String Model Building on Quantum Annealers

Steven Abel [1,2,3],* Luca A. Nutricati [2,3],† and John Rizos [4‡]

[1] *Theoretical Physics Department, Cern, 1211 Geneva 23, Switzerland*
[2] *Institute for Particle Physics Phenomenology, Durham University, Durham DH1 3LE, UK*
[3] *Department of Mathematical Sciences, Durham University, Durham DH1 3LE, UK and*
[4] *Department of Physics, University of Ioannina GR45110 Ioannina, Greece*

We explore for the first time the direct construction of string models on quantum annealers, and investigate their efficiency and effectiveness in the model discovery process. Through a thorough comparison with traditional methods such as simulated annealing, random scans, and genetic algorithms, we highlight the potential advantages offered by quantum annealers, which in this study promised to be roughly fifty times faster than random scans and genetic algorithm and approximately four times faster than simulated annealing.

CONTENTS

I. INTRODUCTION

There is continued interest in the problem of model selection in string theory which, due to the vast number of models available, presents a fascinating "big-data" challenge. Indeed depending on the set-up, the various estimates of the number of available models in the parameter space vary from the original 10^{500} estimate in type IIB flux compactifications [1, 2] to significantly larger numbers, for example $10^{272,000}$ F-theory flux compactifications on a single elliptically fibered four-fold [3]. In fact the number of Standard Model (SM) -like compactifications could itself be as large as 10^{700} [4].

Given this challenge attention has naturally turned to heuristic search methods. The argument for pursuing them is that nature itself finds solutions to problems without any difficulty within similarly large (or indeed much larger) search spaces. Heuristic methods typically attempt to mimic these natural processes. In the string context perhaps the most successful heuristic approach to date has proven to be the genetic algorithm (GA) [5, 6], which mimics evolution by implementing selection and breeding cycles on a population of would-be string solutions. Such methods have been shown capable of achieving orders of magnitude speed-up over blind scans in a string setting [7–18].

In this paper we shall consider a different class of heuristic search method in the string context, namely quantum adiabatic algorithms [19, 20]. Here we focus on the particular implementation of adiabatic computing known as Quantum Annealing (QA), in which the problem to be solved is mapped to to the minimisation of an Ising Hamiltonian. The full Hamiltonian in quantum annealing comprises an admixture of this Ising *problem-Hamiltonian* and a trivial Hamiltonian for which the ground state is known. The original idea behind quantum annealing (and quantum adiabatic algorithms more generally) is to begin in the ground state of the trivial system and adiabatically replace the trivial Hamiltonian with the problem Hamiltonian, while remaining in the ground state throughout. Provided we can remain in the ground state the final configuration will yield a solution to the problem. More modern approaches have extended this idea (for example using reverse annealing, of which more later) however the basic principle of arranging an interplay between a problem Hamiltonian and a trivial Hamiltonian is universal.

Quantum annealing has been utilised in many simple settings, notably for solving network problems, but its application in high energy theory has up to now been somewhat limited (see Refs. [21–35] for some examples). In the string setting, quantum annealing has been employed in hybrid algorithms, for example most recently in Genetic Quantum Annealing [33, 34] in which the GA performance is improved with a quantum annealing stage, however it has not to date been possible to implement a full string search directly on a quantum annealer

---
* steve.abel@durham.ac.uk
† luca.a.nutricati@durham.ac.uk
‡ irizos@uoi.gr

(or indeed on a quantum computer of any kind). Nevertheless the recent study in Ref. [36] laid the foundations for achieving this goal by showing that quantum annealers can solve the kind of discrete problems (for example satisfying anomaly cancellation conditions) that one typically encounters in string model building.

In the present study we will break new ground by for the first time implementing string models directly on a quantum annealer, and using the annealer to search for string theories with SM-like properties. The general approach we shall use is a combined technique which embeds the string consistency conditions (*a.k.a.* the GSO conditions) themselves on the annealer, but which performs certain additional phenomenological checks (such as for example selecting only models with three generations) during a second step. In such an arrangement the quantum annealer is essentially providing a consistency filtering of models, which may then be classically tested against other constraints. We shall see that this approach significantly enhances the overall efficiency of the algorithm. It can successfully be used to search parameter spaces orders of magnitude more quickly than either a blind scan *or* more traditional classical heuristic methods such as GAs.

II. ENCODING STRING MODELS ON A QUANTUM ANNEALER

We begin with the formulation of the models themselves. As described this is done using an Ising Hamiltonian: hence our first and indeed most difficult task is to encode the consistency conditions of the string theory by reformulating their solution as the minimisation of a function $H(\sigma_\ell)$ of spin variables $\sigma_\ell = \pm 1$, where ℓ labels the spin sites. This is the aforementioned *problem-Hamiltonian*, which is quadratic in the spins,

$$H(\sigma_\ell) = \sum_\ell h_\ell \sigma_\ell + \sum_{\ell m} J_{\ell m} \sigma_\ell \sigma_m , \qquad (1)$$

where each σ_ℓ corresponds to physical Z-spins on the machine.

The particular models we shall consider are the $SO(10)$ models described in Ref. [37]. The consistency conditions for these models amount to a set of generalized GSO (GGSO) projections determined by phases. The essential ingredients are described in Appendix A, to which we will continue to refer. The important aspect of these consistency conditions for the present study is that they can be written as a set of single qubit binary equations $f_A(\tau_i) = 0$, where $\tau_i \in \{0, 1\}$ map directly to the GGSO phases of 0 or π which determine the particular string model. The binary τ variables can in turn be mapped to annealer spins as

$$\tau_i = \frac{1}{2}(1 + \sigma_i) . \qquad (2)$$

Thus these are arguably the string models that can most readily be encoded in an Ising spin model. As we are

solving equations of purely single digit binaries we will use the names of the variables themselves to stand for the binary qubit value.

To get a broad idea of the form of the GGSO constraints, they partially consist of six (three for spinorial and three for vectorial representations) systems of four linear equations each, which can be synthesised in the following expression:

$$\Delta^I U_i^I = Y_i^I \mod 2, \quad \text{with } I = 1, 2, 3 \text{ and } i = s, v, \quad (3)$$

where each choice of (i, I) corresponds to a linear system of four equations (specifically $i = s, v$ refers to spinorial/vectorial representations and $I = 1, 2, 3$ refers to the three orbifold planes, respectively. See Appendix A for more details). The GGSO coefficients appear as components of the Y_i^I vectors (defined in Eqs. (A7), (A13)) and also as entries in the Δ^I matrices defined in Eqs. A3, A4, A5. Being related to phases, these coefficients take values of 0 and 1. Finally, U_i^I is a vector of four elements which denote solutions of the (i, I) system, also with entries in $\{0, 1\}$. We shall refer to the set of the solutions of the system as Ξ_i^I: it can contain at most $2^4 = 16$ solutions due to the binary nature of the components of the U_i^I vectors.

