

密度矩阵

表示

- 密度矩阵 ρ 是一个复矩阵，当给定模式数为 M ,每个模式的维度为 n_i 时，一个密度矩阵应该表示为一个 2^M 维的矩阵，其中第 $2M-1$ 和 $2M$ 维度的区间大小为 $n_{M-1} * n_{M-1}$ ；

换言之，这个密度矩阵的总元素数量是 $N = \prod_{i=0}^{M-1} n_i^2$ 。且每个元素是复数。

为了表示方便，我们一般用狄拉克左右矢来表示某个特定的元素，例如一个密度矩阵的元素为

$$\rho_{k_0 l_0, k_1 l_1, \dots, k_{M-1} l_{M-1}} |k_1\rangle \langle l_1| \otimes \dots \otimes |k_{M-1}\rangle \langle l_{M-1}|$$

表示的是矩阵索引为 $[k_0, l_0, k_1, l_1, \dots, k_{M-1}, l_{M-1}]$ 的元素值为 $\rho_{k_0 l_0, k_1 l_1, \dots, k_{M-1} l_{M-1}}$ 。

- 其中 $0 \leq k_i, l_i \leq n_i - 1$ ，如果超出该范围，那么该索引位置上的**值为零**（不是空）。
- 密度矩阵的复对称性允许我们只存储一半的数据，因为

$$\rho_{l_0 k_0, l_1 k_1, \dots, l_{M-1} k_{M-1}} = \rho_{k_0 l_0, k_1 l_1, \dots, k_{M-1} l_{M-1}}^*$$

（即其对角线上一定是实数）

演化及稳态

- 求解一个玻色系统的稳态解是我们程序的目标。某个特定密度矩阵的元素的演化，由哈密顿量和耗散量决定，这两者都会导致密度矩阵的特定元素间的耦合，从而形成一组 $N+1$ 个方程 N 个未知数的线性方程，其中多出来的一个方程是迹条件，即对角元素加起来为1。但这个并不重要，因为其他 N 个方程都是齐次方程，求出非平凡解后整体除一个因子就能满足迹条件了，所以最后一个方程并不重要。
- 下面开始阐述如何推导密度矩阵的演化方程。

算符的作用

- 我们只会有三种基本算符，常算符 \hat{I} ,某个模式的产生算符 \hat{a}^\dagger 和湮灭算符 \hat{a}

$$(\hat{a}^\dagger)^\dagger = \hat{a}$$

- 他们对矩阵元的作用如下(此处为了规避引入量子力学的一些内容，所以阐述不太严谨)

常算符

- 无论作用在矩阵元左边又或是右边矩阵元都不变，一般和系数相乘起到简单的乘数作用即

$$\begin{aligned}\hat{I} \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} &= \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} \\ \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} \hat{I} &= \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}\end{aligned}$$

- 换言之矩阵元和普通数字相乘，就是普通的可交换乘法

$$\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} a = a \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}$$

产生算符

- 作用在左边,对应模式的行索引减一

$$\hat{a}_i^\dagger \rho_{l_0 k_0, \dots, l_i k_i, \dots, l_{M-1} k_{M-1}} = \rho_{l_0 k_0, \dots, (l_i-1) k_i, \dots, l_{M-1} k_{M-1}}$$

- 作用在右边，对应模式的列索引加一且乘上列索引加一

$$\rho_{l_0 k_0, \dots, l_i k_i, \dots, l_{M-1} k_{M-1}} \hat{a}_i^\dagger = (k_i + 1) \rho_{l_0 k_0, \dots, l_i (k_i+1), \dots, l_{M-1} k_{M-1}}$$

湮灭算符

- 作用在左边，对应模式的行索引加一，且乘上列索引加一

$$\hat{a}_i \rho_{l_0 k_0, \dots, l_i k_i, \dots, l_{M-1} k_{M-1}} = (l_i + 1) \rho_{l_0 k_0, \dots, (l_i+1) k_i, \dots, l_{M-1} k_{M-1}}$$

- 作用在右边，对应模式的列索引减一

$$\rho_{l_0 k_0, \dots, l_i k_i, \dots, l_{M-1} k_{M-1}} \hat{a}_i = \rho_{l_0 k_0, \dots, l_i (k_i-1), \dots, l_{M-1} k_{M-1}}$$

