

8. naloga – Metropolisov algoritem

Matic Debeljak

24. november 2020

1 Naloga

1. **Molekularna verižnica:** 17 členkov dolga nitkasta molekula je obešena za oba konca. Vsak členek se lahko povesi od ničelne lege na poljubnega od 19 nivojev in si s tem zmanjša potencialno energijo za eno enoto na nivo. Če pa s tem prenatagne vezi do sosedov, plača s prožnostno energijo, ki je za vsakega sosedu enaka kvadratu razlike v nivojskem številu. Določi ravnovesno energijo v odvisnosti od temperature. Za poteze lahko uporabiš spremembo za en nivo na izbranem slučajnem mestu.
2. **Isingov model:** feromagnetne in anti feromagnetne snovi v dveh dimenzijah v približku dveh stanj opišemo s Hamiltonovim operatorjem

$$\mathcal{H} = -J \sum_{\langle ij \rangle} s_i s_j - H \sum_i s_i, \quad J = \pm 1$$

kjer je $s_i \pm 1$ in vsota $\langle ij \rangle$ teče le po vezeh med najbližjimi sosedi. Če ni zunanega polja ($H = 0$), temperatura T_c faznega prehoda pri feromagnetu zadošča enačbi

$$\sinh \frac{2J}{k_B T_c} = 1 \quad \Rightarrow \quad T_c \approx 2.269185 \frac{J}{k_b}$$

Določi povprečno energijo $\langle E \rangle$ in lastno magnetizacijo $\langle S \rangle$ v odvisnosti od temperature. $S = \sum_{i=1}^N s_i$ je celotna magnetizacija sistema. Oglej si tudi spinsko susceptibilnost in specifično toploto pri različnih jakostih magnetnega polja.

$$\chi = \frac{\langle S^2 \rangle - \langle S \rangle^2}{N k_B T} \quad C = \frac{\langle E^2 \rangle - \langle E \rangle^2}{N k_B T^2}$$

2 Rezultati

2.1 Molekularna verižnica

Energijo molekularne verižice zapišemo kot

$$E = \sum_{i=1}^N mgh_i + \sum_{i=1}^{N-1} \frac{1}{2}kd^2(h_{i+1} - h_i)^2$$

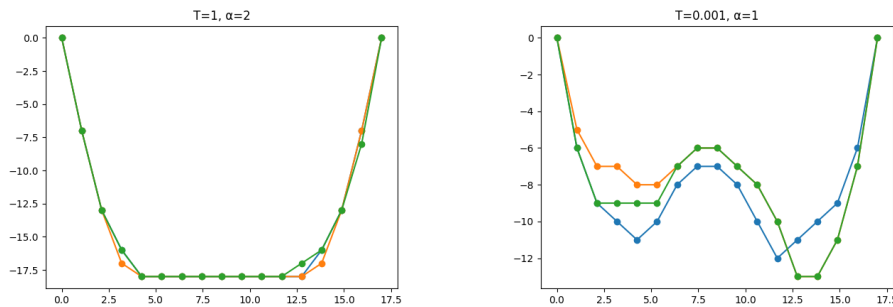
enačbo delimo z kd^2 in dobimo:

$$E' = \sum_{i=1}^2 \alpha h_i + \sum_{i=1}^{N-1} \frac{1}{2}(h_{i+1} - h_i)^2; \quad \alpha = \frac{mg}{kd^2}$$

kjer α predstavlja razmerje med potencialno in prožnosto energijo verižnice. Metropolisov algoritem za verižnico:

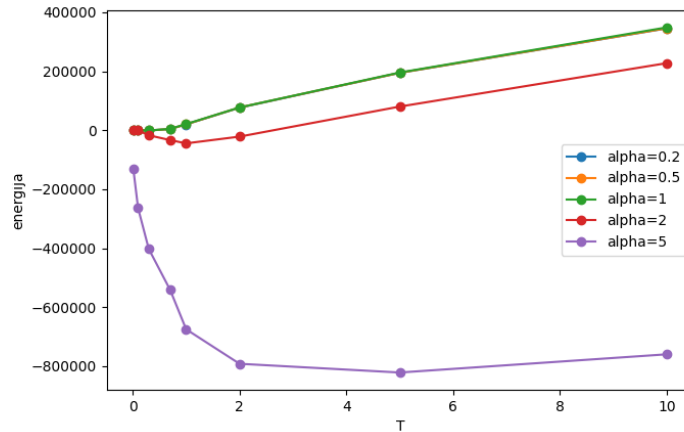
1. Naključno izreba pozicijo i in $\delta_i = \pm 1$
2. Predpostavi potezo $h_i \rightarrow h_i + \delta_i$
3. Izračunaj $\Delta E' = \delta_i^2 - \delta_i(h_{i+1} - 2h_i + h_{i-1} - \alpha)$
4. Če velja $\Delta E' < 0$, potezo sprejmemo
5. Če velja $\Delta E' > 0$, potezo sprejmemo če $\zeta \sim U(0, 1) \leq e^{-\Delta E'/k_B T}$

