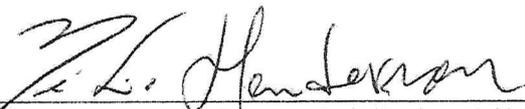
Janaína Imbiriba da Costa

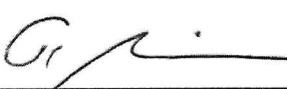
**Inicialização global topográfica aplicada a problemas de autovalores e de
otimização com restrições mistas**

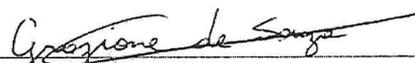
Tese apresentada como requisito parcial para
obtenção do título de Doutor(a) ao Programa de
Pós-Graduação em Modelagem Computacional
do Instituto Politécnico, da Universidade do
Estado do Rio de Janeiro.

Aprovada em 28 de novembro de 2022

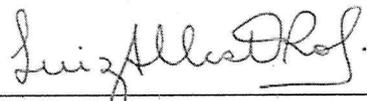
Banca examinadora:


Prof. Luiz Nélio Henderson Guedes de Oliveira, D.Sc. (Orientador)
Instituto Politécnico - UERJ


Prof. Joaquim Teixeira de Assis, D.Sc.
Instituto Politécnico - UERJ


Prof. Graziene de Souza Boy, D.Sc.
Instituto Politécnico - UERJ


Prof. Sanderson Lincoln Gonzaga de Oliveira, D.Sc.
Universidade Federal de São Paulo - Unifesp


Prof. Luiz Alberto Oliveira Rocha, D.Sc.
Universidade do Vale do Rio dos Sinos - UNISINOS

Nova Friburgo

2022

AGRADECIMENTOS

Agradeço, primeiramente, a Deus, pela sua bondade e misericórdia, por me conceder vida através de seu Filho e permitir que eu conquistasse muitas vitórias mesmo quando eu não acreditava ser capaz. Sem a Sua vontade soberana nada poderia ser feito, e mesmo nos momentos mais difíceis sempre senti sua Destra fiel. Obrigada Pai Celeste.

A meu esposo, Marroni, que foi meu maior encorajador para que eu fizesse pós-graduação. Agradeço por sua paciência, carinho e amor quando me dedicou várias horas de sua vida para me ensinar e até ouvir minhas reclamações. Sem dúvida, não há melhor professor e terapeuta.

A meus pais, que mesmo de longe estavam na torcida por mim, sendo meu abrigo depois de cada prova e cada etapa vencida.

A minha irmã e meu cunhado, por sua confiança no meu trabalho, sempre me incentivando e me fortalecendo.

A família da minha esposa, pelas orações e pelo apoio que nos deram em todo o período que estivemos em Nova Friburgo.

Ao professor Dr. Luiz Nélio, pelo incentivo e por ter confiado na realização deste trabalho.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Finalmente, agradeço a Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ) pelo apoio financeiro.

Finalmente, irmãos, tudo o que é verdadeiro, tudo o que é respeitável, tudo o que é justo, tudo o que é puro, tudo o que é amável, tudo o que é de boa fama, se alguma virtude há e se algum louvor existe, seja isso o que ocupe o vosso pensamento.

Filipenses 4, 8.

RESUMO

COSTA, Janaína Imbiriba. *Inicialização Global Topográfica aplicada a problemas de autovalores e de otimização com restrições mistas*. 2022. 119 f. Tese (Doutorado em Modelagem Computacional) - Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, 2022.

Os métodos matemáticos clássicos, desenvolvidos a partir da análise no \mathbb{R}^n , são amplamente empregados na solução de diferentes problemas da matemática, física e engenharias. Em geral, esses métodos realizam uma busca local a partir de uma estimativa inicial. Esse fato, pode apresentar grandes dificuldades, pois em muitos casos, pontos iniciais adequados não estão disponíveis na formulação dos problemas. Nesse contexto, a Inicialização Global Topográfica é um método eficiente para a seleção de estimativas iniciais adequadas. Para isso, o método faz uso de conceitos da teoria de grafos para refinar um conjunto de pontos amostrais uniformemente gerados no conjunto viável. Entretanto, existem casos em que o ponto amostral mais próximo de uma solução não é selecionado devido ao fato de um dos seus vizinhos mais próximos se encontrar em um pequeno vale. Para contornar essa dificuldade, no presente trabalho propomos uma modificação no processo de seleção das estimativas iniciais. Tal modificação permite ainda, em certos casos, o uso de menos pontos amostrais. Utilizando a Inicialização Global Topográfica Modificada resolvemos os problemas de autovalores de matrizes reais simétricas, autovalores complementares e de minimização com restrições de igualdade e desigualdade. Para verificar a eficiência da Inicialização Global Topográfica Modificada na seleção de pontos iniciais para os métodos de busca local, foram realizados testes numéricos utilizando problemas descritos na literatura. Em todos os testes, inclusive os que apresentam várias soluções, a metodologia aqui apresentada se mostrou robusta no cálculo das soluções.

Palavras-chave: Inicialização Global Topográfica Modificada. Problema de Autovalor. Otimização restrita. Complementaridade mista.

ABSTRACT

COSTA, Janaína Imbiriba. *Topographic Global Initialization applied to eigenvalues and optimization problems with mixed constraints*. 2022. 119 f. Tese (Doutorado em Modelagem Computacional) - Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, 2022.

The classical mathematical methods, developed from the \mathbb{R}^n analysis, have widely been used in the solution of different problems in mathematics, physics and engineering. In general, these methods perform a local search based on an initial estimate. This fact can present great difficulties, because in many cases, adequate starting points are not available in the formulation of the problems. In this context, the Global Topographic Initialization is an efficient method for selecting suitable initial estimates. For this, the method makes use of graph theory concepts to refine a set of sample points uniformly generated in the feasible set. However, there are cases in which the closest sample point to a solution is not selected due to the fact that one of its nearest neighbors is in a small valley. To overcome this difficulty, in the present work we propose a modification in the selection process of the initial estimates. Such modification also allows, in certain cases, the use of fewer sample points. Using the Modified Topographic Global Initialization we solve the eigenvalues problem of real symmetric matrices, complementary eigenvalues and minimization with equality and inequality constraints. In order to test the Modified Topographic Global Initialization performance in selection of initial estimates for the local search methods, we have carried out computational experiments using problems described in the literature. In all tests, including those with several solutions, the methodology presented here proved to be robust in calculating the solutions.

Keywords: Modified Topographic Global Initialization. Eigenvalue Problem. Constrained Optimization. Mixed Complementarity.

LISTA DE FIGURAS

Figura 1 - Grafo G	18
Figura 2 - Grafo direcionado G	18
Figura 3 - Exemplos de cones.	22
Figura 4 - Curvas de níveis da função objetivo ϕ na caixa S	27
Figura 5 - O 4^+ -topografo de ϕ	29
Figura 6 - Gráfico da função real ϕ	31
Figura 7 - Pontos gerados a partir da sequência Sobol.	35
Figura 8 - Pontos gerados a partir do gerador Mersenne Twister.	36
Figura 9 - Círculos de Gerschgorin para a matriz A	39
Figura 10 - Transformação de Householder	43
Figura 11 - Gráfico da função polinomial $p_{\bar{A}}$	57
Figura 12 - Gráfico da função auxiliar $f_{\bar{A}}$	58

LISTA DE TABELAS

Tabela 1 - Doze pontos amostrais viáveis gerados pela sequência de Sobol, com os respectivos valores da função objetivo ϕ	26
Tabela 2 - Seis pontos amostrais viáveis gerados pela sequência de Sobol, com os respectivos valores da função objetivo ϕ	32
Tabela 3 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - <i>topografo</i> (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 4$	60
Tabela 4 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - <i>topografo</i> (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 5$	61
Tabela 5 - Número mínimo de pontos amostrais utilizados pelo TGO/Newton-Maehly (TN_{min}) com $k = 4$, o número de minimizadores locais de ϕ no k^+ - <i>topografo</i> (TN_s) e o tempo t_T^* utilizado.	62
Tabela 6 - Número mínimo de pontos amostrais utilizados pelo TGO/Newton-Maehly (TN_{min}) com $k = 5$, o número de minimizadores locais de ϕ no k^+ - <i>topografo</i> (TN_s) e o tempo t_T^* utilizado.	63
Tabela 7 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - <i>topografo</i> (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 4$, sem o uso do logaritmo de $p_{\bar{A}}(\lambda)$	64
Tabela 8 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - <i>topografo</i> (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 5$, sem o uso do logaritmo de $p_{\bar{A}}(\lambda)$	65
Tabela 9 - Limite superior \bar{z} do intervalo no qual a variável auxiliar z está definida em cada teste. Limite inferior l_λ e superior u_λ do intervalo dos autovalores complementares. Número n_λ de autovalores complementares em cada teste.	73
Tabela 10 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 1.	73
Tabela 11 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 2.	74
Tabela 12 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 3.	74

Tabela 13 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 4.	75
Tabela 14 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 5.	75
Tabela 15 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 6.	76
Tabela 16 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 7.	77
Tabela 17 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 8.	77
Tabela 18 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 9.	79
Tabela 19 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 10.	80
Tabela 20 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 11.	80
Tabela 21 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 12.	81
Tabela 22 - Número mínimo de pontos amostrais utilizados pelo MTGO (\overline{NMz}), com as respectivas quantidades de pontos viáveis (NMz) e de estimativas iniciais (NMz_0).	83
Tabela 23 - Número de avaliações $Eval_{\overline{NMz}}$ da função ϕ e o tempo de CPU (em segundos) necessário para solucionar cada problema teste utilizando a quantidade \overline{NMz} de pontos amostrais.	84
Tabela 24 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 1-8.	97
Tabela 25 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 9-12.	97
Tabela 26 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 13-15.	98
Tabela 27 - Número mínimo de pontos amostrais utilizados pelo MTGO-FDIPA (\overline{NMx}) e TGO-FDIPA (\overline{NTx}), com as respectivas quantidade de pontos viáveis (NMx) e (NTx), e de estimativas iniciais (NMx_0) e (NTx_0). Número mínimo de pontos amostrais (\overline{NSx}) utilizados pelo NFDIPA com o respectivo número de pontos viáveis (NSx).	99
Tabela 28 - Número mínimo de pontos amostrais quando não se utilizou a variável auxiliar x_n . \overline{NM} , \overline{NT} e \overline{NS} indicam, respectivamente, o número mínimo necessário para o MTGO-FDIPA, TGO-FDIPA e NFDIPA.	100

Tabela 29 - Número de avaliações $Eval_{NMx}$, $Eval_{NTx}$ e $Eval_{NSx}$ da função ϕ necessárias para solucionar os problemas 1-15, utilizando os métodos MTGO-FDIPA, TGO-FDIPA e NFDIPA, respectivamente.	101
Tabela 30 - Tempo (em segundos) necessário para obter as soluções descritas nas Tabelas 24-26 utilizando os métodos MTGO-FDIPA e NFDIPA	103
Tabela 31 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 16-18.	103
Tabela 32 - Número de avaliações da função objetivo necessárias para solucionar os problemas 16-18 utilizando o MTGO-FDIPA e 6 outros métodos. .	104

SUMÁRIO

	INTRODUÇÃO	13
1	PRELIMINARES	17
1.1	Conceitos Básicos	17
2	INICIALIZAÇÃO GLOBAL TOPOGRÁFICA MODIFICADA	24
2.1	Descrição do método	24
2.2	Gerador de Amostras Uniformes	33
2.3	Algoritmo MTGO/Busca Local	36
3	MÉTODOS NUMÉRICOS PARA O CÁLCULO DE AUTOVA- LORES	37
3.1	Descrição do problema de autovalores	37
3.2	Método de Householder	41
3.2.1	<u>Transformação de Householder</u>	42
3.2.2	<u>Algoritmo de Householder</u>	43
3.3	Método de Newton	48
3.4	Método de Newton-Maehly	49
3.5	Fatoração LU	50
3.6	Diferenciação Algorítmica	54
3.7	Função auxiliar para o MTGO	57
3.8	Testes numéricos	59
4	PROBLEMA DE AUTOVALOR COMPLEMENTAR	67
4.1	Descrição do problema	67
4.2	Algoritmo FDA-MNCP	70
4.3	Testes numéricos	72
5	PROBLEMA DE OTIMIZAÇÃO RESTRITA MODIFICADO .	85
5.1	Formulação do problema	85
5.2	Reformulação do problema utilizando uma variável auxiliar . . .	86
5.3	Método de busca local FDIPA	91
5.4	Testes numéricos	96
	CONCLUSÕES	105
	REFERÊNCIAS	107

APÊNDICE A - Matrizes para o problema de autovalor complementar . 112

APÊNDICE B - Problemas de otimização com restrições mistas 114

INTRODUÇÃO

Ao estudar algum aspecto particular de um problema do mundo real, muitas vezes recorre-se ao uso de modelos matemáticos. Tais modelos, são expressos por equações que descrevem as leis e restrições físicas. Em geral, somente em situações particulares, simplificadas, se pode obter uma solução analítica para essas equações (CHAPRA e CANALE, 2010). Assim, por vezes, é mais favorável definir o modelo do que resolvê-lo.

Para contornar essa dificuldade, recorre-se ao uso de métodos numéricos. Tal abordagem pode ser entendida como um conjunto de técnicas pelas quais os problemas matemáticos são formulados para que possam ser resolvidos com operações aritméticas (CHAPRA e CANALE, 2010). Utilizando esses métodos podemos resolver várias classes de problemas, como por exemplo, minimização de funções sujeitas a restrições (LUENBERGER, 1984), resolução de sistemas de equações não lineares (ISMAILOV e SOLODOV, 2009), solução de equações diferenciais (EPPERSON, 2013), etc.

Os métodos numéricos que derivam do cálculo diferencial, como os métodos tipo Newton, são geralmente muito eficientes e realizam uma busca local a partir de um dado ponto inicial (NOCEDAL e WRIGHT, 2006). Entretanto, pontos iniciais adequados nem sempre estão disponíveis na formulação dos problemas, o que é uma limitação para o uso desses métodos. Tal limitação é ainda mais evidente para problemas de otimização, onde podem existir várias soluções locais (problemas multimodais). Nesse caso, o método pode convergir para soluções locais e não globais.

Uma forma natural para solucionar o problema dos pontos iniciais, é gerar uma quantidade suficientemente grande de pontos amostrais que serão utilizados como estimativas iniciais para o método de busca local. Entretanto, tal metodologia pode apresentar um expressivo custo computacional, pois um grande número de buscas locais serão realizadas. Além disso, podemos ter vários dados iniciais conduzindo a uma mesma solução, ou ainda, podem haver pontos para os quais o método não convergirá.

Assim, vê-se a necessidade de refinar as amostras, ou seja, obter dentre os pontos amostrais aqueles que estão suficientemente próximos das soluções. Dessa forma, o objetivo é selecionar pontos amostrais adequados para os quais o método de busca local convergirá para diferentes soluções, e ao fim do processo espera-se que todas as soluções tenham sido obtidas.

Nesse contexto, a Inicialização Global Topográfica, descrita pelo algoritmo TGO (*Topographical Global Optimization*), é um método eficiente para a tarefa de refinamento das estimativas iniciais, ver Törn (1986, 1990) e Henderson et al. (2015). Nesse método, um ponto amostral P é selecionado como estimativa inicial se o valor de uma determinada função ϕ em P é menor ou igual aos valores que essa função assume nos k pontos amostrais mais próximos de P , onde k é um inteiro maior ou igual a 1. Note que a Inicia-

lização Global Topográfica deve ser empregada antes da etapa de busca local, ou seja, ela inicia o processo de resolução. Por esse motivo, utilizamos o termo “Inicialização”. Por simplicidade, algumas vezes vamos nos referir a esse método apenas por TGO.

Dada uma quantidade \bar{N} de pontos amostrais, quanto maior o valor do parâmetro k menos estimativas iniciais são obtidas pela Inicialização Global Topográfica. Por outro lado, quanto menor o valor de k mais pontos iniciais são obtidos, alguns destes convergindo para a mesma solução. Estes fatos mostram a importância do parâmetro k para uma boa performance desse método de refinamento. Por esse motivo, em Henderson et al. (2015) os autores propuseram uma fórmula para o cálculo do parâmetro k , obtendo na maioria dos seus experimentos os valores 4 e 5.

Recentemente, a Inicialização Global Topográfica foi empregada na solução de uma variedade de problemas, como por exemplo, na solução de problemas de otimização restritos e irrestritos, ver Henderson et al. (2015), Costa (2016) e Henderson et al. (2018). Esse método também foi utilizado na resolução de sistemas de equações não lineares, ver Henderson, Sá Rêgo e Imbiriba (2016).

Apesar da eficiência demonstrada pelo TGO na seleção dos pontos iniciais para resolução de diferentes tipos de problemas, existem casos em que um ponto mais próximo de uma solução não é selecionado devido ao fato de um dos seus vizinhos mais próximos se encontrar em um pequeno vale, apresentando um valor menor para a função ϕ . Tal problema pode ser contornado gerando mais pontos amostrais.

No presente trabalho, propomos uma solução alternativa para esse problema, modificando o processo de seleção das estimativas iniciais. Tal modificação estabelece que um ponto que esteja bem próximo da solução não seja penalizado pelo fato de um dos seus vizinhos se encontrar em um pequeno vale. Em certos casos, essa abordagem, chamada de Inicialização Global Topográfica Modificada, denotada por MTGO, possibilita o uso de menos pontos amostrais.

Objetivos gerais

Utilizar a Inicialização Global Topográfica Modificada na solução do problema de autovalor para matrizes reais simétricas (GANDER, 2008), no cálculo de autovalores complementares (FERNANDES et al., 2014a), (FERNANDES et al., 2014b) e na resolução do problema de minimização com restrições de igualdade e desigualdade (COSTA, 2016).

Objetivos específicos

Os objetivos específicos desta tese incluem:

1. Modificar o processo de seleção das estimativas iniciais na Inicialização Global Topográfica, visando a melhor seleção das estimativas iniciais e a redução do número de pontos amostrais necessários.
2. Obter os autovalores de matrizes reais simétricas sem calcular os coeficientes do polinômio característico. Para isso utilizaremos o MTGO para selecionar pontos iniciais para o método de busca local Newton-Maehly (STOER e BULIRSCH, 1993).
3. Reescrever o problema de autovalores complementares utilizando uma variável auxiliar. Resolver tal problema empregando o MTGO para gerar estimativas iniciais adequadas para o método de direções viáveis descrito em Mazorche (2007).
4. Reescrever o problema de minimização de uma função sujeita a restrições de igualdade e desigualdade utilizando uma variável auxiliar. Solucionar tal problema utilizando o MTGO para gerar estimativas iniciais para o método de ponto interior proposto em Herskovits (1998).

Contribuições

Como contribuições desta tese, temos:

1. A Inicialização Global Topográfica Modificada (MTGO) é uma proposta inédita, sendo uma importante contribuição científica.
2. O uso do MTGO na resolução dos problemas de autovalores de matrizes reais simétricas, autovalores complementares e do problema de minimização com restrições de igualdade e desigualdade, esses dois últimos utilizando uma variável auxiliar, é uma abordagem não encontrada na literatura, tendo sido desenvolvido pela primeira vez no presente trabalho.

Organização da tese

Este trabalho está organizado da seguinte forma:

No Capítulo 1 são apresentadas as notações e definições importantes que serão utilizadas nos demais capítulos.

No Capítulo 2 é feita uma revisão da Inicialização Global Topográfica. Também é descrita a modificação do processo de seleção das estimativas iniciais. Em seguida, é descrito o gerador de amostras uniformes chamado de sequência de Sobol (SOBOL, 1967). Para concluir o capítulo, é apresentado o Algoritmo MTGO.

No Capítulo 3 é descrito o problema de autovalor de matrizes reais simétricas. Em seguida é descrita a transformação de Householder (POOLE, 2011), o método de Newton para solução de sistemas de equações não lineares (ISMAILOV e SOLODOV, 2009) e o método de busca local Newton-Maehly. Nesse mesmo capítulo, é apresentada a fatoração LU e a diferenciação algorítmica (GANDER, 2008). Por fim, são apresentados os resultados dos testes numéricos aplicando o MTGO na seleção das estimativas iniciais para o método Newton-Maehly.

No Capítulo 4 é apresentado o problema de autovalor de complementaridade e a sua reformulação utilizando uma variável auxiliar. Também é descrito o método de direções viáveis, com busca linear de Armijo (ISMAILOV e SOLODOV, 2009). Por fim, são apresentados os resultados da aplicação do MTGO na resolução de alguns problemas testes, onde a busca local foi realizada pelo método FDA-MNCP (*Feasible Direction Algorithm-Mixed Nonlinear Complementarity Problem*), ver Mazorche (2007).

No Capítulo 5 é descrito o problema de minimização de uma função sujeita a restrições de igualdade e desigualdade. Também é apresentada a reformulação desse problema utilizando uma variável auxiliar. Em seguida, é apresentado o método Quasi-Newton BFGS. Para finalizar, é descrito o método FDIPA (HERSKOVITS, 1998) e os resultados dos testes numéricos obtidos empregando o MTGO no refinamento das estimativas iniciais.

Por fim, tem-se a conclusão deste trabalho, além de se apresentar as referências bibliográficas e os apêndices.

1 PRELIMINARES

Neste capítulo, apresentamos algumas definições e notações para facilitar a leitura e o entendimento dos assuntos que serão abordados no desenvolvimento do trabalho.

1.1 Conceitos Básicos

Todos os vetores são considerados aqui como vetores colunas. Denota-se por $\|x\|$ a norma do vetor x no espaço Euclidiano n -dimensional, isto é,

$$\|x\| = \left(\sum_{i=1}^n x_i^2 \right)^{\frac{1}{2}}, \quad (1)$$

onde $x \in \mathbb{R}^n$ e $x_i \in \mathbb{R}$ é a i -ésima componente do vetor x .

O produto interno usual no espaço Euclidiano n -dimensional entre dois vetores $x, y \in \mathbb{R}^n$ é denotado por $x^T y$, ou ainda $\langle x, y \rangle$, e corresponde ao número real dado por

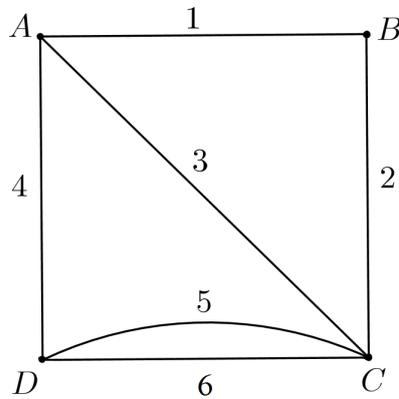
$$x^T y = \sum_{i=1}^n x_i y_i. \quad (2)$$

Como consequência da definição anterior temos que $\|x\|^2 = x^T x$.

A Inicialização Global Topográfica Modificada, que será apresentada no Capítulo 2, faz uso dos conceitos de grafos para selecionar pontos iniciais adequados para os métodos de busca local. Por esse motivo, definimos a seguir o conceito de grafo direcionado.

Definição 1: Um grafo G consiste de um conjunto finito VG de elementos chamados vértices, um conjunto finito AG de elementos de ligação chamados arestas (arcos) e uma função de incidência ψ_G que associa a cada aresta α de G um par de vértices, não necessariamente distintos, chamados de extremos de α (BOLLOBÁS, 1998). Na Fig. 1, temos um exemplo de grafo, com $VG = \{A, B, C, D\}$, $AG = \{1, 2, 3, 4, 5, 6\}$ e $\psi_G(1) = \{A, B\}$, $\psi_G(2) = \{B, C\}$, $\psi_G(3) = \{A, C\}$, $\psi_G(4) = \{A, D\}$, $\psi_G(5) = \{C, D\}$ e $\psi_G(6) = \{C, D\}$.

Figura 1 - Grafo G.

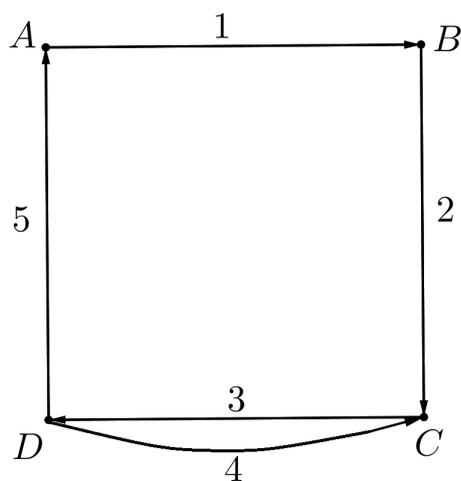


Legenda: Os vértices são indicados por $\{A, B, C, D\}$. As arestas são indicadas como $\{1, 2, 3, 4, 5, 6\}$.

Fonte: A autora, 2022.

Definição 2: Se a função de incidência ψ_G associa a cada aresta α de G um par ordenado de vértices de G, sendo o primeiro a extremidade inicial do arco e o outro a sua extremidade final, então o grafo G é dito direcionado (BOLLOBÁS, 1998). Na Fig. 2, temos um exemplo de grafo direcionado.

Figura 2 - Grafo direcionado G.



Legenda: Os vértices são indicados por $\{A, B, C, D\}$. As arestas (arcos) são indicadas como $\{1, 2, 3, 4, 5\}$.

Fonte: A autora, 2022.

A seguir, apresentamos algumas definições da Álgebra Linear, que serão utilizadas no estudo dos problemas de autovalores e de otimização restrita. Tais definições referem-se aos casos especiais de matrizes, determinantes e as relações entre matrizes.

Definição 3: Uma matriz $A \in \mathbb{C}^{n \times n}$ diz-se simétrica se coincidir com sua transposta, ou seja, $A = A^T$. No caso particular em que A é uma matriz real, temos que A é simétrica se, e somente se, $\langle Ax, y \rangle = \langle x, Ay \rangle$, para todo $x, y \in \mathbb{R}^n$.

Definição 4: Uma matriz simétrica $A \in \mathbb{R}^{n \times n}$ é semidefinida positiva se, $d^T A d \geq 0$ para todo $d \in \mathbb{R}^n \setminus \{0\}$. Quando $d^T A d \leq 0$ para todo $d \in \mathbb{R}^n \setminus \{0\}$, diz-se que a matriz é semidefinida negativa.

Definição 5: Uma matriz simétrica $A \in \mathbb{R}^{n \times n}$ é definida positiva se $d^T A d > 0$ para todo $d \in \mathbb{R}^n \setminus \{0\}$. Quando $d^T A d < 0$ para todo $d \in \mathbb{R}^n \setminus \{0\}$, diz-se que a matriz é definida negativa.

Definição 6: Uma matriz quadrada $A \in \mathbb{R}^{n \times n}$ com todas as entradas acima da diagonal principal nulas é denominada triangular inferior, e uma matriz quadrada com todas as entradas abaixo da diagonal principal nulas é denominada triangular superior (ANTON e RORRES, 2012).

Definição 7: Se a matriz quadrada $A \in \mathbb{R}^{n \times n}$ é simultaneamente triangular inferior e superior, então A é denominada matriz diagonal. Além disso, se A é diagonal com $a_{ii} = 1$, $i = 1, \dots, n$, então A é chamada de matriz identidade, denotada por I_n .

Definição 8: Diz-se que a matriz $A \in \mathbb{R}^{n \times n}$ está na forma de Hessenberg superior se $a_{ij} = 0$ para todos i, j com $i > j + 1$ (ANTON e RORRES, 2012). Por outro lado, se $a_{ij} = 0$ para todos i, j com $j > i + 1$, diz-se que a matriz A está na forma de Hessenberg inferior.

Definição 9: Se $A \in \mathbb{R}^{n \times n}$ está na forma de Hessenberg superior e, ao mesmo tempo, na forma de Hessenberg inferior, então A é dita uma matriz tridiagonal (BURDEN; FAIRES; BURDEN, 2015).

Definição 10: O determinante de uma matriz A é o número definido como

$$\det(A) = \sum_{j=1}^n a_{ij} C_{ij}, \text{ para algum } i = 1, \dots, n. \quad (3)$$

Aqui, C_{ij} é o (i, j) -cofator de A , dado por $C_{ij} = (-1)^{i+j} \det(A_{ij})$, onde A_{ij} é a submatriz da matriz A obtida deletando a linha i e a coluna j (POOLE, 2011).

Do ponto de vista computacional, utilizar a expansão em cofatores da Eq. (3) para o cálculo do determinante de matrizes grandes, pode apresentar um elevado custo.

Por esse motivo, uma alternativa é recorrer aos métodos numéricos baseados na redução por linhas. Tais métodos fazem uso das operações elementares, a saber: multiplicação de uma linha por uma constante não nula c , troca de duas linhas entre si e a soma de uma linha a uma outra linha previamente multiplicada por uma constante c .

De acordo com Poole (2011), uma operação elementar pode ser feita utilizando a multiplicação matricial. Para isso, utiliza-se a matriz elementar E obtida da matriz identidade I_n efetuando a operação elementar sobre linhas. Assim, quando a matriz A é multiplicada à esquerda por uma matriz elementar E , o efeito é o de efetuar uma operação elementar com as linhas de A .

Se B é a matriz que resulta de A quando esta é multiplicada à esquerda por uma sequência de matrizes elementares, então a matriz A poder ser recuperada de B efetuando a multiplicação por uma sequência de matrizes elementares, ou seja, existe uma segunda sequência de matrizes elementares, que sendo aplicada a B , recupera A . Nesse caso, as matrizes A e B são ditas equivalentes por linhas (ANTON e RORRES, 2012). No capítulo 3, será apresentado um método de redução por linhas, utilizando operações elementares, para o cálculo do determinante de uma matriz.

Definição 11: Se $A \in \mathbb{R}^{n \times n}$ for uma matriz quadrada e se pudermos encontrar uma matriz $A^{-1} \in \mathbb{R}^{n \times n}$, tal que $AA^{-1} = A^{-1}A = I_n$, então diremos que A é invertível e que A^{-1} é a inversa de A . Além disso, a matriz A^{-1} existe se, e somente se, $\det(A) \neq 0$ (ANTON e RORRES, 2012).

Definição 12: Dizemos que uma matriz quadrada $A \in \mathbb{R}^{n \times n}$ é ortogonal se sua transposta for sua inversa, ou seja, se $A^{-1} = A^T$.

Definição 13: Se A e B forem matrizes quadradas, dizemos que B é semelhante a A se existir alguma matriz invertível P tal que $B = P^{-1}AP$.

As matrizes semelhantes tem muitas propriedades em comum, por exemplo, a igualdade dos determinantes. Em geral, dizemos que uma propriedade de matrizes é invariante por semelhança ou que a propriedade é um invariante de semelhança, se ela for compartilhada por quaisquer duas matrizes semelhantes (ANTON e RORRES, 2012). No capítulo 3, veremos uma outra propriedade invariante, chamada de autovalor.

Para solucionar os problemas de autovalor, autovalor complementar e de otimização restrita faremos uso de métodos de busca local desenvolvidos a partir da análise matemática clássica. As definições a seguir serão muito úteis no estudo desses métodos e na análise dos problemas.

Definição 14: Uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita diferenciável em $x \in \mathbb{R}^n$ se existe um vetor $\nabla f(x) \in \mathbb{R}^n$ e uma função $\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}$ tal que $\varepsilon(d) \rightarrow 0$ quando $\|d\| \rightarrow 0$ e para todo $d \in \mathbb{R}^n$

$$f(x + d) = f(x) + \nabla f(x)^T d + \|d\| \varepsilon(d). \quad (4)$$

O vetor $\nabla f(x)$ é o gradiente de f em x definido como

$$\nabla f(x) = \left(\frac{\partial f(x)}{\partial x_1}, \dots, \frac{\partial f(x)}{\partial x_n} \right)^T, \quad (5)$$

onde as componentes $\frac{\partial f(x)}{\partial x_n}$, $i = 1, 2, \dots, n$, são as derivadas parciais de primeira ordem da função f . Se as derivadas parciais de f são todas contínuas, então a função f é dita continuamente diferenciável. A classe das funções com tal propriedade é representada por C^1 .

Definição 15: A Hessiana de f em x é a matriz pertencente ao espaço $\mathbb{R}^{n \times n}$ definida por

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \dots & \frac{\partial^2 f(x)}{\partial x_n \partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} & \dots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}, \quad (6)$$

onde os elementos são as derivadas parciais de segunda ordem da função f . Se as derivadas parciais de segunda ordem de f são todas contínuas, então f é dita duas vezes continuamente diferenciável e a classe das funções com tal propriedade é representada por C^2 . Além disso, se $f \in C^2$ então $\nabla^2 f(x)$ é uma matriz simétrica (LIMA, 2010).

Definição 16: A função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita duas vezes diferenciável em $x \in \mathbb{R}^n$ se o vetor gradiente e a matriz Hessiana de f em x estiverem definidos e existir uma função $\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}$ tal que para todo $d \in \mathbb{R}^n$,

$$f(x + d) = f(x) + \nabla f(x)^T d + \frac{1}{2} d^T \nabla^2 f(x) d + \|d\| \varepsilon(d), \quad (7)$$

onde $\varepsilon(d) \rightarrow 0$ quando $\|d\| \rightarrow 0$.

Definição 17: Seja $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ uma aplicação diferenciável em $x \in \mathbb{R}^n$, ou seja, as funções $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$, satisfazem em x a Definição 14. A matriz Jacobiana de g é definida como,

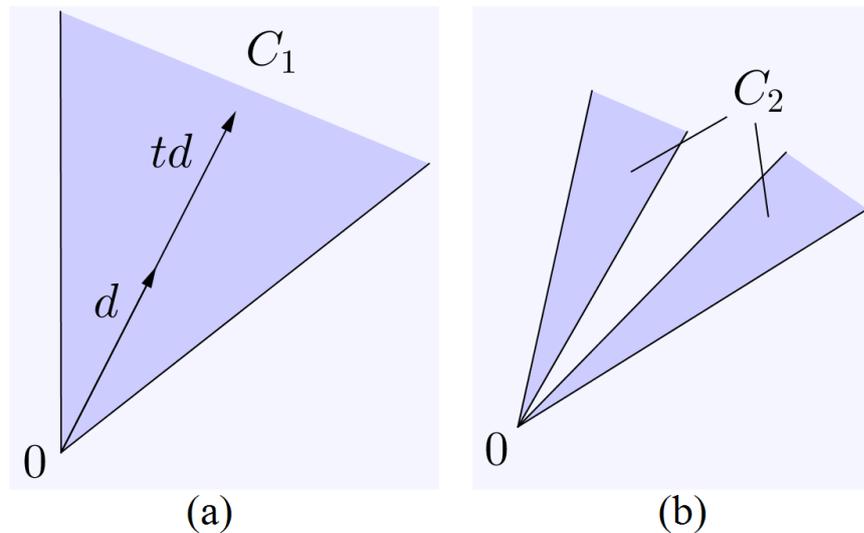
$$J_g = \begin{bmatrix} \frac{\partial g_1(x)}{\partial x_1} & \dots & \frac{\partial g_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_m(x)}{\partial x_1} & \dots & \frac{\partial g_m(x)}{\partial x_n} \end{bmatrix}. \quad (8)$$

Definição 18: Um conjunto $C \subset \mathbb{R}^n$ é chamado de cone se contém todos os múltiplos positivos de seus elementos, ou seja,

$$d \in C \Rightarrow td \in C \quad \forall t \in \mathbb{R}_+. \quad (9)$$

Na Fig. 3 temos dois exemplos de cones no \mathbb{R}^2 .