Once Eq. (3) is satisfied, *i.e.*, once we have a consistent GGSO projection, we will as described in the introduction further demand that a viable model must have 3 generations by imposing the following constraint (in which $i \equiv s$) classically [38]:

$$N_F = \sum_{I=1}^{3} \sum_{p,q,r,s \in \Xi_s^I} X_{pqrs}^{(I)} = 3, \qquad (4)$$

where $X_{pqrs}^{(I)} = \exp(i\pi \chi_{pqrs}^{(I)})$ and $\chi_{pqrs}^{(I)}$ are defined in Eqs. (A8), (A9) and (A10), for $I = 1, 2, 3$, respectively. We shall also require two additional properties: existence of at least one SM Higgs doublet and existence of a top Yukawa coupling. The first requirement corresponds to having at least one solution coming from one of the $\{(v, I), I = 1, 2, 3\}$ systems. The second requirement will be guaranteed by a particular choice of U_v^3 as we shall see in the following.

Having given an overview of the fundamental ingredients and phenomenological requirements that we will impose, let us now describe our method in detail. To encode the GGSO constraints we adopt a technique that is significantly different from those that have been used to analyse this specific class of models before, *e.g.* in Refs. [37–41]. Indeed the U_i^I parameters are typically scanned over along with the other variables corresponding to the GGSO phases. Here by contrast we first fix the values of U_s^I and U_v^I on the three orbifold planes which allows us to then search for suitable values of the GGSO coefficients. That is, following the discussion in

Appendix A, we can first without loss of generality set

$$U_s^1 = U_s^2 = U_v^3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{5}$$

Note that we fix U_v^3 to zero, which guarantees the existence of a top Yukawa coupling as we shall see. However we are still free to fix the residual parameters in U_i^I. For this study it is convenient to compare the different methods by studying the models with

$$U_s^3 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} , \; U_v^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} , \; U_v^2 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}. \tag{6}$$

Of course in a full treatment one would scan through the $2^{12} \approx 4000$ possible choices of U_s^3, U_v^1, U_v^2. Having fixed these parameters, the remainder of the GGSO constraints may be solved by quantum annealing. That is we are required to encode and solve the following equations on the quantum annealer:

$$\begin{cases} \Delta^3 U_s^3 &= Y_s^3 \\ \Delta^1 U_v^1 &= Y_v^1 \\ \Delta^2 U_v^2 &= Y_v^2 \\ \chi_{pqrs}^{(3)} &= 0 \end{cases} \quad \mod 2 \,, \tag{7}$$

where all the quantities involved can be expressed in terms of the GGSO coefficients following the definitions in Appendix A. Comparing Eqs. (7) with Eq. (3), one may wonder why spinorial projectors appear only on the third plane along with the corresponding chirality constraint. A similar question applies to the vectorial projectors, which are only present on the first and second plane. Indeed, it is straightforward to show that the choice in Eq. (5) (see Appendix A for details) trivialises the corresponding constraints on the first and second planes for spinorials as well as those on the third plane for vectorials. In a similar fashion, the chirality constraints on the first and second planes are satisfied by the conventions adopted in Eq. (A29).

The first three lines in Eq. (7) correspond to four linear equations, yielding a total of 13 equations with 27 unknown GGSO coefficients. Thus, we take the corresponding problem Hamiltonian to be effectively a "loss-function" for this set of equations, which is to say that we take it to be the sum of the squares of the 13 equations with additional integer parameters $K_{i=1,\dots,13}$ to absorb

the modulo 2 operation,

$$\begin{aligned} H &= \sum_{i=1}^4 \left(\Delta_{ij}^3 U_{s,j}^3 - Y_{s,i}^3 - 2K_i \right)^2 \\ &+ \sum_{i=1}^4 (\Delta_{ij}^1 U_{v,j}^1 - Y_{v,i}^1 - 2K_{4+i})^2 \\ &+ \sum_{i=1}^4 (\Delta_{ij}^2 U_{v,j}^2 - Y_{v,j}^2 - 2K_{8+i})^2 \\ &+ \left(\chi_{pqrs}^{(3)} - 2K_{13} \right)^2 \,, \end{aligned} \tag{8}$$

where the sum over j in each square is to be understood. The auxiliary variables $K_{i=1,\dots,13}$ are encoded using binary representations and take values in $[-3, 4]$, while Δ and Y are binary variables in $\{0, 1\}$, which are straightforwardly encoded in annealer spins via Eq. (2). This means that even for this restricted choice of U_i^I the parameter space is $2^{28} \approx 10^8$.

Once models have been acquired from this quantum annealing stage, they as mentioned need to be post-filtered classically to satisfy our additional phenomenological requirements. As discussed these conditions include the imposition of three generations, and the requirement of at least one Higgs doublet. However as explained in Appendix A the third constraint, namely the existence of a top Yukawa coupling, is already ensured by our choice of U_ν^3 and by Eqs. (A27) and (A28) and is therefore already encoded in Eq. (8). Thus, from on now on we need focus only on the first two conditions.

Let us start by analysing the 3 generations constraint. As already mentioned, the chirality on the first and on the second planes is set to one thanks to the conventions adopted in Eq. (A29). Therefore, the only chirality that needs to be checked is that on the third plane, which translates into the following equation (also without loss of generality):

$$\sum_{p,q,r,s\in\Xi_s^3} X_{pqrs}^{(3)} = \sum_{p,q,r,s\in\Xi_s^3} e^{i\pi\chi_{pqrs}^{(3)}} = 1\,. \tag{9}$$

As the reader may have already noticed, to ensure the desired number of generations we must sum over *all* the spinorial solutions on the third plane. However, it may happen that the proposed solution U_s^3 in Eq. (6) is not unique. In other words, since we are not imposing any constraint on the number of solutions, nothing prevents the system representing the spinorial projectors on the third plane from having solutions in addition to that already designated in Eq. (6). Indeed, the Hamiltonian in Eq. (8) does not contain a term that enforces the uniqueness of that solution: it simply guarantees that U_s^3 is a solution. Therefore, in order to ensure the fulfillment of the generations constraint, we post-process all the candidate models proposed by the annealer and discard those that have additional solutions besides the one that we have already assigned. Similar arguments hold for the

number of vectorials, however in this case the constraints are less severe as we only require there to be at least one Higgs doublet, that is we require at least one solution coming from one of the planes.

