Quantum Annealing for Distributions Unfolding in High-Energy Physics

28/05/2024 – DIFA, UniBo

Simone Gasperini, *PhD student in Data Science and Computation*

Gianluca Bianco, *PhD student in Physics*
Marco Lorusso, *PhD student in Physics*

Who am I?

- 2^{nd} year PhD student in Data Science and Computation at UniBo and INFN

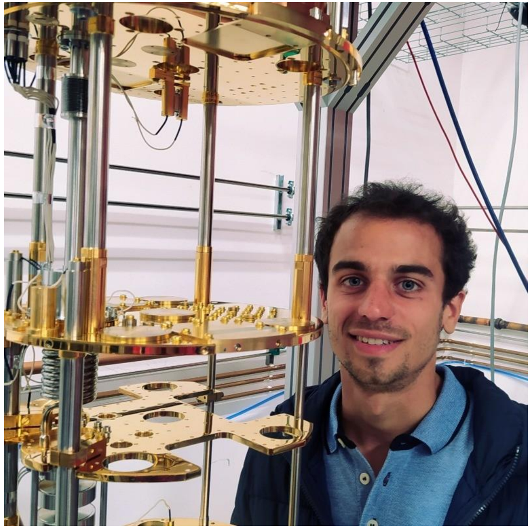
- Education and background in applied computational physics

- Passionate about quantum software development

Research activities

Design, implementation, testing, and benchmarking of quantum algorithms

Applications of quantum computing in experimental High-Energy Physics



Cryostat for superconducting qubits at the Jülich Supercomputing Center (Germany, August 2023)

Outline

Part 1

- History of Simulated Annealing and Quantum Annealing (QA) with D-Wave
- Basic theoretical introduction to QA
- Definition of Quadratic Unconstrained Binary Optimization (QUBO) problems
- Overview about D-Wave hardware and software
- Discussion about features and limitations of QA

D:WAVE
The Quantum Computing Company™

Part 2

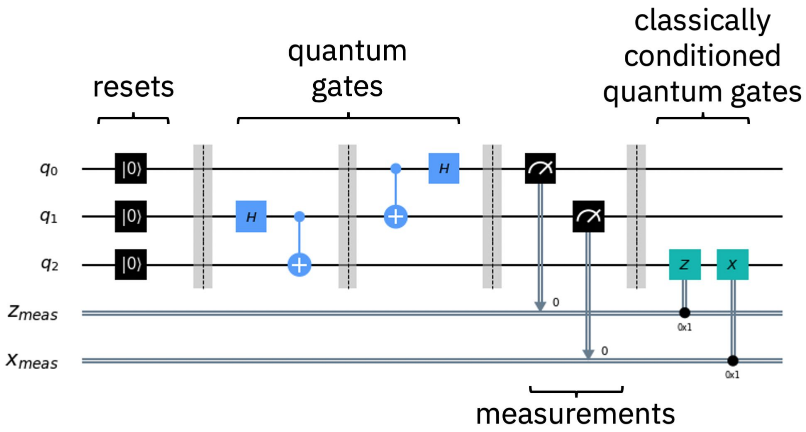
- Introduction to the distribution unfolding problem in High-Energy Physics (HEP)
- Mathematical formulation of the unfolding problem as a QUBO
- Presentation of the QUnfold Python package
- Discussion about experimental results on simulated HEP data

QUNFOLD

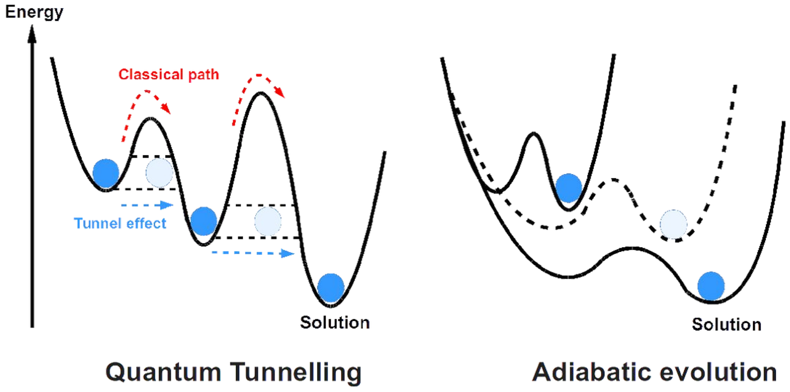
Part 1

Quantum computing models

Gate-based model (IBM, Google, …)



