Experimental applications of quantum algorithms in high-energy physics

C3NT workshop: Quantum Information Science in High Energy Nuclear Physics
(QIS-HENP), June 15-19, 2026

Hideki Okawa

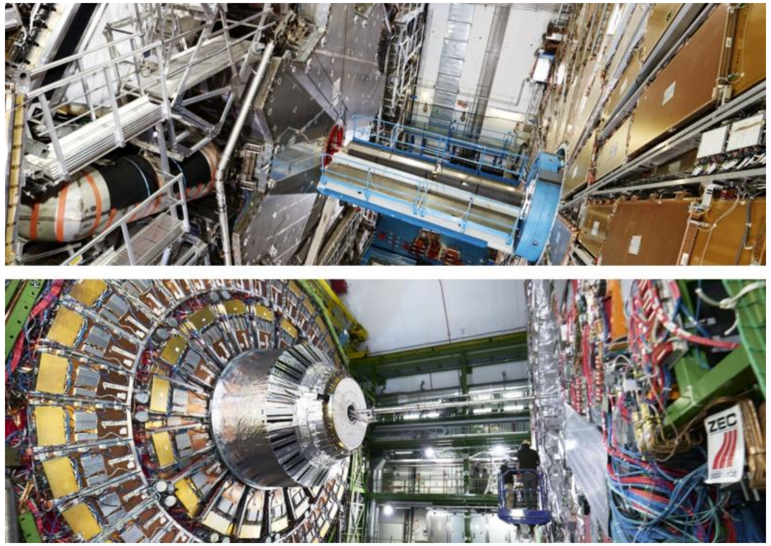
Institute of High Energy Physics, Chinese Academy of Sciences

Picture credits: Qianhui Shi, Yan Liang

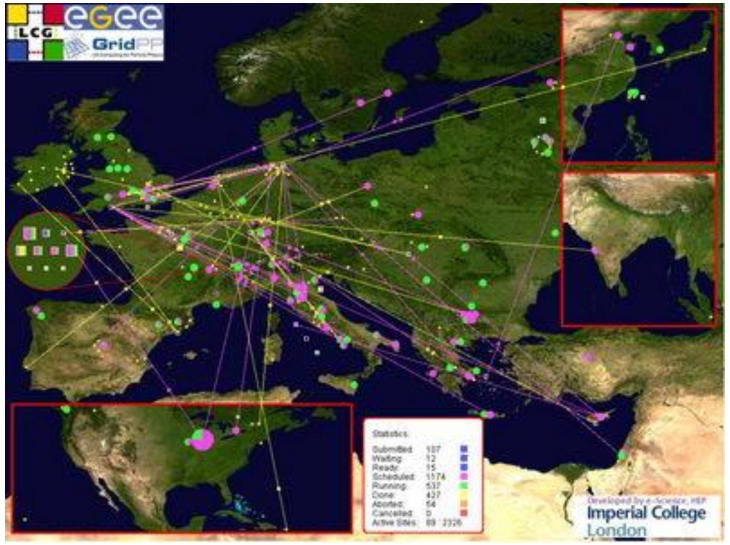
HEP Experiments → Big Data Science

Large Hadron Collider (LHC)

