# Quantum annealing applications in high-energy phenomenology

**Juan Carlos Criado**

CAFPE and Departamento de Física Teórica y del Cosmos,
Universidad de Granada, Campus de Fuentenueva, E–18071 Granada, Spain

E-mail: jccriadoalamo@ugr.es

**Abstract.** The quantum annealing framework provides a general technique for the solution of a wide variety of optimization problems. Recent technological developments have made it possible to build quantum annealing devices with thousands of qubits, bringing them within range to tackle real-world problems in the near future. In this paper, I review this framework, and present a method for its use in the minimization of functions of continuous variables. Then, I discuss how this method can be employed to solve optimization problems with applications in high-energy physics, including training neural networks for classification, solving partial differential equations, and fitting effective field theories to experimental data.

## 1. Introduction

The development of efficient algorithms for the optimization of complex multivariate target functions is essential for theoretical and phenomenological studies in high-energy physics. Such optimization problems arise when performing several different data-analysis tasks. These include fitting the parameters of a theoretical model to experimental data, and training machine-learning models for multiple purposes, e.g., event classification. Purely theoretical calculations may also require the use of numerical optimization techniques. For example, the solution of a differential equation or variational problem can be viewed as the minimum of some loss function [1, 2].

Classical optimization algorithms typically operate by incrementally improving a tentative solution. This can lead to their convergence to local extrema of the target function, instead of the desired global one. While there are several techniques that attempt to address this issue, it remains a problem when local extrema are deep, in the sense that they are separated from the global one by large barriers. Quantum annealing provides an alternative optimization method, based on the dynamics of a quantum system [3, 4]. It may offer a better performance than its classical counterparts, both in terms of the scaling of the time required to find the solution [5], and in the success rate for obtaining the global extremum [6].

The existing implementations of quantum annealing in physical devices allow to find the ground state of a generic Ising model with thousands of qubits. In order to minimize a more complex function, one needs to find a qubit encoding of its arguments that will turn the function into an Ising model Hamiltonian when expressed in terms of the qubits. In this work, I describe several techniques to achieve this for some of the optimization problems arising in high-energy physics [7, 8, 9].

## 2. Quantum Annealing

The quantum annealing method allows to obtain the ground state of a quantum Hamiltonian $H_1$. This is done by means of a physical system whose Hamiltonian is given by

$$H(s) = A(s)H_0 + B(s)H_1, \tag{1}$$

where $H_0$ is a simple Hamiltonian whose ground state is known, $A(s)$ and $B(s)$ are functions such that $A(0) = B(1) = 0 < A(1), B(0)$, and $s$ is an adjustable parameter. The annealing procedure starts by setting $s = 0$ and preparing the system in the ground state of $H_0$. Then, $s$ is turned up from 0 to 1. If this is done sufficiently slowly, the adiabatic theorem ensures that the system ends up in the ground state of $H_1$ with a high probability.

A quantum annealing device performs this procedure on a fixed system with pre-defined $H_0$, $A(s)$ and $B(s)$, allowing the user to control the time evolution of $s$ and the free parameters of the target Hamiltonian $H_1$. The typical system employed in such devices is a collection of qubits (i.e., two-state quantum systems). In particular, the largest currently-available quantum annealers implement the transverse-field Ising model, which is defined as

$$H_0 = \sum_i X_i, \qquad H_1 = \sum_{ij} J_{ij} Z_i Z_j + \sum_i h_i Z_i, \tag{2}$$

where $X_i$ and $Z_i$ are the corresponding Pauli matrices acting on the $i$-th qubit, and $J_{ij}$ and $\theta_i$ are free parameters.

The eigenstates of the quantum Ising model Hamiltonian $H_1$ are simultaneous eigenstates of all the $Z_i$. They are thus labeled by the eigenvalues $\sigma_i = \pm 1$ of the $Z_i$. That is, collecting the eigenvalues in a vector $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots)$, one has $Z_i |\boldsymbol{\sigma}\rangle = \sigma_i |\boldsymbol{\sigma}\rangle$. There is thus a direct correspondence between the quantum Ising model and the discrete classical system consisting of two-state spins $\sigma_i = \pm 1$ with Hamiltonian: $H_{\text{classical}}(\boldsymbol{\sigma}) = \sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_i h_i\sigma_i$, which is just the classical Ising model. Concretely, the eigenvalues of the quantum Hamiltonian $H_1$ are equal to the energies of this classical system: $\langle\boldsymbol{\sigma}| H_1 |\boldsymbol{\sigma}\rangle = H_{\text{classical}}(\boldsymbol{\sigma})$. One can thus view the quantum annealing process for the transverse-field Ising model as a method for minimizing the discrete classical function $H_{\text{classical}}(\boldsymbol{\sigma})$ through quantum dynamics. Alternatively, it is sometimes useful to view it as the minimization of the function