Quantum annealing (D-Wave)



Annealing algorithm history

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

PHYSICAL REVIEW E VOLUME 58, NUMBER 5 NOVEMBER 1998

Quantum annealing in the transverse Ising model

Tadashi Kadowaki and Hidetoshi Nishimori
Department of Physics, Tokyo Institute of Technology, Oh-okayama, Meguro-ku, Tokyo 152-8551, Japan

(Received 30 April 1998)

We introduce quantum fluctuations into the simulated annealing process of optimization problems, aiming at faster convergence to the optimal state. Quantum fluctuations cause transitions between states and thus play the same role as thermal fluctuations in the conventional approach. The idea is tested by the transverse Ising model, in which the transverse field is a function of time similar to the temperature in the conventional method. The goal is to find the ground state of the diagonal part of the Hamiltonian with high accuracy as quickly as possible. We have solved the time-dependent Schrödinger equation numerically for small size systems with various exchange interactions. Comparison with the results of the corresponding classical (thermal) method reveals that the quantum annealing leads to the ground state with much larger probability in almost all cases if we use the same annealing schedule. [S1063-651X(98)02910-9]

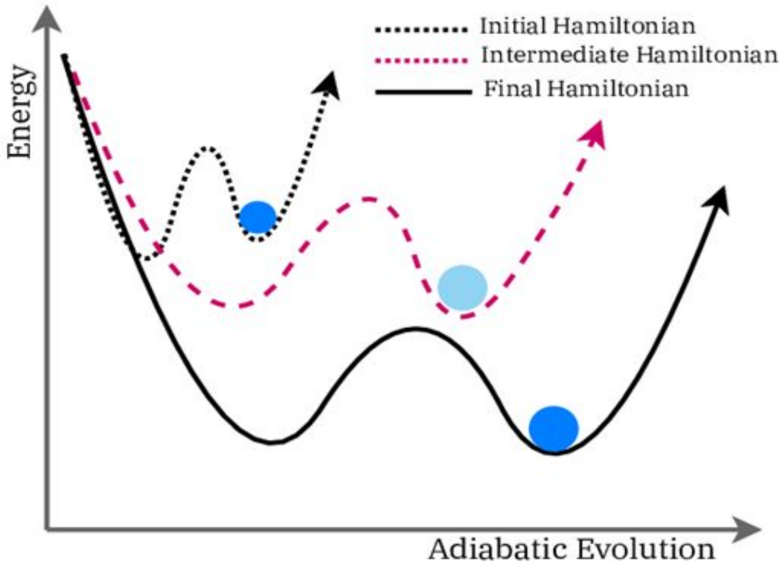
1999: foundation of D-Wave Quantum Systems company in Canada
2007: first prototype of a 16 superconducting qubits quantum annealer
2011: announcement of D-Wave One: *“the world's first commercially available quantum computer”*
2017: launch of D-Wave 2000Q and release of D-Wave Ocean software
2022: launch of D-Wave Advantage 5000Q

Quantum annealing

QA is a quantum optimization process to find the global minimum of an objective function $O(\vec{x})$ over a set of candidate states

