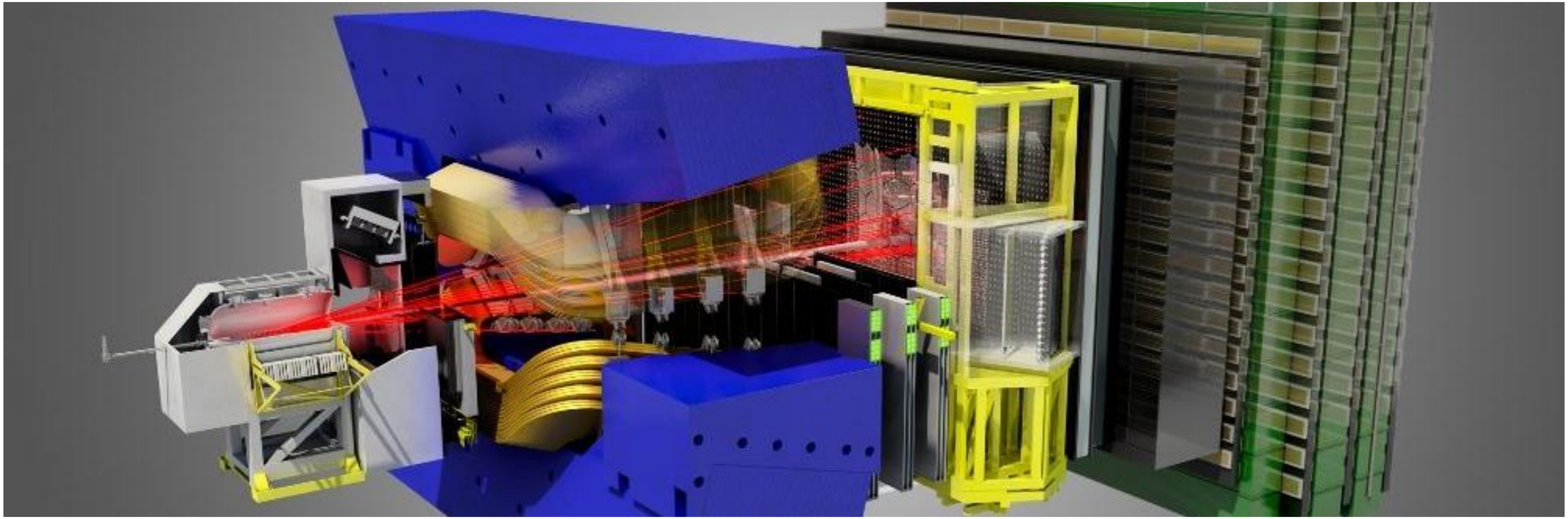


A Toy Model For Reconstructing Particle Tracks at LHCb at CERN with Quantum Computing

Based on previous work and collaboration with Prof. Marcel Merk and his team (The NIKHEF institute and Maastricht University)

ALAIN CHANCÉ

Alain Chancé

President [MolKet SAS](), a member of QED-C and QuIC, offering management consulting and AI services in quantum molecular dynamics, cryptography, neuromorphic computing, with cloud-based hybrid HPC and quantum computing platforms. Alain is the founder and CEO of Quantalain SASU and Alainquant LLC, both business management consulting startups.

He is a Qiskit Advocate, an IEEE Senior Member, a society affiliate of the American Chemical Society (ACS), and a member of the American Physical Society (APS).

Alain also serves on the Advisory Board Committee of the North America Technical Institute (NATI), Virginia, USA, which offers a professional certificate program in quantum computation.

