

An Introduction to Adiabatic Quantum Computing

In addition to the circuit model discussed in the textbook, adiabatic quantum computing (ACQ) shows great promise as a computing platform. ACQ does not require a description in terms of gates and circuit components, nevertheless, the circuit model and ACQ do share common features and it has been argued that the latter is a special case of the former¹

The philosophy behind ACQ is similar to that of classical simulated annealing². Consider a complicated multivariate function $f(x_1 \dots x_n)$. The goal of simulated annealing is to find the point $x_1^* \dots x_n^*$ in which this function has its extremum (usually minimum) value. This is achieved by relying on statistical and thermodynamic arguments to simulate a physical system that evolves into the desired configuration.

Typically, the goal of ACQ is to find the ground state (energy eigenstate with the lowest energy) of a quantum system by simulating, on the computer, Hamiltonian evolution of the latter. Problems of this type are important in many applications. For example, finding the ground state of large molecules can require tremendous time and memory resources on standard digital computers. The latter problem has many applications, including drug design and in the understanding of protein folding to name just a few. A functional quantum computer could greatly advance our capabilities in these arenas.

The goal of ACQ is simple. Given a quantum system that is described by a Hamiltonian H , find its ground state energy eigenvalue and eigenstate. To achieve the desired goal; one takes advantage of the quantum adiabatic theorem, which we paraphrase (for the special case of the ground state) below

Quantum Adiabatic Theorem

Consider a quantum system that is described by a Hamiltonian $H(t)$ that evolves very slowly (i.e. adiabatically) in time t . If this system possesses a non-degenerate ground state $|\psi(t_0)\rangle$ at $t = t_0$, then $e^{i\gamma} |\psi(t)\rangle$, where $|\psi(t)\rangle$ is an instantaneous ground eigenstate of $H(t)$, is a solution to the Schrodinger equation. The phase $e^{i\gamma} = \langle \psi(t) | U(t, t_0) | \psi(t_0) \rangle / | \langle \psi(t) | U(t, t_0) | \psi(t_0) \rangle |$ and $U(t, t_0)$ is the time evolution operator.

From our previous discussions, we know that this theorem obviously applies to the case where H is independent of time. The question is, how slow must $H(t)$ evolve in order for the above statement to hold? We will not address that question here, instead we will experiment with a model system in order to illustrate the point.

The AQC Strategy

We exploit the quantum adiabatic theorem by mapping the time independent Hamiltonian H , whose ground state is sought, into a time dependent Hamiltonian $H(t)$

$$H(t) = H_0 (1 - t/T) + t/T H.$$

Here, H_0 is a Hamiltonian whose eigenstates and eigenvalues we can easily discover, and T is a parameter. At $t = 0$ we find that $H(0) = H_0$ and at $t = T$, $H(T) = H$. So according to the adiabatic theorem if we start at $t = 0$ in the ground state of H_0 we are guaranteed, up to a phase factor, to wind up in the ground state at $t = T$. As the utility of the adiabatic theorem is contingent on adiabatic evolution, we take the time derivative of $H(t)$

$$dH(t)/dt = -H_0/T + H/T$$

and which suggests that slow evolution implies large values for the parameter T .

Model System

```
In[76]:= (* define basic 1-qubit gates *)
```

```
In[77]:= ClearAll["Global`*"];
σX = {{0, 1}, {1, 0}};
σZ = {{1, 0}, {0, -1}};
σY = {{0, -I}, {I, 0}};
unit = {{1, 0}, {0, 1}};
```

We wish to find the ground state of the following Hamiltonian

$$H = H_1 + H_{int}$$

$$H_1 = \hbar\omega (\sigma_X \otimes \mathbb{1} \otimes \mathbb{1} - \mathbb{1} \otimes \sigma_X \otimes \mathbb{1} + \mathbb{1} \otimes \mathbb{1} \otimes \sigma_X)$$

$$H_{int} = \hbar\omega (\sigma_X \otimes \sigma_X \otimes \mathbb{1} + \sigma_X \otimes \mathbb{1} \otimes \sigma_X + \mathbb{1} \otimes \sigma_X \otimes \sigma_X)$$

```
In[82]:= H1 = ħω (KroneckerProduct[σX, unit, unit] -
  KroneckerProduct[unit, σX, unit] + KroneckerProduct[unit, unit, σX]);
Hint = ħΩ (KroneckerProduct[σX, σX, unit] + KroneckerProduct[σX, unit, σX] +
  KroneckerProduct[unit, σX, σX]);
```

```
In[84]:= H = H1 + Hint;
```

```
MatrixForm[H]
```

```
Out[85]//MatrixForm=
```