1. The quantum-mechanical system is prepared in the known ground-state of an **initial Hamiltonian** H_{init}
2. The seeked solution $\vec{x}_{\min} = \operatorname{argmin} O(\vec{x})$ is encoded in the ground-state of a **final Hamiltonian** H_{fin}
3. The quantum system evolution is controlled by the following time-dependent Hamiltonian:

$$H(t) = A(t)H_{init} + B(t)H_{fin} \qquad A(t)\downarrow \quad B(t)\uparrow$$



Quantum Adiabatic theorem
"If the evolution is slow enough, the quantum-mechanical system stays close the ground-state of the instantaneous Hamiltonian"

From Ising to QUBO

The quantum-mechanical Hamiltonian is an operator on the Hilbert space (Ising model with transverse field):

$$H(t) = A(t)H_{init} + B(t)H_{fin}$$
$$= \underbrace{A(t)\sum_i \sigma_i^x}_{\textbf{Transverse field}} + B(t)\left[\underbrace{\sum_i h_i\sigma_i^z + \sum_{i<j} J_{ij}\sigma_i^z\sigma_j^z}_{\textbf{Ising model}}\right]$$

The **Ising model** term encodes the objective function $O(\vec{s})$

Spin variables: $s_i \in \{-1,+1\}$

$$O(\vec{s}) = \sum_i h_i s_i + \sum_{i<j} J_{ij} s_i s_j \xrightarrow[x_i = \frac{1+s_i}{2}]{} O(\vec{x}) = \sum_i a_i x_i + \sum_{i<j} B_{ij} x_i x_j = \vec{x}^T Q \vec{x}$$

Binary variables: $x_i \in \{0,1\}$ **QUBO!!**

QUBO problem

$$O(\vec{x}) = \sum_i a_i x_i + \sum_{i<j} B_{ij} x_i x_j = \vec{x}^T Q \vec{x}$$

a_i are the linear coefficients

B_{ij} are the quadratic coefficients

Q is the so-called QUBO matrix

Quadratic → Highest power of variables is 2

Unconstrained → No constraint is applied to variables

Binary → Variables take only binary values

Optimization → Minimize (or maximize) an objective function

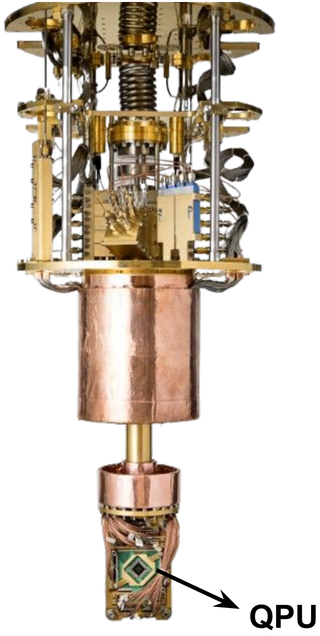
Quantum Annealing allows to solve only problems that can be written as QUBO/Ising

↓

It is not a universal model of quantum computation!

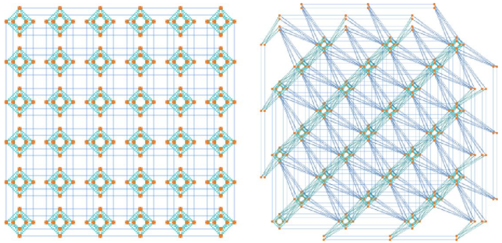
D-Wave QA system

- Shielded from external electric, magnetic, and thermal effects
- Isolated from floor vibrations
- Controlled by a specialized wiring and read-out electronics



Quantum Processing Unit (QPU)

- Lattice of superconducting qubits
- Cooled down to ~15 mK by a liquid Helium refrigeration system
- <25 kW total power consumption
- <u>Physical interaction between qubits limited by a fixed topology</u>