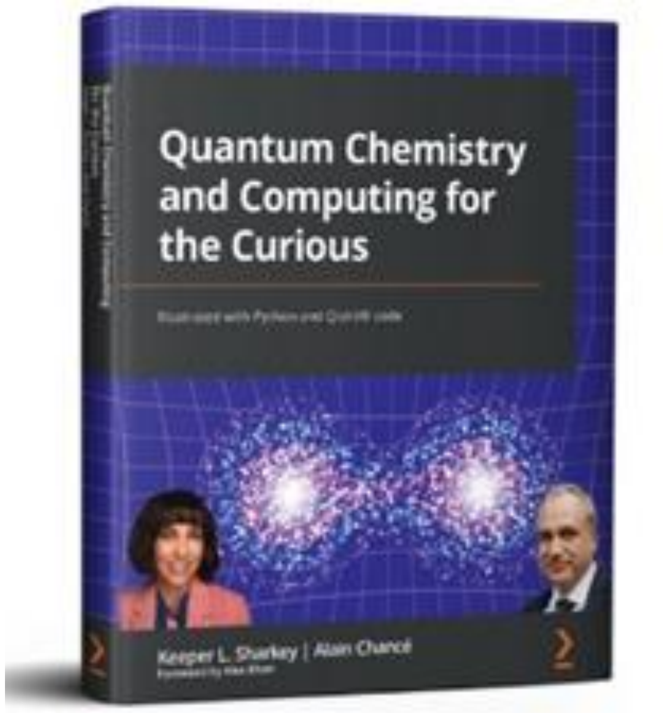
Participant in the IBM Quantum Developer Conference 2025 and 2024.

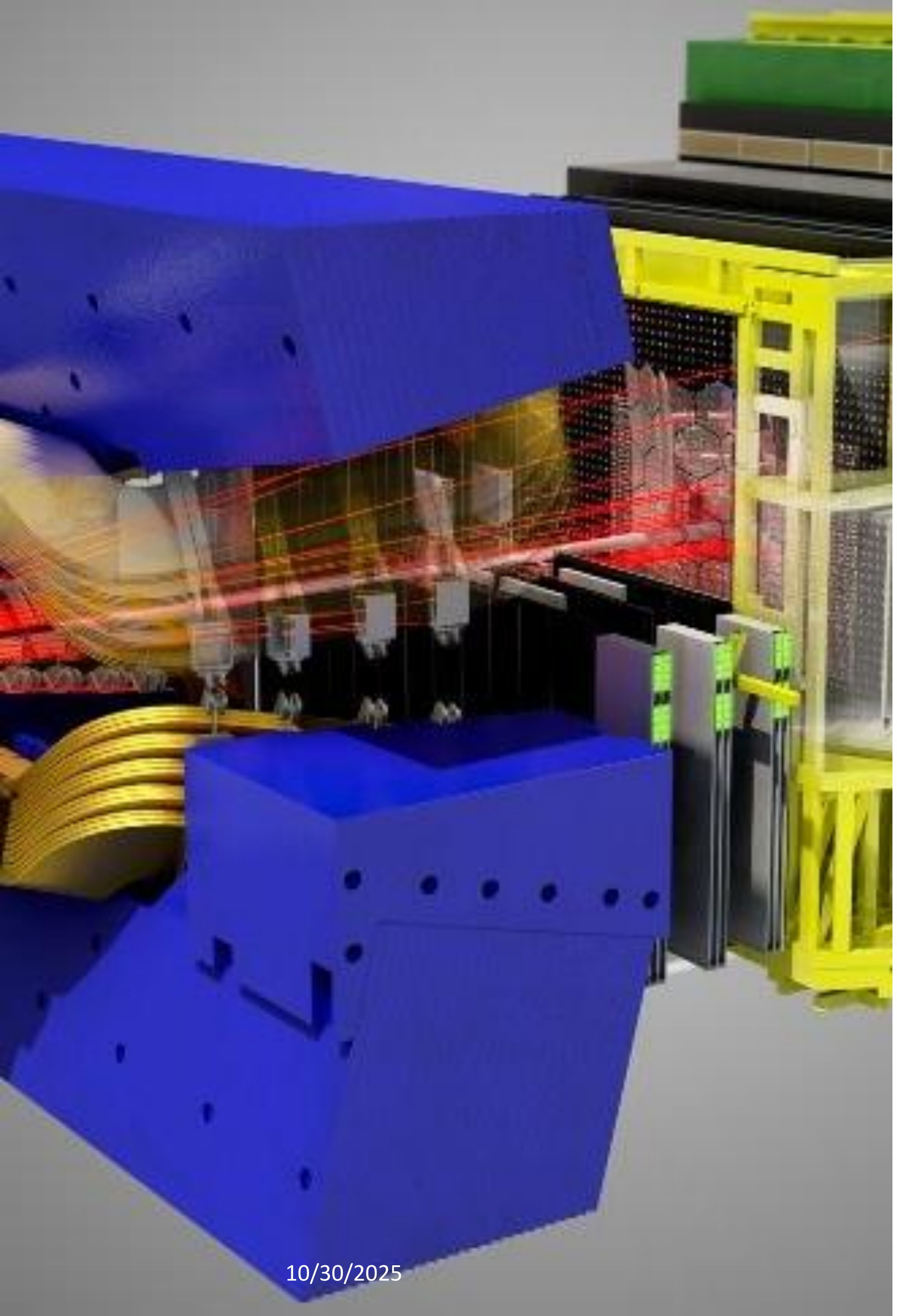
He was a keynote co-speaker at the World Artificial Intelligence Cannes Forum (WAICF), where he presented “Combining the Power of AI with Quantum”, February 2024.

