Many-Body Quantum Systems

6^{th} week assignment

December 13, 2021

Campesan Giulia

Università degli Studi di Padova

- Consider a quantum system formed by N **subsystems**, the state of each one ψ_i belonging to a D-dimensional Hilbert space $\psi \in \mathcal{H}^D$.
  Given a basis $\{|\alpha_1\rangle, \ldots, |\alpha_d\rangle\} = \{|0\rangle, \ldots, |d-1\rangle\}$ of $\mathcal{H}^D$, the wave-function of the total system $\Psi(\psi_1, \ldots, \psi_N) \in \mathcal{H}^{d^N}$ reads

$$\Psi(\psi_1, \ldots, \psi_N) = \sum_{\alpha_1, \ldots, \alpha_N} C_{\alpha_1 \ldots \alpha_N} |\alpha_1\rangle \otimes \cdots \otimes |\alpha_N\rangle = \sum_{i=0}^{d^N-1} C_i |i\rangle$$

- If $\Psi(\psi_1, \ldots, \psi_N)$ is **separable** i.e.

$$\Psi(\psi_1, \ldots, \psi_N) = \bigotimes_{i=0}^{N} |\psi_i\rangle = \bigotimes_{i=0}^{N} \sum_{\alpha_i} C_{\alpha_i} |\alpha_i\rangle$$

  it can be described by $\mathbf{N} \cdot \mathbf{d}$ coefficients

- Given a generic pure state ψ describing a system, its **density matrix** ρ reads

$$\rho = |\psi\rangle \langle\psi| = \sum_{\alpha_i} \sum_{\alpha_j} C_j^* C_i |\alpha_i\rangle \langle\alpha_j|$$

- To retrieve the density matrix of the k_{th} sub-system, we have to **trace-out** all the other degrees of freedom

$$\rho_k = Tr_1 \dots Tr_{k-1} Tr_{k+1} \dots Tr_N \rho \tag{1}$$

- In the hypothesis that the system is **bipartite** into subsystem A and B of size d we can express the **partial trace** operation in (1) leading to the **reduced density matrix** of system A as

$$\rho_A = Tr_B \rho = \sum_i \left(\mathbb{1}_A \otimes \langle i|_B\right) \cdot \rho \cdot \left(\mathbb{1}_A \otimes |i\rangle_B\right)$$

- And, similarly, for system B

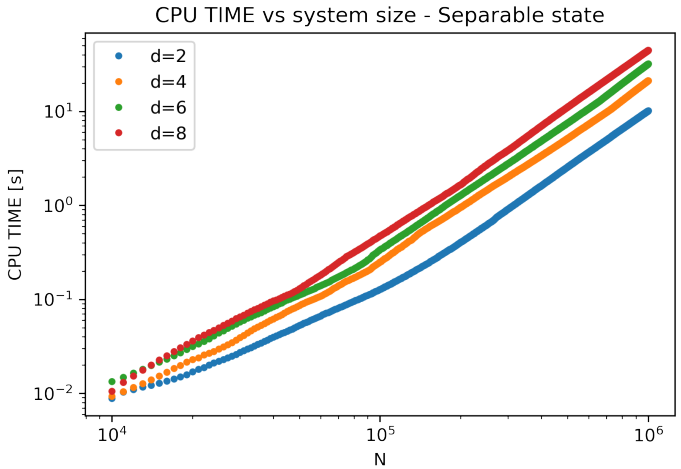
$$\rho_B = Tr_A \rho = \sum_i \left(\langle i|_A \otimes \mathbb{1}_B\right) \cdot \rho \cdot \left(|i\rangle_A \otimes \mathbb{1}_B\right)$$