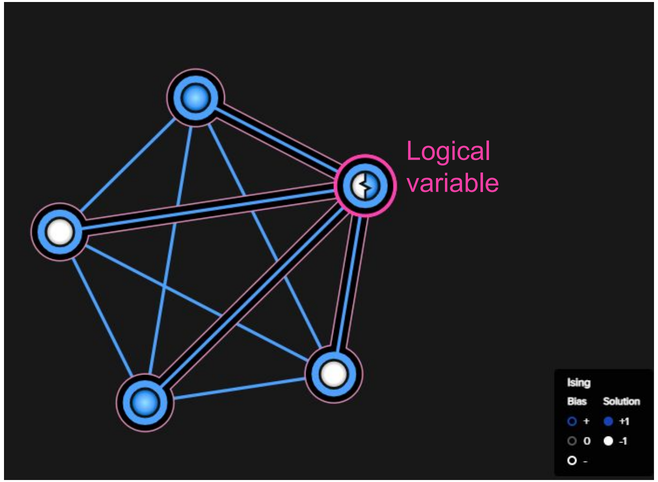


Chimera (left) and Pegasus (right) graph topology

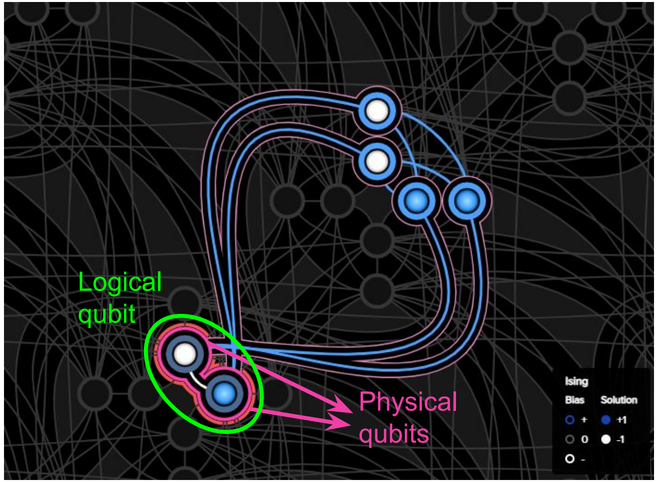
Embedding on D-Wave QPU

Logical qubit ≠ Physical qubit

“Embedding” is the process of mapping a **source graph** (problem topology) to a **target graph** (QPU topology): <u>a cluster of physical qubits may represent a single logical qubit/variable</u>



Problem graph topology



Embedding on QPU Pegasus graph topology

D-Wave software platform

- D-Wave Ocean SDK to define Ising/QUBO model and set annealing parameters
- Implementation of classical solvers to run locally
- Access to real quantum and hybrid solvers via cloud
- D-Wave Leap platform to monitor quantum resources utilization

Part 2

Unfolding problem

In High-Energy Physics (HEP) experiments each measurement apparatus has a unique signature in terms of **detection efficiency**, **space/time resolution**, and **geometric acceptance**

The overall effect is that the distribution of some measured observable is **biased and distorted**

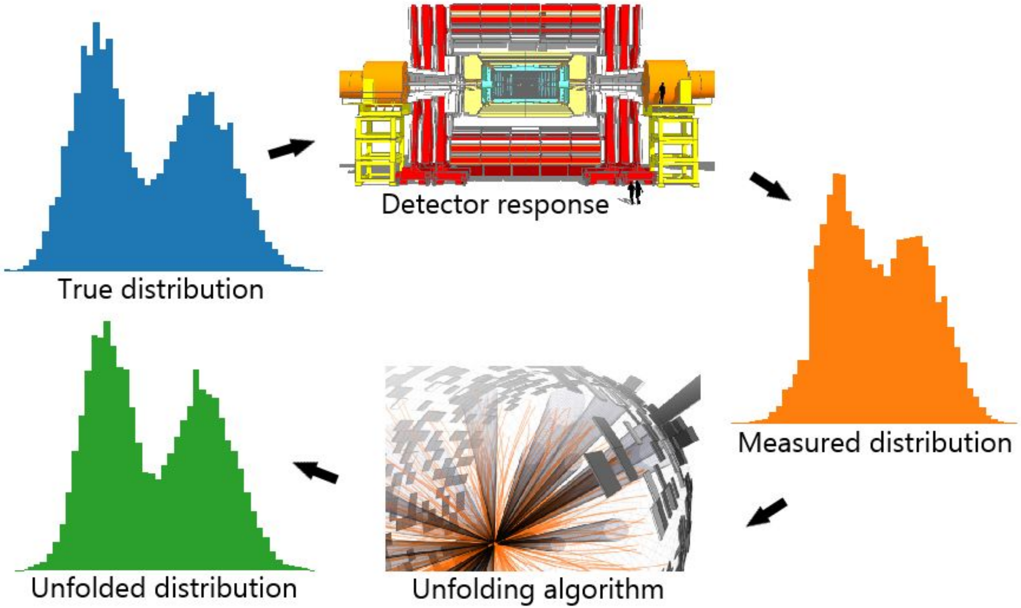
Unfolding is the mathematical technique to correct for this distortion and recover the original distribution