Za lažjo analizo sem predpostavil še $k_B = 1$. Opazimo, da z večanjem temperature narašča število sprejetih potez, ki energijo povečajo. V limiti $T \rightarrow \inf$ sprejmemo vse poteze, v limiti $T \rightarrow 0$ pa sprejmemo le poteze, ki zmanjšajo energijo sistema. Poglejmo si kako izgledajo končne konfiguracije po 10^5 potezah za nekaj različnih parametrov T in α ter enakih začetnih konfiguracijah.

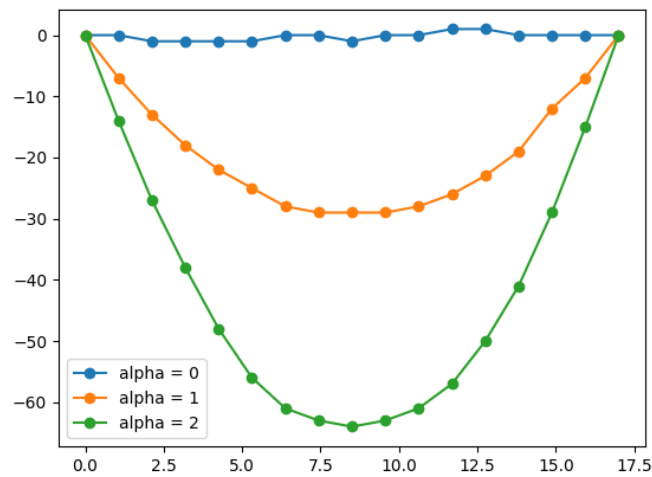


Opazimo da so končne konfiguracije pri enakih parametrih dokaj podobne, a se vseno razlikujejo, to je posledica tega da smo delali naključne poteze (random.seed ni bil definiran). Konfiguracije pa so zelo različne od vrednosti parametrov T in α .

Poglejmo si še kako se spreminja končna energija kot funkcija temperature za različne α .

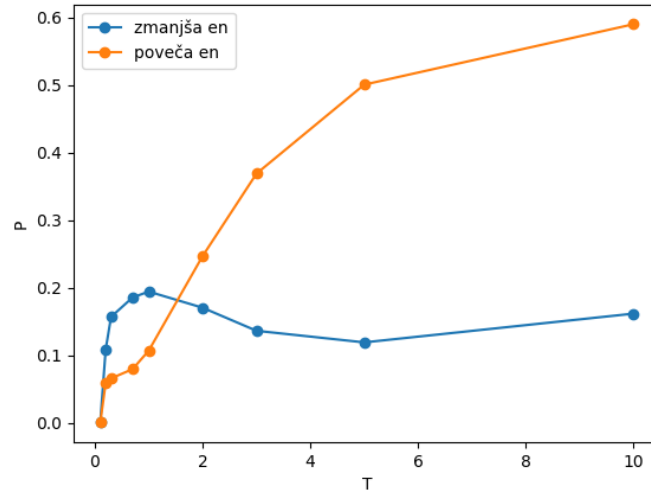


Zanimivo nas tudi, kaj se zgodi z verigami, če ne omejimo števila nivojev. Na sliki 3 so prikazane tri končne konfiguracije, za različne α pri $T = 1$.



Te konfiguracije so oblike $y = a + b \cosh(cx + d)$.

