



# Application of quantum algorithms for event reconstruction in collider experiments

Martin Bureš

Meshcheryakov Laboratory of Information Technologies (MLIT)  
Joint Institute for Nuclear Research (JINR), Dubna, Russia

December 3, 2024

New methods for processing data of a physical experiment  
Moscow Institute of Physics and Technology

## Goals:

- 1 To develop an efficient tracking algorithm for the NICA SPD experiment.
- 2 To assess the feasibility to use quantum algorithms (in simulation mode) for this purpose.

## Method:

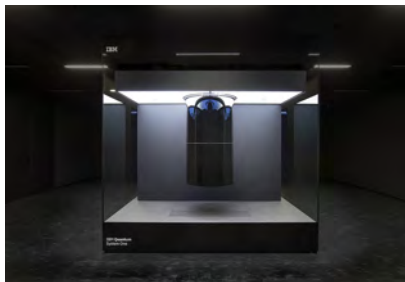
- The tracking problem is expressed as quadratic unconstrained binary optimization (QUBO) and solved by two QUBO solvers.

# NICA - Nuclotron-based Ion Collider fAcility

A new accelerator complex designed at the Joint Institute for Nuclear Research (Dubna, Russia) to study properties of dense baryonic matter.



# Universal (gate-based) vs. adiabatic quantum computer



## Universal quantum computer

(unitary quantum operations,  
measurements)  
general purpose

computer	qubits
IBM Q System One (2019)	20
IBM Osprey (2022)	433



## Adiabatic quantum computer

adiabatic evolution of quantum state  
optimization problems

D-Wave machine	qubits	couplers
2000Q (2017)	2000	6000
Advantage (2020)	5000	35000

# Quantum algorithms for tracking (QUBO)

## Quantum Annealing (QA):

- [Lucy Linder](#). *Using a Quantum Annealer for particle tracking at LHC*. 2019 - The HEPQPR.Qallse project encodes the HEP (ATLAS) pattern recognition problem into a QUBO and solves it using a D-Wave or other classical QUBO libraries (qbsolv, neal).
- [Frédéric Bapst et al.](#) "A pattern recognition algorithm for quantum annealers". In: *Comput. Softw. Big Sci.* 4.1 (2020)
- [Parker S Reid](#). "Applied Quantum Annealing for Particle Tracking: Optimisation for the HL-LHC". MA thesis. Simon Fraser University, 2021
- [Alexander Zlokapa et al.](#) "Charged particle tracking with quantum annealing optimization". In: *Quantum Machine Intelligence* 3.2 (Nov. 2021)

## Variational Quantum Eigensolver (VQE):

- [Tim Schwägerl et al.](#) *Particle track reconstruction with noisy intermediate-scale quantum computers*. 2023

## Quantum Approximate Optimization Algorithm (QAOA):

- [Arianna Crippa et al.](#) *Quantum algorithms for charged particle track reconstruction in the LUXE experiment*. 2023
- [Lena Funcke et al.](#) "Studying quantum algorithms for particle track reconstruction in the LUXE experiment". In: *Journal of Physics: Conference Series* 2438.1 (Feb. 2023), p. 012127

# Tracking with Quantum Annealing

- Track reconstruction as a quadratic unconstrained binary optimization (QUBO) – can be naturally mapped to a quantum annealer (D-Wave).
- For now, simulation.
-

# Track reconstruction as doublet classification

- 1 **doublets generation:** Generate the set of all potential doublets.
- 2 **doublets classification:** Determine the subset which is believed to contain the constituents of true tracks.



Fig. 1. Display of all generated lines for a real  $Z^0 \rightarrow$  hadrons ( $XY$  projection).

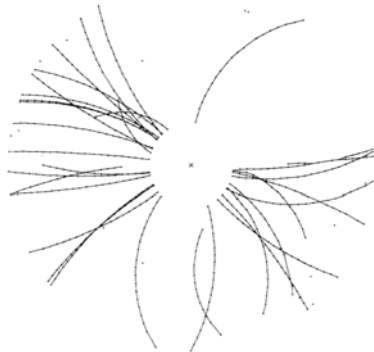


Fig. 3. Display of the activated lines after convergence for a real  $Z^0 \rightarrow$  hadrons ( $XY$  projection).