Figura 3 - Exemplos de cones.



Legenda: (a) Exemplifica um cone convexo e (b) um cone não convexo.

Fonte: A autora, 2022.

O conjunto das possíveis soluções dos problemas de autovalores e de otimização com restrições mistas, estudados no presente trabalho, é chamado de conjunto viável. A definição a seguir, apresenta um conjunto viável contido num hipercubo e definido por restrições de desigualdade.

Definição 19: Definimos o conjunto viável geral Ω , como

$$\Omega = \{x \in S \subset \mathbb{R}^n \mid g(x) \leq 0\}, \quad (10)$$

onde $g : S \rightarrow \mathbb{R}^m$. O conjunto S é o hipercubo dado por

$$S = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n], \quad (11)$$

com $[a_i, b_i] \subset \mathbb{R}$, $i = 1, \dots, n$. Note que se $m = 0$, então $\Omega = S$.

Definição 20: O vetor $d \in \mathbb{R}^n$, $d \neq 0$, é uma direção de descida de $f : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$ em $x^* \in \Omega$, quando existe $t_\delta > 0$ tal que $f(x^* + td) < f(x^*)$ para todo $t \in (0, t_\delta]$. Denota-se por $\Omega_f(x^*)$ o conjunto de todas as direções de descida de f em x^* .

O teorema a seguir, cuja demonstração pode ser encontrada em Ismailov e Solodov (2009), estabelece as condições necessária e suficiente para uma direção $d \in \mathbb{R}^n$ ser de descida para f em $x^* \in \Omega$.

Teorema 1: Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função diferenciável em $x^* \in \mathbb{R}^n$ e d uma direção no \mathbb{R}^n com $d \neq 0$. Então:

$$(a) \ d \in \Omega_f(x^*) \Rightarrow \langle \nabla f(x^*), d \rangle \leq 0.$$

$$(b) \ \text{Se } d \in \mathbb{R}^n \text{ satisfaz } \langle \nabla f(x^*), d \rangle < 0, \text{ então } d \in \Omega_f(x^*).$$

Na Definição 20, vimos que se d é uma direção de descida de f em $x^* \in \Omega$, então $f(x^* + td) < f(x^*)$. Entretanto, $x^* + td$ pode não pertencer a Ω . Por este motivo, é necessário estabelecer o conceito de direções viáveis, cuja definição é dada a seguir.

Definição 21: O vetor $d \in \mathbb{R}^n$ é uma direção viável em relação ao conjunto Ω no ponto $x^* \in \Omega$, se existe $t_\delta > 0$ tal que para todo $t \in [0, t_\delta]$

$$x^* + td \in \Omega. \tag{12}$$

O conjunto das direções viáveis em relação a Ω no ponto $x^* \in \Omega$, denotado por $V_\Omega(x^*)$, é chamado de cone das direções viáveis.

2 INICIALIZAÇÃO GLOBAL TOPOGRÁFICA MODIFICADA

Suponhamos que para calcular numericamente as soluções de um determinado problema, utilizaremos um método de busca local. Tal método realizará, a partir de um ponto inicial adequado, uma pesquisa no conjunto viável Ω , descrito na Eq. (10). Ao fim de um número finito de iterações, espera-se obter um ponto em Ω que satisfaça as equações que descrevem o problema, ou ainda, os critérios que definem as soluções. Note que partimos da premissa que o ponto inicial é conhecido. Entretanto, na prática isso nem sempre acontece, pois tais pontos nem sempre são fornecidos na formulação dos problemas. Nesse contexto, a Inicialização Global Topográfica é um método eficiente para selecionar pontos iniciais adequados em Ω . Isso é feito avaliando o valor de uma determinada função ϕ em um conjunto de pontos amostrais gerados uniformemente.

2.1 Descrição do método

O primeiro passo na Inicialização Global Topográfica consiste em gerar, de modo uniforme, \bar{N} pontos no hipercubo S . Aqui, esse processo é feito usando uma sequência com baixa discrepância (GENTLE, 2003), conhecida como sequência de Sobol, ver Sobol (1967). Observe que os pontos amostrais podem não ser todos viáveis, ou seja, apenas N pontos amostrais pertencem a $\Omega \subset \mathbb{R}^n$, $N \leq \bar{N}$. Os N pontos amostrais viáveis são denotados por P_i , $i = 1, 2, \dots, N$. Para cada ponto P_i , uma lista de índices de pontos é construída. Isso é feito listando os outros $(N - 1)$ pontos de acordo com a menor distância. Assim, para P_i , o j -ésimo elemento na lista é o j -ésimo ponto de amostra mais próximo.

Essa lista é ainda complementada pela atribuição de um sinal em cada índice j , da seguinte forma:

$$j = \begin{cases} +j, & \text{se } \phi(P_j) \geq \phi(P_i) \\ -j, & \text{se } \phi(P_j) < \phi(P_i) \end{cases}, \quad (13)$$

onde a função $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ é definida de forma que seja possível quantificar o quanto um ponto amostral deve ser priorizado no processo de escolha das estimativas iniciais. No caso do problema de otimização, ϕ é a própria função a ser minimizada.

As N listas de referência constituem uma matriz em $\mathbb{R}^{N \times (N-1)}$ chamada matriz topográfica de ϕ , denotada por t -matriz. Tal matriz pode ser representada por um grafo direcionado. Se $+j$ é um índice positivo da linha i , então P_j é a extremidade final do arco cujo ponto de partida é P_i (TÖRN e VIITANEN, 1992).

Dado um inteiro k , com $1 \leq k \leq (N - 1)$, se considerarmos apenas os primeiros k vizinhos mais próximos, obtemos uma submatriz da $t - matriz$. Essa submatriz é denotada como $k - t - matriz$ e o grafo correspondente é chamado de $k^+ - topografo$.

Definição 22: O ponto amostral P_i é dito minimizador local de ϕ no $k^+ - topografo$, se a i -ésima linha da $k - t - matriz$ é uma linha positiva, ou seja, todos os elementos da i -ésima linha são positivos.

Definição 23: Dizemos que um ponto amostral P_i tem uma referência positiva na $k - t - matriz$, se existir $j \neq i$ tal que: (a) A j -ésima linha da $k - t - matriz$ é uma linha positiva, e (b) O número $+i$ é um elemento desta j -ésima linha da $k - t - matriz$.

Definição 24: O ponto amostral P_i é dito minimizador global de ϕ no $k^+ - topografo$, se P_i é um minimizador local de ϕ no $k^+ - topografo$ e, além disso, P_i não tem referência positiva na $k - t - matriz$.

A seleção dos pontos iniciais pela Inicialização Global Topográfica pode seguir dois modelos distintos. No primeiro modelo somente os minimizadores globais de ϕ no $k^+ - topografo$ são selecionados para a busca local. Por sua vez, o segundo modelo estabelece que todos os minimizadores locais de ϕ no $k^+ - topografo$ devem ser selecionados como estimativas iniciais.

Um código computacional baseado no primeiro modelo é resumido em cinco passos: (i) são gerados N pontos amostrais viáveis (ii) constrói-se a $k - t - matriz$, (iii) verifica-se a positividade das linhas dessa matriz, (iv) seleciona-se os pontos amostrais P_i tais que a i -ésima linha da $k - t - matriz$ é uma linha positiva e (v) verifica-se todas as possíveis referências positivas na $k - t - matriz$.

Por outro lado, um código computacional baseado no segundo modelo, realiza somente as tarefas dadas nos itens (i), (ii), (iii) e (iv). Este fato faz com que o segundo código seja mais fácil de programar, o que o torna mais atrativo. Uma vez obtidos os minimizadores locais de ϕ no $k^+ - topografo$, tais pontos serão utilizados como pontos iniciais para o método de busca local.

Para ilustrar os passos da Inicialização Global Topográfica, consideremos o seguinte problema: obter $x \in \Omega$ que minimiza a função

$$\phi(x) = 10(x_1 - 2)^2 + 0.1(x_2^2 - 1)^2 + \cos^2(\pi x_2) + 100|x_1 x_2^2 - 2|, \quad (14)$$

onde $\Omega = \{x \in [-2, 4] \times [-2, 4]; g_1(x) = x_1 + x_2 + x_2^2 - 5 \leq 0 \text{ e } h_1(x) = x_1 x_2^2 - 2 \leq 0\}$.

Inicialmente, foram gerados $\bar{N} = 22$ pontos amostrais na caixa $[-2, 4] \times [-2, 4]$ utilizando a sequência de Sobol. Em seguida, verificando a viabilidade desses pontos, observou-se que 10 deles não são viáveis, a saber: $(-0,5; 2,5)$, $(3,25; 3,25)$, $(1,75; -1,25)$, $(2,125; 2,875)$, $(3,625; -1,625)$, $(2,875; 2,125)$, $(3,8125; 1,5625)$, $(1,5625; 3,8125)$, $(0,0625; 2,3125)$ e $(-1,625; 3,625)$.

Os $N = 12$ pontos restantes que pertencem a Ω são apresentados na Tabela 1 com os respectivos valores da função objetivo ϕ .

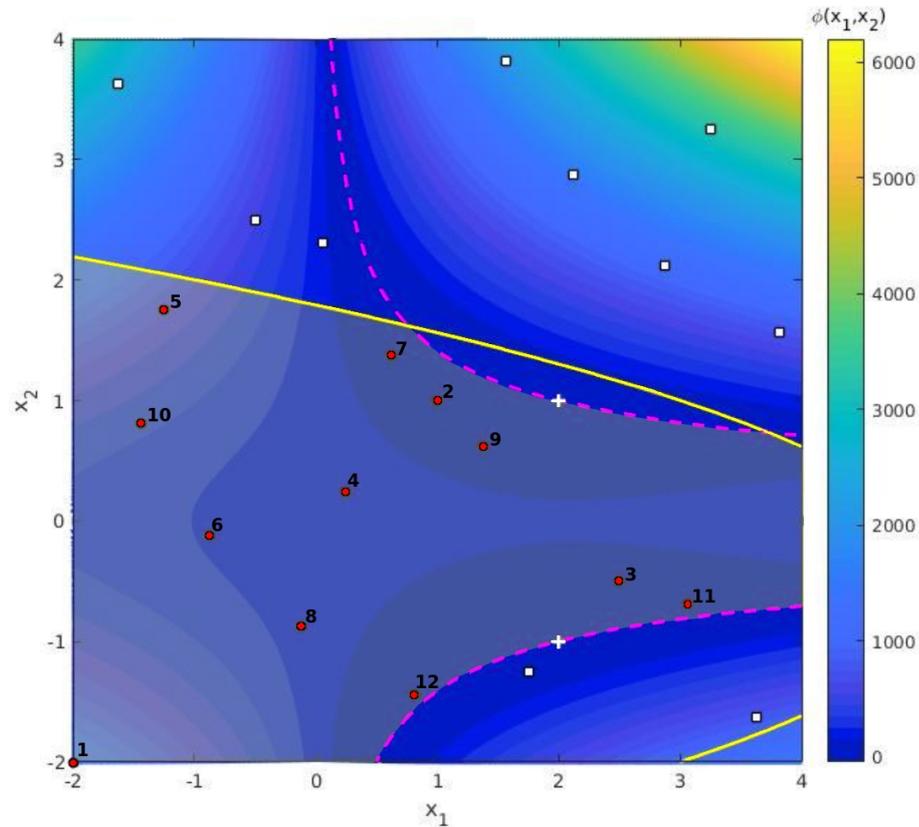
Tabela 1 - Doze pontos amostrais viáveis gerados pela sequência de Sobol, com os respectivos valores da função objetivo ϕ .

i	$P_i = (x_1, x_2)$	$\phi(x_1, x_2)$
1	(-2; -2)	1161,9
2	(1; 1)	111
3	(2,5; -0,5)	140,056
4	(0,25; 0,25)	229,65
5	(-1,25; 1,75)	689,363
6	(-0,875; -0,125)	284,974
7	(0,625; 1,375)	100,968
8	(-0,125; -0,875)	255,586
9	(1,375; 0,625)	150,379
10	(-1,4375; 0,8125)	413,764
11	(3,0625; -0,6875)	66,8746
12	(0,8125; -1,4375)	46,3578

Fonte: A autora, 2022.

Na Fig. 4, são exibidas as curvas de níveis da função objetivo na caixa $S = [-2, 4] \times [-2, 4]$. Também é apresentada a localização dos pontos amostrais em Ω . Olhando para as curvas de níveis, pode-se notar duas bacias de mínimos da função ϕ , onde se encontram as soluções globais do problema. Observe também que existem alguns pontos amostrais viáveis ($P_2, P_3, P_7, P_9, P_{11}$ e P_{12}) localizados nessas bacias.

Figura 4 - Curvas de níveis da função objetivo ϕ na caixa S .



Legenda: A região acinzentada indica a região viável do problema. A curva amarela é formada pelos pontos que satisfazem a igualdade $g_1(x) = 0$. A curva tracejada rosa é formada pelos pontos que satisfazem a igualdade $h_1(x) = 0$. Os pontos viáveis, dados na Tabela 1, são representados pelos círculos vermelhos. Os quadrados brancos indicam os pontos amostrais não viáveis. A escala de cores mostra a magnitude de ϕ . As cruces brancas representam as soluções globais do problema, a saber $(2, 1)$ e $(2, -1)$.

Fonte: A autora, 2022.

O próximo passo é obter a matriz topográfica. Para este exemplo, a t -matriz tem dimensão 12×11 , onde cada linha corresponde a um dos doze pontos amostrais viáveis.

$$t\text{-matriz} = \begin{bmatrix} -8 & -6 & -10 & -12 & -4 & -5 & -2 & -7 & -9 & -3 & -11 \\ -7 & 9 & 4 & 3 & 8 & 6 & 5 & 10 & -12 & -11 & 1 \\ -11 & 9 & -12 & -2 & 4 & -7 & 8 & 6 & 10 & 5 & 1 \\ -2 & -7 & 6 & -9 & 8 & -12 & 10 & 5 & -3 & -11 & 1 \\ -10 & -6 & -7 & -4 & -2 & -9 & -8 & -12 & 1 & -3 & -11 \\ -8 & 10 & -4 & 5 & -7 & -12 & -2 & 1 & -9 & -3 & -11 \\ 2 & 9 & 4 & 5 & 6 & 10 & 8 & 3 & -12 & -11 & 1 \\ 6 & -12 & -4 & -9 & 10 & 1 & -2 & -7 & -3 & 5 & -11 \\ -2 & -7 & 4 & -3 & 8 & -12 & -11 & 6 & 10 & 5 & 1 \\ 5 & -6 & -4 & -8 & -7 & -2 & -9 & 1 & -12 & -3 & -11 \\ 3 & 9 & -12 & 2 & 4 & 7 & 8 & 6 & 10 & 5 & 1 \\ 8 & 4 & 3 & 9 & 6 & 11 & 2 & 7 & 1 & 10 & 5 \end{bmatrix}. \quad (15)$$

Note que na $t\text{-matriz}$, os elementos de cada linha são ordenados de acordo com a ordem dos vizinhos mais próximos e recebem o sinal de acordo com o valor de ϕ . Por exemplo, na linha 2 nós temos que a distância entre os pontos P_2 e P_7 é menor ou igual que a distância entre os pontos P_2 e P_9 . Além disso, o elemento 7 tem sinal negativo, pois $\phi(P_7) < \phi(P_2)$.

Analisando apenas $k = 2$ vizinhos mais próximos, obtemos a submatriz $2\text{-}t\text{-matriz}$, formada pelas 2 primeiras colunas da $t\text{-matriz}$.

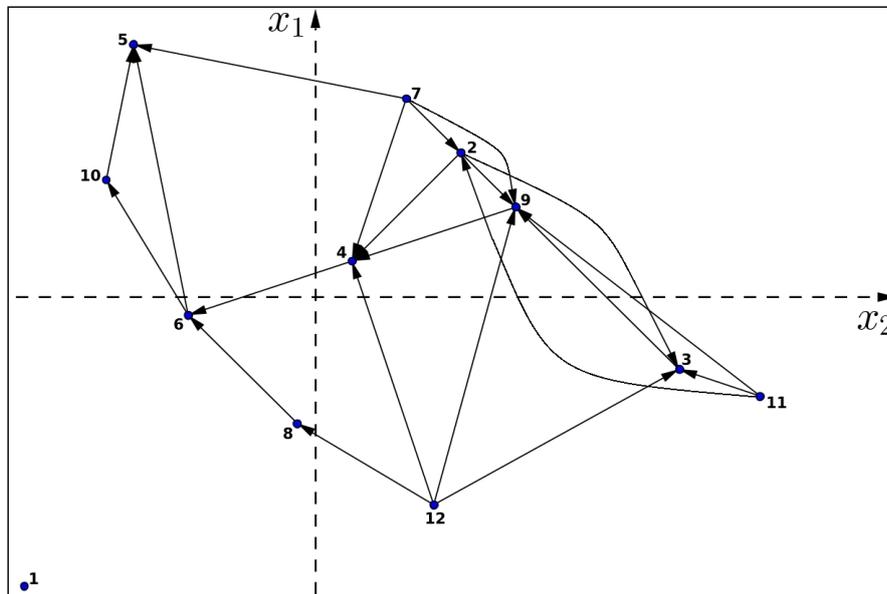
$$2\text{-}t\text{-matriz} = \begin{array}{l} 1 \rightarrow \\ 2 \rightarrow \\ 3 \rightarrow \\ 4 \rightarrow \\ 5 \rightarrow \\ 6 \rightarrow \\ 7 \rightarrow \\ 8 \rightarrow \\ 9 \rightarrow \\ 10 \rightarrow \\ 11 \rightarrow \\ 12 \rightarrow \end{array} \begin{bmatrix} -8 & -6 \\ -7 & 9 \\ -11 & 9 \\ -2 & -7 \\ -10 & -6 \\ -8 & 10 \\ 2 & 9 \\ 6 & -12 \\ -2 & -7 \\ 5 & -6 \\ 3 & 9 \\ 8 & 4 \end{bmatrix}. \quad (16)$$

Por outro lado, se $k = 4$ obtemos a submatriz $4\text{-}t\text{-matriz}$ dada por

$$4-t\text{-matriz} = \begin{array}{l} 1 \rightarrow \\ 2 \rightarrow \\ 3 \rightarrow \\ 4 \rightarrow \\ 5 \rightarrow \\ 6 \rightarrow \\ 7 \rightarrow \\ 8 \rightarrow \\ 9 \rightarrow \\ 10 \rightarrow \\ 11 \rightarrow \\ 12 \rightarrow \end{array} \left[\begin{array}{cccc} -8 & -6 & -10 & -12 \\ -7 & 9 & 4 & 3 \\ -11 & 9 & -12 & -2 \\ -2 & -7 & 6 & -9 \\ -10 & -6 & -7 & -4 \\ -8 & 10 & -4 & 5 \\ 2 & 9 & 4 & 5 \\ 6 & -12 & -4 & -9 \\ -2 & -7 & 4 & -3 \\ 5 & -6 & -4 & -8 \\ 3 & 9 & -12 & 2 \\ 8 & 4 & 3 & 9 \end{array} \right]. \quad (17)$$

Geometricamente, a $4-t$ -matriz pode ser representada por um grafo orientado, chamado de 4^+ -topografo conforme a Fig. 5 a seguir.

Figura 5 - O 4^+ -topografo de ϕ .



Legenda: Os pontos de amostra que apresentam os menores valores de $\phi(x_1, x_2)$ entre os 4 primeiros vizinhos na $4-t$ -matriz, são aqueles que enviam 4 arcos. Neste caso, P_7 e P_{12} . Esses pontos são os minimizadores locais de ϕ neste grafo.
Fonte: A autora, 2022.

Note que as linhas 7 e 12 são as únicas linhas positivas da $4-t$ -matriz. Conseqüentemente, os pontos P_7 e P_{12} , são os minimizadores locais de ϕ no 4^+ -topografo.

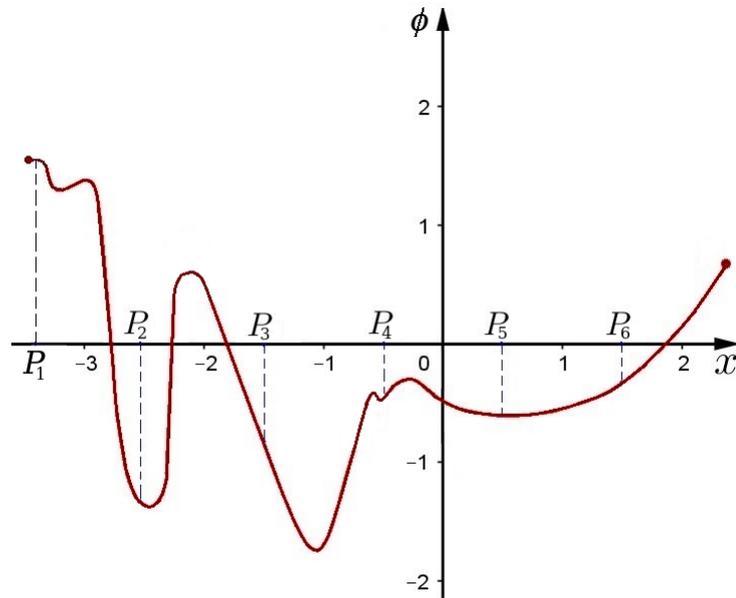
Como ilustrado na Fig. 5, os nós correspondentes a esses pontos, são os únicos pontos dos quais partem 4 arcos. Além disso, não há referência positiva para esses pontos na $4-t$ matriz. Logo, tais pontos são também minimizadores globais de ϕ no 4^+ topografo.

No presente trabalho, utilizaremos o segundo modelo para seleção das estimativas iniciais, ou seja, todos os minimizadores locais de ϕ no k^+ topografo serão selecionados. Entretanto, ao utilizar o modelo 2 deve-se analisar o custo computacional associado, pois o número de minimizadores locais de ϕ no k^+ topografo é maior ou igual ao número de minimizadores globais neste grafo (HENDERSON et al., 2015). Dessa forma, tal modelo pode levar a um significativo custo computacional no passo de busca local, sem contribuir de forma significativa.

Para superar este problema, basta que o parâmetro k escolhido seja suficientemente grande, reduzindo assim o número de minimizadores locais de ϕ . De fato, no exemplo anterior, quando $k = 2$, existem 3 estimativas iniciais (P_7, P_{11} e P_{12}) e quando $k = 4$ existem apenas duas (P_7 e P_{12}).

Portanto, a escolha de k constitui um passo importante para o uso da Inicialização Global Topográfica. Por esse motivo, em Henderson et al. (2015) os autores determinaram uma fórmula para estimar o valor de k a fim de reduzir o número de estimativas iniciais sem descartar aquelas que levarão às soluções do problema da Eq. (14). Na maioria dos testes realizados em Henderson et al. (2015), foram obtidos os valores $k = 4$ e $k = 5$. Por simplicidade, no presente trabalho utilizaremos estes valores como referência.

Ao utilizar a Inicialização Global Topográfica para refinar as estimativas iniciais, pode ocorrer de algumas amostras serem excluídas devido ao fato de um dos seus vizinhos mais próximos possuir o valor de ϕ “excessivamente” menor. Esse fato é ilustrado na Fig. 6, onde temos o gráfico de uma função $\phi : \mathbb{R} \rightarrow \mathbb{R}$, e seis pontos amostrais no seu domínio, a saber: P_1, P_2, P_3, P_4, P_5 e P_6 .

Figura 6 - Gráfico da função real ϕ .

Legenda: Os pontos P_1, P_2, P_3, P_4, P_5 e P_6 são os pontos amostrais. O minimizador global encontra-se entre os pontos P_3 e P_4 .

Fonte: A autora, 2022.

Note que existem cinco bacias de mínimos. Além disso, temos $\phi(P_2) < \phi(P_3) < \phi(P_5) < \phi(P_4) < \phi(P_6) < \phi(P_1)$ e a $4-t$ -matriz é dada por

$$4-t\text{-matriz} = \begin{matrix} 1 \rightarrow \\ 2 \rightarrow \\ 3 \rightarrow \\ 4 \rightarrow \\ 5 \rightarrow \\ 6 \rightarrow \end{matrix} \begin{bmatrix} -2 & -3 & -4 & -5 \\ 1 & 3 & 4 & 5 \\ -2 & 4 & 1 & 5 \\ -3 & -5 & -2 & 6 \\ 4 & 6 & -3 & -2 \\ -5 & -4 & -3 & -2 \end{bmatrix}. \quad (18)$$

Assim, se avaliarmos os 4 vizinhos mais próximos, o único minimizador local de ϕ no 4^+ -topografo será o ponto P_2 , que não se encontra na bacia do minimizador global da função ϕ . Dessa forma, selecionando o ponto P_2 como estimativa inicial para o método de busca local, será obtida uma solução local, mas não global. Por outro lado, o ponto P_3 que se encontra na bacia do minimizador global será descartado pois ϕ apresenta um valor “excessivamente” menor em P_2 . Para contornar esse problema, no presente trabalho propomos a Inicialização Global Topográfica Modificada (MTGO).

Seja N o número de pontos amostrais viáveis obtidos. Se $N > k + 1$, então o MTGO verifica quais linhas da $k-t$ -matriz apresentam ao menos $k-1$ valores positivos. Os pontos associados a essas linhas são chamados de semiminimizadores locais de ϕ no

k^+ - *topografo* e devem ser utilizados como estimativas iniciais para o método de busca local. Caso contrário, se $N = k + 1$, então o MTGO selecionará como estimativas iniciais os minimizadores locais de ϕ no k^+ - *topografo*.

No exemplo ilustrado na Fig. 6, temos $N > k + 1$ e existem dois semiminimizadores locais de ϕ no 4^+ - *topografo*, a saber P_2 e P_3 . Logo, tais pontos serão selecionados pelo MTGO.

Teorema 2: Dada uma quantidade \bar{N} de pontos amostrais, sejam n_M e n_S os números de minimizadores e semiminimizadores locais de ϕ no k^+ - *topografo*, respectivamente. Então, para essa quantidade \bar{N} de pontos, temos $n_M \leq n_S$.

Demonstração: Suponhamos $n_M > n_S$. Logo, existe pelo menos um minimizador local de ϕ no k^+ - *topografo* que não é semiminimizador local de ϕ no k^+ - *topografo*, o que contradiz a definição de semiminimizador local. Portanto, $n_M \leq n_S$. ■

Como consequência do Teorema 2, temos que o MTGO permite ainda o uso de um número menor ou igual de pontos amostrais. Tal fato, é fundamental para a solução de problemas restritos, cujas restrições sejam muito rígidas, especialmente problemas com muitas variáveis e um grande número de restrições de desigualdades. No exemplo descrito na Eq. (14) foram utilizados inicialmente $\bar{N} = 22$ pontos amostrais. Entretanto, utilizando o MTGO esse número pode ser reduzido para 10 pontos amostrais, resultando em 6 pontos viáveis listados na Tabela 2.

Tabela 2 - Seis pontos amostrais viáveis gerados pela sequência de Sobol, com os respectivos valores da função objetivo ϕ .

i	$P_i = (x_1, x_2)$	$\phi(x_1, x_2)$
1	(-2; -2)	1161,9
2	(1; 1)	111
3	(2,5; -0,5)	140,056
4	(0,25; 0,25)	229,65
5	(-1,25; 1,75)	689,363
6	(-0,875; -0,125)	284,974

Fonte: A autora, 2022.

Analisando a $4 - t$ - matriz descrita a seguir, vemos que os pontos P_2 e P_3 são os semiminimizadores locais de ϕ no 4^+ - *topografo* associado aos 6 pontos viáveis.

$$4 - t - \text{matriz} = \begin{matrix} 1 \rightarrow \\ 2 \rightarrow \\ 3 \rightarrow \\ 4 \rightarrow \\ 5 \rightarrow \\ 6 \rightarrow \end{matrix} \begin{bmatrix} -6 & -4 & -5 & -2 \\ 4 & 3 & 6 & 5 \\ -2 & 4 & 6 & 5 \\ -2 & 6 & 5 & -3 \\ -6 & -4 & -2 & 1 \\ -4 & 5 & -2 & 1 \end{bmatrix}. \quad (19)$$

De acordo com Henderson et al. (2015), a uniformidade da distribuição dos pontos amostrais no interior do conjunto viável $\Omega \subset \mathbb{R}^n$, é um aspecto importante para o sucesso do TGO e, portanto, do MTGO. Descrevemos, a seguir, o gerador de amostras uniformes usado neste trabalho, a chamada sequência de Sobol.

2.2 Gerador de Amostras Uniformes

Seja S_n o hipercubo unitário dado por $S_n = [0, 1] \times [0, 1] \times \dots \times [0, 1]$. Uma vez que é sempre possível obter um hipercubo $S \subset \mathbb{R}^n$ a partir do hipercubo unitário, então podemos discutir o processo usado para gerar η pontos em S_n .

A medida do desvio da uniformidade de uma dada sequência de pontos, distribuídos dentro de um conjunto de pontos, é chamada de discrepância. Aqui, é suficiente considerar a discrepância de $B = \{P_1, P_2, \dots, P_\eta\} \subset S_n$, onde o j -ésimo componente do ponto $P_i \in B$ será denotado por $x_{i,j}$, de forma que, $P_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n})^T$, ver Joe e Kuo (2003). A discrepância $D_\eta(B)$ de $B \subset S_n$ é definida como

$$D_\eta(B) = \text{Sup}_V |F_\eta(V) - M(V)|, \quad (20)$$

onde $V \subset S_n$ é o subcubo dado por $V = [0, q_1] \times [0, q_2] \times \dots \times [0, q_n]$, com $q_j \in [0, 1]$, $j = 1, \dots, n$. O termo $F_\eta(V)$ é o número de pontos contidos em $B \cap V$ dividido por η . Por sua vez, $M(V)$ é o hipervolume de V , veja Gentle (2003). A sequência B é considerada de baixa discrepância, se para η suficientemente grande, $D_\eta(B)$ torna-se pequeno. Observe que a magnitude $|F_\eta(V) - M(V)|$ representa o tamanho relativo da fração de pontos de S contidos em V .

Para estudar a sequência de Sobol, precisamos ver inicialmente o conceito de polinômio primitivo. Seja $\mathbb{Z}_2 = \{\bar{0}, \bar{1}\}$ o conjunto (anel) de todas as classes de congruência módulo 2, onde $\bar{a} + \bar{b} = \overline{a+b}$ e $\bar{a} \cdot \bar{b} = \overline{a \cdot b}$, com $\bar{a}, \bar{b} \in \mathbb{Z}_2$, veja Shokranian et al. (1999) e Garcia e Lequain (2012). O conjunto de polinômios com coeficientes em \mathbb{Z}_2 , é denotado por $\mathbb{Z}_2[x]$. Um polinômio $p(z) \in \mathbb{Z}_2[z]$, de grau η , é dito um polinômio irredutível sobre

\mathbb{Z}_2 se a decomposição $p(z) = \alpha(z)\beta(z)$, com $\alpha(z)$ e $\beta(z)$ em $\mathbb{Z}_2[z]$, ocorre se, e somente se $\alpha(z)$ ou $\beta(z)$ tem grau η . Neste caso, $\alpha(z)$ ou $\beta(z)$ tem grau zero. Além disso, se o menor inteiro positivo n para o qual $p(z)$ divide $z^n + 1$, é dado por $n = 2^n - 1$, então $p(z)$ é dito ser um polinômio primitivo.

A sequência de Sobol é uma sequência com baixa discrepância, veja Gentle (2003), em que o componente $x_{i,j}$ do ponto P_i na sequência de Sobol, é dado pela seguinte expressão:

$$x_{i,j} = \beta_1 v_{1,j} \oplus \beta_2 v_{2,j} \oplus \beta_3 v_{3,j} \oplus \dots, \quad (21)$$

onde β_ℓ é o ℓ -ésimo bit da direita para a esquerda, quando i é escrito em binário, ou seja, $i = (\dots\beta_3\beta_2\beta_1)_2$.

Os números $v_{k,j}$, chamados de "números de direção", são definidos por

$$v_{k,j} = \frac{m_{k,j}}{2^k}, \quad (22)$$

onde os valores $m_{k,j}$ são inteiros positivos ímpares menores que 2^k . Tais valores são determinados através da relação de recorrência

$$m_{k,j} = 2\alpha_{1,j}m_{k-1,j} \oplus 2^2\alpha_{2,j}m_{k-2,j} \oplus \dots \oplus 2^{s_j-1}\alpha_{s_j-1,j}m_{k-s_j+1,j} \oplus 2^{s_j}m_{k-s_j,j} \oplus m_{k-s_j,j}, \quad (23)$$

com $k \geq s_j + 1$. Os valores iniciais $m_{1,j}, m_{2,j}, \dots, m_{s_j,j}$ podem ser escolhidos livremente de modo que cada $m_{k,j}$ satisfaça $0 < m_{k,j} < 2^k$ para todo $k = 1, \dots, s_j$. Os valores subsequentes $m_{s_j+1,j}, m_{s_j+2,j}, \dots$, são determinados pela fórmula dada na Eq. (23).

O símbolo \oplus denota o operador bit a bit, tal que $1 \oplus 0 = 0 \oplus 1 = 1$ e $1 \oplus 1 = 0 \oplus 0 = 0$. Por sua vez, as constantes $\alpha_{1,j}, \dots, \alpha_{s_j-1,j}$ são os coeficientes do polinômio primitivo $p_j(z) \in \mathbb{Z}_2[z]$, dado por

$$p_j(z) = z^{s_j} + \alpha_{1,j}z^{s_j-1} + \dots + \alpha_{s_j-1,j}z + 1, \quad (24)$$

onde s_j é o grau de $p_j(z)$. Existem várias tabelas destes polinômios disponíveis na literatura, veja por exemplo Press et al. (1992).

Um código Gray é uma função $G(i)$ definida nos inteiros não negativos, de modo que a representação binária de $G(i)$ e $G(i+1)$ se diferem em exatamente um bit (GENTLE, 2003). Usando um código Gray particular, Antonov e Saleev (1979) mostraram que a sequência de Sobol na Eq.(21) pode ser escrita na seguinte forma equivalente

$$x_{i,j} = g_1 v_{1,j} \oplus g_2 v_{2,j} \oplus g_3 v_{3,j} \oplus \dots, \quad (25)$$

onde $(\dots g_3 g_2 g_1)_2$ é a representação binária do código Gray usado por Antonov e Saleev

(1979) avaliada em i , dada por

$$G(i) = (\dots g_3 g_2 g_1)_2 = \dots \beta_3 \beta_2 \beta_1 \oplus \dots \beta_4 \beta_3 \beta_2. \quad (26)$$

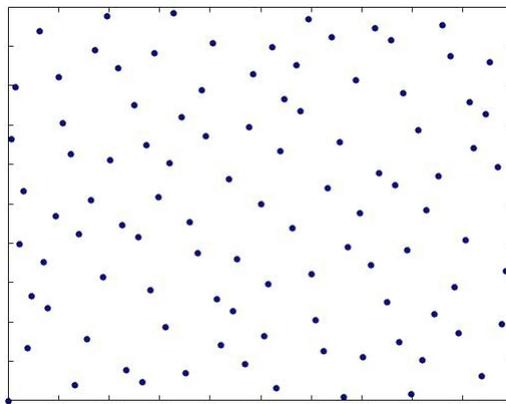
Das Eqs. (25) e (26), a sequência de Sobol mostrada na Eq. (21) pode ser gerada de forma recursiva

$$x_{i,j} = x_{i-1,j} \oplus v_{r,j}, \quad (27)$$

onde $x_{0,j} = 0$. Na Eq. (27), o número r é tal que β_r é o bit zero mais a direita na representação binária de $i-1$. De acordo com Galanti e Jung (1997), a fórmula modificada de Antonov e Sallev (1979), Eq.(27), leva a uma implementação cerca de 20% mais rápida do que a fórmula original de Sobol, dada na Eq. (21).