$$\begin{pmatrix} 0 & \hbar\omega & -\hbar\omega & \hbar\Omega & \hbar\omega & \hbar\Omega & \hbar\Omega & 0 \\ \hbar\omega & 0 & \hbar\Omega & -\hbar\omega & \hbar\Omega & \hbar\omega & 0 & \hbar\Omega \\ -\hbar\omega & \hbar\Omega & 0 & \hbar\omega & \hbar\Omega & 0 & \hbar\omega & \hbar\Omega \\ \hbar\Omega & -\hbar\omega & \hbar\omega & 0 & 0 & \hbar\Omega & \hbar\Omega & \hbar\omega \\ \hbar\omega & \hbar\Omega & \hbar\Omega & 0 & 0 & \hbar\omega & -\hbar\omega & \hbar\Omega \\ \hbar\Omega & \hbar\omega & 0 & \hbar\Omega & \hbar\omega & 0 & \hbar\Omega & -\hbar\omega \\ \hbar\Omega & 0 & \hbar\omega & \hbar\Omega & -\hbar\omega & \hbar\Omega & 0 & \hbar\omega \\ 0 & \hbar\Omega & \hbar\Omega & \hbar\omega & \hbar\Omega & -\hbar\omega & \hbar\omega & 0 \end{pmatrix}$$

We find the ground state of this Hamiltonian by introducing Hamiltonian H_0

$$H_0 = \hbar\omega \left(\sigma_Z \otimes \mathbb{1} \otimes \mathbb{1} + \mathbb{1} \otimes \sigma_Z \otimes \mathbb{1} + \mathbb{1} \otimes \mathbb{1} \otimes \sigma_Z \right)$$

whose eigenstates are easy to calculate. For example in this case, the eigenstates of H_0 are simply the 3-qubit computational basis vectors $|0\rangle_3, |1\rangle_3, \dots, |7\rangle_3$. The ground state of H_0 is $|7\rangle_3$ with eigenvalue $-3\hbar\omega$.

```
In[86]:= H0 = \hbar\omega (KroneckerProduct[\sigmaZ, unit, unit] +
```

```
KroneckerProduct[unit, \sigmaZ, unit] + KroneckerProduct[unit, unit, \sigmaZ]);
```

```
MatrixForm[
```

```
H0]
```

```
Out[87]//MatrixForm=
```

$$\begin{pmatrix} 3\hbar\omega & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \hbar\omega & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \hbar\omega & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\hbar\omega & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \hbar\omega & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\hbar\omega & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\hbar\omega & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3\hbar\omega \end{pmatrix}$$

```
In[88]:= (* now we construct the adiabatic hamiltonian *)
```

```
Had[t_, T_] = (1 - t/T) H0 + t/T H;
```

```
% // MatrixForm
```

```
Out[89]//MatrixForm=
```