(From Georg Stimpfl-Abele and Lluís Garrido. “Fast track finding with neural networks”.  
In: *Computer Physics Communications* 64.1 (Apr. 1991))

# Track reconstruction as doublet classification

A binary classification problem: for each potential doublet, assign a 1 if kept and 0 if ignored in the final solution.

- 1 Construct **all possible segments** (doublets) of hits (in consecutive modules of the detector).

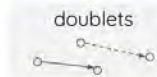
- 2 Associate each doublet with a **binary variable**:

$$t_i = \begin{cases} 1 & \text{if the doublet is part of a track,} \\ 0 & \text{otherwise.} \end{cases}$$

- 3 **Determine vector**  $\mathbf{t} = (t_1, t_2, \dots, t_N)$ , containing the correct **activation state** of every doublet.

# Model building: using triplets instead of doublets

doublets model (Stimpfl)



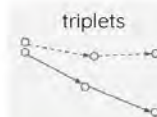
combine into



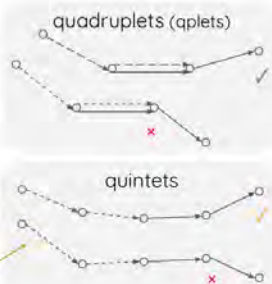
conflicts



triplets model  
(HEP.QPR)



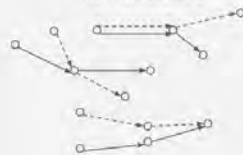
combine into



*should be used in the model,  
but are currently ignored*

conflicts

*non-exhaustive*

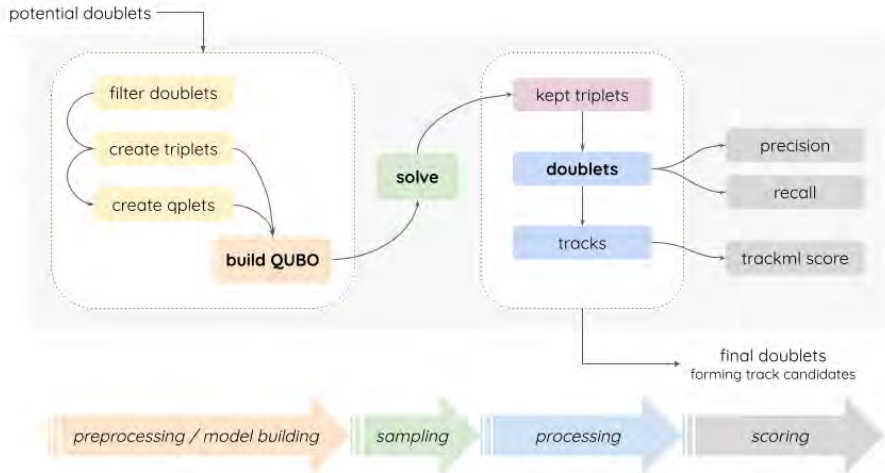


*2+ doublets in conflict*

✓ valid, generated by the model  
✗ invalid, not generated

Lucy Linder. *Using a Quantum Annealer for particle tracking at LHC*. 2019  
Georg Stimpfl-Abele and Lluís Garrido. "Fast track finding with neural networks". In:  
*Computer Physics Communications* 64.1 (Apr. 1991) (doublet model)

# Algorithm overview – triplet model



Lucy Linder. *Using a Quantum Annealer for particle tracking at LHC.* 2019

# Track reconstruction problem as QUBO

- A **triplet**,  $T_i = (h_{i_1}, h_{i_2}, h_{i_3})$ ,  $i = 1, \dots, N$  is a set of three hits,  $i_1, i_2, i_3 \in \{1, \dots, N_{\text{hit}}\}$ .
- QUBO objective function (to minimize) is constructed so that it activates the best of triplets.

$$O(a, b, t) = \sum_{i=1}^N a_i t_i + \sum_i^N \sum_{j<i}^N b_{ij} t_i t_j, \quad t_i, t_j \in \{0, 1\}.$$

- $t_i = \text{act}(T_i)$  is the  $T_i$  triplet activation.
- Linear term: *bias*  $a_i$  – quality of individual triplets (for now,  $a_i = 0$ , for  $i = 1, \dots, N$ ).
- Quadratic term: *coupling strengths*  $b_{ij}$  – computed from the relation between  $T_i$  and  $T_j$ .

