Composite quantum systems

Abstract

In this assignment a quantum system composed by N subsystems, each of them of dimension D, is analyzed. In particular the total wave function of the system is computed through a `FORTRAN` program, in both interacting and non interacting case. Moreover, fixed $N = 2$, another task is to compute the density matrix of a generic pure state and the reduced density matrix using both 'left' and 'right' subsystems.

Theory

Given a quantum system composed by N subsystems, each described by its wave function ψ_i with $i = 1, ..., N$, that lives in the $D-$dimensional Hilbert space $\mathcal{H}^D$, the total wave function of the system is a vector of dimension D^N. Generally, if

$$|\alpha_i\rangle \in \{|1\rangle, |2\rangle, ..., |D\rangle\} \tag{1}$$

is the basis of the i-th subsystem, the total wave function Ψ can be written as

$$|\Psi\rangle = \sum_{\alpha_1,...,\alpha_N} \psi_{\alpha_1,...,\alpha_N} |\alpha_1\rangle|\alpha_2\rangle...|\alpha_N\rangle \tag{2}$$

where $\psi_{\alpha_1,...,\alpha_N}$ are the components of Ψ. The latter, of size D^N, can be considered in a "compressed version", doing a *Mean-Field approximation*. Physically this means to consider non interacting subsystems: the total wave function can be written as the tensor product of the single subsystem's wave functions (separable state)

$$|\Psi_{MF}\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes ... \otimes |\psi_N\rangle = \prod_{i=1}^{N} \sum_{j=1}^{D} A_j |\alpha_j\rangle \qquad with \qquad \sum_j |A_j|^2 = 1 \tag{3}$$

So in this case only $D \cdot N$ instead of D^N coefficients are needed to describe the whole system. This means, from a computational point of view, less memory to allocate.
If Ψ is a pure state, its density matrix is nothing else than a projector

$$\rho = |\Psi\rangle\langle\Psi| \tag{4}$$

The following properties, for a density matrix, hold:

- It is Hermitian $\rho = \rho^\dagger$
- Its trace is unitary, $\mathrm{Tr}[\rho] = 1$
- If it is a projector (pure states), $\rho = \rho^2$

In general, if one knows the density matrix of a composite quantum system composed by two subsystems, A and B, the density matrix of one of the two systems, let's say A, is given by the following , known as *partial trace*

$$\rho_A = Tr_B(\rho_{AB}) = \sum_\beta \langle\beta|\rho_{AB}|\beta\rangle \tag{5}$$

where β runs over the possible states that B can assume. In the case of a separable system the reduced density matrix that remains when we trace out a subsystem can be obtained also with the tensor product of the remaining subsystems.

Code development

In the program, called `ex8`, is included the usual `MODULE 'DEBUGMOD'`, used also in previous exercises, to debug a generic code.
Firstly, at each run the program asks the user the number of subsystems N and their dimension D. The aforementioned `MODULE 'DEBUGMOD'`, activated if the logical flag `'debug_flag'` is set to `TRUE`, is used to check if the inserted quantities are positive, otherwise a warning message is printed.

```
CALL DEBUG(debug_flag,N,N .LE. 0, "Insert a positive number of subsystems!")
CALL DEBUG(debug_flag,D,D .LE. 0, "Insert a positive dimension!")
```

Listing 1: 'DEBUG' check

Not separable systems

The wave function is a vector of D^N components, that are complex numbers drawn randomly from the uniform distribution in the interval [-1,1]. Then the wave function is normalized in the way shown in listing 2. Once computed this, the density matrix is calculated, applying the definition (4). Using the intrinsic function `'MATMUL'` also the square of the density matrix is computed.

```
! wave function vector in the not separable case (vector of D**N components)
DO ii=1,D**N
   CALL RANDOM_NUMBER(re_part) ! re_part is a random number ~ U([0,1])
   CALL RANDOM_NUMBER(im_part) ! im_part is a random number ~ U([0,1])
   psi_not_sep(ii) = CMPLX(2*re_part-1.,2*im_part-1) ! the component are complex
  values in the interval [-1,1]
END DO

psi_not_sep(:) = psi_not_sep(:) / SQRT(SUM(ZABS(psi_not_sep)**2)) ! normalization
```

Listing 2: `Wave function construction`

The program writes on the file `'not_separable_case.dat'`, shown in the next section, the

- Norm of the wave function
- The wave function of the system, both the real and imaginary part of the D^N components
- The density matrix
- The trace of ρ^2, that it is expected to be 1, since Tr[ρ] has to be 1 and ρ=ρ^2.
- A logical flag that tells if the density matrix is Hermitian.

Separable systems

The procedure is similar to the one described before and the file produced, named `'separable_case.dat'`, has the same structure. However in this case only $D \cdot N$ values are stored, initialized using uniform random numbers in the interval [-1,1], like before. The trick that has been used here is the fact that every coefficient is indexed by a number in base D, written using N digits.