III. RESULTS AND PERFORMANCE COMPARISON WITH OTHER SEARCH METHODS

In order to perform our analysis the system in Eq. (8) was implemented on D-Wave's `Advantage_system4.1` architecture [42]: this annealer contains 5627 qubits, connected in a *Pegasus* structure, but only has a total of 40279 couplings between them. For more details of the physical realisation of the system the reader is referred to Ref. [36][1].

The crucial aspect of these systems for our discussion is the anneal process itself. The entire Hamiltonian on the annealer takes the form

$$\mathcal{H}(t) = A(t)H(\sigma_\ell^Z) + B(t)\sum_\ell \sigma_\ell^X , \qquad (10)$$

where the overall couplings $A(t)$ and $B(t)$ are adjusted during the anneal, and where the second piece, $\sum_\ell \sigma_\ell^X$ is the trivial Hamiltonian.

The adiabatic annealing paradigm chooses $A(t_{\mathrm{init}}) = 0$ and $A(t_{\mathrm{final}}) = 1$ while $B(t_{\mathrm{init}}) = 1$ and $B(t_{\mathrm{final}}) = 0$. However the quantum annealing method we use consists of several iterations of reverse annealing. Reverse annealing is a variation of the traditional quantum annealing approach: instead of starting the anneal in the groundstate of the trivial Hamiltonian, reverse annealing allows one to begin with the Hamiltonian as pure problem Hamiltonian $\mathcal{H}(t_{\mathrm{init}}) = H(\sigma_\ell^Z)$, with the qubits initialised in a specific eigenstate (most likely not its ground state). One then adjusts the system to parametrically approach the trivial Hamiltonian which induces a controllable "hopping" of σ_ℓ^Z spins before then returning to the pure problem Hamiltonian. In other words in a single anneal iteration we take $A(t_{\mathrm{init}}) = A(t_{\mathrm{final}}) = 1$ and $B(t_{\mathrm{init}}) = B(t_{\mathrm{final}}) = 0$, with non-zero values in between. The spins are then read from the annealer and used to initialise the next iteration.

Repeating the process and initialising each time with the best solution (in a fashion reminiscent of *elitism* in a GA), induces a kind of *quantum gradient descent* towards a solution over several iterations. This can in general lead to faster convergence and improved solutions.

To assess the different methods we first note that a comprehensive scan of this system is possible, and this

---

[1] It is worth mentioning that the quadratic Hamiltonian restriction does not apply to implementations of adiabatic quantum algorithms on gate quantum computers (at the expense of gate depth), as has been implemented in the Qibo architecture [43]. We return to this point later.
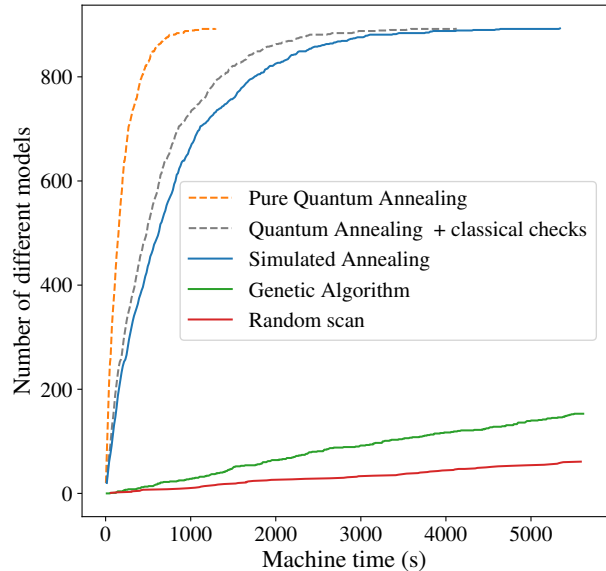

Figure 1. Comparison of the machine-time efficiency of various techniques for finding viable models. The methods analysed are: random scans, genetic algorithms, simulated annealing and quantum annealing. For the first three methods "machine-time" is equivalent to CPU-time. In the latter case it implies time on the annealer. The number of independent GGSO coefficients is 28, yielding to a search space of $2^{28} \sim 10^8$ possible models. The superiority of the methods using annealers is clearly evident in this case, surpassing the other techniques. The total number of models found using simulated annealing is 894 in approximately $6000s$ *vs* 153 and 61 models found using a genetic algorithm and a random scan, respectively. For the quantum annealer each anneal run had 2000 reads and used auxiliary variables $K_i \in [-3, 4]$. The dashed lines both indicate quantum annealing and are projections based on the simulated annealing data. In these two cases the machine-time does not take into account the dead-time required due to bottlenecks in exchanging information with the annealer (which incurs a delay of order a second, and which could be removed if a portion of the machine were dedicated to the search). The machine time is computed based on the anneal schedule time only, which is fixed to $160\,\mu s$. The grey line takes also into account the time required to do all the classical checks on the models found by the annealer. The orange line is an estimation of the performance of a quantum annealer supposing that all the classical checks are also encoded in the annealer Hamiltonian.

was performed for comparison: this took about 48 hours on a DELL PowerEdge R630 workstation with 32 GB of memory which resulted in a total number of 1024 models, *i.e.*, the search space of the set of models we have described contains one viable SM-like model in 10^5. The second method we considered was simulated annealing optimised with a linear β-schedule. We also for comparison implemented the same system in a genetic algorithm.

The GA was implemented with a fitness function defined linearly by the ranking, and optimised to find the best values for the mutation rate and for the learning rate (*i.e.*, the number of times breeding occurs for the fittest individual compared to the least fit): these were found to be 3% and 4, respectively.

To compare these four methods, we examined the rate at which each finds acceptable models in terms of machine-time. Apart from quantum annealing, all the other algorithms were run on standard computers. The results are collected in Fig. 1 where we plot number of found SM-like theories against machine-time. The definition of machine-time on quantum annealers is complicated by the fact that currently they are shared resources, and need to handle the influx of traffic and requests from multiple users simultaneously. This can lead to bottlenecks, resulting in longer wait times of a few seconds for accessing the machine or retrieving spin values after an anneal run. Since these are external factors (and even depend on the traffic on the machine) we have excluded them from our definition of quantum annealing machine-time. Indeed, these factors are the main obstruction preventing the reproduction of the saturation plot in Fig. 1. Hence the dashed quantum annealer lines are built upon the simulated annealing data by extrapolating comparative early performance. Specifically, the grey dashed line represents a projection of the quantum annealer performance, including the time required to perform the classical checks.

