MIT Open Access Articles

Quantum Adiabatic Algorithms, Small Gaps, and Different Paths

Massachusetts Institute of Technology

Quantum Adiabatic Algorithms, Small Gaps, and Different Paths

Edward Farhi,[1] Jeffrey Goldstone,[1] David Gosset,[1]

Sam Gutmann,[2] Harvey B. Meyer,[1,3] and Peter Shor[1,4]

[1]*Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, MA 02139*

[2]*Department of Mathematics, Northeastern University, Boston, MA 02115*

[3]*Physics Department, CERN, 1211 Geneva 23, Switzerland*

[4]*Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139*

Abstract

We construct a set of instances of 3SAT which are not solved efficiently using the simplest quantum adiabatic algorithm. These instances are obtained by picking random clauses all consistent with two disparate planted solutions and then penalizing one of them with a single additional clause. We argue that by randomly modifying the beginning Hamiltonian, one obtains (with substantial probability) an adiabatic path that removes this difficulty. This suggests that the quantum adiabatic algorithm should in general be run on each instance with many different random paths leading to the problem Hamiltonian. We do not know whether this trick will help for a random instance of 3SAT (as opposed to an instance from the particular set we consider), especially if the instance has an exponential number of disparate assignments that violate few clauses. We use a continuous imaginary time Quantum Monte Carlo algorithm in a novel way to numerically investigate the ground state as well as the first excited state of our system. Our arguments are supplemented by Quantum Monte Carlo data from simulations with up to 150 spins.

I. INTRODUCTION

Quantum adiabatic algorithms are designed for classical combinatorial optimization problems [8]. In the simplest case, such algorithms work by adiabatically evolving in the ground state of a system with Hamiltonian $H(s) = (1-s)H_B + sH_P$ that is a function of a parameter s which is increased from 0 to 1 as a function of time. H_B is called the beginning Hamiltonian and H_P, which is instance dependent, is called the problem Hamiltonian. The minimum (for $s \in [0,1]$) eigenvalue gap between the ground state and first excited state of $H(s)$ is related to the runtime of the adiabatic algorithm. If $H(s)$ has an exponentially small minimum gap then the corresponding algorithm is inefficient, whereas a minimum gap which scales inverse polynomially corresponds to an efficient quantum adiabatic algorithm.

Whether or not quantum adiabatic algorithms can be used to solve classically difficult optimization problems efficiently remains to be seen. Some numerical studies have examined the quantum adiabatic algorithm on random sets of instances of optimization problems where these sets are thought to be difficult for classical algorithms. These studies have reported polynomial scaling of the minimum gap out to about 100 bits [6, 12, 20]. Whether this scaling persists at high bit number has recently been called into question [21]. Meanwhile there has been no rigorous analytical result that characterizes the performance of the quantum adiabatic algorithms on random instances of NP-complete problems.

Over the years there have been a number of proposed examples which were meant to demonstrate failures of the adiabatic algorithm on specific problems. In reference [18], van Dam et al constructed examples intended to foil the adiabatic algorithm, but these examples used a nonlocal cost function. Related 3SAT examples of van Dam and Vazirani [19] indeed cannot be solved efficiently using the quantum adiabatic algorithm. However it was shown in [5] that such 3SAT instances do not pose a problem for the quantum adiabatic algorithm if, having fixed a specific problem Hamiltonian, one randomly chooses multiple interpolating paths between the initial and final Hamiltonians and runs the adiabatic algorithm once for each random path. Fisher [9] has constructed an interesting but specialized example on which the quantum adiabatic algorithm is inefficient. (A later example of Reichardt [17] is based on this.) We do not know if random path change helps here. (It is interesting to note that in this case the runtime scales like $c^{\sqrt{n}}$ where n is the number of bits.) The authors of reference [22] pointed out that a certain adiabatic algorithm for 3SAT is not efficient. However, this was due to a perverse and avoidable nonlocal choice of beginning Hamiltonian H_B [7].

One purpose of this paper is to discuss a different type of challenge to adiabatic optimization
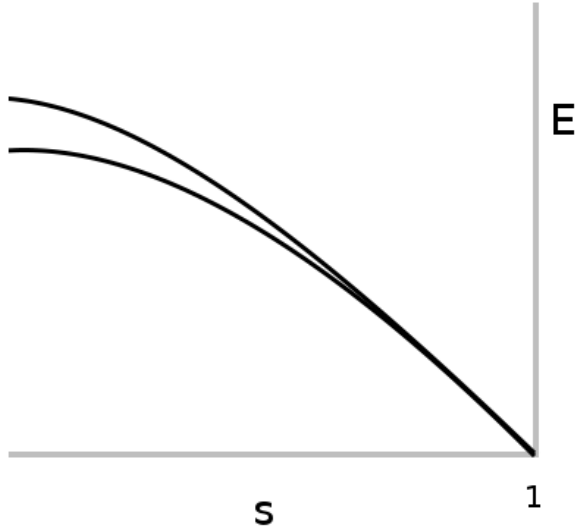
Figure 1: Energy levels of the Hamiltonian $H'(s)$ before adding the last term h to the problem Hamiltonian. The lower curve coincides with the upper curve only at $s = 1$.

which has recently come to light. (See references [2–4] and in a different context [15].) It seems to us that in the history of challenges to adiabatic optimization, this may be the most serious. The takeaway message of this work is that a small minimum gap for $H(s)$ can arise when the Hamiltonian H_P has features which are seen in the following construction. First suppose that we start with a problem Hamiltonian H'_P which has two degenerate ground states $|z_1\rangle$ and $|z_2\rangle$ corresponding to bit strings of length n that differ in order n bits. We then expect that the two lowest eigenvalues of $H'(s) = (1 - s)H_B + sH'_P$ will look something like figure 1. Note that one curve is always below the other except at $s = 1$ even though in the figure they appear to meld because they have the same slope at $s = 1$. Suppose that the upper curve (the first excited state for s near 1) approaches the state $|z_2\rangle$ and the lower curve approaches the state $|z_1\rangle$ as $s \to 1$. We now form the problem Hamiltonian $H_P = H'_P + h$, where h is a term that penalizes the state $|z_1\rangle$ but not the state $|z_2\rangle$. Then, for the two lowest eigenvalues of $H(s) = (1 - s)H_B + sH_P$, we expect to have the situation pictured in figure 2, where there is a small gap near $s = 1$.

In this paper we discuss a simple way of generating random instances of 3SAT where the corresponding adiabatic Hamiltonian $H(s)$ has the difficulty discussed above. We argue in section III that this problem can be overcome (for the set of instances we consider) by randomizing the choice of beginning Hamiltonian.

We present a continuous imaginary time Quantum Monte Carlo algorithm, which is a modifica-
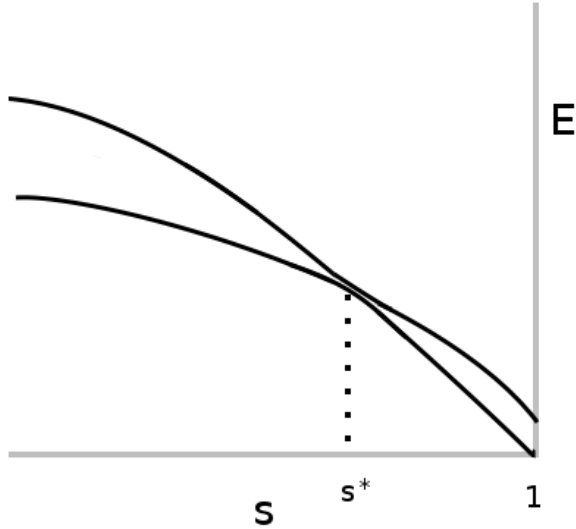
Figure 2: Energy levels of the Hamiltonian $H(s)$. There is a tiny gap at $s^\star$.

tion of the Heat Bath algorithm of Krzakala et al [14]. ("Quantum Monte Carlo" is a completely classical numerical technique for finding properties of quantum systems and is not a quantum algorithm.) We use this numerical technique in a novel manner which allows us to investigate both the ground state and the first excited state of our Hamiltonian and thereby detect the presence or absence of a small gap between them. This differs from the standard application of the Quantum Monte Carlo method in that we are able to obtain information about the first excited state using a simple procedure which we have validated at low bit number by comparing our results to exact numerical diagonalization. These Quantum Monte Carlo simulations support our claim that the problem we describe can be overcome using random path change.

II. PROBLEMATIC INSTANCES

We now describe the method we use to generate n bit random instances of 3SAT that lead to quantum adiabatic Hamiltonians with small minimum gaps. To do this we first generate an instance with exactly two satisfying assignments given by the bit strings 111...1 and 000...0 . Each clause c of the 3SAT instance specifies a subset of 3 bits $i_1(c), i_2(c), i_3(c) \in \{1, .., n\}$ and a particular assignment $w_1(c)w_2(c)w_3(c)$ to those three bits which is disallowed. In order to only generate instances which are consistent with the bit strings 111...1 and 000...0, we only use clauses for which

$$w_1(c)w_2(c)w_3(c) \in \{100, 010, 001, 110, 101, 011\}.$$

4

We add such clauses one at a time uniformly at random and stop as soon as these two bit strings are the only bit strings which satisfy all of the clauses that have been added. (In practice to check whether or not this is the case we use a classical 3SAT solver.) We write m for the total number of clauses in the instance. We note that the number of clauses m obtained using this procedure scales like $n \log n$. We need this many clauses in order to ensure that each bit is involved in some clause. On the other hand, when the number of clauses is $5n \log n$ the probability of additional satisfying assignments goes to zero as $n \to \infty$.

We now consider the problem Hamiltonian H'_P corresponding to this instance, which we define to be

$$H'_P = \sum_{c=1}^{m} \left(\frac{1 + (-1)^{w_1(c)} \sigma_z^{i_1(c)}}{2} \right) \left(\frac{1 + (-1)^{w_2(c)} \sigma_z^{i_2(c)}}{2} \right) \left(\frac{1 + (-1)^{w_3(c)} \sigma_z^{i_3(c)}}{2} \right).$$

Each term in this sum is 1 if $i_1(c) i_2(c) i_3(c)$ violates the clause and 0 otherwise. For the beginning Hamiltonian we choose

$$H_B = \sum_{i=1}^{n} \left(\frac{1 - \sigma_x^i}{2} \right). \tag{1}$$

The two lowest eigenvalues of the Hamiltonian $H'(s) = (1-s)H_B + sH'_P$ will then both approach 0 as $s \to 1$, as in figure 1. The ground state for values of s which are sufficiently close to 1 will approach either $|000...0\rangle$ or $|111...1\rangle$ as $s \to 1$. Suppose for values of s close to 1 the state that approaches $|000...0\rangle$ has lowest energy. Then we add an extra term h_0 which acts on bits $1, 2$ and 3 and which penalizes this state but not $|111...1\rangle$

$$h_0 = \frac{1}{2} \left(\frac{1 + \sigma_z^1}{2} \right) \left(\frac{1 + \sigma_z^2}{2} \right) \left(\frac{1 + \sigma_z^3}{2} \right)$$

and pick

$$H_P = H'_P + h_0.$$

(Note that this extra term has a multiplicative factor of $\frac{1}{2}$, to avoid any degeneracy of the first excited state at $s = 1$.) In the case where the lowest energy state near $s = 1$ approaches $|111...1\rangle$ we instead add

$$h_1 = \frac{1}{2} \left(\frac{1 - \sigma_z^1}{2} \right) \left(\frac{1 - \sigma_z^2}{2} \right) \left(\frac{1 - \sigma_z^3}{2} \right).$$

The Hamiltonian $H(s) = (1-s)H_B + sH_P$ is then expected to have a small gap near $s = 1$ as depicted in figure 2. We expect the location $s^\star$ of the minimum gap to approach $s = 1$ as $n \to \infty$.

Location of the Minimum Gap

In order to determine the dependence of the location $s^\star$ of the avoided crossing on the number of spins n and the number of clauses m, we consider the perturbative corrections to the energies of the states $|000...0\rangle$ and $|111...1\rangle$ around $s = 1$. We will show that the low order terms in the perturbation series reliably predict a crossing at[1] $s^\star = 1 - \Theta(\frac{1}{n^{1/4}} \left(\frac{m}{n}\right)^{\frac{3}{4}})$.

With the Hamiltonian constructed in the previous section, one of these states has energy 0 at $s = 1$ (call this the lower state $|z_L\rangle$) and the other state has energy $\frac{1}{2}$ at $s = 1$ (we call this the upper state $|z_U\rangle$). We write our Hamiltonian as

$$H(s) = (1-s)\frac{n}{2} + s\left[-\left(\frac{1-s}{s}\right)\sum_{i=1}^{n}\frac{\sigma_x^i}{2} + H_P\right] \tag{2}$$

and we then consider the term $-\left(\frac{1-s}{s}\right)\sum_{i=1}^{n}\frac{\sigma_x^i}{2}$ as a perturbation to H_P expanding around $s = 1$.

To understand why we trust perturbation theory to predict the location of the near crossing, consider a system which is composed of two disconnected sectors, A and B, so the corresponding Hamiltonian is of the form

$$\begin{pmatrix} H_A(s) & 0 \\ 0 & H_B(s) \end{pmatrix}.$$

In this situation the generic rule that levels do not cross does not apply and we can easily imagine that the two lowest levels look like what we show in figure 3, where the levels actually cross at $s^\star$.

Imagine that low order perturbation theory around $s = 1$ can be used to get good approximations to the ground state energies of $H_A(s)$ and $H_B(s)$ for s near $s^\star$, even for s somewhat to the left of $s^\star$. Then it is possible to accurately predict $s^\star$. Our situation is very close to this. We can think of A as consisting of the states close to z_L in Hamming weight, and B as states close to z_U in Hamming weight. Similarly, we view H_A and H_B as the restrictions of H to these sectors. Note that it takes n powers of the perturbation to connect $|000...0\rangle$ and $|111...1\rangle$ and this is why we view A and B as essentially disconnected.

Although figure 3 looks like figure 2, it is figure 2 that depicts the actual situation, where the two levels avoid crossing. This true near cross means that the perturbation series in the actual theory will diverge very close to $s^\star$. However this divergence will only be seen at high order, in fact at an order which is proportional to n. The low order terms of the perturbation series in figure 3

---

[1] The notation $f(n) = \Theta(g(n))$ means that, for n sufficiently large, $bg(n) \le f(n) \le cg(n)$ for some constants b and c.
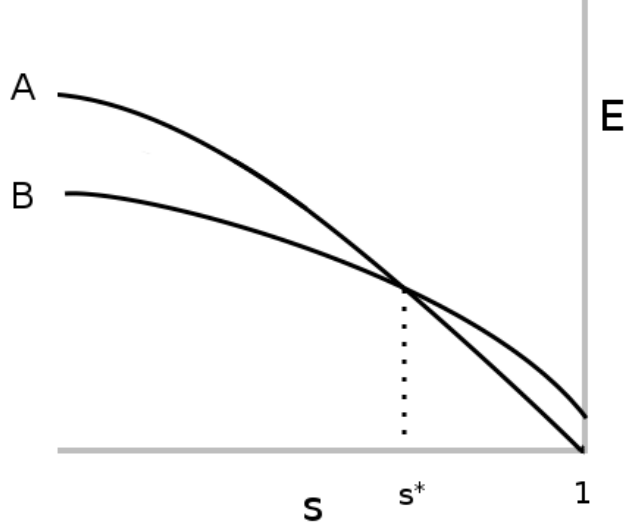
Figure 3: A true energy level crossing can arise from two disconnected sectors.

are the same as the low order terms of the perturbation series in figure 2, so we can trust low order perturbation theory to locate $s^\star$.

(We argue below that as a function of the number of bits, n, $s^\star$ goes to 1 as n goes to infinity. This implies that the radius of convergence of the perturbation theory for the full $H(s)$, expanded about $s = 1$, goes to 0 as n goes to infinity. This fact has no bearing on our argument that low order perturbation theory can be used to accurately predict $s^\star$.)

For small values of the parameter $\frac{1-s}{s}$, the energies of the two states under consideration can be expanded as

$$E_L(s) = (1-s)\frac{n}{2} + s\left[0 + \left(\frac{1-s}{s}\right)^2 e_L^{(2)} + \left(\frac{1-s}{s}\right)^4 e_L^{(4)} + \ldots\right] \tag{3}$$

$$E_U(s) = (1-s)\frac{n}{2} + s\left[\frac{1}{2} + \left(\frac{1-s}{s}\right)^2 e_U^{(2)} + \left(\frac{1-s}{s}\right)^4 e_U^{(4)} + \ldots\right]. \tag{4}$$

It is easy to see that each expansion (inside the square brackets) only contains even powers. Note that $e_L^{(2)}$ is guaranteed to be negative as is always the case for 2nd order perturbation theory of the ground state. In addition, since we added a term to penalize the state which had smaller energy near $s = 1$ before adding the clause, we expect that $e_U^{(2)} < e_L^{(2)}$. We are interested in the behaviour of the difference $E_L(s) - E_U(s)$ for randomly generated instances (as generated using the prescription of the previous section) as a function of the number of spins n in the limit $n \to \infty$. This requires us to further investigate the behaviour of each coefficient $e_L^{(k)}$ and $e_U^{(k)}$ as a function of n and m.