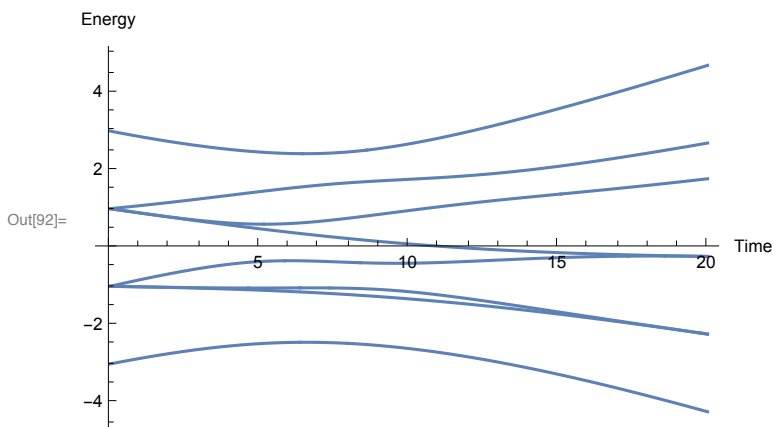
$$\begin{pmatrix} 3\left(1 - \frac{t}{T}\right)\hbar\omega & \frac{t\hbar\omega}{T} & -\frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\Omega}{T} \\ \frac{t\hbar\omega}{T} & \left(1 - \frac{t}{T}\right)\hbar\omega & \frac{t\hbar\Omega}{T} & -\frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\omega}{T} & 0 & \frac{t\hbar\Omega}{T} \\ -\frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & \left(1 - \frac{t}{T}\right)\hbar\omega & \frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & 0 & \frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} \\ \frac{t\hbar\Omega}{T} & -\frac{t\hbar\omega}{T} & \frac{t\hbar\omega}{T} & -\left(1 - \frac{t}{T}\right)\hbar\omega & 0 & \frac{t\hbar\Omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\omega}{T} \\ \frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\Omega}{T} & 0 & \left(1 - \frac{t}{T}\right)\hbar\omega & \frac{t\hbar\omega}{T} & -\frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} \\ \frac{t\hbar\Omega}{T} & \frac{t\hbar\omega}{T} & 0 & \frac{t\hbar\Omega}{T} & \frac{t\hbar\omega}{T} & -\left(1 - \frac{t}{T}\right)\hbar\omega & \frac{t\hbar\Omega}{T} & -\frac{t\hbar\omega}{T} \\ \frac{t\hbar\Omega}{T} & 0 & \frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\Omega}{T} & -\frac{t\hbar\omega}{T} & -\left(1 - \frac{t}{T}\right)\hbar\omega & \frac{t\hbar\Omega}{T} \\ 0 & \frac{t\hbar\Omega}{T} & \frac{t\hbar\Omega}{T} & \frac{t\hbar\omega}{T} & \frac{t\hbar\Omega}{T} & -\frac{t\hbar\omega}{T} & \frac{t\hbar\omega}{T} & -3\left(1 - \frac{t}{T}\right)\hbar\omega \end{pmatrix}$$

Exact Diagonalization

Before we embark on an adiabatic journey with this Hamiltonian, let's find, using standard methods, the instantaneous eigenvalues of $H(t)$ for all values of t . This includes the sought for

eigenvalue $H(t)$. Because we are working with a 3-qubit system, standard matrix diagonalization is computationally straightforward. However this become computationally intractable for Hamiltonians in a large Hilbert space (e.g. for a 100 qubits). As we illustrate below, quantum adiabatic evolution offers an alternative path toward this goal.

```
In[90]:= (* We set the parameters  $\hbar\omega$ ,  $\hbar\Omega$  to some arbitrary values,
and take the value for  $T=20$  *)
 $\hbar\omega = 1.0$ ;  $\hbar\Omega = 1.23$ ;
Tend = 20.0;
Plot[Sort[Eigenvalues[Had[t, Tend]]],
{t, 0, Tend}, PlotRange -> All, AxesLabel -> {Time, Energy}]
```



The above graph is a plot of all eigenvalues, shown on the vertical axis, of $H(t)$ for all values of $0 < t < T$ on the horizontal axis.

The bottom-most curve represent the ground state energy as a function of t .

Quantum Adiabatic Evolution

We now employ quantum adiabatic evolution to estimate the ground state wavefunction and energy eigenvalue at $t = T$. First, we express the system wavefunction as an expansion over the computational basis of the 3-qubit Hilbert space, i.e.

$$|\psi(t)\rangle = \sum_{i=0}^7 c_i(t) |i\rangle$$

or, as a matrix representation

```
In[93]:= psi[t_] = {c0[t], c1[t], c2[t], c3[t], c4[t], c5[t], c6[t], c7[t]};
```

Inserting this expression into the Schrodinger equation, we obtain a set of coupled equations for the coefficients $c_i(t)$.

```
In[94]:= equations = Had[t, T].psi[t] - I psi'[t];
```

```
In[95]:= set[T_] = Table[equations[[i]] == 0, {i, 1, 8}]
```

```
Out[95]= {3. (1 - \frac{t}{T}) c0[t] + \frac{1. t c1[t]}{T} - \frac{1. t c2[t]}{T} +
```

$$\frac{1.23 t c3[t]}{T} + \frac{1. t c4[t]}{T} + \frac{1.23 t c5[t]}{T} + \frac{1.23 t c6[t]}{T} - i c0'[t] = 0,$$

```
\frac{1. t c0[t]}{T} + 1. (1 - \frac{t}{T}) c1[t] + \frac{1.23 t c2[t]}{T} - \frac{1. t c3[t]}{T} +
```

$$\frac{1.23 t c4[t]}{T} + \frac{1. t c5[t]}{T} + \frac{1.23 t c7[t]}{T} - i c1'[t] = 0,$$

```
- \frac{1. t c0[t]}{T} + \frac{1.23 t c1[t]}{T} + 1. (1 - \frac{t}{T}) c2[t] + \frac{1. t c3[t]}{T} +
```