Poglejmo si še delež sprejetih potez v odvisnosti od T . Pričakovano število



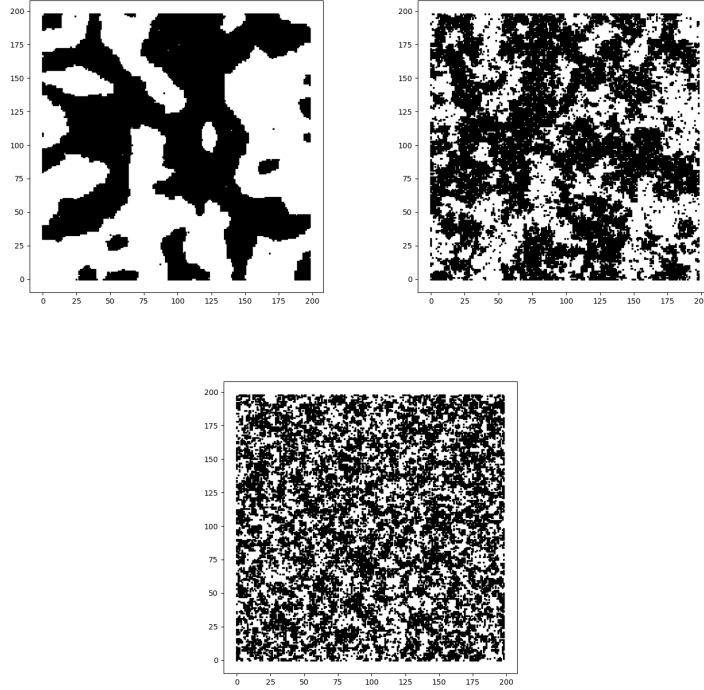
sprejetih potez, ki poveča energijo narašča z višanjem temperature. Malo pa me je presenetilo, da je pri majhnih T ($T \leq 0.1$) sprejetih tudi precej manj potez, ki zmanjša energijo, to pa je verjetno posledica tega, da se algoritem zatakne v lokalnem minimumu precej hitro in je zato delež sprejetih potez majhen.

2.1.1 Isingov model

Podobno kot pri prvi nalogi uporabimo Metropolisov algoritem.

1. Naključno izreba pozicijo spina (i, j)
2. Predpostavi potezo $S(i, j) \rightarrow -S(i, j)$
3. $\Delta E = 2JS(i, j)[S(i-1, j) + S(i+1, j) + S(i, j-1) + S(i, j+1)] + 2HS(i, j)$
4. Če velja $\Delta E' < 0$, potezo sprejmemo
5. Če velja $\Delta E' > 0$, potezo sprejmemo če $\zeta \sim U(0, 1) \leq e^{-\Delta E/k_B T}$

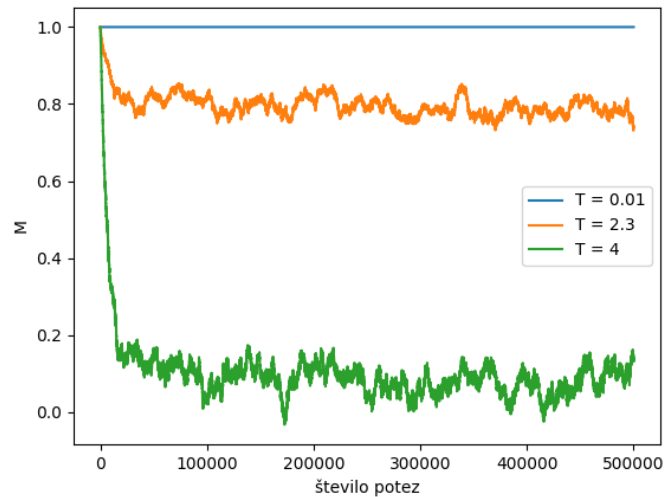
V celotni nalogi sem obravnaval isingov model za feromagnete ($J = 1$). Za lažjo predstavbo sem najprej narisal končne konfiguracije za tri različna temperaturna območja in $H = 0$.