$$\mathrm{Q}(\boldsymbol{\tau}) = \sum_{ij} Q_{ij}\tau_i\tau_j, \tag{3}$$

where $\tau_i = (\sigma_i + 1)/2 \in \{0, 1\}$. The minimization problem for $\mathrm{Q}(\boldsymbol{\tau})$ is known as Quadratic Unconstrained Binary Optimization (QUBO) [10, 11, 12].

In order to solve a general optimization problem in a quantum annealer, one needs to find a way of encoding it as the minimization of a function of the form given in Eq. (2) or Eq. (3). It has been shown that, for several such encodings, quantum annealing is more consistent than its classical alternatives in finding the global minimum of some non-convex functions [6, 7]. In Sec. 3, I will present a method for obtaining encodings for a wide range of optimization problems.

The practical limitations of quantum annealing come from the number of available qubits, their connectivity, and the finite amount of time that can be used in an annealing run. The state-of-the-art D-Wave `Advantage_system6.1` device contains 5616 qubits with a total of 40135 couplings between them. Each qubit is connected to 15 other qubits. This means that only some of the elements of the $J_{ij}$ matrix can be non-vanishing. If the $J_{ij}$ matrix corresponding to the problem at hand is denser than what is allowed by the physical device, several qubits are be chained together by setting a strong $J_{ij}$ connection between them. The chain of strongly bound qubits behaves as a single one with more available connections. In this way, highly-connected Ising models can be embedded in a physical device, at the price of using more qubits.

The annealing time plays a role in the likelihood to get the ground state after the annealing process. The adiabatic theorem is valid only in the limit in which the annealing time goes to infinity. In practical applications with finite time, this is accounted for by running the annealing process several times. The number of times this is done is called the *number of reads.* The final candidate for the ground state is selected as the final state with the least energy among all those that have been generated.

## 3. Encoding

One of the key elements for the implementation of an optimization problem in a quantum annealer is its formulation as a QUBO (or, equivalently, the minimization of a classical Ising model Hamiltonian). Here, I present a method that allows to do this for a large class of problems, consisting of the minimization of a polynomial function $L(\mathbf{c})$ of real variables $\mathbf{c} = (c_1, \ldots, c_n)$ inside a box

$$B = [a_1, b_1] \times \ldots \times [a_n, b_n], \tag{4}$$

up to a given precision $(b_i - a_i)\,\epsilon$ for each $c_i$. The function $L$ to be minimized is referred to as the *loss function.* It should be noted that, since polynomials can efficiently approximate a wide variety of functions on a bounded domain, this method effectively allows to encode many non-polynomial functions too.

The first step is to write $L$ as a function of $n \times (p+1)$ binary variables $\tau_{i\alpha} = 0, 1$, indexed by $i = 1, \ldots, n$ and $\alpha = 0, \ldots p$, where $p$ is such that $1/(2^p - 1) < \epsilon$. The real variables $c_i$ can be expressed as functions of the binary ones $\tau_{i\alpha}$, as

$$c_i(\tau) = a_i + \frac{b_i - a_i}{1 - 2^{-p}} \sum_{\alpha=0}^{p} \frac{\tau_{i\alpha}}{2^p}, \tag{5}$$

Substituting this expansion in the quadratic loss function turns it into a polynomial $L(\mathbf{c}(\tau))$ in the QUBO variables $\tau_{i\alpha}$.

The second step is to turn this polynomial into a quadratic one, at the price of introducing auxiliary binary variables $\tilde{\tau}$. This can be done by iteratively applying the following procedure:

(i) Select a monomial $\tau_1\tau_2\tau_3\ldots$ in the loss function polynomial with degree higher than 2.

(ii) Replace the product $\tau_1\tau_2$ by a new binary variables $\tilde{\tau}$ in this monomial. This decreases its degree by one unit.