Large Scale Detectors

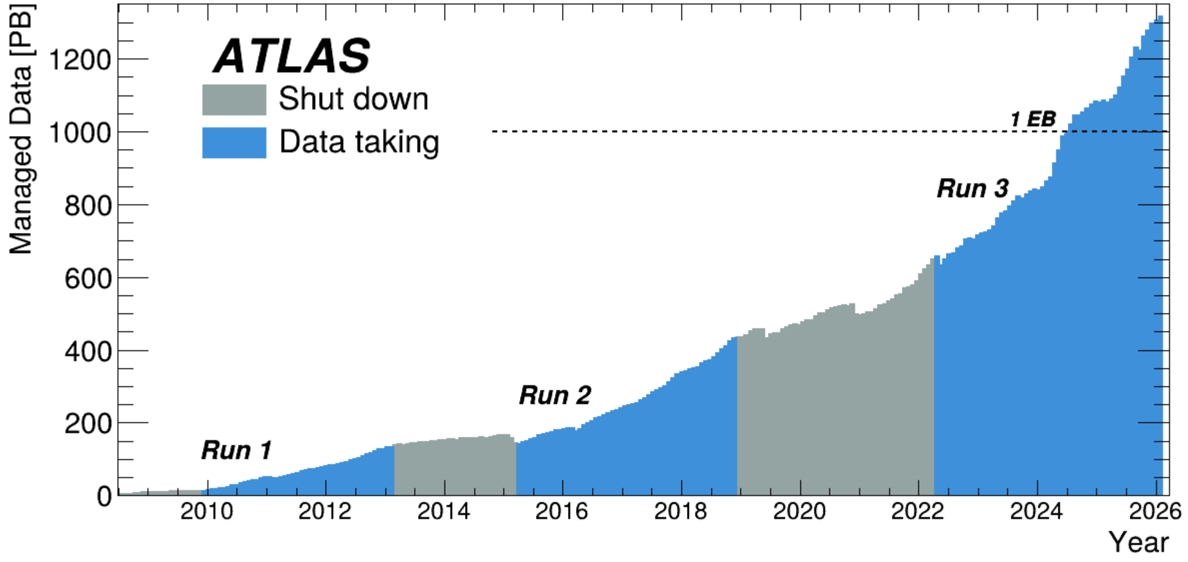


HPC & Worldwide Grid Computing

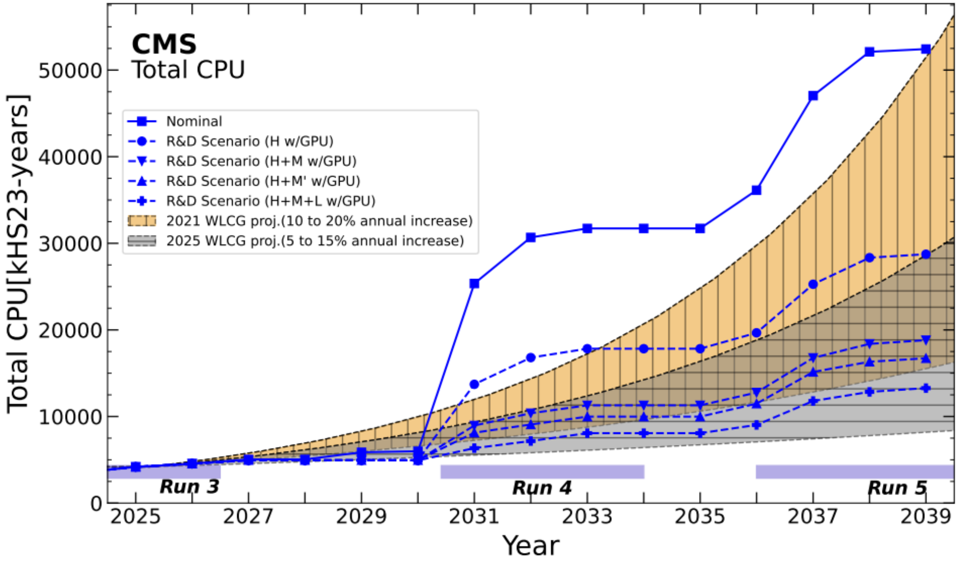


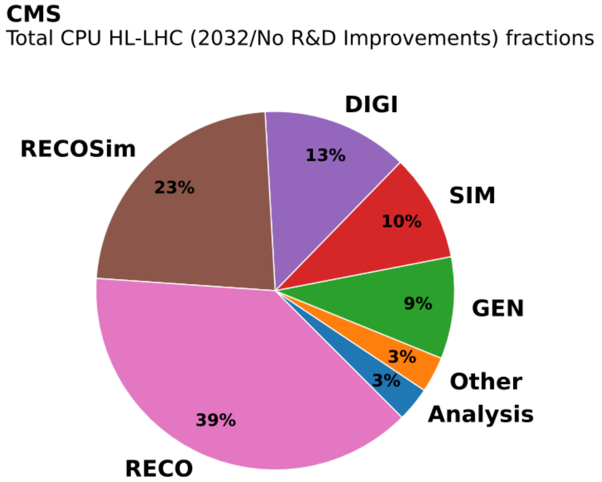
- **Experimental high-energy particle/nuclear physics is a big data science.**
- It uses cutting-edge detector/computing technologies & analyzes one of most complicated high-dimensional datasets in science.