As we can see from the figure, both quantum and simulated annealing are far more efficient than random scans and genetic algorithms (red and green solid lines, respectively). Modulo the above caveats, in this case we estimate that the quantum annealing method is 1.3 times faster even than simulated annealing. The orange dashed line is an estimate of a hypothetical pure quantum annealing implementation, in which we assume that *all* of the additional classical checks are instead also encoded on the annealer (using methods that we discuss below), with no additional classical post-processing being required. This would result in a 4.125 times better performance than simulated annealing.

Overall then, in this study annealing methods appear to surpass other techniques in terms of performance, efficacy. They outshine alternative approaches and show superior outcomes. Quantitatively, the initial model discovery rate (*i.e.*, within the first 1000s) is one model in 0.66 seconds for simulated annealing, compared to one models in 33 seconds for the genetic algorithm and one in 100 seconds for the random scan. Using quantum annealing, the rate increases to one model in 0.50 seconds when model checks are performed classically (the grey line in Fig. 1) and to one model in 0.16 seconds for a hypothetical purely quantum annealing implementation (orange dashed line). However one caveat we should add for completeness is that the search in this study may be somewhat disadvantageous for the GA as the number of SM-like models in the search space is relatively dense: as

discussed in Ref. [8] a GA becomes more effective when the search is very difficult, say one model in 10^6 or more. The future challenge then is to compare quantum annealers and GAs when they are both confronted by much harder problems such as those discussed in Refs. [16, 34].

IV. TOWARDS PURE QA MODEL BUILDING

In Fig. 1 we included a line for hypothetical purely quantum annealing in which no classical post-processing would need to be done. In this section we briefly consider how such a complete implementation might be achieved. This discussion will illustrate the potential and also the complexity of performing pure quantum computing analyses.

The first hurdle for a complete quantum implementation of the models under discussion is the fact that the only spin Hamiltonians that can currently be considered are quadratic Ising models. Let us suppose for this discussion that this will remain the case for the foreseeable future. (We shall comment on alternative possibilities at the end of this section.) The problem then is how do we encode the higher order constraints on the annealer. For example if we do not fix the values of U_s^3, U_v^1, U_v^2 in advance but allow all the U parameters to be set by the quantum annealing (in which case we are searching the larger space of size 10^{12}) we are then obliged to encode the squares of Eqs. (6) in the Hamiltonian, which would be quartic in the spins.

Thus we will need to know how to turn a higher order polynomial of spins into a quadratic Ising Hamiltonian on the annealer. This is done by the method of reduction, described in Refs. [36, 44]. This method involves introducing auxilliary spins to represent pairs of spins in the original Hamiltonian, and it works as follows.

Let us begin with the raw high order polynomial $\tilde{H}(\sigma_\ell)$ written as a function of binary QUBO variables using Eq. (2). Suppose $\tilde{H}$ has terms involving products of two binary variables τ_1 and τ_2. Now consider adding to the polynomial $\tilde{H}$ a quadratic term that involves the binary variables together with a new auxiliary variable τ_{12}, which is of the form

$$Q(\tau_{12}; \tau_1, \tau_2) = \Lambda(\tau_1\tau_2 - 2\tau_{12}(\tau_1 + \tau_2) + 3\tau_{12}) . \quad (11)$$

Inspection shows that a sufficiently large and positive overall coupling Λ enforces $\tau_{12} = \tau_1\tau_2$ whatever the values of τ_1 and τ_2 happen to be. Importantly the minimum at this point has $Q = 0$. Therefore we may replace the product $\tau_1\tau_2$ with τ_{12} wherever it appears within $\tilde{H}$, and the new Hamiltonian is guaranteed to have the same set of minima as the original $\tilde{H}$. This process can be iterated until one arrives at the desired problem-Hamiltonian which is quadratic in spins, and which is schematically

of the form

$$H = \tilde{H}(\tau_1, \tau_2, \ldots, \tau_{12}, \tau_{13}, \ldots, \tau_{12,34}, \tau_{12,56} \ldots)$$
$$+ \sum_{i>j} Q(\tau_{ij}; \tau_i, \tau_j) + \sum_{i<j,k<m} Q(\tau_{ij,km}; \tau_{ij}, \tau_{km})$$
$$+ \ldots \qquad (12)$$

with the constraints imposed by the Q terms ensuring that this quadratic Hamiltonian has the same minima as the original polynomial. Although this procedure may seem laborious it can easily be automated to systematically reduce any higher order polynomial in spins to a quadratic, as in the explicit examples in Ref. [36].

Thus in principle all the remaining degrees of freedom may be put on to the annealer with the GGSO constraints being entirely enforced by the Hamiltonian, at the expense of using an additional ancillary qubit for every term in the Hamiltonian that requires reducing.

There remains the problem of incorporating phenomenological counting constraints. That is, suppose that the annealer with this larger system has found a solution to Eqs. (7) having thereby determined a set of U's and Δ's. We are required to include something in the Hamiltonian which will now implement the three generations check in Eq. (4) which previously we did classically. This requires the introduction of "counting" qubits and it can be achieved as follows.

We are required to impose Eq. (4) which boils down to imposing $\#(\chi = \text{even}) = 3 + \#(\chi = \text{odd})$, or in other words

$$\sum_{I=1}^{3} \sum_{p,q,r,s \in \Xi_s^I} \left(2\,\overline{\chi}_{pqrs} - 1 \right) = -3 , \qquad (13)$$

where $\overline{\chi} \equiv \chi \mod(2)$. Note that the quantity on the left is actually just given by the sum of the σ qubits corresponding to the binary $\overline{\chi}$ variable. This is sufficient to ensure that even if we have cancelling positive and negative chiralities the nett number of generations is 3.

In these equations the χ_{pqrs} are functions of the p, q, r, s as in Eqs.(A8),(A9),(A10), so they are also polynomial objects in spins. However each χ can be mapped to a single ancillary binary qubit $\overline{\chi}$ by adding the terms

$$H \supset (\chi + 2K_\chi - \overline{\chi})^2 \qquad (14)$$

for every χ, where here $\overline{\chi}$ is another ancillary binary qubit and K_χ is as before a binary encoded integer, $K_\chi \in \mathbb{Z}$. Thus $\overline{\chi} = \chi$ is enforced at the minimum. Of course given that the χ_{pqrs} in Eqs. (A8),(A9),(A10) are not linear in spins, Eq. (14) will also require reduction. Finally to impose the three generation constraint in Eq. (13) we then add to the quadratic Hamiltonian the term

$$H \supset \left(\sum_{I=1}^{3} \sum_{p,q,r,s \in \Xi_s^I} \left(2\overline{\chi}_{pqrs} - 1 \right) + 3 \right)^2 \qquad (15)$$

which is quadratic and therefore requires no further reduction.