Code

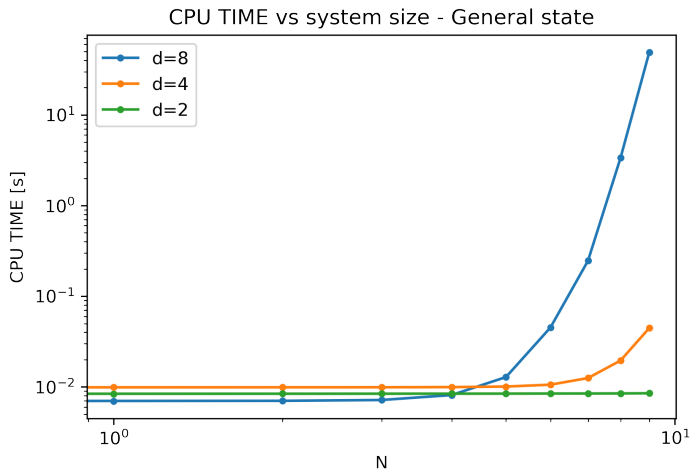
```
function reduced_density_matrix(rho, system, dim_A, dim_B) result(reduced_rho)

    double complex :: rho(:, :)
    double complex, allocatable :: reduced_rho(:,:)
    character(1) :: system
    integer ii,  jj, dim_A, dim_B
    real(8), allocatable :: id(:,:), ket(:,:), bra(:,:), bra_id(:,:), ket_id(:,:)

    if (system=='A') then
        allocate( reduced_rho(dim_B, dim_B) )
        allocate(ket(dim_A, 1))
        allocate(bra(1, dim_A))

        reduced_rho = 0
        id = ID_matrix(dim_B)


        do ii=1, dim_A
            ket(:, 1) = versor(dim_A, ii)
            bra(1, :) = versor(dim_A, ii)

            allocate(bra_id(size(bra, 1)*size(id, 1), size(bra, 2)*size(id, 2)))
            allocate(ket_id(size(ket, 1)*size(id, 1), size(ket, 2)*size(id, 2)))

            call kronecker_product(bra, id, bra_id)
            call kronecker_product(ket, id, ket_id)

            reduced_rho = reduced_rho + MATMUL (MATMUL(bra_id, rho), ket_id)

            deallocate(bra_id) ; deallocate(ket_id)

        enddo
```

Figure: *reduced_density_matrix* function

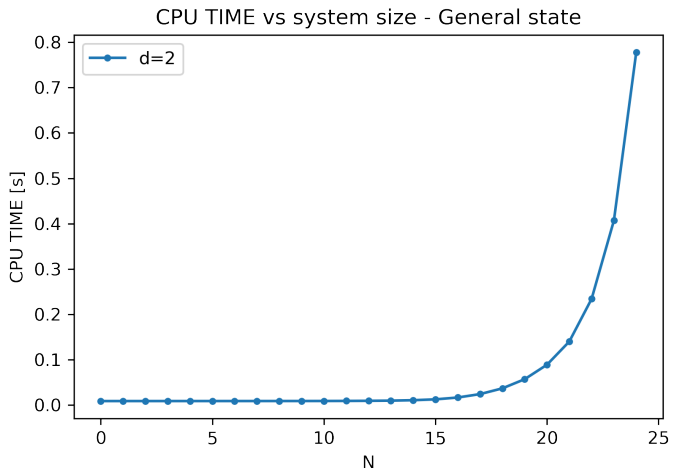
Results



(a) scaling of cpu-time vs system size for separable many-body state



(b) scaling of cpu-time vs system size for generic many-body state



(c) scaling of cpu-time vs system size for separable many-body state, d=2

- computation of reduced density matrices ρ_A, ρ_B for $d = 2, N = 2$
- in this case is particularly easy to verify the result since given th total matrix

$$\rho = \begin{vmatrix} a & b & e & f \\ c & d & g & h \\ i & j & m & n \\ k & l & o & p \end{vmatrix}$$

then

$$\rho_A = \begin{vmatrix} a+d & e+h \\ i+l & m+p \end{vmatrix}, \qquad \rho_B = \begin{vmatrix} a+m & b+n \\ c+o & d+p \end{vmatrix}$$

```
(base) giulia@Air-di-Giulia week6 % ./dmat.out 2 2 .FALSE. 1
 state coefficients
( 0.073810 -0.294803i)
( 0.371076 +0.076118i)
( 0.452853 -0.418971i)
(-0.464968 -0.409078i)
 density matrix
( 0.092357 +0.000000i) ( 0.004949 -0.115013i) ( 0.156939 -0.102578i) ( 0.086278 +0.167268i)
( 0.004949 +0.115013i) ( 0.143492 +0.000000i) ( 0.136152 +0.189940i) (-0.203677 +0.116407i)
( 0.156939 +0.102578i) ( 0.136152 -0.189940i) ( 0.380612 -0.000000i) (-0.039170 +0.380060i)
( 0.086278 -0.167268i) (-0.203677 -0.116407i) (-0.039170 -0.380060i) ( 0.383540 -0.000000i)
 reduced density matrix A-tracing out B
( 0.235848 +0.000000i) (-0.046738 +0.013828i)
(-0.046738 -0.013828i) ( 0.764152 -0.000000i)
 reduced density matrix B-tracing out A
( 0.472969 -0.000000i) (-0.034221 +0.265047i)
(-0.034221 -0.265047i) ( 0.527031 -0.000000i)
(base) giulia@Air-di-Giulia week6 %
```

Figure: Computation of reduced density matrices