(iii) Add a term $\lambda C(\tau_1, \tau_2, \tilde{\tau})$ to the loss function, where $\lambda$ is a large coefficient, and $C$ is the quadratic function:

$$C(x, y, z) = xy - 2z(x + y) + 3z. \tag{6}$$

The values of the original $\tau$ variables at the global minimum of the loss function are unchanged by this procedure. This is because $C(x, y, z)$ attains its minimal value if and only if $xy = z$. Every time the procedure is applied, the degree of one monomial is decreased. The process stops when the loss becomes quadratic.

Through the application of these two steps, binary encoding and quadratization, one can always turn the minimization of a polynomial in continuous variables on a bounded domain into a QUBO. In practice, the number of variables $c_i$ and the precision $\epsilon \gtrsim 1/(2^p - 1)$ that can be achieved are limited by the number of available qubits. The precision can be improved through what is known as a *zooming* procedure [13]. Once the optimal $\mathbf{c}$ has been obtained, the boundaries $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_n)$ can be updated to center the $n$-dimensional box $B$ around $\mathbf{c}$ and reduce its size by a factor $0 < f < 1$. Each zooming step is known as an *epoch*, and $f$ is the *zooming factor.* Denoting the boundaries of epoch $N$ by $\mathbf{a}_N$ and $\mathbf{b}_N$, and the solution $\mathbf{c}$ resulting from that epoch by $\mathbf{c}_N$, the update rule is

$$\mathbf{a}_{N+1} = (1 - f)\,\mathbf{a}_N + f\,\mathbf{c}_N, \qquad \mathbf{b}_{N+1} = (1 - f)\,\mathbf{b}_N + f\,\mathbf{c}_N, \tag{7}$$

The zooming procedure allows to obtain a greater precision with a relatively low number of qubits, at the price of intrucing classical updating steps. Some of the advantage of the quantum computation may be lost, if the first zooming steps, having low precision, generate wrong guesses for the $c_i$ that are far away from the optimal values for a higher precision. This can be mitigated by choosing a zooming factor close to one, e.g. $f = 0.9$, which allows successive epochs to correct the potential wrong values produced by previous ones. In an ideal setting in which sufficient qubits are available to reach the desired precision, this issue can be completely removed by performing a single epoch, taking full advantage of the quantum dynamics of the annealer. This may be realized in future annealing devices, for which the number of qubits and connections is expected to grow quickly.

## 4. Applications

This section focuses on three specific applications of the general encoding method outlined in Sec. 3, to three different problems: training neural networks, solving differential equations, and fitting effective field theories to experimental data. I will show examples of such problems in which the QUBO obtained with this procedure have been embedded solved in quantum annealers. Generally, these are small proof-of-concept examples that allow to test the viability of the proposed methodology, and suggest that it has the potential to solve real-world problems when larger quantum annealing devices with more qubits and couplings are available.

### *4.1. Training neural networks*

A neural network is a family of non-linear functions $\mathbf{Y}_{\boldsymbol{\theta}}(\mathbf{x})$ that depend on a collection of parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots)$. In a wide range of applications, they are able to efficiently approximate the functional relation described by a set of points $(\mathbf{x}_a, \mathbf{y}_a)$, as:

$$\mathbf{y}_a \simeq \mathbf{Y}_{\boldsymbol{\theta}}(\mathbf{x}_a) \tag{8}$$

by adjusting a relatively small number of free parameters $\theta_i$. In their simplest version, known as densely connected feed-forward networks, the $\mathbf{Y}$ functions are defined by successive application of layers, with each layer consisting of an affine transformation $A_{ij}x_j + B_i$, followed by the element-wise application of a non-linear function $f$, known as the activation function. For example, the action of 2-layer network is given by

$$Y_{\boldsymbol{\theta},i}(\mathbf{x}) = \sum_j f\left[A^{(2)}_{ij} f\left(\sum_k A^{(1)}_{jk} x_k + B^{(1)}_j\right) + B^{(2)}_i\right]. \tag{9}$$

In general, the elements of the $A$ matrices and $B$ vectors for all layers constitute the collection $\boldsymbol{\theta}$ of free parameters.

The process by which the parameters $\boldsymbol{\theta}$ are adjusted is known as the training algorithm. This is typically done by minimizing a loss function that quantifies the deviation from Eq. (8). A common loss function in several applications is the mean squared error:

$$L(\boldsymbol{\theta}) = \sum_a |\mathbf{Y}_{\boldsymbol{\theta}}(\mathbf{x}_a) - \mathbf{y}_a|^2 . \tag{10}$$

Training is usually the most computationally expensive task in machine learning, and the loss functions that arise in this context are highly non-convex, leading to difficulties in their minimization through classical algorithms. Using the method presented here, neural networks can be trained on quantum annealers, which allows to take advantage of the tunneling that happens in them to find the global minimum of the loss function more efficiently.