$$\vec{\mu} = R\vec{x}$$

"reco" histogram

response matrix

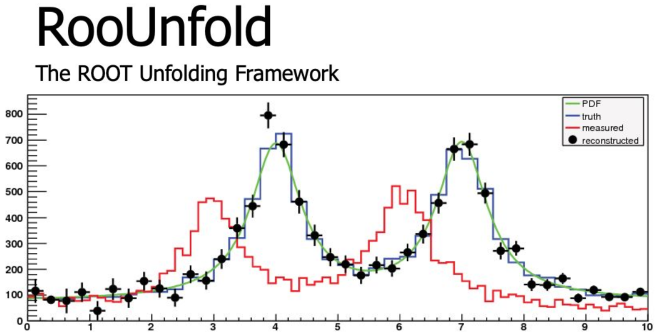
unfolded histogram



Unfolding techniques

Classical unfolding methods in High-Energy Physics:

- Standard matrix inversion (never used in practice)
- Bin-by-bin unfolding (never used in practice)
- Likelihood-based unfolding (SVD)
- Iterative Bayesian unfolding (IBU)



QUnfold is our Python package implementing a quantum approach to tackle the unfolding problem leveraging D-Wave quantum annealing systems: the mathematical formulation is based on the previous work by R. Di Sipio et al. in 2019 [1]

Unfolding problem ➡ **QUBO formulation** ➡ **QUnfold implementation** ➡ **D-Wave systems**

- 5000+ (physical) qubits
- 35000+ couplers
- *Pegasus* topology

[1] Cormier, K., Sipio, R.D. & Wittek, P. Unfolding measurement distributions via quantum annealing. *J. High Energy Phys.*, 128 (2019)

Quadratic model formulation

Log-likelihood maximization unfolding: $\max_{\vec{x}} \left(\log \mathcal{L}(\vec{\mu}|\vec{d}) + \lambda \mathcal{S}(\vec{\mu}) \right)$

measured histogram

regularization

$\vec{\mu} = R\vec{x}$

"reco" histogram

response matrix

unfolded histogram

Quadratic model minimization unfolding: $\min_{\vec{x}} \left(||R\vec{x} - \vec{d}||^2 + \lambda ||G\vec{x}||^2 \right)$

Tikhonov regularization
discrete 2^{nd}-order derivative
(Laplacian operator)

$\rightarrow$ This is a <u>classical model</u> (nothing quantum yet!)
$\rightarrow$ $\vec{x}$ is an array of <u>integer numbers</u> (need binary variables!)

