ITPO 2019, Problem 3: Disentanglement algorithms

April 15, 2019

Problem Statement:

Suppose you have a chain of N spins placed adjacent to one another around a circle, as in Figure 1. Some of these spins may be entangled within M subchains of indefinite length, where $1 \leq M \leq N$. You are provided with the density matrix ρ corresponding to the system.

1. Construct an efficient algorithm which will isolate the M subchains of the system. Assume that these subchains will be visible in the form of ρ within a finite number b of bases for your Hilbert space. Include an analysis of the complexity of your algorithm. Do so assuming that you have crafted this spin chain in a laboratory so that entangled subchains will consist only of adjacent spins.

2. A competing scientist sees that you are close to a result on your spin chains breakthrough paper and, in a dastardly attempt to jeopardize your grant funding, decides to rearrange the spins in your apparatus. Thankfully, this evildoer does so without disrupting the entanglement of the spins. Modify your algorithm to relax the assumption that subchains consist of adjacent spins. You may now encounter troubling exponential runtimes, so try your best to reduce them.

Rubric:

	Criterion	Point value	Total
Part A:	Reasonable approach	2	12
	Block-diagonalizes rho / block operations for submatrices	8	
	Accurate complexity analysis	2	
Part B:	Reasonable modification of Part A	6	8
	Accurate complexity analysis	2	

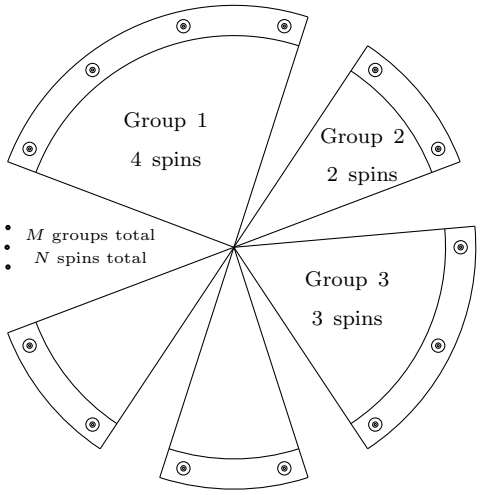

Figure 1: Schematic for Exercise 3.

Solution:

Not so much is known about this specific computational task, so an optimal solution may still be out there. We could not identify a polynomial-time algorithm to achieve the disentanglement, but that does not mean that one does not exist.

We begin our solution with an intuitive approach. An entanglement structure like the one in Part (A) imposes the following structure on the Hilbert space for our system:

$$\mathcal{H}_A = H_{A_1} \oplus H_{A_2} \oplus \cdots \oplus H_{A_M} \tag{1}$$

For some number of blocks M. This means our density matrix will be block-diagonal. In other words, our Hilbert space will contain M invariant subspaces,[1] which implies the existence of M projection matrices P_i satisfying:

$$P_i \, \rho \, P_i = \rho \, P_i \qquad\qquad i = 1, \ldots, M \tag{2}$$

We will block diagonalize ρ by a search through all possible projection matrices. This is the reason we allowed solutions which specified a finite number of relevant bases b in Hilbert space, because otherwise we would have to search through a continuum of projection matrices. The worst-case complexity of this algorithm will be the number of partitions of N objects into M boxes, multiplied by the number of bases we have to check: $\mathcal{O}(b \cdot M^N)$, which is exponential in N.

Part (B) utterly ruins the runtime. The most straightforward way we found to block-diagonalize ρ is to perform the same search over all possible spin groups, where spins in each group are not necessarily adjacent. We thus need to repeat our algorithm $N!$ times (the number of possible permutations of the N spins). One can speed up the worst-case scenario by a smart use of recursion, but the improvements are still limited.

[1] We recommend reading H. Georgi's *Lie Algebras in Particle Physics*, Chapter 1 for more clarity.

Another solution involves calculating the *entanglement entropy* of different blocks of spins with respect to the rest of the system, but this will have a similar runtime to the solution above (it is equivalent).

We end our analysis by eliminating the naïve approaches. Simply diagonalizing ρ might seem to be enough, and would lead to an $\mathcal{O}(N^3)$-complexity algorithm, but this does not work in general because if two eigenvectors in different blocks happen to share an eigenvalue, we can mix them together in an arbitrary proportion.