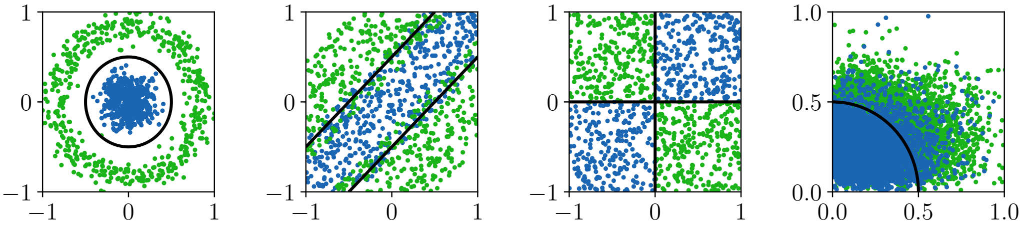

**Figure 1.** Example datasets (dots) and decision boundaries (lines) from simple neural networks trained on a quantum annealer. Points with label $y_a = -1$ ($+1$) are shown in blue (green)

In order to formulate the training as a QUBO, a bounded domain must be chosen for the parameters $\boldsymbol{\theta}$. Then, the activation function will only be applied to a bounded set of values, arising from combinations of $\boldsymbol{\theta}$ and $\mathbf{x}_a$, so they can be approximated arbitrarily well by a (single-variable) polynomial. The same argument applies to the loss, viewed as a function of $\mathbf{Y}_{\boldsymbol{\theta}}(\mathbf{x}_a)$ and $\mathbf{y}_a$. In the case of Eq. (10), this is not necessary, as it is already quadratic. After these approximations have been made, the loss function becomes a polynomial in $\boldsymbol{\theta}$. Then, the procedure in Sec. 3 can be applied to convert it into QUBO form.

In Ref. [7], a simple version of the network defined in Eq. (9), without the final activation function and the first layer's biases $B_j^{(1)}$, and with one qubit per parameter, was trained using the method presented here to fit three toy datasets in which the $\mathbf{x}_a$ vectors are two-dimensional and $y_a = \pm 1$. The resulting decision boundaries, computed as the set of points $\mathbf{x}$ for which $\mathbf{Y}_{\boldsymbol{\theta}}(\mathbf{x}) = 1/2$ are shown in Fig. 1. The quantum training algorithm was compared to a classical analog of the same network, with binarized weights. The quantum one was shown to be more reliable in finding the global minimum of the loss function.

### *4.2. Solving differential equations*

An arbitrary system of partial differential equations with any initial or boundary conditions can be equivalently formulated as a QUBO (up to finite precision), as long as the equations and conditions are polynomials in the functions $f_i(\mathbf{x})$ to be solved for and their derivatives. In order to achieve this, the first step is to parametrize the solution as a linear combination

$$f_i(\mathbf{x}) = \sum_a w_{ia} \Phi_a(\mathbf{x}) \tag{11}$$

of some set of basis functions $\Phi_a(\mathbf{x})$. The functions $\Phi_a(\mathbf{x})$ should be chosen taking into account the type of problem under consideration, e.g., using a set of sines and cosines for oscillatory problems.

The second step is to define a loss function $L(w)$ such that its global minimum is achieved at values of the $w_{ia}$ that make $f_i(\mathbf{x})$ an approximate solution of the system. I denote both the differential equations and the boundary conditions at a given point $\mathbf{x}$ as $E_a(\mathbf{x})[f] = 0$, with the understanding that boundary conditions identically vanish outside the region on which they are applied. That is, $E_a(\mathbf{x})[f] = 0$ if $\mathbf{x}$ is outside the domain of $E_a$. The loss function can then be defined as

$$L = \sum_{ak} \left(E_a(\mathbf{x}_k)[f]\right)^2 , \tag{12}$$

where the $\mathbf{x}_k$ denote the points of a lattice that fills the domain of the equations. $L$ is a non-negative quantity that vanishes when all equations are satisfied. It thus achieve its global minimum at values of $w$ that solve them. Plugging the expansion from Eq. (11) into Eg. (12), gives the loss function as a polynomial in the parameters $w_{ia}$. The method in Sec. 3 can then be applied, providing a QUBO representation for the problem.