QUBO formulation

Define $\vec{a} = -2R^T\vec{d}$
and $B = R^TR + \lambda G^TG$

Encode integer numbers
into binary variables

$$H(\vec{x}) = \sum_i a_i x_i + \sum_{i,j} b_{ij} x_i x_j = \vec{a} \cdot \vec{x} + \vec{x}^T B \vec{x}$$

The total number of binary variables (logical qubits) scales as:

$$N_{\text{qubits}} \propto n_{\text{bins}} \cdot \log_2(n_{\text{entries}})$$

n_{bins} is the number of bins of the histogram
n_{entries} is the total number of entries in the histogram

For instance, for a histogram with 20 bins and 50M entries we need **~350 logical qubits**

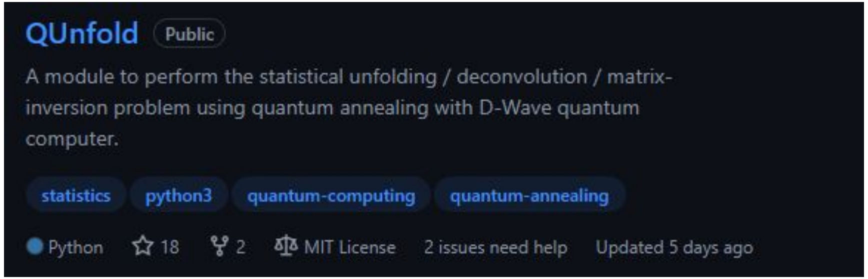
QUnfold software

- Implemented using **D-Wave Ocean SDK** but fully compatible with ROOT framework for HEP
- Minimal and intuitive **Python interface**
- **Open-source** repository available on GitHub
- Easy to install in Python/Conda environment:

```
pip install Qunfold
```

Available solver methods:

- Simulated annealing sampler (CPU only)
- Hybrid sampler (CPU + QPU)
- Quantum annealing sampler (QPU only)

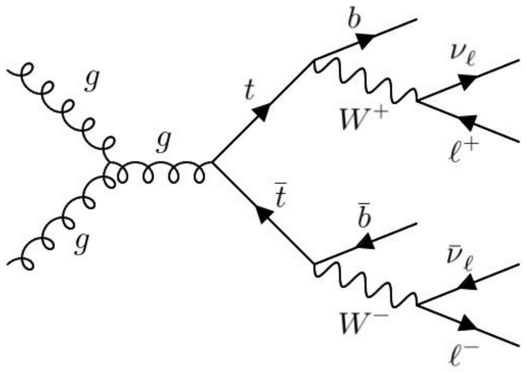



```
unfolder = QUnfoldQUBO(response, measured, lam=0.1)
unfolder.initialize_qubo_model()
unfolded_SA, error_SA = unfolder.solve_simulated_annealing(num_reads=10)
```

Test on HEP data

Simulated HEP dataset

- $t\bar{t}$ **process** in the dileptonic channel (2 leptons and 2 b-jets required in the final state)
- 2.5M truth-level events generated using the *MadGraph* generator (*truth* histogram)
- Detector-level data generated using the *Delphes* simulator (*measured* histogram)