$$b_{ij} = \begin{cases} -S(T_i, T_j), & \text{if } (T_i, T_j) \text{ forms a valid quadruplet,} \\ C & \text{if } (T_i, T_j) \text{ is in conflict,} \\ 0 & \text{otherwise.} \end{cases}$$

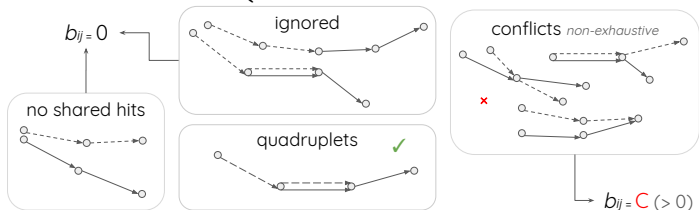
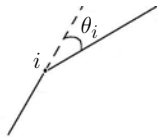


Figure from:  
[Frédéric Bapst et al.](#) "A pattern recognition algorithm for quantum annealers". In: *Comput. Softw.*

# Track reconstruction problem as QUBO

Key triplet  $T_i$  properties:

- $H_i$ : the number of holes in the triplet (input parameter),
- $q/p_{T_i}$ : the curvature (inverse of the transverse momentum with the sign determined by the charge  $q$  of the particle – input parameter),
- $\theta_i$ : the angle between two doublets of triplet  $T_i$ .



To quantify the compatibility of track parameters between the two triplets in a valid quadruplet  $(T_i, T_j)$ , we define:<sup>1</sup>

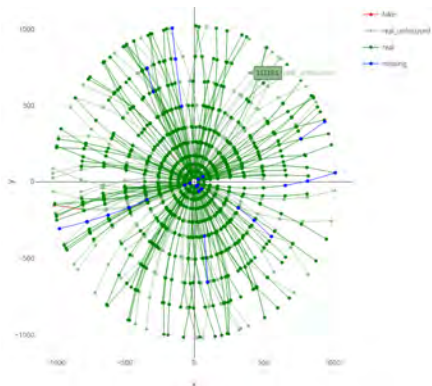
$$S(T_i, T_j) = z_1 \cdot \frac{z_2(1 - |q/p_{T_i} - q/p_{T_j}|)^{z_3} + (1 - z_2)(1 - \max(\theta_i, \theta_j))^{z_4}}{(1 + H_i + H_j)^{z_5}}$$

- Parameter  $z_2$ : encodes the relative importance of curvature with respect to  $\theta$ .
- Parameters  $z_1, z_3, z_4, z_5$ : require tuning; favor high  $p_T$  tracks.  
Default values:  $z_2 = 0.5$  (equal weights),  $z_5 = 2$ ,  $z_1 = z_3 = z_4 = 1$ .

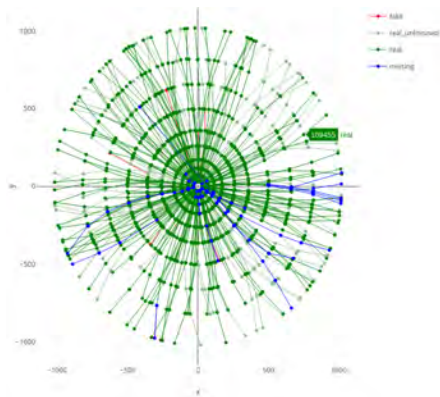
---

<sup>1</sup>Lucy Linder. *Using a Quantum Annealer for particle tracking at LHC*. 2019.

# Reconstructed event



(c) 15% event density



(d) 30% event density

A TrackML event reconstructed by an implementation of the HEPQPR.Qallse<sup>2</sup> algorithm on the HybriLIT Cluster. (Focused dataset – a dataset focusing on particles with a  $p_t \geq 1$  GeV and at least 5 hits.)

<sup>2</sup><https://github.com/derlin/hepppr-qallse>

# Solving QUBO: `neal` and `qbsolv` libraries

- `neal`: is an implementation of a simulated annealing algorithm.<sup>3</sup>

(<https://github.com/dwavesystems/dwave-neal>)

- `qbsolv`: uses a sub-QUBO method, in which the original QUBO matrix is partitioned into pieces. These smaller subproblems are then solved either via a D-Wave system or a classical solver.

