

Inteligência Computacional e Otimização - ICO

Trabalho/Projeto 1 – ano letivo 2023/2024

Cada grupo (3-5 elementos, preferencialmente 4 elementos) escolhe uma das 4 propostas de trabalho/projeto abaixo expostas.

As 4 propostas têm em comum:

- **Submissão**, no local do Moodle criado para tal, **por apenas um elemento do grupo**, que deve ser o mesmo em todas as entregas; **todos os ficheiros** devem conter o **nome de todos** os elementos do grupo (exemplo: Proposta1_AntonioDias_MariaSilva_...)
- **1ª entrega até final do dia 11 de março** (2ª feira), com **peso de 15%**, dos elementos:
 - composição do grupo e proposta escolhida
 - ficheiro de 5-7 páginas (número indicativo, não obrigatório) escritas em word ou latex, com descrição sumária do trabalho que vão desenvolver (estrutura, metodologia, consulta inicial, bibliografia, etc...)
- **2ª entrega até final do dia 01 de abril** (2ª feira), com **peso de 70%**, dos elementos:
 - ficheiro de 17-23 páginas (sem contar com apêndice(s)) escritos em word ou latex, de relatório do trabalho desenvolvido; pode incorporar conteúdos do ficheiro da 1ª entrega; o(s) apêndice(s) podem conter figuras ou código;
 - ficheiros de código que conduziram ao exposto no ficheiro de relatório.
- **Apresentação e discussão no dia 02 março (18h-21h)** (3ª feira), com **peso de 15%**,: 10-15 minutos para exposição do trabalho desenvolvido, seguida de 5-10 minutos para discussão e/ou colocação de questões.

Podem incluir abordagem analítica quando tal for esclarecedor e enriquecedor do trabalho.

Para cada proposta existe uma pasta de elementos que podem consultar, a qual não é exaustiva.

Proposta 1.

Trabalho sobre **otimização através dos métodos de busca em linha** (no inglês, *line-search methods*).

O que se pretende: exploração de alguns métodos de busca em linha, para diferentes funções, e comparação da performance dos métodos, com e sem as condições de Wolfe, tendo como ponto de partida os slides disponíveis na pasta “Métodos de busca em linha” (Moodle), alguns elementos

de consulta (Notas+Referências, ver abaixo) e os tópicos de exploração de alguns métodos para uma lista de funções (ver abaixo).

Devem iniciar o trabalho com a exploração dos 11 ficheiros MATLAB (*descidaMaximaV1aula.m*, *descidaMaximaWolfe.m*, *Newton2015.m*, entre outros) disponíveis no Moodle, que implementam alguns dos métodos *line-search*. De seguida, devem avancar para outros ficheiros (em MATLAB e/ou em Python) e/ou outras funções e/ou outros métodos *line-search*, sempre numa perspetiva crítica de comparação e de deteção das vantagens e desvantagens de cada método (em termos, por exemplo, do tipo de convergência e da taxa de convergência). Podem ainda apresentar comparação dos métodos *line-search* com outros métodos de otimização.

Lista de funções:

$$r(x, y) = 100(y - x^2)^2 + (1 - x)^2 \quad (\text{função de Rosenbrock})$$

$$f(x, y) = (x^2 - 100)(x^2 - 61) + (y^2 - 25)(y^2 - 16) - 15 \cos\left(x - \frac{7}{5}\right) \cos\left(y - \frac{9}{2}\right)$$

$$g(x, y) = x(x + 3)(x - 3)(x - 5) + y(y + 3)(y - 3)(y - 5)$$

Tópicos de exploração: Considere as funções acima listadas e os m.file *descidaMaximaV1aula.m*, *descidaMaximaWolfe.m* e *Newton2015.m*. Estes implementam o Método de Descida Máxima e o Método de Newton, com escolha pseudoaleatória de pontos iniciais, para determinação de valores aproximados dos minimizantes e dos mínimos locais da função f .

1. Altere os m.files acima referidos de modo a implementar os dois métodos às diferentes funções com:
 - a. tolerância do erro igual a 0.15,
 - b. um número máximo de iterações escolhido entre 35 e 50 e
 - c. um número de pontos iniciais escolhido entre 8 e 13 e a tomar valores no intervalo $(x, y) \in [-11, 7] \times [-11, 7]$.

Para permitir a comparação de *outputs* pedida em 3., escolha os mesmos valores nos vários ficheiros e para as várias funções.

2. Apresente os *outputs* obtidos para estes problemas de minimização, devidamente identificados (mínimos locais, minimizantes locais, vetor gradiente, etc.).
3. Analise e interprete os *outputs* obtidos, fazendo a comparação:
 - a. do vetor gradiente nos minimizantes
 - b. do número de iterações por ponto inicial e o número médio de iterações, face ao objetivo das condições de Wolfe
 - c. de outros elementos que considere relevantes.

Notas:

- em “Complemento aos slides das aulas (Line-search methods).pdf” encontram informação adicional aos métodos de busca em linha

- em “Exercícios Propostos.pdf” (disponível no Moodle) encontram 7 exercícios que podem ser tomados como indicadores de exploração dos métodos de busca em linha (páginas 3 e 4);
- as condições de Wolfe são um elemento importante nos métodos de busca em linha, que devem ter em conta nesta proposta de trabalho; encontram no Moodle o ficheiro “Condições de Wolfe.pdf” de explicação detalhada destas condições (para além dos slides).