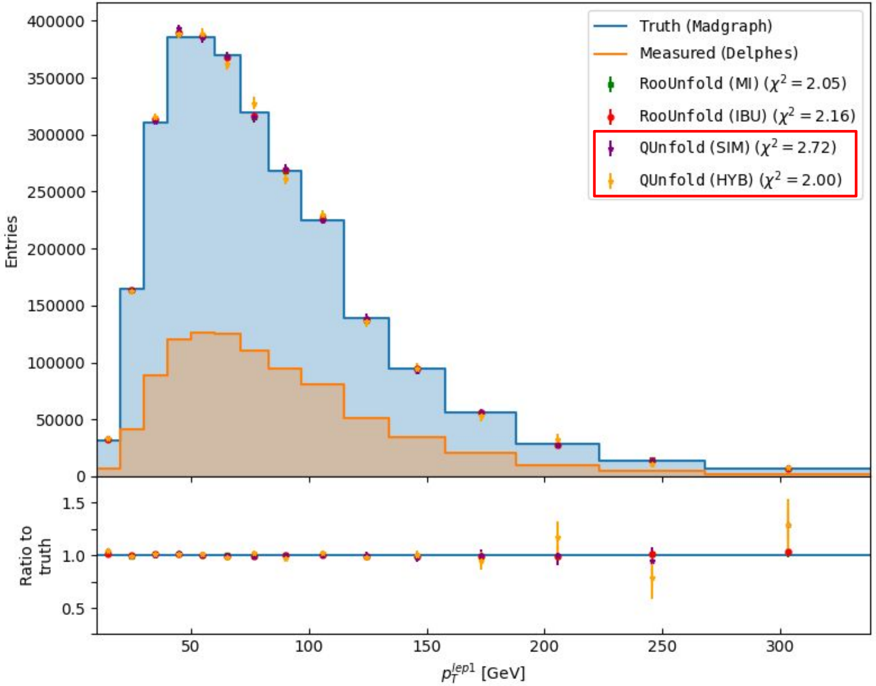


Kinematic variables unfolding

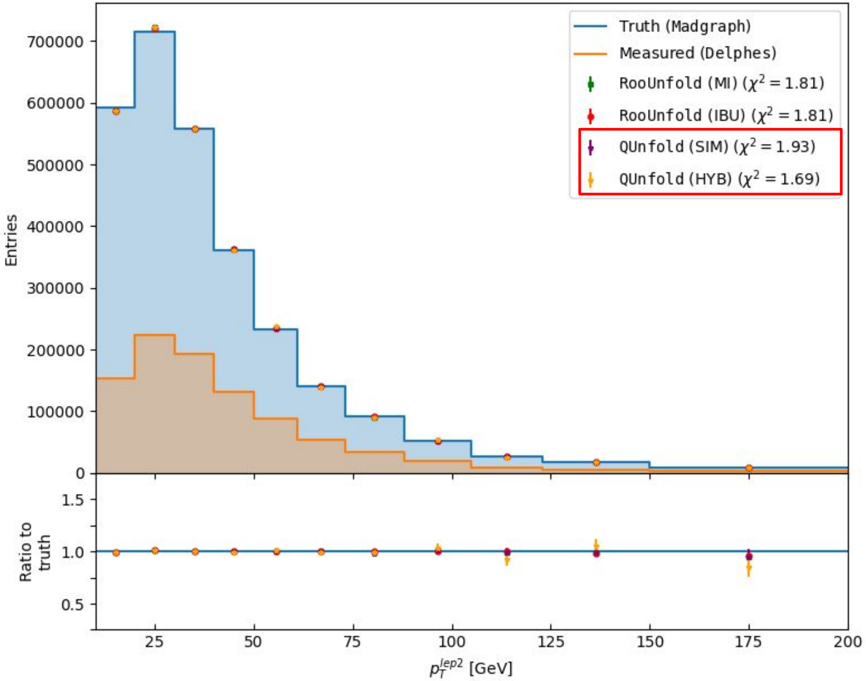
- D-Wave **simulated annealing** and **hybrid quantum-classical** solvers are used
- Results are compared with standard HEP methods (MI and IBU, *RooUnfold*)
- Multiple experiments are run to evaluate the quality of the result (χ^2 test) and to estimate the statistical errors

