Koda v python-u 3.7, uporabljam pa še numpy in numbo.

Ne dela mi klasični del, problem pa mislim da je v dinamiki. Poskusil sem dobiti enake rezultate kot v https://journals.aps.org/pre/abstract/10.1103/PhysRevE.96.062144, torej ista dinamika in začetno stanje. Algoritem bom opisal po korakih. Indeksi q, p tečejo $0, \ldots, N-1$.

1. Začetno $N \times N$ stanje je OK (torej en nesklopljen sistem z enim q in p), se ujema z kvantnim, je torej 2D gaussovka s $\sigma = \sqrt{\hbar}$. Tako združim dve enaki NxN stanji v en sam $N^2 \times N^2$ sistem

$$\boldsymbol{\rho}_{N^2\times N^2}[q_1 * N + p_1][q_2 * N + p_2] = \boldsymbol{\rho}_1[q_1][p_1] * \boldsymbol{\rho}_2[q_2][p_2], \tag{1}$$

kar se ujema z s kvantnim združenjem stanj (sem preveril z Wignerjevo funkcijo.)

2. Nato si naračunam "matrike za dinamiko" $\boldsymbol{Ar_1}$ in $\boldsymbol{Ar_2}$ in sicer tako:

$$\boldsymbol{Ar_1}[q_1 * N + p_1][q_2 * N + p_2] = \tilde{q}_1 * N + \tilde{p}_1, \tag{2}$$

$$\boldsymbol{Ar_2}[q_1 * N + p_1][q_2 * N + p_2] = \tilde{q}_2 * N + \tilde{p}_2, \tag{3}$$

kjer sta

$$\begin{bmatrix}\tilde{q}_1/N \\ \tilde{p}_1/N\end{bmatrix} = \boldsymbol{M}\begin{bmatrix} q_1/N \\ p_1/N + \epsilon(q_1/N) + \kappa(q_1/N, q_2/N)\end{bmatrix}. \tag{4}$$

in simetrično za $\tilde{q}_2, \tilde{p}_2$. $\boldsymbol{M}$ je matrika, ki spravi dinamiko v hiperbolični ali pa eliptični režim (enako kot v članku). Je pa treba še poskrbeti za to, da $\tilde{q}/N, \tilde{p}/N$ ostaneta na torusu

$$\tilde{q} := \mathrm{int}((\tilde{q}/N \;\%\!= 1.0) * N), \tag{5}$$

isto za ostale.

3. Dejanska dinamika poteka tako, da vzamem nov, zgolj z ničlami napolnjen $\tilde{\boldsymbol{\rho}}_{N^2\times N^2}$ in ga uporabim kot

$$\tilde{\boldsymbol{\rho}}_{N^2\times N^2}[\boldsymbol{Ar_1}[q_1 * N + p_1][q_2 * N + p_2]][\boldsymbol{Ar_2}[q_1 * N + p_1][q_2 * N + p_2]] \mathrel{+}= \boldsymbol{\rho}_{N^2\times N^2}[q_1 * N + p_1][q_2 * N + p_2], \tag{6}$$

torej verjetnosti samo premikam in če pridejo na isto mesto, se seštejejo, zato imam tudi $\boldsymbol{\rho}_{N^2\times N^2}$ normiran na 1, torej $\sum_{i,j} \boldsymbol{\rho}_{N^2\times N^2}[i][j] = 1$. 4. Izračun entropije je preprost. $\boldsymbol{\rho}_{N^2\times N^2}$ dam v SVD algoritem:

$$\boldsymbol{u}, \boldsymbol{s}, \boldsymbol{v^h} = \mathrm{np.linalg.svd}(\boldsymbol{\rho}^{0.5}_{N^2\times N^2}), \tag{7}$$

kjer potem seštejem vrednosti v s kot $-\sum s^2[i] \log s^2[i]$, da dobim entropijo. $\boldsymbol{\rho}_{N^2\times N^2}$ je pod korenom, da imam isto definicijo za entropijo kot v članku: $\sum \boldsymbol{\rho}^2_{N^2\times N^2} = 1$.

5. Možni problemi: N v članku je sod, sam pa sem zaradi Wignerjeve funkcije izbral liho. Potem vpliv absolutnega položaja gaussovke na dinamiko....