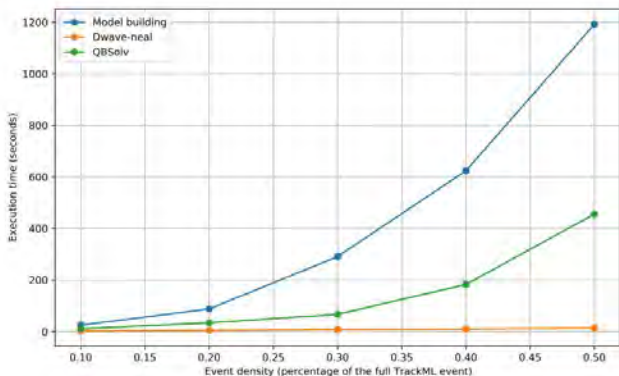
Sub-QUBO methods are mainly motivated by implementation in actual quantum annealing hardware, which has limitations in terms of the problem size due to the available number of qubits.

(<https://github.com/dwavesystems/qbsolv>)

---

<sup>3</sup>Scott Kirkpatrick, C. Gelatt, and M. Vecchi. “Optimization by Simulated Annealing”. In: *Science (New York, N.Y.)* 220 (June 1983).

# Timing performance: `neal` and `qbsolv` libraries



**Figure:** Timing performance as a function of event density<sup>5</sup> for the QUBO model building and solving by the `qbsolv` (in simulation mode) and the `neal` library. The datasets were created from TrackML events, calculations were performed using the HEPQPR.Qallse algorithm on the HybriLIT Cluster. `neal` performs better by about 2 orders of magnitude than `qbsolv` – the reason was not properly understood.

<sup>5</sup>Event density — what proportion of the full event is used in the tracking algorithm.

## Results, TrackML score

Event density (%)	Number of doublets	Precision (%)	Recall (%)	TrackML score (%)
~10	51482	99.13/99.13	97.86/97.86	98.57/98.57
~20	188662	98.80/98.57	97.17/97.64	95.94/96.90
~30	456760	99.12/98.61	97.68/97.26	96.81/96.47
~40	785624	98.30/96.94	96.75/96.28	96.20/95.80
~50	1175314	98.08/95.65	96.72/95.37	95.88/95.01

**Table:** Scoring performance as a function of event density for the `qbsolv` (in classical mode) and the `neal` library. The datasets were created from TrackML events, calculations were performed implementing the `HEPQPR.Qallse` algorithm on the HybriLIT Cluster. The values refer to `neal/qbsolv`.

# Conclusions and future work

## Conclusions

- Based on the HEPQPR.Qallse project, a pilot study of the tracking method on TrackML data has been successfully performed on the HybriLIT Cluster and we intend to adapt this approach to SPD model data.
- The algorithm runtime is given by model building and QUBO solving (the main portion is the combination of doublets into triplets) – a superlinear rise with the number of hits.
- The speed of `qbsolv` is more than two orders of magnitude slower than that of `neal`, with seemingly no gain in performance in track reconstruction.<sup>6</sup>

## Future work

- Improve model building.
- Reduce the number of fake tracks.
- Add [better segment filtering](#), which will possibly reduce the influence of noise hits when used for SPD datasets (and also allow to test the method on TrackML competition data and compare the results).

---

<sup>6</sup>Lucy Linder. *Using a Quantum Annealer for particle tracking at LHC*. 2019.

# Backup slides

# TrackML particle tracking challenge<sup>7</sup>

- **TrackML** – simulation of a typical LHC detector with *high luminosity*.
- **Competition goal** – build an algorithm that quickly reconstructs particle tracks from 3D points left in the silicon detectors.
- **TrackML score** - intersection between reconstructed tracks and the ground truth particles (scored only for overlap > 50%).