Algumas implementações da sequência de Sobol estão disponíveis na literatura, ver Bratley e Fox (1988) e Press et al. (1992). No presente trabalho, vamos utilizar uma modificação do código conhecido como Algoritmo 659 (JOE e KUO, 2003). A Fig. 7 mostra os pontos gerados dentro de um quadrado unitário usando o gerador Sobol, onde foram gerados 100 pontos.

Figura 7 - Pontos gerados a partir da sequência Sobol.

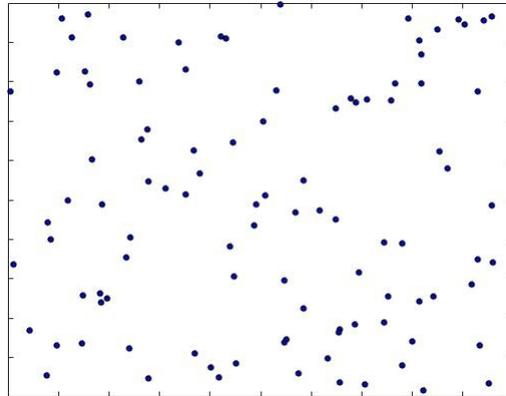


Legenda: Os círculos indicam as posições dos pontos gerados. Os espaços vazios na caixa são pequenos devido a uniformidade da distribuição.

Fonte: A autora, 2022.

A Fig. 8 mostra os resultados obtidos realizando o mesmo procedimento, mas agora usando o gerador de números aleatórios "Mersenne Twister", proposto por Matsumoto e Nishimura (1998). A sequência Sobol fornece os pontos que estão melhor distribuídos na caixa.

Figura 8 - Pontos gerados a partir do gerador Mersenne Twister.



Legenda: Os círculos indicam as posições dos pontos gerados. Os espaços vazios na caixa são grandes devido a distribuição não uniforme.

Fonte: A autora, 2022.

2.3 Algoritmo MTGO/Busca Local

No Algoritmo 1 constam os principais passos da Inicialização Global Topográfica Modificada, que visa determinar pontos amostrais adequados no conjunto limitado $\Omega \subset \mathbb{R}^n$, dado pela Eq. (10), com $\Omega^0 \neq \emptyset$.

Algoritmo 1- MTGO/Busca Local

Passo 1. Utilizando a sequência de Sobol, gerar \bar{N} pontos amostrais no hipercubo S .

Passo 2. Escolher N ($\leq \bar{N}$) pontos viáveis, dentre os \bar{N} pontos gerados no Passo 1.

Passo 3. Usando esses N pontos, e o parâmetro k escolhido, obter o conjunto de todos os s semiminimizadores locais de ϕ no k^+ - *topográfico*.

Passo 4. Utilizar cada semiminimizador $x^{0,i}$ como estimativa inicial para o método de busca local. Obtendo um $x^{*,i} \in \Omega$ para cada $x^{0,i}$, $i = 1, \dots, s$.

Passo 5. Dentre todos $x^{*,i} \in \Omega$, obtidos no Passo 4, selecionar aqueles que solucionam o problema.

3 MÉTODOS NUMÉRICOS PARA O CÁLCULO DE AUTOVALORES

O cálculo dos autovalores de uma matriz é um problema que possui destaque na Álgebra linear, sendo a sua resolução necessária em problemas nas mais diversas áreas da Física, da Química, da Biologia, da Engenharia e da Economia (ANTON e RORRES, 2012). Na engenharia, por exemplo, os autovalores podem ser aplicados na resolução de sistemas de equações diferenciais que descrevem a flexão de uma viga, ou ainda, as vibrações de um edifício (LEON, 2011). Na Termodinâmica, mais precisamente no estudo dos estados críticos de misturas termodinâmicas, os autovalores são utilizados para caracterizar os pontos multicríticos de uma mistura multicomponente, ver Henderson et al. (2013). Na teoria dos números, os autovalores podem ser utilizados para obter uma fórmula fechada para a sequência de Fibonacci, que por sua vez pode ser empregada no estudo do crescimento de plantas (POOLE, 2011).

Assim, em vista do amplo campo de aplicações dos autovalores, faz-se necessário o estudo adequado de tal problema. Em geral, para solucionar o problema de autovalor, recorre-se aos métodos numéricos, cujas características dependem da abordagem utilizada. Nesse contexto, vários trabalhos já foram desenvolvidos com o objetivo de elaborar e estudar métodos numéricos eficientes, ver Francis (1961), Wilkinson (1965), Parlett (1980), Saad (1992), Gander (2008), Watkins (2010) e Burden; Faires e Burden (2015).

No presente trabalho, utilizaremos o MTGO para obter estimativas iniciais adequadas para um método de busca local, baseado no método de Newton, que será empregado no cálculo de autovalores de matrizes reais simétricas.

3.1 Descrição do problema de autovalores

Seja $A \in \mathbb{C}^{n \times n}$ a matriz, na base canônica, do operador linear complexo $\tilde{A} : \mathbb{C}^n \rightarrow \mathbb{C}^n$, onde \mathbb{C}^n é o espaço das listas $v = (z_1, \dots, z_n)$ de n números complexos (LIMA, 2014). Um vetor $v \neq 0$ em \mathbb{C}^n chama-se um autovetor da matriz A quando existe $\lambda \in \mathbb{C}$ tal que

$$Av = \lambda v. \tag{28}$$

O número $\lambda \in \mathbb{C}$, por sua vez, chama-se autovalor da matriz A quando existe um vetor não nulo $v \in \mathbb{C}^n$ tal que $Av = \lambda v$. Diz-se então que o autovalor λ corresponde ao autovetor v e, vice-versa, que o autovetor v também corresponde ao autovalor λ (LIMA, 2014).

Pela Eq. (28) temos que

$$Av - \lambda v = (A - \lambda I_n) v = 0, \tag{29}$$

onde $I_n \in \mathbb{C}^{n \times n}$ é a matriz identidade de ordem n . Assim, o número complexo λ é um autovalor de A se, e somente se, existe $v \neq 0$ em \mathbb{C}^n tal que a Eq. (29) é satisfeita. De acordo com Hoffman e Kunze (1971), $(A - \lambda I_n)v = 0$ se, e somente se, $\det(A - \lambda I_n) = 0$, onde $\det(A - \lambda I_n)$ é um polinômio de grau n em λ . Esse polinômio, indicado por $p_A(\lambda)$ é chamado de polinômio característico da matriz A e as raízes complexas da equação $p_A(\lambda) = 0$ são chamadas de raízes características e coincidem com os autovalores de A . O conjunto dos autovalores de A , denotado por $sp(A)$, é chamado de espectro de A .

Lembre-se que existe uma relação bijetiva entre os números complexos $z = a + bi \in \mathbb{C}$ e os pares ordenados (a, b) . Assim, o número complexo $z \in sp(A)$ pode ser representado geometricamente por um ponto no plano de Argand-Gauss, ver Evans (2006). Esse fato nos permite definir os chamados círculos de Gerschgorin (GERSCHGORIN, 1931). Tais círculos, denotados por R_i , possuem centro a_{ii} e raio r_i , e são dados por

$$R_i = \{z \in \mathbb{C}; \|z - a_{ii}\| \leq r_i\} \quad (30)$$

onde

$$r_i = \sum_{\substack{j=1 \\ j \neq i}}^n \|a_{ij}\|, i = 1, \dots, n. \quad (31)$$

Aqui, $\|\cdot\|$ é a norma do número complexo.

O teorema a seguir nos permite restringir a localização dos autovalores de uma matriz num círculo contido no plano de Argand-Gauss (GERSCHGORIN, 1931).

Teorema 3: Dada a matriz $A \in \mathbb{C}^{n \times n}$, o conjunto $sp(A)$ está contido na reunião $R_1 \cup \dots \cup R_n$ dos círculos R_i descritos na Eq. (30).

Demonstração: Ver Wilkinson (1965).

Corolário 1: Seja r o valor máximo das somas $r_i + \|a_{ii}\|$, $i = 1, \dots, n$. Então, o conjunto $sp(A)$ está contido no círculo R de raio r e centro na origem do plano Argand-Gauss.

Demonstração: Pela definição de R temos que $R_i \subset R$, $i = 1, \dots, n$. Portanto, $R_1 \cup \dots \cup R_n \subset R$. Logo, pelo Teorema 3 temos que $sp(A) \subset R$. ■

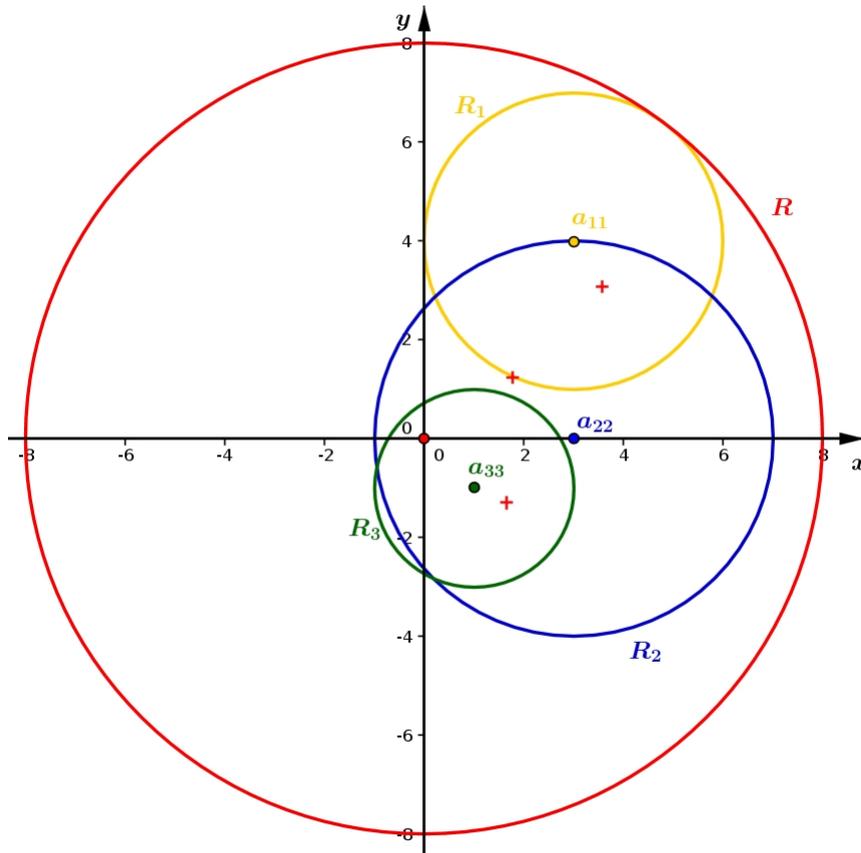
Consideremos, por exemplo, a matriz A dada por

$$A = \begin{bmatrix} 3 + 4i & 1 & 2 \\ 2 & 3 & 2 \\ 1 & -1 & 1 - i \end{bmatrix}. \quad (32)$$

Os autovalores dessa matriz são $\lambda_1 = 3,57678 + 3,07323i$, $\lambda_2 = 1,76413 + 1,2304i$ e $\lambda_3 = 1,65909 - 1,30363i$. Pela Eq. (31) temos $r_1 = 3$, $r_2 = 4$ e $r_3 = 2$. Nesse caso, $r = r_1 + \|a_{11}\| = 8$.

Na Fig. 9, temos os círculos R_1 , R_2 e R_3 para a matriz A . Observa-se que $R_1 \cup R_2 \cup R_3 \subset R$, o que implica em $\lambda_1, \lambda_2, \lambda_3 \in R$. Tal resultado está em concordância com o Corolário 1.

Figura 9 - Círculos de Gerschgorin para a matriz A .



Legenda: O círculo R_1 (em amarelo) tem raio $r_1 = 3$ e centro $a_{11} = (3, 4)$. O círculo R_2 (em azul) tem raio $r_2 = 4$ e centro $a_{22} = (3, 0)$. O círculo R_3 (em verde) tem raio $r_3 = 2$ e centro $a_{33} = (1, -1)$. O círculo R (em vermelho) tem centro na origem e raio $r = 8$. Os autovalores λ_1, λ_2 e λ_3 são representados pelas cruces vermelhas.

Fonte: A autora, 2022.

Para solucionar o problema descrito na Eq. (28), existe uma ampla variedade de métodos, cujas características dependem da abordagem utilizada. O mais antigo desses métodos remonta ao trabalho de Jacobi em meados do século XIX, que utilizando uma sequência de rotações planas busca reduzir uma matriz simétrica real à forma diagonal. Entretanto, por se tratar de um problema essencialmente mais complicado do que resolver sistemas lineares, foi somente por volta de 1950 que, impulsionado pelo surgimento do computador eletrônico, houve um desenvolvimento altamente significativo de métodos numéricos para calcular os autovalores de matrizes (GOLUB e VORST, 2000).

De acordo com Golub e Vorst (2000), o método mais simples para obter auto-

valores e autovetores é o Método da Potência. Tal técnica iterativa busca determinar o autovalor dominante da matriz, ou seja, o autovalor com o maior módulo. Uma vez obtido o autovalor dominante, utiliza-se as técnicas de deflação para a obtenção das aproximações dos outros autovalores. Uma característica útil nesse método é que ele produz o autovalor e um autovetor associado. Por esse motivo, por vezes esse método é empregado para determinar um autovetor para um autovalor determinado por outro método. Entretanto, o Método da Potência tem a desvantagem de que não se sabe, no início, se a matriz tem ou não um autovalor dominante. Além disso, o processo de deflação é suscetível a erros de arredondamento (BURDEN; FAIRES; BURDEN, 2015).

Alguns dos métodos para o cálculo de autovalores, utilizam transformações matriciais, que permitem obter a cada iteração uma nova matriz semelhante a matriz original A , ou seja, a cada iteração é obtida uma nova matriz mais simples (vários elementos nulos) e que possui os mesmos autovalores de A . Após um número finito de iterações, obtém-se uma matriz cujos elementos da diagonal principal são aproximações dos autovalores da matriz A .

Dentre os métodos que utilizam essa abordagem, um dos métodos mais conhecidos é dado pelo Algoritmo QR (GOLUB e LOAN, 1996). De acordo Burden; Faires e Burden (2015), o primeiro passo ao aplicar o método QR , é transformar a matriz inicial A numa matriz semelhante na forma de Hessenberg superior. Particularmente, se a matriz A é simétrica, então a matriz transformada será tridiagonal. Feita tal transformação, o método QR decompõe a matriz $A = A_1$ no produto $A_1 = Q_1 R_1$, onde Q_1 é uma matriz ortogonal ($Q_1^T Q_1 = I_n$) e R_1 é uma matriz triangular superior. Em seguida, o processo de decomposição é aplicado à matriz $A_2 = R_1 Q_1$, resultando na matriz $A_2 = Q_2 R_2$. Continuando com esse processo, temos que a sequência $\{A_k\}$, com $A_k = R_{k-1} Q_{k-1} = Q_k R_k$, converge para uma matriz triangular superior cujos elementos da diagonal principal são os autovalores da matriz A (POOLE, 2011).

Outra abordagem para o problema (28), consiste em obter as raízes do polinômio característico $p_A(\lambda)$ associado a matriz A . Entretanto, os zeros do polinômio podem ser muito sensíveis a pequenas mudanças nos seus coeficientes, ou seja, o problema de determinar os autovalores de uma matriz A utilizando os coeficientes do polinômio característico $p_A(\lambda)$ é mal condicionado (WILKINSON, 1994).

Para contornar essa dificuldade, uma alternativa é calcular os zeros de $p_A(\lambda)$, sem representar explicitamente os coeficientes. Algoritmos desse tipo são estáveis, no entanto, muitas vezes necessitam de um grande esforço computacional, principalmente no caso de matrizes de maior dimensão, devido o cálculo de determinantes. Por outro lado, o avanço da tecnologia e o desenvolvimento de processadores modernos, permite o uso de tais métodos. Além disso, através de operações elementares é possível obter matrizes mais simples para o cálculo do determinante (ANTON e RORRES, 2012).

No presente trabalho, vamos calcular os autovalores de uma matriz simétrica

$A \in \mathbb{R}^{n \times n}$ utilizando o polinômio característico sem representar explicitamente os seus coeficientes. Nesse caso, as raízes características do polinômio $p_A(\lambda)$ são valores reais. Com efeito, suponhamos que λ seja um autovalor de A com autovetor correspondente

$$v = \begin{bmatrix} a_1 + b_1 i \\ a_2 + b_2 i \\ \vdots \\ a_n + b_n i \end{bmatrix} \neq 0. \quad (33)$$

Pela definição de autovalor temos,

$$A\bar{v} = \overline{A}v = \overline{Av} = \overline{\lambda v} = \bar{\lambda}\bar{v}, \quad (34)$$

onde \bar{v} é o conjugado de v e \overline{A} é a matriz conjugada de A , ou seja, $\overline{a_{ij}} = \bar{a}_{ij}$, com $i, j = 1, \dots, n$.

Além disso, como a matriz A é simétrica, obtemos:

$$\bar{v}^T A = \bar{v}^T A^T = (A\bar{v})^T = (\bar{\lambda}\bar{v})^T = \bar{\lambda}\bar{v}^T. \quad (35)$$

Logo,

$$\lambda(\bar{v}^T v) = \bar{v}^T(\lambda v) = \bar{v}^T(Av) = (A\bar{v})^T v = (\bar{v}^T A)v = (\bar{\lambda}\bar{v}^T)v = \bar{\lambda}(\bar{v}^T v), \quad (36)$$

ou seja,

$$(\lambda - \bar{\lambda})(\bar{v}^T v) = 0. \quad (37)$$

Como $\bar{v}^T v \neq 0$, temos $\lambda - \bar{\lambda} = 0 \Leftrightarrow \lambda = \bar{\lambda}$. Portanto, λ é um número real. Observe que nesse caso, o círculo R descrito no Corolário 1, é reduzido à um intervalo nos reais.

A seguir, passamos a descrever a metodologia aqui empregada para o cálculo dos autovalores de matrizes reais simétricas. Visando simplificar o cálculo do determinante, o primeiro passo é transformar a matriz A em uma matriz semelhante mais simples. Tal transformação será feita utilizando o método desenvolvido por Alston Householder (BURDEN; FAIRES e BURDEN, 2015).

3.2 Método de Householder

O método Householder aplicado a uma matriz A resulta em uma matriz \overline{A} Hessenberger superior semelhante à matriz A . No caso particular de A ser simétrica, a matriz

\bar{A} é tridiagonal. Note que se $A \in \mathbb{R}^{2 \times 2}$, então $A = \bar{A}$. Para melhor compreensão deste método precisamos definir o que é a transformação de Householder, também conhecida como Refletor de Householder (POOLE, 2011).

3.2.1 Transformação de Householder

A transformação de Householder $\tilde{H} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ faz corresponder a cada $x \in \mathbb{R}^n$ o vetor $Hx \in \mathbb{R}^n$, que é a reflexão de x em torno do hiperplano $\{w\}^\perp$ perpendicular a w , onde $w \in \mathbb{R}^n$ é tal que $\|w\|^2 = w^T w = 1$. A matriz $H \in \mathbb{R}^{n \times n}$ da transformação \tilde{H} , na base canônica do \mathbb{R}^n , é dada por

$$H = I_n - 2ww^T. \quad (38)$$

Logo, para $x \in \mathbb{R}^n$ temos

$$Hx = (I_n - 2ww^T)x = x - 2ww^T x = x - 2(w^T x)w, \quad (39)$$

onde $Px = (w^T x)w$ é a projeção ortogonal de x em w (GOLUB e LOAN, 1996). Note que $\|Hx\| = \|x\|$, ou seja, a transformação de Householder preserva a norma.

Suponhamos que $\bar{x} = x + \bar{e} \in \mathbb{R}^n$ é uma aproximação do vetor $x \in \mathbb{R}^n$. Se $\bar{e} \in \mathbb{R}^n$ é o erro cometido nessa aproximação, temos que $H\bar{x} = Hx + H\bar{e}$, onde o erro $H\bar{e}$ cometido na aproximação do vetor Hx por $H\bar{x}$, é tal que $\|H\bar{e}\| = \|\bar{e}\|$. Isso indica que a transformação de Householder é estável em relação ao aumento de tamanho do erro (HANSEN, 1990).

Observe que a matriz $H \in \mathbb{R}^{n \times n}$ é simétrica e ortogonal. De fato, pela definição de H temos

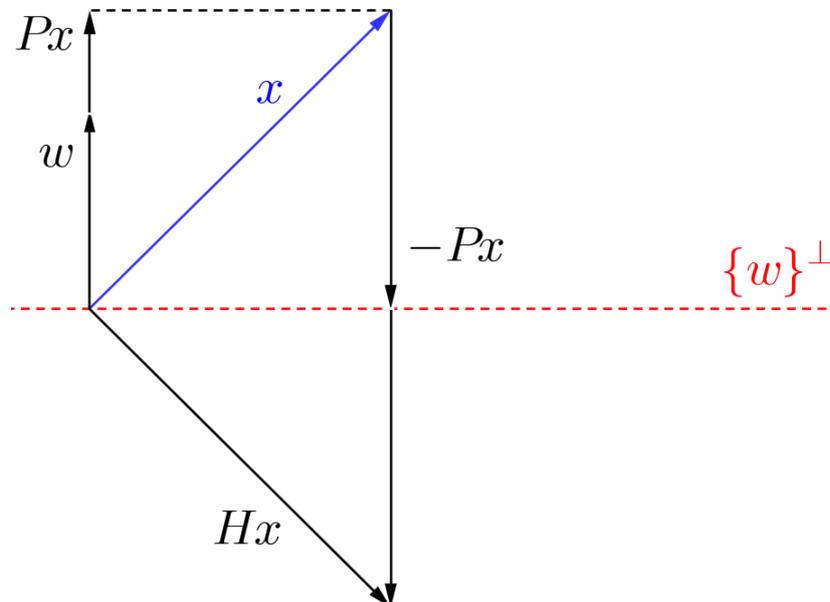
$$H^T = (I_n - 2ww^T)^T = I_n - 2(w^T)^T w^T = I - 2ww^T = H, \quad (40)$$

ou seja, H é simétrica. Por outro lado,

$$HH^T = HH = (I_n - 2ww^T)(I_n - 2ww^T) = I_n \Rightarrow H^{-1} = H^T, \quad (41)$$

isto é, H é ortogonal. Na Fig. 10 temos uma ilustração da transformação de Householder.

Figura 10 - Transformação de Householder



Legenda: Hx é a reflexão de x em torno do hiperplano $\{w\}^\perp$, representado pela linha tracejada vermelha. O vetor Px é a projeção ortogonal de x em w , onde $w^T w = 1$
 Fonte: A autora, 2022.

3.2.2 Algoritmo de Householder

No caso da matriz A ser simétrica, as transformações \tilde{H} podem ser utilizadas para zerar, de modo estável em relação ao erro de arredondamento, blocos de elementos nas colunas de A de tal modo que ao fim de $n - 2$ passos, obtém-se uma matriz simétrica tridiagonal semelhante a A (BURDEN; FAIRES e BURDEN, 2015).

De acordo com Burden; Faires e Burden (2015), o primeiro passo é determinar a matriz $H^{(1)} = I_n - 2w^{(1)}(w^{(1)})^T$, com $(w^{(1)})^T w^{(1)} = 1$, tal que $A^{(2)} = H^{(1)}AH^{(1)}$ é uma matriz cujos elementos $a_{11}^{(2)} = a_{11}$, $a_{j1}^{(2)} = 0$, $j = 3, \dots, n$. Note que $H^{(1)}$ é a matriz da transformação de Householder $\tilde{H}^{(1)}$. Portanto, por simetria temos $a_{1j}^{(2)} = 0$, $j = 3, \dots, n$.

Se $w_1^{(1)} = 0$, temos que $a_{11}^{(2)} = a_{11}$. Assim, a matriz $H^{(1)}$ deve satisfazer a igualdade

$$H^{(1)} \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ \vdots \\ a_{n1} \end{bmatrix} = \begin{bmatrix} a_{11} \\ \alpha \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \alpha \in \mathbb{R}. \quad (42)$$

Seja $\widehat{H} : \mathbb{R}^{n-1} \rightarrow \mathbb{R}^{n-1}$ a transformação de Householder, cuja matriz $\widehat{H} \in \mathbb{R}^{(n-1) \times (n-1)}$ é dada por $\widehat{H} = I_{n-1} - 2\widehat{w}\widehat{w}^T$, onde $\widehat{w} = [w_2^{(1)} \ w_3^{(1)} \ \dots \ w_n^{(1)}]^T \in \mathbb{R}^{n-1}$. Definindo $\widehat{y} = [a_{21} \ a_{31} \ \dots \ a_{n1}]^T \in \mathbb{R}^{n-1}$, a Eq. (42) pode ser reescrita como

$$H^{(1)} \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ \vdots \\ a_{n1} \end{bmatrix} = \begin{bmatrix} a_{11} \\ \dots \\ \widehat{H}\widehat{y} \\ \dots \end{bmatrix} = \begin{bmatrix} a_{11} \\ \dots \\ \alpha \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (43)$$

com

$$\widehat{H}\widehat{y} = (I_{n-1} - 2\widehat{w}\widehat{w}^T)\widehat{y} = \widehat{y} - 2(\widehat{w}^T\widehat{y})\widehat{w} = \begin{bmatrix} a_{21} - 2rw_2^{(1)} \\ a_{31} - 2rw_3^{(1)} \\ a_{41} - 2rw_4^{(1)} \\ \vdots \\ a_{n1} - 2rw_n^{(1)} \end{bmatrix} = \begin{bmatrix} \alpha \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (44)$$

onde $r = \widehat{w}^T\widehat{y}$. Portanto,

$$2rw_2^{(1)} = a_{21} - \alpha \quad (45)$$

e

$$2rw_j^{(1)} = a_{j1}, \text{ para cada } j = 3, 4, \dots, n. \quad (46)$$

Para obter os valores $w_j^{(1)}$, $j = 2, \dots, n$, utilizando as Eqs. (45) e (46) é necessário determinar r e α . Isso pode ser feito elevando ao quadrado ambos os lados das Eqs. (45) e (46) e somando membro a membro, de onde resulta

$$4r^2 \sum_{j=2}^n (w_j^{(1)})^2 = (a_{21} - \alpha)^2 + \sum_{j=3}^n a_{j1}^2 = \alpha^2 - 2\alpha a_{21} + \sum_{j=2}^n a_{j1}^2. \quad (47)$$

Note que $\sum_{j=2}^n (w_j^{(1)})^2 = 1$, pois $(w^{(1)})^T w^{(1)} = 1$ e $w_1^{(1)} = 0$. Além disso, da ortogonalidade de \widehat{H} temos

$$\alpha^2 = (\widehat{H}\widehat{y})^T (\widehat{H}\widehat{y}) = \widehat{y}^T \widehat{H}^T \widehat{H} \widehat{y} = \widehat{y}^T \widehat{y} = \sum_{j=2}^n a_{j1}^2. \quad (48)$$

Assim, podemos reescrever a Eq. (47) como

$$2r^2 = \sum_{j=2}^n a_{j1}^2 - \alpha a_{21}. \quad (49)$$

Observe que $2r^2 = \alpha(\alpha - a_{21})$. Logo, $2r^2 = 0$ se, e somente se, $\alpha = 0$ ou $\alpha - a_{21} = 0$.

De acordo com Burden; Faires e Burden (2015), para garantir que $2r^2 = 0$ somente se $a_{21} = a_{31} = \dots = a_{n1} = 0$, deve-se escolher

$$\alpha = -\text{sign}(a_{21}) \sqrt{\sum_{j=2}^n a_{j1}^2}. \quad (50)$$

Assim, se $a_{21} = a_{31} = \dots = a_{n1} = 0$, então $r = 0$ e $A^{(2)} = A$.

Em resumo, para encontrar a matriz $H^{(1)} = I_n - 2w^{(1)}(w^{(1)})^T$, deve-se obter α conforme a Eq. (50). O vetor $w^{(1)} \in \mathbb{R}^n$ é então definido por

$$w^{(1)} = \frac{1}{2r} \begin{bmatrix} 0 \\ a_{21} - \alpha \\ a_{31} \\ a_{41} \\ \vdots \\ a_{n1} \end{bmatrix}, \quad (51)$$

onde

$$r = \sqrt{\frac{1}{2}(\alpha^2 - \alpha a_{21})}. \quad (52)$$

De modo análogo, as matrizes $H^{(k)} = I_n - 2w^{(k)}(w^{(k)})^T$, $k = 2, 3, \dots, n-2$, podem ser obtidas definindo $w_1^{(k)} = \dots = w_k^{(k)} = 0$,

$$w_{k+1}^{(k)} = \frac{a_{k+1,k}^{(k)} - \alpha}{2r}, \quad (53)$$

e

$$w_j^{(k)} = \frac{a_{jk}^{(k)}}{2r}, \quad j = k+2, k+3, \dots, n, \quad (54)$$

onde $r = \sqrt{\frac{1}{2}(\alpha^2 - \alpha a_{k+1,k})}$ e

$$\alpha = -\text{sign}(a_{k+1,k}^{(k)}) \cdot \left(\sum_{j=k+1}^n (a_{jk}^{(k)})^2 \right)^{\frac{1}{2}}. \quad (55)$$

Dessa forma, obtém-se a matriz simétrica e tridiagonal $A^{(n-1)}$ dada por

$$A^{(n-1)} = H^{(n-2)} H^{(n-3)} \dots H^{(1)} A H^{(1)} \dots H^{(n-3)} H^{(n-2)}, \quad (56)$$

com $A^{(k+1)} = H^{(k)} A^{(k)} H^{(k)}$ (BURDEN; FAIRES e BURDEN, 2015).

O Algoritmo 2 apresentado a seguir, descreve os passos do método de Householder para matrizes simétricas.

Algoritmo 2- Método de Householder

Dados: $k = 1$, $A^{(k)} = A \in \mathbb{R}^{n \times n}$.

Enquanto $k \leq n - 2$, faça.

Passo 1. $q = \sum_{j=k+1}^n (a_{jk}^{(k)})^2$.

Passo 2. Se $a_{k+1,k}^{(k)} = 0$, então $\alpha = -\sqrt{q}$.

Caso contrário, faça

$$\alpha = -\frac{\sqrt{q} a_{k+1,k}^{(k)}}{|a_{k+1,k}^{(k)}|}.$$

Passo 3. Faça $\bar{r} = \alpha^2 - \alpha a_{k+1,k}^{(k)}$, onde $r = \sqrt{\bar{r}/2}$

Passo 4. Faça $v_j = 0$, $j = 1, \dots, k$; $v_{k+1} = a_{k+1,k}^{(k)} - \alpha$ e $v_j = a_{jk}^{(k)}$, $j = k + 2, \dots, n$.

Passo 5. Para $j = k, k + 1, \dots, n$ faça

$$u_j = \left(\frac{1}{\bar{r}} \right) \sum_{i=k+1}^n a_{ji}^{(k)} v_i.$$

Passo 6. Faça $\varphi = \sum_{i=k+1}^n v_i u_i$.

Passo 7. Para $j = k, k + 1, \dots, n$ faça $z_j = u_j - \left(\frac{\varphi}{2\bar{r}} \right) v_j$.

Passo 8. Para $l = k + 1, k + 2, \dots, n - 1$ execute os Passos 9 e 10.

Passo 9. Para $j = l + 1, \dots, n$ faça $a_{jl}^{(k+1)} = a_{jl}^{(k)} - v_l z_j - v_j z_l$ e $a_{lj}^{(k+1)} = a_{jl}^{(k+1)}$.

Passo 10. Faça $a_{ll}^{(k+1)} = a_{ll}^{(k)} - 2v_l z_l$.

Passo 11. Faça $a_{nn}^{(k+1)} = a_{nn}^{(k)} - 2v_n z_n$.

Passo 12. Para $j = k + 2, \dots, n$ faça $a_{kj}^{(k+1)} = a_{jk}^{(k+1)} = 0$.

Passo 13. Faça $a_{k+1,k}^{(k+1)} = a_{k+1,k}^{(k)} - v_{k+1} z_k$ e $a_{k,k+1}^{(k+1)} = a_{k+1,k}^{(k+1)}$.

Fim

Retornar a matriz $A^{(n-1)}$ simétrica, tridiagonal e semelhante a A .

Note que pelo passo 4 do Algoritmo 2, temos $w = \left(\frac{1}{\sqrt{2r}}\right)v = \frac{1}{2r}v$. Logo,

$$u = \left(\frac{1}{r}\right)A^{(k)}v = \frac{1}{2r^2}A^{(k)}v = \frac{1}{r}A^{(k)}v. \quad (57)$$

Assim, utilizando a Eq. (57), vê-se que nos Passos 6 e 7, temos

$$\varphi = v^T u = \frac{1}{2r^2}v^T A^{(k)}v \quad (58)$$

e

$$z = u - \frac{1}{4r^2}v^T u v = u - ww^T u = \frac{1}{r}A^{(k)}w - ww^T \frac{1}{r}A^{(k)}w. \quad (59)$$

Portanto, a matriz $A^{(k+1)}$ dos Passos 9-13 pode ser escrita como

$$A^{(k+1)} = A^{(k)} - vz^T - zv^T = \left(I_n - 2ww^T\right)A^{(k)}\left(I_n - 2ww^T\right). \quad (60)$$

De acordo com Burden; Faires e Burden (2015), para aplicar o Algoritmo 2 a uma matriz $A \in \mathbb{R}^{n \times n}$ arbitrária, as seguintes modificações devem ser feitas:

i) No Passo 5, para faça

$$u_j = \left(\frac{1}{r}\right) \sum_{i=k+1}^n a_{ji}^{(k)} v_i \text{ e } y_j = \left(\frac{1}{r}\right) \sum_{i=k+1}^n a_{ij}^{(k)} v_i$$

ii) No Passo 7, para $j = 1, 2, \dots, n$ faça $z_j = u_j - \left(\frac{\varphi}{r}\right)v_j$.

iii) No Passo 8, para $l = k + 1, k + 2, \dots, n$ execute os Passos 9 e 10.

iv) No Passo 9, para $j = 1, 2, \dots, k$ faça $a_{jl}^{(k+1)} = a_{jl}^{(k)} - v_l z_j$ e $a_{lj}^{(k+1)} = a_{lj}^{(k)} - v_l y_j$.

v) No Passo 10, para $j = k + 1, \dots, n$ faça $a_{jl}^{(k+1)} = a_{jl}^{(k)} - v_l z_j - v_j y_l$.