He is a co-author of the book, Keeper L. Sharkey, A. Chancé, "Quantum Chemistry and Computing for the Curious: Illustrated with Python and Qiskit® code", Packt Publishing (2022), ISBN-13: 978-1803243900. https://a.co/d/hIVmgQl

Alain has over 30 years of experience in major enterprise transformation projects. He holds a diploma as an Ingénieur civil des Mines from École des Mines de Saint-Étienne in France (1981).

LinkedIn: https://www.linkedin.com/in/alainchance/





Abstract

We present a toy model for reconstructing particle tracks at LHCb experiment at CERN with quantum computing derived from the following work:

Xenofon Chiotopoulos, Miriam Lucio Martinez, Davide Nicotra, Jacco A. de Vries, Kurt Driessens, Marcel Merk, and Mark H.M. Winands, TrackHHL: A Quantum Computing Algorithm for Track Reconstruction at the LHCb, EPJ Web of Conferences 337, 01181 (2025), https://doi.org/10.1051/epjconf/202533701181

Image credit: LHCb Detector Performance, JINST 3 (2008) S08005, Int. J. Mod: Phys. A 30, 1530022 (2015), CERN-LPCC-2018-04, https://doi.org/10.1142/S0217751X15300227

Agenda

1. The Large Hadron Collider beauty (LHCb) experiment at CERN.
2. Particle track reconstruction in the VELO.
3. Parameterizing an Ising-like Hamiltonian in terms of doublets S.
4. Minimizing an Ising-like Hamiltonian using matrix inversion.
5. Solving the system of linear equations with the 1-Bit HHL algorithm.
6. Efficient hybrid 1-Bit HHL classical implementation.
7. GitHub repository LHCb_VeLo_Toy_Model_1-Bit_HHL.
8. Set-up your own 1-Bit HHL track simulation toy model simulation
9. 1-bit HHL quantum circuit for 2 particles and 3 layers simulation.

The Large Hadron Collider beauty (LHCb) experiment at CERN

The LHCb Experiment at CERN is a general-purpose detector at the Large Hadron Collider (LHC) and specializes in investigating the slight differences between matter and antimatter by studying a type of particle called the "beauty quark", or "b quark".

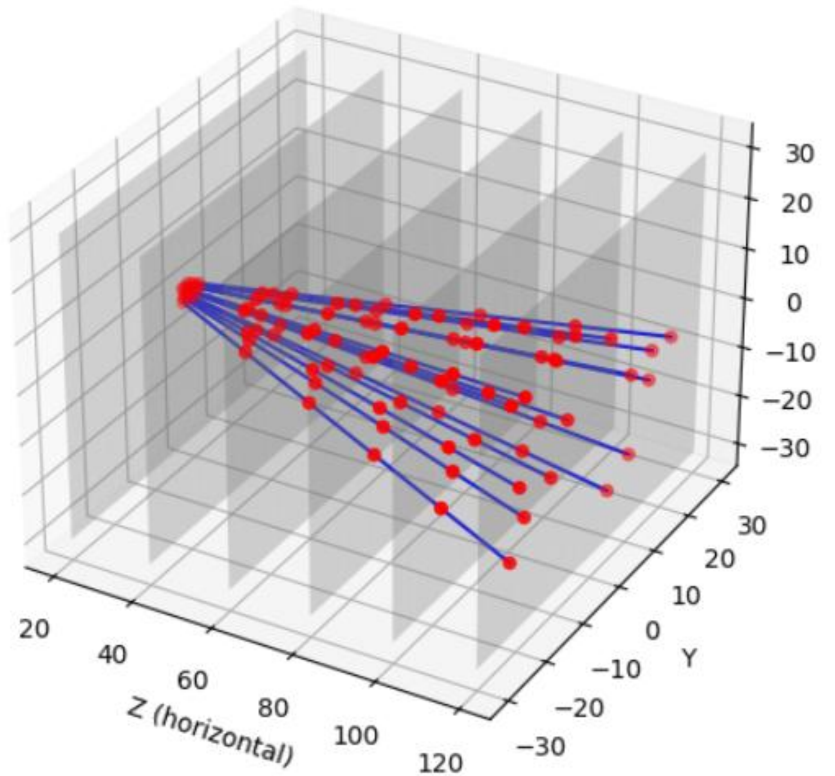
It uses a series of subdetectors to detect mainly forward particles – those thrown forwards by the collision in one direction. The first subdetector is mounted close to the collision point, with the others following one behind the other over a length of 20 meters.

