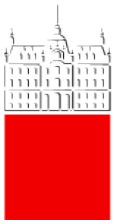
Univerza *v Ljubljani*
Fakulteta za *matematiko in fiziko*



Lastne vrednosti in lastni vektorji

3. naloga pri Matematično-fizikalnem praktikumu

Avtor: Marko Urbanč (28191096)
Predavatelj: prof. dr. Borut Paul Kerševan

27.10.2021

Kazalo

1 Uvod

Kompleksno število α imenujemo *lastna vrednost* za matriko A, če obstaja tak neničeln vektor $\mathbf{v} \in C^n$, da je $A\mathbf{v} = \alpha\mathbf{v}$ oz. $(A - \alpha I)\mathbf{v} = 0$. Vektor $\mathbf{v}$ imenujemo *lastni vektor* za A pri lastni vrednosti α.
Problem lastnih vrednosti in lastnih vektorjev je zelo pomemben v kvantni mehaniki, kjer gledamo funkcije kot vektorje v Hilbertovem prostoru L^2. Lastne vrednosti predstavljajo možne vrednosti, ki jih pri meritvah lahko zavzamejo opazljivke.
Primer so lastne energije, ki ustrezajo lastnim funkcijam Hamiltonovega operatorja. Enodimenzionalni linearni harmonski oscilator (delec mase m s kinetično energijo $T(p) = p^2/2m$ v kvadratičnem potencialu $V(q) = m\omega^2 q^2/2$) opišemo z brezdimenzijsko Hamiltonovo funkcijo

$$H_0 = \frac{1}{2}\left(p^2 + q^2\right) \ ,$$

kjer energijo merimo v enotah $\hbar\omega$, gibalne količine v enotah $(\hbar m\omega)^{1/2}$ in dolžine v enotah $(\hbar/m\omega)^{1/2}$. Lastna stanja nemotenega Hamiltonovega operatorja $|n\rangle$ lahko zapišemo s Hermitovimi polinomi $\mathcal{H}_n$ kot

$$|n\rangle = (2^n n!\sqrt{\pi})^{-1/2} \mathrm{e}^{-q^2/2}\, \mathcal{H}_n(q) \ .$$

Te lastne funkcije morajo zadoščati stacionarni Schrödingerjevi enačbi

$$H_0|n^0\rangle = E_n^0|n^0\rangle \ ,$$

kjer prepoznamo, da gre pravzaprav samo za problem lastnih vrednosti in lastnih funkcij. $E_n^0 = n + 1/2$ za $n = 0, 1, 2, \ldots$ so nedegenerirane lastne vrednosti energije. Hamiltonianu linearnega harmonskega oscilatorja dodamo *anhilacijski člen*

$$H = H_0 + \lambda q^4$$

kjer je $0 < \lambda < 1$ in q matrika matričnih elementov, ki uteleša izbirno pravilo za električni dipolni prehod med nivoji harmonskega oscilatorja $\langle i|H|j\rangle$. Z njimi lahko ugotovomo kako motnja anhilacijskega člena spremeni lastne energije. Pričakovano vrednost prehodnega matričnega elementa za posplošeno koordinato lahko zapišemo kot

$$q_{ij} = \langle i|q|j\rangle = \frac{1}{2}\sqrt{i + j + 1}\ \delta_{|i-j|,1} \ .$$

Matematično lahko gledamo harmonski oscilator v neskončno dimenzijah, v numeriki pa se moramo omejiti na neko siselno razsežnost N.

2 Naloga

Naloga od nas zahteva, da uporabimo postopek numerične diagonalizacije in z njim poiščemo nekaj najnižjih lastnih vrednosti in lastnih funkcij za moten Hamiltonian. Nove valovne funkcije izrazimo v bazi starih, torej kot linearna kombinacija lastnih funkcij za nemoten harmonski oscilator. Preveriti moramo, da v limiti $\lambda \to 0$ velja $E_n \to E_n^0$ in raziskati moramo kako so rezultati odvisni od razsežnosti N in kako konvergirajo lastne vrednosti pri velikih N. Poleg tega lahko matrične elemente računamo tudi kot matrilne elemente kvadrata koordinate ali pa četrte potence koordinate. Naloga želi, da primerjamo te načine.
Iz rekurzivnih zvez za Hermitove polinome lahko izpeljemo