算符在程序内的表示

- 按照上次我和可可做的程序

0用来表示常算符，2*i*用来表示*i*模式的产生湮灭算符，2*i* - 1用来表示产生算符。

- 但在输入的时候为了方便，我们一般用大写字母A表示第一个模式的产生算符，小写字母a表示第一个模式的湮灭算符；B表示第二个模式，依次类推。空格表示常算符(一般用不到)，输入进去后再转成数字存储。

演化方程

- 演化方程可以被简单地写为

$$\frac{d}{dt} \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} = i [\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}, \hat{H}] + \sum \gamma_j \mathcal{L}_{\hat{O}_j} [\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}]$$

其中

$$[\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}, \hat{H}] = \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} \hat{H} - \hat{H} \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}$$

$$\mathcal{L}_{\hat{O}_j} [\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}] = (2\hat{O}_j \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} \hat{O}_j^\dagger - \hat{O}_j^\dagger \hat{O}_j \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} - \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} \hat{O}_j^\dagger \hat{O}_j)$$

γ_j 是一个复系数， \hat{O}_j 是某个模式的产生算符或者湮灭算符。

哈密顿量的处理

- 哈密顿量 $\hat{H} = \sum_j C_j [\prod (\hat{a}_j^\dagger)^{n_j} (\hat{a}_j)^{m_j}]$, C_j 第*j*项的复系数，根据上述推导，当作用在左边时

$$\hat{H} \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} = \sum_j C_j \frac{(l_j + m_j)!}{l_j!} \rho_{l_0 k_0, \dots, (l_j+m_j-n_j) k_j, \dots, l_{M-1} k_{M-1}}$$

右边同理。

- 输入的时候一般是二维列表的形式，即一项对应一个系数地输入，比如

$$[[3, AAa], [5, BBbCc], \dots, [10, aDDdd]] \Rightarrow 3\hat{a}_1^\dagger \hat{a}_1^\dagger \hat{a}_1 + 5\hat{a}_2^\dagger \hat{a}_2^\dagger \hat{a}_2 \hat{a}_3^\dagger \hat{a}_3 + \dots + 10\hat{a}_1 \hat{a}_4^\dagger \hat{a}_4^\dagger \hat{a}_4 \hat{a}_4$$

- 由于系数可能会在计算中更新，为了不重复推导表达式，系数一般单独用一个数组存起来，最后计算时再乘上。

- 存储推导表达式的时候，其实从上面可见把每一项的 $[j, [n_j, m_j]]$ 存起来就行了，计算时根据上面表达式复现就行。 $(\mathcal{L}_{\hat{O}_j})$ 的处理也是一样的)

稳态表达式的存储

- 稳态就是把上述演化方程的左边设置成零。
- 一步是找出索引不变的那一项即 $\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}$ ，然后用链表或者数组的形式把它的系数存起来，即把 $[C_{address}, j, [n_j = m_j]]$ 拿一个链表存起来， $C_{address}$ 是系数的地址（因为系数是单独拿数组存的）。然后把这些项扔到表达式的左边，**即系数全部乘上一个负号。**
- 其他项也是把索引相同的项汇合到一起，对每个索引项，拿链表或数组来存它的乘子表达式 $P(l_0^i, k_0^i, \dots, l_{M-1}^i, k_{M-1}^i)$
- 然后再把这些**不同的索引项**用一个链表存起来，用来表示当前这个索引矩阵元 $\rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}}$ 会和哪些矩阵元耦合在一起（这些耦合的矩阵元称为该矩阵元的邻居，即把它的邻居都存起来）。

可以看到一个显然的事实，这个表达式是对所有矩阵元通用的，所有以函数的形式存一份就够了，这个稳态表达式可以形象地写为

$$P(l_0, k_0, \dots, l_{M-1}, k_{M-1}) \rho_{l_0 k_0, \dots, l_{M-1} k_{M-1}} = \sum_{i \in Neighbors} P(l_0^i, k_0^i, \dots, l_{M-1}^i, k_{M-1}^i) \rho_{l_0^i k_0^i, \dots, l_{M-1}^i k_{M-1}^i}$$