Obtida a matriz \bar{A} , o próximo passo é calcular os autovalores. Para isso, vamos utilizar um método de busca local baseado no método de Newton para solução de sistemas de equações não lineares. Por esse motivo, antes de descrever a metodologia aqui utilizada, vamos revisar o método de Newton.

3.3 Método de Newton

O método de Newton clássico é introduzido para o problema de se determinar um $y \in \Omega \subseteq \mathbb{R}^n$, tal que

$$F(y) = 0, \quad (61)$$

onde $F : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ é uma aplicação diferenciável no \mathbb{R}^n , ver Ismailov e Solodov (2007) e Luenberger (1984).

Seja $y^k \in \Omega$, uma aproximação de alguma solução y^* da Eq. (61). Em torno de y^k podemos aproximar F pela sua linearização, obtendo

$$F(y^k) + J_F(y^k)(y^{k+1} - y^k) = 0. \quad (62)$$

A Eq. (62) é chamada de iteração do método de Newton. Definindo $\bar{d}_k = y^{k+1} - y^k$, temos $y^{k+1} = y^k + \bar{d}_k$, onde o vetor \bar{d}_k é chamado de direção de Newton. O ponto y^{k+1} é obtido tomando um passo unitário na direção de Newton, partindo de y^k .

A seguir, é descrito o algoritmo do método de Newton.

Algoritmo 3- Método de Newton

Dados: $y^0 \in \Omega$, $k = 0$.

Passo 1. Calcular $y^{k+1} \in \Omega$ utilizando a fórmula descrita na Eq. (62).

Passo 2. Se o critério de parada foi satisfeito. Pare.

Passo 3. Caso contrário faça $k = k + 1$ e retorne ao Passo 1.

Suponhamos que $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ é diferenciável em uma vizinhança de $y^* \in \Omega$, com derivada contínua neste ponto e $\det(J_F(y^*)) \neq 0$. Então, para qualquer ponto inicial $y^0 \in \Omega$ suficientemente próximo a y^* , o método de Newton descrito no Algoritmo 3, gera uma sequência $\{y^k\}$ bem definida que converge para y^* (ISMAILOV e SOLODOV, 2007).

Definindo $F : R \subset \mathbb{R} \rightarrow \mathbb{R}$ como $F(\lambda) = p_{\bar{A}}(\lambda)$, podemos utilizar o método de Newton para calcular as raízes características do polinômio $p_{\bar{A}}(\lambda)$, ou seja, para calcular os valores $\lambda^* \in R \subset \mathbb{R}$ tais que $p_{\bar{A}}(\lambda) = 0$, onde $R \subset \mathbb{R}$ é o intervalo definido conforme o Corolário 1.

Nesse caso, a iteração de Newton é dada por

$$p_{\bar{A}}(\lambda^k) + p'_{\bar{A}}(\lambda^k)(\lambda^{k+1} - \lambda^k) = 0, \quad k = 0, 1, \dots, \quad (63)$$

ou ainda,

$$\lambda^{k+1} = \lambda^k - \frac{p_{\bar{A}}(\lambda^k)}{p'_{\bar{A}}(\lambda^k)}, \quad k = 0, 1, \dots, \quad (64)$$

Pelo que vimos anteriormente, se $\lambda^0 \in R$ é uma estimativa inicial suficientemente próxima a λ^* , a sequência $\{\lambda^k\}$ gerada pelo método de Newton converge para λ^* . Note que se $p_{\bar{A}}(\lambda)$ possui n raízes características distintas, serão necessárias no mínimo, n estimativas iniciais, pois ao aplicar sucessivamente o método de Newton com essas n estimativas, pode-se obter raízes já calculadas, ou seja, o método pode não convergir para algumas raízes, enquanto outras podem ser obtidas mais de uma vez. Nesse contexto, o Método de Newton-Maehly, que será apresentado a seguir, busca contornar essa dificuldade.

3.4 Método de Newton-Maehly

Suponhamos que já foram computados m autovalores $\lambda_1, \lambda_2, \dots, \lambda_m$. Nesse caso, as raízes restantes de $p_{\bar{A}}(\lambda)$ correspondem às raízes do polinômio deflacionado de grau $n - m$, dado por

$$p_{n-m}(\lambda) = \frac{p_{\bar{A}}(\lambda)}{(\lambda - \lambda_1)(\lambda - \lambda_2) \dots (\lambda - \lambda_m)}. \quad (65)$$

Assim, para evitar a avaliação dos zeros já calculados, podemos aplicar a iteração de Newton ao polinômio $p_{n-m}(\lambda)$ (GANDER, 2008), ou seja,

$$\lambda^{k+1} = \lambda^k - \frac{p_{n-m}(\lambda^k)}{p'_{n-m}(\lambda^k)}. \quad (66)$$

A derivada de $p_{n-m}(\lambda)$ em relação a λ é dada por

$$p'_{n-m}(\lambda) = \frac{p'_{\bar{A}}(\lambda)}{(\lambda - \lambda_1) \dots (\lambda - \lambda_m)} - \frac{p_{\bar{A}}(\lambda)}{[(\lambda - \lambda_1) \dots (\lambda - \lambda_m)]^2} \cdot r(\lambda), \quad (67)$$

onde $r : \mathbb{R} \rightarrow \mathbb{R}$ é dada por

$$r(\lambda) = \sum_{j=1}^m \left[\prod_{i=1, i \neq j}^m (\lambda - \lambda_i) \right]. \quad (68)$$

Assim, a Eq. (67) pode ser reescrita como

$$p'_{n-m}(\lambda) = \frac{p'_{\bar{A}}(\lambda)}{(\lambda - \lambda_1) \dots (\lambda - \lambda_m)} - \frac{p_{\bar{A}}(\lambda)}{(\lambda - \lambda_1) \dots (\lambda - \lambda_m)} \sum_{j=1}^m \frac{1}{(\lambda - \lambda_j)}. \quad (69)$$

Pelas Eqs. (65), (66) e (69), temos:

$$\lambda^{k+1} = \lambda^k - \frac{p_{\bar{A}}(\lambda^k)}{\left[p'_{\bar{A}}(\lambda^k) - p_{\bar{A}}(\lambda^k) \cdot \sum_{j=1}^m \frac{1}{(\lambda^k - \lambda_j)} \right]}, \quad (70)$$

ou ainda,

$$\lambda^{k+1} = \lambda^k - \frac{p_{\bar{A}}(\lambda^k)}{p'_{\bar{A}}(\lambda^k)} \cdot \frac{1}{\left[1 - \frac{p_{\bar{A}}(\lambda^k)}{p'_{\bar{A}}(\lambda^k)} \cdot \sum_{j=1}^m \frac{1}{(\lambda^k - \lambda_j)} \right]}. \quad (71)$$

A Eq. (71) é chamada de iteração de Newton-Maehly (STOER e BULIRSCH, 1993), que encontra uma nova raiz λ_{m+1} suprimindo as m raízes já computadas. Esse método pode ser aplicado na determinação dos zeros de uma função diferenciável $f : \mathbb{R} \rightarrow \mathbb{R}$ qualquer.

Uma grande vantagem do método Newton-Maehly é a convergência (local) quadrática para λ_{m+1} . Além disso, o cálculo de λ_{m+1} não é sensível aos erros incorridos no cálculo das raízes anteriores (STOER e BULIRSCH, 1993).

Note que, na Eq. (71), deve-se obter o valor do polinômio característico e da sua derivada em um determinado ponto λ^k . Entretanto, conforme foi dito anteriormente, não vamos calcular de forma explícita os coeficientes de $p_{\bar{A}}$. Em vez disso, vamos avaliar de forma implícita o valor de $p_{\bar{A}}$ em λ^k , utilizando para isso a chamada fatoração LU (GANDER, 2008).

3.5 Fatoração LU

Considere a transformação linear $\tilde{M}_k : \mathbb{R}^n \rightarrow \mathbb{R}^n$ cuja matriz $M_k \in \mathbb{R}^{n \times n}$, na base canônica, é dada por

$$M_k = I_n - \tau e_k^T = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\tau_{k+1} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\tau_n & 0 & \cdots & 1 \end{bmatrix}, \quad (72)$$

onde $e_k \in \mathbb{R}^n$ é o k -ésimo vetor da base canônica do \mathbb{R}^n e $\tau \in \mathbb{R}^n$ é um vetor cujos primeiros k componentes são zeros e $\tau_i = x_i/x_k$, $i = k+1, \dots, n$ (GOLUB e LOAN,

1996). Observe que a matriz M_k definida na Eq. (72), é invertível com $M_k = M_k^{-1}$. Além disso, para $x \in \mathbb{R}^n$ temos

$$M_k x = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\tau_{k+1} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\tau_n & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (73)$$

ou seja, a transformação \tilde{M}_k aplicada no vetor x tem por função zerar os componentes x_i , $i = k + 1, \dots, n$.

Por praticidade, vamos adotar as seguintes notações: $C(a : b, q : w)$ indica a matriz cujos elementos são c_{ij} , $i = a, \dots, b$ e $j = q, \dots, w$, com $1 \leq a \leq b \leq n$ e $1 \leq q \leq w \leq n$. Se $a = b$, temos a matriz linha $C(a, q : w)$. Por sua vez, $\tau(r : s)$, denota o vetor cujos componentes são os elementos τ_i , $i = r, \dots, s$, com $1 \leq r \leq s \leq n$.

Assim, dada uma matriz $C \in \mathbb{R}^{n \times n}$, o produto matricial $M_k C$ é tal que

$$M_k C = (I_n - \tau e_k^T) C = I_n C - \tau (e_k^T C) = C - \tau C(k, 1 : n). \quad (74)$$

Note que no produto $M_k C$, somente $C(k + 1 : n, 1 : n)$ é modificada. Portanto, a atualização $C = M_k C$ pode ser feita linha por linha utilizando a fórmula

$$C(i, 1 : n) = C(i, 1 : n) - \tau C(k, 1 : n), \quad i = k + 1, \dots, n. \quad (75)$$

Suponhamos que existam matrizes M_1, \dots, M_{n-1} tais que $M_{n-1} M_{n-2} \dots M_1 C = U$, é triangular superior. A matriz U pode ser obtida utilizando o seguinte algoritmo.

Algoritmo 4- Eliminação de Gauss - Cálculo da matriz U .

Dados: $k = 1$, $C^k = C \in \mathbb{R}^{n \times n}$.

Enquanto $(C^k(k, k) \neq 0)$ e $(k \leq n - 1)$, faça

$$\begin{aligned} C^{k+1}(k, 1 : n) &= C^k(k, 1 : n) \\ \tau(k + 1 : n) &= C^k(k + 1 : n, k) / C^k(k, k) \\ C^{k+1}(k + 1 : n, 1 : n) &= C^k(k + 1 : n, 1 : n) - \tau(k + 1 : n) C^k(k, 1 : n) \\ k &= k + 1 \end{aligned}$$

Fim

Faça $U = C^n$.

Observe que no Algoritmo 4, as matrizes M_k são dadas por

$$\begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\frac{C^k(k+1,k)}{C^k(k,k)} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\frac{C^k(n,k)}{C^k(k,k)} & 0 & \cdots & 1 \end{bmatrix}. \quad (76)$$

Nesse caso, faz-se necessário que os elementos $C^k(k, k)$, chamados de pivôs, sejam verificados para evitar uma divisão por zero.

Definindo $L = M_1^{-1}M_2^{-1} \dots M_{n-1}^{-1}$, temos que

$$C = LU, \quad (77)$$

onde L é uma matriz triangular inferior cujos elementos da diagonal principal são iguais a 1

$$L = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ l_{21} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix}. \quad (78)$$

A decomposição descrita na Eq. (77) é chamada de fatoração LU , ver Golub e Loan (1996). Em resumo, dada uma matriz $C \in \mathbb{R}^{n \times n}$, dizemos que as matrizes L e U formam uma decomposição LU para a matriz C se:

- i) L e U são matrizes quadradas e $C = LU$.
- ii) U é uma matriz triangular superior.
- iii) L é uma matriz triangular inferior com os elementos da diagonal principal iguais a 1.

O teorema a seguir estabelece uma condição suficiente (mas não necessária) para a existência e unicidade da decomposição LU .

Teorema 4: Seja $C_k = C(1:k, 1:k)$, $k = 1, \dots, n-1$. A matriz $C \in \mathbb{R}^{n \times n}$ possui fatoração LU se $\det(C_k) \neq 0$. Se além disso ocorrer $\det(C) \neq 0$, então tal fatoração é única e $\det(C) = u_{11} \cdots u_{nn}$.

Demonstração: Ver Golub e Loan (1996).

Note que a exigência $C^k(k, k) \neq 0$ na Eq. (76), nem sempre pode ser satisfeita, violando assim as condições do Teorema 4. Além disso, o cálculo dos valores $\tau(k+1:n)$ na k -ésima iteração do Algoritmo 4, pode apresentar instabilidade numérica se o valor $C^k(k, k)$ estiver muito próximo de zero (RUGGIERO e LOPES, 2000). Para contornar essas dificuldades, deve-se adotar uma estratégia de pivoteamento, que consiste num processo para a escolha adequada do pivô em cada iteração.

Aqui, utilizaremos a estratégia do pivoteamento parcial que consiste em:

- i) A cada iteração, escolher para pivô o elemento $C^k(i, k)$ de maior módulo, onde $i = k, k+1, \dots, n$.
- ii) Se $i \neq k$, então deve-se permutar as linhas k e i da matriz C^k .

A priori, não se sabe quais são (nem mesmo se serão necessárias) as transposições de linhas durante o processo de Eliminação de Gauss de uma matriz C . Entretanto, depois de efetuado o processo, dispomos da relação de todas as transposições feitas. Efetuando, na mesma ordem em que foram feitas, todas essas transposições nas linhas da matriz identidade, obtemos uma matriz $M \in \mathbb{R}^{n \times n}$, chamada de matriz de permutação. O produto MC corresponde a efetuar sobre a matriz C , antecipadamente, todas as transposições de linhas que seriam necessárias durante o escalonamento (LIMA, 2014). Portanto a matriz MC pode ser escrita na forma LU , ou seja, $MC = LU$.

Utilizando a Eliminação de Gauss com pivoteamento parcial para decompor a matriz $C = \bar{A} - \lambda I_n$, obtém-se

$$M(\lambda)C(\lambda) = L(\lambda)U(\lambda), \quad (79)$$

onde $M(\lambda)$ é uma matriz de permutação, $L(\lambda)$ é uma matriz triangular inferior, e $U(\lambda)$ é uma matriz triangular superior (GANDER, 2008). Pelas propriedades de determinantes temos,

$$\det M(\lambda) \cdot \det C(\lambda) = \det L(\lambda) \cdot \det U(\lambda) = u_{11} \cdot u_{22} \cdot \dots \cdot u_{nn}, \quad (80)$$

com $\det L(\lambda) = 1$ e $\det M(\lambda) = \pm 1$. Portanto, o polinômio $p_{\bar{A}}(\lambda)$ é dado por

$$p_{\bar{A}}(\lambda) = \det C(\lambda) = \pm 1 \prod_{j=1}^n u_{jj}(\lambda). \quad (81)$$

O Algoritmo 5, descrito a seguir, detalha o cálculo de $p_{\bar{A}}(\lambda)$ utilizando a Eliminação de Gauss com pivoteamento parcial.

Algoritmo 5- Cálculo de $p_{\bar{A}}(\lambda)$.

Dados: $k = 1$, $C^k = C \in \mathbb{R}^{n \times n}$, $\epsilon > 0$, $f = 1$.

Enquanto $k \leq n - 1$

Passo 1. Faça $c_{max} = \text{maior dos valores } |C^k(k : n, k)|$ e i_{max} a linha correspondente ao maior valor.

Se $c_{max} = 0$, faça $p_{\bar{A}}(\lambda) = f = 0$. FIM

Passo 2. Se $i_{max} \neq k$ então

Faça a transposição das linhas i_{max} e k da matriz C^k .

Defina $f = -f$.

Passo 3. Calcule $f = f.C^k(k, k)$.

Passo 4.

$$\tau(k + 1 : n) = C^k(k + 1 : n, k) / C^k(k, k)$$

$$C^{k+1}(k + 1 : n, k + 1 : n) = C^k(k + 1 : n, k + 1 : n) - \tau(k + 1 : n) C^k(k, k + 1 : n)$$

Passo 5. $k = k + 1$

Fim

Faça $p_{\bar{A}}(\lambda) = f = f.C^n(n, n)$.

No método Newton-Maehly, apresentado na subseção 3.4, faz-se necessário o cálculo não somente de $p_{\bar{A}}(\lambda)$ mas também de sua derivada $p'_{\bar{A}}(\lambda)$. Para essa tarefa, no presente trabalho utilizamos a Diferenciação Algorítmica, também chamada de Diferenciação Automática que calcula a derivada de uma função a partir de uma série de operações elementares (GANDER, 2008).

3.6 Diferenciação Algorítmica

A derivada de $p_{\bar{A}}(\lambda)$ pode ser calculada utilizando a fórmula de Jacobi

$$p'_{\bar{A}}(\lambda) = \det(C(\lambda)) \operatorname{tr}(C^{-1}(\lambda) C'(\lambda)). \quad (82)$$

Entretanto, o uso dessa fórmula exige o cálculo da matriz inversa de $C(\lambda)$ a cada avaliação de $p'_{\bar{A}}(\lambda)$, o que incorre em um expressivo custo computacional (GANDER, 2008).

Por outro lado, utilizando a Eq. (81), temos que a derivada de $p_{\bar{A}}(\lambda)$ é dada por

$$p'_{\bar{A}}(\lambda) = \pm 1 [u'_{11}(\lambda).u_{22}(\lambda).\dots.u_{nn}(\lambda) + \dots + u_{11}(\lambda).u'_{22}(\lambda).\dots.u'_{nn}(\lambda)], \quad (83)$$

ou ainda,

$$p'_{\bar{A}}(\lambda) = \pm 1 \sum_{i=1}^n u'_{ii}(\lambda) \left[\prod_{j \neq i}^n u_{jj}(\lambda) \right] = \pm 1 \sum_{i=1}^n \frac{u'_{ii}(\lambda)}{u_{ii}(\lambda)} \prod_{j=1}^n u_{jj}(\lambda). \quad (84)$$

Logo,

$$p'_A(\lambda) = \pm 1 \sum_{i=1}^n \frac{u'_{ii}(\lambda)}{u_{ii}(\lambda)} p_A(\lambda). \quad (85)$$

Nesse caso, para determinar $p'_A(\lambda)$ utilizando a Eq. (85), faz-se necessário calcular as derivadas u'_{ii} dos elementos da diagonal de $U(\lambda)$. Entretanto, obter uma fórmula explícita para essa expressão não é possível.

Visando contornar as dificuldades apresentadas para o cálculo de $p'_A(\lambda)$, vamos recorrer ao procedimento de Diferenciação Algorítmica (GANDER, 2008), que aplica as regras de derivação (produto e quociente) em cada passo do Algoritmo 5. Tal procedimento é descrito no Algoritmo 6.

Algoritmo 6- Cálculo de $p_A(\lambda)$ e $p'_A(\lambda)$ - Diferenciação Algorítmica.

Dados: $k = 1$, $C^k = C \in \mathbb{R}^{n \times n}$, $Cs^k = C'(\lambda) = -I_n \in \mathbb{R}^{n \times n}$, $f = 1$, $fs = 0$.

Enquanto $k \leq n - 1$

Passo 1. Faça $c_{max} =$ maior dos valores $|C^k(k : n, k)|$ e i_{max} a linha correspondente ao maior valor.

Se $c_{max} = 0$, faça $p_A(\lambda) = f = 0$ e $p'_A(\lambda) = fs = 1$. FIM

Passo 2. Se $i_{max} \neq k$ então

Faça a transposição das linhas i_{max} e k das matrizes C^k e Cs^k .

Defina $fs = -fs$ e $f = -f$.

Passo 3. Calcule $fs = fs.C^k(k, k) + f.Cs^k(k, k)$ e $f = f.C^k(k, k)$.

Passo 4.

$$\tau_s(k + 1 : n) = \frac{Cs^k(k + 1 : n, k)C^k(k, k) - C^k(k + 1 : n, k)Cs^k(k, k)}{C^k(k, k).C^k(k, k)}$$

$$\tau(k + 1 : n) = C^k(k + 1 : n, k) / C^k(k, k)$$

$$Cs^{k+1}(k + 1 : n, k + 1 : n) = Cs^k(k + 1 : n, k + 1 : n) - \tau_s(k + 1 : n)C^k(k, k + 1 : n) - \tau(k + 1 : n)Cs^k(k, k + 1 : n)$$

$$C^{k+1}(k + 1 : n, k + 1 : n) = C^k(k + 1 : n, k + 1 : n) - \tau(k + 1 : n)C^k(k, k + 1 : n)$$

Passo 5. $k = k + 1$

Fim

Faça $p'_A(\lambda) = fs = fs.C^n(n, n) + f.Cs^n(n, n)$ e $p_A(\lambda) = f = f.C^n(n, n)$.

O Algoritmo 6 pode ser utilizado em cada iteração do método Newton-Maehly para determinar a direção de Newton $d = \frac{p_A(\lambda)}{p'_A(\lambda)} = \frac{f}{fs}$. Entretanto, para matrizes maiores, existe o risco de *overflows* (GANDER, 2008). Tal problema pode ser contornado empregando-se o logaritmo complexo de $p_A(\lambda)$. Assim, em lugar de computar

$f = f.C^k(k, k)$, deve-se obter

$$lf = lf + \log(C^k(k, k)), \quad (86)$$

onde lf indica o $\log(f)$.

Note que,

$$\frac{d}{d\lambda} \log(p_{\bar{A}}(\lambda)) = \frac{p'_{\bar{A}}(\lambda)}{p_{\bar{A}}(\lambda)}, \quad (87)$$

ou seja, $lfs = \frac{d}{d\lambda} \log(p_{\bar{A}}(\lambda))$ é o inverso da direção d . Além disso, temos

$$lfs = lfs + \frac{C_s^k(k, k)}{C^k(k, k)}. \quad (88)$$

O Algoritmo 7, descrito a seguir, utiliza as Eqs. (86) e (88) para calcular a direção d utilizada pelo método Newton-Maehly.

Algoritmo 7- Cálculo de $d = \frac{p_{\bar{A}}(\lambda)}{p'_{\bar{A}}(\lambda)}$.

Dados: $k = 1$, $C^k = C \in \mathbb{R}^{n \times n}$, $C_s^k = C'(\lambda) = -I_n \in \mathbb{R}^{n \times n}$, $lfs = 0$.

Enquanto $k \leq n - 1$

Passo 1. Faça $c_{max} =$ maior dos valores $|C^k(k : n, k)|$ e i_{max} a linha correspondente ao maior valor.

Se $c_{max} = 0$, faça $d = 0$. FIM

Passo 2. Se $i_{max} \neq k$ então

Faça a transposição das linhas i_{max} e k das matrizes C^k e C_s^k .

Passo 3. Calcule $lfs = lfs + \frac{C_s^k(k, k)}{C^k(k, k)}$.

Passo 4.

$$\tau_s(k + 1 : n) = \frac{C_s^k(k + 1 : n, k) C^k(k, k) - C^k(k + 1 : n, k) C_s^k(k, k)}{C^k(k, k) \cdot C^k(k, k)}$$

$$\tau(k + 1 : n) = C^k(k + 1 : n, k) / C^k(k, k)$$

$$C_s^{k+1}(k + 1 : n, k + 1 : n) = C_s^k(k + 1 : n, k + 1 : n) - \tau_s(k + 1 : n) C^k(k, k + 1 : n) - \tau(k + 1 : n) C_s^k(k, k + 1 : n)$$

$$C^{k+1}(k + 1 : n, k + 1 : n) = C^k(k + 1 : n, k + 1 : n) - \tau(k + 1 : n) C^k(k, k + 1 : n)$$

Passo 5. $k = k + 1$

Fim

Faça $lfs = lfs + \frac{C_s^n(n, n)}{C^n(n, n)}$ e $d = \frac{1}{lfs}$.

Para utilizar o método Newton-Maehly é necessário o uso de pontos iniciais adequados. Nesse contexto, o uso do MTGO ainda não é possível, pois conforme descrito no

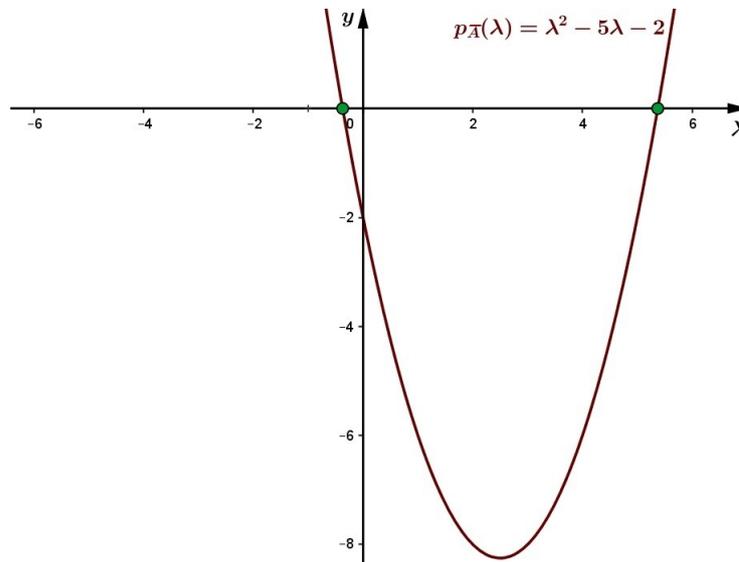
Capítulo 2, o MTGO seleciona pontos amostrais na bacia dos mínimos de uma função objetivo ϕ , ou seja, pontos que estejam suficientemente próximos dos minimizadores de ϕ . Entretanto, para resolver o problema dos autovalores, devemos obter pontos iniciais em $\Omega = R$ que estejam próximos dos zeros do polinômio característico $p_{\bar{A}}$. Para contornar essa dificuldade, vamos utilizar a função auxiliar descrita a seguir.

3.7 Função auxiliar para o MTGO

Consideremos a função polinomial definida pela regra $p_{\bar{A}}(\lambda) = \lambda^2 - 5\lambda - 2$, cujos zeros $\lambda_1 = -0,37$ e $\lambda_2 = 5,37$, correspondem aos autovalores da matriz $A = \bar{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$, definidos no intervalo de Gerschgorin $R = [-6, 6]$. O minimizador de $p_{\bar{A}}$ é $\hat{\lambda} = 2,5$ e, portanto, o MTGO irá priorizar os pontos amostrais próximos de $\hat{\lambda}$.

Na Fig. 11 temos o gráfico de $p_{\bar{A}}$. Note que a região de mínimo encontra-se no parte central do intervalo $[-0,37; 5,37]$. Portanto, o MTGO irá priorizar os pontos amostrais localizados nessa região.

Figura 11 - Gráfico da função polinomial $p_{\bar{A}}$.



Legenda: A curva em bordô descreve o gráfico da função polinomial $p_{\bar{A}}$. Os autovalores λ_1 e λ_2 são representados pelos círculos verdes.

Fonte: A autora, 2022.

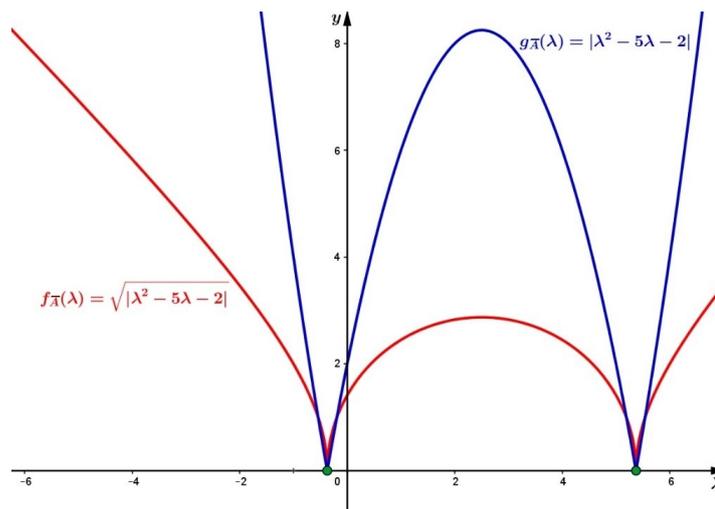
Para contornar essa dificuldade, uma possibilidade é utilizar uma função auxiliar $f_{\bar{A}}$ cujos minimizadores coincidem com os zeros de $p_{\bar{A}}$. Nesse caso, uma forma natural é

definir $f_{\bar{A}} = \left| \det(\bar{A} - \lambda I_n) \right|$. Entretanto, para matrizes maiores, existe o risco de *overflow* (GANDER, 2008).

Conforme vimos anteriormente, o determinante de $\bar{A} - \lambda I_n$ será calculado como um produto, onde os fatores serão obtidos de forma iterativa utilizando a eliminação de Gauss com pivoteamento parcial, conforme a Eq. (81). Isso nos permite definir $f_{\bar{A}}$ como $f_{\bar{A}}(\lambda) = \sqrt[n]{\left| \det(\bar{A} - \lambda I_n) \right|}$. Nesse caso, a cada iteração a raiz n -ésima tem como objetivo reduzir o valor de cada fator na multiplicação.

Na Fig. 12, é apresentado o gráfico da função $f_{\bar{A}}(\lambda) = \sqrt{\left| \det(\bar{A} - \lambda I_n) \right|}$ para a matriz $\bar{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$. Também é ilustrada a curva $|p_{\bar{A}}|$. Observe que o gráfico de $f_{\bar{A}}(\lambda)$ apresenta picos menores no intervalo $[-6, 6]$.

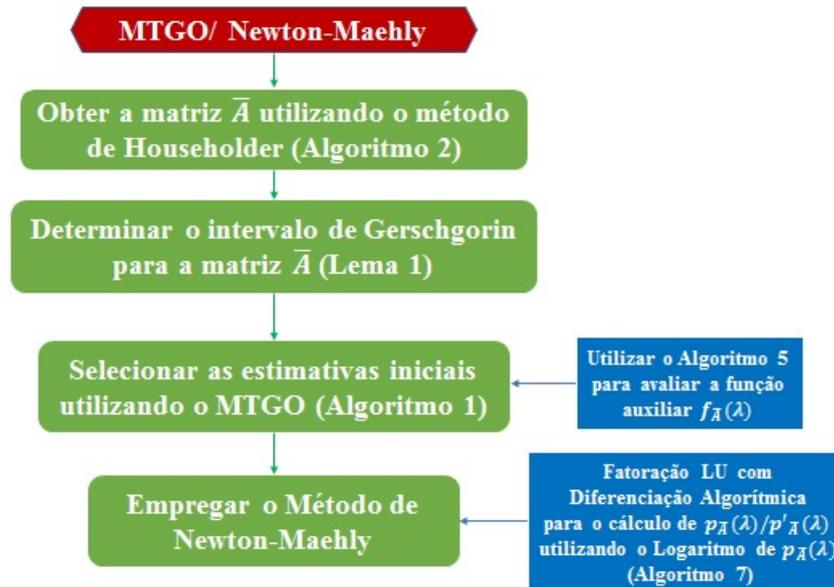
Figura 12 - Gráfico da função auxiliar $f_{\bar{A}}$.



Legenda: A curva em azul descreve o gráfico da função módulo de $p_{\bar{A}}$. A curva em vermelho descreve o gráfico da função raiz quadrada do módulo de $p_{\bar{A}}$. Os autovalores λ_1 e λ_2 são representados pelos círculos verdes.

Fonte: A autora, 2022.

O valor da função $f_{\bar{A}}(\lambda)$ pode ser calculado utilizando o Algoritmo 5. Para isso, no Passo 3 deve-se calcular $f_{\bar{A}} = f_{\bar{A}} \cdot \sqrt[n]{|C^k(k, k)|}$ e no fim $\phi = f_{\bar{A}} = f_{\bar{A}} \cdot \sqrt[n]{|C^n(n, n)|}$. Dessa forma, é possível empregar o MTGO na obtenção de estimativas iniciais adequadas para o método Newton-Maehly, sendo tal método denotado por MTGO/Newton-Maehly. O fluxograma a seguir, mostra o funcionamento desse método.



3.8 Testes numéricos

Para avaliar o desempenho da metodologia aqui proposta, foram selecionados 22 matrizes testes, cujas dimensões variam de 5 a 200. Os resultados numéricos foram obtidos utilizando um computador equipado com processador Intel® Core™ i7-4510U, 8 GB de RAM, sistema operacional Ubuntu 15.04. O código foi programado em MATLAB.

As matrizes simétricas $A \in \mathbb{R}^{n \times n}$ aqui utilizadas, tem autovalores $1, 2, \dots, n$ (GANDER, 2008). Tais matrizes foram obtidas utilizando o Algoritmo 8.

Algoritmo 8- Matriz simétrica randômica.

Dado inicial: $n > 1, n \in \mathbb{Z}$.

Passo 1. Definir $x = (1, 2, \dots, n)^T$.

Passo 2. Gerar uma matriz randômica não singular $Q_1 \in \mathbb{R}^{n \times n}$.

Passo 3. Obter uma base ortonormal β dos vetores colunas de $Q_1 \in \mathbb{R}^{n \times n}$. Os vetores de β definem as colunas de uma matriz ortogonal $Q \in \mathbb{R}^{n \times n}$.

Passo 4. Calcular a matriz simétrica $A \in \mathbb{R}^{n \times n}$ utilizando a fórmula $A = QD_xQ^T$, onde $D_x \in \mathbb{R}^{n \times n}$ é a matriz diagonal $D_{x_{ii}} = x_i, i = 1, \dots, n$.

Nas tabelas a seguir, n indica a dimensão da matriz, o número mínimo de pontos amostrais necessários para obter os autovalores de cada matriz é denotado por N_{min} , o número de semiminimizadores locais de ϕ no k^+ – *topografo* é indicado por N_s . Por sua vez, t^* representa o tempo (em segundos) necessário para resolver cada problema.

Os primeiros testes resolvidos pelo método MTGO/Newton-Maehly, foram realizados utilizando $k = 4$ vizinhos mais próximos. Neste caso, para que um ponto seja

selecionado como estimativa inicial, basta que ele seja melhor que 3 desses vizinhos mais próximos. A direção de Newton foi obtida utilizando o Algoritmo 7. Os valores N_{min} , N_s e t^* para esses testes são apresentados na Tabela 3.

Tabela 3 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - *topografo* (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 4$.

n	N_{min}	N_s	$t^*(s)$
5	13	5	0,0019
10	34	10	0,0029
15	50	15	0,0045
20	71	20	0,0082
25	77	25	0,0108
30	102	30	0,0147
35	129	35	0,0208
40	137	40	0,0276
45	145	45	0,0328
50	158	50	0,0399
55	210	55	0,0532
60	216	60	0,0610
65	213	66	0,0666
70	233	70	0,0748
75	266	75	0,0923
80	276	80	0,1035
85	279	85	0,1199
90	286	90	0,1335
95	309	95	0,1604
100	318	101	0,1669
150	513	150	0,3749
200	651	206	0,7723

Fonte: A autora, 2022.