$$\langle i|q^2|j\rangle = \frac{1}{2}\left[\sqrt{j(j-1)}\,\delta_{i,j-2} + (2j+1)\,\delta_{i,j} + \sqrt{(j+1)(j+2)}\,\delta_{i,j+2}\right],$$

$$\langle i|q^4|j\rangle = \frac{1}{2^4}\sqrt{\frac{2^i\,i!}{2^j\,j!}}\left[\delta_{i,j+4} + 4\,(2j+3)\,\delta_{i,j+2} + 12\left(2j^2+2j+1\right)\,\delta_{i,j}\right.$$
$$\left. + \left(2j^2-3j+1\right)\,\delta_{i,j-2} + 16j\left(j^3-6j^2+11j-6\right)\,\delta_{i,j-4}\right].$$

3 Opis reševanja

Problema sem se zopet lotil v Pythonu, kjer sem veliko koristil knjižnice `NumPy` za matematične operacije in delo z matrikami in `Matplotlib` za risanje in animacije. Po originalnem načrtu sem prvo spisal funkcijo, ki s pomočjo *Householderjevih transformacij* spravi matriko v tridiagonalno obliko. Householderjeve transformacije so linearne trasformacije, ki predstavljajo zrcaljenje preko ravnine (oz. hiperravnine), ki je podana z enotsko normalo $\mathbf{v}$ pravokotno na ravnino. Zrcaljenje točke x lahko zapišemo kot

$$x' = x - 2\mathbf{v}(\mathbf{v}^H x).$$

Matriko te transformacije (Householderjeva matrika) lahko zapišemo z tenzorskim produktom kot:

$$P = I - 2\mathbf{v}\mathbf{v}^H\,.$$

Algoritem in njegova izpeljava sta podrobneje opisana v knjigi Numerical Analysis [1]. V grobem ga za $k = 1, \ldots, N$ lahko povzamemo kot:

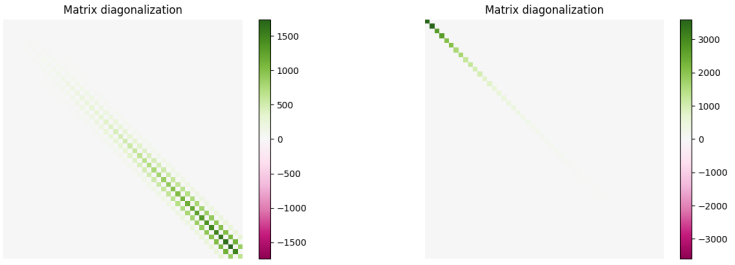
$$\alpha = -\mathrm{sgn}(a^k_{k+1,k})\sqrt{\sum_{j=k+1}^{n}(a^k_{jk})^2}\,,$$
$$r = \sqrt{\frac{1}{2}(\alpha^2 - a^k_{k+1,k}\alpha)}\,,$$
$$v^k_1, v^k_2, \cdots = v^k_k = 0\,,$$
$$v^k_{k+1} = \frac{a^k_{k+1,k} - \alpha}{2r}\,,$$

$$v_j^k = \frac{a_{jk}^k}{2r}; \text{ for } j = k+2, k+3, ..., N$$
$$P^k = I - 2\mathbf{v}^{(k)}(\mathbf{v}^{(v)})^T \,,$$
$$A^{(k+1)} = P^k A^{(k)} P^k \,.$$

Problem je, da moja implementacija postopka tridiagonalizacije ne deluje v nekaj primerih. Probleme ima, ko ima nad diagonalnim elementom vrednost 0. Takrat pride do deljenja z 0 in postopek vrne polno matriko *nan*. Žal po knjigah ali spletu nisem zasledil kako postopati v tej obliki in tudi nekoliko mi je zmanjkalo časa za "razhroščevanje"moje kode. Skrbi me, da je v celoti nekaj nekje narobe, kar očitno ne najdem, ker je nadaljna QR dekompozicija **hitrejša** brez tega prvotnega postopka tridiagonalizacije, kar sicer ni smiselno. Tudi komponente lastnih vektorjev so bile ravno prezrcaljene od tega, kar vrne `NumPy`. Torej je lahko kje kaka napaka v indeksih oz. da je nekaj narobe obrnjeno.

