# My research: quantum computing and information

Daniel Malz

November 20, 2023

My research focuses on understanding quantum computing devices and their applications, especially to solve many-body problems in physics.

I have always been captivated by computer science. In an effort to understand how problems can be broken down into elementary steps, it has emerged that there are different formal languages. The most powerful ones, Turing machines or $\lambda$-calculus, are computationally universal, which means they are equivalent and define what is and is not "computable". Interestingly, there are limits to what is computable. For example, undecidable problems do not admit an algorithmic solution. Problems can also be unsolvable in practise, if finding the solution takes too long. The complexity of an algorithm is the number of elementary steps needed to solve any problem of a given fixed size $N$. Algorithms for which the complexity grows (asymptotically) as a polynomial of $N$ are said to be efficiently solvable, and can typically be solved by throwing enough computational power at them (for example, multiplying matrices).

Unfortunately, many important problems do not fall into this family. For example, part of our job is to prove true statements. It is widely believed that searching for a proof of a bounded length $N$ has a complexity that grows exponentially with $N$ (otherwise $\mathsf{P} = \mathsf{NP}$), meaning that a computer cannot efficiently search for proofs. As a result, our jobs cannot straightforwardly be done by computers.

Ever since statistical physics was invented in the second half of the 19th century, a central theme of physics has been to understand how phases of matter emerge from the complexity of applying simple rules to many particles. Computers have played an important role as practical tools in statistical physics as they allow simulating complex systems. In turn, physics has inspired a number of important algorithms.

A snag in this success story has been that our world microscopically is described by quantum theory, which cannot be efficiently simulated by computers. The reason is that classical representations of quantum problems intrinsically scale exponentially with problem size. This problem has sparked the idea of a quantum computer—a computing device that follows the laws of quantum rather than classical mechanics. In 1994, Peter Shor showed that a quantum computer can find prime factors in polynomial time, unlike classical computers. This is very exciting, because it establishes that in spite of computational universality, not all computing models are equally powerful (if you are willing to believe that factoring is hard for classical computers). It also provides more support for the belief that simulating Nature microscopically requires a quantum computer in general and that there will always be problems we cannot solve on classical computers.

These fundamental questions are fascinating, but what makes the current times particularly exciting is that first quantum computers are being built, and scaling them up seems eminently possible. Thus, we may soon witness to a revolution in computational quantum many-body physics and related fields. This prospect motivates the search for fast and robust algorithms that can exploit these devices, especially for quantum simulation.[1]

My research centers in particular around finding, proposing, and understanding quantum algorithms to simulate quantum many-body systems. The two subfields of this general direction that I currently mainly work on concern (i) algorithms to prepare many-body states, and (ii) the robustness of these and other algorithms to errors.

The first problem, *state preparation*, is in some sense "Problem 0" of quantum simulation. A quantum algorithm might for example solve the problem of computing the time evolution of a state under a given Hamiltonian. To use this algorithm, one has to first prepare the state at the beginning of the time evolution. Conceptually, we can think of state preparation as "uploading" data to a quantum computer. It requires an efficient classical description of the state and an efficient algorithm to compile this description into native quantum operations.

The second problem is inspired by current devices, which do not posses error correction mechanisms and thus can only perform quantum computations with errors. Processes in Nature are robust against some types and amount of errors. Chemical reactions, for instance, predictably happen despite the molecules being constantly objected to random stochastic fluctuations of the environment. Similarly, materials retain their properties at finite temperature and in the presence of structural defects. Thus, when studying quantum algorithms for many-body problems, it may be the case that similar robustness can be identified. A better understanding of this will make the existing devices more usable in practise.

[1]A common misconception is that quantum computers are superior to conventional computers in all ways—this is emphatically not the case. Most things that can be done well with classical computers will not benefit from quantum computers. At the moment, it appears like quantum many-body simulation and perhaps cryptography are the most promising application of quantum computers. This said, it may well be that there are really cool applications of quantum computers that we do not know about yet. (Incidentally, simulation and cryptography were the first known applications of classical computers, too.)