Note que para $n \in \{65, 100, 200\}$ o número de semiminimizadores locais de ϕ no k^+ - *topografo* é superior ao número de autovalores. Por esse motivo, ao utilizar a iteração de Newton-Maehly descrita na Eq. (71), deve-se verificar para cada nova estimativa, a ocorrência da igualdade $(\lambda^k - \lambda_j) = 0$, $j = 1, \dots, m$, onde λ_j é o j -ésimo autovalor já computado.

Em seguida, realizamos os testes utilizando $k = 5$. Neste caso, para que um ponto seja selecionado como estimativa inicial, basta que este seja melhor do que 4 dos seus vizinhos mais próximos. Na Tabela 4, temos os valores N_{min} , N_s e t^* para esses testes.

Tabela 4 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - *topografo* (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 5$.

n	N_{min}	N_s	$t^*(s)$
5	19	5	0,0016
10	42	10	0,0031
15	66	15	0,0059
20	92	20	0,0086
25	102	25	0,0140
30	133	30	0,0175
35	164	35	0,0243
40	164	40	0,0288
45	190	45	0,0381
50	203	50	0,0449
55	244	55	0,0579
60	269	60	0,0658
65	262	65	0,0735
70	284	70	0,0859
75	322	75	0,1007
80	339	81	0,1172
85	341	86	0,1282
90	402	90	0,1481
95	404	95	0,1664
100	407	101	0,1830
150	580	150	0,4056
200	808	201	0,8267

Fonte: A autora, 2022.

Observe pelas Tabelas 3 e 4 que, em geral, conforme a dimensão da matriz aumenta, o número mínimo de pontos amostrais necessários também aumenta. Além disso, em todos os problemas testes, o número mínimo de pontos amostrais necessários para obter os autovalores de cada matriz é menor quando utilizado $k = 4$. Tal resultado já era

esperado, pois conforme o número de vizinhos aumenta o número de semiminimizadores locais de ϕ no k^+ - *topografo* diminui.

Com o objetivo de comparar o desempenho do MTGO/Newton-Maehly, os testes foram repetidos utilizando o método Newton-Maehly equipado com o TGO. Tal método é denotado por TGO/Newton-Maehly. Os resultados obtidos quando $k = 4$, são descritos na Tabela 5, onde o número mínimo de pontos amostrais necessários para obter os autovalores de cada matriz é denotado por TN_{min} , o número de minimizadores locais de ϕ no k^+ - *topografo* é indicado por TN_s e t_T^* representa o tempo (em segundos) necessário para resolver cada problema.

Tabela 5 - Número mínimo de pontos amostrais utilizados pelo TGO/Newton-Maehly (TN_{min}) com $k = 4$, o número de minimizadores locais de ϕ no k^+ - *topografo* (TN_s) e o tempo t_T^* utilizado.

n	TN_{min}	TN_s	$t_T^*(s)$
5	20	5	0,0018
10	52	10	0,0033
15	104	15	0,0066
20	117	20	0,0094
25	137	25	0,0141
30	209	30	0,0227
35	233	35	0,0295
40	361	40	0,0476
45	376	45	0,0548
50	281	50	0,0484
55	360	55	0,0687
60	712	60	0,1376
65	465	65	0,0982
70	492	70	0,1179
75	745	75	0,1801
80	713	80	0,1788
85	681	85	0,1808
90	741	90	0,2005
95	824	95	0,2422
100	881	100	0,2764
150	1329	150	0,6336
200	2040	200	1,3306

Comparando os resultados descritos na Tabela 5 com os resultados da Tabela 3, temos $N_{min} < TN_{min}$ em todos os testes. Esse fato está de acordo com a proposta do MTGO, que busca evitar a exclusão de bons pontos iniciais, resultando na diminuição dos pontos amostrais necessários.

Em seguida, realizamos os testes utilizando o TGO/Newton-Maehly para $k = 5$ vizinhos mais próximos. Na Tabela 6, temos os valores TN_{min} , TN_s e t_T^* para esses testes. Comparando esses resultados com os apresentados na Tabela 4, temos novamente $N_{min} < TN_{min}$ em todos os testes.

Tabela 6 - Número mínimo de pontos amostrais utilizados pelo TGO/Newton-Maehly (TN_{min}) com $k = 5$, o número de minimizadores locais de ϕ no k^+ - *topografo* (TN_s) e o tempo t_T^* utilizado.

n	TN_{min}	TN_s	$t_T^*(s)$
5	24	5	0,0017
10	57	10	0,0034
15	104	15	0,0063
20	117	20	0,0104
25	177	25	0,0158
30	220	30	0,0250
35	316	35	0,0372
40	361	40	0,0478
45	377	45	0,0541
50	344	50	0,0554
55	420	55	0,0719
60	840	60	0,1662
65	497	65	0,1028
70	504	70	0,1198
75	873	75	0,2055
80	984	80	0,2490
85	873	85	0,2253
90	869	90	0,2419
95	936	95	0,2868
100	881	100	0,2741
150	1329	150	0,6372
200	2040	200	1,3271

Fonte: A autora, 2022.

Na Subseção 3.6 vimos que, para matrizes maiores, existe o risco de *overflows*

quando se utiliza $p_{\bar{A}}(\lambda)$ para determinar a direção de Newton. Por esse motivo, recorre-se ao uso do logaritmo complexo de $p_{\bar{A}}(\lambda)$, conforme a Eq. (86). Nas Tabelas 7 e 8, são apresentados os resultados obtidos utilizando o MTGO/ Newton-Maehly, sem o uso do logaritmo de $p_{\bar{A}}(\lambda)$, para $k = 4$ e $k = 5$, respectivamente.

Tabela 7 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - *topografo* (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 4$, sem o uso do logaritmo de $p_{\bar{A}}(\lambda)$.

n	N_{min}	N_s	$t^*(s)$
5	13	5	0,0016
10	34	10	0,0030
15	50	15	0,0046
20	71	20	0,0081
25	84	26	0,0115
30	102	30	0,0162
35	129	35	0,0235
40	137	40	0,0283
45	145	45	0,0339
50	158	50	0,0392
55	210	55	0,0533
60	216	60	0,0635
65	213	66	0,0684
70	233	70	0,0797
75	266	75	0,0998
80	276	80	0,1084
85	279	85	0,1208
90	286	90	0,1448
95	309	95	0,1687
100	318	101	0,1683
150	513	150	0,3791
200	overflow		

Fonte: A autora, 2022.

Tabela 8 - Número mínimo de pontos amostrais utilizados pelo MTGO (N_{min}), o número de semiminimizadores locais de ϕ no k^+ - *topografo* (N_s) e o tempo t^* necessário para resolver cada problema utilizando o método MTGO/Newton-Maehly com $k = 5$, sem o uso do logaritmo de $p_{\bar{A}}(\lambda)$.

n	N_{min}	N_s	$t^*(s)$
5	19	5	0,0017
10	42	10	0,0032
15	66	15	0,0054
20	91	20	0,0095
25	102	25	0,0135
30	133	30	0,0187
35	164	35	0,0233
40	164	40	0,0285
45	190	45	0,0381
50	203	50	0,0503
55	244	55	0,0575
60	269	60	0,0678
65	262	65	0,0765
70	284	70	0,0931
75	322	75	0,1031
80	339	81	0,1211
85	341	86	0,1353
90	402	90	0,1529
95	404	95	0,1720
100	407	101	0,1812
150	581	151	0,4183
200	overflow		

Fonte: A autora, 2022.

Observe pelas Tabelas 3 e 7, que para $k = 4$ o número mínimo de pontos amostrais foi o mesmo, exceto para a matriz de dimensão $n = 25$. Por sua vez, pelas Tabelas 4 e 8, vê-se que para $k = 5$ o número mínimo de pontos amostrais é diferente apenas nos casos com $n = 20$ e $n = 150$. Isso indica que o uso do logaritmo complexo de $p_{\bar{A}}(\lambda)$, não afeta de forma expressiva o número de pontos amostrais necessários. Entretanto, quando $n = 200$ ocorreu *overflow*, o que justifica o uso do logaritmo.

Pelas tabelas anteriores, temos que o método MTGO/Newton-Maehly foi eficaz no cálculo dos autovalores das matrizes testes, sendo que o melhor resultado foi obtido

empregando o método MTGO/Newton-Maehly com $k = 4$ vizinhos mais próximos e utilizando o logaritmo complexo de $p_{\bar{A}}(\lambda)$.

4 PROBLEMA DE AUTOVALOR COMPLEMENTAR

O Problema de Autovalor Complementar pode ser encontrado em várias aplicações importantes nas engenharias e na física, por exemplo, na simulação de circuitos elétricos, processamento de sinais e no estudo da instabilidade de sistemas na presença de obstáculos com atrito, ver Costa et al. (2004), Costa e Seeger (2010) e Adly e Rammal (2013). Tal problema tem sido estudado por vários autores que utilizaram diferentes abordagens para obter as suas soluções, por exemplo: Adly e Seeger (2011), Ma (2012), Fernandes et al. (2014a) e Fernandes et al. (2014b).

Aqui, utilizaremos um método de direções viáveis denominado FDA-MNCP (*Feasible Direction Algorithm-Mixed Nonlinear Complementarity Problem*). Esse método, proposto por Mazonche (2007), será equipado com a Inicialização Global Topográfica Modificada. Para avaliar a performance da metodologia utilizada aqui, serão realizados experimentos computacionais com 12 problemas testes.

4.1 Descrição do problema

Seja $\bar{Q} : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ uma aplicação de classe C^1 em $\mathbb{R}^n \times \mathbb{R}$ definida por $\bar{Q}(x, \lambda) = (\sum_{i=1}^n x_i) - 1$. O problema de autovalor complementar consiste em encontrar um escalar $\lambda \in \mathbb{R}$, chamado de autovalor complementar, e um vetor correspondente $x \in \mathbb{R}^n / \{0\}$ denominado autovetor complementar, tal que,

$$\begin{cases} x_i \geq 0, \bar{F}_i(x, \lambda) \geq 0, i = 1, \dots, n. \\ x \bullet \bar{F}(x, \lambda) = 0 \\ \bar{Q}(x, \lambda) = 0 \end{cases} \quad (89)$$

onde $\bar{F}(x, \lambda) = (A - \lambda B)x$ é uma função definida no \mathbb{R}^{n+1} , e $A, B \in \mathbb{R}^{n \times n}$ são matrizes dadas, com B definida positiva (ADLY e RAMMAL, 2013). Aqui, “ \bullet ” denota o produto de Hadamard (STYAN, 1973) definido como

$$x \bullet \bar{F}(x, \lambda) = \begin{bmatrix} x_1 \bar{F}_1(x, \lambda) \\ \vdots \\ x_n \bar{F}_n(x, \lambda) \end{bmatrix}. \quad (90)$$

Note que $\bar{Q}(x, \lambda) = 0$ e $0 \leq x_i$ implicam que $x_i \leq 1$. Por outro lado, se $B = I_n \in \mathbb{R}^{n \times n}$, então o intervalo $[l_\lambda, u_\lambda] \subset \mathbb{R}$ no qual a variável $\lambda \in \mathbb{R}$ está definida, pode ser

obtido da seguinte forma,

$$l_\lambda = -\min \{\|A_1\|, \|A_\infty\|\} \leq \lambda \leq \min \{\|A_1\|, \|A_\infty\|\} = u_\lambda, \quad (91)$$

onde $\|A_1\| = \max_{1 \leq j \leq n} \{\sum_{i=1}^n |a_{ij}|\}$ e $\|A_\infty\| = \max_{1 \leq i \leq n} \{\sum_{j=1}^n |a_{ij}|\}$ (FERNANDES et al., 2014b). Além disso, de acordo com Costa e Seeger (2010), existem no máximo $\delta_n = n2^{n-1} - (n-1)$ autovalores complementares no intervalo $[l_\lambda, \mu_\lambda]$.

Definindo o conjunto $\bar{\Omega} = \{(x, \lambda) \in \mathbb{R}^{n+1}; 0 \leq x_i \leq 1, \bar{F}(x, \lambda) \geq 0\}$, cujo interior é o conjunto não vazio, denotado por $\bar{\Omega}_0$, temos que solucionar o problema descrito na Eq. (89) significa determinar os pontos em $\bar{\Omega}$ que satisfazem o sistema de equações

$$\begin{cases} x \bullet \bar{F}(x, \lambda) = 0 \\ \bar{Q}(x, \lambda) = 0 \end{cases}. \quad (92)$$

Assim, utilizando uma estimativa inicial viável adequada, podemos empregar um método de busca local, como o FDA-MNCP, para obter $(x, \lambda) \in \bar{\Omega}$ que satisfaz a Eq. (92). Entretanto, dependendo da quantidade de soluções e da rigidez das restrições, para o MTGO obter pontos viáveis próximos o suficiente de cada solução, podem ser necessário gerar muitos pontos amostrais Sobol. Por esse motivo, vamos reescrever o problema de autovalor complementar utilizando uma variável auxiliar, que tem por função relaxar as restrições do conjunto $\bar{\Omega}$, permitindo o uso de menos pontos amostrais.

Seja $Q : \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ a aplicação de classe C^1 em $\mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$ definida por $Q(x, z, \lambda) = \bar{Q}(x, \lambda) + z$, onde $z \in \mathbb{R}$ é uma variável auxiliar definida em $[0, \bar{z}] \subset \mathbb{R}$, com $\bar{z} > 0$. Consideremos então o problema de encontrar $(x, z, \lambda) \in \mathbb{R}^{n+2}$ tal que,

$$\begin{cases} H(x, z, \lambda) = X \bullet F(x, z, \lambda) = 0 \\ Q(x, z, \lambda) = 0 \\ X \geq 0, F(x, z, \lambda) \geq 0 \end{cases} \quad (93)$$

com

$$F(x, z, \lambda) = \begin{bmatrix} (A - \lambda B)x + Z \\ 1 \end{bmatrix}, \quad (94)$$

onde $X = (x, z) \in \mathbb{R}^n \times \mathbb{R}$ e $Z = (z, \dots, z) \in \mathbb{R}^n$. O produto $X \bullet F(x, z, \lambda)$ é dado por

$$X \bullet F(x, z, \lambda) = \begin{pmatrix} x_1 \cdot F_1 \\ \vdots \\ x_n \cdot F_n \\ z \cdot F_{n+1} \end{pmatrix}. \quad (95)$$

Observe que $z \cdot F_{n+1} = z \cdot 1 = 0$ se, e somente se, $z = 0$. Assim, estamos buscando

pontos do tipo $(x, 0, \lambda) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$. O teorema a seguir mostra que podemos resolver o problema descrito na Eq. (89) solucionando o problema da Eq. (93). Além disso, a recíproca também é verdadeira.

Teorema 5: O ponto $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}$ é solução do problema da Eq. (89) se, e somente se, $(x, 0, \lambda) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$ é solução do problema descrito na Eq. (93).

Demonstração: Suponhamos que (x, λ) seja solução do problema da Eq. (89), temos que $Q(x, z, \lambda) = z$. Definindo $z = 0$, resulta $Q(x, z, \lambda) = 0$, $X \geq 0$ e $F(x, z, \lambda) \geq 0$. Como $x \bullet \bar{F}(x, \lambda) = 0$ então $X \bullet F(x, z, \lambda) = 0$, ou seja, $(x, 0, \lambda)$ é solução da Eq. (93). Por outro lado, suponhamos que $(x, 0, \lambda)$ seja solução do problema descrito na Eq. (93), é imediato que $x \geq 0$, $\bar{F}(x, \lambda) \geq 0$, $x \bullet \bar{F}(x, \lambda) = 0$ e $\bar{Q}(x, \lambda) = 0$ pois $z = 0$. Logo, (x, λ) satisfaz a Eq. (89). \square

Assim, para resolver o problema de autovalor complementar descrito na Eq. (89), basta encontrar as soluções do problema apresentado na Eq. (93). Note que a variável z da Eq. (93) está definida no intervalo $[0, \bar{z}] \subset \mathbb{R}$. Para determinar a extremidade \bar{z} , propomos o Algoritmo a seguir.

Algoritmo 9- Cálculo de \bar{z} .

Dados: $s = 0$, $n_{\bar{F}} = 0$.

Passo 1. Gerar uma quantidade $N^* > k$ de pontos Sobol no hipercubo $A \subset \mathbb{R}^{n+1}$ dado por

$$A = [0, 1] \times [0, 1] \times \dots \times [0, 1] \times [l_\lambda, u_\lambda],$$

onde k é quantidade de vizinhos utilizados no MTGO e $[l_\lambda, u_\lambda] \subset \mathbb{R}$ é o intervalo no qual a variável $\lambda \in \mathbb{R}$ está definida.

Passo 2. Verificar para cada $i = 1, \dots, n$ e para todo ponto Sobol $(x, \lambda) \in A$.

Se $\bar{F}_i(x, \lambda) \leq 0$, então faça $s = s - \bar{F}_i(x, \lambda)$ e $n_{\bar{F}} = n_{\bar{F}} + 1$.

Passo 3. Se $n_{\bar{F}} = 0$, faça $X = x$ e resolva o problema descrito na Eq. (89).

Caso contrário, faça $\bar{z} = \frac{s}{n_{\bar{F}}}$ e resolva o problema descrito na Eq. (93).

Definido o intervalo $[0, \bar{z}]$, utilizaremos o MTGO para selecionar estimativas iniciais no hipercubo $S = [0, 1] \times [0, 1] \times \dots \times [0, 1] \times [0, \bar{z}] \times [l_\lambda, u_\lambda]$ para o método de busca local FDA- MNCP proposto por Mazorche (2007). Tal método utiliza a iteração de Newton para solucionar o seguinte sistema de equações

$$P(X, \lambda) = \begin{bmatrix} H(X, \lambda) \\ Q(X, \lambda) \end{bmatrix} = 0. \quad (96)$$

4.2 Algoritmo FDA-MNCP

Seja $\phi(X, \lambda) = f(X, \lambda) + Q^2(X, \lambda)$, com $f(X, \lambda) = X^T F(X, \lambda)$, uma função definida no conjunto $\Omega = \{(X, \lambda) \in \mathbb{R}^{n+2}; X \geq 0, F(X, \lambda) \geq 0\}$. Partindo de um ponto inicial, o algoritmo FDA-MNCP gera uma sequência monótona decrescente contida em Ω_c^0 , que converge para uma solução do problema descrito na Eq. (93) (MAZORCHE, 2007). Aqui, Ω_c^0 denota o interior da região compacta $\Omega_c = \{(X, \lambda) \in \Omega; \phi(X, \lambda) \leq c\}$, com $\Omega_c^0 \neq \emptyset$. Por hipótese, para todo $(X, \lambda) \in \Omega_c^0$ temos $X_i > 0$ e $F_i(X, \lambda) > 0$, $i = 1, \dots, n+1$.

Aplicando uma iteração do método de Newton ao sistema de equações descrito na Eq. (96) temos:

$$J_P(X^k, \lambda^k)d_k^\alpha = -P(X^k, \lambda^k), \quad (97)$$

onde $J_P(X^k, \lambda^k)$ é a Jacobiana de P , avaliada em (X^k, λ^k) , dada por

$$J_P(X^k, \lambda^k) = \begin{bmatrix} \nabla_X H(X^k, \lambda^k) & D_{X^k} \nabla_\lambda F(X^k, \lambda^k) \\ \nabla_X Q(X^k, \lambda^k) & \nabla_\lambda Q(X^k, \lambda^k) \end{bmatrix}, \quad (98)$$

com $\nabla_X H(X^k, \lambda^k) = D_{F(X^k, \lambda^k)} + D_{X^k} \nabla_X F(X^k, \lambda^k)$. Aqui, $D_{F(X^k, \lambda^k)}$ e D_{X^k} são matrizes diagonais definidas, respectivamente, por $D_{F(X^k, \lambda^k)}_{ii} = F_i(X^k, \lambda^k)$ e $D_{X^k}_{ii} = X_i^k$, para todo $i = 1, \dots, n+1$. Por sua vez, $\nabla_X F(X^k, \lambda^k)$ é a matriz das primeiras derivadas de F em relação a X , avaliada em (X^k, λ^k) e $\nabla_\lambda F(X^k, \lambda^k)$ é o vetor da derivada de F em relação a λ .

O vetor $d_k^\alpha \in \mathbb{R}^{n+2}$ é uma direção de descida para $\phi(x, \lambda)$ que pode ou não ser viável (MAZORCHE, 2007). Para garantir a viabilidade utiliza-se um vetor $d_k^\beta \in \mathbb{R}^{n+2}$ que tem por função causar uma deflexão no vetor d_k^α no sentido do interior de Ω . Para calcular $d_k^\beta \in \mathbb{R}^{n+2}$ deve-se resolver o seguinte sistema linear

$$J_P(X^k, \lambda^k)d_k^\beta = \rho^k \bar{E}, \quad (99)$$

onde $\bar{E} = (1, \dots, 1, 0) \in \mathbb{R}^{n+2}$, $\rho^k = \frac{\rho_0 f(X^k, \lambda^k)^\beta}{n+1} \in (0, 1)$ e $\rho_0 = \alpha \min \left\{ 1, \frac{1}{c^{\beta-1}} \right\}$, com $\alpha \in (0, 1)$ e $\beta \in (1, 2]$.

Somando membro a membro as Eqs. (97) e (99), obtemos

$$J_P(X^k, \lambda^k) (d_k^\alpha + d_k^\beta) = -P(X^k, \lambda^k) + \rho^k \bar{E}. \quad (100)$$

Assim, resolvendo o sistema da Eq. (100) obtemos d_k que é uma direção viável e

de descida para $\phi(X, \lambda)$, dada como

$$d_k = d_k^\alpha + d_k^\beta. \quad (101)$$

Os passos do FDA- MNCP são descritos no Algoritmo 10. No passo 3, é formada uma busca linear inexata que obedece ao chamado critério de decréscimo de Armijo (NOCEDAL e WRIGHT, 2006), cujo objetivo é computar um comprimento de passo na direção d_k que resulta em um decréscimo suficiente da função ϕ em relação ao valor $\phi(X^k, \lambda^k)$.

Algoritmo 10- FDA-MNCP

Parâmetros: $c, \varepsilon > 0, \alpha, \nu, \eta \in (0, 1), \beta \in (1, 2]$.

Dados: $k = 0, (X^0, \lambda^0)$ um ponto interior de Ω tal que $\phi(X^k, \lambda^k) \leq c$.

Passo 1. Calcule $\phi(X^k, \lambda^k) = f(X^k, \lambda^k) + Q^2(X^k, \lambda^k)$.

Passo 2. (Cálculo da direção de busca) Calcule d_k resolvendo o sistema linear descrito na Eq. (100).

Passo 3. (Busca linear) Calcule t , o primeiro número da sequência $\{1, \nu, \nu^2, \nu^3, \dots\}$ satisfazendo

$$X_i^k + t^k d_{k_i} \geq 0, i = 1, \dots, n + 1,$$

$$F((X^k, \lambda^k) + t^k d_k) \geq 0,$$

e

$$\phi((X^k, \lambda^k) + t^k d_k) \leq \phi(X^k, \lambda^k) + t^k \eta \nabla \phi(X^k, \lambda^k)^T d_k.$$

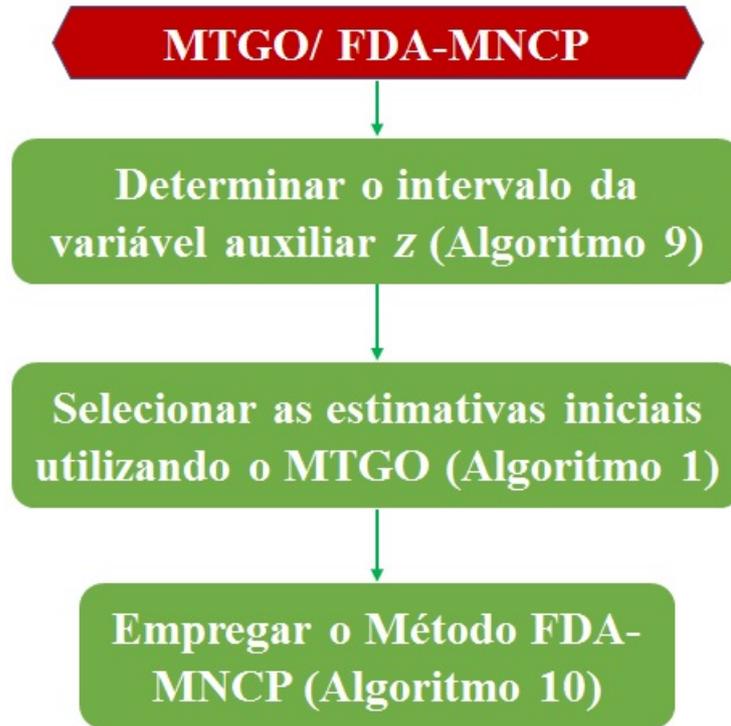
Passo 4. (Atualizações)

4.1 Faça $(X^{k+1}, \lambda^{k+1}) = (X^k, \lambda^k) + t^k d_k$.

4.2 Faça $\phi(X^{k+1}, \lambda^{k+1}) = f(X^{k+1}, \lambda^{k+1}) + Q^2(X^{k+1}, \lambda^{k+1})$.

Se $|\phi(X^{k+1}, \lambda^{k+1})| < \varepsilon$, pare. Caso contrário, faça $k = k + 1$ e volte para o passo 2.

O método de ponto interior descrito no Algoritmo 10 realiza uma busca local a partir de um ponto inicial obtido pelo MTGO. Aqui, o método FDA-MNCP equipado com o MTGO é indicado por MTGO/FDA-MNCP. O fluxograma a seguir, mostra o funcionamento desse método.



4.3 Testes numéricos

Para avaliar o desempenho da metodologia aqui proposta, foram selecionados 12 problemas descritos na literatura, ver Ma (2012), Fernandes et al. (2014a), Costa e Seeger (2010) e Adly e Rammal (2013). As matrizes A destes problemas encontram-se no apêndice A, e em todos testes utilizou-se $B = I_n \in \mathbb{R}^{n \times n}$. O número de soluções varia de 3 a 57, entretanto em alguns casos os autovalores complementares não estão descritos na literatura.

Foram utilizados os seguintes parâmetros no FDA-MNCP: $\varepsilon = 10^{-10}$, $\beta = 1, 4$, $\alpha = 0,05$, $\eta = 0,5$ e $\nu = 5/7$ e $c = \phi(X^0, \lambda^0)$. Os resultados numéricos foram obtidos utilizando um computador equipado com processador Intel® Core™ i7-4510U, 8 GB de RAM, sistema operacional Ubuntu 15.04. O código foi programado em linguagem C++.

Na Tabela 9 são descritos os valores de \bar{z} obtidos para cada teste utilizando o Algoritmo 9. Em todos os 12 testes definimos $N^* = 50000$. Também são apresentados os limites inferiores l_λ e superiores u_λ dos intervalos dos autovalores complementares e o número n_λ de autovalores complementares em cada intervalo.

Tabela 9 - Limite superior \bar{z} do intervalo no qual a variável auxiliar z está definida em cada teste. Limite inferior l_λ e superior u_λ do intervalo dos autovalores complementares. Número n_λ de autovalores complementares em cada teste.

Problema	\bar{z}	l_λ	u_λ	n_λ
1	0,737001	-9	9	3
2	1,78593	-13	13	9
3	73,154	-340,18	340,18	9
4	24,0778	-112	112	7
5	80,5598	-375	375	23
6	504,092	-2129,27	2129,27	21
7	111,124	-480	480	15
8	3160,26	-12991,63	12991,63	45
9	475,139	-1984	1984	31
10	60,4411	-226	226	9
11	74,7294	-300	300	17
12	942,389	-2541	2541	57

Fonte: A autora, 2022.

Nas Tabelas 10-21 são descritas as soluções dos problemas testes obtidas utilizando o método MTGO/FDA-MNCP. Aqui, (λ^*) indica os autovalores complementares calculados e (X^*) denota os respectivos autovetores (x, z) .

Tabela 10 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 1.

λ^*	X^*
5	(0,25; 0,75; 0)
7	(0,5; 0,5; 0)
8	(1; 0; 0)

Fonte: A autora, 2022.

Tabela 11 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 2.

λ^*	X^*
4,13397	(0; 0,78868; 0,21133; 0)
4,60208	(0,14497; 0,7825; 0,07249; 0)
5	(0,25; 0,75; 0; 0)
5,86603	(0; 0,21133; 0,78868; 0)
6	(0; 0; 1; 0)
7	(0,5; 0,5; 0; 0)
8	(1; 0; 0; 0)
9,39792	(0,47572; 0,28642; 0,23786; 0)
10	(0,66667; 0; 0,33333; 0)

Fonte: A autora, 2022.

Tabela 12 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 3.

λ^*	X^*
12,3081	(0,82542; 0,05061; 0,12397; 0)
12,36431	(0,84977; 0; 0,15023; 0)
18	(0,5505; 0,44949; 0; 0)
24	(0,44949; 0,55051; 0; 0)
36	(0; 1; 0; 0)
209,6357	(0,15023; 0; 0,84977; 0)
216	(0; 0; 1; 0)
245,692	(0,11066; 0,25782; 0,63152; 0)
252	(0; 0,2899; 0,7101; 0)

Fonte: A autora, 2022.

Tabela 13 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 4.

λ^*	X^*
4	(1; 0; 0; 0)
16	(0; 1; 0; 0)
20	(0,33333; 0,66667; 0; 0)
64	(0; 0; 1; 0)
68	(0,2; 0; 0,8; 0)
80	(0; 0,33333; 0,66667; 0)
84	(0,14286; 0,28571; 0,57143; 0)

Fonte: A autora, 2022.

Tabela 14 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 5 (continua).

λ^*	X^*
26,28229	(0,43143; 0,0762; 0; 0,49237; 0)
26,41492	(0,45581; 0,036779; 0,05812; 0,4493; 0)
28,71137	(0,45273; 0; 0,19135; 0,35593; 0)
29,1341	(0,22658; 0,24915; 0; 0,52428; 0)
32,60802	(0; 0,44613; 0; 0,55387; 0)
32,86353	(0,4258; 0; 0,28443; 0,28974; 0)
37,57671	(0,22382; 0; 0,77618; 0; 0)
41,01624	(0,12414; 0,06809; 0,80777; 0; 0)
46,46815	(0; 0,17708; 0,8229; 0; 0)
49,14347	(0,1561; 0,15887; 0,48744; 0,19759; 0)
66,97002	(0; 0,34289; 0,45656; 0,2006; 0)
77,4251	(0,78144; 0; 0,0009868; 0,21757; 0)
77,45751	(0,78229; 0; 0; 0,2177; 0)
99,42329	(0,96896; 0; 0,03104; 0; 0)
100	(1; 0; 0; 0; 0)
107,501	(0; 0,51321; 0,30194; 0,18485; 0)
127,392	(0; 0,76743; 0; 0,2326; 0)
148,5319	(0; 0,7171; 0,2829; 0; 0)
158	(0; 1; 0; 0; 0)

Tabela 14 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 5 (conclusão).

197,173	(0,34149; 0,42379; 0,11547; 0,11925; 0)
204,5836	(0,38741; 0,48201; 0; 0,13057; 0)
226,2813	(0,39347; 0,48876; 0,1178; 0; 0)
231,9223	(0,44552; 0,55448; 0; 0; 0)

Fonte: A autora, 2022.

Tabela 15 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 6.

λ^*	X^*
12,04706	(0,9171; 0,008775; 0,02149; 0,05265; 0)
12,04819	(0,92354; 0; 0,02217; 0,05429; 0)
12,0548	(0,92769; 0,01033; 0; 0,06198; 0)
12,05634	(0,93573; 0; 0; 0,06427; 0)
12,3081	(0,8254; 0,05061; 0,12397; 0; 0)
12,3643	(0,84977; 0; 0,15023; 0; 0)
18	(0,55051; 0,44949; 0; 0; 0)
24	(0,44949; 0,55051; 0; 0; 0)
36	(0; 1; 0; 0; 0)
209,6357	(0,15023; 0; 0,8498; 0; 0)
216	(0; 0; 1; 0; 0)
245,692	(0,11066; 0,25782; 0,6315; 0; 0)
252	(0; 0,2899; 0,7101; 0; 0)
1289,944	(0,06427; 0; 0; 0,9357; 0)
1296	(0; 0; 0; 1; 0)
1325,945	(0,05558; 0,13492; 0; 0,8095; 0)
1332	(0; 0,14286; 0; 0,85714; 0)
1505,952	(0,04644; 0; 0,27643; 0,67712; 0)
1512	(0; 0; 0,2899; 0,7101; 0)
1541,953	(0,04173; 0,10141; 0,2484; 0,60846; 0)
1548	(0; 0,10583; 0,25922; 0,63496; 0)

Fonte: A autora, 2022.

Tabela 16 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 7.

λ^*	X^*
4	(1; 0; 0; 0; 0)
16	(0; 1; 0; 0; 0)
20	(0,33333; 0,66667; 0; 0; 0)
64	(0; 0; 1; 0; 0)
68	(0,2; 0; 0,8; 0; 0)
80	(0; 0,33333; 0,66667; 0; 0)
84	(0,14286; 0,28571; 0,57143; 0; 0)
256	(0; 0; 0; 1; 0)
260	(0,11111; 0; 0; 0,88889; 0)
272	(0; 0,2; 0; 0,8; 0)
276	(0,09091; 0,18182; 0; 0,72727; 0)
320	(0; 0; 0,33333; 0,66667; 0)
324	(0,076923; 0; 0,30769; 0,61539; 0)
336	(0; 0,14286; 0,28571; 0,57143; 0)
340	(0,06667; 0,13333; 0,26667; 0,53333; 0)

Fonte: A autora, 2022.

Tabela 17 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 8 (continua).

λ^*	X^*
12,007737	(0,96329; 0,00152; 0,00372; 0,00912; 0,02234; 0)
12,007767	(0,964624; 0; 0,0037437; 0,009170; 0,022462; 0)
12,007921	(0,96613; 0,001561; 0; 0,009366; 0,022942; 0)
12,007952	(0,967516; 0; 0; 0,009417; 0,023067; 0)
12,008989	(0,9678; 0,001774; 0,004346; 0; 0,026078; 0)
12,009029	(0,96939; 0; 0,0043732; 0; 0,026239; 0)
12,009238	(0,97127; 0,00183; 0; 0; 0,026897; 0)
12,009281	(0,97293; 0; 0; 0; 0,027068; 0)
12,04706	(0,91708; 0,0087752; 0,02149; 0,052651; 0; 0)
12,048194	(0,92354; 0; 0,022166; 0,054295; 0; 0)
12,054797	(0,92769; 0,010329; 0; 0,061976; 0; 0)
12,056341	(0,93573; 0; 0; 0,064266; 0; 0)

Tabela 17 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 8 (conclusão).