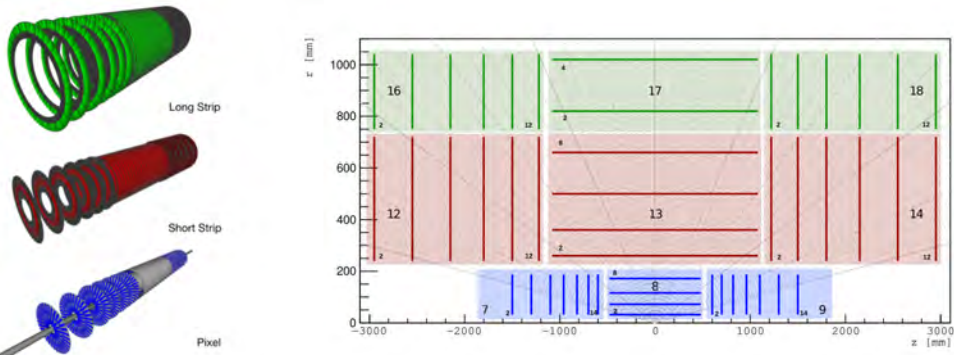


Figure: Moritz Kiehn et al. “The TrackML high-energy physics tracking challenge on Kaggle”.  
In: *EPJ Web of Conferences* 214 (2019)

<sup>7</sup><https://kaggle.com/competitions/trackml-particle-identification>

# Solving QUBO: `neal` and `qbsolv` libraries<sup>9</sup>

- **neal**: uses random moves in the solution space to search for the global minimum. Analogous to finding ground states of matter, a “temperature” is introduced to first “melt” a substance at a high temperature, and then, the temperature is slowly lowered until the system “freezes”, similar to growing a crystal from a melt. A simple algorithm<sup>8</sup> is employed for the iteration, in which a small random displacement is applied to the original spin configuration, and the energy difference  $\Delta E$  is computed. If  $\Delta E < 0$ , then the new configuration is accepted and the algorithm moves on to the next iteration. If  $\Delta E > 0$ , then the new configuration is accepted only probabilistically according to the Boltzmann factor:  $P(\Delta E) = \exp(-\Delta E/kBT)$ .
- **qbsolv**: uses a sub-QUBO method, in which the original QUBO matrix is partitioned into pieces. These smaller subproblems are then solved using a classical solver running the tabu algorithm. Sub-QUBO methods are mainly motivated by implementation in actual quantum annealing hardware, which has limitations in terms of the problem size due to the available number of qubits.

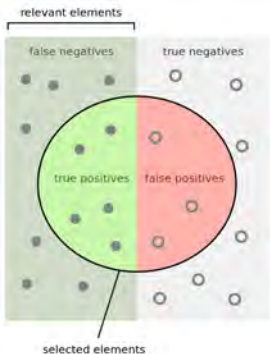
---

<sup>8</sup>N. Metropolis et al. “Equation of state calculations by fast computing machines”. In: *J. Chem. Phys.* 21 (1953).

<sup>9</sup>Hideki Okawa et al. *Quantum Annealing Inspired Algorithms for Track Reconstruction at High Energy Colliders*. 2024.

# Tracking Metrics

- TP (True Positives) - the number of segments correctly identified by the model as real;
- TN (True Negatives) - the number of segments correctly identified by the model as false;
- FP (False Positives) - the number of segments incorrectly identified by the model as real;
- FN (False Negatives) - the number of segments incorrectly identified by the model as false;
- $D^* := TP+FN$  - the total number of all real segments (known from Monte Carlo simulations or datasets like TrackML);
- $D := TP+FP$  - the total number of all segments reconstructed by the model (the set of tracks selected by the algorithm);



**purity/precision:**

How many selected elements are relevant?

$$p = \frac{|D \cap D^*|}{|D|} = \frac{TP}{TP + FP}$$

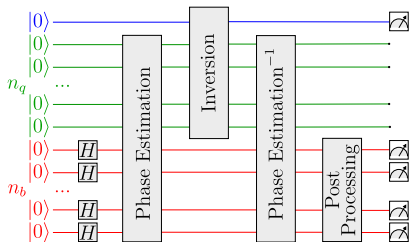
**efficiency/recall:**

How many relevant elements are selected?

$$r = \frac{|D \cap D^*|}{|D^*|} = \frac{TP}{TP + FN}$$

Lucy Linder. *Using a Quantum Annealer for particle tracking at LHC*. 2019

# Universal (gate-based) vs. adiabatic quantum computer



## Universal quantum computer

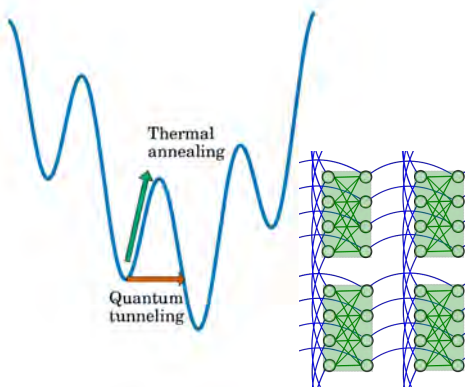
(unitary quantum operations,  
measurements)

general purpose

(QUBO solution:

