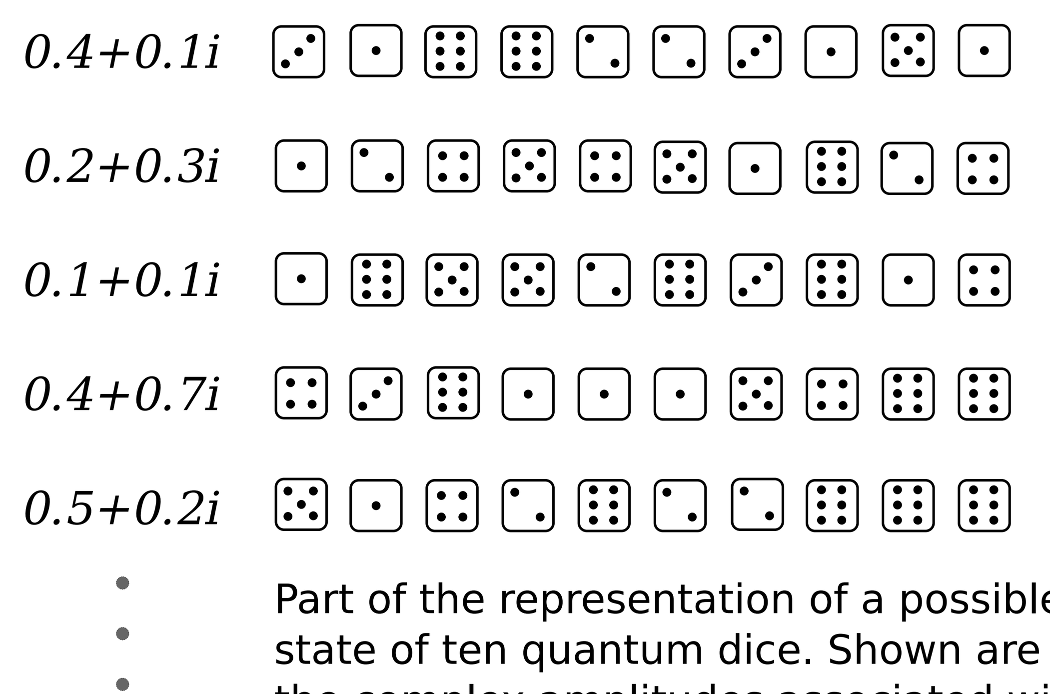
Gregory Crosswhite
University of Washington