In Ref. [9], this technique has been used to solve several simple examples of differential equations (ordinary, partial and coupled) in a quantum annealing device. An accurate approximation to the analytical solution is found reliably with it. While the quantum method cannot currently compete with specialized classical solvers, due to limitations in the number of qubits, these results suggest that it may become useful in practice when larger annealing devices are avaiable.

### *4.3. Fitting effective field theories to experimental data*

The task of finding the values of free parameters $\boldsymbol{\theta}$ of a theory that describe the experimentally measured data best can also be seen as an optimization problem. Assuming a Gaussian likelihood, the function to be minimized is the negative log-likelihood

$$\chi^2 = \left(\mathcal{O}_a^{\text{exp}} - \mathcal{O}_a^{\text{th}}(\boldsymbol{\theta})\right) C_{ab}^{-1} \left(\mathcal{O}_b^{\text{exp}} - \mathcal{O}_b^{\text{th}}(\boldsymbol{\theta})\right), \tag{13}$$

where the $\mathcal{O}_a^{\text{exp}}$ are the experimental values of the observables under consideration, the $\mathcal{O}_a^{\text{th}}(\boldsymbol{\theta})$ are the $\boldsymbol{\theta}$-dependent theoretical predictions for their values, and $C_{ij}^{-1}$ is the inverse covariance matrix.

In an effective field theory, all quantities are expanded perturbatively in inverse powers of the cutoff scale $1/\Lambda$. The free parameters are the Wilson coefficients, which have an associated power of $1/\Lambda$. In practice, when working at a finite precision, the pertubative expansion can be cut at a finite order. Then, the loss function $\chi^2$ becomes polynomial in the Wilson coefficients. Applying the procedure described in Sec. 3, with bounds given by the range of values of coefficients in the pertubative regime, one obtains a QUBO that approximates the minimization problem for $\chi^2$.

Several effective field theory fits were performed in Ref. [8] using this method to compute them in quantum annealers. While the number of coefficients that can be fitted is limited by the amount of qubits (to ∼8–10 coefficients on current devices), the number of observables that can be included is arbitrary. This allowed to perform a quantum annealing fit for a set dimension-6 Wilson coefficients to a set of Higgs observables, matching the results of previous classical calculations [14]. A scenario in which an experimental anomaly generates a local minimum in the $\chi^2$ function away from the global one was considered. The quantum method was shown to be more consistent than its classical alternatives in finding the global minimum in that setup.

## References

[1] Piscopo M L, Spannowsky M and Waite P 2019 *Phys. Rev. D* **100** 016002 (*Preprint* 1902.05563)

[2] Araz J Y, Criado J C and Spannowsky M 2021 (*Preprint* 2103.14575)

[3] Finilla A B, Gomez M A, Sebenik C and Doll J D 1994 *Chem. Phys. Lett.* **219** 343

[4] Albash T and Lidar D A 2018 *Rev. Mod. Phys.* **90**(1) 015002 URL https://link.aps.org/doi/10.1103/RevModPhys.90.015002

[5] King A D *et al.* 2021 *Nature Communications* **12** 1113 URL https://doi.org/10.1038/s41467-021-20901-5

[6] Abel S, Blance A and Spannowsky M 2022 *Phys. Rev. A* **106** 042607 (*Preprint* 2105.13945)

[7] Abel S, Criado J C and Spannowsky M 2022 *Phys. Rev. A* **106** 022601 (*Preprint* 2202.11727)

[8] Criado J C, Kogler R and Spannowsky M 2023 *Phys. Rev. D* **107** 015023 (*Preprint* 2207.10088)

[9] Criado J C and Spannowsky M 2023 *Quantum Sci. Technol.* **8** 015021 (*Preprint* 2204.03657)

[10] Anthony M, Boros E, Crama Y and Gruber A 2017 *Mathematical Programming* **162** 115–144

[11] Hahn G and Djidjev H 2017 *2017 IEEE International Conference on Rebooting Computing (ICRC)* (IEEE) pp 1–8

[12] Date P, Patton R, Schuman C and Potok T 2019 *Quantum Information Processing* **18** 1–31

[13] Zlokapa A, Mott A, Job J, Vlimant J R, Lidar D and Spiropulu M 2020 *Phys. Rev. A* **102** 062405 (*Preprint* 1908.04480)

[14] Englert C, Kogler R, Schulz H and Spannowsky M 2017 *Eur. Phys. J. C* **77** 789 (*Preprint* 1708.06355)