

Iterativne metode v numerični linearni algebri 2013/2014

2. domača naloga

Rešitve stisnite v ZIP datoteko z imenom `ime-priimek-vpisna-2.zip` in jih oddajte preko spletne učilnice (<http://ucilnica.fmf.uni-lj.si>) najkasneje do 26. januarja 2014, do 12. ure. Priložite poročilo, v katerem za vsako nalogo opišete postopek reševanja, zapišete rešitev, in komentirajte rezultat. Prav tako dodajte izjavo, da ste nalogo reševali samostojno. Če poročilo skenirate, mora biti obvezno oddano v pdf obliki. Priložite programe, s katerimi ste naloge rešili. Programi naj bodo smiselno poimenovani in razporejeni v mapah, ki naj bo bodo poimenovane `nal1`, `nal2`, ... Naloge morajo biti rešene v Matlabu (uporabite lahko tudi Octave ali Scilab), razen če je v nalogi drugače navedeno.

Če imate kakšno vprašanje o nalogah ali Matlabu, se obrnite na asistenta ali profesorja. Če menite, da je vprašanje zanimivo tudi za ostale, uporabite forum.

Naj bodo $c_1c_2c_3c_4$ zadnje 4 cifre vaše vpisne številke in $V = (6 * c_1c_2 + c_3c_4)/1000$.

Skupaj morate zbrati 4. točke. Če želite več točk, je potrebno rešiti vsaj kako nalogo ovrednoteno z dvema točkama.

1. točka Pri računanju Googleovega rangiranja strani moramo izračunati dominantno lastno vrednost matrike A , kjer je $A = \alpha P + (1 - \alpha)\frac{1}{n}ee^T$ in $P = Q + \frac{1}{n}ed^T$. Matrika Q je utežena matrika povezav med internetnimi stranmi in jo naložiš z ukazom `loadStanfordMatrix`, n je dimenzija matrike, $\alpha \in [0, 1]$ je faktor teleportacije, e je vektor samih enic. Vektor d ima i -to komponento enako 1, če je i -ti stolpec matrike Q ničeln, vse njegove preostale komponente pa so enake 0. Potrebne datoteke so priložene. Množenje z A implementiraj kot

$$Az = \alpha(Q + \frac{1}{n}ed^T)z + \frac{(1 - \alpha)}{n}ee^Tz = \alpha Qz + \beta \frac{1}{n}e,$$

$$\beta = \alpha d^Tz + (1 - \alpha)e^Tz.$$

Izračunaj dominantno lastno vrednost, kjer $\alpha = V$ in še dodatno za

$$\alpha = 0.01, 0.04, 0.1, 0.3, 0.5, 0.8, 0.99.$$

Kaj opaziš?

Za izračun dominantne lastne vrednosti primerjaj uporabo potenčne metode in Arnoldijeve metode. Primerjaj hitrost konvergence in preveri, da dobiš pravi rezultat še z uporabo Matlabove funkcije `eigs`.

1. točka Za Poissonovo 2D matriko in $m = n = 9$ izračunaj dve največji in dve najmanjši lastnosti z uporabo Lanczosove metode in Rayleigh-Ritzovih vrednosti. Uporabi metodo brez reortogonalizacije in z naključno izbranim začetnim približkom. Kjer dobiš tridiagonalne bločne sisteme, jih poizkusi rešiti učinkovito (* učinkovitost ni obvezna in je dodatna naloga). Nariši sliko, kjer za vsak primer narišeš vse lastne vrednosti in vse Ritzove vrednosti. Ali lahko na tak način dobiš vse lastne vrednosti, do večkratnosti natančno?

Nasvet Poissonovo matriko lahko dobite z uporabo ukaza `gallery` v Matlabu.

1. točka Za matriko iz naloge 2 poizkusi dobiti vse lastne vrednosti, vključno z večkratnostmi, tako da v svoji kodi uporabiš reortogonalizacijo. Uporabi popolno reortogonalizacijo, kjer reortogonaliziraš enkrat ali dvakrat, kar primerjaj s selektivno reortogonalizacijo. Ali kje dobiš bistveno boljše rezultate?

Ali lahko po zmernem številu korakov dobiš šest največjih (šest najmanjših) ekstremnih lastnih vrednosti z večkratnostjo vred?