Why Quantum Computing for HEP?



CMS, HL-LHC Software/Computing CDR, CERN-LHCC-2026-003 (2026)

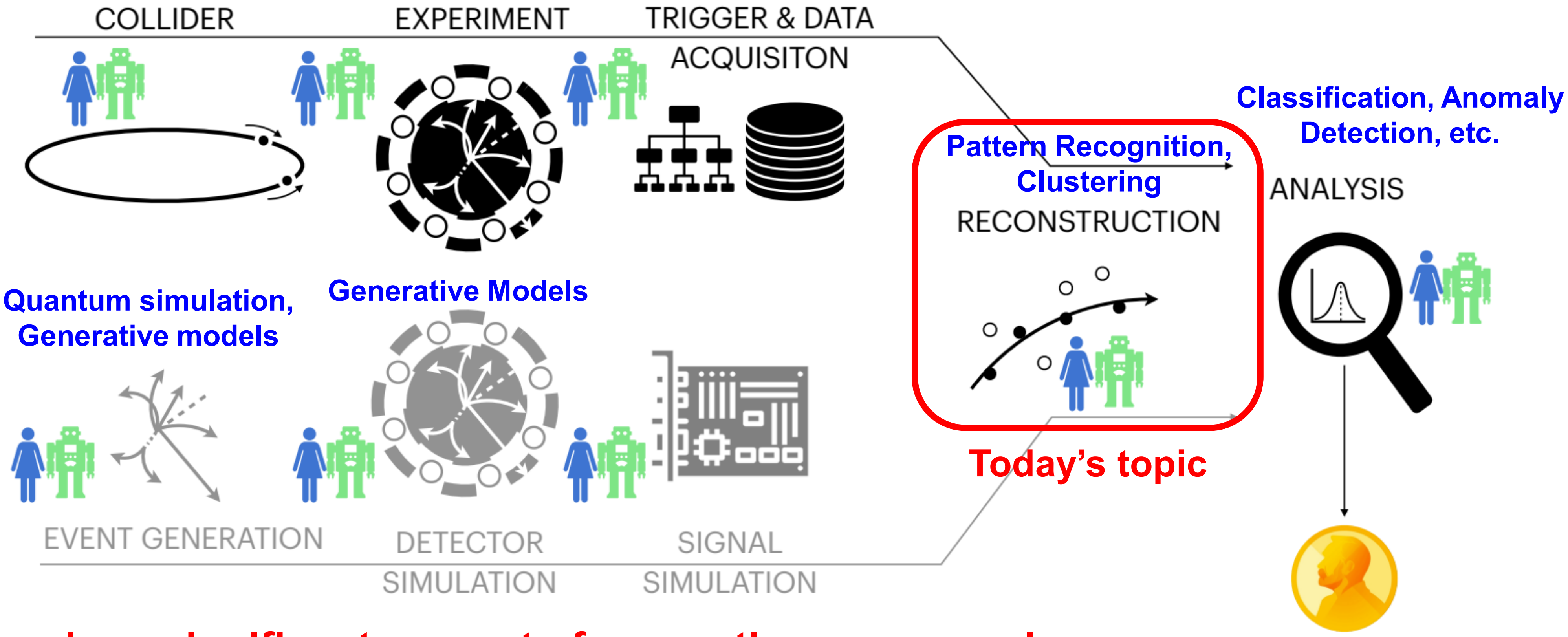




- **LHC has already entered the exabyte era!**
- Computing demands will continue to increase during the High-Luminosity LHC era (>2030).
- Other future colliders will face similar challenges (CEPC, FCC-ee, EIC, etc.)
- **CPU demands for reconstruction is particularly enormous.**