Matrix Product States

An Excalibur in the quest for a self-correcting quantum memory

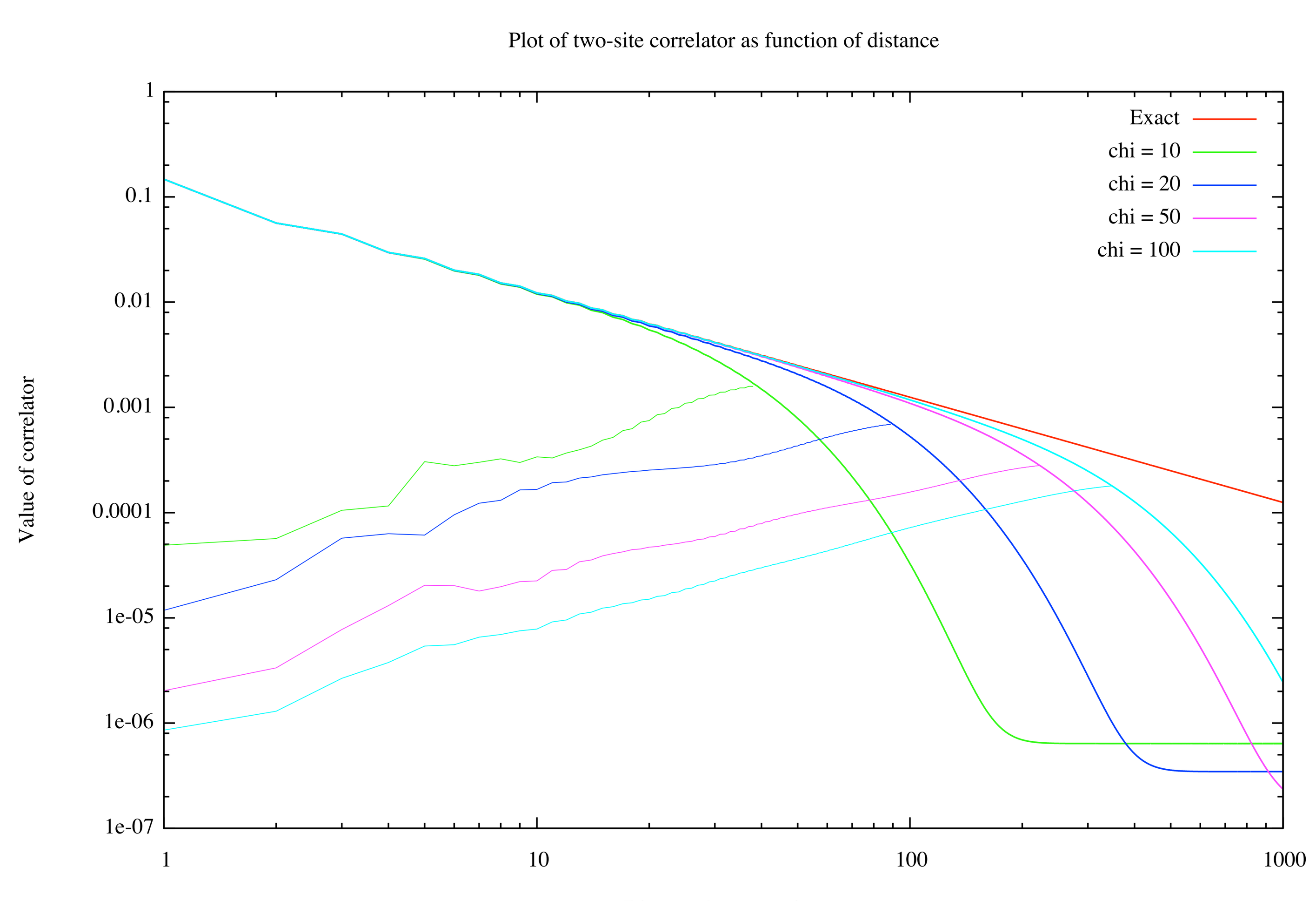
The Challenge

One of the holy grails of quantum error correction theory is to design a practical self-correcting memory: a physical system which is naturally robust against loss of quantum information, and so needs no active error correction process. The quest to do this is vexed by the presence of entanglement, which forces us to work in an exponentially large state space, such as the one illustrated to the right. Fortunately, many quantum systems are susceptible to attack by matrix product states (1D) and projected entangled-pair states (2+D), which serve to cut the working space down to a more manageable size while still capturing essential properties of the system. In this poster we overview these techniques, and present recent results of applying them to the Haldane-Shastry model.



Part of the representation of a possible state of ten quantum dice. Shown are the complex amplitudes associated with 5 of the 60,466,176 possible configurations of the system.

The Battle of Haldane-Shastry



Armed with matrix product states, we attacked the Haldane-Shastry model, a one-dimensional spin lattice featuring a long-range antiferromagnetic spin interaction whose strength falls off with the square of the distance. We computed a translationally invariant matrix product state approximation to the ground state. Our code is (as far as we know) the first matrix product based to do this. Previous algorithms have been limited to modeling systems with short-ranged interactions, but we employ a trick that allows us to approximate any potential that can be systematically approximated by a sum of decaying exponentials by writing the (approximated) hamiltonian as a matrix product operator.

Plotted on the left is the two-point correlator for our computed ground state. As you can see, our approximation does a very good job of reproducing this correlator to long distance. Furthermore, as we increase "chi" (the number of vertices in our diagram), we get a better result, so we have a systematic way to improve our approximation.