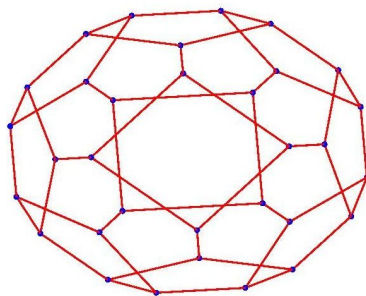
1. točka Eden izmed algoritmov za risanje kubičnih poliedrskih grafov (graf predstavlja oglišča in stranice nekega poliedra, kjer se v vsakem oglišču srečajo tri stranice), je uporaba lastnih vektorjev in lastnih vrednosti. Če vzamemo incidenčno matriko A (matrika $n \times n$, ki ima na (i, j) -tem mestu 1, če sta točki i in j povezani, sicer pa 0), se hitro vidi, da je ena lastna vrednost (ki je ravno največja) ravno 3, normiran lastni vektor pa je enak $e = \frac{1}{\sqrt{n}}[1 \ 1 \ \dots \ 1]^T$. Sedaj vzamemo λ_2 , λ_3 in λ_4 ter normirane lastne vektorje, ki naj bodo po vrsti x , y , z . Če sedaj za koordinate i -te točke vzamemo kar $T_i = (x_i, y_i, z_i)$, dobimo lepo tridimenzionalno sliko grafa. Na ta način poiščite koordinate grafa, ki vam ga vrne podprogram `G=kubicnigraf(vpisna)`, ki je priložen in tudi na voljo na spletni strani. i -ta vrstica matrike G ima obliko

$$i \quad a_{i1} \quad a_{i2} \quad a_{i3},$$

kjer je i indeks točke, a_{i1}, a_{i2}, a_{i3} pa so indeksi njenih treh sosedov. Iz teh podatkov sestavite incidenčno matriko A , izračunajte lastne vektorje in tako določite koordinate ter narišite graf.

Za račun lastnih druge, tretje in četrte lastne vrednosti uporabite dva načina:

- (a) Z uporabo Matlabove funkcije `eigs`, ki uporablja implicitno Arnoldijevo metodo.
- (b) Z uporabo Lanczosove metode z reortogonalizacijo.



Pri risanju grafa v Matlabu si pomagajte z ukazom `gplot3`. Implementacija množenja naj bo učinkovita.

2. točki Na strani 19 v knjigi lahko najdete opis modela Brusselator za reševanje sistema parcialnih diferencialnih enačb iz kemije. V Matlabu sestavite program, ki bo generiral matriko A , ki jo dobimo z diskretizacijo in je odvisna od parametra L . Sistem je stabilen, če za vse lastne vrednosti λ dobljene matrike velja, da je $Re(\lambda) < 0$. Izkaže se, da je sistem pri $L = 0.5$ stabilen, pri $L = 1$ pa nestabilen. Potrebno je določiti vrednost parametra L , pri katerem pride do prehoda iz stabilnega v nestabilen sistem. Za preverjanje pravilne konstrukcije lahko matrike primerjate z matrikami, ki so objavljene v zbirki, ki je na voljo tudi v spletni učilnici. Tako je pod številom 1630 matrika velikosti 200×200 pri $L = 0.5$, pod številom 1631 pa matrika pri $L = 1$. Pri vašem računanju uporabite na koncu matrike velikosti vsaj 10000×10000 . Opišite postopek in katere numerične metode so najbolj primerne.
2. točki Rešite nalogo 11 (točke a, b in c) na strani 227 iz knjige, ki zahteva razvoj bločne variante BiLanczosovega algoritma za reševanje nesimetričnega linearne sistema. Metodo implementirajte v Matlabu, pri čemer naj bo velikost bloka parameter metode. Preizkusite jo na večjem testnem primeru in jo primerjajte z navadno BiLanczosovo metodo (z velikostjo bloka 1). Za test uporabite matriki številka 214 in 2335, ki ju dobite z ukazom `Ufget`.

1. točka Denimo, da tako kot pri Lanczosevi metodi, začnemo z normiranim vektorjem v_1 , preostale pa izračunamo po tričlenski rekurzivni formuli (kjer privzamemo $\beta_0 = 0$ in $v_0 = 0$)

$$\tilde{v}_{j+1} = Av_j - \alpha_j v_j - \beta_{j-1} v_{j-1} \quad (1)$$

$$v_{j+1} = \tilde{v}_{j+1} / \gamma_j, \text{ kjer je } \gamma_j = \|\tilde{v}_{j+1}\| \quad (2)$$

Skalarji γ_j so izbrani tako, da so vektorji v_j normirani, medtem ko so skalarji α_j in β_j poljubni. Denimo, da to rekurzijo izvajamo do največ $(n+2)/2$ korakov. Pokažite, da potem obstaja tak neničelni vektor w_1 , da velja

$$\langle v_j, (A^H)^k w_1 \rangle = 0$$

za vse $k < j - 1$, kjer je $j = 2, \dots, \lfloor \frac{n+2}{2} \rfloor$.

2. točki Zapišite metodo podobno CG za reševanje linearnega sistema $Ax = b$ z matriko oblike $A = I - B$, kjer je matrika B poševno hermitska, torej $B = -B^H$. Metoda naj na vsakem koraku minimizira 2-normo ostanka $b - Ax_j$. Metodo implementirajte v Matlabu in preizkusite na večjem testnem primeru.