The 5600-tonne LHCb detector is made up of a forward spectrometer and planar detectors. It is 21 meters long, 10 meters high and 13 meters wide, and sits 100 meters below ground near the town of Ferney-Voltaire, France.

As of 2024, more than 1600 members from 98 institutes in 22 countries, including 1100 authors.

Source: https://home.cern/science/experiments/lhcb

Particle track reconstruction in the VELO



In the High Luminosity phase of the Large Hadron Collider (HL-LHC), thousands of particles are produced simultaneously. Particles leave energy hits in detector layers. Hits are reconstructed into particle tracks. Tracks reveal Primary Vertices (collision points).

Tracks in the LHCb Vertex Locator (VELO) can be modeled as straight lines because it is the sub-detector closest to the LHCb collision point and it contains a negligible magnetic field.

Parameterizing an Ising-like Hamiltonian in terms of doublets S

The Hamiltonian $H(S)$ is parameterized in terms of doublets S, these doublets are possible connections between two hits in successive detector layers and take a binary value to indicate if they actively contribute to a track, $S_i \in \{0,1\}$.

$$H(S) = -\frac{1}{2}\left[\sum_{abc} f(\theta_{abc}, \varepsilon) S_{ab} S_{bc} + \gamma \sum_{ab} S_{ab}^2 + \delta \sum_{ab} (1 - 2S_{ab})^2\right]$$

$$f(\theta_{abc}, \varepsilon) = \begin{cases} 1, \text{if } cos(\theta_{abc}, \varepsilon) \geq 1 \\ 0, \text{otherwise} \end{cases}$$

Term	Definition
$H_{ang} = \sum_{abc} f(\theta_{abc}, \varepsilon) S_{ab} S_{bc}$	The angular term assigns values for straight doublets. It is the most important as it determines if a set of doublets S_i and S_j are aligned within ε.
$H_{spec} = \sum_{ab} S_{ab}^2$	The regularization term makes the spectrum of the matrix A positive, where $\nabla_S H = -AS + b$ and $b = \|1,1,1,\ldots,1\rangle$.
$H_{gap} = \sum_{ab} (1 - 2S_{ab})^2$	The gap term ensures a gap in the solution spectrum.

Minimizing an Ising-like Hamiltonian using matrix inversion

By relaxing $S_i \in \mathbb{R}$, we find its minimum by taking the derivative of the quadratic H, obtaining a system of linear equations:

$$\nabla_S H = -AS + b = 0, \; AS = b$$

Matrix inversion yields the solution of reconstructed tracks.

The system of linear equations $AS = b$ is solved classically using SciPy:

```
vector_b = np.ones(len(A))
sol, _ = sci.sparse.linalg.cg(A, vector_b, atol=0)
```

The resulting vector S of real values is subsequently discretized to obtain an "on"/"off" status by setting a threshold T_classical:

```
T_classical = param["T_classical"]
disc_sol = (sol > T_classical ).astype(int)
```

Solving the system of linear equations with the 1-Bit HHL algorithm

The Harrow–Hassidim–Lloyd (HHL) algorithm promises a complexity improvement over the best classical alternatives for solving sparse systems of linear equations.

However, its practical implementation faces considerable challenges. The Quantum Phase Estimation (QPE) step results in prohibitively deep circuits, making the algorithm unfeasible on currently available hardware short of fault-tolerant quantum computing.

The 1-Bit HHL algorithm, presented in the paper [TrackHHL: A Quantum Computing Algorithm for Track Reconstruction at the LHCb](), applies a first-order Suzuki–Trotter decomposition to approximate the time-evolution operator. By restricting the QPE accuracy to a single bit, the algorithm can efficiently determine whether a phase is close to zero or significantly different.