$$\frac{1.23 t c4[t]}{T} + \frac{1. t c6[t]}{T} + \frac{1.23 t c7[t]}{T} - i c2'[t] = 0,$$

```
\frac{1.23 t c0[t]}{T} - \frac{1. t c1[t]}{T} + \frac{1. t c2[t]}{T} - 1. (1 - \frac{t}{T}) c3[t] +
```

$$\frac{1.23 t c5[t]}{T} + \frac{1.23 t c6[t]}{T} + \frac{1. t c7[t]}{T} - i c3'[t] = 0,$$

```
\frac{1. t c0[t]}{T} + \frac{1.23 t c1[t]}{T} + \frac{1.23 t c2[t]}{T} + 1. (1 - \frac{t}{T}) c4[t] +
```

$$\frac{1. t c5[t]}{T} - \frac{1. t c6[t]}{T} + \frac{1.23 t c7[t]}{T} - i c4'[t] = 0,$$

```
\frac{1.23 t c0[t]}{T} + \frac{1. t c1[t]}{T} + \frac{1.23 t c3[t]}{T} + \frac{1. t c4[t]}{T} -
```

$$1. (1 - \frac{t}{T}) c5[t] + \frac{1.23 t c6[t]}{T} - \frac{1. t c7[t]}{T} - i c5'[t] = 0,$$

```
\frac{1.23 t c0[t]}{T} + \frac{1. t c2[t]}{T} + \frac{1.23 t c3[t]}{T} - \frac{1. t c4[t]}{T} + \frac{1.23 t c5[t]}{T} -
```

$$1. (1 - \frac{t}{T}) c6[t] + \frac{1. t c7[t]}{T} - i c6'[t] = 0,$$

```
\frac{1.23 t c1[t]}{T} + \frac{1.23 t c2[t]}{T} + \frac{1. t c3[t]}{T} + \frac{1.23 t c4[t]}{T} -
```

$$\frac{1. t c5[t]}{T} + \frac{1. t c6[t]}{T} - 3. (1 - \frac{t}{T}) c7[t] - i c7'[t] = 0\}$$

These coupled first order equations are now solved, starting from the initial ground state at $t = 0$, for the ground state amplitude at $t = T$. Since at $t = 0$, the ground state amplitude is the basis ket $|7\rangle_3$ or $|111\rangle$,

```
In[96]:= groundinitial = {0, 0, 0, 0, 0, 0, 0, 1};
H0.groundinitial = -3 groundinitial
```



```
Out[97]= True
```

```
In[98]:= total = Flatten[Append[set[Tend], {c0[0] == 0, c1[0] == 0,
c2[0] == 0, c3[0] == 0, c4[0] == 0, c5[0] == 0, c6[0] == 0, c7[0] == 1}]];
```

which we now solve using the NDSOLVE command.

```
In[99]:= Tend = 20.0;
```

```
sols = Flatten[NDSolve[total, {c0, c1, c2, c3, c4, c5, c6, c7}, {t, 0, Tend}]]
```

```
Out[100]:= {c0 → InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar ],
```

$c1 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$,

$c2 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$,

$c3 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$,

$c4 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$,

$c5 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$,

$c6 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$,

$c7 \rightarrow \text{InterpolatingFunction}[\text{Domain: } \{0., 20.\}, \text{Output: scalar}]$ }