Tridiagonalno matriko naj bi z QR dekompozicijo na podlagi Householderjevih transformacij spravil v diagonalno obliko. QR dekompoziciji, kot hitro rešitev v časovni stiski, podam kar prvotno matriko. Funkcija za diagonalizacijo ima pogoj za ustavitev, ko je vsota absolutnih vrednosti elementov matrike za nek majhen epsilon različna, torej skoraj enaka, kot vsota absolutnih vrednosti diagonalnih elementov. Zaradi težav z natančnostjo floatov je potrebno gledati enakost v neki epsilonski okolici. Za dodatno pospešitev diagonalizacije uporabljam preprosti "filter", ki preverja za elemente, ki so blizu nič. Ko padejo pod prag tolerance (recimo 10^{-15}) jih filter postavi direktno na 0. Zraven pa sproti še množim vse ortogonalne matrike Q iz dekompozicije. Njihov produkt mi na koncu da pripadajoče lastne vektorje matrike.

Spisal sem še kodo, ki tvori matrike za problem anharmonskega oscilatorja.



(a) Pred postopkom (b) Po postopku

Slika 1: Diagonalizacija matrike za $N = 50$ in $\lambda = 0.5$

Hamiltionian anharmonskega oscilatorja izračunam kot

$$H = H_0 + \lambda q^4 \,,$$

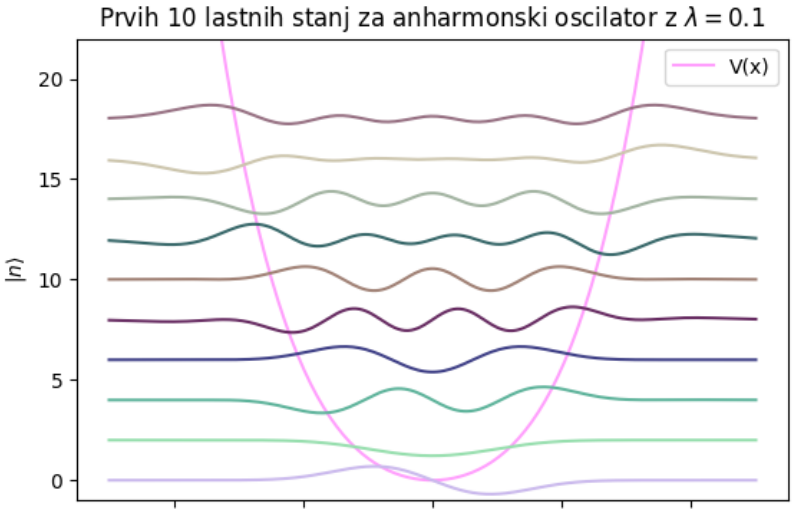
kjer matriko matričnih elementov q lahko izračunam na 3 načine. Ti so kot četrta potenca pričakovane vrednosti matričnega elementa, kvadrat pričakovane vrednosti kvadrata matričnega elementa in kot pričakovano vrednost četrte potence matričnega elementa. Med prvim in drugima dvema načinoma je nekaj

razlike, ki jo mislim pokomentirati pri rezultatih.
V kodo sem dodal še bazne vektorje za harmonski oscilator in možnost, da razvijem nek vektor po tej bazi. To vse skupaj, s potencialom in tabelo lastnih, energij sem risal nato kot končen rezultat. Postopek diagonalizacije pa sem za zabavo in zgled tudi animiral.