In principle therefore all the consistency conditions may be straightforwardly implemented on the annealer in this fashion. Currently the limiting factor is the size of the architecture and the relatively large number of ancillary qubits that would be generated by this method. Therefore one might also contemplate an alternative approach which is to implement annealing on a quantum gate computer. In such an approach the Hamiltonian is "Trotterized" in order to evolve the system in small time-steps on a universal gate quantum computer [43]. Thus in principle spin Hamiltonians of high order are allowed and no reduction would be required in order to implement all the consistency conditions. Currently no such system of large enough physical size is available to encode the system under discussion, and unfortunately one cannot simulate more than approximately 30 qubits. However this approach would be a promising avenue to explore once universal gate machines of sufficient size become available.

V. CONCLUSIONS

In this study, we have employed quantum annealing to construct string models, focusing on their efficiency and effectiveness in the model discovery process. By comparing quantum annealing with other established methods such as simulated annealing, random scans, and genetic algorithms, we have gained valuable insights into the possible advantages of using quantum annealers for this purpose.

We should add that annealers are possibly most advantageous when the search space consists of relatively dense regions of SM-like models (in this study one model in 10^5), a situation in which genetic algorithms do not usually lead to significant improvement with respect to alternative methods such as random scans. By contrast, genetic algorithms are known to excel in scenarios with more challenging searches, where the exploration of extremely large solution spaces is required. Therefore, it would be interesting in future investigations to compare these methods in more difficult problem domains in order to provide a comprehensive assessment of their respective strengths and weaknesses.

Appendix A: Construction of $SO(10)$ models

In this article we focus on heterotic string models defined in the free fermionic formulation using the basis $b = \{\beta_1, \ldots, \beta_{12}\}$, where

$$\beta_1 = \mathbb{1} = \{\psi^\mu, x^{1,\ldots,6}, y^{1,\ldots,6}, \omega^{1,\ldots,6};$$
$$\bar{y}^{1,\ldots,6}, \bar{\omega}^{1,\ldots,6}, \bar{\psi}^{1,\ldots,5}, \bar{\eta}^{1,2,3}, \bar{\phi}^{1,\ldots,4}, \bar{\phi}^{5,\ldots,8}\},$$
$$\beta_2 = S = \{\psi^\mu, x^{1,\ldots,6}\},$$
$$\beta_{2+i} = e_i = \{y^i \omega^i; \bar{y}^i, \bar{\omega}^i\}, i = 1, \ldots, 6, \quad \text{(A1)}$$
$$\beta_9 = b_1 = \{x^{34}, x^{56}, y^{3,4}, y^{5,6}; \bar{y}^{3,4}, \bar{y}^{5,6}, \bar{\psi}^{1,\ldots,5}, \bar{\eta}^1\},$$
$$\beta_{10} = b_2 = \{x^{12}, x^{56}, y^{1,2}, y^{5,6}; \bar{y}^{1,2}, \bar{y}^{5,6}, \bar{\psi}^{1,\ldots,5}, \bar{\eta}^2\},$$
$$\beta_{11} = z_1 = \{\bar{\phi}^{1,2,3,4}\},$$
$$\beta_{12} = z_2 = \{\bar{\phi}^{5,6,7,8}\},$$

and a set of phases $c\begin{bmatrix}\beta_1\\\beta_1\end{bmatrix} = \pm 1, c\begin{bmatrix}\beta_i\\\beta_j\end{bmatrix} = \pm 1, i > j = 1, \ldots, 6$. The basis vectors β_i describe the parallel transportation properties of the fermionic coordinates along the world-sheet torus while the phases link to generalised GSO projections (GGSO). Following the standard notation, included fermions are periodic, while all rest are anti-periodic. For $c\begin{bmatrix}S\\e_i\end{bmatrix} = c\begin{bmatrix}s\\z_a\end{bmatrix} = -1, i = 1, \ldots, 6, a = 1, 2$ and generic choice of the remaining GGSO phases, the above basis describes a $\mathcal{N} = 1$ supersymmetric model possessing $SO(10) \times U(1)^3 \times SO(8)^2$ gauge symmetry.

$SO(10)$ spinorials arise from the sectors $\mathcal{S}^I_{\vec{P}^I_s} = S + b_I + \vec{P}^I_s \cdot \vec{E}$, $I = 1, 2, 3$ where $P^1_s = (0, 0, p^1_s, q^1_s, r^1_s, s^1_s)$, $P^2_s = (p^2_s, q^2_s, 0, 0, r^2_s, s^2_s)$, $P^3_s = (p^3_s, q^3_s, r^3_s, s^3_s, 0, 0)$ and $\vec{E} = (e_1, e_2, e_3, e_4, e_5, e_6)$. Here, $b^3 = b^1 + b^2 + x$ with $x = \mathbb{1} + S + \sum_{i=1}^6 e_i + \sum_{a=1}^2 z_a$. Similarly, $SO(10)$ vectorials come from the sectors $\mathcal{V}^I_{\vec{P}^I_v} = S + b_I + x + \vec{P}^I_v \cdot \vec{E}$, $I = 1, 2, 3$ where $P^1_v = (0, 0, p^1_v, q^1_v, r^1_v, s^1_v)$, $P^2_s = (p^2_v, q^2_v, 0, 0, r^2_v, s^2_v)$, $P^3_v = (p^3_v, q^3_v, r^3_v, s^3_v, 0, 0)$.