QAOA/VQE/HHL)

computer	qubits
IBM Q System One (2019)	20
IBM Osprey (2022)	433



## Adiabatic quantum computer

adiabatic evolution of quantum state

optimization problems

D-Wave machine	qubits	couplers
2000Q (2017)	2000	6000
Advantage (2020)	5000	35000

# Simulated and quantum annealing

- **simulated annealing**<sup>10</sup> (SA) can be regarded as a random walk on the search space, whose steps are parameterized by a temperature parameter called  $T$ . At each iteration, a random step is taken, and the step is accepted if it has a lower cost. If the new step has a higher cost, then it is accepted with a probability determined by the temperature  $T$  and the difference between the existing and the new costs. At higher temperatures, the transition between the states occurs more frequently. According to a cooling schedule, at each iteration, the parameter  $T$  is decremented, and the optimal solution is found with the help of the thermal fluctuations.
- **quantum annealing** (QA) - a quantum mechanical heuristic method for solving optimization problems (quantum fluctuations are used instead of thermal fluctuations)

---

<sup>10</sup>Scott Kirkpatrick, C. Gelatt, and M. Vecchi. "Optimization by Simulated Annealing". In: *Science (New York, N.Y.)* 220 (June 1983).

# Quadratic unconstrained binary optimization (QUBO)

**QUBO** model or problem is defined as finding the binary vector  $\mathbf{x}$  corresponding to<sup>11</sup>:

$$\min_{\mathbf{x} \in \{0,1\}^n} \mathbf{x}^T W \mathbf{x} + \mathbf{c}^T \mathbf{x}$$

- $\mathbf{x} \in \mathbb{B}^n$  - set of vectors of fixed length  $n > 0$
- $\mathbb{B} = \{0, 1\}$  - set of binary values (bits)
- $\mathbf{c} \in \mathbb{R}^n$ ;  $W \in \mathbb{R}^{n \times n}$  - symmetric matrix whose elements  $w_{\alpha\beta}$  determine the weights of each pair of indices  $\alpha, \beta \in \{1, \dots, n\}$  of the binary vector
- Since  $x_i^2 = x_i$  for  $i \in \{1, \dots, n\} \implies \mathbf{x}^T W \mathbf{x} + \mathbf{c}^T \mathbf{x} = \mathbf{x}^T (W + \text{diag}(\mathbf{c})) \mathbf{x}$
- Computational complexity of QUBO: the number of binary vectors-candidates rises exponentially with  $n$  ( $|\mathbb{B}^n| = 2^n$ ).

---

<sup>11</sup>A.P. Punnen. *The Quadratic Unconstrained Binary Optimization Problem: Theory, Algorithms, and Applications*. Springer International Publishing, 2022.

# Connection to Ising models

Replacing  $x_i = \frac{1+\sigma_i}{2}$ ,  $x_i \in \{0, 1\}$ , **QUBO is computationally equivalent to the Ising model**, with the Hamiltonian:

$$H(\sigma) = - \sum_{\alpha < \beta} J_{\alpha\beta} \sigma_{\alpha} \sigma_{\beta} - \mu \sum_{\beta} h_{\beta} \sigma_{\beta}, \quad \sigma_{\alpha} \in \{-1, +1\}.$$

**QUBO on D-Wave:** In order to solve a problem using QA, one can formulate the problem as a quadratic unconstrained binary optimization problem (QUBO) which is then easily recast into the problem Hamiltonian.

- coefficients of the quadratic model need to be set, so the **linear terms (biases)** evaluate qubits and **quadratic terms** assess the interactions between pairs of qubits (couplers).
- QUBO **cannot contain constraints** (by definition) - any constraints have to be added to the objective function as penalties.
- $\mathcal{C}$  - the actual set of couplers between qubits is **restricted** in a real quantum annealer (chimera graph in D-Wave)

Note that:

- 1 Not all couplings are available on the hardware, and therefore a process called minor embedding is needed to map the logical qubits to the physical ones.
- 2 Furthermore, there are specific ranges for  $h_i$  and  $J_{ij}$ , so that the coupling and the qubit bias require scaling.