4 Rezultati

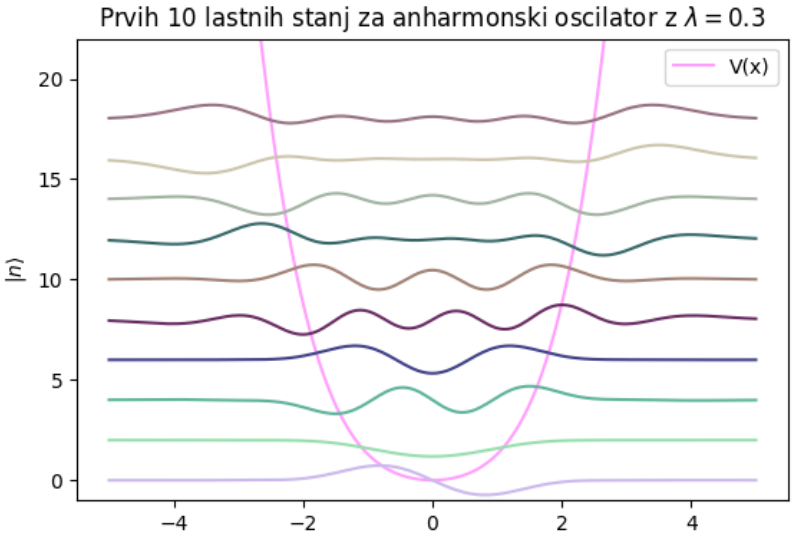
4.1 Lastna stanja

Zdelo se mi je smiselno, da velikost matrike odrežem pri $N = 10$. Več podatkov je težko smiselno prikazati na grafu, energije stanj rastejo dalje. Slike so narisane z uporabo četrte potence matričnih elementov. Pri vsakem grafu je še tabela lastnih energij za lastna stanja.



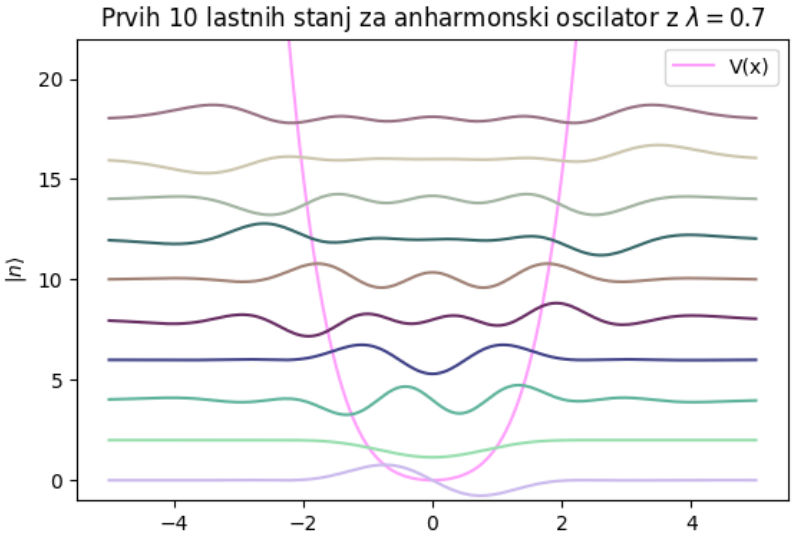
	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.56	1.77	3.14	4.63	6.17	7.9	10.26	10.76	21.49	21.57

Slika 2: Graf lastnih stanj za $\lambda = 0.1$



	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.64	2.09	3.84	5.88	7.95	9.24	18.16	18.32	49.29	49.34

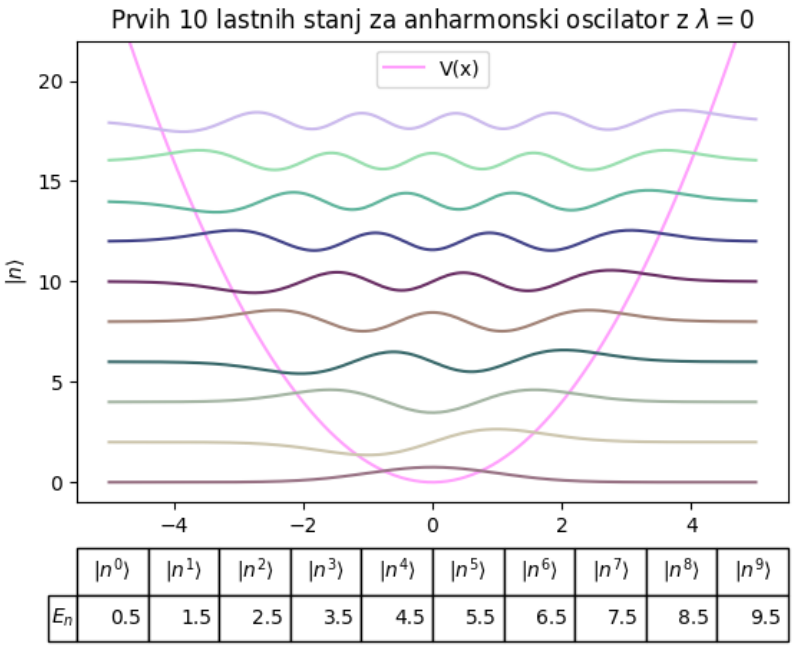
Slika 3: Graf lastnih stanj za $\lambda = 0.3$



	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.74	2.51	4.62	7.11	11.49	12.08	34.5	34.61	105.02	105.07

Slika 4: Graf lastnih stanj za $\lambda = 0.7$

Preveril sem tudi limito $\lambda \to 0$ in res dobimo vezana stanja za harmonski oscilator.