12,308098	(0,82542; 0,050611; 0,12397; 0; 0; 0)
12,364307	(0,84977; 0; 0,15023; 0; 0; 0)
18	(0,55051; 0,44949; 0; 0; 0; 0)
24	(0,44949; 0,55051; 0; 0; 0; 0)
36	(0; 1; 0; 0; 0; 0)
209,63569	(0,150228; 0; 0,84977; 0; 0; 0)
216	(0; 0; 1; 0; 0; 0)
245,6919	(0,11066; 0,25782; 0,63152; 0; 0; 0)
252	(0; 0,2899; 0,7101; 0; 0; 0)
1289,9437	(0,064266; 0; 0; 0,93573; 0; 0)
1296	(0; 0; 0; 1; 0; 0)
1325,9452	(0,055583; 0,134917; 0; 0,8095; 0; 0)
1332	(0; 0,14286; 0; 0,85714; 0; 0)
1505,9518	(0,04644; 0; 0,27643; 0,67712; 0; 0)
1512	(0; 0; 0,2899; 0,7101; 0; 0)
1541,9529	(0,04173; 0,10141; 0,2484; 0,60846; 0; 0)
1548	(0; 0,10583; 0,25922; 0,63495; 0; 0)
7769,9907	(0,02707; 0; 0; 0; 0,9729; 0)
7776	(0; 0; 0; 0; 1; 0)
7805,9908	(0,025387; 0,062089; 0; 0; 0,91252; 0)
7812	(0; 0,063707; 0; 0; 0,93629; 0)
7985,991	(0,02329; 0; 0,13953; 0; 0,83718; 0)
7992	(0; 0; 0,14286; 0; 0,85714; 0)
8021,991	(0,022035; 0,053893; 0,13201; 0; 0,79206; 0)
8028	(0; 0,05511; 0,13498; 0; 0,80991; 0)
9065,992	(0,019369; 0; 0; 0,28428; 0,69635; 0)
9072	(0; 0; 0; 0,289898; 0,710102; 0)
9101,9921	(0,018492; 0,045237; 0; 0,27142; 0,66485; 0)
9108	(0; 0,046089; 0; 0,27654; 0,67737; 0)
9281,9922	(0,017354; 0; 0,10399; 0,25472; 0,62394; 0)
9288	(0; 0; 0,10583; 0,25922; 0,63495; 0)
9317,9923	(0,01665; 0,04073; 0,09975; 0,2443; 0,59853; 0)
9324	(0; 0,04141; 0,10144; 0,24848; 0,6087; 0)

Fonte: A autora, 2022.

Tabela 18 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 9.

λ^*	X^*
4	(1; 0; 0; 0; 0; 0)
16	(0; 1; 0; 0; 0; 0)
20	(0,33333; 0,66667; 0; 0; 0; 0)
64	(0; 0; 1; 0; 0; 0)
68	(0,2; 0; 0,8; 0; 0; 0)
80	(0; 0,33333; 0,66667; 0; 0; 0)
84	(0,142857; 0,285714; 0,571429; 0; 0; 0)
256	(0; 0; 0; 1; 0; 0)
260	(0,11111; 0; 0; 0,88889; 0; 0)
272	(0; 0,2; 0; 0,8; 0; 0)
276	(0,09091; 0,18182; 0; 0,727273; 0; 0)
320	(0; 0; 0,33333; 0,66667; 0; 0)
324	(0,076923; 0; 0,30769; 0,615385; 0; 0)
336	(0; 0,142857; 0,285714; 0,571429; 0; 0)
340	(0,06667; 0,13333; 0,26667; 0,53333; 0; 0)
1024	(0; 0; 0; 0; 1; 0)
1028	(0,058824; 0; 0; 0; 0,941176; 0)
1040	(0; 0,11111; 0; 0; 0,88889; 0)
1044	(0,052632; 0,105263; 0; 0; 0,84211; 0)
1092	(0,047619; 0; 0,19048; 0; 0,7619; 0)
1104	(0; 0,0909091; 0,18182; 0; 0,7273; 0)
1108	(0,043478; 0,086957; 0,173913; 0; 0,695652; 0)
1280	(0; 0; 0; 0,33333; 0,66667; 0)
1284	(0,04; 0; 0; 0,32; 0,64; 0)
1088	(0; 0; 0,2; 0; 0,8; 0)
1296	(0; 0,076923; 0; 0,30769; 0,61538; 0)
1300	(0,037037; 0,07407; 0; 0,2963; 0,59259; 0)
1344	(0; 0; 0,142857; 0,285714; 0,57143; 0)
1348	(0,034483; 0; 0,137931; 0,275862; 0,551724; 0)
1360	(0; 0,066667; 0,13333; 0,266667; 0,53333; 0)
1364	(0,03226; 0,06452; 0,129032; 0,258065; 0,51613; 0)

Fonte: A autora, 2022.

Tabela 19 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 10.

λ^*	X^*
-44,5908	(0,255001; 0,531518; 0,21348; 0)
-38,1664	(0,440145; 0,53979; 0,020063; 0)
-37,3528	(0,460889; 0,53911; 0; 0)
8,35279	(0,704; 0,296; 0; 0)
34	(1; 0; 0; 0)
36,67275	(0; 0,09118; 0,90882; 0)
45	(0; 0; 1; 0)
98,75721	(0,373839; 0,10175; 0,52441; 0)
115,0927	(0,41699; 0; 0,583012; 0)

Fonte: A autora, 2022.

Tabela 20 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 11.

λ^*	X^*
-51,9024	(0,040683; 0,496934; 0,13284; 0,32955; 0)
-37,3528	(0,46089; 0,53911; 0; 0; 0)
-37,9223	(0,447247; 0,53972; 0,00024; 0,012793; 0)
8,35279	(0,704; 0,296; 0; 0; 0)
34	(1; 0; 0; 0; 0)
36,3939	(0; 0,083031; 0; 0,91697; 0)
36,67275	(0; 0,09118; 0,90882; 0; 0)
44	(0; 0; 0; 1; 0)
45	(0; 0; 1; 0; 0)
92,39614	(0; 0,057998; 0,53457; 0,40744; 0)
98,75721	(0,373839; 0,101753; 0,524408; 0; 0)
99,0379	(0,375872; 0,098769; 0; 0,52536; 0)
102,846	(0; 0; 0,56126; 0,43874; 0)
114,941	(0,41744; 0; 0; 0,58256; 0)
115,0927	(0,41699; 0; 0,58301; 0; 0)
165,9022	(0,26605; 0,062836; 0,36182; 0,309297; 0)
181,1699	(0,282693; 0; 0,38438; 0,33293; 0)

Fonte: A autora, 2022.

Tabela 21 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 12 (continua).

λ^*	X^*
1,02707	(0,40571; 0,31899; 0,2753; 0; 0; 0)
68,5186	(0,34311; 0,27093; 0,23239; 0,15356; 0; 0)
86,9134	(0,34074; 0,2687; 0,23242; 0; 0,15814; 0)
159,951	(0,29229; 0,23167; 0,19905; 0,13758; 0,13942; 0)
170,1649	(0,55801; 0,44199; 0; 0; 0; 0)
246,06	(0,4334; 0,34537; 0; 0,2212; 0; 0)
262,1	(0,43657; 0,34728; 0; 0; 0,2161; 0)
346,577	(0,34922; 0,27988; 0; 0,191; 0,17987; 0)
618,656	(0,60187; 0; 0,39813; 0; 0; 0)
685,127	(0,48122; 0; 0,31659; 0,20219; 0; 0)
692,76	(0; 0,5289; 0,4711; 0; 0; 0)
698,314	(0,47896; 0; 0,3187; 0; 0,20231; 0)
762,019	(0; 0,40229; 0,35485; 0,24286; 0; 0)
774,886	(0,389979; 0; 0,25803; 0,17965; 0,17234; 0)
780,2769	(0; 0,3968; 0,35641; 0; 0,24678; 0)
788	(1; 0; 0; 0; 0; 0)
862	(0; 1; 0; 0; 0; 0)
864,3796	(0; 0,30925; 0,27471; 0,21164; 0,204387; 0)
874,326	(0,64376; 0; 0; 0,3562; 0; 0)
878,1878	(0,67926; 0; 0; 0; 0,32074; 0)
955,8978	(0; 0,54396; 0; 0,45604; 0; 0)
965,253	(0; 0,5807; 0; 0; 0,419297; 0)
995,095	(0,452671; 0; 0; 0,3084; 0,23893; 0)
1158,91	(0; 0,29534; 0; 0,42181; 0,28285; 0)
1181,945	(0; 0,2795; 0; 0,43921; 0,28131; 0)
1266,45	(0,25868; 0; 0; 0,50933; 0,23199; 0)
1295,819	(0,085625; 0,11419; 0; 0,5795; 0,22068; 0)
1299,01	(0; 0,15227; 0,11688; 0,50837; 0,22248; 0)
1308	(0; 0; 1; 0; 0; 0)
1308,1022	(0; 0,20068; 0; 0,79932; 0; 0)
1309,115	(0,070435; 0,10068; 0,05151; 0,57297; 0,2044; 0)

Tabela 21 - Soluções obtidas utilizando o MTGO/FDA-MNCP na resolução do problema 12 (conclusão).

1311,67	(0,17658; 0; 0,10369; 0,52722; 0,19251; 0)
1315,674	(0,22818; 0; 0; 0,77182; 0; 0)
1324,238	(0,069243; 0,11546; 0; 0,8153; 0; 0)
1326,802	(0; 0,15105; 0,082363; 0,76659; 0; 0)
1328,41	(0,062192; 0,108175; 0,02776; 0,80187; 0; 0)
1329,14	(0; 0; 0,83883; 0,16117; 0; 0)
1331,69	(0,17518; 0; 0,08113; 0,74368; 0; 0)
1355,04	(0; 0; 0,716688; 0; 0,28331; 0)
1363,57	(0; 0; 0,67908; 0,05001; 0,27091; 0)
1380,87	(0; 0; 0,60154; 0,39846; 0; 0)
1395,33	(0; 0; 0,5683; 0,1937; 0,23799; 0)
1418,747	(0; 0,20436; 0; 0; 0,79564; 0)
1424,977	(0,046413; 0,157128; 0; 0; 0,79646; 0)
1431,81	(0,228794; 0; 0; 0; 0,77121; 0)
1450,33	(0,002496; 0,07193; 0,26627; 0; 0,6593; 0)
1450,65	(0; 0,07207; 0,27108; 0; 0,65685; 0)
1458,646	(0,068485; 0; 0,29528; 0; 0,63624; 0)
1461,0395	(0,056137; 0; 0,31878; 0; 0,62508; 0)
1461,069	(0; 0,035669; 0,3499; 0; 0,6144; 0)
1461,351	(0,001429; 0,03392; 0,35084; 0; 0,61381; 0)
1462,915	(0,00181; 0,02797; 0,36219; 0,01087; 0,59716; 0)
1463,06	(0; 0,02835; 0,36474; 0,01442; 0,59248; 0)
1465,526	(0,031769; 0; 0,36336; 0,02811; 0,57676; 0)
1467,551	(0; 0,01277; 0,3944; 0,06732; 0,52546; 0)
1467,918	(0,01906; 0; 0,3854; 0,06085; 0,534696; 0)
1467,972	(0,00268; 0,009967; 0,39497; 0,06967; 0,5227; 0)

Fonte: A autora, 2022.

Note que alguns dos problemas aqui estudados apresentam soluções distintas com autovalores complementares muito próximos. No problema 8, por exemplo, temos as soluções (0,97127; 0,00183; 0; 0; 0,026897; 0; 12,009238) e (0,97293; 0; 0; 0; 0,027068; 0; 12,009281). Por sua vez, no problema 11 temos (0,46089; 0,53911; 0; 0; 0; -37,3528) e (0,447247; 0,53972; 0,00024; 0,012793; 0; -37,9223). Em problemas como esses, os métodos que dependem de pontos iniciais apresentam dificuldades no cálculo das soluções, devido a proximidade entre elas. Para contornar essa dificuldade é fundamental a seleção de estimativas iniciais suficientemente próximas de cada solução. Aqui, o MTGO foi eficiente

em selecionar as estimativas iniciais adequadas.

Na Tabela 22 mostramos o número mínimo \overline{NMz} de pontos amostrais necessários para obter as soluções descritas nas Tabelas 10 - 21 utilizando o MTGO. A quantidade de pontos viáveis obtidos utilizando os \overline{NMz} pontos amostrais e o número de estimativas iniciais selecionadas pelo MTGO são denotados, respectivamente, por NMz e NMz_0 .

Tabela 22 - Número mínimo de pontos amostrais utilizados pelo MTGO (\overline{NMz}), com as respectivas quantidades de pontos amostrais viáveis (NMz) e de estimativas iniciais (NMz_0) selecionadas.

Problema	\overline{NMz}	NMz	NMz_0
1	14	12	5
2	97	79	29
3	112	75	26
4	65	46	16
5	879	565	184
6	850	550	160
7	433	291	91
8	4740	2902	794
9	1326	837	278
10	293	170	52
11	3529	2168	831
12	31012	13627	4237

Fonte: A autora, 2022.

Note que nos problemas 8, 11 e 12 a quantidade de pontos iniciais selecionados pelo MTGO é muito superior ao número de soluções. Tal fato pode ser explicado pela proximidade entre algumas soluções. Além disso, a ocorrência de regiões de atração em torno de alguma solução pode exigir mais pontos amostrais, o que por sua vez pode resultar no aumento do número de pontos iniciais. No problema 11, por exemplo, existe uma região de atração em torno da solução (0,040683; 0,496934; 0,13284; 0,32955; 0; -51,9024), pois das 831 estimativas iniciais, o método convergiu para tal solução em 457.

Na Tabela 23 é apresentado o número de avaliações ($Aval_{\overline{NMz}}$) da função ϕ necessárias para solucionar cada problema teste utilizando as quantidades mínimas \overline{NMz} descritas na Tabela 22. Também é apresentado o tempo t_{MTGO} (em segundos) necessário para solucionar cada um dos problemas testes.

Tabela 23 - Número de avaliações $Aval_{\overline{NMz}}$ da função ϕ e o tempo de CPU (em segundos) necessário para resolver cada problema teste utilizando as quantidades \overline{NMz} .

Problema	$Aval_{\overline{NMz}}$	t(s)
1	182	0,008
2	1215	0,025
3	2552	0,024
4	881	0,018
5	19485	0,096
6	22486	0,104
7	6149	0,056
8	150077	0,740
9	29703	0,215
10	5329	0,027
11	68554	0,473
12	482396	10,445

Fonte: A autora, 2022.

Comparando o número de avaliações, observa-se que nos problemas 8, 11 e 12 foram necessárias muitas avaliações da função ϕ . Isso pode ser explicado pelo grande número de estimativas iniciais. Além disso, os problemas 8 e 12 apresentam mais soluções. Observe ainda que o problema 12, que apresenta 57 soluções, foi solucionado em 10,445 segundos, ou seja, o tempo médio para obter cada solução foi 0,18 segundos.

Esses resultados indicam que o MTGO/FDA-MNCP é uma ferramenta que pode ser utilizada para solucionar o Problema de Autovalor Complementar.

5 PROBLEMA DE OTIMIZAÇÃO RESTRITA MODIFICADO

Nosso objetivo é utilizar a Inicialização Global Topográfica Modificada para obter estimativas iniciais adequadas para o método de busca local FDIPA (Feasible Direction Interior Point Algorithm), que será empregado na resolução do problema de otimização com restrições de igualdade e desigualdade. Tal problema será reescrito utilizando uma variável auxiliar, visando relaxar as restrições que definem o conjunto viável, facilitando a obtenção de pontos amostrais viáveis.

5.1 Formulação do problema

Seja $D \subset \mathbb{R}^w$ o conjunto viável definido como

$$D = \{y \in A; r_i(y) \leq 0 \text{ e } s_j(y) = 0, i = 1, \dots, m, j = 1, \dots, q\}, \quad (102)$$

onde $A \subset \mathbb{R}^w$ é o hipercubo dado por $A = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_w, b_w]$. O problema de otimização restrita considerado neste trabalho pode ser definido como

$$\text{minimize } \zeta(y) \text{ sujeito a } y \in D, \quad (103)$$

onde as funções $\zeta : \mathbb{R}^w \rightarrow \mathbb{R}$, $r_i : \mathbb{R}^w \rightarrow \mathbb{R}$ e $s_j : \mathbb{R}^w \rightarrow \mathbb{R}$ são por hipótese funções suaves, diferenciáveis, mas não necessariamente convexas (HERSKOVITS, 1998).

Diz-se que $y^* \in D$ é uma solução local (minimizador local) do problema da Eq. (103), se existe $\epsilon > 0$ tal que $\zeta(y) \geq \zeta(y^*)$, para todo $y \in D \setminus \{y^*\}$ e $\|y - y^*\| < \epsilon$. Por outro lado, se $\zeta(y) \geq \zeta(y^*)$, para todo $y \in D \setminus \{y^*\}$, então $y^* \in D$ é a solução global (minimizador global) do problema descrito na Eq. (103) e $\zeta(y^*)$ é o valor ótimo.

Note que para solucionar o problema descrito na Eq. (103) devemos avaliar as informações tanto da função objetivo ζ como das restrições. Por esse motivo, neste trabalho vamos recorrer à função Lagrangiana (LAGRANGE, 1788). Tal função agrega informações da função objetivo e das restrições do problema. Além disso, a condição clássica de otimalidade é dada em termos da Lagrangiana.

Definição 25: Sejam $y \in \mathbb{R}^w$, $\bar{\lambda} \in \mathbb{R}^m$ e $\bar{\mu} \in \mathbb{R}^q$. A função Lagrangiana associada ao problema (103) é a função $L_\zeta : \mathbb{R}^w \times \mathbb{R}^m \times \mathbb{R}^q \rightarrow \mathbb{R}$, definida como

$$L_\zeta(y, \bar{\lambda}, \bar{\mu}) = \zeta(y) + \bar{\lambda}^T r(y) + \bar{\mu}^T s(y), \quad (104)$$

onde $\bar{\lambda}$ e $\bar{\mu}$ são chamados de multiplicadores de Lagrange associados ao ponto y .

Definição 26: O gradiente da função Lagrangiana descrita na Eq. (104) é a aplicação $\nabla L_\zeta : \mathbb{R}^w \times \mathbb{R}^m \times \mathbb{R}^q \rightarrow \mathbb{R}^w$, dada por

$$\nabla L_\zeta (y, \bar{\lambda}, \bar{\mu}) = \nabla \zeta (y) + \nabla r (y) \bar{\lambda} + \nabla s (y) \bar{\mu}. \quad (105)$$

Definição 27: A Hessiana da função Lagrangiana da Eq. (104) é a função $H : \mathbb{R}^w \times \mathbb{R}^m \times \mathbb{R}^q \rightarrow \mathbb{R}^{w \times w}$, definida por

$$H (y, \bar{\lambda}, \bar{\mu}) = \nabla^2 L_{yy} (y, \bar{\lambda}, \bar{\mu}) = \nabla^2 \zeta (y) + \sum_{i=1}^m \bar{\lambda}_i \nabla^2 r_i (y) + \sum_{j=1}^q \bar{\mu}_j \nabla^2 s_j (y), \quad (106)$$

onde $\nabla^2 \zeta (y)$, $\nabla^2 r_i (y)$ e $\nabla^2 s_j (y)$ correspondem às matrizes hessianas de ζ , r_i e s_j , respectivamente.

Definição 28: Um ponto $y \in D$ é dito regular se para todo i tal que $r_i (y) = 0$, os vetores $\nabla r_i (y)$, $i = 1, \dots, m$, são linearmente independentes.

O teorema a seguir, provado por William Karush, Harold Kuhn e Albert Tucker, ver Karush (1939) e Kuhn e Tucker (1951), estabelece as condições necessárias para um ponto y ser solução local do problema descrito na Eq. (103).

Teorema 6: [Condição necessária de otimalidade de Karush-Kuhn-Tucker (KKT)] Se um ponto regular $y \in D$ é uma solução local do problema da Eq. (103), então o seguinte sistema de equações e inequações é satisfeito,

$$\nabla \zeta (y) + \nabla r (y) \bar{\lambda} + \nabla s (y) \bar{\mu} = 0, \quad (107)$$

$$R (y) \bar{\lambda} = 0, \quad (108)$$

$$s_j (y) = 0, \quad j = 1, \dots, q, \quad (109)$$

$$r_i (y) \leq 0, \quad i = 1, \dots, m, \quad (110)$$

$$\bar{\lambda}_i \geq 0, \quad i = 1, \dots, m, \quad (111)$$

onde $R (y) \in \mathbb{R}^{m \times m}$ é uma matriz diagonal, com $R_{ii} (y) = r_i (y)$, para todo $i = 1, \dots, m$, $\bar{\lambda}$ e $\bar{\mu}$ são as variáveis duais associadas às restrições de desigualdade e igualdade, respectivamente. Este sistema é chamado de sistema de Karush-Kuhn-Tucker.

5.2 Reformulação do problema utilizando uma variável auxiliar

Para resolver problemas como o descrito na Eq. (103), métodos como o FDIPA, que será apresentado ainda neste capítulo, fazem uso de um conjunto viável formado

apenas por restrições de desigualdade. No caso do problema (103) o conjunto viável D assume a forma

$$\bar{\Omega} = \{y \in A; r_i(y) \leq 0 \text{ e } s_j(y) \leq 0, i = 1, \dots, m, j = 1, \dots, q\}. \quad (112)$$

Para obter determinada quantidade \bar{N} de pontos $y \in \bar{\Omega}$, podemos utilizar um gerador de amostras aleatórias, por exemplo, a sequência de Sobol. Tais pontos são gerados no hipercubo A . Em seguida, verifica-se quais dos pontos satisfazem as restrições de desigualdade. Dependendo da rigidez das restrições $r_i(y) \leq 0$ e $s_j(y) \leq 0$, a tarefa de obter pontos amostrais viáveis demanda um grande número de pontos amostrais.

Para ilustrar esse fato, consideremos o conjunto $\bar{\Omega}$ definido pelas seguintes desigualdades:

$$\left\{ \begin{array}{l} r_1(y) = -35,82 + 0,222y_{10} + by_9 \leq 0, \\ r_2(y) = 133 - 3y_7 + ay_{10} \leq 0, \\ r_3(y) = 35,82 - 0,222y_{10} - by_9 - y_9(1/b - b) \leq 0, \\ r_4(y) = -133 + 3y_7 - ay_{10} - y_{10}(1/a - a) \leq 0, \\ r_5(y) = -1,12y_1 - 0,13167y_1y_8 + 0,00667y_1y_8^2 + ay_4 \leq 0, \\ r_6(y) = -57,425 - 1,098y_8 + 0,038y_8^2 - 0,325y_6 + ay_7 \leq 0, \\ r_7(y) = 1,12y_1 + 0,13167y_1y_8 - 0,00667y_1y_8^2 - ay_4 - y_4(1/a - a) \leq 0, \\ r_8(y) = 57,425 + 1,098y_8 - 0,038y_8^2 + 0,325y_6 - ay_7 - y_7(1/a - a) \leq 0, \\ s_1(y) = 1,22y_4 - y_1 - y_5 \leq 0, \\ s_2(y) = (98000y_3)/(y_4y_9 + 1000y_3) - y_6 \leq 0, \\ s_3(y) = (y_2 + y_5)/y_1 - y_8 \leq 0, \\ 0,00001 \leq y_1 \leq 2000, 0; 0,00001 \leq y_2 \leq 16000, 0; \\ 0,00001 \leq y_3 \leq 120, 0; 0,00001 \leq y_4 \leq 5000, 0; \\ 0,00001 \leq y_5 \leq 2000, 0; 85,0 \leq y_6 \leq 93, 0; \\ 90,0 \leq y_7 \leq 95, 0; 3,0 \leq y_8 \leq 12, 0; \\ 1,2 \leq y_9 \leq 4, 0; 145,0 \leq y_{10} \leq 162, 0; \\ a = 0,99; b = 0,9. \end{array} \right. \quad (113)$$

Utilizando a sequência de Sobol, é necessário gerar 563683 pontos amostrais para obter o primeiro ponto viável em $\bar{\Omega}$. Para obter 5 pontos viáveis são necessários 1571473 pontos amostrais. Tal exemplo, mostra que restrições muito rígidas exigem que muitos pontos amostrais sejam gerados. Para contornar essa dificuldade e relaxar as restrições $r_i(y) \leq 0$ e $s_j(y) \leq 0$ propomos o uso de uma variável auxiliar.

Seja $S \subset \mathbb{R}^n$ o hipercubo definido por $S = A \times [a_n, b_n]$, onde $n = w + 1$, $b_n = 0$ e $a_n < 0$. Vamos definir as funções $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ e $h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ da seguinte

forma:

$$\begin{cases} g_i(x) = r_i(y) + x_n, & i = 1, \dots, m, \\ h_j(x) = s_j(y) + x_n, & j = 1, \dots, q, \\ h_l(x) = x_n, & l = q + 1, \\ f(x) = \zeta(y), \end{cases} \quad (114)$$

onde $x = (y, x_n) \in \mathbb{R}^n$. Note que $r_i(y) \leq 0 \Rightarrow g_i(x) \leq 0$ e $s_j(y) \leq 0 \Rightarrow h_j(x) \leq 0$. Por outro lado, a recíproca não é verdadeira. De fato, podemos ter $g_i(x) \leq 0$ mesmo quando $r_i(y) > 0$. Para isso, basta tomar $x_n \leq -r_i(y)$. De forma análoga, temos que se $s_j(y) > 0$, então $h_j(x) \leq 0$ para todo $x_n \leq -s_j(y)$.

Pela Eq. (114) temos que o vetor gradiente de f , denotado por $\nabla f(x)$ é dado por

$$\nabla f(x) = \begin{bmatrix} \nabla \zeta(y) \\ 0 \end{bmatrix}. \quad (115)$$

Por sua vez, as matrizes $\nabla g(x) \in \mathbb{R}^{n \times m}$ das primeiras derivadas parciais de g e $\nabla h(x) \in \mathbb{R}^{n \times l}$ das primeiras derivadas parciais de h são dadas, respectivamente, por $\left\{ \right.$

$$\nabla g(x) = \begin{bmatrix} \nabla g_1(x) & \dots & \nabla g_m(x) \end{bmatrix} = \begin{bmatrix} \nabla r_1(y) & \dots & \nabla r_m(y) \\ 1 & \dots & 1 \end{bmatrix} \quad (116)$$

e

$$\nabla h(x) = \begin{bmatrix} \nabla h_1(x) & \dots & \nabla h_l(x) \end{bmatrix} = \begin{bmatrix} \nabla s_1(y) & \dots & \nabla s_{l-1}(y) & \bar{0} \\ 1 & \dots & 1 & 1 \end{bmatrix}, \quad (117)$$

onde $\bar{0} \in \mathbb{R}^{n-1}$ é o vetor nulo.

Utilizando as funções $f : S \subset \mathbb{R}^n \rightarrow \mathbb{R}$, $g : S \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$ e $h : S \subset \mathbb{R}^n \rightarrow \mathbb{R}^l$ vamos definir o seguinte problema de otimização

$$\text{minimize } f(x) \text{ sujeito a } g_i(x) \leq 0 \text{ e } h_j(x) = 0, \quad i = 1, \dots, m, \quad j = 1, \dots, l. \quad (118)$$

Sob certas hipóteses de regularidade, o teorema a seguir tem como consequência que para toda solução do problema (118) podemos obter uma solução do problema (103).

Teorema 7: Seja $(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^l$ um ponto que satisfaz as condições KKT do problema descrito na Eq. (118). Então o ponto $(y, \bar{\lambda}, \bar{\mu}) \in \mathbb{R}^{n-1} \times \mathbb{R}^m \times \mathbb{R}^{l-1}$, com $y = (x_1, \dots, x_{n-1})$, $\bar{\lambda} = \lambda$ e $\bar{\mu} = (\mu_1, \dots, \mu_{l-1})$, é um ponto KKT do problema (103).

Demonstração: Como $(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^l$ satisfaz as condições de otimalidade de primeira ordem de Karush-Kuhn-Tucker (KKT) do problema descrito na Eq. (118), então

$$\nabla f(x) + \nabla g(x) \lambda + \nabla h(x) \mu = 0, \quad (119)$$

$$G(x) \lambda = 0, \quad (120)$$

$$h_j(x) = 0, \quad j = 1, \dots, l, \quad (121)$$

$$g_i(x) \leq 0, \quad i = 1, \dots, m, \quad (122)$$

$$\lambda_i \geq 0, \quad i = 1, \dots, m, \quad (123)$$

onde $G(x) \in \mathbb{R}^{m \times m}$ é uma matriz diagonal, com $G_{ii}(x) = g_i(x)$, para todo $i = 1, \dots, m$, λ e μ são as variáveis duais associadas às restrições de desigualdade e igualdade, respectivamente.

Utilizando as Eqs. (114)-(117), podemos reescrever a Eq. (119) como

$$\begin{bmatrix} \nabla \zeta(y) \\ 0 \end{bmatrix} + \sum_{i=1}^m \lambda_i \begin{bmatrix} \nabla r_i(y) \\ 1 \end{bmatrix} + \sum_{j=1}^{l-1} \mu_j \begin{bmatrix} \nabla s_j(y) \\ 1 \end{bmatrix} + \mu_l \begin{bmatrix} \bar{O} \\ 1 \end{bmatrix} = 0, \quad (124)$$

o que implica,

$$\nabla \zeta(y) + \sum_{i=1}^m \lambda_i \nabla r_i(y) + \sum_{j=1}^{l-1} \mu_j \nabla s_j(y) = \nabla \zeta(y) + \nabla r(x) \bar{\lambda} + \nabla s(x) \bar{\mu} = 0, \quad (125)$$

onde $y = (x_1, \dots, x_{n-1})$, $\bar{\mu} = (\mu_1, \dots, \mu_{l-1})$ e $\bar{\lambda} = \lambda$.

Por sua vez, pela Eq. (121) temos

$$h(x) = \begin{bmatrix} s_1(y) + x_n \\ \vdots \\ s_{l-1}(y) + x_n \\ x_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \Rightarrow x_n = 0.$$

Daí resulta que

$$s_j(y) = 0, \quad j = 1, \dots, l-1, \quad (126)$$

$$r_i(y) = g_i(x) \leq 0, \quad i = 1, \dots, m, \quad (127)$$

e

$$G(x)\lambda = \begin{bmatrix} r_1(y) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & r_m(y) \end{bmatrix} \bar{\lambda} = 0. \quad (128)$$

■

Pelo teorema 7, podemos obter uma solução global do problema (103), a partir de uma solução global do problema (118). Por outro lado, o teorema a seguir tem como consequência que para toda solução do problema descrito na Eq. (118), pode ser obtido utilizando uma solução correspondente da Eq. (103).

Teorema 8: Seja $(y, \bar{\lambda}, \bar{\mu}) \in \mathbb{R}^{n-1} \times \mathbb{R}^m \times \mathbb{R}^{l-1}$ um ponto KKT do problema descrito na Eq. (103). Então o ponto $(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^l$, com $x = (y, 0)$, $\bar{\lambda} = \lambda$ e $\mu = (\bar{\mu}, \mu_l)$, $\mu_l = -\lambda_1 - \dots - \lambda_m - \mu_1 - \dots - \mu_{l-1}$, é um ponto KKT do problema (118).

Demonstração: Como $(y, \bar{\lambda}, \bar{\mu}) \in \mathbb{R}^{n-1} \times \mathbb{R}^m \times \mathbb{R}^{l-1}$ satisfaz o sistema descrito nas Eqs. (107)-(111), então definindo $x_n = 0$, temos de imediato que

$$\begin{bmatrix} s_1(y) \\ \vdots \\ s_{l-1}(y) \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \Rightarrow h(x) = \begin{bmatrix} s_1(y) + x_n \\ \vdots \\ s_{l-1}(y) + x_n \\ x_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad (129)$$

$$g_i(x) = r_i(y) + x_n = r_i(y) \leq 0, \quad i = 1, \dots, m, \quad (130)$$

e

$$G(x)\lambda = \begin{bmatrix} r_1(y) + x_n & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & r_m(y) + x_n \end{bmatrix} \bar{\lambda} = 0. \quad (131)$$

Por fim, $\mu_l = -\bar{\lambda}_1 - \dots - \bar{\lambda}_m - \bar{\mu}_1 - \dots - \bar{\mu}_{l-1}$, se, e somente se, $\bar{\lambda}_1 + \dots + \bar{\lambda}_m + \bar{\mu}_1 + \dots + \bar{\mu}_{l-1} + \mu_l = 0$. Temos, então, pelas Eqs. (107) e (115)-(117),

$$\nabla f(x) + \sum_{i=1}^m \bar{\lambda}_i \nabla g_i(x) + \sum_{j=1}^{l-1} \bar{\mu}_j \nabla h_j(x) + \mu_l \nabla h_l(x) = 0, \quad (132)$$

ou seja,

$$\nabla f(x) + \nabla g(x)\lambda + \nabla h(x)\mu = 0. \quad (133)$$

■

Para definir o hipercubo $S \subset \mathbb{R}^n$, utilizou-se o intervalo $[a_n, 0]$ no qual a variável x_n está definida. Para calcular o valor a_n , propomos o Algoritmo 11 a seguir.

Algoritmo 11- Cálculo de a_n .

Dados: $s_g = 0$, $s_h = 0$, $n_{gh} = 0$ e a quantidade k de vizinhos utilizados no MTGO.

Passo 1. Gerar uma quantidade $N^* > k$ de pontos Sobol no hipercubo A dado por

$$A = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_{n-1}, b_{n-1}].$$

Passo 2. Verificar para cada $i = 1, \dots, m$ e para todo ponto Sobol $y \in A$.

Se $r_i(y) \geq 0$, então faça $s_g = s_g + r_i(y)$ e $n_{gh} = n_{gh} + 1$.

Passo 3. Verificar para cada $j = 1, \dots, q$ e para todo ponto Sobol $y \in A$.

Se $s_j(y) \geq 0$, então faça $s_h = s_h + s_j(y)$ e $n_{gh} = n_{gh} + 1$.

Passo 4. Se $\left(\frac{s_g + s_h}{n_{gh}}\right) > 10^3$, faça $a_n = -\alpha \left(\frac{s_g + s_h}{n_{gh}}\right)$, $g_i(x) = \alpha g_i(x)$, $i = 1, \dots, m$ e $h_j(x) = \alpha h_j(x)$, $j = 1, \dots, l - 1$, onde α é uma constante tal que $\alpha \left(\frac{s_g + s_h}{n_{gh}}\right) < 1$.

Caso contrário, se $\left(\frac{s_g + s_h}{n_{gh}}\right) < 10^3$, faça $a_n = -\left(\frac{s_g + s_h}{n_{gh}}\right)$.

Caso contrário, se $n_{gh} = 0$, faça $n = w$ e resolva o problema descrito na Eq. (103).

Uma vez obtido o valor a_n utilizaremos o MTGO para selecionar estimativas iniciais no hipercubo S para o método de busca local FDIPA. A seguir apresentamos o método FDIPA para solução do problema descrito na Eq. (118).