7

In fact to locate $s^\star$ we need only to go to second order in perturbation theory. From equation 2 we view H_P as the unperturbed Hamiltonian, and $-\frac{1}{2}\sum_{i=1}^{n}\sigma_x^i$ as the perturbation, with $\frac{1-s}{s}$ as the expansion parameter. Now $\sigma_x^i|z\rangle = |z \oplus \hat{e}_i\rangle$ so at second order we get

$$
\begin{aligned}
e_L^{(2)} &= \frac{1}{4}\sum_{i=1}^{n}\frac{|\langle z_L \oplus \hat{e}_i|\sigma_x^i|z_L\rangle|^2}{\langle z_L|H_P|z_L\rangle - \langle z_L \oplus \hat{e}_i|H_P|z_L \oplus \hat{e}_i\rangle} \\
&= -\frac{1}{4}\sum_{i=1}^{n}\frac{1}{\langle z_L \oplus \hat{e}_i|H_P|z_L \oplus \hat{e}_i\rangle}\;.
\end{aligned}
$$

Similarly,

$$
e_U^{(2)} = \frac{1}{4}\sum_{i=1}^{n}\frac{1}{\frac{1}{2} - \langle z_U \oplus \hat{e}_i|H_P|z_U \oplus \hat{e}_i\rangle}
$$

where the $\frac{1}{2}$ in the denominator is $\langle z_U|H_P|z_U\rangle$.

The expected energy penalty incurred when flipping a bit of either z_U or z_L is of order $\frac{m}{n}$ since each bit is typically involved in $\Theta(\frac{m}{n})$ clauses. So the coefficients $e_L^{(2)}$ and $e_U^{(2)}$ are of order $n\left(\frac{n}{m}\right)$ since the energy denominators involved are $\Theta(\frac{m}{n})$. We now show that their difference is of order $\sqrt{n}\left(\frac{n}{m}\right)^{\frac{3}{2}}$. Write

$$
e_U^{(2)} - e_L^{(2)} = \sum_{i=1}^{n}d_i
$$

where for each i,

$$
d_i = \frac{1}{4}\left(\frac{1}{\frac{1}{2} - \langle z_U \oplus \hat{e}_i|H_P|z_U \oplus \hat{e}_i\rangle} + \frac{1}{\langle z_L \oplus \hat{e}_i|H_P|z_L \oplus \hat{e}_i\rangle}\right). \tag{5}
$$

Recall that $H_P = H_P' + h$ where h is the penalty term from the final clause which acts only on the first 3 bits. Therefore, for $i = 4, 5, ..., n$

$$
d_i = \frac{1}{4}\left(-\frac{1}{\langle z_U \oplus \hat{e}_i|H_P'|z_U \oplus \hat{e}_i\rangle} + \frac{1}{\langle z_L \oplus \hat{e}_i|H_P'|z_L \oplus \hat{e}_i\rangle}\right).
$$

Our procedure for generating instances is symmetric between the strings 000...0 and 111...1 so averaging over instances it is clear that the mean of d_i for $i = 4, 5..., n$ is 0. Thus we expect $\sum_{i=4}^{n}d_i$ to be (approximately) Gaussian with mean 0 and standard deviation proportional to $\sqrt{n}\sigma(d)$, where $\sigma(d)$ is the standard deviation of each d_i for $i \in \{4, 5, ..., n\}$. To compute $\sigma(d)$ we note that

$$
d_i = \frac{1}{4}\left(\frac{\langle z_U \oplus \hat{e}_i|H_P'|z_U \oplus \hat{e}_i\rangle - \langle z_L \oplus \hat{e}_i|H_P'|z_L \oplus \hat{e}_i\rangle}{\langle z_L \oplus \hat{e}_i|H_P'|z_L \oplus \hat{e}_i\rangle\langle z_U \oplus \hat{e}_i|H_P'|z_U \oplus \hat{e}_i\rangle}\right).
$$

Again using the symmetry between all zeros and all ones, we conclude that the numerator is of order $\sqrt{\frac{m}{n}}$ and the denominator is of order $\left(\frac{m}{n}\right)^2$. Hence we expect $\sigma(d)$ to be $\Theta\left(\left(\frac{n}{m}\right)^{\frac{3}{2}}\right)$. So $e_U^{(2)} - e_L^{(2)}$

is of order $\sqrt{n}\left(\frac{n}{m}\right)^{\frac{3}{2}}$. We will now locate $s^\star$ using second order perturbation theory and afterwards argue that higher orders do not change the result. Returning to equations 3 and 4, equating the two energies at second order we have

$$0 = s^\star \left[\frac{1}{2} + \left(\frac{1 - s^\star}{s^\star} \right)^2 \left(e_U^{(2)} - e_L^{(2)} \right) \right]$$

so

$$s^\star = 1 - \Theta\left(\frac{1}{n^{1/4}} \left(\frac{m}{n} \right)^{\frac{3}{4}} \right). \tag{6}$$

The 4th order correction to the energy of the lower state is given by

$$e_L^{(4)} = \langle z_L | V \left(\frac{\phi_L}{H_P} \right)^2 V | z_L \rangle \langle z_L | V \frac{\phi_L}{H_P} V | z_L \rangle - \langle z_L | V \frac{\phi_L}{H_P} V \frac{\phi_L}{H_P} V \frac{\phi_L}{H_P} V | z_L \rangle$$

where

$$V = -\frac{1}{2} \sum_{i=1}^n \sigma_x^i$$

and $\phi_L = 1 - |z_L\rangle\langle z_L|$. Writing $\mathcal{H}_P(z) = \langle z | H_P | z \rangle$, this can be expressed as

$$\begin{aligned}
e_L^{(4)} &= \frac{1}{16} \sum_{i=1}^n \sum_{j=1}^n \frac{1}{(\mathcal{H}_P(z_L \oplus \hat{e}_i))^2 \, \mathcal{H}_P(z_L \oplus \hat{e}_j)} \\
&\quad - \frac{1}{16} \sum_{i \neq j} \frac{1}{\mathcal{H}_P(z_L \oplus \hat{e}_i) \mathcal{H}_P(z_L \oplus \hat{e}_j) \mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j)} \\
&\quad - \frac{1}{16} \sum_{i \neq j} \frac{1}{(\mathcal{H}_P(z_L \oplus \hat{e}_i))^2 \, \mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j)} \\
&= \sum_{i=1}^n \frac{1}{16} \frac{1}{(\mathcal{H}_P(z_L \oplus \hat{e}_i))^3} + \sum_{i \neq j} \frac{1}{16} \frac{\mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j) - \mathcal{H}_P(z_L \oplus \hat{e}_i) - \mathcal{H}_P(z_L \oplus \hat{e}_j)}{(\mathcal{H}_P(z_L \oplus \hat{e}_i))^2 \, \mathcal{H}_P(z_L \oplus \hat{e}_j) \, (\mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j))}.
\end{aligned}$$

Now consider the terms in this expression corresponding to indices i, j for which $i \neq j$ and the bits i and j do not appear in a clause together. Under these conditions we have

$$\mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j) = \mathcal{H}_P(z_L \oplus \hat{e}_i) + \mathcal{H}_P(z_L \oplus \hat{e}_j).$$

So we can write

$$\begin{aligned}
e_L^{(4)} &= \sum_{i=1}^n \frac{1}{16} \frac{1}{(\mathcal{H}_P(z_L \oplus \hat{e}_i))^3} \\
&\quad + \sum_{i \neq j \text{ clausemates}} \frac{1}{16} \frac{\mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j) - \mathcal{H}_P(z_L \oplus \hat{e}_i) - \mathcal{H}_P(z_L \oplus \hat{e}_j)}{(\mathcal{H}_P(z_L \oplus \hat{e}_i))^2 \, \mathcal{H}_P(z_L \oplus \hat{e}_j) \, (\mathcal{H}_P(z_L \oplus \hat{e}_i \oplus \hat{e}_j))}.
\end{aligned} \tag{7}$$

Here the subscript "clausemates" indicates that we sum only over pairs of indices which appear together in at least one clause of the 3SAT instance corresponding to H_P. For $e_U^{(4)}$, since the unperturbed energy is $\frac{1}{2}$ we obtain

$$
e_U^{(4)} = \sum_{i=1}^{n} \frac{1}{16} \frac{1}{\left(\mathcal{H}_P(z_U \oplus \hat{e}_i) - \frac{1}{2}\right)^3} \tag{8}
$$
$$
+ \sum_{i \neq j \text{ clausemates}} \frac{1}{16} \frac{\left(\mathcal{H}_P(z_U \oplus \hat{e}_i \oplus \hat{e}_j) - \frac{1}{2}\right) - \left(\mathcal{H}_P(z_U \oplus \hat{e}_i) - \frac{1}{2}\right) - \left(\mathcal{H}_P(z_U \oplus \hat{e}_j) - \frac{1}{2}\right)}{\left(\mathcal{H}_P(z_U \oplus \hat{e}_i) - \frac{1}{2}\right)^2 \left(\mathcal{H}_P(z_U \oplus \hat{e}_j) - \frac{1}{2}\right) \left(\mathcal{H}_P(z_U \oplus \hat{e}_i \oplus \hat{e}_j) - \frac{1}{2}\right)} .
$$

Let's look at the first sum in each of equations 7 and 8 where i goes from 1 to n. Each term as i goes from 4 to n is of order $\left(\frac{n}{m}\right)^3$ and so the difference of the two sums is $\Theta(\sqrt{n}\left(\frac{n}{m}\right)^3)$. The second sums (those which are restricted to clausemates) contain of order m terms. Each denominator is of order $\left(\frac{m}{n}\right)^4$ and the numerators are $\Theta(1)$ since the only contribution to the numerator is from clauses in H_P which involve bits i and j together. Separating out the terms where i and/or j are $1, 2$ or 3, we conclude that the contribution to $e_U^{(4)} - e_L^{(4)}$ from the clausemate terms is $\Theta(\sqrt{n}\left(\frac{n}{m}\right)^{\frac{7}{2}})$. For our instance generation m grows like $n \log n$ so this clausemate contribution is asymptotically dominated by the first term which scales like $\Theta(\sqrt{n}\left(\frac{n}{m}\right)^3)$. So the fourth order contribution to the difference of energies $E_U(s) - E_L(s)$ is $\Theta(s\left[\sqrt{n}\left(\frac{n}{m}\right)^3\left(\frac{1-s}{s}\right)^4\right])$. At $s^\star$ which we determined at second order to be $1 - \Theta(\frac{1}{n^{1/4}}\left(\frac{m}{n}\right)^{\frac{3}{4}})$, the fourth order contribution to the energy difference is $\Theta\left(\frac{1}{\sqrt{n}}\right)$. The fourth order corrections can therefore be neglected in determining the location of $s^\star$. Sixth order and higher contributions to the difference are even smaller.

III. FIXING THE PROBLEM BY PATH CHANGE

In our instance generation we manufactured a small gap by penalizing the planted assignment corresponding to the energy eigenstate with the smallest energy near $s = 1$. Since the slopes of the two curves in figure 1 are the same at $s = 1$, the second derivatives determine which eigenvalue is smaller near $s = 1$. After penalization we have $e_U^{(2)} < e_L^{(2)}$ which is consistent with the near crossing in figure 2. Suppose instead that we penalize the assignment corresponding to larger energy in figure 1. Then we expect the situation depicted in figure 4 where no level crossing is induced.

We imagine that the instances that we manufacture with a small gap as in figure 2 are a model for what might be encountered in running the quantum adiabatic algorithm on some instance a quantum computer is actually trying to solve. There is a strategy for overcoming this problem. The idea is to produce figure 4, with reasonable probability, by randomly modifying the adiabatic path which ends at H_P, of course making no use of the properties of the particular instance. For this
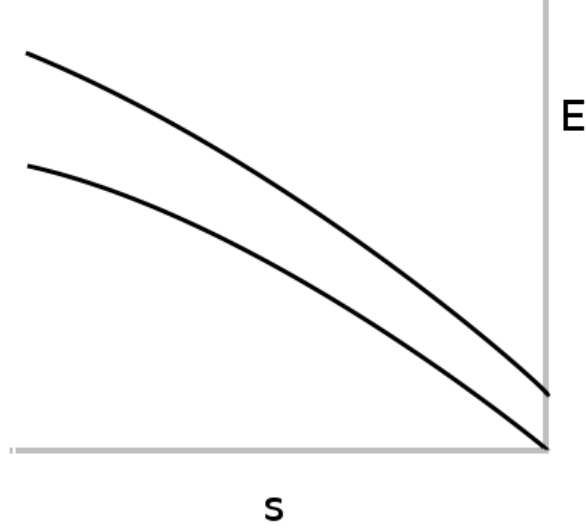
Figure 4: Energy levels with no avoided crossing near $s = 1$. Here the second derivative of the upper curve B is greater than the second derivative of the lower curve A.

purpose one could use any random ensemble of paths $H(s)$ such that $H(1) = H_P$ and the ground state of $H(0)$ is simple to prepare. However in this paper we only consider randomly changing the beginning Hamiltonian. We have made this choice so that we are able to use our Quantum Monte Carlo method to numerically verify our arguments. Like other Quantum Monte Carlo methods, the method we use does not work when the Hamiltonian has nonzero off-diagonal matrix elements with positive sign.

Starting with an instance where $H(s) = (1 - s)H_B + sH_P$ has a tiny gap due to the problem discussed above, we now consider a different adiabatic path $\tilde{H}(s) = (1 - s)\tilde{H}_B + sH_P$ obtained by keeping the same problem Hamiltonian H_P but choosing a different beginning Hamiltonian $\tilde{H}_B$ in a random fashion which we will prescribe below. We argue that the small gap near $s = 1$ is then removed with substantial probability, so that by repeating this procedure a constant number of times it is possible to find an adiabatic path without a small gap near $s = 1$.

The way that we choose a random Hamiltonian $\tilde{H}_B$ is to first draw n random variables c_i for $i = 1, 2, ..., n$, where each c_i is chosen to be $\frac{1}{2}$ or $\frac{3}{2}$ with equal probability. We then take

$$\tilde{H}_B = \sum_{i=1}^{n} \frac{c_i(1 - \sigma_x^i)}{2} .$$

(9)

We write $\tilde{e}_U^{(2)}$ and $\tilde{e}_L^{(2)}$ for the analogous quantities to $e_L^{(2)}$ and $e_U^{(2)}$ for the new Hamiltonian $\tilde{H}(s) = (1 - s)\tilde{H}_B + sH_P$. The point is that by randomizing $\tilde{H}_B$ in the way we prescribe, there is a

substantial probability that one will obtain $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > 0$, and in that case one expects no avoided crossing near $s = 1$. Write

$$\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} = \sum_{i=1}^{n} c_i^2 d_i$$

where the $\{d_i\}$ are fixed by the instance (and are defined in equation 5). Since we have fixed the problem Hamiltonian H_P, the only random variables appearing in the above equation are the c_i. We have $\overline{c_i^2} = \frac{5}{4}$ so the mean value of $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ is then

$$\overline{\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}} = \frac{5}{4}(e_U^{(2)} - e_L^{(2)}) < 0 \,.$$

But more importantly

$$\overline{\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}} = \Theta(\sqrt{n} \left(\frac{n}{m}\right)^{\frac{3}{2}}) \,.$$

The variance of this difference is

$$\mathrm{Var}\left(\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}\right) = \sum_{i=1}^{n} d_i^2 \mathrm{Var}(c_i^2)$$

$$= \sum_{i=1}^{n} d_i^2 \cdot 1 \,,$$

which is $\Theta(n \left(\frac{n}{m}\right)^3)$. For a fixed instance with a corresponding fixed set $\{d_i\}$ the random variable $\sum_i c_i^2 d_i$ is approximately Gaussian and from its mean and variance we see that the probability that $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ is positive and in fact greater than $a\sqrt{n} \left(\frac{n}{m}\right)^{\frac{3}{2}}$, for $a > 0$, is bounded away from 0 independent of n. This means that there is a good chance that randomizing H_B turns the situation depicted in figure 2 into the situation depicted in figure 4.

In the case of two planted satisfying assignments with one penalized to produce a small gap when the beginning Hamiltonian is H_B of equation 1, we have shown that a random choice for the beginning Hamiltonian $\tilde{H}_B$ of equation 9 can with substantial probability remove the small gap. This gives further weight to the idea that when running the quantum adiabatic algorithm on a single instance of some optimization problem, the programmer should run the quantum adiabatic algorithm repeatedly with different paths ending at H_P [5].

What if there are $k > 2$ satisfying assignments and all but one are penalized?

We have considered instances of 3SAT for which the corresponding problem Hamiltonian has a unique ground state and a nondegenerate first excited state that is far from the ground state in

12

Hamming weight. In this case we have shown that the path change strategy succeeds in removing a tiny gap. What happens when we have k mutually disparate solutions and we penalize all but one of them? We show (under some assumptions) that for large n, path change will succeed after a number of tries polynomial in k. We can therefore hope for success if there are k disparate assignments which violate few clauses and k scales polynomially with n. On the other hand when k is superpolynomial in n we have no reason to be optimistic about the performance of the quantum adiabatic algorithm.