The resulting vector S of real values is subsequently discretized to obtain an "on"/"off" status by setting a threshold T_hhl:

$$\text{T\_hhl} = \text{param}[\text{"T\_hhl"}]$$

$$\text{disc\_x\_hhl} = (\text{x\_hhl} > \text{T\_hhl}).\text{astype}(\text{int})$$

Efficient hybrid 1-Bit HHL classical implementation

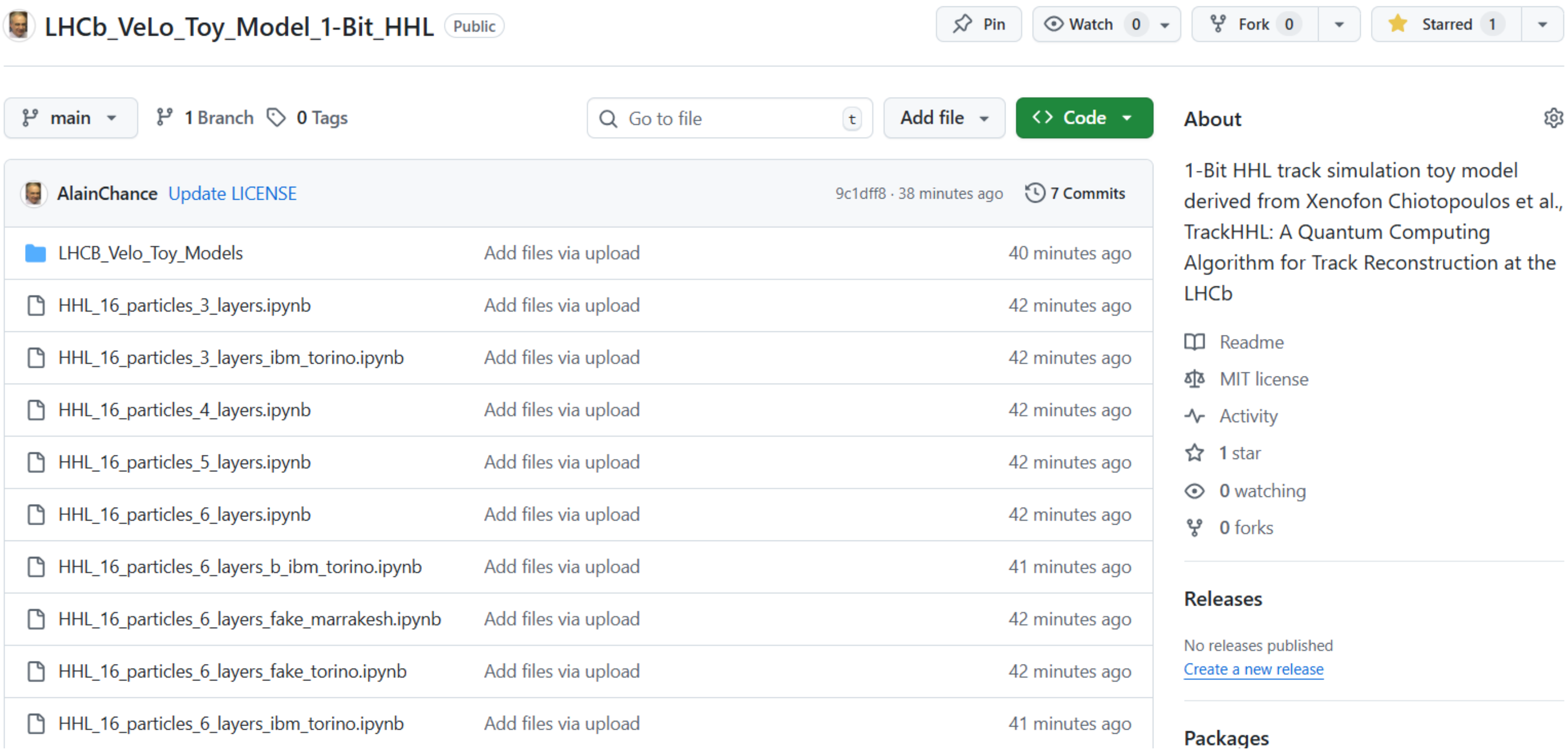
```
Extracted HHL solution (normalized):
[0.43349225 0.36955737 0.30330749 0.33215972 0.36844616 0.34602271
 0.31801489 0.34142137]

Computed T_hhl: 0.3293444389048007

Discretized HHL solution:
[1 1 0 1 1 1 0 1]

Indices of HHL solution:
[0, 1, 3, 4, 5, 7]

Found primary vertices:
(0.0, 0.0, 10.0)

Correct indices of HHL solution is equal to correct indices of classical solution: False

Good indices of HHL solution:
[0, 3, 4, 7]
Good indices of HHL solution is equal to correct indices of classical solution: True

Reconstructed event tracks from discretized HHL solution

Track ID: 0
  Hit ID: 3, x: -1.577509905705905, y: -0.048043641520914566, z: 20, Module ID: 1
  Hit ID: 4, x: -4.732529717117715, y: -0.1441309245627437, z: 40, Module ID: 2
  Hit ID: 5, x: -7.887549528529525, y: -0.24021820760457285, z: 60, Module ID: 3

Track ID: 1
  Hit ID: 0, x: 0.3930194759052289, y: 1.8827028137012705, z: 20, Module ID: 1
  Hit ID: 1, x: 1.1790584277156868, y: 5.648108441103812, z: 40, Module ID: 2
  Hit ID: 2, x: 1.9650973795261446, y: 9.413514068506352, z: 60, Module ID: 3
```