Utilising $e_i \cap \mathcal{S}^2_{\vec{P}^1_s} = \emptyset, i = 1, 2$, $e_i \cap \mathcal{S}^2_{\vec{P}^1_s} = \emptyset, i = 3, 4$, $e_i \cap \mathcal{S}^3_{\vec{P}^1_s} = \emptyset, i = 5, 6$ and $z_a \cap \mathcal{S}^I_{\vec{P}^1_s} = \emptyset, a = 1, 2, I = 1, 2, 3$ the spinorial projectors can be recast in the form

$$\Delta^I U^I_s = Y^I_s \, , \quad I = 1, 2, 3 \, , \quad \text{(A2)}$$

where

$$\Delta^1 = \begin{pmatrix} (e_1|e_3) & (e_1|e_4) & (e_1|e_5) & (e_1|e_6) \\ (e_2|e_3) & (e_2|e_4) & (e_2|e_5) & (e_2|e_6) \\ (z_1|e_3) & (z_1|e_4) & (z_1|e_5) & (z_1|e_6) \\ (z_2|e_3) & (z_2|e_4) & (z_2|e_5) & (z_2|e_6) \end{pmatrix}, \quad \text{(A3)}$$

$$\Delta^2 = \begin{pmatrix} (e_3|e_1) & (e_3|e_2) & (e_3|e_5) & (e_3|e_6) \\ (e_4|e_1) & (e_4|e_2) & (e_4|e_5) & (e_4|e_6) \\ (z_1|e_1) & (z_1|e_2) & (z_1|e_5) & (z_1|e_6) \\ (z_2|e_1) & (z_2|e_2) & (z_2|e_5) & (z_2|e_6) \end{pmatrix}, \quad \text{(A4)}$$

$$\Delta^3 = \begin{pmatrix} (e_5|e_1) & (e_5|e_2) & (e_5|e_3) & (e_5|e_4) \\ (e_6|e_1) & (e_6|e_2) & (e_6|e_3) & (e_6|e_4) \\ (z_1|e_1) & (z_1|e_2) & (z_1|e_3) & (z_1|e_4) \\ (z_2|e_1) & (z_2|e_2) & (z_2|e_3) & (z_2|e_4) \end{pmatrix}, \quad \text{(A5)}$$

and

$$U^1_s = \begin{pmatrix} p^1_s \\ q^1_s \\ r^1_s \\ s^1_s \end{pmatrix} \, , \; U^2_s = \begin{pmatrix} p^2_s \\ q^2_s \\ r^2_s \\ s^2_s \end{pmatrix} \, , \; U^3_s = \begin{pmatrix} p^3_s \\ q^3_s \\ r^3_s \\ s^3_s \end{pmatrix} \quad \text{(A6)}$$

and

$$Y^1_s = \begin{pmatrix} (e_1|b_1) \\ (e_2|b_1) \\ (z_1|b_1) \\ (z_2|b_1) \end{pmatrix} \, , Y^2_s = \begin{pmatrix} (e_3|b_2) \\ (e_4|b_2) \\ (z_1|b_2) \\ (z_2|b_2) \end{pmatrix} \, , Y^3_s = \begin{pmatrix} (e_5|b_3) \\ (e_6|b_3) \\ (z_1|b_3) \\ (z_2|b_3) \end{pmatrix}$$
$$\text{(A7)}$$

For the surviving spinorials we calculate chiralities using the formulae [38] , $X^{(I)}_{pqrs} = \exp(i\pi \chi^{(I)}_{pqrs})$, with

$$\chi^{(1)}_{pqrs} = \alpha_0 + (1-r)(e_5|b_1) + (1-s)(e_6|b_1)$$
$$+ p(e_3|b_2) + q(e_4|b_2) + r(e_5|b_2) + s(e_6|b_2)$$
$$+ p(1-r)(e_3|e_5) + p(1-s)(e_3|e_6)$$
$$+ q(1-r)(e_4|e_5) + q(1-s)(e_4|e_6)$$
$$+ (r+s)(e_5|e_6) \, , \quad \text{(A8)}$$

$$\chi^{(2)}_{pqrs} = \alpha_0 + (1-r)(e_5|b_2) + (1-s)(e_6|b_2)$$
$$+ p(e_1|b_1) + q(e_2|b_1) + r(e_5|b_1) + s(e_6|b_1)$$
$$+ p(1-r)(e_1|e_5) + q(1-r)(e_2|e_5)$$
$$+ p(1-s)(e_1|e_6) + q(1-s)(e_2|e_6)$$
$$+ (r+s)(e_5|e_6) \, , \quad \text{(A9)}$$

$$\chi^{(3)}_{pqrs} = \alpha_0 + (1-p)(e_1|b_1) + (1-q)(e_2|b_1)$$
$$+ (1-r)(e_3|b_2) + (1-s)(e_4|b_2)$$
$$+ (1-r)(1-p)(e_1|e_3) + (1-r)(1-q)(e_2|e_3)$$
$$+ (1-s)(1-p)(e_1|e_4) + (1-s)(1-q)(e_2|e_4)$$
$$+ (1-r)[(e_3|e_5) + (e_3|e_6)] + (1-s)[(e_4|e_5) + (e_4|e_6)]$$
$$+ (1-r)[(e_3|z_1) + (e_3|z_2)] + (1-s)[(e_4|z_1) + (e_4|z_2)]$$
$$+ (e_5|b_1) + (e_6|b_1) + (z_1|b_1) + (z_2|b_1) \, , \quad \text{(A10)}$$

where we can set $\alpha_0 = 0$ as it depends on conventions, $e^{i\pi\alpha_0} = -\text{ch}(\psi^\mu) c\begin{bmatrix}\mathbb{1}\\S\end{bmatrix} c\begin{bmatrix}S\\b_1\end{bmatrix} c\begin{bmatrix}S\\b_2\end{bmatrix} c\begin{bmatrix}b_1\\b_2\end{bmatrix}$.

Similarly, vectorial projectors can be recast in the form

$$\Delta^I U^I_v = Y^I_v \, , \quad I = 1, 2, 3 \, , \quad \text{(A11)}$$

where

$$U^1_v = \begin{pmatrix} p^1_v \\ q^1_v \\ r^1_v \\ s^1_v \end{pmatrix} \, , \; U^2_v = \begin{pmatrix} p^2_v \\ q^2_v \\ r^2_v \\ s^2_v \end{pmatrix} \, , \; U^3_v = \begin{pmatrix} p^3_v \\ q^3_v \\ r^3_v \\ s^3_v \end{pmatrix}, \quad \text{(A12)}$$

and

$$Y^1_v = \begin{pmatrix} (e_1|b^x_1) \\ (e_2|b^x_1) \\ (z_1|b^x_1) \\ (z_2|b^x_1) \end{pmatrix} \, , Y^2_v = \begin{pmatrix} (e_3|b^x_2) \\ (e_4|b^x_2) \\ (z_1|b^x_2) \\ (z_2|b^x_2) \end{pmatrix} \, , Y^3_v = \begin{pmatrix} (e_5|b^x_3) \\ (e_6|b^x_3) \\ (z_1|b^x_3) \\ (z_2|b^x_3) \end{pmatrix} \, ,$$
$$\text{(A13)}$$

with $b_i^x = b_i + x$. The GGSO associated coefficients in $Y_s^3, Y_v^I, i = 1, 2, 3$ can be reduced as follows