The energies $E_r(s)$ for $r = 0, 1, .., k-1$ of the k states under consideration can be expanded as in equations 3 and 4. Write $e_r^{(2)}$ for the second order corrections in these expansions. If we add a small number of clauses to this instance which penalize all the solutions except for a randomly chosen one which we call z_0 (forming a new problem Hamiltonian $\hat{H}_p$) then the differences $e_r^{(2)} - e_0^{(2)}$ will not change substantially. So we are interested in the differences

$$e_r^{(2)} - e_0^{(2)} = \sum_{i=1}^{n} d_{ri},$$

where

$$d_{ri} = \frac{1}{4}\left(-\frac{1}{\langle z_r \oplus \hat{e}_i | H_p | z_r \oplus \hat{e}_i \rangle} + \frac{1}{\langle z_0 \oplus \hat{e}_i | H_P | z_0 \oplus \hat{e}_i \rangle}\right), \tag{10}$$

and where $z_0, z_1, z_2, ..., z_{k-1}$ are the k ground states of H_p. The adiabatic algorithm applied to this instance with problem Hamiltonian $\hat{H}_p$ will succeed if all of the above differences are positive, for $r = 1, ..., k-1$. There is a $\frac{1}{k}$ chance of this occurring. If they are not all positive, then we have encountered an instance which we expect to have a small gap. In this case, we randomize the beginning Hamiltonian as in the previous subsection. This produces new values $\tilde{e}_r^{(2)}$ for the second order energy corrections and the relevant differences are given by

$$\tilde{e}_r^{(2)} - \tilde{e}_0^{(2)} = \sum_{i=1}^{n} c_i^2 d_{ri}, \tag{11}$$

where the random variables c_i are $\frac{1}{2}$ or $\frac{3}{2}$ with equal probability (and the d_{ri} are fixed). With this fixed problem Hamiltonian, how many times do we have to randomize the beginning Hamiltonian (by drawing a random set $\{c_i\}$) to produce algorithmic success? For large n, the random variables $\tilde{e}_r^{(2)} - \tilde{e}_0^{(2)}$ are approximately jointly Gaussian with expectation and variance given by

$$\overline{\tilde{e}_r^{(2)} - \tilde{e}_0^{(2)}} = \frac{5}{4}\sum_{i=1}^{n} d_{ri}$$

$$\mathrm{Var}\left(\tilde{e}_r^{(2)} - \tilde{e}_0^{(2)}\right) = \sum_{i=1}^{n} (d_{ri})^2.$$

13

The correlation (the covariance divided by the product of the standard deviations) between any two of these random variables is given by

$$\text{Corr}\left(\tilde{e}_q^{(2)} - \tilde{e}_0^{(2)}, \tilde{e}_r^{(2)} - \tilde{e}_0^{(2)}\right) = \frac{\sum_i d_{qi} d_{ri}}{\left[\sum_j (d_{qj})^2 \sum_j (d_{rj})^2\right]^{\frac{1}{2}}}. \tag{12}$$

We now argue that the above correlation is typically equal to $\frac{1}{2}$ when n is large. To see this, we can expand the above expression using equation 10. We imagine that the instances have been generated so that the law of large numbers applies and that in the limit of large n the quantity

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{\langle z_r \oplus \hat{e}_i | H_p | z_r \oplus \hat{e}_i\rangle} \cdot \frac{1}{\langle z_r \oplus \hat{e}_i | H_p | z_r \oplus \hat{e}_i\rangle}\right)$$

approaches an r independent value and the quantity

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{\langle z_q \oplus \hat{e}_i | H_p | z_q \oplus \hat{e}_i\rangle} \cdot \frac{1}{\langle z_r \oplus \hat{e}_i | H_p | z_r \oplus \hat{e}_i\rangle}\right)$$

approaches a value independent of r and q for $r \neq q$. Using this in equation 12, we obtain that the correlation is equal to $\frac{1}{2}$ (for $r \neq q$ and in the limit of large n).

Recall that the problem Hamiltonian has been fixed and therefore the $\{d_{ri}\}$ are set. We have argued that for large n the differences $\left\{\tilde{e}_r^{(2)} - \tilde{e}_0^{(2)}\right\}$ have approximately a joint normal distribution with pairwise correlations equal to $\frac{1}{2}$. The probability that a random choice of the $\{c_i\}$ make the algorithm succeed is

$$\begin{aligned}
\Pr\left[\tilde{e}_r^{(2)} - \tilde{e}_0^{(2)} > 0 \ , \ r = 1, ..., k-1\right] &= \Pr\left[\sum_{i=1}^{n} c_i^2 d_{ri} > 0 \ , \ r = 1, ..., k-1\right] \\
&= \Pr\left[\frac{\sum_{i=1}^{n} c_i^2 d_{ri} - \frac{5}{4}\sum_{i=1}^{n} d_{ri}}{\sqrt{\sum_{i=1}^{n}(d_{ri})^2}} > \frac{-\frac{5}{4}\sum_{i=1}^{n} d_{ri}}{\sqrt{\sum_{i=1}^{n}(d_{ri})^2}}, r = 1, ..., k-1\right].
\end{aligned}$$

Note that in the last line we have subtracted the mean and divided by the standard deviation. Using a standard trick we can rewrite the above probability as (in the limit of large n),

$$\Pr\left[w_r - w_0 > \frac{-\frac{5}{4}\sum_{i=1}^{n} d_{ri}}{\sqrt{\sum_{i=1}^{n}(d_{ri})^2}}, r = 1, ..., k-1\right], \tag{13}$$

where w_0 and $\{w_r\}$ are independent normal random variables with mean zero and variance $\frac{1}{2}$. This follows from the fact that the joint normal distribution is specified by its marginal distributions and pairwise correlations. Essentially w_r and w_0 are the (normalized) changes in the quantities $e_r^{(2)}$ and $e_0^{(2)}$ provided by the randomization.

At this point, in order to evaluate equation 13 we need to understand the magnitude of the RHS of the inequality. Consider typical values for the sum $\sum_{i=1}^{n} d_{ri}$. As in section II we assume that

14

this sum is Gaussian in the limit of large n with mean 0 and standard deviation approximated by $\sqrt{\sum_{i=1}^{n} d_{ri}^2}$.

(In the argument below, we use some properties of the Gaussian distribution which follow from the inequalities

$$\int_a^\infty e^{-x^2} dx \leq \int_a^\infty \frac{x}{a} e^{-x^2} dx = \frac{1}{2a} e^{-a^2}$$

$$\int_a^\infty e^{-x^2} dx \geq \int_a^{a+1} e^{-x^2} dx \geq e^{-(a+1)^2}.)$$

For the $k-1$ approximately Gaussian random variables $\frac{-\sum_{i=1}^{n} d_{ri}}{\sqrt{\sum_{i=1}^{n} (d_{ri})^2}}$ with mean 0 and variance 1, for some constant B it will hold with high probability (in the limit of large n) that

$$\frac{-\frac{5}{4}\sum_{i=1}^{n} d_{ri}}{\sqrt{\sum_{i=1}^{n} (d_{ri})^2}} < B\sqrt{\log k} \text{ for } r = 1, ..., k-1.$$

Now the random variable w_0 satisfies $w_0 < -2B\sqrt{\log k}$ with probability which is only polynomially small in k. For each w_r the probability that $w_r < -B\sqrt{\log k}$ is polynomially small in k and we use the bound

$$\Pr\left[w_r - w_0 \geq B\sqrt{\log k}, \, r = 1, ..., k-1\right] \geq \Pr\left[w_0 < -2B\sqrt{\log k}\right] \cdot \prod_{r=1}^{k-1} \left(1 - \Pr\left[w_r < -B\sqrt{\log k}\right]\right).$$

The first term on the RHS is only polynomially small, and each of the $k-1$ terms in the product are at least $1 - \frac{1}{k^2}$ if B is large enough. So the probability that the adiabatic algorithm succeeds is at worst polynomially small in k. By repeating the randomization a polynomial number of times we expect to succeed with high probability.

IV. QUANTUM MONTE CARLO AND NUMERICAL RESULTS

Continuous Imaginary Time Quantum Monte Carlo

This section is a review of continous imaginary time Quantum Monte Carlo (which is a classical path integral simulation technique for extracting properties of quantum systems [16]). In particular we will show how this method can be used to compute thermal expectation values of Hermitian operators at inverse temperature β. We start with a Hamiltonian H which we write as

$$H = H_0 + V$$

where H_0 is diagonal in some known basis $\{|z\rangle\}$, and V is purely off diagonal in this basis. We require that all the nonzero matrix elements of V are negative. For the Hamiltonian $H(s) = (1-s)H_B + sH_P$

with H_B as in equation 1, we have

$$H_0 = sH_P + \frac{(1-s)n}{2}$$

$$V = -(1-s)\sum_{i=1}^{n}\frac{\sigma_x^i}{2}.$$

Here we include the factor of $1-s$ in the definition of V since we are not doing perturbation theory in this quantity.

The partition function can be expanded using the Dyson series as follows

$$Tr\left[e^{-\beta H}\right] = Tr\left[e^{-\beta H_0}\sum_{m=0}^{\infty}(-1)^m\int_{t_m=0}^{\beta}dt_m\int_{t_{m-1}=0}^{t_m}dt_{m-1}...\int_{t1_1=0}^{t_2}dt_1 V_I(t_m)...V_I(t_1)\right].$$

Here we use the notation $V_I(t) = e^{tH_0}Ve^{-tH_0}$. We can then insert complete sets of states and take the trace to obtain the path integral

$$Tr\left[e^{-\beta H}\right] =$$

$$\sum_{m=0}^{\infty}\sum_{\{z_1,...,z_m\}}\left[(-1)^m\langle z_1|V|z_m\rangle\langle z_m|V|z_{m-1}\rangle...\langle z_2|V|z_1\rangle\right.$$

$$\left.\int_{t_m=0}^{\beta}dt_m\int_{t_{m-1}=0}^{t_m}dt_{m-1}...\int_{t_1=0}^{t_2}dt_1 e^{-\int_{t=0}^{\beta}\mathcal{H}_0(z(t))dt}\right]. \tag{14}$$

In this formula we have used the notation $\mathcal{H}_0(z(t)) = \langle z(t)|H_0|z(t)\rangle$, where the function $z(t)$ is defined by

$$z(t) = \begin{cases} z_1, & 0 \leq t < t_1 \\ z_2, & t_1 \leq t < t_2 \\ \vdots \\ z_m, & t_{m-1} \leq t < t_m \\ z_1, & t_m \leq t \leq \beta. \end{cases}$$

So in particular

$$\int_{t=0}^{\beta}\mathcal{H}_0(z(t))dt = \langle z_1|H_0|z_1\rangle(t_1 + \beta - t_m) + \langle z_2|H_0|z_2\rangle(t_2 - t_1) + ...\langle z_m|H_0|z_m\rangle(t_m - t_{m-1}).$$

We view the function $z(t)$ as a path in imaginary time which begins at $t = 0$ and ends at $t = \beta$. Then equation 14 is a sum over paths, where every path is assigned a positive weight according to a measure $\tilde{\rho}$

$$\tilde{\rho}(P) = (-1)^m\langle z_1|V|z_m\rangle\langle z_m|V|z_{m-1}\rangle...\langle z_2|V|z_1\rangle dt_1...dt_m e^{-\int_{t=0}^{\beta}\mathcal{H}_0(z(t))dt}.$$

Note that the fact that $\tilde{\rho}$ is positive semidefinite follows from our assumption that all matrix elements of V are negative. We write

$$\rho = \frac{\tilde{\rho}}{Z(\beta)} \tag{15}$$

for the normalized distribution over paths.

We now discuss how one can obtain properties of the quantum system as expectation values with respect to the classical measure ρ. We are interested in ground state properties, but in practice we select a large value of β and compute thermal expectation values at this inverse temperature. If β is sufficiently large then these expectation values will agree with the corresponding expectation values in the ground state. It is straightforward to show that for any operator A which is diagonal in the $|z\rangle$ basis, the thermal expectation value can be written as

$$\frac{Tr[Ae^{-\beta H}]}{Tr[e^{-\beta H}]} = \langle \frac{1}{\beta} \int_0^\beta \mathcal{A}(z(t))dt \rangle_\rho \tag{16}$$

where $\mathcal{A}(z(t)) = \langle z(t)|A|z(t)\rangle$. The notation $\langle \rangle_\rho$ means average with respect to the classical probability distribution ρ over paths.

This immediately allows us to estimate quantities such as the diagonal part of the Hamiltonian

$$\frac{Tr[H_0 e^{-\beta H}]}{Tr[e^{-\beta H}]} = \langle \frac{1}{\beta} \int_0^\beta \mathcal{H}_0(z(t))dt \rangle_\rho . \tag{17}$$

Another quantity that we find useful in our study is the Hamming weight operator defined by

$$W = \sum_{i=1}^n \left(\frac{1 - \sigma_z^i}{2} \right) . \tag{18}$$

which can also be estimated using equation 16. In order to estimate the thermal expectation value of the full Hamiltonian $H_0 + V$ we use equation 17 as well as the expression

$$\frac{Tr[Ve^{-\beta H}]}{Tr[e^{-\beta H}]} = -\langle \frac{m}{\beta} \rangle_\rho \tag{19}$$

where m is the number of transitions in the path. (Equation 19 is not as simple to derive as equation 16.) So by generating paths from the distribution ρ and then computing averages with respect to ρ, we can evaluate the thermal expectation value of the energy at a given inverse temperature β and by taking β sufficiently large we can approximate the ground state energy.

Generating paths from the distribution ρ is itself a challenging task. We use a modified version of the heat bath algorithm of Krzakala et al [14] which we describe in the appendix. As with other Quantum Monte Carlo methods, this algorithm is a Markov Chain Monte Carlo method. In order to sample from the distribution ρ, one defines a Markov Chain over the state space consisting of

all paths, where the limiting distribution of the chain is ρ. To obtain samples from ρ one starts in an initial path $z_{init}(t)$ and then applies some number N_{equil} of iterations of the Markov Chain. If N_{equil} is sufficiently large then the distribution of the paths found by further iteration will be close to ρ. We use these subsequent paths to compute averages with respect to ρ.

Equilibration of the Quantum Monte Carlo and Identification of Level Crossings

The discussion in the previous section demonstrates how one can estimate expectation values of various quantities in the ground state of a quantum system. The method is "continuous time" so there is no discretization error and for fixed β quantities can be arbitrarily well approximated with enough statistics. We use the Quantum Monte Carlo method in a nonstandard way in order to be able to study the lowest two eigenstates of our Hamiltonian (as opposed to just the ground state).

As described in the previous section, the standard procedure for generating configurations is to equilibrate to the distribution ρ from some initial path $z_{init}(t)$ (we call this the seeded path) by applying the Monte Carlo update N_{equil} times. In order to ensure that one has equilibrated after N_{equil} Monte Carlo updates, one could for example do simulations with two or more different initial paths (seeded paths) and check that the values of Monte Carlo observables appear to converge to the same seed independent values.

For each instance we consider, we run two different Monte Carlo simulations that are seeded with two different paths $z^0_{init}(t)$ and $z^1_{init}(t)$. These seeds are paths with no flips in them, corresponding to the two states 000...0 and 111...1,

$$z^0_{init}(t) = 000...0 \text{ for all } t \in [0, \beta]$$
$$z^1_{init}(t) = 111...1 \text{ for all } t \in [0, \beta].$$

With each seed, we run the Monte Carlo simulation for some total number N_{total} of Monte Carlo sweeps, taking data every k sweeps. Here a single sweep is defined to be n iterations of our Monte Carlo update rule as described in the appendix (where n is the number of spins). We then remove the first N_{equil} Monte Carlo samples, and use the remaining Monte Carlo samples to estimate the thermal averages

$$\langle H \rangle = \frac{Tr[He^{-\beta H}]}{Tr[e^{-\beta H}]}$$
$$\langle W \rangle = \frac{Tr[We^{-\beta H}]}{Tr[e^{-\beta H}]}$$

using equations 17,19 and 18.

In order to convince the reader that the Monte Carlo algorithm works correctly, and that we can obtain information about the first excited state, we show data at 16 bits where exact numerical diagonalization is possible. In figures 5 and 6 we show the result of this procedure for a 3SAT instance with 16 bits, with problem Hamiltonian H'_P (before adding the penalty term h, so that the levels are degenerate at $s = 1$). This instance has 122 clauses. The inverse temperature is $\beta = 150$. The total number of Monte Carlo sweeps at each value of s is $N_{total} = 200000$, with data taken after every fifth sweep (this gave us 40000 data samples). We use $N_{equil} = 2500$ samples solely for equilibration at each value of s. Note that the Monte Carlo simulations with the two different seeds (corresponding to the circles and crosses in the figures) only agree for values of s less than roughly 0.4. We interpret this to mean that at these values of s the Quantum Monte Carlo has equilibrated to the proper limiting distribution ρ regardless of the seed. As s increases past 0.4, the two simulations abruptly begin to give different results. In this case the simulation seeded with $z^1_{init}(t)$ finds a metastable equilibrium of the Markov Chain (i.e not the limiting distribution ρ), which corresponds in this case to the first excited state of the Hamiltonian. For s above 0.4 we can see from the comparison with exact diagonalization that the two differently seeded values allow us to compute the energies (figure 5) and Hamming weights (figure 6) of the lowest two energy levels of our Hamiltonian.

What is happening here is that for large s, the quantum system can be thought of as consisting of two disconnected sectors. One sector consists of states in the z basis with low Hamming weight and the other with Hamming weight near n. The Quantum Monte Carlo "equilibrates" in each sector depending on the initial seed as can be seen by the smooth data in each sector. (Note that data is taken independently at each value of s.) The lower of the cross and circle at each value of s in figure 5 is the ground state energy. Of course if the classical algorithm ran for long enough then the circles in figure 5 would lie on the crosses for every value of s.