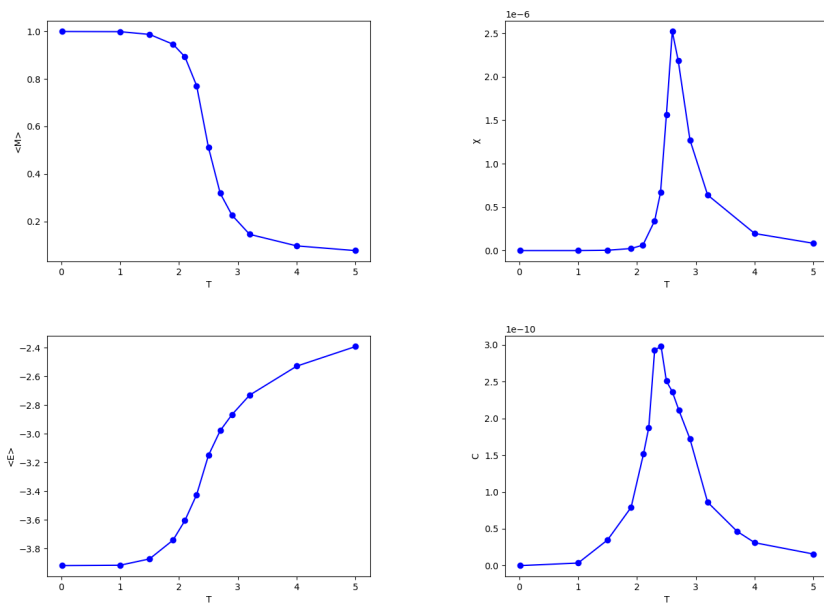
Slika 1: Končne konfiguracije pri $T = 1$ (levo zgoraj) $T \approx T_C \approx 2.3$ (desno zgoraj) in $T = 3$ (spodaj). Vse slike vsebujejo 200×200 veliko kvadratno mrežo. Število korakov je bilo $200 \times 200 \times 200$.

Zaradi časovne zahtevnosti sem od tu naprej vse delal na mrežah velikih 50×50 , ter številu korakov $200 \times 50 \times 50$. Prav tako sem v začetnem stanju vse spine postavil v stanje 1. Poglejmo si, kako se spreminja magnetizacija vzorca v odvisnosti števila potez za različne temperature (slika 2).

Opazimo, da se po začetni spremembi magnetizacija približno ustali na neki vrednosti, ki je odvisna od temperature, povprečje te vrednosti definiramo kot $\langle M \rangle$. S pomočjo enačb v navodilih izračunamo potrebne količine (slika 3).



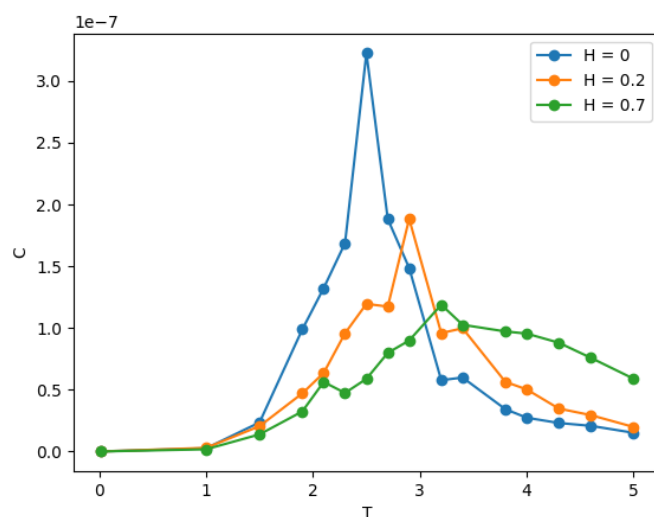
Slika 2: Magnetizacija v odvisnosti od števila potez za različne T



Slika 3: Prikaz povprečne magnetizacije (levo zgoraj), magnetne susceptibilnosti (desno zgoraj), povprečne energije (levo spodaj) in specifične toplote (desno spodaj) v odvisnosti od temperature.

Energija pričakovano narašča s temperaturo, hitreje v okolici kritične temperature. Magnetizacija pada s temperaturo, saj pri $T = 0$ ne obrne nobenega spina iz začetnega stanja, saj je to energijsko neugodno, pri večjih temperaturah, pa je magnetizacija 0 saj je približno pol spinov = 1 in druga polovica = -1. Magnetna susceptibilnost in specifična toplota pa imata obe vrhblizu T_c drugje pa padata proti 0.

Poglejmo si še, kaj se zgodi ko vklopimo magnetno polje ($H \neq 0$). Vpliv magnetnega polja se najbolj vidi na grafu specifične toplote v odvisnosti od temperature (slika 4).



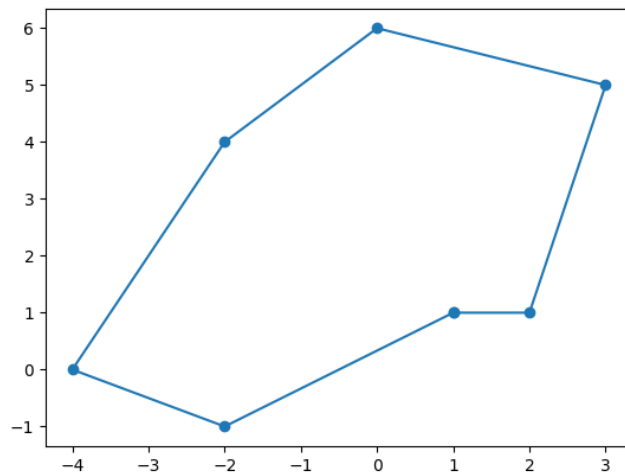
Slika 4: Specifične toplota v odvisnosti od temperature za različne jakosti magnetnega polja.

