**MPRI: Quantum Algorithms and Complexity (QUALCO)** **2025 - 2026**

# Lecture 3: Adiabatic quantum computation

*Lecturer: Simon Apers (apers@irif.fr)*

The quantum circuit model is a discrete model of quantum computation, and it has much appeal to computer scientists. Physicists however are more inclined to think about quantum dynamics as following Schrödinger's equation

$$\partial_t |\psi_t\rangle = -iH_t |\psi_t\rangle ,$$

where $H_t$ is the (potentially time-dependent) *Hamiltonian* of the quantum system. In the time-independent case ($H_t = H$), this solves to $|\psi_t\rangle = e^{-iHt} |\psi_0\rangle$. In physics, an important role is played by the ground state of the Hamiltonian $H$, corresponding to the eigenvector with the smallest eigenvalue.

The idea of adiabatic quantum computation is to (i) encode the solution of a problem in the ground state of a Hamiltonian, and (ii) use "adiabatic evolution" to prepare that ground state.

## 1 Quantum adiabatic algorithm

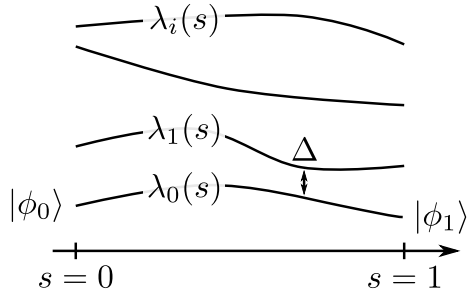
In its simplest form, the quantum adiabatic algorithm takes the following form:

- **Input:** initial Hamiltonian $H_0$ with an easy-to-prepare ground state $|\phi_0\rangle$, final Hamiltonian $H_1$ whose ground state $|\phi_1\rangle$ we wish to prepare
- **Evolution:** Prepare $|\psi_0\rangle = |\phi_0\rangle$, and let it evolve according to the Schrödinger equation

  $$\partial_t |\psi_t\rangle = -iH(t/T) |\psi_t\rangle ,$$

  where $H(s) = (1-s)H_0 + sH_1$ for $0 \leq s \leq 1$.
- **Output:** state $|\psi_T\rangle$

The algorithm succeeds if $|\psi_T\rangle \approx |\phi_1\rangle$. The reason why this algorithm should succeed at all is called the *quantum adiabatic theorem*. It says that if we start in the ground state of a "slowly-varying" Hamiltonian, then we stay in the ground state of that Hamiltonian. Clearly, as $T \to \infty$ then our Hamiltonian $H(t/T)$ will be slowly varying, and so the output state $|\psi_T\rangle$ should equal $|\phi_1\rangle$, the ground state of the final Hamiltonian.




A more quantitative statement is given by the following theorem. Let $\lambda_0(s) \leq \lambda_1(s) \leq \cdots \leq \lambda_{N-1}(s)$ denote the eigenvalues of $H(s)$. The theorem states that the evolution time depends on

the *spectral gap* of $H(s)$, which is the distance between the ground energy $\lambda_0(s)$ and the energy $\lambda_1(s)$ of the first excited state (see figure above).

**Theorem 1** (Quantum adiabatic theorem)**.** *Let the spectral gap of $H(s)$ be at least $\Delta$ for all $0 \leq s \leq 1$. For any $\epsilon > 0$, there exists*

$$T = \text{poly}\left(\frac{1}{\Delta}, \frac{1}{\epsilon}, \|H_0\| + \|H_1\|\right)$$

*such that $|\psi_T\rangle$ will be $\epsilon$-close to $|\phi_1\rangle$.*

## 2 Quantum adiabatic optimization algorithm

As an example of its usage, we describe the *quantum adiabatic optimization algorithm*, as proposed by Farhi, Goldstone, Gutman and Sipser [FGGS00]. Consider a classical optimization problem over $n$-bit strings, encoded by a function

$$h : \{0,1\}^n \mapsto \mathbb{R}.$$

Our goal is to find a string $x$ that minimizes $h(x)$. We can easily frame this as a ground state problem. Indeed, such a minimizer $x$ would correspond to a ground state $|x\rangle$ of the (diagonal) Hamiltonian

$$H_1 = \sum_{z \in \{0,1\}^n} h(z) \, |z\rangle \langle z| \, . \tag{1}$$

We then use the quantum adiabatic algorithm to try and prepare a ground state $|\phi_1\rangle$ of $H_1$. Such a state must be a linear combination of minimizers of $h$, and so measuring the state must return a minimizer of $h$.

It remains to specify the initial Hamiltonian $H_0$. A naive approach is to pick again a diagonal Hamiltonian such as $H_0 = I - |0^n\rangle \langle 0^n|$ or $H_0 = -\sum_j Z_j$, where $Z_j$ is shorthand for applying the Pauli $Z$-gate to the $j$-th qubit while leaving the other qubits invariant. Both Hamiltonians have a unique (and trivial to prepare) ground state $|0^n\rangle$.

**Exercise 1.** *What goes wrong if we try to apply the adiabatic theorem to the resulting Hamiltonian $H(s) = (1-s)(I - |0^n\rangle \langle 0^n|) + sH_1$?*

Surprisingly, this problem can be easily avoided by picking an initial Hamiltonian that does not commute with $H_1$ (equivalently, $H_1$ is not diagonal in the same basis as $H_0$). A phenomenon called "avoided crossing" then predicts that no energy levels $\lambda_i(s)$ and $\lambda_j(s)$ will cross for $i \neq j$. This ensures a nonzero gap $\Delta > 0$ throughout the evolution. A typical choice is the Hamiltonian

$$H_0 = -\sum_j X_j, \tag{2}$$

where $X_j$ is shorthand for applying the Pauli $X$-gate to the $j$-th qubit while leaving the other qubits invariant. The unique ground state of $H_0$ is $|\phi_0\rangle = |+\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{z \in \{0,1\}^n} |z\rangle$.

The resulting quantum adiabatic optimization algorithm will succeed if run for a time $T \geq \text{poly}(1/\Delta)$, with $\Delta$ the gap of the intermediate Hamiltonian $H(s)$. The algorithm will hence be efficient precisely when $H(s)$ has a large gap (at least inversely polynomial with $n$). Unfortunately, as is the case with classical variants of this algorithm (e.g., simulated annealing), this is generally hard to verify. This makes the quantum adiabatic optimization algorithm mostly a heuristic.

**Exercise 2** (Max cut)**.** *Finding the maximum cut in a graph is a canonical problem in combinatorial optimization. For a graph $G$ with vertex set $[n]$ and (symmetric) edge set $E \subseteq [n]^2$, a maximum cut is described by a subset $Z \subset [n]$ that cuts a maximum number of edges (edges crossing from $Z$ to $Z^c$). Equivalently, it maximizes the cut function*

$$c(A) = \sum_{(i,j)\in E} I_{i\in A, j\notin A}.$$

- *Identify a subset $A$ with the indicator $a \in \{0,1\}^n$ ($i \in A \Leftrightarrow a_i = 1$). Express the cut function $c(A)$ as a degree-2 polynomial $h$ in $a$.*
- *Rewrite the cost Hamiltonian $H_1 = -\sum_{z\in\{0,1\}^n} c(z)\,|z\rangle\langle z|$ in terms of identity and Pauli-Z matrices.*

## References

[FGGS00] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv:quant-ph/0001106*, 2000.