Experimental results

Leading lepton p_T

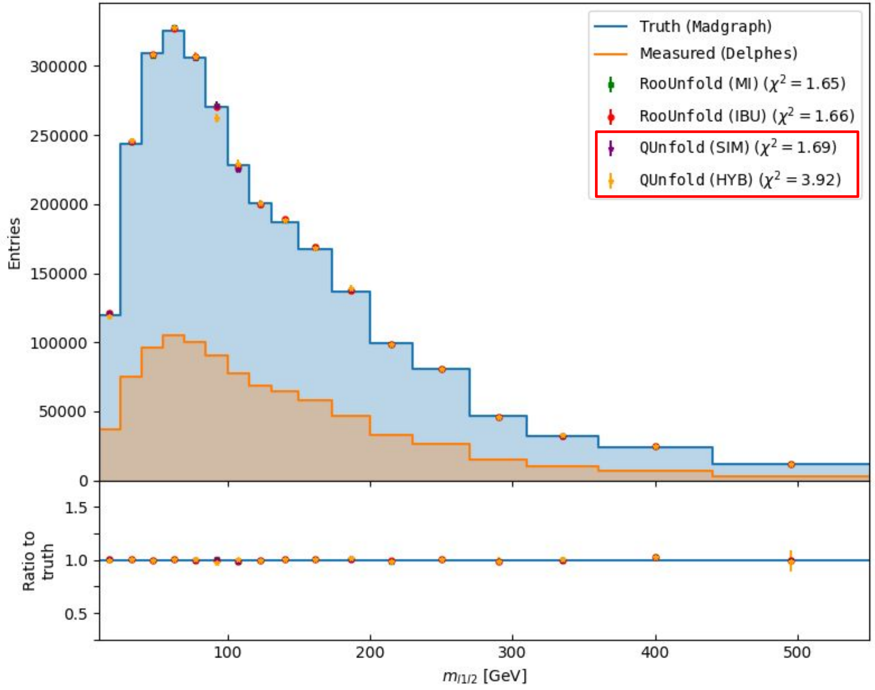


Subleading lepton p_T

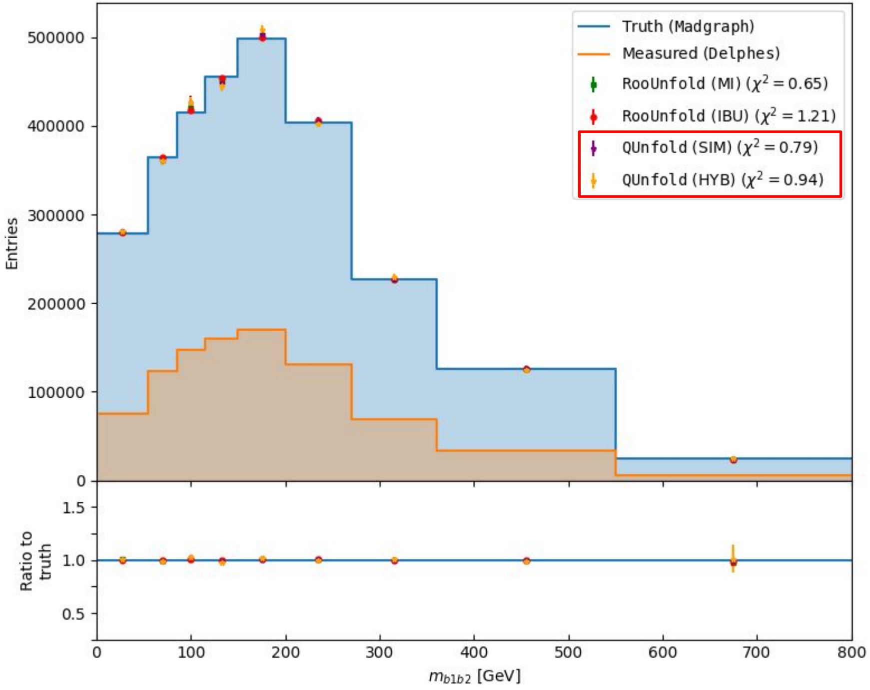


Experimental results

Leptons invariant mass



***b*-jets invariant mass**



Conclusion

- New unfolding approach based on the **QUBO formulation** of the problem and **quantum annealing**
- Model implemented in the **QUnfold Python package**
- Algorithm tested on simulated High-Energy Physics data

Next steps

- Further optimize the algorithm: integer model *binarization*, QUBO matrix *preconditioning*, graph embedding on D-Wave QPU
- Perform more experiments on **real quantum hardware** (we recently got D-Wave QPU access time by CINECA)
- Design, implement, and test a **gate-based approach** for the same problem (we started a collaboration with CERN QTI and IONQ)

Thank you!
Any questions?

https://github.com/JustWhit3/QUnfold

Simone Gasperini

Gianluca Bianco

Marco Lorusso

Simone Gasperini - simone.gasperini4@unibo.it