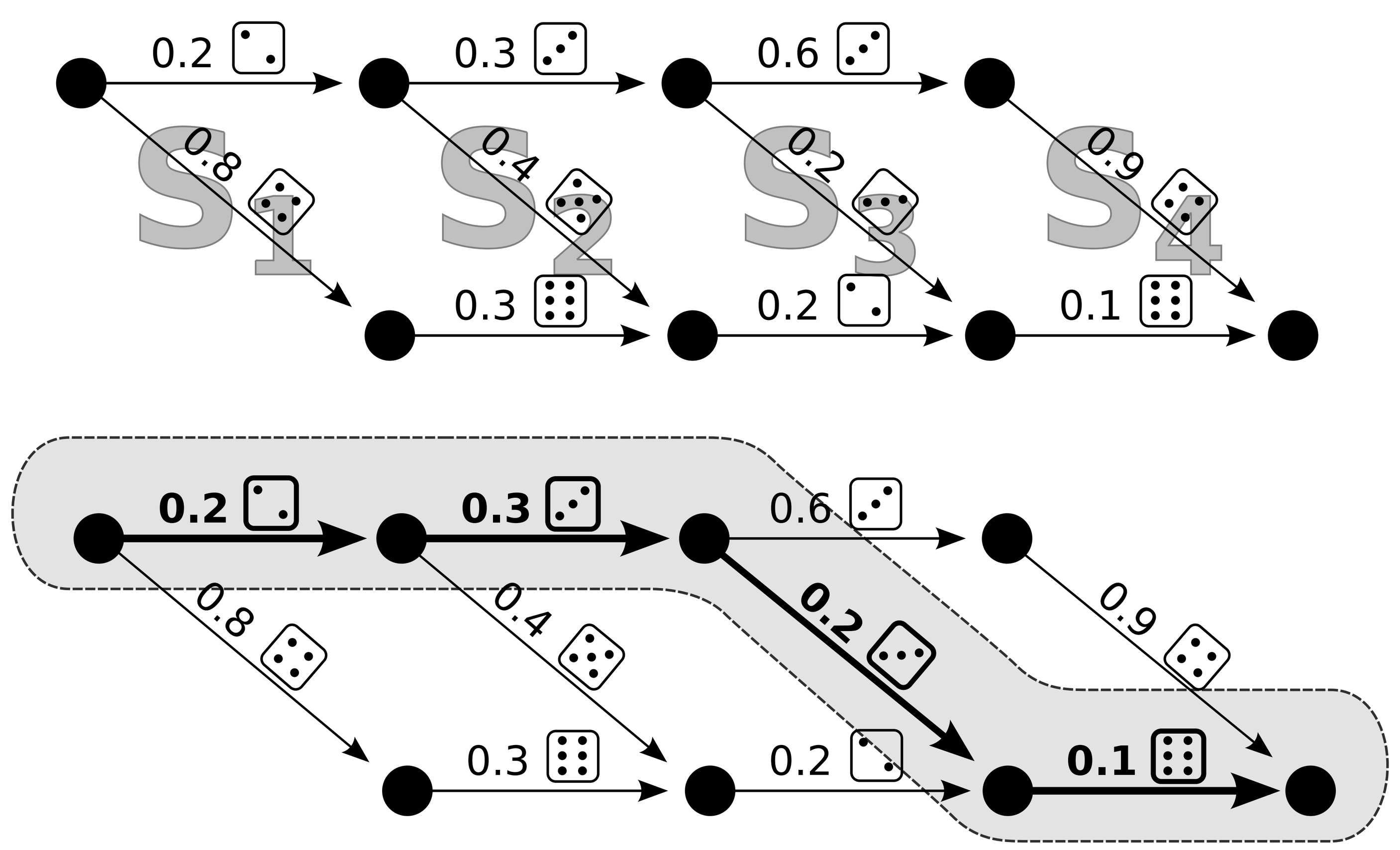
The Weapons

Matrix product states (for a four-site system of quantum dice) are states of the form,