5.3 Método de busca local FDIPA

O método FDIPA foi desenvolvido para resolver problemas de otimização não lineares e diferenciáveis com restrições de igualdade e/ou desigualdade (HERSKOVITS, 1998). Como as restrições de igualdade devem estar ativas na solução, o FDIPA faz uso da função de penalidade ϕ , dada por,

$$\phi(x) = f(x) + \sum_{j=1}^l c_j |h_j(x)|, \quad (134)$$

onde $c_j > 0$, $j = 1, \dots, l$. Neste caso, existe $c \in \mathbb{R}^l$ tal que o valor mínimo de ϕ sujeito apenas a restrições de desigualdade ocorre na solução do problema descrito na Eq. (118). No entanto, embora não existam parâmetros de penalidade indefinidamente crescentes, a função ϕ não possui derivadas em pontos onde as restrições de igualdade estão ativas (HERSKOVITS, 1998).

Para superar essa limitação e permitir o uso da função de penalidade ϕ , as restrições de igualdade são redefinidas de forma que todas tenham o mesmo sinal (não

positivo). Assim, o conjunto viável do problema assume a forma

$$\Omega = \{x \in S; g_i(x) \leq 0 \text{ e } h_j(x) \leq 0, i = 1, \dots, m, j = 1, \dots, l\}, \quad (135)$$

Aqui, o interior de Ω é denotado por Ω^0 .

Conforme descrito em Herskovits (1998), em relação ao problema (118), as seguintes hipóteses deverão ser observadas.

Hipótese 1: Existe um número real a tal que o conjunto $\Omega_a = \{x \in \Omega; f(x) \leq a\}$ é compacto e tem um interior Ω_a^0 .

Hipótese 2: $x \in \Omega_a^0 \Rightarrow g_i(x) < 0, i = 1, \dots, m$.

Hipótese 3: As funções f, g e h são continuamente diferenciáveis em Ω_a e têm derivadas que satisfazem a condição de Lipschitz, ou seja, existem constantes positivas K, L_i e M_j tais que

$$\|\nabla f(x') - \nabla f(x)\| \leq K \|x' - x\|, \quad (136)$$

$$\|\nabla g_i(x') - \nabla g_i(x)\| \leq L_i \|x' - x\|, i = 1, \dots, m, \quad (137)$$

e

$$\|\nabla h_j(x') - \nabla h_j(x)\| \leq M_j \|x' - x\|, j = 1, \dots, l. \quad (138)$$

Hipótese 4: (Condição de Regularidade) Para todo $x \in \Omega_a$ e para todo i tal que $g_i(x) = 0$, os vetores $\nabla g_i(x)$ são linearmente independentes.

Sob essas hipóteses, Herskovitz demonstrou que dado um ponto $x_0 \in \Omega^0$, o método FDIPA gera uma sequência de pontos $\{x_k\} \subset \Omega^0$ tal que $\phi(x_{k+1}) < \phi(x_k)$ (HERSKOVITS, 1998). Esta sequência converge para um ponto que satisfaz as condições de otimalidade de primeira ordem de Karush-Kuhn-Tucker (KKT) do problema descrito na Eq. (118).

A k -ésima iteração de Newton para resolver as Eqs. (119)-(121) é definida na forma matricial como

$$\begin{aligned} & \begin{bmatrix} B_k & \nabla g(x_k) & \nabla h(x_k) \\ \Lambda_k \nabla g^T(x_k) & G(x_k) & 0 \\ \nabla h^T(x_k) & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{k+1} - x_k \\ \lambda_{k+1} - \lambda_k \\ \mu_{k+1} - \mu_k \end{bmatrix} \\ & = - \begin{bmatrix} \nabla f(x_k) + \nabla g(x_k) \lambda_k + \nabla h(x_k) \mu_k \\ G(x_k) \lambda_k \\ h(x_k) \end{bmatrix}, \quad (139) \end{aligned}$$

onde $\Lambda_k \in \mathbb{R}^{m \times m}$ é a matriz diagonal com $\Lambda_{ii} = \lambda_i, i = 1, \dots, m$, e $B_k = H(x_k, \lambda_k, \mu_k)$. No entanto, devido ao alto custo computacional para calcular $H(x_k, \lambda_k, \mu_k)$, podemos definir

B_k como uma aproximação quasi-Newton de H ou a matriz identidade (HERSKOVITS, 1998).

Seja d_k^α a direção no espaço primal definida por $d_k^\alpha = x_{k+1} - x_k$. O sistema da Eq. (139) pode ser reescrito como

$$B_k d_k^\alpha + \nabla g(x_k) \lambda_k^\alpha + \nabla h(x_k) \mu_k^\alpha = -\nabla f(x_k), \quad (140)$$

$$\Lambda_k \nabla g^T(x_k) d_k^\alpha + G(x_k) \lambda_k^\alpha = 0, \quad (141)$$

$$\nabla h^T(x_k) d_k^\alpha = -h(x_k), \quad (142)$$

onde $\lambda_k^\alpha = \lambda_{k+1}$ e $\mu_k^\alpha = \mu_{k+1}$. Observe que o sistema definido pelas equações (140)-(142) é independente do valor atual de μ .

De acordo com Herskovits (1998), o vetor d_k^α é uma direção de descida para ϕ , mas pode não ser viável em relação a Ω . De fato, se $g_i(x) \rightarrow 0$, para algum $i = 1, \dots, m$, então pela Eq. (141) temos $\nabla g_i^T(x_k) d_k^\alpha \rightarrow 0$, ou seja, d_k^α tende a uma direção tangente ao conjunto viável. Da mesma forma, se $h_j(x) \rightarrow 0$, para algum $j = 1, \dots, l$, então pela Eq. (142) obtemos $\nabla h_j^T(x_k) d_k^\alpha \rightarrow 0$.

Para garantir a viabilidade da direção de busca, utiliza-se o vetor $d_k^\beta \in \mathbb{R}^n$ para desviar o vetor $d_k^\alpha \in \mathbb{R}^n$ para o interior da região viável. O vetor d_k^β é obtido resolvendo o seguinte sistema linear

$$B_k d_k^\beta + \nabla g(x_k) \lambda_k^\beta + \nabla h(x_k) \mu_k^\beta = 0, \quad (143)$$

$$\Lambda_k \nabla g^T(x_k) d_k^\beta + G(x_k) \lambda_k^\beta = -\Lambda_k \omega^g, \quad (144)$$

$$\nabla h^T(x_k) d_k^\beta = -\omega^h, \quad (145)$$

onde $\omega^g > 0$ e $\omega^h > 0$ são fatores de deflexão associados a restrições de desigualdade e igualdade, respectivamente.

Assim, obtemos d_k que é uma direção viável e de descida, definida como uma composição das direções d_k^α e d_k^β ,

$$d_k = d_k^\alpha + \rho_k d_k^\beta. \quad (146)$$

O escalar $\rho_k > 0$ determina o grau de deflexão no sentido do interior da região viável, garantindo que d_k seja também uma direção de descida para ϕ . Isso é feito impondo limites superiores para ρ_k (HERSKOVITS, 1998). Com efeito, se $\nabla \phi^T(x_k) d_k^\alpha < 0$, obtemos $\nabla \phi^T(x_k) d_k < 0$ impondo $\nabla \phi^T(x_k) d_k \leq \xi \nabla \phi^T(x_k) d_k^\alpha$, com $\xi \in (0, 1)$. Assim, usando a Eq. (146) obtemos $\rho_k < (\xi - 1) \frac{\nabla \phi^T(x_k) d_k^\alpha}{\nabla \phi^T(x_k) d_k^\beta}$. A seguir, descrevemos o algoritmo FDIPA.

Algoritmo 12- FDIPA

Parâmetros: Escolher $\varepsilon > 0$, $\varphi > 0$, $\xi \in (0, 1)$, $\eta \in (0, 1)$ e $\nu \in (0, 1)$.

Dados: $k = 0$, $x_k \in \Omega^0$, $0 < \lambda_k \in \mathbb{R}^m$, $0 < c \in \mathbb{R}^l$, $0 < \omega^g \in \mathbb{R}^m$, $0 < \omega^h \in \mathbb{R}^l$ e $B_k \in \mathbb{R}^{n \times n}$ simétrica e definida positiva.

Passo 1. Calcular $\nabla L_x(x_k, \lambda_k, \mu_k) = \nabla f(x_k) + \nabla g(x_k) \lambda_k + \nabla h(x_k) \mu_k$.

Passo 2. Cálculo da direção de busca d_k .

2.1. Calcular d_k^α e λ_k^α resolvendo o seguinte sistema linear

$$\begin{cases} B_k d_k^\alpha + \nabla g(x_k) \lambda_k^\alpha + \nabla h(x_k) \mu_k^\alpha = -\nabla f(x_k) \\ \Lambda_k \nabla g^T(x_k) d_k^\alpha + G(x_k) \lambda_k^\alpha = 0 \\ \nabla h^T(x_k) d_k^\alpha = -h(x_k). \end{cases}$$

Caso $\|d_k^\alpha\| \leq \varepsilon$, pare.

2.2. Calcular d_k^β e λ_k^β resolvendo o seguinte sistema linear

$$\begin{cases} B_k d_k^\beta + \nabla g(x_k) \lambda_k^\beta + \nabla h(x_k) \mu_k^\beta = 0 \\ \Lambda_k \nabla g^T(x_k) d_k^\beta + G(x_k) \lambda_k^\beta = -\Lambda_k \omega^g \\ \nabla h^T(x_k) d_k^\beta = -\omega^h \end{cases}$$

2.3. Se $c_j < 1, 2\mu_k^\alpha$, então faça $c_j = -2\mu_k^\alpha$, $j = 1, \dots, l$.

2.4. Se $\nabla \phi^T(x_k) d_k^\beta > 0$, faça

$$\rho_k = \min\{\varphi \|d_k^\alpha\|^2; (\xi - 1) \nabla \phi^T(x_k) d_k^\alpha / \nabla \phi^T(x_k) d_k^\beta\}.$$

Caso contrário, faça

$$\rho_k = \varphi \|d_k^\alpha\|^2.$$

2.5. Calcule a direção de busca

$$d_k = d_k^\alpha + \rho_k d_k^\beta$$

e

$$\bar{\lambda}_k = \lambda_k^\alpha + \rho_k \lambda_k^\beta.$$

Passo 3. (Busca Linear) Calcule t , o primeiro número da sequência $\{1, \nu, \nu^2, \nu^3, \dots\}$ que satisfaz

$$\phi(x_k + t_k d_k) \leq \phi(x_k) + t_k \eta \nabla \phi^T(x_k) d_k,$$

$$h_j(x_k + t_k d_k) \leq 0, \quad j = 1, \dots, l,$$

$$g_i(x_k + t_k d_k) < 0, \quad \text{se } \bar{\lambda}_{k_i} \geq 0,$$

$$g_i(x_k + t_k d_k) \leq g_i(x_k), \quad \text{se } \bar{\lambda}_{k_i} < 0.$$

Algoritmo 12- FDIPA (continuação)

Passo 4. Atualizações.

4.1 Faça $x_{k+1} = x_k + t_k d_k$.

4.2. Faça

$$\nabla L_x(x_{k+1}, \lambda_k, \mu_k) = \nabla f(x_{k+1}) + \nabla g(x_{k+1}) \lambda_k + \nabla h(x_{k+1}) \mu_k,$$

$$\gamma = \nabla L_x(x_{k+1}, \lambda_k, \mu_k) - \nabla L_x(x_k, \lambda_k, \mu_k),$$

$$\delta = t_k d_k,$$

$$\phi = \begin{cases} 1, & \text{se } \delta^T \gamma \geq 0, 2\delta^T B_k \delta \\ (0, 8\delta^T B_k \delta) / (\delta^T B_k \delta - \delta^T \gamma), & \text{se } \delta^T \gamma < 0, 2\delta^T B_k \delta \end{cases},$$

$$\sigma = \phi \gamma + (1 - \phi) B_k \delta,$$

$$B_{k+1} = B_k - \frac{B_k \delta \delta^T B_k}{\delta^T B_k \delta} + \frac{\sigma \sigma^T}{\delta^T \sigma}.$$

4.3. Para todo $i = 1, \dots, m$, faça $\lambda_{k+1_i} = -1/g(x_{k+1})$.

Se $\lambda_{k+1_i} > 1$, então faça $\lambda_{k+1_i} = 1$.

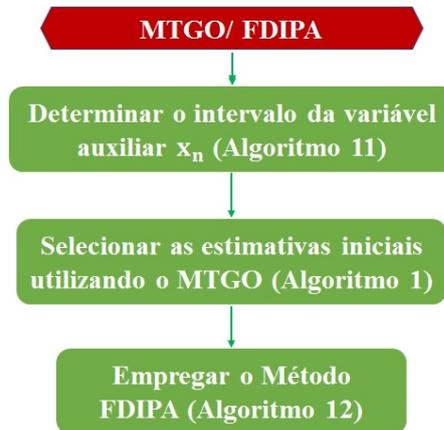
4.4. Faça $k = k + 1$. Volte para o Passo 2.

Observa-se que para a seleção do comprimento do tamanho do passo (no Passo 3), neste algoritmo realiza-se uma busca linear inexata que obedece a condição de Armijo. Assim, no Passo 4.1, o novo ponto é determinado por $x_{k+1} = x_k + t_k d_k$, onde t_k é o comprimento do passo.

No Passo 4.2, a matriz B (simétrica e definida positiva) é atualizada usando a fórmula de atualização quasi-Newton BFGS. Para garantir B definida positiva, a atualização BFGS utiliza a chamada condição de curvatura $\delta^T \gamma > 0$, que pode não ser satisfeita quando a função objetivo é a Lagrangiana. Para solucionar esse problema, em Herskovits (1998) recomenda-se o uso da fórmula BFGS com a modificação de Powell, Powell (1978), conforme descrito no Passo 4.2.

Por outro lado, no Passo 4.3 as variáveis duais associadas às restrições de desigualdade são incrementadas conforme a restrição correspondente tende para zero. Desta forma, em lugar da direção de busca apontar para a fronteira da restrição, ela desliza ao longo da fronteira, tornando os passos mais longos e conseqüentemente melhorando a eficiência do algoritmo (HERSKOVITS, 1998). A atualização de λ deve garantir convergência aos multiplicadores ótimos na solução, preservando desta forma a convergência global. Por sua vez, a atualização das variáveis duais associadas às restrições de igualdade, μ , não é considerada, uma vez que o sistema de equações (140)-(142), independe destas variáveis.

O método de pontos interiores descrito no Algoritmo 12, realiza uma busca local a partir de um ponto inicial. No presente trabalho, as estimativas iniciais foram obtidas utilizando a Inicialização Global Topográfica Modificada e o método é denominado MTGO-FDIPA. O fluxograma a seguir, mostra o funcionamento desse método.



5.4 Testes numéricos

Resolvemos 18 problemas testes, onde 7 deles possuem mais de uma solução, veja Apêndice B. Os primeiros 15 problemas apresentam restrições de igualdade e desigualdade e foram escolhidos para avaliar a eficiência do método MTGO-FDIPA na resolução de problemas com diferentes características, como o número de variáveis, a linearidade, o tipo e a quantidade de restrições. Por sua vez, os problemas 16-18, são casos particulares do problema descrito na Eq. (103), com restrições apenas de desigualdades. Tais problemas, de importância na engenharia, foram escolhidos para comparar o desempenho do MTGO-FDIPA com o de outros métodos descritos na literatura.

Primeiramente, analisamos os problemas 1-15 que apresentam restrições gerais. Para solucionar tais problemas foram utilizados os seguintes parâmetros no método FDIPA: $\varepsilon = 10^{-12}$; $\varphi = 0,8$; $\xi = 0,7$; $\eta = 0,1$ e $\nu = 5/8$, $\lambda_0 = (1, \dots, 1)$, $c_j = 100$, $\omega^g = (1, \dots, 1) \in \mathbb{R}^m$, $\omega^h = (1, \dots, 1) \in \mathbb{R}^l$ e $B_0 = I_n \in \mathbb{R}^{n \times n}$. Em todos os testes definimos $N^* = 5000$. Os resultados numéricos foram obtidos utilizando um computador equipado com processador Intel® Core™ i7-4510U, 8 GB de RAM, sistema operacional Ubuntu 15.04. O código foi programado em linguagem C++.

Nas Tabelas 24-26 são descritas as soluções dos problemas testes obtidas utilizando a variável auxiliar x_n . Nessas tabelas, (x^*) e $\phi(x^*)$ denotam, respectivamente, os minimizadores globais e o valor ótimo obtido pelo MTGO-FDIPA. O limite inferior do intervalo da variável x_n é dado por a_n . Os resultados obtidos para os problemas 1-8 são

apresentados na Tabela 24.

Tabela 24 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 1-8.

Problema	a_n	x^*	$\phi(x^*)$
1	-635,898	(679, 942; 1026, 07; 0, 11888; -0, 39623; 0)	5126,5
2	-76,4133	(0; 100; 0; 100; 0; 0; 100; 200; 0, 01; 0)	-400
3	-90,14	(0, 82288; 0, 91144; 0)	1,39495
4	-0,734218	(0; 0; 0, 99991; 0)	1
5	-2,83266	(1; 4, 743; 3, 8212; 1, 3794; 0)	17,014
6	-1,15571	(0, 63552; 0; 0, 3127; 0, 05178; 0)	29,8944
7	-580,75	(1698, 09; 15817, 8; 54, 1055; 3031, 22; 2000; 90, 1158; 95; 10, 4929; 1, 56164; 153, 535; 0)	-1768,8069
8	-599,157	(776, 159; 925, 195; 0, 0511089; -0, 428891; 0)	5174,41

Fonte: A autora, 2022.

Os problemas 9-12 são modificações de problemas multimodais sujeitos à restrições de desigualdades apresentados em Hock e Schittkowski (1981) e Li et al. (2020). Para isso, foram adicionadas restrições de igualdade, que correspondem às condições de otimalidade de primeira ordem de Karush-Kuhn-Tucker (KKT) dos problemas com restrições de desigualdade (ISMAILOV e SOLODOV, 2009). Na Tabela 25 apresentamos os resultados obtidos pelo MTGO-FDIPA para esses problemas.

Tabela 25 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 9-12.

Problema	a_n	x^*	$\phi(x^*)$
9	-9,51959	(4; 2, 82843; 2; 0, 70711; 0), (-4; -2, 82843; 2; 0, 70711; 0), (4; -2, 82843; -2; 0, 70711; 0), (-4; 2, 82843; -2; 0, 70711; 0)	-22,6274
10	-35,9991	(3, 14159; 2, 275; 0; 0; 0), (-3, 14159; 12, 275; 0; 0; 0), (9, 42478; 2, 475; 0; 0; 0)	0,397887
11	-19,6559	(0, 089842; -0, 712656; 0; 0; 0; 0) (-0, 089842; 0, 712656; 0; 0; 0; 0)	-1,03163
12	-33,234	(0; 0; 0; 1; 0, 0625; 0), (0; 0; 0; -1; 0, 0625; 0)	1

Fonte: A autora, 2022.

Por fim, na Tabela 26 apresentamos os resultados obtidos para os problemas 13-15. Tais problemas são reformulações dos problemas de autovalores complementares

descritos em MA (2012). Conforme mostrado na Tabela 26, tais problemas têm até 23 soluções globais.

Tabela 26 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 13-15.

Problema	a_n	x^*	$\phi(x^*)$
13	-45,7688	(0, 43143; 0, 076204; 0; 0, 49237; 26, 2823; 0) (0, 455806; 0, 03678; 0, 05812; 0, 449297; 26, 4149; 0) (0, 45273; 0; 0, 19135; 0, 35593; 28, 7114; 0) (0, 22658; 0, 24915; 0; 0, 52428; 29, 1341; 0) (0; 0, 44613; 0; 0, 55387; 32, 608; 0) (0, 42583; 0; 0, 28443; 0, 28974; 32, 8635; 0) (0, 22382; 0; 0, 77618; 0; 37, 5767; 0) (0, 12414; 0, 06809; 0, 80777; 0; 41, 0162; 0) (0; 0, 17708; 0, 82292; 0; 46, 4681; 0) (0, 1561; 0, 15887; 0, 487439; 0, 19759; 49, 1435; 0) (0; 0, 342887; 0, 456563; 0, 20055; 66, 97; 0) (0, 78144; 0; 0, 000987; 0, 21757; 77, 4251; 0) (0, 78229; 0; 0; 0, 21771; 77, 4575; 0) (0, 96896; 0; 0, 031045; 0; 99, 4233; 0) (1; 0; 0; 0; 100; 0) (0; 0, 513208; 0, 301942; 0, 18485; 107, 501; 0) (0; 0, 76743; 0; 0, 23257; 127, 392; 0) (0; 0, 7171; 0, 2829; 0; 148, 532; 0) (0; 1; 0; 0; 158; 0) (0, 341489; 0, 423788; 0, 115468; 0, 119254; 197, 173; 0) (0, 387412; 0, 482014; 0; 0, 130574; 204, 584; 0) (0, 393472; 0, 488755; 0, 117773; 0; 226, 281; 0) (0, 445524; 0, 554476; 0; 0; 231, 922; 0)	1
14	-8,3324	(0; 0, 788675; 0, 211325; 4, 13397; 0), (0, 144971; 0, 782543; 0, 0724857; 4, 60208; 0), (0, 25; 0, 75; 0; 5; 0), (0; 0, 211325; 0, 788675; 5, 86603; 0), (0; 0; 1; 6; 0), (0, 5; 0, 5; 0; 7; 0), (1; 0; 0; 8; 0), (0, 475718; 0, 286423; 0, 237859; 9, 39792; 0), (0, 666667; 0; 0, 333333; 10; 0)	1
15	-0,464482	(0, 25; 0, 75; 5; 0), (0, 5; 0, 5; 7; 0), (1; 0; 8; 0)	1

Buscando analisar o desempenho do MTGO-FDIPA na resolução dos problemas testes 1-15, utilizamos ainda duas outras abordagens. A primeira, chamada de TGO-FDIPA, consiste no emprego do TGO com busca local feita pelo FDIPA. Na outra abordagem, empregou-se o método FDIPA para realizar a busca local a partir dos pontos iniciais viáveis sem o uso do método topográfico. Tal método é indicado por NFDIPA.

Na Tabela 27 mostramos o número mínimo \overline{NMx} de pontos amostrais necessários para obter as soluções descritas nas Tabelas 24-26 utilizando o MTGO-FDIPA. A quantidade de pontos viáveis obtidos utilizando os \overline{NMx} pontos amostrais e o número de estimativas iniciais selecionadas pelo MTGO, são denotados, respectivamente, por NMx e NMx_0 . Para comparação, também apresentamos o número mínimo \overline{NTx} de pontos amostrais necessários quando utilizado o TGO-FDIPA, com a respectiva quantidade NTx de pontos viáveis e o número NTx_0 de estimativas iniciais. O número mínimo de pontos amostrais necessários ao utilizar o NFDIPA é denotado por \overline{NSx} e o número de pontos viáveis é dado por NSx .

Tabela 27 - Número mínimo de pontos amostrais utilizados pelo MTGO-FDIPA (\overline{NMx}) e TGO-FDIPA (\overline{NTx}), com as respectivas quantidade de pontos viáveis (NMx) e (NTx), e de estimativas iniciais (NMx_0) e (NTx_0). Número mínimo de pontos amostrais (\overline{NSx}) utilizados pelo NFDIPA com o respectivo número de pontos viáveis (NSx).

Problema	\overline{NMx}	NMx	NMx_0	\overline{NTx}	NTx	NTx_0	\overline{NSx}	NSx
1	86	5	1	86	5	1	20	1
2	377	5	1	377	5	1	257	3
3	8	5	1	8	5	1	2	1
4	7	5	1	7	5	1	2	1
5	9	6	1	16	13	3	2	1
6	26	5	1	26	5	1	8	1
7	89	6	3	116	9	2	36	3
8	86	5	1	86	5	1	20	1
9	65	18	7	137	38	8	37	11
10	59	17	9	105	32	10	59	17
11	95	36	12	165	62	9	12	5
12	20	7	3	74	25	4	20	7
13	3812	2268	908	9698	5788	1172	3511	2086
14	99	93	40	346	308	41	55	52
15	13	10	4	41	26	5	4	4

Fonte: A autora, 2022.

Observe ainda que para todos os problemas temos $\overline{NSx} < \overline{NMx} \leq \overline{NTx}$. Tal resultado já era esperado, pois tanto o TGO quanto o MTGO exigem pelo menos $k + 1$ pontos amostrais. Note ainda que no problema 13 a quantidade de pontos iniciais selecionados pelo MTGO é muito superior ao número de soluções globais. Esse fato pode ser explicado pela ocorrência das regiões de atração em torno de uma solução. De fato, na ocorrência de uma região de atração o método tende a convergir para determinada solução.

Tal dificuldade pode ser contornada obtendo-se um ponto suficientemente próximo de cada solução, o que é feito aumentando o valor \overline{NMx} resultando no aumento de NMx_0 . Além disso, quando dois minimizadores estão muito próximos, pode ser necessário o uso de mais estimativas iniciais para haver convergência para ambas soluções. No problema 13, a distância euclidiana entre as soluções $(0, 78144; 0; 0, 000987; 0, 21757; 77, 4251; 0)$ e $(0, 78229; 0; 0; 0, 21771; 77, 4575; 0)$ é 0,032.

Para mostrar a importância do uso da variável auxiliar x_n , apresentamos na Tabela 28 o número mínimo \overline{NM} de pontos amostrais necessários para o MTGO-FDIPA solucionar os problemas 1-15 sem o uso da variável auxiliar x_n . Também apresentamos o número mínimo \overline{NT} de pontos amostrais necessários quando utilizado o TGO-FDIPA e o número mínimo \overline{NS} de pontos amostrais necessários para o NFDIPA.

Tabela 28 - Número mínimo de pontos amostrais quando não se utilizou a variável auxiliar x_n . \overline{NM} , \overline{NT} e \overline{NS} indicam, respectivamente, o número mínimo necessário para o MTGO-FDIPA, TGO-FDIPA e NFDIPA (continua).

Problema	\overline{NM}	\overline{NT}	\overline{NS}
1	7302	7302	142
2	22097	25217	1905
3	1260	1260	332
4	7	7	2
5	13	13	2
6	313	313	97
7	1571473	1571473	1571473
8	32526	32526	3686
9	838	3150	662
10	315787	400011	230571
11	588	1436	588
12	7577	11269	2801

Tabela 28 - Número mínimo de pontos amostrais quando não se utilizou a variável auxiliar x_n . \overline{NM} , \overline{NT} e \overline{NS} indicam, respectivamente, o número mínimo necessário para o MTGO-FDIPA, TGO-FDIPA e NFDIPA (conclusão).

13	1475977	1475977	1475977
14	99349	220196	37957
15	36	61	8

Fonte: A autora, 2022.

Note que nos testes 1-15 temos $\overline{NMx} \leq \overline{NM}$, $\overline{NTx} \leq \overline{NT}$ e $\overline{NSx} \leq \overline{NS}$. Esses resultados mostram a importância do uso da variável auxiliar para reduzir a quantidade de pontos amostrais necessários.

Para mostrar a importância de utilizar o MTGO, apresentamos na Tabela 29 o número de avaliações da função ϕ necessárias para solucionar cada problema teste utilizando as quantidades \overline{NMx} , \overline{NTx} e \overline{NSx} , descritas na Tabela 27. Aqui, $Eval_{\overline{NMx}}$ denota o número de avaliações de ϕ para \overline{NMx} pontos, utilizando o MTGO-FDIPA. Por sua vez, $Eval_{\overline{NTx}}$ e $Eval_{\overline{NSx}}$ indicam o número de avaliações de ϕ utilizando os métodos TGO-FDIPA e NFDIPA, para as quantidades \overline{NTx} e \overline{NSx} , respectivamente.

Tabela 29 - Número de avaliações $Eval_{\overline{NMx}}$, $Eval_{\overline{NTx}}$ e $Eval_{\overline{NSx}}$ da função ϕ necessárias para solucionar os problemas 1-15, utilizando os métodos MTGO-FDIPA, TGO-FDIPA e NFDIPA, respectivamente (continua).

Problema	$Eval_{\overline{NMx}}$	$Eval_{\overline{NTx}}$	$Eval_{\overline{NSx}}$
1	7984	7984	3337
2	4081	4081	8510
3	151	151	201
4	14331	14331	14900
5	6330	1786	7879
6	523	523	619
7	368438	661503	1515971
8	2909	2909	5050
9	902	1378	1559
10	1087	4478	1903
11	4730	5582	1218
12	55	92	187

Tabela 29 - Número de avaliações $Eval_{NMx}$, $Eval_{NTx}$ e $Eval_{NSx}$ da função ϕ necessárias para solucionar os problemas 1-15, utilizando os métodos MTGO-FDIPA, TGO-FDIPA e NFDIPA, respectivamente (conclusão).

13	172840	235891	343184
14	5537	6664	7313
15	220	403	241

Fonte: A autora, 2022.

Note que $Eval_{NSx}$ foi o menor apenas nos problemas 1 e 11. No caso do problema 11, temos $NSx < NMx_0$ e $NSx < NTx_0$, o que significa que mais buscas locais foram realizadas quando se utilizou o refinamento, justificando assim o número maior de avaliações de ϕ . Por outro lado, nos demais problemas, o uso de pontos de partida obtidos utilizando refinamento foi mais adequado do que o uso dos pontos de amostrais viáveis obtidos sem o uso de refinamento.

Nos problemas 7 e 13 foram necessárias muitas avaliações da função de penalidade. No problema 13, isso ocorreu devido ao grande número de estimativas iniciais. Por sua vez, o problema 7 apresenta o efeito Maratos (HERSKOVITS, 1998), necessitando de muitas avaliações da função de penalidade ϕ apesar de possuir apenas uma única solução.

Note que, com exceção do problema 5, temos $Eval_{NMx} \leq Eval_{NTx}$. Isso ocorre mesmo nos casos onde $NTx_0 < NMx_0$. Nesses casos, o MTGO selecionou pontos que o TGO descartou e que exigiram menos do método de busca local, ou seja, os pontos de partida são distintos e aqueles obtidos pelo MTGO foram mais adequados do que os pontos selecionados pelo TGO.

Aqui, avaliamos também o tempo de CPU (em segundos) necessário para obter as soluções descritas nas Tabelas 24-26. Na Tabela 30 são apresentados os tempos necessários para os métodos MTGO-FDIPA e NFDIPA determinarem os minimizadores globais dos problemas 1-15. Note que, o MTGO-FDIPA necessitou de menos tempo em 86,67% dos 15 problemas considerados.

Tabela 30 - Tempo (em segundos) necessário para obter as soluções descritas nas Tabelas 24-26 utilizando os métodos MTGO-FDIPA e NFDIPA.

Problema	MTGO-FDIPA (s)	NFDIPA (s)
1	0,066	0,031
2	0,12	0,235
3	0,006	0,009
4	0,102	0,14
5	0,04	0,063
6	0,01	0,016
7	11,923	65,153
8	0,018	0,023
9	0,023	0,028
10	0,032	0,037
11	0,042	0,021
12	0,009	0,011
13	3,176	5,098
14	0,131	0,156
15	0,007	0,015

Fonte: A autora, 2022.

Na Tabela 31 são descritas as soluções obtidas pelo MTGO-FDIPA para os problemas 16-18. Para obter tais soluções, foram utilizados os seguintes parâmetros no método FDIPA: $\varepsilon = 10^{-12}$; $\varphi = 0,8$; $\xi = 0,5$; $\eta = 0,1$ e $\nu = 10/19$, $\lambda_0 = (10, \dots, 10)$, $c_j = 100$, $\omega^g = (1, \dots, 1) \in \mathbb{R}^m$, $\omega^h = (1, \dots, 1) \in \mathbb{R}^l$ e $B_0 = I_n \in \mathbb{R}^{n \times n}$. Além disso, nos problemas 16 e 17 utilizou-se $\bar{N} = 200$ pontos amostrais Sobol e no problema 18 foi utilizado $\bar{N} = 100$.

Tabela 31 - Soluções obtidas utilizando o MTGO-FDIPA na resolução dos problemas 16-18.

Problema	a_n	x^*	$\phi(x^*)$
16	-0,45067	(0, 20573; 3, 47049; 9, 03662; 0, 20573; 0)	1,7248523
17	-0,76974	(0, 0516575; 0, 355959; 11, 3336; 0)	0,012665
18	-8,09234	(0, 788674; 0, 40825; 0)	263,895843

Fonte: A autora, 2022.

Para avaliar o desempenho do MTGO-FDIPA, comparamos o presente método, com 6 métodos robustos, que foram usados anteriormente com sucesso para determinar

as soluções dos problemas 16-18, a saber: (1) MVDE (*Multi-View Differential Evolution*) (MELO e CAROSIO, 2013), (2) CMA-ES (*Covariance Matrix Adaptation Evolution Strategy*) (MELO e IACCA, 2014), (3) IAPSO (*Improved Accelerated Particle Swarm Optimization*) (GUEDRIA, 2016), (4) IABC-MAL (*Improved Artificial Bee Colony with Modified Augmented Lagrangian*) (LONG et al., 2018), (5) C-ITGO (*Constrained Iterative Topographical Global Optimization*) (FERREIRA; ROCHA; NETO e SACCO, 2018) e (6) DAPSO-GA (*Dynamic Adaptive Particle Swarm Optimization and Genetic Algorithm*) (ZHU; HU e ZHU, 2019).

Na Tabela 32 mostramos o número de avaliações da função objetivo para o método MTGO-FDIPA e para esses 6 outros métodos modernos. No caso dos métodos CMA-ES e C-ITGO os valores apresentados correspondem ao número médio de avaliações da função em diferentes execuções.

Tabela 32 - Número de avaliações da função objetivo necessárias para solucionar os problemas 16-18 utilizando o MTGO-FDIPA e 6 outros métodos.

Métodos	Problema 16	Problema 17	Problema 18
MVDE	15000	10000	7000
CMA-ES	4659	19444,9	1706
IAPSO	12500	2000	-
IABC-MAL	15000	15000	15000
C-ITGO	940,68	535,08	136,48
DAPSO-GA	13356	2000	7131
MTGO-FDIPA	678	404	120

Fonte: A autora, 2022.

Observe que para a configuração utilizada, o método MTGO-FDIPA realizou menos avaliações da função objetivo para resolver cada um dos problemas testes 16-18, seguido pelo método C-ITGO. Isso indica que o MTGO foi capaz de fornecer bons pontos iniciais para o método de busca local FDIPA, que por sua vez, se mostrou eficaz na solução desses problemas.

CONCLUSÕES E PERSPECTIVAS

Neste trabalho, utilizamos a Inicialização Global Topográfica Modificada para gerar pontos iniciais adequados para os métodos de busca local, utilizados aqui para resolver os problemas de autovalores de matrizes reais simétricas, autovalores complementares e minimização de funções sujeitas a restrições de igualdade e desigualdade.

Para obter os autovalores de matrizes reais simétricas foi utilizado o método MTGO/Newton- Maehly. O problema de computar os autovalores a partir dos coeficientes de polinômio característico é mal condicionado. Tal dificuldade foi contornada avaliando o polinômio característico de forma direta sem representar explicitamente os coeficientes. O primeiro passo, foi modificar a matriz inicial através da transformação de Householder. Os autovalores foram, então, calculados utilizando a iteração de Newton-Maehly, a partir de pontos iniciais gerados pelo MTGO. Para a avaliação da direção de Newton utilizou-se a Eliminação de Gauss com pivoteamento parcial, empregando também a diferenciação algorítmica. Essa abordagem do problema de autovalor utilizando o MTGO/Newton-Maehly, não é encontrada na literatura, e constitui uma importante contribuição do presente trabalho. Para avaliar o desempenho dessa metodologia foram utilizadas 22 matrizes com diferentes dimensões e em todos os experimentos o método obteve todos os autovalores.