In figures 7 and 8 we show data taken for the same instance after the penalty clause is added that removes the degeneracy at $s = 1$. In this case the two levels have an avoided crossing as expected and which can be seen by exact numerical diagonalization in figure 7. However in the Monte Carlo simulation with two seeds there are two essentially disconnected sectors and the two levels being tracked do cross. When we see this behaviour in the Monte Carlo simulation, we interpret it as compelling evidence that there is a tiny gap in the actual system, which occurs where the curves cross. In figure 8 first look at the curves coming from the exact numerical diagonalization. The Hamming weight of the ground state decreases and then abruptly rises at the location of the minimum gap. The Hamming weight of the first excited state also undergoes an abrupt change

at the location of the minimum gap. Not surprisingly the exact diagonalization clearly shows the behaviour we expect with the manufactured tiny gap. Now look at the Monte Carlo data which is interpreted by first looking at figure 7. In figure 7 the true ground state is always the lower of the circles and the crosses. For s below $s^\star \approx 0.423$ it is the crosses and for s larger than $s^\star$ it is the circles. Accordingly in figure 8 the Hamming weight of the ground state is tracked by the crosses for s below $s^\star$ and by the circles for s above $s^\star$. We can therefore conclude, based only on the Monte Carlo data, that the Hamming weight of the ground state changes abruptly. At higher bit number we do not have exact numerical diagonalization available but we can still use the Quantum Monte Carlo to extract key information.

Randomizing the Beginning Hamiltonian

For the 16 spin Hamiltonian depicted above, we now consider randomizing the beginning Hamiltonian H_B as described in section III. In order to minimize the number of times we run the Monte Carlo algorithm (this will be more of an issue at high bit number where simulations are very time consuming), we generated many different sets of coefficients $\{c_i\}$ and calculated the differences $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ with fixed problem Hamiltonian H'_P and beginning Hamiltonians

$$\tilde{H}_B = \sum_{i=1}^{n} c_i \left(\frac{1 - \sigma_x^i}{2} \right) .$$

According to our discussion in section III, we expect the avoided crossing near $s = 1$ to be removed for choices of coefficients c_i such that the difference $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > 0$. We made a histogram of the $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ (shown in figure 9) and randomly chose three sets of coefficients $\{c_i\}$ such that $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > \frac{1}{2}$ for each. Our analysis predicts that in these cases the crossing will be removed. As expected, each of these three sets of coefficients resulted in an adiabatic Hamiltonian with the small gap removed, although in one there was another small gap at a smaller value of s. We plot the Monte Carlo data for one of these sets in figures 10 and 11.

In section II we argued that we could manufacture a small gap in a 3SAT instance as sketched in figure 2. Our 16 bit instance is a concrete example of this and the tiny gap can be seen in figure 7. In section III we argued that by randomizing the beginning Hamiltonian we should be able to produce figure 4. This is what we see concretely at 16 bits in figure 10. We now tell the same story at 150 bits using only Monte Carlo data since exact numerical diagonalization is not possible.

Data for an Instance of 3SAT with 150 bits

In addition to validating our method at 16 bits, we studied 3SAT instances with $25, 75$ and 150 bits using our Quantum Monte Carlo simulator. The data from these simulations for the most part supported our arguments. In this section we present Monte Carlo data taken for a double plant instance of 3SAT with 150 spins and 1783 clauses. In this case the inverse temperature $\beta = 300$, and the total number of Monte Carlo sweeps at each value of s is $N_{total} = 100000$, with data taken every fifth sweep (giving 20000 data samples). The first 2500 data samples at each value of s are removed for equilibration. We ran our simulations on a 648 processor SiCortex computer cluster in embarrassingly parallel fashion. We used a different processor for each value of s and each value of the seed. Data taken at the lower values of s took the longest to accumulate, in some cases more than 10 days on a single processor for a single data point.

Figures 12 and 13 show the energy and the Hamming weight before the penalty clause is added, running with two different seeds. In figure 12 for large values of s, the crosses are always below the circles and track the ground state which ends at 000...0 as can be seen in figure 13. We have also plotted the second order perturbation theory energies for these two levels, expanding around $s = 1$. Note the good agreement between the second order perturbation theory and the Monte Carlo data all the way down to $s = 0.35$.

Since the lower curve corresponds to 000...0, we penalize this assignment by the addition of a single clause to form H_P attempting to manufacture a near crossing. In figure 14, the circle data is below the cross data for s near 1 but the two curves cross near $s = .49$. This is seen clearly in figure 15 where the energy difference is plotted. From the Monte Carlo data we conclude that $H(s)$ has a tiny gap. The location of the avoided crossing is well predicted by second order perturbation theory as can be seen in figure 15.

The Monte Carlo data at 150 bits shows that we can make an instance of 3SAT with a tiny gap following the procedure outlined in section II. We now use the Monte Carlo simulator to show that a randomly chosen $\tilde{H}_B$ can alter the Hamiltonian $H(s)$ so that this small gap becomes large. For the instance at hand we first compute the difference $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ for 100000 randomly chosen sets of coefficients. The histogram of these differences is plotted in figure 17. After doing this we randomly selected (from this set) two sets of coefficients such that $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > \frac{1}{2}$. For both of these sets of coefficients we saw that the crossing at $s \approx 0.49$ was no longer present, although in one case there appeared to be a new crossing at a much lower value of s. Figures 18 and 19 show the Monte Carlo data for the choice of $\tilde{H}_B$ which does not appear to have any crossing. At 150 bits we see

compelling evidence that the story outlined in sections II and III is true.

Is $s^\star$ near 1?

We argued that $s^\star$ should go to 1 for large enough n. However our 16 bit example has $s^\star \approx 0.42$ and the 150 bit example has $s^\star \approx 0.49$. Recall from equation 6 that $s^\star = 1 - \Theta(\frac{1}{n^{1/4}} \left(\frac{m}{n}\right)^{\frac{3}{4}})$. At 16 bits with $m = 122$ we have $\frac{1}{n^{1/4}} \left(\frac{m}{n}\right)^{\frac{3}{4}} = 2.29$ and at 150 bits with $m = 1783$ we have $\frac{1}{n^{1/4}} \left(\frac{m}{n}\right)^{\frac{3}{4}} = 1.83$. Although asymptotically m is of order $n \log n$, for these values of n we are not yet in the regime where $\frac{1}{n^{1/4}} \left(\frac{m}{n}\right)^{\frac{3}{4}} \ll 1$.

Even though $s^\star$ is not near 1 for our instance at 150 bits, we see from figure 15 that second order perturbation theory can be used to predict the location of $s^\star$. This is because the fourth order contribution to the energy difference is quite small. So already at 150 bits we can predict the presence or absence of an avoided crossing using second order perturbation theory.

V. CONCLUSIONS

We have introduced a new Quantum Monte Carlo technique to analyze the performance of quantum adiabatic algorithms for instances of satisfiability. Using seeded configurations (in the Monte Carlo simulation) corresponding to disparate low lying states, our technique exposed the presence or absence of an exponentially small gap without ever actually computing the gap.

We used this method to numerically investigate a set of random instances of 3SAT which were designed to expose a weakness of the adiabatic algorithm. We confirmed that this weakness can be overcome for our set of instances by using path change. Our numerical work and the main part of our analysis pertains to instances of 3SAT with 2 planted satisfying assignments.

We have also considered the scenario where an instance has k satisfying assignments and then all but one are penalized with a small number of clauses. However, in the $k > 2$ case we have made certain assumptions which make our analysis possible and we do not know if they apply to randomly generated instances. In our scenario the adiabatic algorithm with path change will succeed in a number of tries which is polynomially large in k. Here k is fixed and n goes to infinity but we take the calculation in section III as an indication that the algorithm will succeed when k grows polynomially with n.

Throughout this paper we have assumed that our instances have a unique satisfying assignment. In this case we believe that the crucial distinction is whether there are polynomially many or

exponentially many low lying mutually disparate states above the unique satisfying assignment. If there are polynomially many, we have argued that the adiabatic algorithm with path change will succeed, but if there are exponentially many we have no reason to be optimistic about the performance of the algorithm.

When there are exponentially many satisfying assignments our analysis does not apply. This situation was considered in recent work by Knysh and Smelyanskiy [13].

Our results give further evidence that path change must be considered an integral part of the quantum adiabatic algorithm. For any given instance, the algorithm should be run with many different randomly selected paths which end at the problem Hamiltonian. As long as the algorithm succeeds on at least a polynomially small fraction of the trials, it can be used to solve decision problems.

VI.   ACKNOWLEDGEMENTS

[1] Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. Adiabatic quantum computation is equivalent to standard quantum computation. *SIAM JOURNAL OF COMPUTING*, 37:166, 2007. arXiv:quant-ph/0405098.

[2] Boris Altshuler, Hari Krovi, and Jeremie Roland. Adiabatic quantum optimization fails for random instances of NP-complete problems, 2009. arXiv:0908.2782.

[3] M. H. S. Amin and V. Choi. First order quantum phase transition in adiabatic quantum computation, 2009. arXiv:0904.1387.

[4] Yale Fan. Adiabatic quantum algorithms for boolean satisfiability, 2009. (RSI summer project at MIT).

[5] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. Quantum adiabatic evolution algorithms with different paths, 2002. arXiv:quant-ph/0208135.

[6] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. *Science*, 292:472–475, 2001. arXiv:quant-ph/0104129.

[7] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Daniel Nagaj. How to make the quantum adiabatic algorithm fail. *International Journal of Quantum Information*, 6:503, 2008. arXiv:quant-ph/0512159.

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

[9] Daniel Fisher. Private communication based on references [10] and [11].

[10] Daniel S. Fisher. Critical behavior of random transverse-field ising spin chains. *Phys. Rev. B*, 51(10):6411–6461, Mar 1995.

[11] Daniel S. Fisher and A. P. Young. Distributions of gaps and end-to-end correlations in random transverse-field ising spin chains. *Phys. Rev. B*, 58(14):9131–9141, Oct 1998.

[12] Tad Hogg. Adiabatic quantum computing for random satisfiability problems. *Physical Review A*, 67:022314, 2003. quant-ph/0206059.

[13] S. Knysh and V. Smelyanskiy. On the relevance of avoided crossings away from quantum critical point to the complexity of quantum adiabatic algorithm. May 2010. arXiv:1005.3011.

[14] Florent Krzakala, Alberto Rosso, Guilhem Semerjian, and Francesco Zamponi. On the path integral representation for quantum spin models and its application to the quantum cavity method and to monte carlo simulations. *Physical Review B*, 78:134428, 2008. arXiv:0807.2553.

[15] Yoshiki Matsuda, Hidetoshi Nishimori, and Helmut G. Katzgraber. Ground-state statistics from annealing algorithms: Quantum vs classical approaches. *New Journal of Physics*, 11:073021, 2009. arXiv:0808.0365.

[16] N. V. Prokof'ev, B. V. Svistunov, and I. S. Tupitsyn. Exact quantum monte carlo process for the statistics of discrete systems. *ZH.EKS.TEOR.FIZ.*, 64:853, 1996. arXiv:cond-mat/9612091.

[17] Ben W. Reichardt. The quantum adiabatic optimization algorithm and local minima. In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 502–510, New York, NY, USA, 2004. ACM.

[18] Wim van Dam, Michele Mosca, and Umesh Vazirani. How powerful is adiabatic quantum computation?, 2002. arXiv:quant-ph/0206003.

[19] Wim van Dam and Umesh Vazirani. Limits on quantum adiabatic optimization (unpublished manuscript).

[20] A. P. Young, S. Knysh, and V. N. Smelyanskiy. Size dependence of the minimum excitation gap in the quantum adiabatic algorithm. *Physical Review Letters*, 101:170503, 2008.

[21] A.P Young. Private communication, 2009.

[22] Marko Znidaric and Martin Horvat. Exponential complexity of an adiabatic algorithm for an NP-complete problem. *Physical Review A*, 73:022329, 2006. arXiv:quant-ph/0509162.

Appendix: A modified Version of the Heat Bath algorithm of Krzakala et al

The authors of [14] give a Quantum Monte Carlo algorithm for spin systems in a transverse field. The algorithm we use, which is described in this section, is a modified version of that algorithm. The modification which we have made is described in subsection 2; everything else in this section constitutes a review of reference [14]. Like other worldline Quantum Monte Carlo techniques, the algorithm samples the appropriate probability distribution ρ (see section IV) over paths in imaginary time via Markov Chain Monte Carlo. However this algorithm is only applicable to the case where the Hamiltonian is of the form $H = H_0 + V$, where H_0 is diagonal in the computational basis $|z\rangle$ and

$$V = -\sum_{i=1}^{n} c_i \sigma_x^i$$

for some set of coefficients $\{c_i\}$ which are all positive.

For a Hamiltonian of this form, the distribution ρ over paths (from equation 15) is given by

$$\rho(P) = \frac{1}{Z(\beta)} \left(\Pi_{r=1}^{m} c_{i_r} \right) dt_1 ... dt_m e^{-\int_{t=0}^{\beta} \mathcal{H}_0(z(t)) dt}$$

where in this case a path is specified by an n bit string z_1 (call this the starting state) at time $t = 0$ and a sequence of flips which occur in bits labeled $i_1, ..., i_m$ at times $t_1, ..., t_m$ (which are ordered), where each $i_r \in \{1, ..., n\}$ and $t_r \in [0, \beta]$ for $r \in \{1, ..., m\}$. Another way of specifying a path is to specify the path P_j of each spin $j \in \{1, ..., n\}$. So a path P of the n spin system can be written $P = (P_1, P_2, ..., P_n)$. For each $j \in \{1, .., n\}$, P_j specifies the jth bit of the starting state z_1 as well as the times at which bit flips occur in bit j. Note that we only need to consider paths which flip each bit an even number of times, since only these paths occur with nonzero probability. An example of a path for a system with 2 spins is given in figure 20.
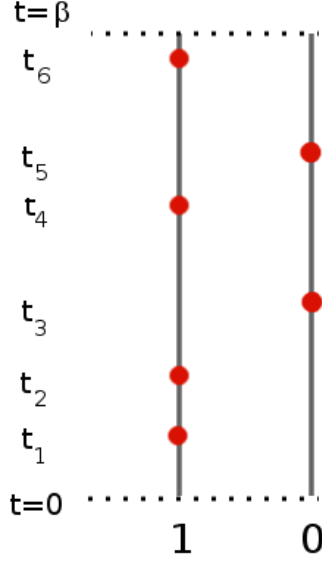
Figure 20: An example of a path P for 2 spins. The red dots indicate bit flips. In this example the starting state $z_1 = 10$ and there are four flips in the first spin and 2 in the second (i.e $i_1, i_2, i_4, i_6 = 1$ and $i_3, i_5 = 2$). The times of these flips are labeled t_1 through t_6.

We now define in detail the Markov Chain which has limiting distribution ρ. It will be useful to define $p_j(P_j|P_1, ..., P_{j-1}, P_{j+1}, ..., P_n)$ (for each $j \in \{1, ..., n\}$) to be the conditional probability distribution of the path of the jth bit, conditioned on the remainder of the path being fixed.

As in [14], the update rule for the Monte Carlo algorithm consists of the following 3 steps:

1. Randomly and uniformly choose a spin j (where $j \in \{1, ..., n\}$).

2. Remove all flips in the path which occur in bit j. Also remove the jth bit from the starting state z_1. This step corresponds to wholly removing the path P_j of spin j.

3. Draw a new path P_j (starting value and flip times) for spin j from the conditional distribution $p_j(P_j|P_1, ..., P_{j-1}, P_{j+1}, ..., P_n)$.

Of course, the nontrivial part of this algorithm is in specifying a procedure which executes step (3) in the above. To do this, the authors of reference [14] note that for any index $j \in \{1, ..., n\}$, the diagonal part of the Hamiltonian can be written as

$$H_0 = g_j + f_j \sigma_z^j \tag{20}$$

where g_j and f_j are operator valued functions of all $\{\sigma_z^k\}$ except for σ_z^j. For a given path P, we define $\mathcal{F}_j(z(t)) = \langle z(t)|f_j|z(t)\rangle$ for $t \in [0, \beta]$. The function $\mathcal{F}_j(z(t))$ is then piecewise constant, and

26

for a given path can be written as

$$
\mathcal{F}_j(z(t)) = \begin{cases}
\langle z_1|f_j|z_1\rangle, & 0 \leq t < t_1 \\
\langle z_2|f_j|z_2\rangle, & t_1 \leq t < t_2 \\
\vdots \\
\langle z_m|f_j|z_m\rangle, & t_{m-1} \leq t < t_m \\
\langle z_1|f_j|z_1\rangle, & t_m \leq t \leq \beta\,.
\end{cases}
$$

Although the above expression involves all the bit flips in the path, the function $\mathcal{F}_j(z(t))$ can actually only change value at times where bit flips occur in bits other than the jth bit, since the operator f_j does not involve σ_z^j. Let us then write

$$
\mathcal{F}_j(z(t)) = \begin{cases}
h_0, & 0 = \tilde{t}_0 \leq t \leq \tilde{t}_1 \\
h_1, & \tilde{t}_1 \leq t \leq \tilde{t}_2 \\
\vdots \\
h_q, & \tilde{t}_q \leq t \leq \beta = \tilde{t}_{q+1}\,.
\end{cases}
$$

In this expression the times $\tilde{t}_s$ correspond to times at which bit flips occur in bits other than bit j (also, $h_q = h_0$).

