Lab7D (5 p, pytanie 1p) 14.01.2022

Termin odesłania **21.01.2022 godz. 16.15** na platformie **Ms Teams** (we właściwym zespole **lab** przypisanym dla przedmiotu **Programowanie Matematyczne**). **Opóźnione** przesłanie rozwiązania zadania będzie rozliczane zgodnie z regulaminem przedmiotu.

Rozwiązanie zadania tj. wszystkie źródłowe **m-pliki, raport** (*obowiązkowy, zawierający oświadczenie o samodzielności*) w formacie **zip** o nazwie `pm7d_swojeimie_swojenazwisko.zip`

Raport (plik **pdf**) powinno być w formacie **A4** i powinno obejmować:
Dane studenta (imię, nazwisko, grupa, data)
Treść zadania (postać rozwiązywanego problemu)
Opis kroków przekształcania zadania, krótki opis algorytmu
Ciekawe przykłady obliczeniowe (również dodatkowo wskazane w treści zadania)
Analizę (omówienie) wyników obliczeniowych, testów
Ponadto należy załączyć:
Kody źródłowe wszystkich funkcji/procedur i skryptów (**brak** kompletu jest traktowany jak **brak** przesłania zadania w terminie)

Napisz **skrypt**, w którym proszę wykonać całe zadanie i wywołać odpowiednie funkcje.

Plik **`dane.m`** tworzy tablicę ***T*** losowo wygenerowanego **wielokąta wypukłego** (w R^2), kolejne **kolumny** tablicy zawierają współrzędne wygenerowanych wierzchołków (*możesz zaproponować też inne własne generowanie?*).

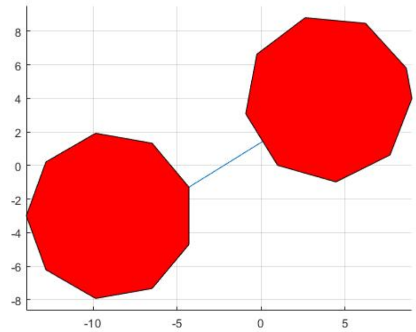
Problem
Rozważmy dwa zbiory punktów $P = \{p_1, \dots, p_r\}$ oraz $Q = \{q_1, \dots, q_s\}$ definiujące wierzchołki **dwóch wypukłych *rozłącznych* wielokątów** (w R^2, $r + s = m$)
Znaleźć minimalną odległość wielokątów, tzn. min $\|p - q\|$, gdzie $p = \sum_{i=1}^{r} \lambda_i p_i$ oraz $q = \sum_{i=1}^{s} \mu_i q_i$
($\lambda_i, \mu_i \in [0,1]$, $\sum \lambda_i = 1$, $\sum \mu_i = 1$)

Problem można sprowadzić do rozwiązania **zadania programowania kwadratowego**:

$$\min_{x\in\Omega}(x^T D x)$$

$$\Omega: \begin{cases} \sum_{i=1}^{r} x_i = 1 \\ \sum_{i=r+1}^{m} x_i = 1 \\ x \geq 0 \end{cases}$$

gdzie $\boldsymbol{D} = \boldsymbol{C}^T\boldsymbol{C}$, $\boldsymbol{C} =?$



1 pkt
Jak zdefiniować ***C*** wykorzystując ***P*** oraz ***Q***? Opisz to w raporcie.

Zdefiniuj odpowiednie macierze dla **funkcji celu** oraz dla **ograniczeń**.

Rozwiązać problem za pomocą funkcji `quadprog` dla ***N=100*** wygenerowanych **par rozłącznych wielokątów** (punkt zaczepienia wielokąta losowy)
W **`optimoptions`** ustaw:
`ConstraintTolerance: 1.0000e-10`
`OptimalityTolerance: 1.0000e-10`

Gdy **`exitflag`** jest równe **1**, to podaj znaleziony wektor **x**, znalezioną wartość funkcji, ponadto podaj *mnożniki Lagrange'a* **`lambda`** (*dla odpowiednich ograniczeń*).

Narysuj oba wielokąty (np. funkcja **`fill`**) oraz linię łączącą punkty **`p`** oraz **`q`** (np. funkcja **`line`**)

1 pkt (w raporcie)
Omów **własności** rozwiązywanego zadania, czy zawsze jest RO? Jednoznaczne?
Podaj **funkcję Lagrange'a** zadania.
Podaj komplet **WKT** zadania (*zapisane macierzowo*).

3 pkt
Proszę rozwiązać **problem** *(i porównać z powyższymi wynikami* ***quadprog****-a)* za pomocą **własnej funkcji** wykorzystującej **algorytm punktu wewnętrznego** ***IPM*** dla odpowiedniego **zadania kwadratowego**

W raporcie (***obowiązkowo***) należy podać **opis algorytmu** wykorzystanego **w swojej implementacji** oraz **uzasadnić** wszystkie podejmowane **kroki**.
Uzasadnij, jak wyznacza się **kierunek**, jak wyznacza się **krok** oraz **kluczowe parametry** (np. punkt startowy, warunek stopu, itp).

[RO, f_opt, exitflag, it, LL_eqlin, LL_lower]=IPM(parametry);
e=? parametr definiujący dokładność obliczeń (zbieżność, itp.), np**. e=1e-6?**
x0 – punkt startowy
MAX_IT = ?
M = ?

it – liczba iteracji
LL – obliczone **mnożniki Lagrange'a** (*które zmienne w algorytmie IPM stanowią mnożniki Lagrange'a?*)

Jeśli istnieje rozwiązanie zadania (**exitflag=1**), to zbadaj skuteczność algorytmu (*może wykres?):*
Zbadaj **norm(fval-f_opt)** oraz **norm(x-RO)**

Jeśli istnieje rozwiązanie zadania (**exitflag=1**), to wyświetl **mnożniki Lagrange'a** (oraz **porównaj** je z uzyskanymi z **quadprog**) Czy są spełnione **WKT**?

TESTY

- Należy przeprowadzić **testy algorytmu** dla ***N=100*** różnych wygenerowanych danych w celu zbadania **liczby iteracji**, której wymaga algorytm, by uzyskać rozwiązanie z podaną dokładnością.
- Zaproponuj własny sposób generowania **wypukłych wielokątów** (*ile mają wierzchołków? 20? 50?)*
- Oczekiwana jest **wysoka skuteczność** własnej implementacji IPM (dopracuj parametry algorytmu)

Opis testów
Wnioski