[HHL_2_particles_3_layers_ibm_torino.ipynb]

In our implementation, both classical and 1-Bit HHL simulations only use the hits in the first three layers.

The function get_tracks_smart() in the module [One_Bit_HHL_Simulation.py] completes the list of active segments with hits in all the outer layers.

It uses only active segments that intersect with the z-axis to reconstruct tracks. Their intersection with the z-axis gives a list of found primary vertices.



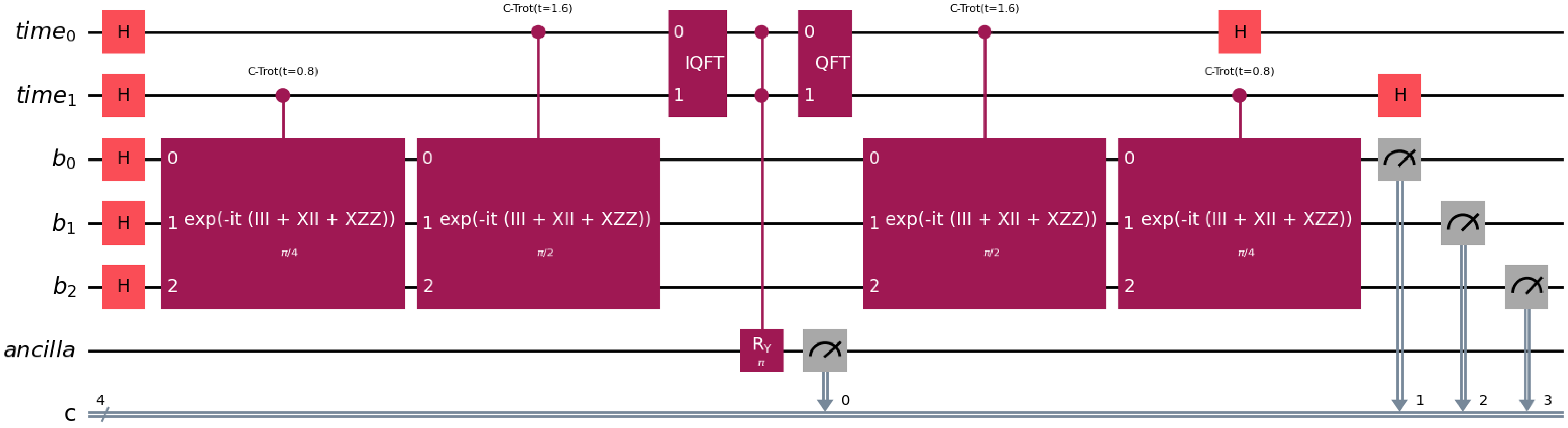
GitHub repository LHCb_VeLo_Toy_Model_1-Bit_HHL