Reduced density matrix

Finally the procedure to evaluate the reduced density matrix is implemented. For this task the code works only for the case $N = 2$. One can choose the particle/system to trace out by inserting 1 or 2 when the program asks for it. As can be seen below, the reduced density matrix is computed for the separable case, implementing the formula (5) and in a more directed way.
The procedures, as expected from the theory, are equivalent and their results are printed in the file `'reduced_matrix.dat'`.

```
        DO jj=0,D**(N-1)-1
           DO ii_red=0,D-1
              index_i = ii*D**(subsys-1)+ii_red*D**(N-subsys) + 1
              index_j = jj*D**(subsys-1)+ii_red*D**(N-subsys) + 1
              rho_reduced(ii+1,jj+1) = rho_reduced(ii+1,jj+1) + rho_sep(index_i,
   index_j)
           END DO
        END DO
```

Listing 3: Reduced matrix with the *partial trace* formula

```
      DO ii=1,D
         DO jj=1,D
            WRITE(3,"(F15.8,F13.8)",advance="yes") psi_sep_temporary((N-subsys)*D+jj
    )*CONJG(psi_sep_temporary((N-subsys)*D+ii))
         END DO
      END DO
```

Listing 4: Reduced matrix in a more direct way (only separable case!)

Results

In this section the `'.dat'` files described before are shown, for $N = 2$ and $D = 2$. In both cases, with a certain precision, we can see that the trace of ρ^2 is unitary (this means that the $\mathrm{Tr}[\rho] = 1$, as required for a density matrix, and that $\rho = \rho^2$, as required for a projector).

```
 Norm of the wave function   1.0000000000000000
 Wave function of the system
-0.46035128  -0.13080280
 0.07686678   0.33519041
 0.43292498  -0.44722766
 0.44216281  -0.26412430
 Density matrix:
    0.22903267   0.00000000   -0.07922957   0.14425094   -0.14079894  -0.26250962   -0.16900202  -0.17942609
   -0.07922957  -0.14425094    0.11826111   0.00000000   -0.11662887   0.17948925   -0.05454430   0.16851112
   -0.14079894   0.26250962   -0.11662887  -0.17948925    0.38743662   0.00000000    0.30954702  -0.08340143
   -0.16900202   0.17942609   -0.05454430  -0.16851112    0.30954702   0.08340143    0.26526960   0.00000000
 Trace of density matrix^2:   1.0000000000000000
 Is the density matrix Hermitian? T
```

Figure 1: `'not_separable_case.dat'` file.

```
 Wave function of the system
 0.23753033   0.02675128
 0.28281728  -0.66234064
 0.01565233  -0.20456095
-0.57680074  -0.22224693
Density matrix:
    0.05713629   0.00000000    0.04945922  -0.16489172   -0.00175436  -0.04900815   -0.14295306  -0.03736023
    0.04945922   0.16489172    0.51868073   0.00000000    0.13991578  -0.04748620   -0.01592604  -0.44489385
   -0.00175436   0.04900815    0.13991578   0.04748620    0.04209018   0.00000000    0.03643477  -0.12146959
   -0.14295306   0.03736023   -0.01592604   0.44489385    0.03643477   0.12146959    0.38209280   0.00000000
Trace of density matrix^2:  0.99999999999999978
 Is the density matrix Hermitian? T
```

Figure 2: `'separable_case.dat'` file.

The figure 3 shows that the results of the procedures described at the end of the previous section are equivalent.

```
 Evaluated reduced matrix
     0.09922647   0.00000000        0.08589399   0.28636131
     0.08589399  -0.28636131        0.90077353   0.00000000

 Expected reduced matrix
     0.09922647   0.00000000        0.08589399   0.28636131
     0.08589399  -0.28636131        0.90077353   0.00000000
```

Figure 3: `'reduced_matrix.dat'` file.

Comment on the maximum N

Fixed $D = 2$, the program starts to give this error when $N = 15$. In fact the required memory to allocate a `COMPLEX*16` vector of 2^{15} components exceeds the avaiable memory of my laptop, that is 2 GB.

```
camilla@camilla-KPL-W0X:~/Desktop/quantum info/camilla_quaglia_Ex8$ ./Ex8-Quaglia-CODE.x
 Insert the number of subsystems:
15
 Insert the dimension of the subsystems:
2
Operating system error: Cannot allocate memory
Allocation would exceed memory limit
```

Figure 4: Memory error

Self-Evaluation

This exercise was the occasion to practise with composite quantum systems in a more 'hands on way' than the usual study in quantum mechanics. Moreover to implement the formulas one has to have a real understanding of their meaning, and this is useful even to study the topic.