$$(e_i|b_3) = (e_i|b_1) + (e_3|b_2) + (e_i|x) \qquad (A14)$$

$$(z_a|b_3) = (z_a|b_1) + (z_a|b_2) + (z_a|x) \qquad (A15)$$

$$(e_i|b_I + x) = (e_i|b_I) + (e_i|x), I = 1, 2 \qquad (A16)$$

$$(z_a|b_I + x) = (z_a|b_I) + (z_a|x), I = 1, 2 \qquad (A17)$$

$$(e_i|b_3 + x) = (e_i|b_1) + (e_i|b_2), \qquad (A18)$$

$$(z_a|b_3 + x) = (z_a|b_1) + (z_a|b_2), \qquad (A19)$$

with

$$(e_i|x) = \sum_{\substack{j=1 \\ j \neq i}}^{6} (e_i|e_j) + (e_i|z_1) + (e_i|z_2), \qquad (A20)$$

$$(z_a|x) = 1 + \sum_{j=1}^{6} (z_a|e_j) + (z_1|z_2). \qquad (A21)$$

In the notation employed here

$$c \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = e^{i\pi(\alpha|\beta)}. \qquad (A22)$$

Moreover,

$$(e_i|e_j) = (e_j|e_i), \ (e_i|z_a) = (z_a|e_i), \qquad (A23)$$

$$(z_1|z_2) = (z_2|z_1), \ (e_i|b_k) = (b_k|e_i). \qquad (A24)$$

For a given set of spin structure coefficients $c \begin{bmatrix} \beta_i \\ \beta_j \end{bmatrix}$ the solutions $U_s^I, U_v^I, I = 1, 2, 3$ of (A2), (A11) determine the

number of surviving spinorials/vectorials. Without loss of generality, we can assume that one spinorial comes from the $S + b_1$ sector, that is we have a solution with $p_s^1 = q_s^1 = r_s^1 = s_s^1 = 0$. This amounts to setting

$$(e_1|b_1) = (e_2|b_1) = (z_1|b_1) = (z_2|b_1) = 0 \qquad (A25)$$

as dictated by the relevant equation of (A2), i.e $\Delta^1 U_s^1 = Y_s^1$. Furthermore, we can also assume that the second spinorial arises from $S + b_2$, i.e. that $p_s^2 = q_s^2 = r_s^2 = s_s^2 = 0$ is a solution of $\Delta^2 U_s^2 = Y_s^2$, which then implies

$$(e_3|b_2) = (e_4|b_2) = (z_1|b_2) = (z_2|b_2) = 0. \qquad (A26)$$

The existence of a coupling of the form $\mathbf{16} \times \mathbf{16} \times \mathbf{10}$ at the trilinear effective superpotential (top mass Yukawa coupling) requires at least one vectorial coming from $S + b_1 + b_2 + x$, that is Eq. (A11) has a solution with $p_v^3 = q_v^3 = r_v^3 = s_v^3 = 0$ [45]. Consequently, we also set

$$(e_5|b_1) = (e_5|b_2), \ (e_6|b_1) = (e_6|b_2), \qquad (A27)$$

$$(z_1|b_1) = (z_1|b_2), \ (z_2|b_1) = (z_2|b_2). \qquad (A28)$$

Finally additional constraints come from adjusting spinorial chiralities, in order to satisfy the chirality constraints on the first and second plane – i.e. to solve the equivalent of Eq. (9) for Eqs. (A8) and (A9). These conditions are

$$(e_5|b_1) = (e_6|b_1), \ (e_5|b_2) = (e_6|b_2). \qquad (A29)$$

[1] M. R. Douglas, "The Statistics of string / M theory vacua," JHEP **05** (2003) 046, [arXiv:hep-th/0303194].

[2] S. Ashok and M. R. Douglas, "Counting flux vacua," JHEP **01** (2004) 060, [arXiv:hep-th/0307049 [hep-th]].

[3] W. Taylor and Y.-N. Wang, "The F-theory geometry with most flux vacua," JHEP **12** (2015) 164, [arXiv:1511.03209 [hep-th]].

[4] A. Constantin, Y.-H. He, and A. Lukas, "Counting String Theory Standard Models," Phys. Lett. B **792** (2019) 258–262, [arXiv:1810.00444 [hep-th]].

[5] J. H. Holland, *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, MI, USA, 1975.

[6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley Professional, Boston, MA, USA, 1989.

[7] J. Blåbäck, U. Danielsson, and G. Dibitetto, "Fully stable dS vacua from generalised fluxes," JHEP **08** (2013) 054, [arXiv:1301.7073 [hep-th]].

[8] S. Abel and J. Rizos, "Genetic Algorithms and the Search for Viable String Vacua," JHEP **08** (2014) 010, [arXiv:1404.7359 [hep-th]].

[9] F. Ruehle, "Evolving neural networks with genetic algorithms to study the String Landscape," JHEP **08** (2017) 038, [arXiv:1706.07024 [hep-th]].

[10] A. Cole, A. Schachner, and G. Shiu, "Searching the Landscape of Flux Vacua with Genetic Algorithms," JHEP **11** (2019) 045, [arXiv:1907.10072 [hep-th]].

[11] S. AbdusSalam, S. Abel, M. Cicoli, F. Quevedo, and P. Shukla, "A systematic approach to Kähler moduli stabilisation," JHEP **08** no. 08, (2020) 047, [arXiv:2005.11329 [hep-th]].

[12] F. Ruehle, "Data science applications to string theory," Phys. Rept. **839** (2020) 1–117.

[13] I. Bena, J. Blåbäck, M. Graña, and S. Lüst, "The tadpole problem," JHEP **11** (2021) 223, [arXiv:2010.10519 [hep-th]].

[14] M. Larfors and R. Schneider, "Explore and Exploit with Heterotic Line Bundle Models," Fortsch. Phys. **68** no. 5, (2020) 2000034, [arXiv:2003.04817 [hep-th]].

[15] I. Bena, J. Blåbäck, M. Graña, and S. Lüst, "Algorithmically Solving the Tadpole Problem," Adv. Appl. Clifford Algebras **32** no. 1, (2022) 7, [arXiv:2103.03250 [hep-th]].

[16] S. Abel, A. Constantin, T. R. Harvey, and A. Lukas, "Evolving Heterotic Gauge Backgrounds: Genetic Algorithms versus Reinforcement Learning," Fortsch. Phys. **70** no. 5, (2022) 2200034, [arXiv:2110.14029

[hep-th]].

[17] A. Cole, S. Krippendorf, A. Schachner, and G. Shiu, "Probing the Structure of String Theory Vacua with Genetic Algorithms and Reinforcement Learning," in *35th Conference on Neural Information Processing Systems.* 11, 2021. [arXiv:2111.11466 [hep-th]].