https://github.com/AlainChance/LHCb_VeLo_Toy_Model_1-Bit_HHL/tree/main

Set-up your own 1-Bit HHL track simulation toy model simulation

Duplicate one of the Jupyter notebooks from the GitHub repo, for instance HHL_2_2_particles_3_layers_fake_marrakesh.ipynb, rename it and customize the configuration

```
config = {
    #--------------------
    # Simulation options
    #--------------------
    "dz": 20,                                # Layer spacing (mm)
    "layers": 3,                             # Number of layers
    "n_particles": [2,2],                    # Number of particles
    "p_vertices": [(0,0,5),(0,0,10)],        # Primary vertices
    "do_draw": True,                         # Whether to draw the HHL circuit
    "measurement_error": 0.0,                # HIT RESOLUTION (sigma on measurement) (sigma)
    "collision_noise": 0.0,                  # MULTIPLE SCATTERING (angular noise proxy)
    "ghost_rate": 1e-2,                      # Ghost (fake) track rate
    "drop_rate": 0.0,                        # Hit drop (inefficiency) rate
    "display_hits": True,                    # Whether to display hits
    "display_tracks": True,                  # Whether to display events and ghost tracks
    "T_classical": 0.45,                     # Threshold for discretizing classical solutions
    "T_hhl": None,                           # Threshold for discretizing 1-Bit HHL solutions - None:
    "do_spectrum": True,                     # Whether to analyze the classical solution spectrum
    "do_print_counts": True,                 # Whether to print raw measurement counts
    "max_abs_eigen": None,                   # Compute max_abs_eigen = round(np.max(np.abs(np.linalg.
    #-----------------------------------------
    # Files containing token (API key) and CRN
    #-----------------------------------------
    "token_file": "Token.txt",               # Token file
    "CRN_file": "CRN.txt",                   # CRN file
    #-------------
    # Run options
    #-------------
    "backend_name": "fake_marrakesh",        # AerSimulator noiseless or Fake QPU or real IBM cloud b
    "run_on_QPU": True,                      # Whether to run the quantum circuit on the target hardw
    "nshots": 100000,                        # Number of shots
    'opt_level':1,                           # Optimization level
    "poll_interval": 5,                      # Poll interval in seconds for job monitor
    "timeout": 600,                          # Time out in seconds for gob monitor
}
```



1-bit HHL quantum circuit for 2 particles and 3 layers simulation

https://github.com/AlainChance/LHCb_VeLo_Toy_Model_1-Bit_HHL/blob/main/HHL_circuit.png

Glossary

Term	Definition
Detector	The job of a particle detector is to record and visualize the explosions of particles that result from the collisions at accelerators. The information obtained on a particle's speed, mass, and electric charge help physicists to work out the identity of the particle. http://www.cernmg.free.bg/en/detectors.html
Hit	In the High Luminosity phase of the Large Hadron Collider (HL-LHC), thousands of particles are produced simultaneously and traverse sensitive detection layers where they deposit small amounts of energy, resulting in so-called hits in the detectors. https://doi.org/10.1051/epjconf/202533701181
Large Hadron Collider (LHC)	The Large Hadron Collider (LHC) is the world's most energetic proton-proton collider, located at the CERN laboratory on the Swiss-French border just outside of Geneva. It was designed and built to discover physics Beyond the Standard Model (BSM), either through direct detection of new massive particles at the CMS and/or ATLAS experiments, or through the indirect detection at the LHCb experiment. https://hep.syr.edu/quark-flavor-physics/lhcb-cern/
VELO	The LHCb Vertex Locator (VELO) is the sub-detector closest to the LHCb collision point and it contains a negligible magnetic field.

Bibliography
The Large Hadron Collider beauty (LHCb) experiment at CERN

1. The Large Hadron Collider beauty (LHCb) experiment, https://home.cern/science/experiments/lhcb
2. LHCb Taking a closer look at LHC, https://www.lhc-closer.es/taking_a_closer_look_at_lhc/0.lhcb
3. LHCb @ Syracuse: An overview, https://hep.syr.edu/quark-flavor-physics/lhcb-cern/
4. LHCb Upgrade II, Detector and Physics Prospects, Vincenzo Vagnoni (INFN Bologna, CERN), for the LHCb collaboration, The 2024 International Workshop on Future Tau Charm Facilities, 15 January 2024, https://indico.pnp.ustc.edu.cn/event/91/contributions/6351/attachments/1859/3063/FTCF%2015%20January%202024.pdf
5. Davide Zuliani, Search for Higgs decay to charm quark pair at LHCb, https://cds.cern.ch/record/2882378/files/higgs2023_zuliani_v3.pdf
6. LHCb reveals secret of antimatter creation in cosmic collisions. The finding may help determine whether or not any antimatter seen by experiments in space originates from dark matter, 7 April 2022, https://home.cern/news/news/physics/lhcb-reveals-secret-antimatter-creation-cosmic-collisions
7. LHCb Detector Performance, JINST 3 (2008) S08005, Int. J. Mod: Phys. A 30, 1530022 (2015), CERN-LPCC-2018-04, https://doi.org/10.1142/S0217751X15300227
8. LHCb publications, https://lhcbproject.web.cern.ch/Publications/LHCbProjectPublic/Summary_all.html