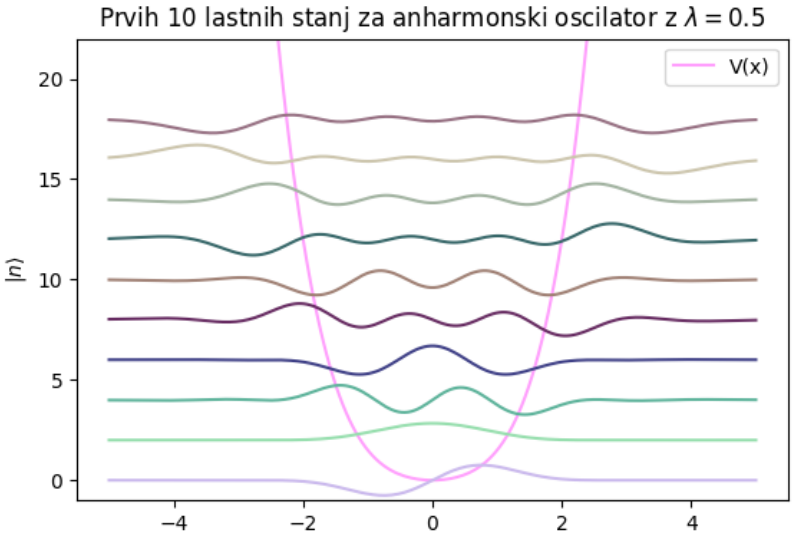


	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.5	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5

Slika 5: Graf lastnih stanj za $\lambda = 0.0$, kar ustreza linearnemu harmonskemu oscilatorju

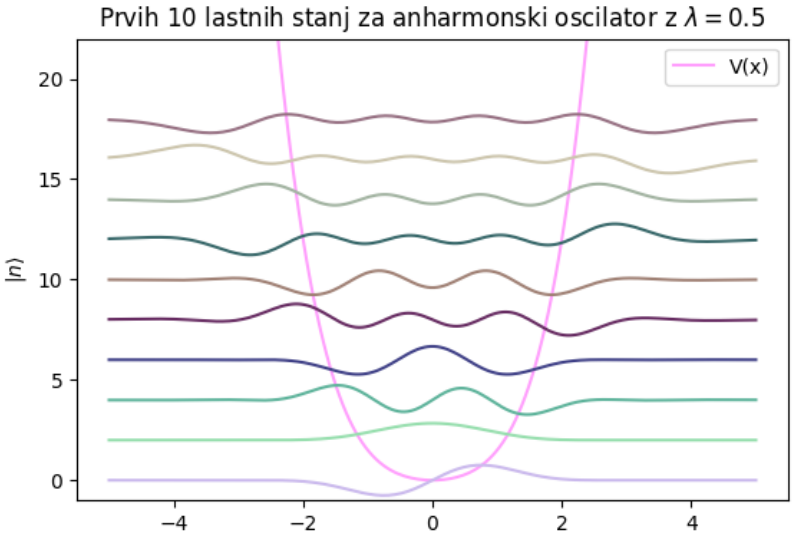
4.2 Računanje matrike q na drugačne načine

Kot prej omenjeno, želim predstaviti tudi slike, kjer sem uporabil druga načina računanja matričnih elementov. Vsi načini so med seboj podobni, ampak vrnejo nekoliko drugačne matrike. Sploh se to pozna pri velikih dimenzijah. Sumim, da gre tu za neke sorte trik, kot recimo ko imamo povprečen odmik $\langle x \rangle = 0$ ampak $\langle x^2 \rangle \neq 0$. Posledično so lastne energije in lastna stanja nekoliko medseboj različna pri vseh treh metodah.