[18] G. J. Loges and G. Shiu, "Breeding Realistic D-Brane Models," Fortsch. Phys. **70** no. 5, (2022) 2200038, [arXiv:2112.08391 [hep-th]].

[19] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, "Quantum computation by adiabatic evolution," arXiv preprint quant-ph/0001106 (2000) .

[20] A. Das and B. K. Chakrabarti, *Quantum Annealing and Related Optimization Methods.* Springer, Berlin, Germany, 2008.

[21] A. Mott, J. Job, J. R. Vlimant, D. Lidar, and M. Spiropulu, "Solving a Higgs optimization problem with quantum annealing for machine learning," Nature **550** no. 7676, (2017) 375–379.

[22] S. Abel, N. Chancellor, and M. Spannowsky, "Quantum Computing for Quantum Tunnelling," [arXiv:2003.07374 [hep-ph]].

[23] S. Abel and M. Spannowsky, "Observing the fate of the false vacuum with a quantum laboratory," P. R. X. Quantum. **2** (2021) 010349, [arXiv:2006.06003 [hep-th]].

[24] S. A Rahman, R. Lewis, E. Mendicelli, and S. Powell, "SU(2) lattice gauge theory on a quantum annealer," Phys. Rev. D **104** no. 3, (2021) 034501, [arXiv:2103.08661 [hep-lat]].

[25] S. Schenk and M. Spannowsky, "Exploring instantons with spin-lattice systems," Phys. Rev. B **103** no. 14, (2021) 144436, [arXiv:2012.10325 [hep-th]].

[26] S. Abel, A. Blance, and M. Spannowsky, "Quantum Optimisation of Complex Systems with a Quantum Annealer," [arXiv:2105.13945 [quant-ph]].

[27] O. Pomponio, M. A. Werner, G. Zaránd, and G. Takacs, "Bloch oscillations and the lack of the decay of the false vacuum in a one-dimensional quantum spin chain," SciPost Phys. **12** no. 2, (2022) 061, [arXiv:2105.00014 [cond-mat.stat-mech]].

[28] G. Lagnese, F. M. Surace, M. Kormos, and P. Calabrese, "False vacuum decay in quantum spin chains," Phys. Rev. B **104** no. 20, (2021) L201106, [arXiv:2107.10176 [cond-mat.stat-mech]].

[29] Y. Bando, K.-W. Yip, H. Chen, D. A. Lidar, and H. Nishimori, "Breakdown of the Weak-Coupling Limit in Quantum Annealing," Phys. Rev. Applied **17** no. 5, (2022) 054033, [arXiv:2111.07560 [quant-ph]].

[30] M. Illa and M. J. Savage, "Basic elements for simulations of standard-model physics with quantum annealers: Multigrid and clock states," Phys. Rev. A **106** no. 5, (2022) 052605, [arXiv:2202.12340 [quant-ph]].

[31] J. C. Criado, R. Kogler, and M. Spannowsky, "Quantum fitting framework applied to effective field theories," Phys. Rev. D **107** no. 1, (2023) 015023,

[arXiv:2207.10088 [hep-ph]].

[32] M. Fromm, O. Philipsen, and C. Winterowd, "Dihedral Lattice Gauge Theories on a Quantum Annealer," [arXiv:2206.14679 [hep-lat]].

[33] S. Abel, L. A. Nutricati, and M. Spannowsky, "A Genetic Quantum Annealing Algorithm," [arXiv:2209.07455 [quant-ph]].

[34] S. Abel, A. Constantin, T. R. Harvey, A. Lukas, and L. A. Nutricati, "Decoding Nature with Nature's Tools: Heterotic Line Bundle Models of Particle Physics with Genetic Algorithms and Quantum Annealing," [arXiv:2306.03147 [hep-th]].

[35] J. Kim, T. Luu, and W. Unger, "$U(N)$ gauge theory in the strong coupling limit on a quantum annealer," [arXiv:2305.18179 [hep-lat]].

[36] S. Abel and L. A. Nutricati, "Ising Machines for Diophantine Problems in Physics," [arXiv:2206.09956 [hep-th]].

[37] A. E. Faraggi, C. Kounnas, S. E. M. Nooij, and J. Rizos, "Towards the classification of Z2 x Z2 fermionic models," in *2nd String Phenomenology 2003*, pp. 143–151. 11, 2003. [arXiv:hep-th/0311058].

[38] A. E. Faraggi, C. Kounnas, S. E. M. Nooij, and J. Rizos, "Classification of the chiral Z(2) x Z(2) fermionic models in the heterotic superstring," Nucl. Phys. B **695** (2004) 41–72, [arXiv:hep-th/0403058].

[39] A. E. Faraggi, C. Kounnas, and J. Rizos, "Chiral family classification of fermionic Z(2) x Z(2) heterotic orbifold models," Phys. Lett. B **648** (2007) 84–89, [arXiv:hep-th/0606144].

[40] A. E. Faraggi, C. Kounnas, and J. Rizos, "Spinor-Vector Duality in fermionic Z(2) X Z(2) heterotic orbifold models," Nucl. Phys. B **774** (2007) 208–231, [arXiv:hep-th/0611251].

[41] A. E. Faraggi, C. Kounnas, and J. Rizos, "Spinor-vector duality in N=2 heterotic string vacua," Nucl. Phys. B **799** (2008) 19–33, [arXiv:0712.0747 [hep-th]].

[42] T. Lanting, "The D-Wave 2000Q Processor," 2017. `https://www.youtube.com/watch?v=_Y9sVY-XBfI&index=2&list=PLAlerseOylzZw8A-R4lyU3mqfYfQSggiz`. presented at AQC 2017.

[43] S. Efthymiou, S. Ramos-Calderer, C. Bravo-Prieto, A. Pérez-Salinas, D. García-Martín, A. Garcia-Saez, J. I. Latorre, and S. Carrazza, "Qibo: a framework for quantum simulation with hardware acceleration," Quantum Science and Technology **7** no. 1, (Dec, 2021) 015018. `https://doi.org/10.1088/2058-9565/ac39f5`.

[44] S. Abel, J. C. Criado, and M. Spannowsky, "Completely Quantum Neural Networks," [arXiv:2202.11727 [quant-ph]].

[45] J. Rizos, "Top quark mass coupling and classification of weakly-coupled heterotic superstring vacua," Eur. Phys. J. C **74** no. 6, (2014) 2905, [arXiv:1404.0819 [hep-ph]].