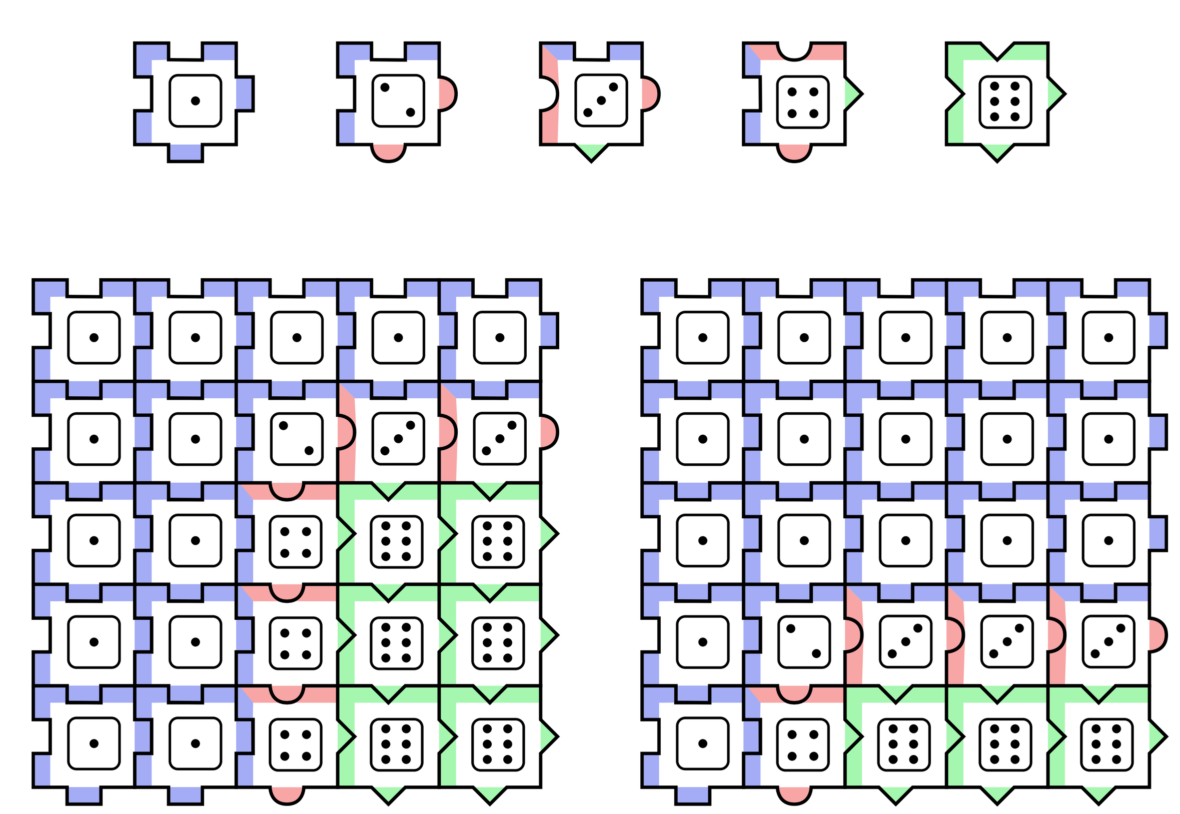
$$\psi^{\alpha\beta\gamma\delta} = \sum_{ijk} (S_1)_i^{\alpha} (S_2)_{ij}^{\beta} (S_3)_{jk}^{\gamma} (S_4)_k^{\delta}$$

where the left-hand side is a tensor denoting the amplitude of a given roll, and the right-hand side is a matrix product decomposition of this tensor. The structure of these states is illustrated in the diagrams to the right. Each of the four sets of edges (S through S) characterizes one of the four dice; the die outcome corresponds to the Greek index of the S tensor and the vertices correspond to the Roman indices. The amplitude of a particular roll is given by the product of the edge weights of a walk through the diagram containing that roll (or zero if there is no such walk). For example, as shown on the right, the weight of rolling 2, 3, 3, 6 is (0.2)(0.3)(0.2)(0.1)=0.0012.

Entanglement is captured by the vertices, which allow each die to "communicate" with its neighbors and thus attain some correlations. The number of indices may be increased arbitrarily to allow an arbitrary amount of entanglement to be modelled in the system.



One particular allowed roll of our line of dice, corresponding to a "walk".



Two allowed rolls of our grid of dice, assembled from the pieces above. Note that all possible rolls may only have a single "2".

For systems with interactions in two or higher dimensions, it is useful to have a form of representation that allows for "communication" in multiple directions. Projected entangled-pair states are the natural generalization of matrix product states that allow this to happen.

If a system is two-dimensional and translationally invariant, the projected entangled-pair state can be visualized as a set of puzzle pieces as shown to the left. The different shapes/colors are analogous to the vertices in the matrix product diagrams and can be thought of as "signals" that each site receives from above and to the left and transmits down and to the right. Each piece will in general have a complex weight, but these have been omitted to save space.

The weight of a particular roll of our "grid" of dice is given by the product of the weights on the pieces needed to assemble it.