Referências: Podem consultar as referências seguintes na pasta “Apoio à Proposta 1” (Moodle):

- Link <https://github.com/topics/line-search>
- **(pp. 34–63)** Jorge Nocedal and Stephen J. Wright (1999) Numerical Optimization, Springer
- **(pp. 111–154)** J.E. Dennis, Jr. and Robert B. Schnabel (1996) Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice-Hall
- **(pp. 71–117)** Wenyu Sun and Ya-Xiang Yuan (2006) Optimization Theory and Methods: Nonlinear Programming, Springer
- E. Bergou, Y. Diouane, V. Kunc, V. Kungurtsev and C. W. Royer (2021) A Subsampling Line-Search Method with Second-Order Results (arxiv)
- Zhensheng Yu and Ji Lin, General Form of a Nonmonotone Line Search Technique for Unconstrained Optimization, Algorithmic Operations Research Vol.4 (2009) 70–75
- Maren Mahsereci and Philipp Hennig, Probabilistic Line Searches for Stochastic Optimization, Journal of Machine Learning Research 18 (2017) 1-59
- Leidy Yissedt Lara Díaz, Um Estudo sobre o Método de Gradientes Conjugados para Minimização Irrestrita, tese de mestrado (2017)
- **(Cap. 11 + 12)** (Lab Manual) Labs for Foundations of Applied Mathematics - Algorithm Design and Optimization (Vol. 2), Jeffrey Humpherys and Tyler J. Jarvis (Eds.)
- **(Cap. 3 + 4)** Bruna Alves Ferraz, Métodos Computacionais de Otimização, tese de mestrado (2017)
- 2019_slides_Daniel P. Robinson_Line-Search Methods for Smooth Unconstrained Optimization
- 2011_slides_Enrico Bertolazzi_Unconstrained Minimization
- **(Cap. 3)** Argimiro R. Secchi (2015) Otimização de Processos, sebenta do COQ-897
- **(Cap. 3)** Elizabeth Christine Marins Valente, Otimização em Problemas de Engenharia Civil, tese de mestrado Univ. Porto (2020)
- **(pp.14-40 Chapter 4)** Alan R. Parkinson, Richard J. Balling and John D. Hedengren (2013) Optimization Methods for Engineering Design - Applications and Theory
- Ed Overman (2018) A MATLAB Tutorial

Proposta 2.

Trabalho sobre **otimização através de redes complexas**, com programação em Python (ou em MATLAB).

O que se pretende: descrição e exploração de alguns problemas de otimização cuja resolução beneficia da modelação/estruturação através de redes complexas, tendo como ponto de partida os slides com iniciação à teoria de grafos e a este “tipo” de problemas de fluxos em rede (disponíveis no Moodle); são ainda sugeridos alguns elementos de consulta (ver Referências abaixo).

Devem fazer um resumo dos “clássicos” problemas de otimização que são colocados em rede complexa (problemas de roteamento: Problema do caixeiro viajante e Problema do carteiro chinês), e avançar para a utilização das redes complexas em problemas de otimização menos convencionais e/ou extensões dos problemas “clássicos”, sempre numa perspetiva de análise das vantagens da

estruturação em rede. Podem ainda comparar a resolução através das redes complexas com outras formas de abordar esses problemas de otimização.

Referências: Podem consultar as referências da pasta “Apoio à Proposta 2” (Moodle).

Proposta 3.

Trabalho em **otimização/programação linear**, com uso de código em Python (ou em MATLAB).

O que se pretende: apresentação/descrição e exploração de alguns problemas de otimização linear, após um resumo dos elementos teóricos de estruturação, resolução e análise subjacentes a este “tipo” de otimização (descrição matricial, método Simplex e extensão big-M, análise de sensibilidade, problema dual, etc.); são sugeridos alguns elementos de consulta (Nota+Referências, ver abaixo). Devem incluir os problemas de afetação e de transporte como casos particulares, e avançar para a programação por metas (*goal programming*) e/ou outras “extensões” da otimização linear. Podem ainda apresentar comparação entre diferentes formas de estruturar um problema de otimização linear.

Nota: podem consultar os elementos da pasta “Otimização/Programação Linear (Moodle).

Referências: Podem consultar as referências seguintes na pasta “Apoio à Proposta 3” (Moodle).

Proposta 4.

Trabalho em **otimização/análise multicritério**, preferencialmente com uso de código em Python.

O que se pretende: apresentação/descrição e exploração de alguns métodos de otimização multicritério, tendo como ponto de partida os slides disponíveis na pasta “Otimização/Análise multicritério” (Moodle), relativos ao *Analytic Hierarchy Process* (AHP) e ao MACBETH (*Measuring Attractiveness by a Categorical Based. Evaluation TechNique*); são sugeridos alguns elementos de consulta (Notas+Referências, ver abaixo).

Devem fazer um resumo dos elementos comuns a qualquer método de análise multicritério (fazendo a sua identificação em alguns métodos), e avançar para a resolução de um problema multicritério concreto através de dois métodos multicritério (sejam eles os já referidos ou outros), o que permite indicar vantagens e desvantagens entre os métodos. Podem ainda apresentar comparação da otimização multicritério com outra(s) metodologia(s) de otimização.

Notas:

- a versão free do MACBETH pode ser obtida em <https://m-macbeth.com/demo/> (que contém vários tutoriais de utilização) mas apenas permite salvar quando as matrizes envolvidas não envolvem mais do que 5 linhas (ou colunas), o que não invalida a sua utilização neste trabalho.
- a pasta acima indicada contém um ficheiro Excel que podem usar para bom entendimento do AHP.

Referências: Podem consultar as referências seguintes na pasta “Apoio à Proposta 4” (Moodle).

BOM TRABALHO 😊