Opazimo, da se maksimum specifične toplote pojavi pri višji temperaturi z naraščanjem jakosti magnetnega polja. To pomeni, da z naraščanjem jakosti magnetnega polja naršča temperatura pri kateri pride do faznega prehoda.

2.2 Trgovski potnik

Problem trgovskega potnika sem najprej sprogramiral tako, da je program preveril vsako možno kombinacijo poti, ta program je sicer dal točen rezultat, a je imel veliko časovno zahtevnost. Vseno pa ga je bilo smiselno uporabiti, da sem preveril ali deluje program z metropolsevim algoritmom.

Pogljemo si kako ta program reši problem osmih mest (slika 5).

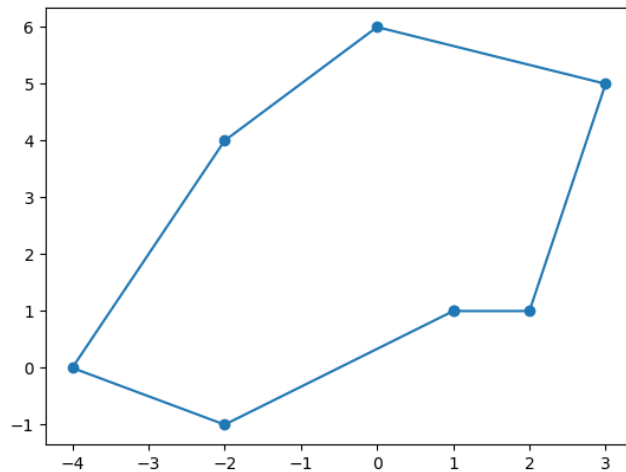


Slika 5: Točna rešitev problema osmih mest.

Metropolisov algoritem za trgovskega potnika sem napisal kot:

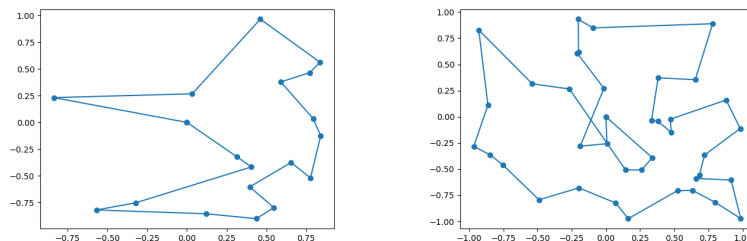
1. izreba 2 mesti in jih zamenja
2. izračuna ΔL
3. potezo sprejmi, če velja $\delta L < 0$, sicer jo sprejmi z verjetnostjo $P = e^{-\Delta L/T}$
4. zmanjšaj temperaturo

Poglejmo si še graf za metropolisev algoritem.(slika 5)



Slika 6: Rešitev problema osmih mest z metropolisevim algoritmom.

Kot smo pričakovali je rešitev enaka. Torej program deluje, pogledjmo pa še kako deluje pri večjem številu mest. (slika 6)



Slika 7: Rešitev problema 20 in 40 mest z metropolisevim algoritmom.

Rešitve sicer ne moram preveriti ampak po občutku zgleda pravilna. Program deluje precej hitro a ima majhno pomankljivost (ki bi jo z nekaj dodatnega časa lahko odpravil) in sicer da pri $e^{-\Delta L/T}$ pri nizkih vrednosti T prihaja do "overflowa" zato T ne moram spraviti čisto na 0, da bi program nakoncu psrejel samo poteze, ki pot skrajšajo.

3 Zaključek

Pri tej nalogi smo spoznali in uporabljali metropolisov algoritem. Pri zadnji nalogi sem ta algoritem še malo dodelal in tekom tega spreminjal temperaturo oziroma količino, ki vpliva na verjetnost za sprejem "slabe" poteze, saj me je ta postopek pripeljal do najboljšega rezultata. Tak način bi vrjetno dobro deloval tudi pri prvi nalogi, a te nisem spreminjal.