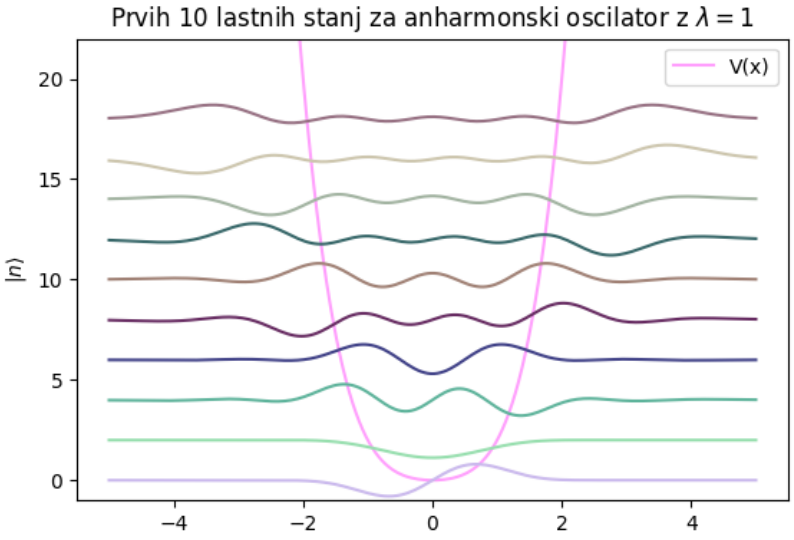
	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.7	2.33	4.28	6.46	9.69	13.96	26.31	36.5	77.15	98.88

Slika 6: $\lambda = 0.5$ in kvadrat q^2



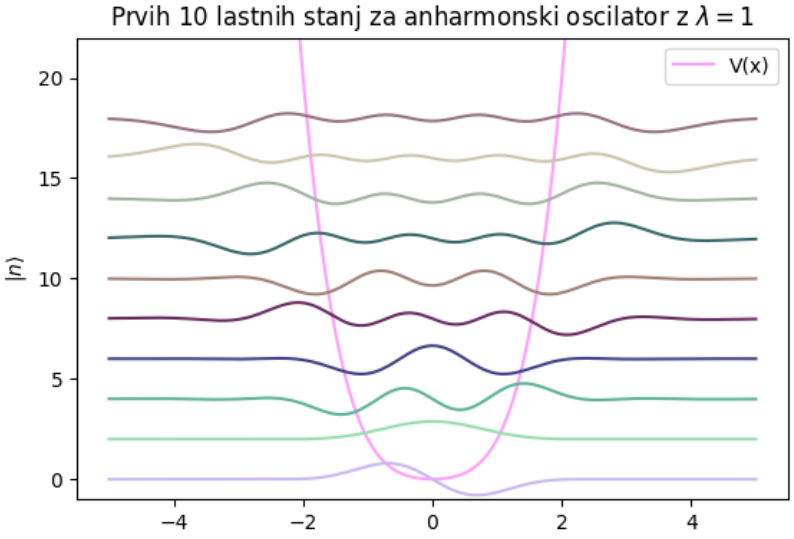
	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.7	2.33	4.34	6.73	10.87	15.75	29.61	40.55	83.85	106.53

Slika 7: $\lambda = 0.5$ in q^4



	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.81	2.73	5.04	7.94	14.25	22.2	46.82	66.45	146.84	189.42