Bibliography

Algorithms for track reconstruction

1. Xenofon Chiotopoulos, TrackHHL: A Quantum Computing Algorithm for Track Reconstruction at the LHCb, https://indico.cern.ch/event/1338689/contributions/6010017/attachments/2951297/5188722/CHEP_ppt.pdf
2. Xenofon Chiotopoulos, Miriam Lucio Martinez, Davide Nicotra, Jacco A. de Vries, Kurt Driessens, Marcel Merk, and Mark H.M. Winands, TrackHHL: A Quantum Computing Algorithm for Track Reconstruction at the LHCb, EPJ Web of Conferences 337, 01181 (2025), https://doi.org/10.1051/epjconf/202533701181
3. Okawa, Hideki, Quantum Algorithms for Track Reconstruction at High Energy Colliders, Workshop of Tracking in Particle Physics Experiments, May 17-19, 2024, https://indico.ihep.ac.cn/event/21775/contributions/155907/attachments/78247/97329/okawa_QTrack_20240517.pdf
4. Quantum pathways for charged track finding in high-energy collisions, Front. Artif. Intell., 30 May 2024, Sec. Big Data and AI in High Energy Physics, Volume 7 – 2024, https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2024.1339785/full
5. D. Nicotra et al., arXiv:2308.00619v2, 7 Oct 2023, A quantum algorithm for track reconstruction in the LHCb vertex detector, https://arxiv.org/pdf/2308.00619
6. Primary Vertex Reconstruction at LHCb, LHCb-PUB-2014-044, October 21, 2014, https://cds.cern.ch/record/1756296/files/LHCb-PUB-2014-044.pdf

Bibliography

Harrow–Hassidim–Lloyd (HHL) Algorithm

1. V., Shwetha, Abinaya Selvarajan, Aarthi A., and Sneka R.. "Quantum Speedup for Linear Systems: An Analysis of the HHL Algorithm Using IBM Qiskit." Journal of Electronics and Informatics 6, no. 4 (2024): 317-331, https://irojournals.com/iroei/article/view/6/4/3
2. Solving linear systems of equations using HHL and its Qiskit implementation, Qiskit Tutorial, https://github.com/Qiskit/textbook/blob/main/notebooks/ch-applications/hhl_tutorial.ipynb

Bibliography
Quantum Phase Estimation (QPE) Algorithm

1. Stefano Scali, Josh Kirsopp, Antonio Márquez Romero, Michał Krompiec, Spectral subspace extraction via incoherent quantum phase estimation, 16 Oct 2025, arXiv:2510.14744 quant-ph, https://doi.org/10.48550/arXiv.2510.14744
2. Antoine Lemelin, Christophe Pere, Olivier Landon-Cardinal, Camille Coti, Mid-circuit measurement as an algorithmic primitive, 2 Sep 2025, arXiv:2506.00118 quant-ph, https://doi.org/10.48550/arXiv.2506.00118
3. Phase estimation variants and its implication for quantum/classical architecture by Microsoft, From the need to hybridize algorithmically to the need to integrate QPUs with CPUs, J. Mikael, EDF, E. Vergnaud, Teratec TQCI, Conference on QPU/CPU Integration, https://www.teratec.eu/library/seminaires/2022/TQCI/Microsoft_Hybrid_QC_EDF.pdf
4. quantum-phase-estimation.ipynb, Qiskit Textbook, https://github.com/Qiskit/textbook/blob/main/notebooks/ch-algorithms/quantum-phase-estimation.ipynb
5. qiskit.circuit.library.phase_estimation, https://quantum.cloud.ibm.com/docs/en/api/qiskit/qiskit.circuit.library.phase_estimation
6. Non-variational and Phase Estimation algorithms, Quantinuum's InQuanto 5.1.0, https://docs.quantinuum.com/inquanto/manual/algorithms/non_variational_overview.html