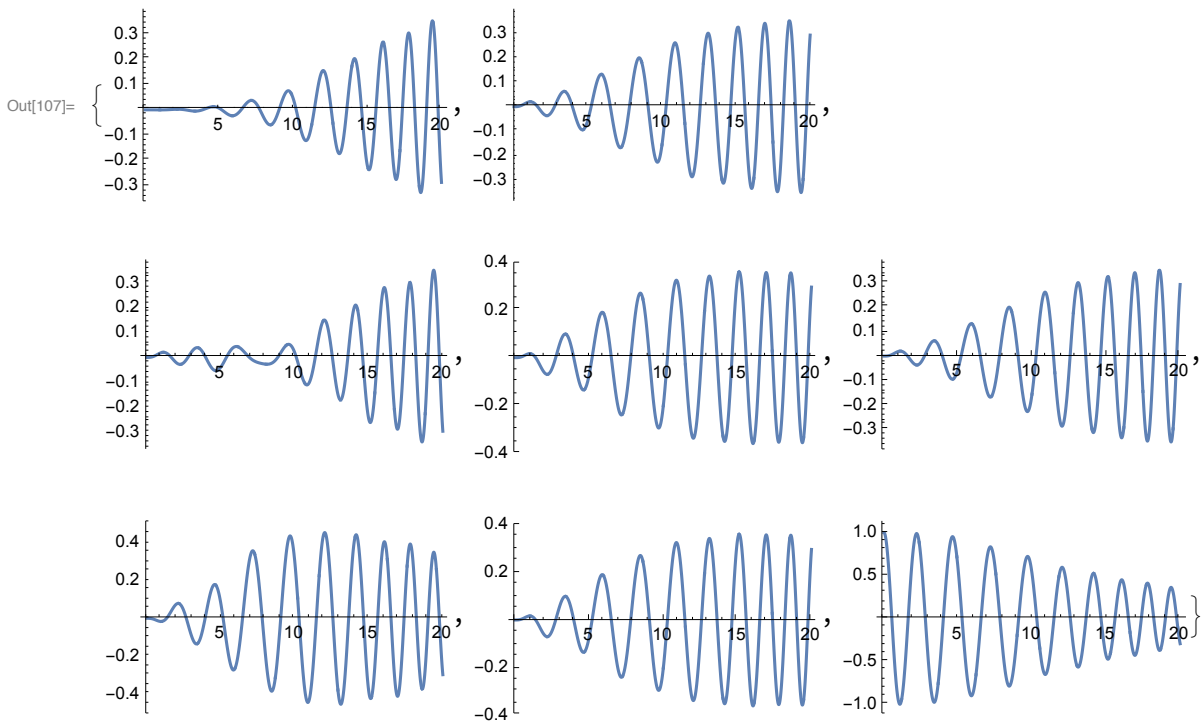
```
In[106]:= wave[t_] = psi[t] /. sols
```

```
Out[106]= {InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t],  
InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t],  
InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t],  
InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t],  
InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t],  
InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t],  
InterpolatingFunction[ Domain: {{0., 20.}}  
Output: scalar][t]}
```

Solution to Adiabatic Evolution.

The adiabatic wavefunction $|\psi(t)\rangle$ is contained in the wavefunction `wave[t]`

```
In[107]:= Table[Plot[Re[wave[t][[i]]], {t, 0, Tend}], {i, 1, 8}]
```



This diagram plots the real part of the amplitude, as a function of time, for each basis component of the amplitude. Note that at $t = 0$ all amplitudes, save for the graph with $i = 7$ which corresponds to the initial ground state wavefunction, vanishes. At $t = T$ all components are occupied. We can estimate the energy eigenvalue of the ground state by evaluating

$$E_{\text{ground}} \approx \langle \psi(T) | H | \psi(T) \rangle$$

```
In[108]:= Conjugate[wave[Tend]] . H.wave[Tend]
```

```
Out[108]:= -4.22956 + 2.77556 × 10-17 i
```

Let's compare this estimate with the exact result, using Mathematica's `Eigenvalues` routine

```
In[109]:= (*?Eigenvalues*)
```

```
Eigenvalues[H]
```

```
Out[109]:= {4.69, -4.23, 2.69, -2.23, -2.23, 1.77, -0.23, -0.23}
```

Notice that the lowest (i.e. ground state), exact eigenvalue has the value $E = -4.23$, within 0.005 the value obtained using adiabatic evolution.

Quantum Supremacy

In the 3-qubit example given above, we demonstrated how quantum adiabatic evolution can be used to estimate the ground energy eigenvalue.

In order to arrive at our estimate we needed to solve the time dependent Schrodinger equation for the system. This is not very efficient, indeed imagine solving a set of (2^{100}) coupled equations

in a 100 qubit system. So what advantages are conferred by quantum adiabatic evolution? If we simulate the latter with a classical computer, as done here, there are no advantages. But for a real quantum system containing, let's say a 100 qubits, all we need to do is encode the Hamiltonian H , load the initial ground state $|\psi(0)\rangle$ and simply allow the system to evolve according to Postulate V discussed in the text. In this way, the quantum computer effectively, and with a modest 100 qubit overhead, targets the solution that we harvest by interrogating the qubits at $t = T$.

¹ "Adiabatic Quantum Computation is equivalent to Standard Quantum Computation", D. Aharonov et. al, arXiv:quant-ph/0405098

² https://en.wikipedia.org/wiki/Simulated_annealing