With this notation, the procedure of reference [14] that generates a new path for bit j consists of the following:

1. Compute the value of $\mathcal{F}_j(z(t))$ as a function of imaginary time along the path.

2. In this step we generate boundary conditions for the path of bit j at times $\tilde{t}_0, \tilde{t}_1, ...\tilde{t}_q$. In other words we choose $q+1$ values $s_0, ..., s_q \in \{0,1\}$ such that at time $\tilde{t}_r$ the bit j will be set to the value s_r in the new path that we are generating. To do this, we sample the values $s_0, ..., s_q \in \{0,1\}$ for spin j at the times $\tilde{t}_0, ..., \tilde{t}_q$ from their joint distribution, which is given by

$$
Z(s_0, s_1, ..., s_q | \{h_0, ..., h_q\}, \{\tilde{t}_1, ..., \tilde{t}_q\}) = \frac{\langle s_0|A_q|s_q\rangle\langle s_q|A_{q-1}|s_{q-1}\rangle...\langle s_1|A_0|s_0\rangle}{Tr[A_q A_{q-1}...A_0]}
$$

where

$$
A_i = e^{-\lambda_i\left[h_i\sigma_z^j - c_j\sigma_x^j\right]}
$$

and $\lambda_i = \tilde{t}_{i+1} - \tilde{t}_i$. We also define $s_{q+1} = s_0$.

3. Having chosen boundary conditions, we now generate subpaths for bit j on each interval $[\tilde{t}_i, \tilde{t}_{i+1}]$ of length λ_i. Such a subpath is specified by a number of flips w and the time offsets $\tau_1, \tau_2, ..., \tau_w \in [0, \lambda_i]$ at which flips occur (note that the starting value of the bit is determined by the boundary conditions). The number of flips is restricted to be either even or odd depending on the boundary conditions that were chosen for this interval in the previous step. In this step, the subpath for each interval $[\tilde{t}_i, \tilde{t}_{i+1}]$ is drawn from the distribution

$$g_i(\tau, ..., \tau_w) = \frac{1}{\langle s_{i+1}|A_i|s_i \rangle} c_j{}^w e^{-s_i h_i [(\tau_1 - 0) - (\tau_2 - \tau_1) + (\tau_3 - \tau_2) - ... + (\lambda_i - \tau_w)]} d\tau_1 ... d\tau_w \qquad (21)$$

4. Put all the subpaths together to form a new path P_j for bit j on $[0, \beta]$.

In section 1 we review the method outlined in [14] for sampling from the distribution $Z(s_0, s_1, ..., s_q | (h_0, ..., h_q), (\tilde{t}_1, ..., \tilde{t}_q))$ in step (2) of the above. In section 2 we outline our method for sampling from the distribution $g_i(\tau_1, ..., \tau_w)$, which differs from the method suggested in reference [14].

1. Generating Boundary Conditions for A Single Spin Path

The prescription outlined in [14] for generating a set $s_0, s_1, ..., s_q$ from the distribution

$$Z(s_0, s_1, ..., s_q | (h_0, ..., h_q), (\tilde{t}_1, ..., \tilde{t}_q)) = \frac{\langle s_0|A_q|s_q \rangle \langle s_q|A_{q-1}|s_{q-1} \rangle ... \langle s_1|A_0|s_0 \rangle}{Tr[A_q A_{q-1} ... A_0]}$$

is as follows. First generate $s_0 \in \{0, 1\}$ according to the distribution

$$p(s_0) = \frac{1}{Tr[A_q A_{q-1} ... A_0]} \langle s_0|A_q A_{q-1} ... A_0|s_0 \rangle .$$

(computing these probabilities involves multiplying q two by two matrices). Then generate $s_1 \in \{0, 1\}$ according to

$$p(s_1|s_0) = \frac{1}{\langle s_0|A_q A_{q-1} ... A_1 A_0|s_0 \rangle} \langle s_0|A_q A_{q-1} ... A_1|s_1 \rangle \langle s_1|A_0|s_0 \rangle .$$

Then generate $s_2 \in \{0, 1\}$ from the distribution

$$p(s_2|s_1, s_0) = \frac{1}{\langle s_0|A_q A_{q-1} ... A_1|s_1 \rangle} \langle s_0|A_q A_{q-1} ... A_2|s_2 \rangle \langle s_2|A_1|s_1 \rangle$$

and so on. Note that this generates the correct distribution since

$$p(s_0) p(s_1|s_0) p(s_2|s_1, s_0) ... p(s_q|s_{q-1}, ..., s_0) = Z(s_0, s_1, ..., s_q | (h_0, ..., h_q), (\tilde{t}_1, ..., \tilde{t}_q)) .$$

28

2. Algorithm for Sampling from the Single Spin Path Integral with Fixed Boundary Conditions

We now present an algorithm which samples from the normalized probability distribution over single spin paths $S(t)$ for $t \in [0, \lambda]$ ($S(t)$ takes values in $\{0,1\}$). Here we only present the case of paths with boundary conditions $S(0) = 0$ and $S(\lambda) = 1$. The other three cases are completely analogous. We can parameterize such a path with fixed boundary conditions by the times $\{\tau_1, ..., \tau_w\}$ at which $S(t)$ changes value. Note that the number w of such flips is odd due to our choice of boundary conditions. The distribution over paths that we aim to sample from is given by

$$\frac{c^w e^{-h[(\tau_1-0)-(\tau_2-\tau_1)+(\tau_3-\tau_2)-...+(\lambda-\tau_w)]} d\tau_1...d\tau_w}{\langle 1|e^{-\lambda[h\sigma_z-c\sigma_x]}|0\rangle}. \tag{22}$$

It will also be useful for us to make the change of variables from the times $(\tau_1, ..., \tau_w)$ to the waiting times $(u_1, ..., u_w)$ defined by

$$u_1 = \tau_1$$

$$u_j = \tau_j - \tau_{j-1} \quad j \geq 2$$

Then the weight assigned to each path is

$$\frac{c^w e^{-h[u_1-u_2+u_3-...+\lambda-\sum_{k=1}^w u_k]} du_1...du_w}{\langle 1|e^{-\lambda[h\sigma_z-c\sigma_x]}|0\rangle}$$

which we can also write as

$$\frac{c^w e^{-\int_{t=0}^\lambda h(1-2S(t))dt} du_1...du_w}{\langle 1|e^{-\lambda[h\sigma_z-c\sigma_x]}|0\rangle}.$$

The algorithm is as follows:

1. Start at t=0 in state $S(0)$ defined by the boundary conditions. Define $B_1 = 1 - 2S(0)$. Set i=1.

2. Draw the waiting time u_i until the next flip from the distribution

$$f(u_i) = [\sqrt{h^2 + c^2} + B_i h]e^{-u_i[\sqrt{h^2+c^2}+B_i h]}$$

If $\sum_{j=1}^i u_j > \lambda$ then go to step 3. Otherwise define $B_{i+1} = -B_i$ and set $i \to i+1$ and repeat step 2.

3. Take the path you have generated (which will in general be longer than λ), and look at the segment $[0, \lambda]$. If this path satisfies the boundary condition at $t = \lambda$ then take this to be the generated path. Otherwise, throw away the path and repeat from step (1).

We now show that this algorithm generates paths from the distribution (22). Before conditioning on the boundary conditions being satisfied, the probability of generating a sequence of waiting times in $(u_1, u_1 + du_1), (u_2, u_2 + du_2), ...(u_w, u_w + du_w)$ followed by any waiting time u_{w+1} such that $u_{w+1} > \lambda - \sum_{j=1}^{w} u_i$ is given by

$$f(u_1)f(u_2)...f(u_w)du_1 du_2...du_w \mathrm{Prob}\left(u_{w+1} > \lambda - \sum_{j=1}^{w} u_i\right)$$

$$= f(u_1)f(u_2)...f(u_w)du_1 du_2...du_w e^{-(\lambda - \sum_{j=1}^{w} u_i)[\sqrt{h^2+c^2}+B_{w+1}h]}$$

$$= \left[\left(\prod_{i=1}^{w}[\sqrt{h^2+c^2}+B_i h]\right) e^{-\sum_{i=1}^{w} u_i \sqrt{h^2+c^2}} e^{-\sum_{i=1}^{w} u_i B_i h} du_1...du_w e^{-(\lambda - \sum_{j=1}^{w} u_j)[\sqrt{h^2+c^2}+SB_{w+1}h]}\right]$$

$$= \begin{cases} c^w e^{-\lambda\sqrt{c^2+h^2}} e^{-\int_{t=0}^{\lambda}(1-2S(t))h} du_1...du_w, & \text{if } w \text{ is even} \\ c^w \left[\sqrt{1+(\frac{h}{c})^2} + B_1\left(\frac{h}{c}\right)\right] e^{-\lambda\sqrt{c^2+h^2}} e^{-\int_{t=0}^{\lambda}(1-2S(t))h} du_1...du_w, & \text{if } w \text{ is odd }. \end{cases}$$