Slika 8: $\lambda = 1$ in kvadrat q^2

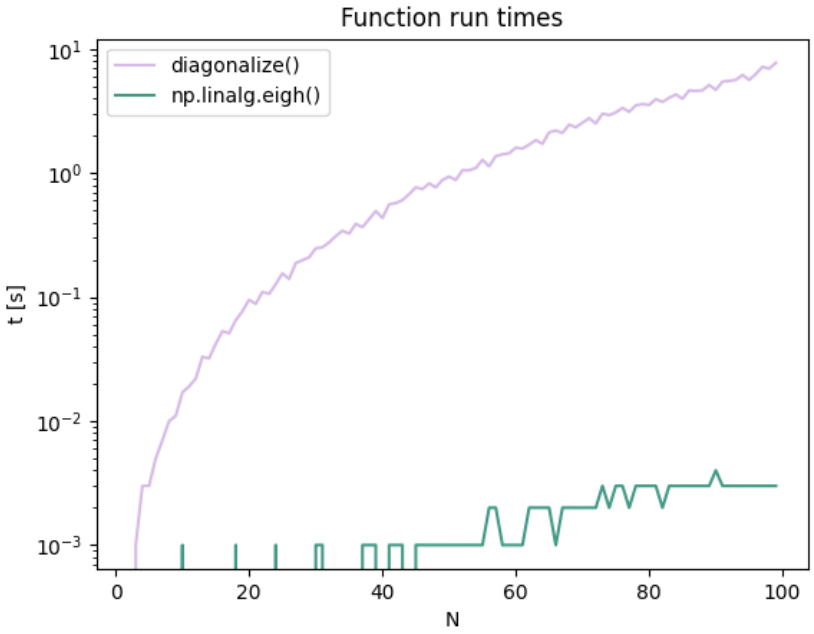


	$\|n^0\rangle$	$\|n^1\rangle$	$\|n^2\rangle$	$\|n^3\rangle$	$\|n^4\rangle$	$\|n^5\rangle$	$\|n^6\rangle$	$\|n^7\rangle$	$\|n^8\rangle$	$\|n^9\rangle$
E_n	0.81	2.74	5.36	8.84	16.79	25.82	53.28	74.37	160.01	204.47

Slika 9: $\lambda = 1$ in q^4

4.3 Hitrost QR dekompozicije

Zdelo se mi je smiselno primerjati hitrost moje kode za QR dekompozcijo proti že pripravljenim metodam. Žal je prvotni del za tridiagonalizacijo trenutno dokaj nekoristen tako da sem pomeril hitrost res samo QR dekompozicije. Narisal sem graf, kjer primerjam hitrost moje funkcije `diagonalize()` proti `np.linalg.eigh()`, ki je vgrajena specifično za diagonalnih hermitskih (oz. v primeru realnih, simetričnih) matrik. Precej očitno je, da je funkcija iz `NumPy`

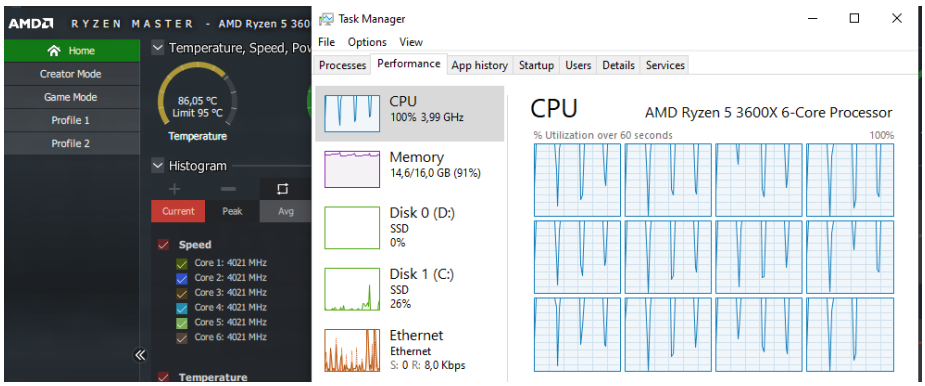


Slika 10: Čas izvajanja funkcije v logaritemski skali

mnogo hitrejša. Za manjše matrike je tako hitra, da je pravzaprav instantna. Sicer bi verjetno dobil boljše podatke po večih ponovitvah postopka, ampak za grobo oceno je očitno, da je moja metoda strahotno počasna.

4.4 Komentarji in izboljšave

Zdi se mi, da sem zopet večina komentarjev že napisal sproti kjer so bili potrebni. Naloga mi je vzela bistveno več časa kot sem si pričakoval. Zelo sem se trudil, da bi usposobil tridiagonalizacijo ampak mi je na koncu res zmanjkalo časa in moram to pustiti za drugič. Zabavno se je bilo naučiti animirati grafe. Jasni izboljšavi, ki bi ju lahko naredil sta optimizacija QR metode in pa recimo večkratno ponavljanje "štopanja"funkcij za graf časa izvajanja.
Za dodatek, da res ne crashamo računalnikov z igricami, je tu moj računalnik precej jezen, ko je dobil $N = 10000$ matriko [11]. Smešno je kako opazno padeta poraba in s tem temperatura med iteracijami. Antiklimaktično se postopek ni izvedel do konca, ker mi je zmanjkalo RAMa.



Slika 11: Hardware torture

Literatura

[1] Richard L. Burden and J. Douglas Faires. *Numerical analysis*. Brooks/Cole Pub. Co., 9 edition, 2011.