Em seguida foi estudado o problema de autovalores complementares. O problema original foi reescrito utilizando uma variável auxiliar permitindo a obtenção de mais pontos amostrais viáveis. Para obter as soluções desse problema, foi utilizado o método MTGO/FDA-MNCP. Tal abordagem mostrou-se muito adequada, sendo uma importante contribuição deste trabalho. Para avaliar a performance foram estudados 12 problemas descritos na literatura que possuem até 57 soluções. Em todos os testes o MTGO/FDA-MNCP foi capaz de obter todas as soluções.

Por fim, para calcular as soluções do problema de minimização de uma função sujeita a restrições de igualdade e desigualdade, foi utilizado o método MTGO-FDIPA. Para isso, o problema foi reescrito utilizando uma variável auxiliar. Em geral, para uma certa quantidade de pontos amostrais, tal abordagem permitiu obter um número maior de pontos amostrais viáveis para serem refinados pelo MTGO. Para avaliar a eficiência do MTGO-FDIPA, foram realizados 18 testes numéricos, e em todos o método obteve as soluções globais.

Assim, em vista dos resultados obtidos, concluimos que o trabalho atingiu seus objetivos, mostrando que o uso da Inicialização Global Topográfica Modificada para gerar estimativas iniciais adequadas para os métodos de busca local é uma metodologia com potencial na resolução dos problemas aqui estudados.

Perspectivas futuras

Como perspectivas futuras imediatas, tem-se utilizar a Inicialização Global Topográfica Modificada na solução do problema de autovalor para matrizes reais não simétricas.

Já para o caso do problema de autovalores complementares, tem-se o objetivo de utilizar a matriz B diferente da matriz identidade (I_n) e realizar aplicações na área de Termodinâmica.

REFERÊNCIAS

- ADLY, S.; RAMMAL, H. A new method for solving Pareto eigenvalue complementarity problems. *Computational Optimization and Applications*, v. 55, p. 703-731, 2013.
- ADLY, S.; SEEGER, A. A non smooth algorithm for cone-constrained eigenvalue problems. *Computational Optimization and Applications*, v. 49, p. 299-318, 2011.
- ANTON, H.; RORRES, C. Álgebra Linear com aplicações. 10^a edição. Porto Alegre: Bookman, 2012.
- ANTONOV, I. A.; SALEEV, V. M. An economic method of computing LP_T – sequences. *USSR Computational Mathematics and Mathematical Physics*, v. 19, p. 252-256, 1979.
- BOLLOBÁS, B. Modern Graph Theory. New York: Springer, 1998.
- BRATLEY, P.; FOX, B. L. Algorithm 659: Implementing Sobol’s quasirandom sequence generator. *ACM Transactions on Mathematical Software*, v. 14, p. 88-100, 1988.
- BURDEN, R. L.; FAIRES, D. J.; BURDEN, A. M. Análise Numérica. 3^a edição. São Paulo: Cengage Learning, 2015.
- CHAPRA, S. C.; CANALE, R. P. Numerical methods for engineers. 6^a edição. Boston: McGraw-Hill Higher Education, 2010.
- COSTA, A.; MARTINS, J.; FIGUEIREDO, I.; JÚDICE, J. The directional instability problem in systems with frictional contacts. *Computer Methods in Applied Mechanics and Engineering*, v. 193, p. 357-384, 2004.
- COSTA, A.; SEEGER, A. Cone-constrained eigenvalue problems: theory and algorithms. *Computational Optimization and Applications*. v. 45, p. 25-57, 2010.
- COSTA, J. *Otimização global topográfica aplicada a problemas de otimização com restrições de igualdade e desigualdade*. Dissertação (Mestrado) – Instituto Politécnico, Universidade do Estado do Rio de Janeiro, Nova Friburgo, 2016.
- EPPERSON, J. F. An Introduction to Numerical Methods and Analysis. 2^a edição. New Jersey: John Wiley & Sons, 2013.
- EVANS, L. Complex Numbers and Vectors. Camberwell: ACER Press, 2006.
- FERNANDES, L. M.; JÚDICE, J.; SHERALI, H. D.; FUKUSHIMA, M. On the computation of all eigenvalues for the eigenvalue complementarity problems. *Journal of Global Optimization*, v. 59, p. 307-326, 2014a.
- FERNANDES, L. M.; JÚDICE, J.; SHERALI, H. D., FORJAZ, M. On an enumerative algorithm for solving eigenvalue complementarity problems. *Computational Optimization and Applications*, v. 59, p. 113-134, 2014b.
- FERREIRA, M. P.; ROCHA, M. L.; NETO, A. J. S.; SACCO, W. F.. A constrained

ITGO heuristic applied to engineering optimization. *Expert Systems with Applications*, v. 110, p. 106-124, 2018.

FRANCIS, J. G. F. The QR transformation: a unitary analogue to the LR transformation - Part I. *The Computer Journal*, v. 4, p. 265-271, 1961.

GALANTI, S.; JUNG, A. Low-discrepancy sequences: Monte Carlo Simulation of Option Prices. *The Journal of Derivatives*, v. 5, p. 63-83, 1997.

GANDER, W. Zeros of Determinants of λ -Matrices. In: OLSHEVSKY, V.; TYRTYSHNIKOV, E. (eds.). *Matrix Methods: Theory, Algorithms, Applications*. Singapura: World Scientific Publishing, 2008. p. 238-246.

GARCIA, A.; LEQUAIN, Y. Elementos de Álgebra. 6ª edição. Rio de Janeiro: IMPA, 2012.

GUEDRIA, N. B.. Improved accelerated PSO algorithm for mechanical engineering optimization problems. *Applied Soft Computing*, v. 40, p. 455-467, 2016.

GENTLE, J. E. Random number generation and Monte Carlo Methods. 2ª edição. New York: Springer, 2003.

GERSCHGORIN, S. Über die Abgrenzung der Eigenwerte einer Matrix. *Bulletin de l'Académie des Sciences de l'URSS. Classe des sciences mathématiques et na.* p. 749-754, 1931.

GOLUB, G. H.; VAN DER VORST, H. A. Eigenvalue computation in the 20th century. *Journal of Computational and Applied Mathematics*, v. 123, p. 35 –65, 2000.

GOLUB, G.; VAN LOAN, C. Matrix computations. 3ª edition. London: The Johns Hopkins University Press, 1996.

HANSEN, P. B. **Householder Reduction**. Syracuse: Electrical Engineering and Computer Science, 1990. (Technical Reports, 78).

HENDERSON, N.; SACCO, W. F.; RODRIGUES JR, R. A. A Deduction of the Multicriticality Conditions of Mixtures from the Gibbs Tangent Plane Criterion. *Fluid Phase Equilibria*, v. 360, p. 305-308, 2013.

HENDERSON, N.; SÁ RÊGO, M.; SACCO, W.F.; RODRIGUES, R.A. A new look at the topographical global optimization method and its application to the phase stability analysis of mixtures. *Chemical Engineering Science*, v. 127, p. 151-174, 2015.

HENDERSON, N.; SÁ RÊGO, M.; IMBIRIBA, J. Topographical global initialization for finding all solutions of nonlinear systems with constraints. *Applied Numerical Mathematics*, v. 112, p. 155-166, 2016.

HENDERSON, N.; SÁ RÊGO, M.; IMBIRIBA, J.; DE SÁ, M.; SACCO, W.F. Testing the topographical global initialization strategy in the framework of an unconstrained optimization method. *Optimization Letters*, v. 12, p. 727-741, 2018.

HERSKOVITS, J. Feasible direction interior-point technique for nonlinear optimization.

Journal of Optimization Theory and Applications, v. 99, p. 121-146, 1998.

HOCK, W.; SCHITTKOWSKI, K. Test Examples for Nonlinear Programming Codes, Lecture Notes in Economics and Mathematical Systems. *Springer-Verlag*. Berlin, 1981.

HOFFMAN, K.; KUNZE, R. Álgebra Linear. 2ª edição. New Jersey: Prentice-Hall, 1971.

ISMAILOV, A; SOLODOV, M. V. Otimização: Métodos Computacionais. 1ª edição. Rio de Janeiro: IMPA, 2007.

ISMAILOV, A; SOLODOV, M. V. Otimização: Condições de Otimalidade, Elementos de Análise Convexa e de Dualidade. 2ª edição. Rio de Janeiro: IMPA, 2009.

JOE, S.; KUO, F. Y. Remark on Algorithm 659: Implementing Sobol's quasirandom sequence generator. *ACM Transactions on Mathematical Software*, v. 29, p. 49-57, 2003.

KARUSH, W. *Minima of Functions of Several Variables with Inequalities as Side Constraints*. Dissertação (Mestrado) - Department of Mathematics, University of Chicago, Chicago, 1939.

KUHN, H. W.; TUCKER, A. W. Nonlinear Programming. In: BERKELEY SYMPOSIUM ON MATHEMATICAL STATISTICS AND PROBABILITY, 1951, Berkeley. *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*. Berkeley: University of California Press, 1951, p. 481-492.

LAGRANGE, J. L. Mécanique analitique. 1ª edição. Paris: Desaint, 1788.

LEON, S. J. Álgebra linear com aplicações. 8ª edição. Rio de Janeiro: LTC, 2011.

LI, H.; ZOU, P.; HUANG, Z.; ZENG, C.; LIU, X. Multimodal optimization using whale optimization algorithm enhanced with local search and niching technique. *Mathematical Biosciences and Engineering*, v. 17, p. 1-27, 2020.

LIMA, E. L. Álgebra Linear. 1ª edição. Rio de Janeiro: IMPA, 2014.

LIMA, E. L. Análise Real: Funções de n variáveis. 5ª edição. Rio de Janeiro: IMPA, 2010.

LONG, W.; LIANG, X.; CAI, S.; JIAO, J.; ZHANG, W.. An improved artificial bee colony with modified augmented lagrangian for constrained optimization. *Soft Computing*, v. 22, p. 4789-4810, 2018.

LUENBERGER, D. G. Linear and nonlinear programming. 2ª edição. Massachusetts: Addison-Wesley, 1984.

MA, C. F. The semismooth and smoothing Newton methods for solving Pareto eigenvalue problem. *Applied Mathematical Modelling*, v. 36, no. 1, p. 279-287, 2012.

MATSUMOTO, L.; NISHIMURA, T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation*, v. 8, p. 3-20, 1998.

- MAZORCHE, S.R. *Algoritmos para problemas de complementaridade não linear*. 2007. Tese (Doutorado) - COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007.
- MELO, V.; CAROSIO, G.. (2013). Investigating multi-view differential evolution for solving constrained engineering design problems. *Expert Systems with Applications*, v. 40, p. 3370-3377, 2013.
- MELO, V.; IACCA, G. (2014). A modified covariance matrix adaptation evolution strategy with adaptive penalty function and restart for constrained optimization. *Expert Systems with Applications*, v. 41, p. 7077-7094, 2014.
- NOCEDAL, J.; WRIGHT, S. J. *Numerical Optimization*, 2ª edição. Springer: New York, 2006.
- PARLETT, B. N. *The Symmetric Eigenvalue Problem*. New Jersey: Prentice-Hall, 1980.
- POOLE, D. *Linear Algebra: A Modern Introduction*. 3ª edição. Canadá: Cengage Learning, 2011.
- POWELL, M. J. D. The convergence of variable metric methods for nonlinearly constrained optimization calculations. In: MANGASARIAN, O. L. et al. (eds.) *Nonlinear Programming 3*. Madison: Academic Press, 1978. p. 27-63.
- PRESS, W.; TEUKOLSKY, S.; VETTERLING, W. T.; FLANNERY, B. P. *Numerical Recipes in C: The art of scientific computing*. 2ª edição. Cambridge: Cambridge University Press, 1992.
- RAGSDELL, K. M.; PHILLIPS, D. T.. Optimal Design of a Class of Welded Structures Using Geometric Programming. *Journal of Engineering for Industry*, v. 98, p. 1021-1025, 1976.
- RAY, T.; SAINI, P.. Engineering design optimization using a swarm with an intelligent information sharing among individuals. *Journal of Engineering Optimization*, v. 33, p. 735-748, 2001.
- RUGGIERO, M. A. G.; LOPES, V. L. R. *Cálculo Numérico: Aspectos Teóricos e Computacionais*, 2ª edição. São Paulo: Pearson Universidades, 2000.
- SAAD, Y. *Numerical Methods for Large Eigenvalue Problems*. Manchester: University Press, 1992.
- SHOKRANIAN S.; SOARES M.; GODINHO, H. *Teoria dos Números*. 2ª edição. Brasília: Editora da Universidade de Brasília, 1999.
- SOBOL, I. M. The distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, v. 7, p. 86-112, 1967.
- STOER, J. BULIRSCH, R. *Introduction to Numerical Analysis*. New York: Springer, 1993.

STYAN, G. P. H. Hadamard Products and Multivariate Statistical Analysis. *Linear Algebra and its Applications*, v. 6, p. 217-240, 1973.

TÖRN, A. Clustering methods in global optimization. In: SYMPOSIUM ON STOCHASTIC CONTROL, 1986, Vilnius. *Preprints of the Second IFAC*. Vilnius: IFAC Proceedings Series, 1986. p. 138-143.

TÖRN, A. Topographical global optimization. Turku: Abo Akademi University, 1990.

TÖRN, A.; VIITANEN, S. V. Topographical global optimization. In: FLOUDAS, C. A.; PARDALOS, P. M. (eds.). *Recent Advances in Global Optimization*. Princeton: Princeton University Press, 1992. p. 384-398.

WATKINS, D. S. Fundamentals of matrix computations. 3^a edição. New York: John Wiley & Sons, 2010.

WILKINSON, J.H. The Algebraic Eigenvalue Problem. Oxford: Oxford University Press, 1965.

WILKINSON, J. H. Rounding errors in algebraic processes. New York: Dover Publications, 1994.

ZHU, H.; HU, Y.; ZHU, W.. A dynamic adaptive particle swarm optimization and genetic algorithm for different constrained engineering design optimization problems. *Advances in Nonlinear Dynamics and Vibrations on Mechanical Systems*, v. 11, p. 1-27, 2019.

APÊNDICE A - Matrizes para o problema de autovalor complementar

Problema 1

$$A = \begin{bmatrix} 8 & -1 \\ 3 & 4 \end{bmatrix}.$$

Problema 2

$$A = \begin{bmatrix} 8 & -1 & 4 \\ 3 & 4 & 0,5 \\ 2 & -0,5 & 6 \end{bmatrix}.$$

Problema 3

$$A = \begin{bmatrix} s^2 & s^3 & s^4 \\ -s^3 & s^4 & s^5 \\ -s^4 & s^5 & s^6 \end{bmatrix},$$

onde $s = \sqrt{6}$.

Problema 4

$$A = \begin{bmatrix} 4 & 8 & 16 \\ 8 & 16 & 32 \\ 16 & 32 & 64 \end{bmatrix}.$$

Problema 5

$$A = \begin{bmatrix} 100 & 106 & -18 & -81 \\ 92 & 158 & -24 & -101 \\ 2 & 44 & 37 & -7 \\ 21 & 38 & 0 & 2 \end{bmatrix}.$$

Problema 6

$$A = \begin{bmatrix} s^2 & s^3 & s^4 & s^5 \\ -s^3 & s^4 & s^5 & s^6 \\ -s^4 & s^5 & s^6 & s^7 \\ -s^5 & s^6 & s^7 & s^8 \end{bmatrix},$$

onde $s = \sqrt{6}$.

Problema 7

$$A = \begin{bmatrix} 4 & 8 & 16 & 32 \\ 8 & 16 & 32 & 64 \\ 16 & 32 & 64 & 128 \\ 32 & 64 & 128 & 256 \end{bmatrix}.$$

Problema 8

$$A = \begin{bmatrix} s^2 & s^3 & s^4 & s^5 & s^6 \\ -s^3 & s^4 & s^5 & s^6 & s^7 \\ -s^4 & s^5 & s^6 & s^7 & s^8 \\ -s^5 & s^6 & s^7 & s^8 & s^9 \\ -s^6 & s^7 & s^8 & s^9 & s^{10} \end{bmatrix},$$

onde $s = \sqrt{6}$.

Problema 9

$$A = \begin{bmatrix} 4 & 8 & 16 & 32 & 64 \\ 8 & 16 & 32 & 64 & 128 \\ 16 & 32 & 64 & 128 & 256 \\ 32 & 64 & 128 & 256 & 512 \\ 64 & 128 & 256 & 512 & 1024 \end{bmatrix}.$$

Problema 10

$$A = \begin{bmatrix} 34 & -61 & 58 \\ 30 & -63 & 10 \\ 98 & -83 & 45 \end{bmatrix}.$$

Problema 11

$$A = \begin{bmatrix} 34 & -61 & 58 & 58 \\ 30 & -63 & 10 & 9 \\ 98 & -83 & 45 & 74 \\ 99 & -84 & 46 & 44 \end{bmatrix}.$$

Problema 12

$$A = \begin{bmatrix} -788 & 780 & 256 & -156 & -191 \\ 548 & -862 & 190 & -112 & -143 \\ 456 & 548 & -1308 & -110 & -119 \\ 292 & 374 & 14 & -1402 & -28 \\ 304 & 402 & 66 & -38 & -1522 \end{bmatrix}.$$

APÊNDICE B - Problemas de otimização com restrições mistas

Problema 1

$$\left\{ \begin{array}{l} \zeta(y) = 3y_1 + 0,000001y_1^3 + 2y_2 + (0,000002/3)y_2^3 \\ r_1(y) = -y_4 + y_3 - 0,55 \leq 0, \\ r_2(y) = -y_3 + y_4 - 0,55 \leq 0, \\ s_1(y) = 1000\text{sen}(-y_3 - 0,25) + 1000\text{sen}(-y_4 - 0,25) + 894,8 - y_1 = 0, \\ s_2(y) = 1000\text{sen}(y_3 - 0,25) + 1000\text{sen}(y_3 - y_4 - 0,25) + 894,8 - y_2 = 0, \\ s_3(y) = 1000\text{sen}(y_4 - 0,25) + 1000\text{sen}(y_4 - y_3 - 0,25) + 1294,8 = 0, \\ 0 \leq y_1, y_2 \leq 1200; -0,55 \leq y_3, y_4 \leq 0,55. \end{array} \right.$$

Problema 2

$$\left\{ \begin{array}{l} \zeta(y) = -9y_5 - 15y_8 + 6y_1 + 16y_2 + 10(y_6 + y_7) \\ r_1(y) = y_9y_3 + 0.02y_6 - 0.025y_5 \leq 0, \\ r_2(y) = y_9y_4 + 0.02y_7 - 0.015y_8 \leq 0, \\ s_1(y) = y_1 + y_2 - y_3 - y_4 = 0, \\ s_2(y) = 0,03y_1 + 0,01y_2 - y_9(y_3 + y_4) = 0, \\ s_3(y) = y_3 + y_6 - y_5 = 0, \\ s_4(y) = y_4 + y_7 - y_8 = 0, \\ 0 \leq y_1, y_2, y_6 \leq 300; 0 \leq y_4, y_8 \leq 200; \\ 0 \leq y_3, y_5, y_7 \leq 100; 0,01 \leq y_9 \leq 0,03. \end{array} \right.$$

Problema 3

$$\left\{ \begin{array}{l} \zeta(y) = (y_1 - 2)^2 + (y_2 - 1)^2 \\ r_1(y) = 0,25y_1^2 + y_2^2 - 1 \leq 0, \\ s_1(y) = y_1 - 2y_2 + 1 = 0, \\ -10 \leq y_1, y_2 \leq 10 \end{array} \right.$$

Problema 4

$$\left\{ \begin{array}{l} \zeta(y) = (y_1 + 3y_2 + y_3)^2 + 4(y_1 - y_2)^2 \\ r_1(y) = 3 + y_1^3 - 6y_2 - 4y_3 \leq 0, \\ s_1(y) = 1 - y_1 - y_2 - y_3 = 0, \\ 0 \leq y_1, y_2, y_3 \leq 1 \end{array} \right.$$

Problema 5

$$\left\{ \begin{array}{l} \zeta(y) = y_1 y_4 (y_1 + y_2 + y_3) + y_3 \\ r_1(y) = -y_1 y_2 y_3 y_4 + 25 \leq 0, \\ s_1(y) = y_1^2 + y_2^2 + y_3^2 + y_4^2 - 40 = 0, \\ 1 \leq y_1, y_2, y_3, y_4 \leq 5 \end{array} \right.$$

Problema 6

$$\left\{ \begin{array}{l} \zeta(y) = 24,55y_1 + 26,75y_2 + 39y_3 + 40,5y_4 \\ r_1(y) = -2,3y_1 - 5,6y_2 - 11,1y_3 - 1,3y_4 + 5 \leq 0, \\ r_2(y) = 1,645\sqrt{0,28y_1^2 + 0,19y_2^2 + 20,5y_3^2 + 0,62y_4^2} - 12y_1 - 11,9y_2 - 41,8y_3 - \\ \hspace{15em} 52,1y_4 + 21 \leq 0, \\ s_1(y) = y_1 + y_2 + y_3 + y_4 - 1 = 0, \\ 0 \leq y_1, y_2, y_3, y_4 \leq 1 \end{array} \right.$$

Problema 7

$$\left\{ \begin{array}{l} \zeta(y) = 5,04y_1 + 0,035y_2 + 10y_3 + 3,36y_5 - 0,063y_4y_7 \\ r_1(y) = -35,82 + 0,222y_{10} + by_9 \leq 0, \\ r_2(y) = 133 - 3y_7 + ay_{10} \leq 0, \\ r_3(y) = 35,82 - 0,222y_{10} - by_9 - y_9(1/b - b) \leq 0, \\ r_4(y) = -133 + 3y_7 - ay_{10} - y_{10}(1/a - a) \leq 0, \\ r_5(y) = -1,12y_1 - 0,13167y_1y_8 + 0,00667y_1y_8^2 + ay_4 \leq 0, \\ r_6(y) = -57,425 - 1,098y_8 + 0,038y_8^2 - 0,325y_6 + ay_7 \leq 0, \\ r_7(y) = 1,12y_1 + 0,13167y_1y_8 - 0,00667y_1y_8^2 - ay_4 - y_4(1/a - a) \leq 0, \\ r_8(y) = 57,425 + 1,098y_8 - 0,038y_8^2 + 0,325y_6 - ay_7 - y_7(1/a - a) \leq 0, \\ s_1(y) = 1,22y_4 - y_1 - y_5 \leq 0, \\ s_2(y) = (98000y_3)/(y_4y_9 + 1000y_3) - y_6 \leq 0, \\ s_3(y) = (y_2 + y_5)/y_1 - y_8 \leq 0, \\ 0,00001 \leq y_1 \leq 2000; 0,00001 \leq y_2 \leq 16000; \\ 0,00001 \leq y_3 \leq 120; 0,00001 \leq y_4 \leq 5000; \\ 0,00001 \leq y_5 \leq 2000; 85 \leq y_6 \leq 93; \\ 90 \leq y_7 \leq 95; 3 \leq y_8 \leq 12; \\ 1,2 \leq y_9 \leq 4; 145 \leq y_{10} \leq 162; \\ a = 0,99; b = 0,9. \end{array} \right.$$

Problema 8

$$\left\{ \begin{array}{l} \zeta(y) = 3y_1 + 0,000001y_1^3 + 2y_2 + (0,000002/3)y_2^3 \\ r_1(y) = -y_4 + y_3 - 0,55 \leq 0, \\ r_2(y) = -y_3 + y_4 - 0,55 \leq 0, \\ s_1(y) = 1000\text{sen}(-y_3 - 0,25) + 1000\text{sen}(-y_4 - 0,25) + 894,8 - y_1 = 0, \\ s_2(y) = 1000\text{sen}(y_3 - 0,25) + 1000\text{sen}(y_3 - y_4 - 0,25) + 894,8 - y_2 = 0, \\ s_3(y) = 1000\text{sen}(y_4 - 0,25) + 1000\text{sen}(y_4 - y_3 - 0,25) + 1294,8 = 0, \\ 0 \leq y_1, y_2 \leq 1200; -0,55 \leq y_3, y_4 \leq 0,55. \end{array} \right.$$

Problema 9

$$\left\{ \begin{array}{l} \zeta(y) = -y_1y_2y_3 \\ r_1(y) = y_1^2 + 2y_2^2 + 4y_3^2 - 48 \leq 0, \\ s_1(y) = -y_2y_3 + 2y_1y_4 = 0, \\ s_2(y) = -y_1y_3 + 4y_2y_4 = 0, \\ s_3(y) = -y_1y_2 + 8y_3y_4 = 0, \\ s_4(y) = y_4(y_1^2 + 2y_2^2 + 4y_3^2 - 48) = 0, \\ 0 \leq y_1, y_4 \leq 5; -4 \leq y_2 \leq 4; -3 \leq y_3 \leq 3. \end{array} \right.$$

Problema 10

$$\left\{ \begin{array}{l} \zeta(y) = a(y_2 - by_1^2 + cy_1 - r)^2 + s(1-t)\cos(y_1) + s \\ r_1(y) = y_1y_2 - 23.5 \leq 0, \\ r_2(y) = y_1 + y_2 - 15 \leq 0, \\ s_1(y) = 2a(y_2 - by_1^2 + cy_1 - r)(2by_1 + c) - s(1-t)\text{sen}(y_1) + y_3y_2 + y_4 = 0, \\ s_2(y) = 2a(y_2 - by_1^2 + cy_1 - r) + y_3y_1 + y_4 = 0, \\ s_3(y) = y_3(y_1y_2 - 23.5) = 0, \\ s_4(y) = y_4(y_1 + y_2 - 15) = 0, \\ -4 \leq y_1 \leq 10; 1 \leq y_2 \leq 13; 0 \leq y_3, y_4 \leq 5. \\ a = 1; b = 5.1/(4\pi^2); c = 5/\pi; r = 6; s = 10; t = 1/(8\pi). \end{array} \right.$$

Problema 11

$$\left\{ \begin{array}{l} \zeta(y) = (4 - 2, 1y_1^2 + y_1^4/3) y_1^2 + y_1 y_2 + (4y_2^2 - 4) y_2^2 \\ r_1(y) = y_1 y_2^3 \leq 0, \\ r_2(y) = y_1^3 - y_2^2 \leq 0, \\ r_3(y) = y_1 + y_2^2 + 2y_2 - 3 \leq 0, \\ s_1(y) = 8y_1 - 8.4y_1^3 + 2y_1^5 + y_2 + y_3 y_2^3 + 3y_4 y_1^2 + y_5 = 0, \\ s_2(y) = y_1 - 8y_2 + 16y_2^3 + 3y_1 y_2^2 y_3 - 2y_2 y_4 + y_5 (2y_2 + 2) = 0, \\ s_3(y) = y_3 (y_1 y_2^3) = 0, \\ s_4(y) = y_4 (y_1^3 - y_2^2) = 0, \\ s_5(y) = y_5 (y_1 + y_2^2 + 2y_2 - 3) = 0, \\ -3 \leq y_1 \leq 3; -2 \leq y_2 \leq 2; 0 \leq y_3, y_4, y_5 \leq 5. \end{array} \right.$$

Problema 12

$$\left\{ \begin{array}{l} \zeta(y) = y_1^2 + y_2^2 + y_3^2 + y_4^2 \\ r_1(y) = -y_1^2 - 4y_2^2 - 9y_3^2 - 16y_4^2 + 16 \leq 0, \\ s_1(y) = 2y_1 - 2y_1 y_5 = 0, \\ s_2(y) = 2y_2 - 8y_2 y_5 = 0, \\ s_3(y) = 2y_3 - 18y_3 y_5 = 0, \\ s_4(y) = 2y_4 - 32y_4 y_5 = 0, \\ s_5(y) = y_5 (-y_1^2 - 4y_2^2 - 9y_3^2 - 16y_4^2 + 16) = 0, \\ -5 \leq y_1, y_2, y_3, y_4 \leq 5; 0 \leq y_5 \leq 2. \end{array} \right.$$

Problema 13

$$\left\{ \begin{array}{l} \zeta(y) = 1 \\ r_1(y) = -(100 - y_5) y_1 - 106y_2 + 18y_3 + 81y_4 \leq 0, \\ r_2(y) = -92y_1 - (158 - y_5) y_2 + 24y_3 + 101y_4 \leq 0, \\ r_3(y) = -2y_1 - 44y_2 - (37 - y_5) y_3 + 7y_4 \leq 0, \\ r_4(y) = -21y_1 - 38y_2 - (2 - y_5) y_4 \leq 0, \\ s_1(y) = -(100 - y_5) y_1^2 - 106y_1 y_2 + 18y_1 y_3 + 81y_1 y_4 = 0, \\ s_2(y) = -92y_1 y_2 - (158 - y_5) y_2^2 + 24y_2 y_3 + 101y_2 y_4 = 0, \\ s_3(y) = -2y_1 y_3 - 44y_2 y_3 - (37 - y_5) y_3^2 + 7y_3 y_4 = 0, \\ s_4(y) = -21y_1 y_4 - 38y_2 y_4 - (2 - y_5) y_4^2 = 0, \\ s_5(y) = y_1 + y_2 + y_3 + y_4 - 1 = 0, \\ 0 \leq y_1, y_2, y_3, y_4 \leq 1; -346 \leq y_5 \leq 346. \end{array} \right.$$

Problema 14

$$\left\{ \begin{array}{l} \zeta(y) = 1 \\ r_1(y) = -(8 - y_4)y_1 + y_2 - 4y_3 \leq 0, \\ r_2(y) = -3y_1 - (4 - y_4)y_2 - 0, 5y_3 \leq 0, \\ r_3(y) = -2y_1 + 0, 5y_2 - (6 - y_4)y_3 \leq 0, \\ s_1(y) = -(8 - y_4)y_1^2 + y_1y_2 - 4y_1y_3 = 0, \\ s_2(y) = -3y_1y_2 - (4 - y_4)y_2^2 - 0, 5y_2y_3 = 0, \\ s_3(y) = -2y_1y_3 + 0, 5y_2y_3 - (6 - y_4)y_3^2 = 0, \\ s_4(y) = y_1 + y_2 + y_3 - 1 = 0, \\ 0 \leq y_1, y_2, y_3 \leq 1; -13 \leq y_4 \leq 13. \end{array} \right.$$

Problema 15

$$\left\{ \begin{array}{l} \zeta(y) = 1 \\ r_1(y) = -8y_1 + y_1y_3 + y_2 \leq 0, \\ r_2(y) = -3y_1 - 4y_2 + y_2y_3 \leq 0, \\ s_1(y) = -8y_1^2 + y_1^2y_3 + y_1y_2 = 0, \\ s_2(y) = -3y_1y_2 - 4y_2^2 + y_2^2y_3 = 0, \\ s_3(y) = y_1 + y_2 - 1 = 0, \\ 0 \leq y_1, y_2 \leq 1; -9 \leq y_3 \leq 9. \end{array} \right.$$

Problema 16

Esse problema da engenharia, presente nos projetos de vigas soldadas, é frequentemente usado para testar a eficiência de diferentes métodos de otimização. O objetivo é encontrar o custo mínimo para fabricação da viga soldada sujeita às restrições de tensão de cisalhamento, tensão de flexão, carga de flambagem, deflexão final e restrição lateral (RAGSDELL e PHILLIPS, 1976). As funções desse problema são dadas a seguir.

$$\left\{ \begin{array}{l} \zeta(y) = 1, 10471y_1^2y_2 + 0, 04811y_3y_4(14 + y_2) \\ r_1(y) = \tau - t_{max} \leq 0, \\ r_2(y) = \sigma - s_{max} \leq 0, \\ r_3(y) = y_1 - y_4 \leq 0, \\ r_4(y) = 0, 10471y_1^2 + 0, 04811y_3y_4(14 + y_2) - 5 \leq 0, \\ r_5(y) = \delta - d_{max} \leq 0, \\ r_6(y) = P - P_c \leq 0, \\ 0, 125 \leq y_1 \leq 2; 0, 1 \leq y_2, y_3 \leq 10; 0, 1 \leq y_4 \leq 2. \end{array} \right.$$

onde

$$M = P \left(L + \frac{y_2}{2} \right); R = \sqrt{0, 25 \left(y_2^2 + (y_1 + y_3)^2 \right)}; J = 2\sqrt{2}y_1y_2 \left[\frac{y_2^2}{12} + 0, 25 \left(y_1 + y_3 \right)^2 \right]; \\ P = 6000; L = 14; P_c = \left(\frac{4, 013E}{6 * L * L} \right) y_3y_4^3 \left(1 - \frac{y_3\sqrt{E/H}}{4L} \right); \sigma = \frac{6PL}{y_4y_3^2}; t_1 = \frac{P}{\sqrt{2}y_1y_2}; t_2 = \frac{MR}{J};$$

$$\tau = \sqrt{t_1^2 + \frac{t_1 t_2 y_2}{R} + t_2^2}; \delta = \frac{4PL^3}{Ey_4 y_3^3}; E = 30000000; t_{max} = 13600; s_{max} = 30000; d_{max} = 0,25 \text{ e } H = 12000000.$$

Problema 17

Esse problema da engenharia, consiste em minimizar o peso de uma mola de tração ou compressão, sujeita a restrições de tensão de cisalhamento, frequência de surto, deflexão mínima e do diâmetro externo (GUEDRIA, 2016).

$$\left\{ \begin{array}{l} \zeta(y) = y_1^2 y_2 (y_3 + 2) \\ r_1(y) = 1 - \frac{y_2^3 y_3}{71785 y_1^4} \leq 0, \\ r_2(y) = \frac{4y_2^2 - y_1 y_2}{12566 y_1^3 y_2 - y_1^4} + \frac{1}{5108 y_1^2} - 1 \leq 0, \\ r_3(y) = 1 - \frac{140,54 y_1}{y_3 y_2^2} \leq 0, \\ r_4(y) = \frac{y_1 + y_2}{1,5} - 1 \leq 0, \\ 0,05 \leq y_1 \leq 0,2; 0,25 \leq y_2 \leq 1,3; 0,2 \leq y_3 \leq 15. \end{array} \right.$$

Problema 18

Nesse problema o objetivo é minimizar o volume da estrutura de um treliça de três barras, sujeita a restrições de tensão (RAY e SAINI, 2001). A formulação matemática é apresentada a seguir.

$$\left\{ \begin{array}{l} \zeta(y) = 200\sqrt{2}y_1 + 100y_2 \\ r_1(y) = 2 \frac{\sqrt{2}y_1 + y_2}{\sqrt{2}y_1^2 + 2y_1 y_2} - 2 \leq 0, \\ r_2(y) = \frac{2y_2}{\sqrt{2}y_1^2 + 2y_1 y_2} - 2 \leq 0, \\ r_3(y) = \frac{2}{\sqrt{2}y_2 + y_1} - 2 \leq 0, \\ 0 \leq y_1, y_2, y_3 \leq 1. \end{array} \right.$$