In the last line we have used the difference of squares formula to simplify consecutive terms: $[\sqrt{c^2 + h^2} + h][\sqrt{c^2 + h^2} - h] = c^2$. When we condition on fixed boundary conditions (whatever they may be), this generates the correct distribution over paths.
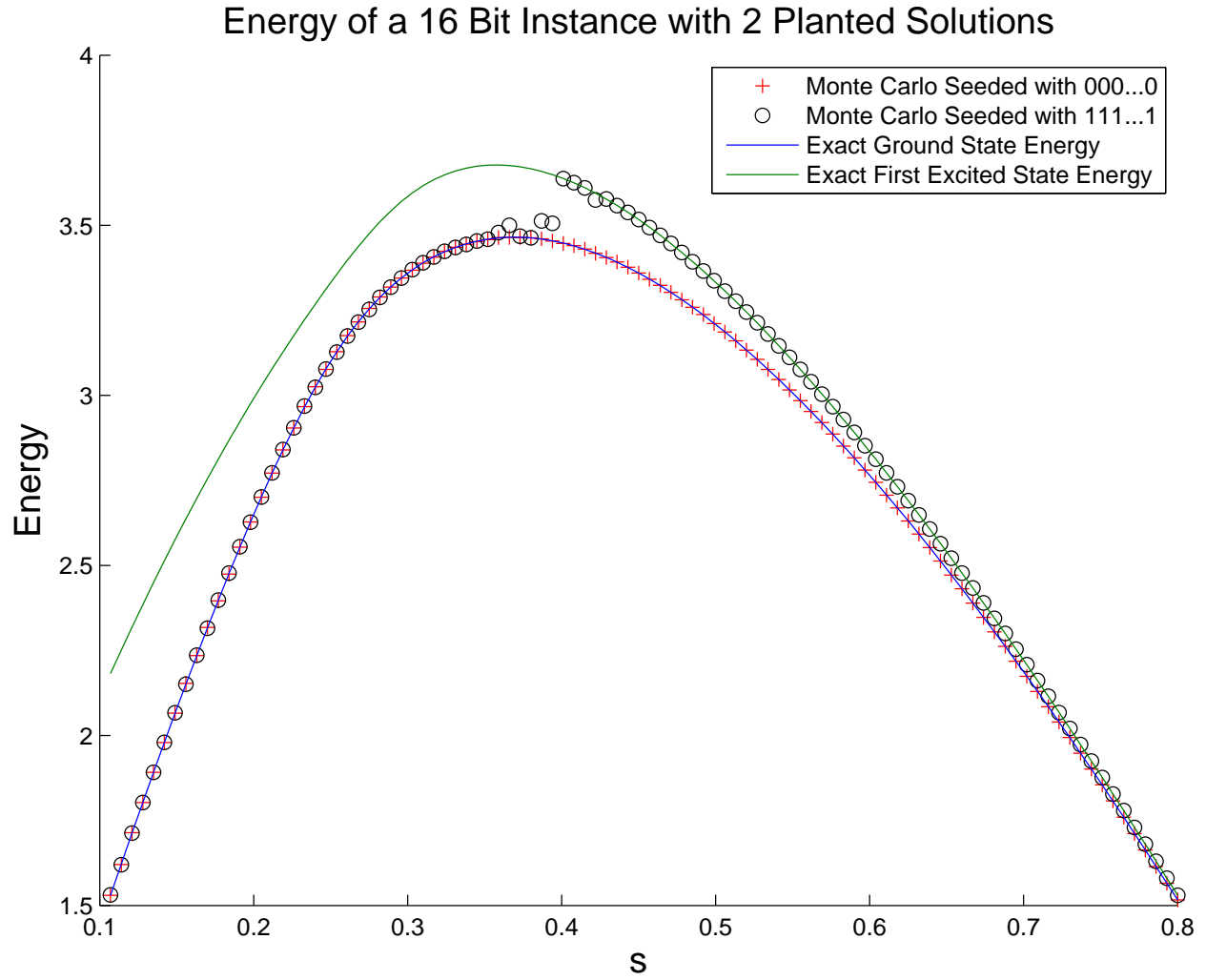
Figure 5: The discontinuity in the circle data that occurs near $s = 0.4$ is a Monte Carlo effect that we understand. As can be seen from the exact numerical diagonalization there is no true discontinuity in either the ground state energy or the first excited state energy. For s greater than 0.4 the Monte Carlo simulation is in a metastable equilibrium that corresponds to the first excited state. The true ground state energy at each s is always the lower of the circle and the cross at that value of s.
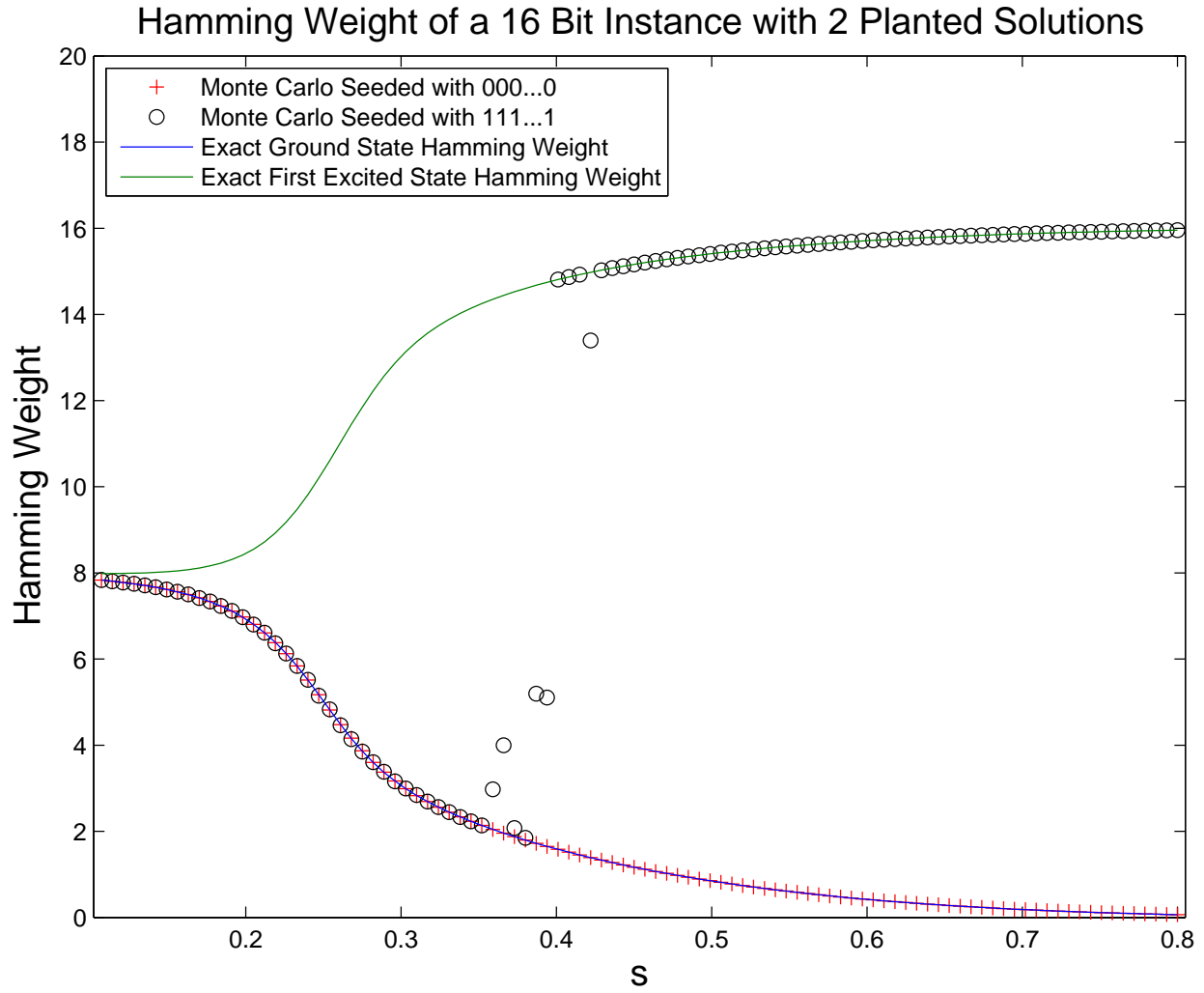
Figure 6: Together with figure 5, we see that the discontinuity in the circles appears in data for both the energy and the Hamming weight. This is not indicative of a phase transition in the physical system (as evidenced by the smooth curves computed by exact numerical diagonalization), but is purely a result of the way in which we use the Monte Carlo method.
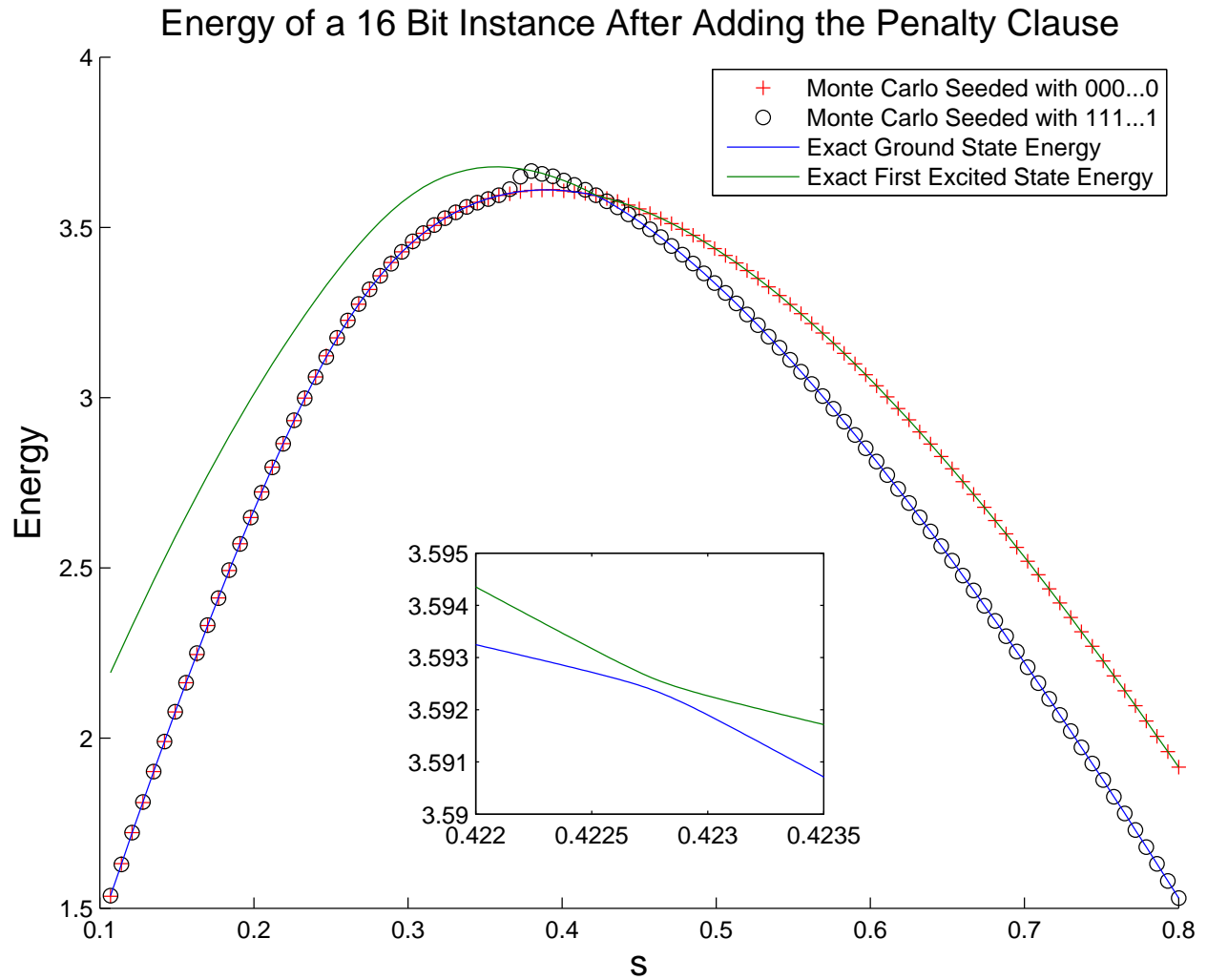
Figure 7: After adding the penalty clause we see that the energy levels have an avoided crossing at $s \approx 0.423$. The inset shows exact numerical diagonalization near the avoided crossing where we can resolve a tiny gap. The ground state energy is well approximated by the lower of the circle and cross at each value of s.
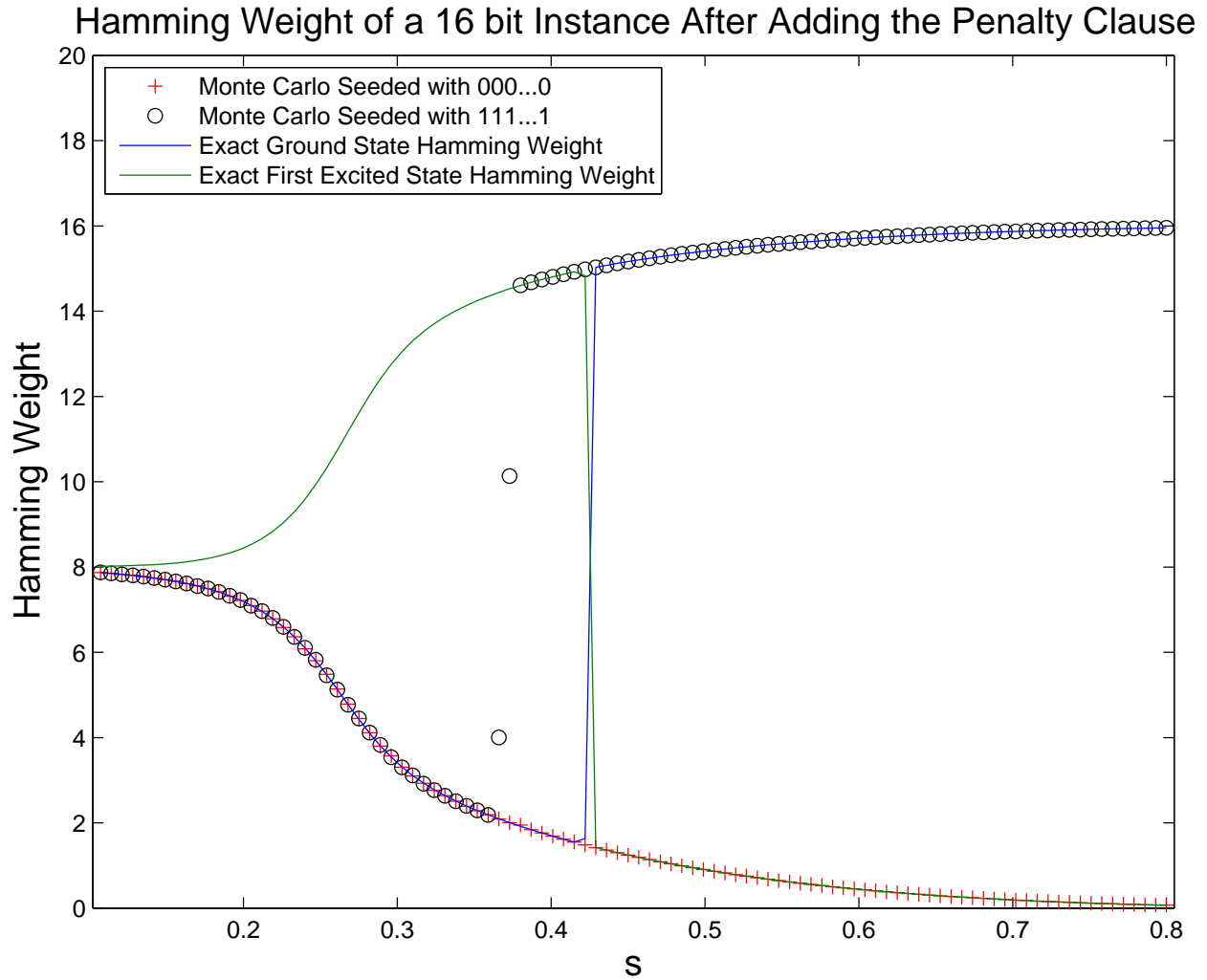
Figure 8: In this figure there is a phase transition which occurs near $s \approx 0.423$. We see from the exact numerical diagonalization that the Hamming weights of the first excited state and ground state undergo abrupt transitions at the point where there is a tiny avoided crossing in figure 7. There is also a jump in the Monte Carlo data plotted with circles that occurs before the avoided crossing: this is a Monte Carlo effect as discussed earlier and has no physical significance. If we look at the Monte Carlo data in figure 7 we conclude that below $s \approx 0.423$ the crosses represent the ground state and after this point the circles represent the ground state. From the current figure, along with figure 7, we conclude that the Hamming weight of the ground state jumps abruptly at $s^\star \approx 0.423$.
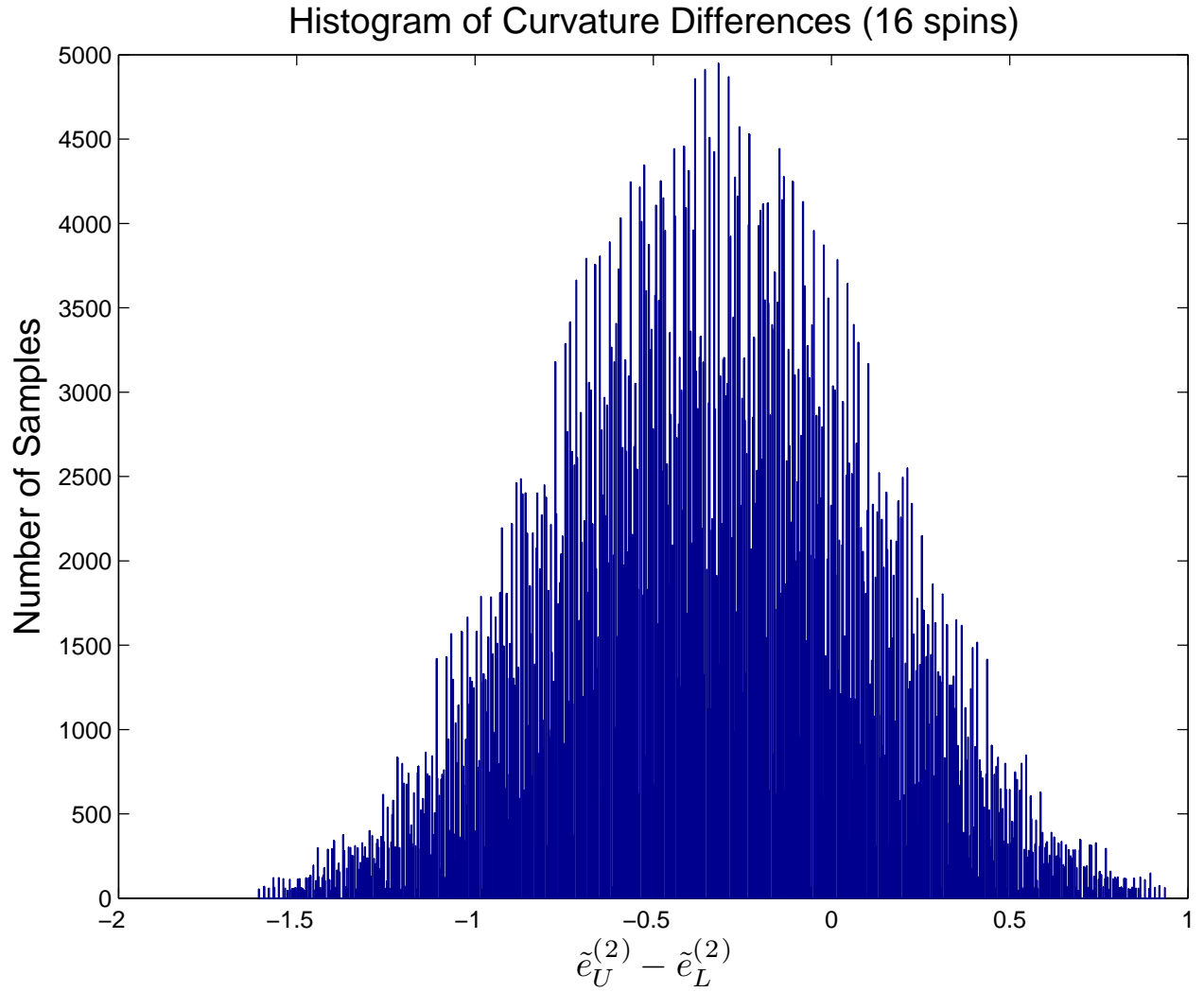
Figure 9: The histogram of $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ for 1 million different choices of coefficients c_i shows a substantial tail for which $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > 0$. These sets of coefficients correspond to beginning Hamiltonians $\tilde{H}_B$ for which we expect the small gap in figure 7 to be removed.
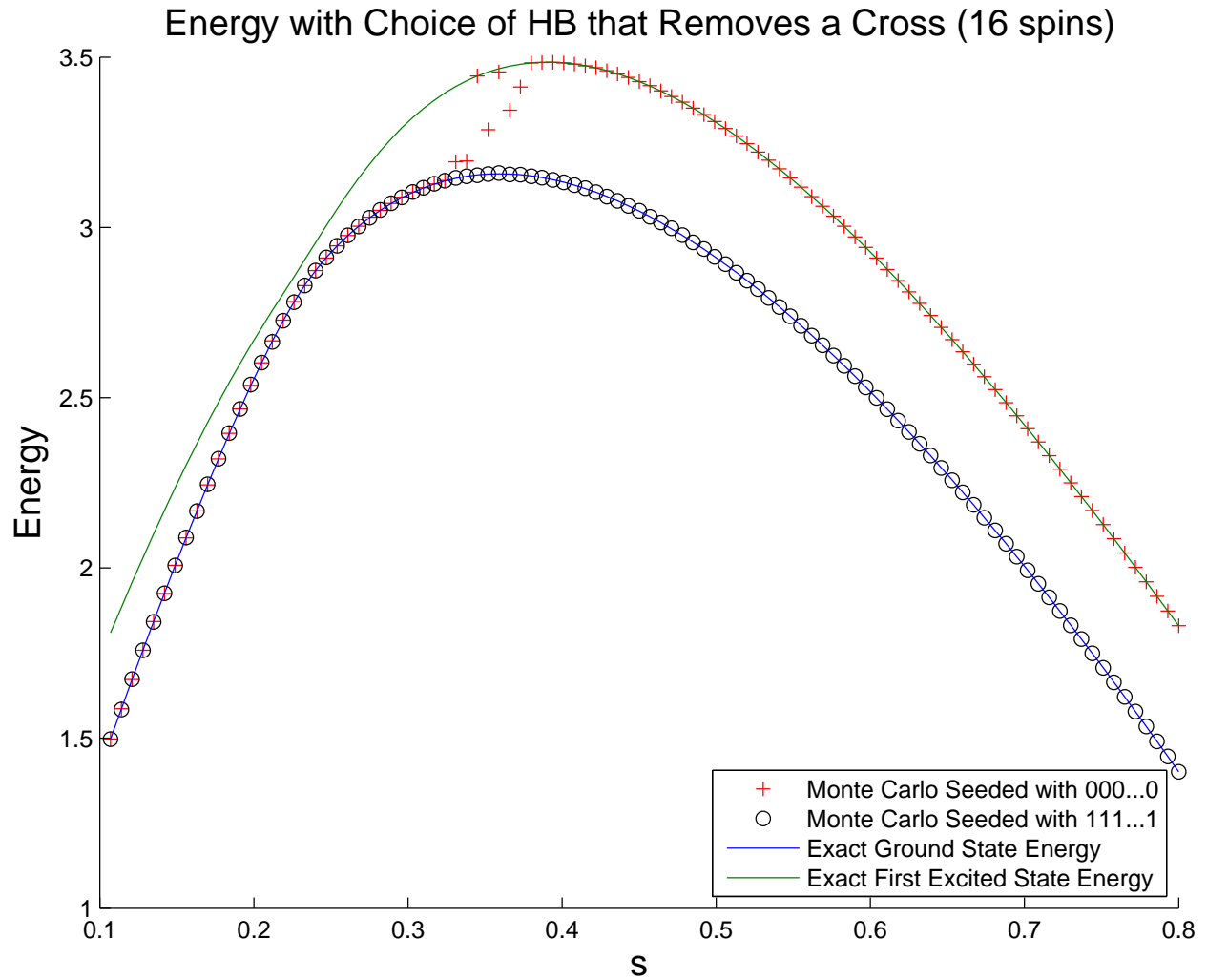
Figure 10: A random beginning Hamiltonian removes the crossing seen in figure 7. The problem Hamiltonian is the same as that in figure 7. The circles are always below (or equal to) the crosses for all values of s. This means that the circles track the ground state for all s and we see no sign of a small gap in the Monte Carlo data. This is consistent with the displayed exact numerical diagonalization.
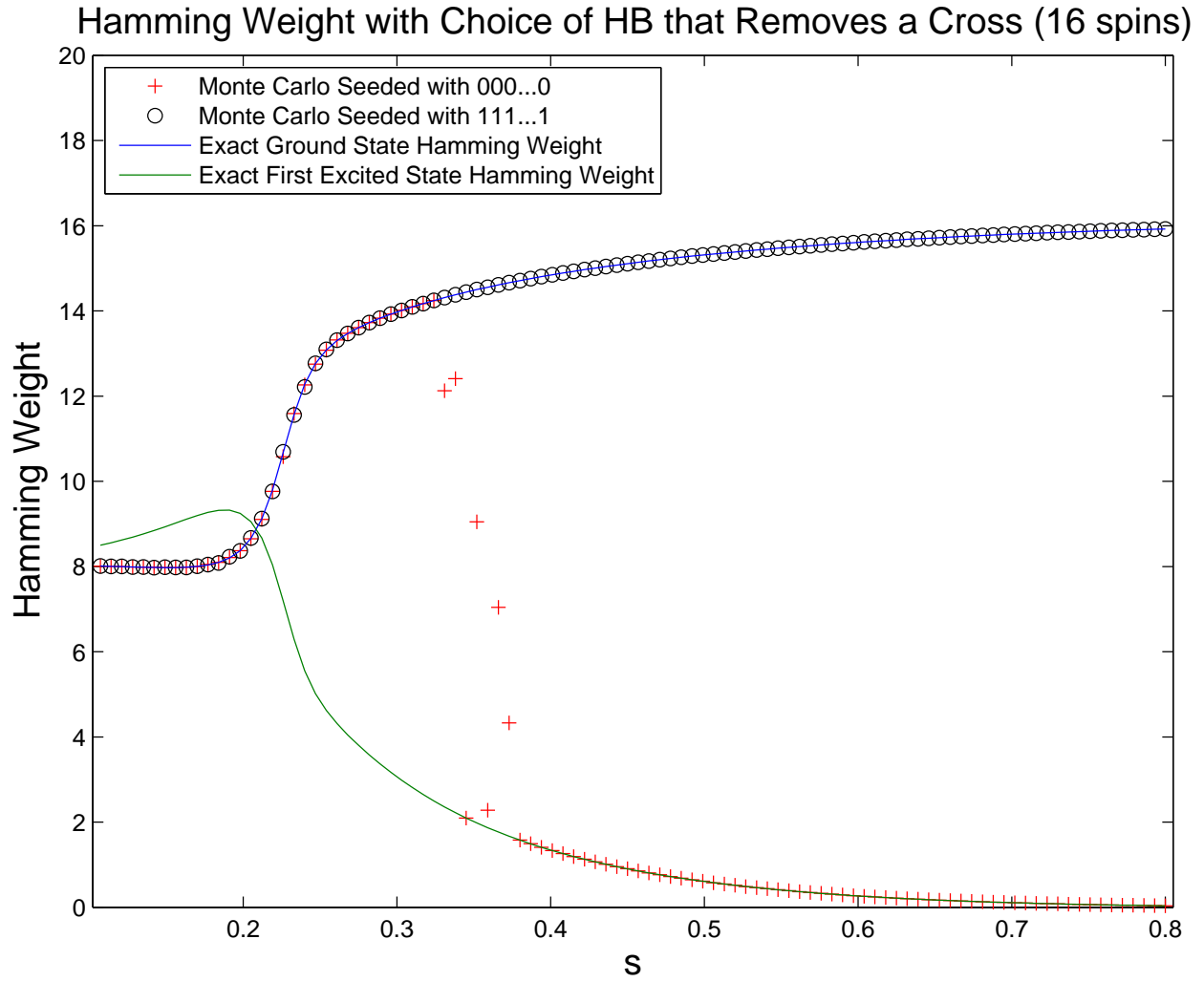
Figure 11: From figure 10 we see that the ground state of the Hamiltonian corresponds to the circles for all values of s and the Hamming weight of the circles here goes smoothly to the Hamming weight of the unique satisfying assignment. (The jump in the Monte Carlo data corresponding to the crosses is due to the Monte Carlo effect discussed earlier.)
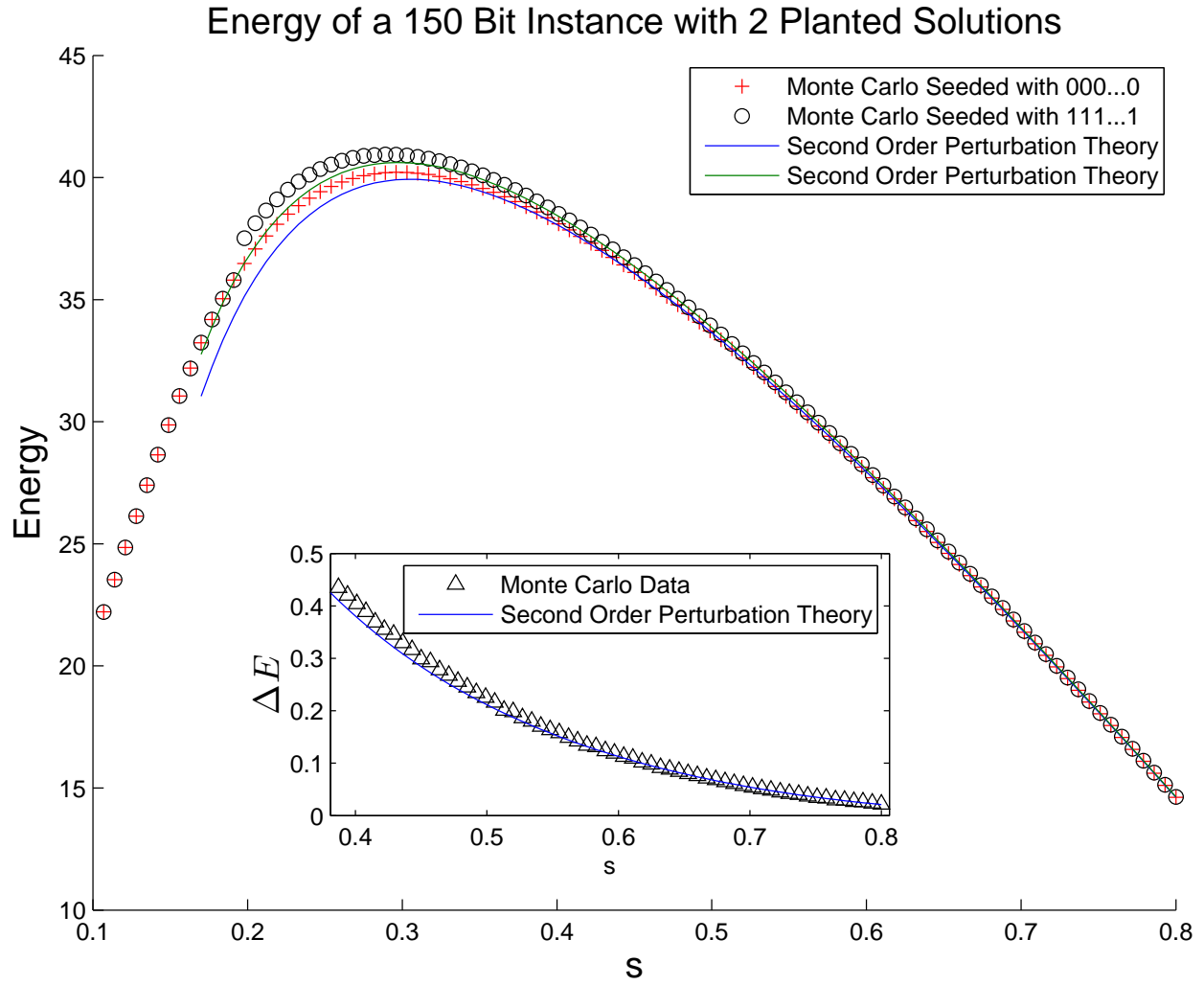
Figure 12: The crosses, which represent the Monte Carlo data seeded with 111...1, are below (or equal to) the circle data for all values of s. (This is seen more clearly in the inset which shows the positive difference between the circle and cross values). We conclude that the crosses track the ground state energy which is smoothly varying. The jump in the circle data is a Monte Carlo effect and for s above 0.2 the circles track the first excited state.
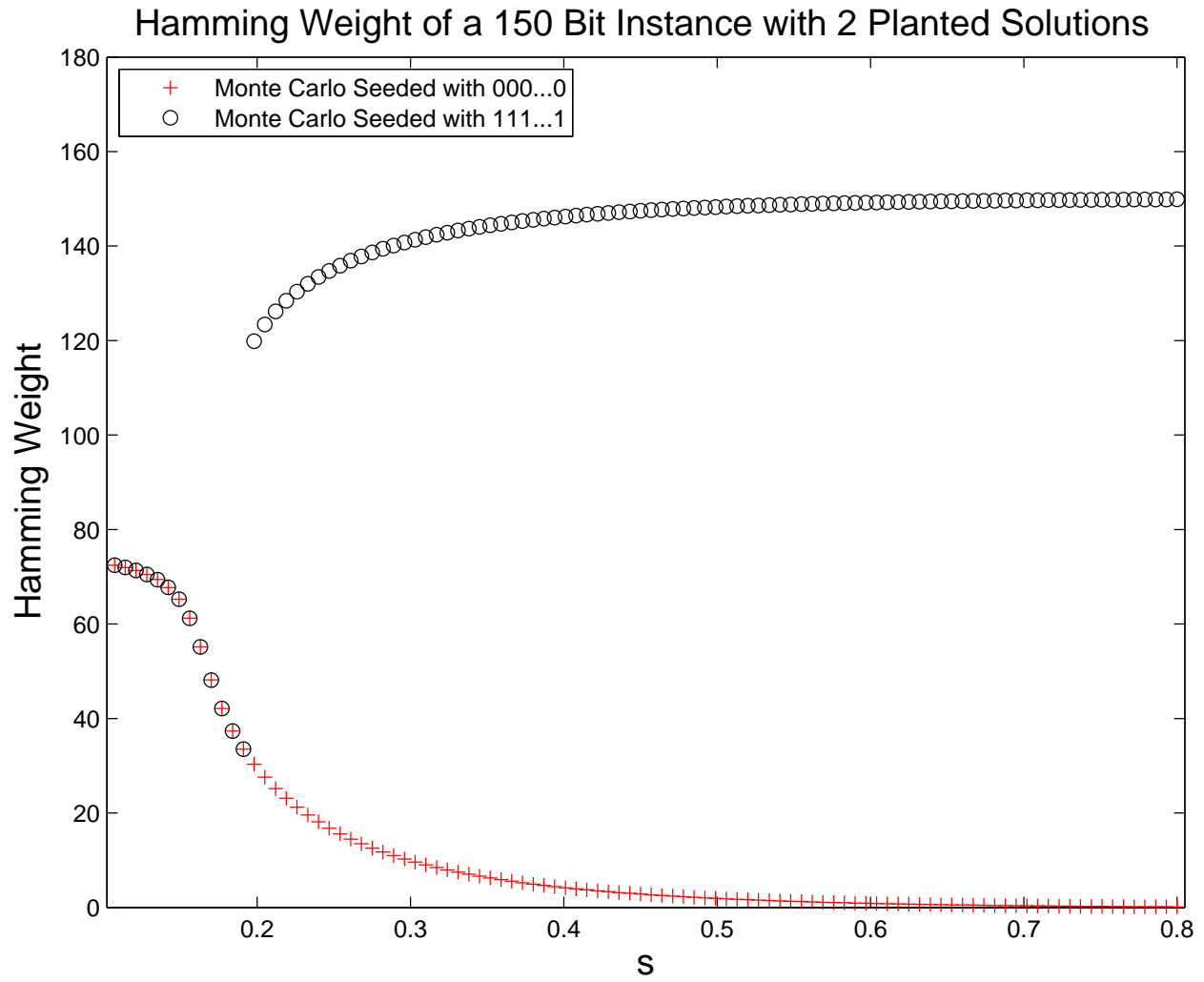
Figure 13: Comparing with figure 12 we see that the Hamming weight for the first excited state and the ground state are continous functions of s. We only obtain data for the first excited state for values of s larger than $s \approx 0.2$.
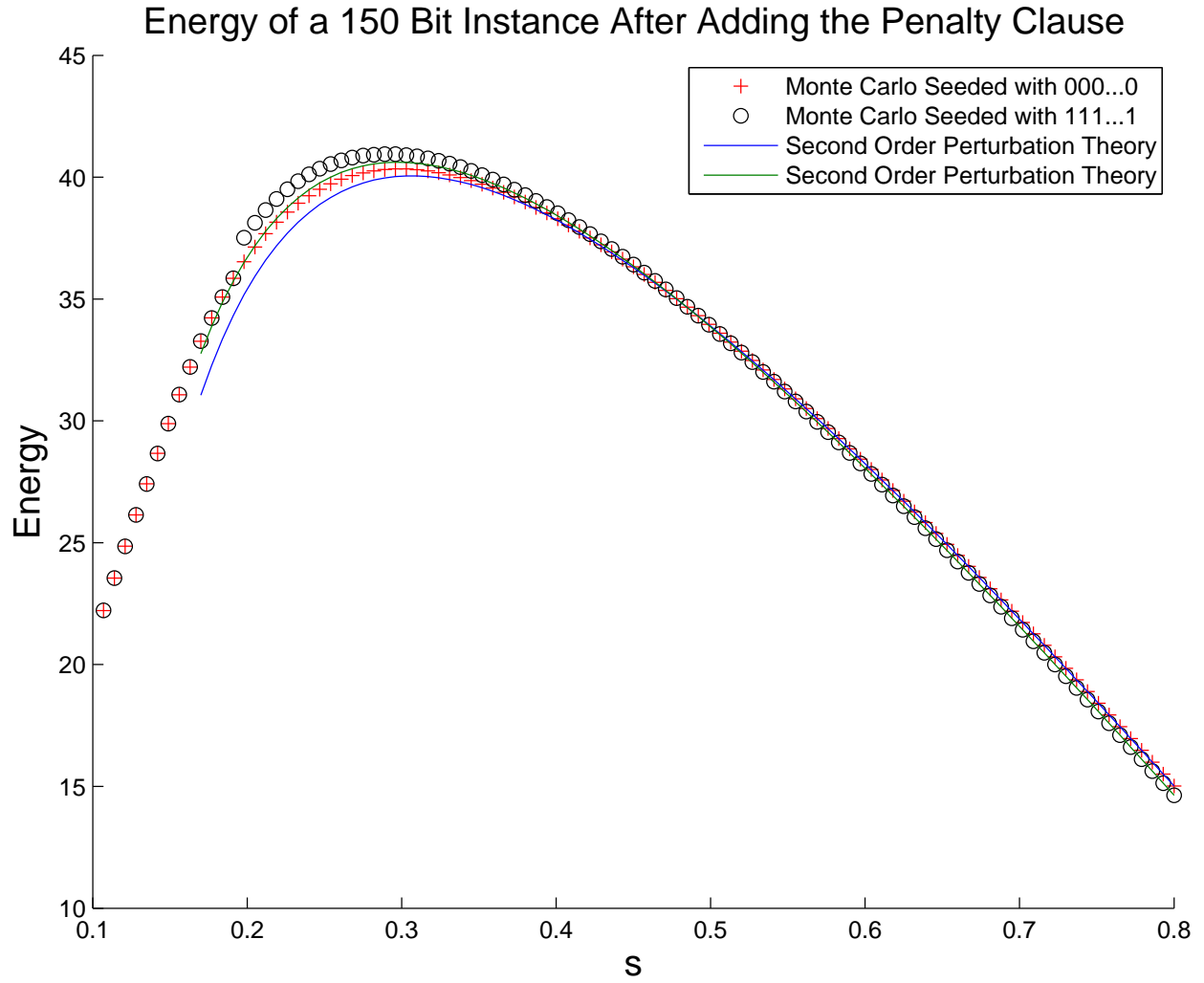
Figure 14: Adding the penalty clause makes the cross data go above the circle data at $s \approx 0.49$. This is shown in more detail in figure 15 where we plot the energy difference between the first two levels as a function of s. We interpret this to mean that the Hamiltonian $H(s)$ has a tiny gap at $s^\star \approx 0.49$.
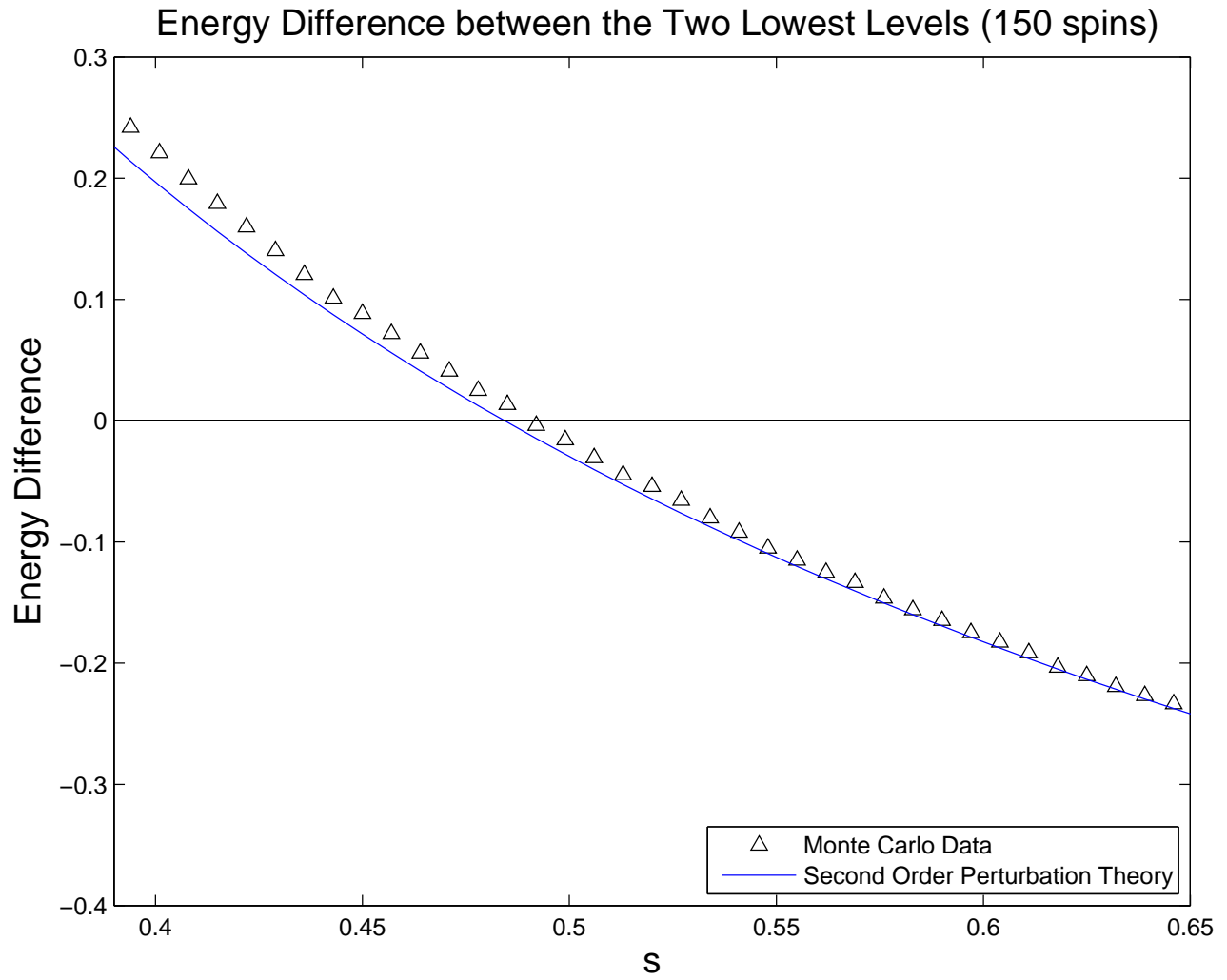
Figure 15: The energy difference, circles minus crosses, from figure 14 near the value of s where the difference is 0. Note that second order perturbation theory does quite well in predicting where the difference goes through zero.
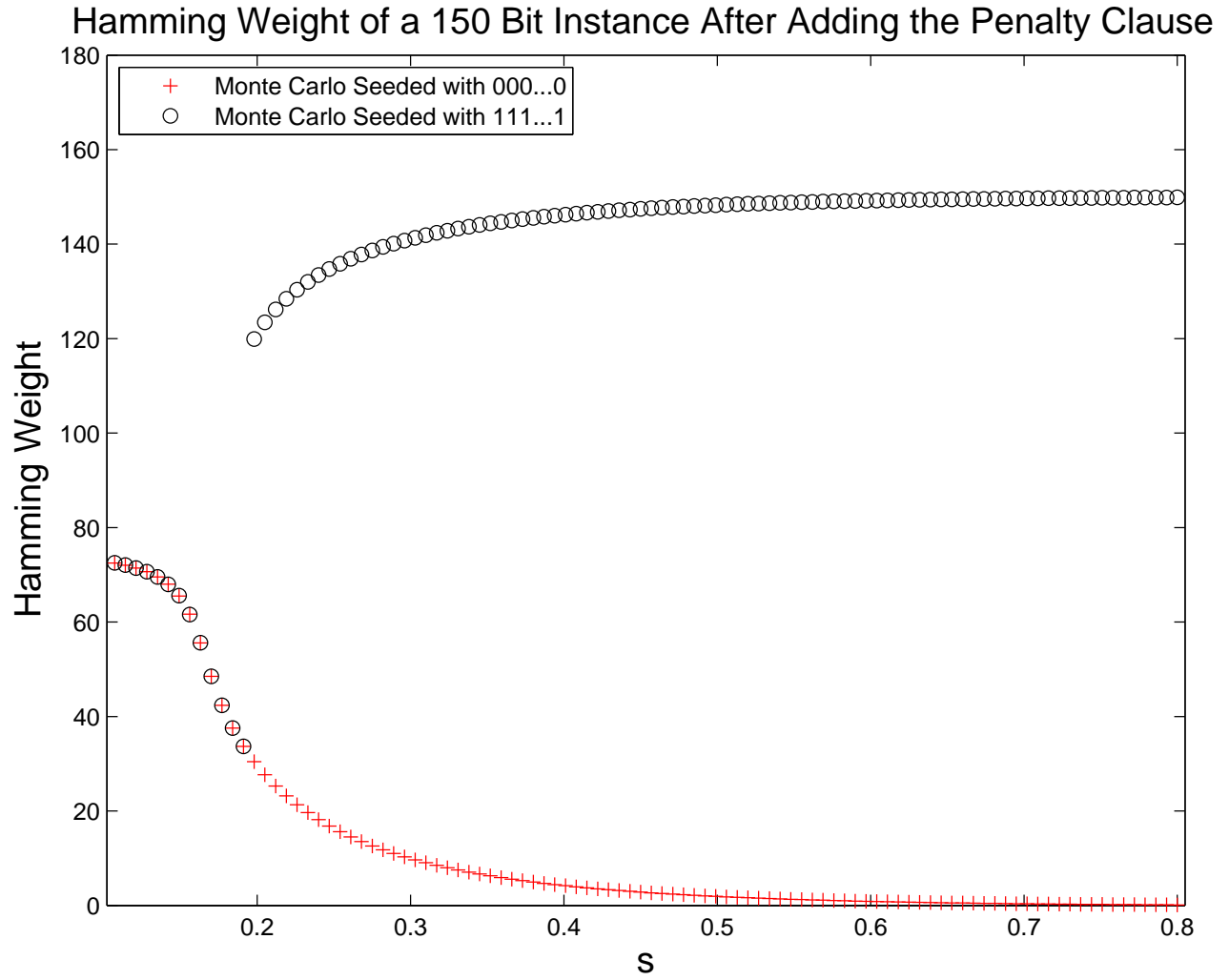
Figure 16: Looking at figure 14 we see that the ground state is represented by the crosses to the left of $s \approx 0.49$ and is represented by the circles after this value of s. Tracking the Hamming weight of the ground state, we conclude that it changes abruptly at $s \approx 0.49$.
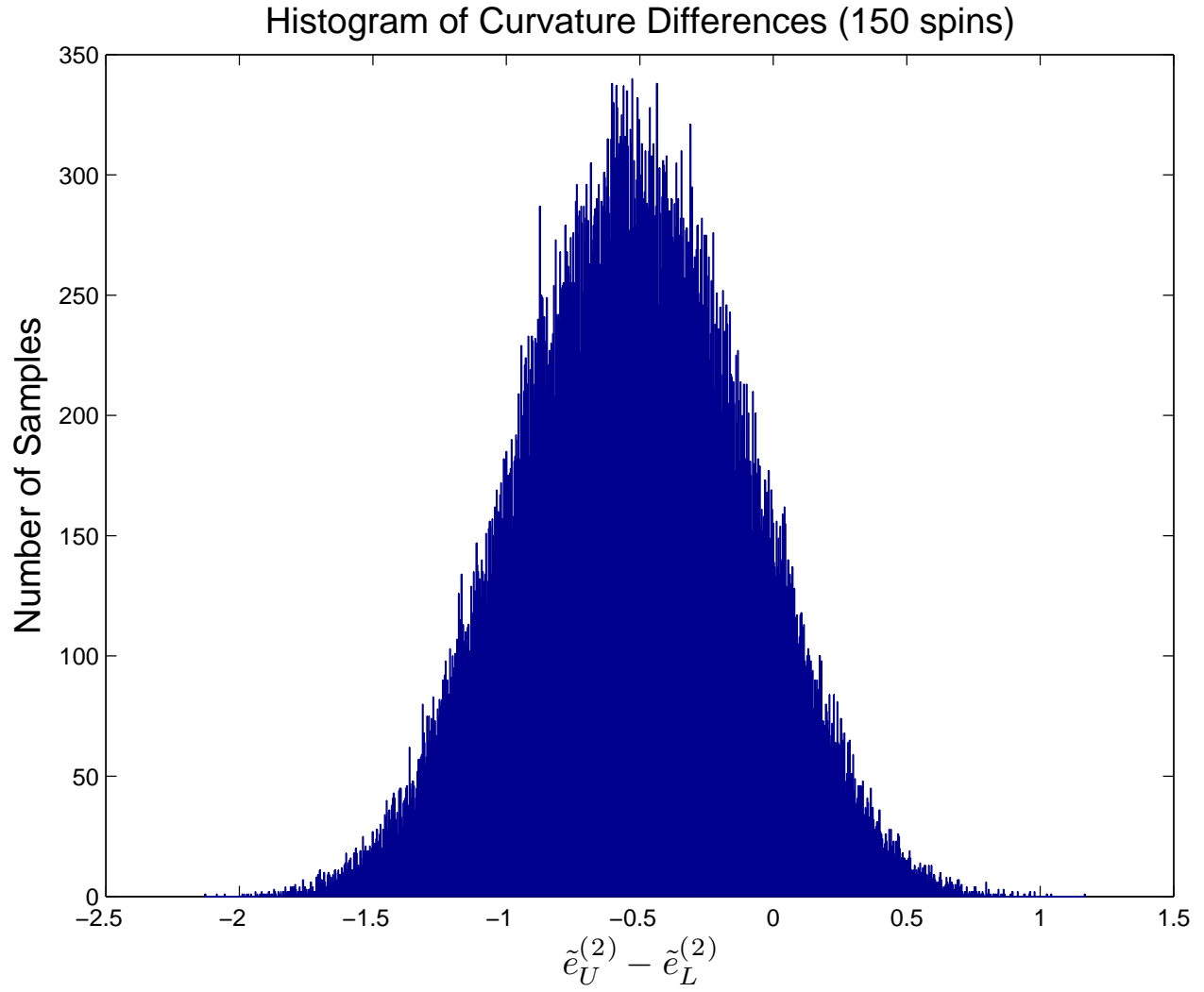
Figure 17: Histogram of $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)}$ for 100000 choices of coefficients c_i for our 150 spin instance. Note that a good fraction have $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > 0$.
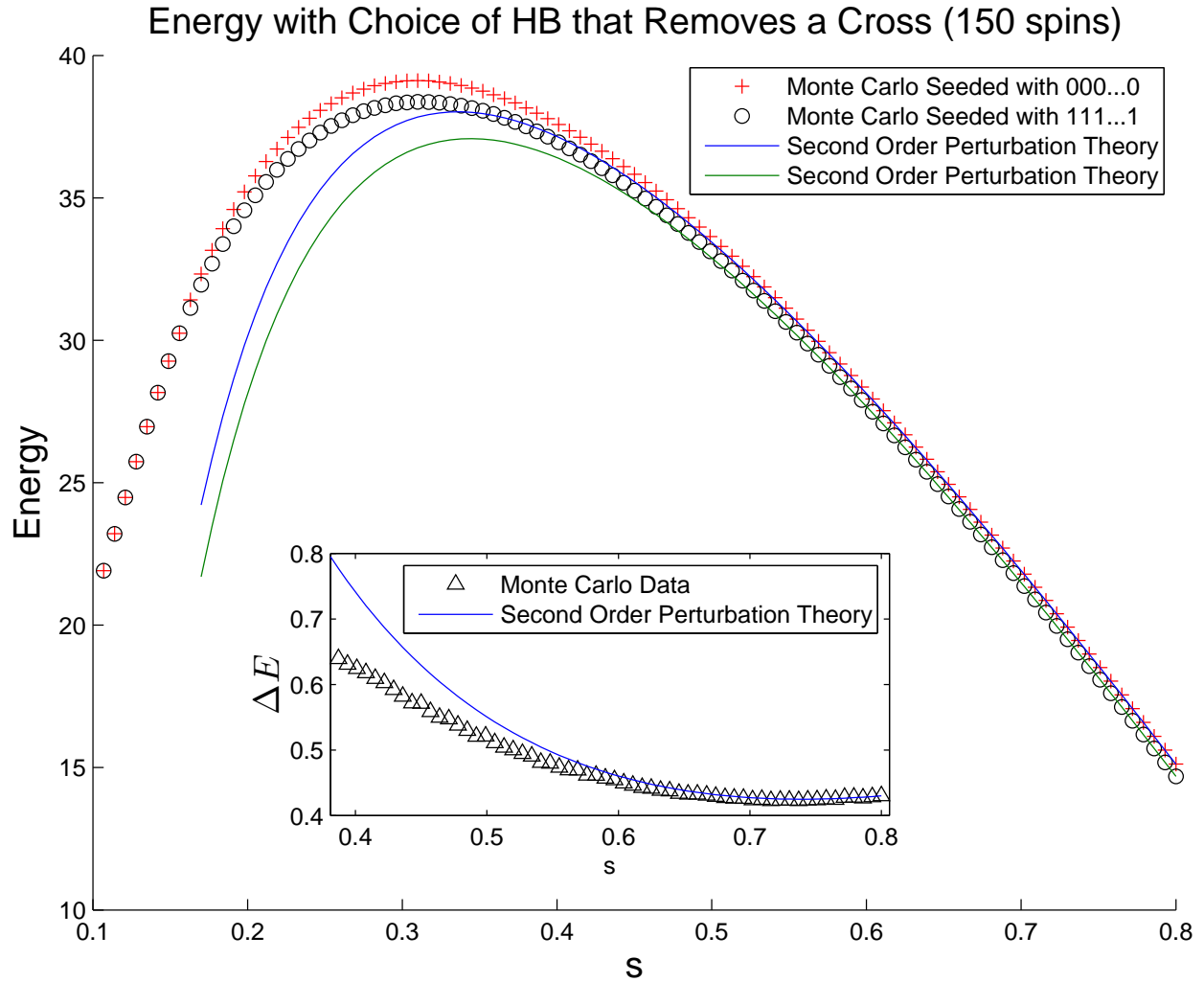
Figure 18: A random choice of coefficients such that $\tilde{e}_U^{(2)} - \tilde{e}_L^{(2)} > \frac{1}{2}$ gives rise to an $H(s)$ where there is no longer an avoided crossing. The circles here correspond to the ground state for all s since the cross data is always above (or equal to) the circle data for all s. This can be seen in the inset where we have plotted the energy difference, crosses minus circles. The crosses have a Monte Carlo discontinuity near $s \approx 0.2$, after which they correspond to the first excited state.
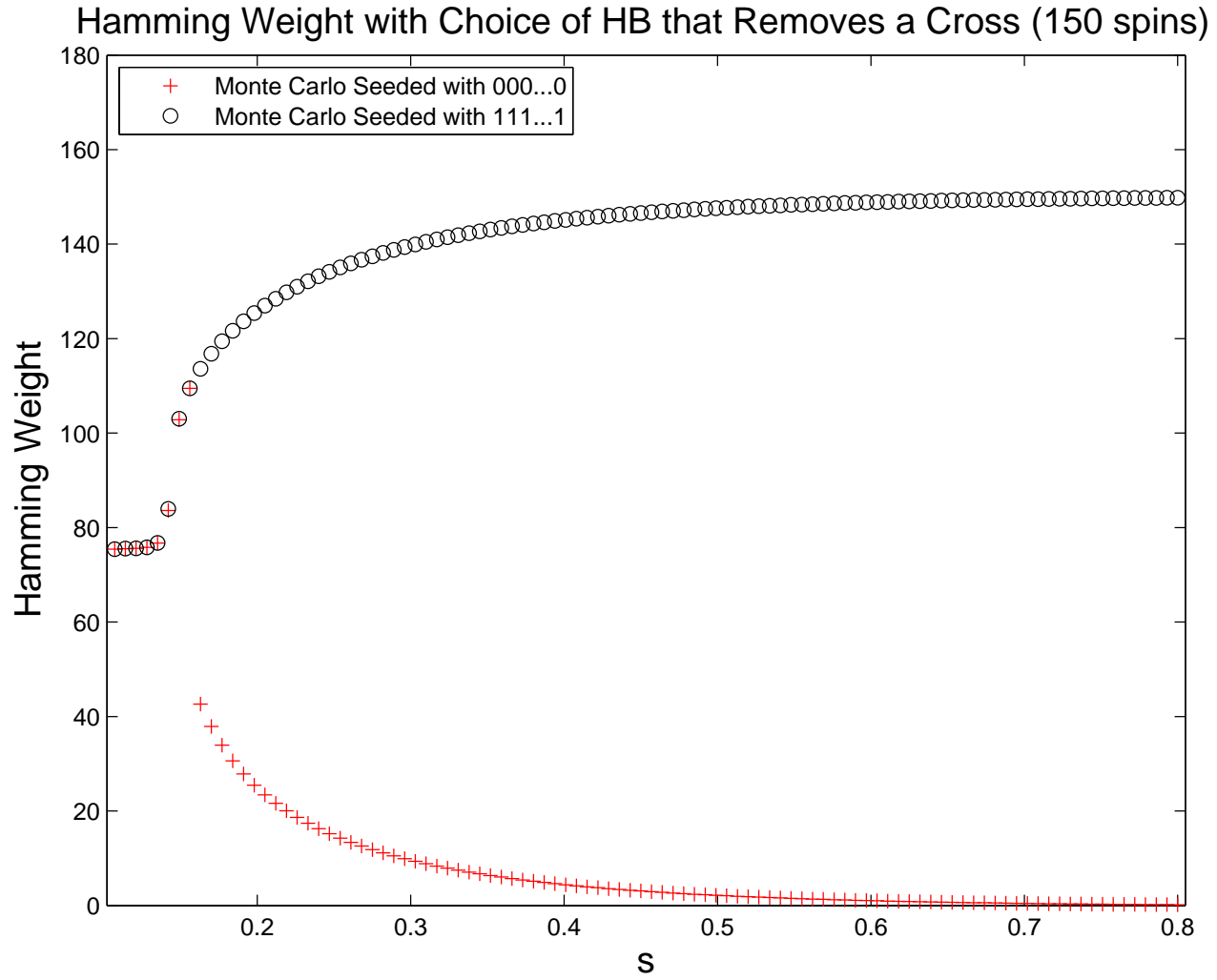
Figure 19: Looking at figure 18 we see that the ground state corresponds to the circles for all values of s so we see here that the Hamming weight of the ground state goes smoothly to its final value as s is increased. We take this as further evidence that this choice of $\tilde{H}_B$ would correspond to success for the quantum adiabatic algorithm for this instance.