Experimental Workflow at Colliders

Credits: Andreas Salzburger



Experimental Data

Monte Carlo simulation

COLLIDER

EXPERIMENT

TRIGGER & DATA ACQUISITON

Quantum simulation, Generative models

Generative Models

EVENT GENERATION

DETECTOR SIMULATION

SIGNAL SIMULATION

Pattern Recognition, Clustering

RECONSTRUCTION

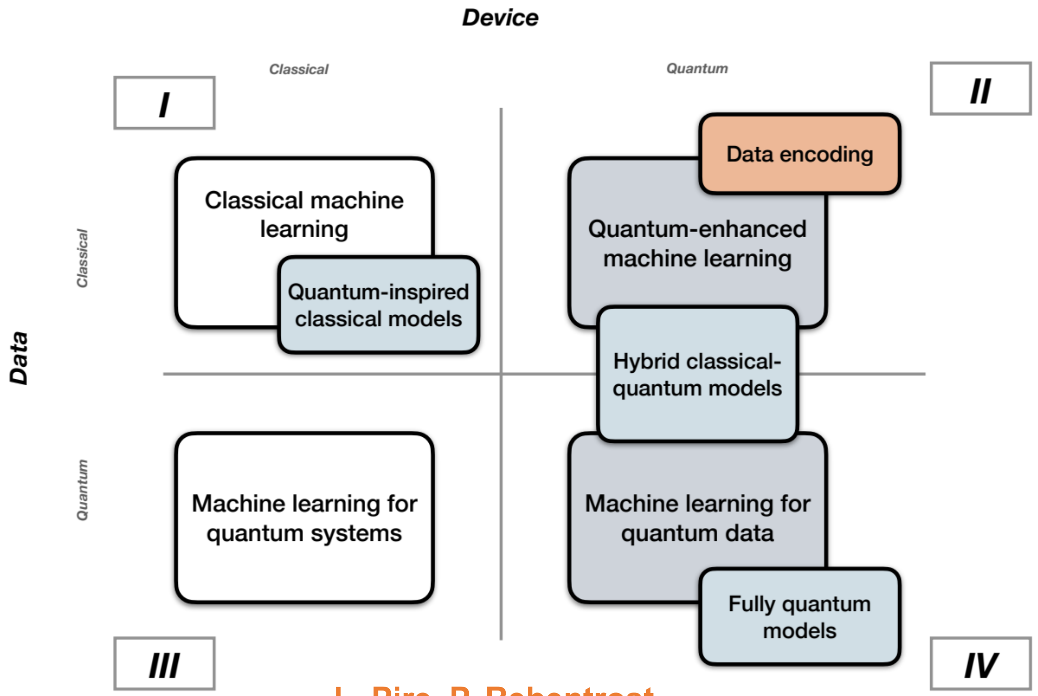
Today's topic

Classification, Anomaly Detection, etc.

ANALYSIS

- All steps require a significant amount of computing resources!
- State-of-the-art AI is currently being introduced/considered at all levels.
- Can we introduce quantum computing to these tasks?

Quantum Machine Learning & HEP



Device

Classical

Quantum

Data

Classical

Quantum

I

II

III

IV

Classical machine learning

Quantum-inspired classical models

Data encoding

Quantum-enhanced machine learning

Hybrid classical-quantum models

Machine learning for quantum systems

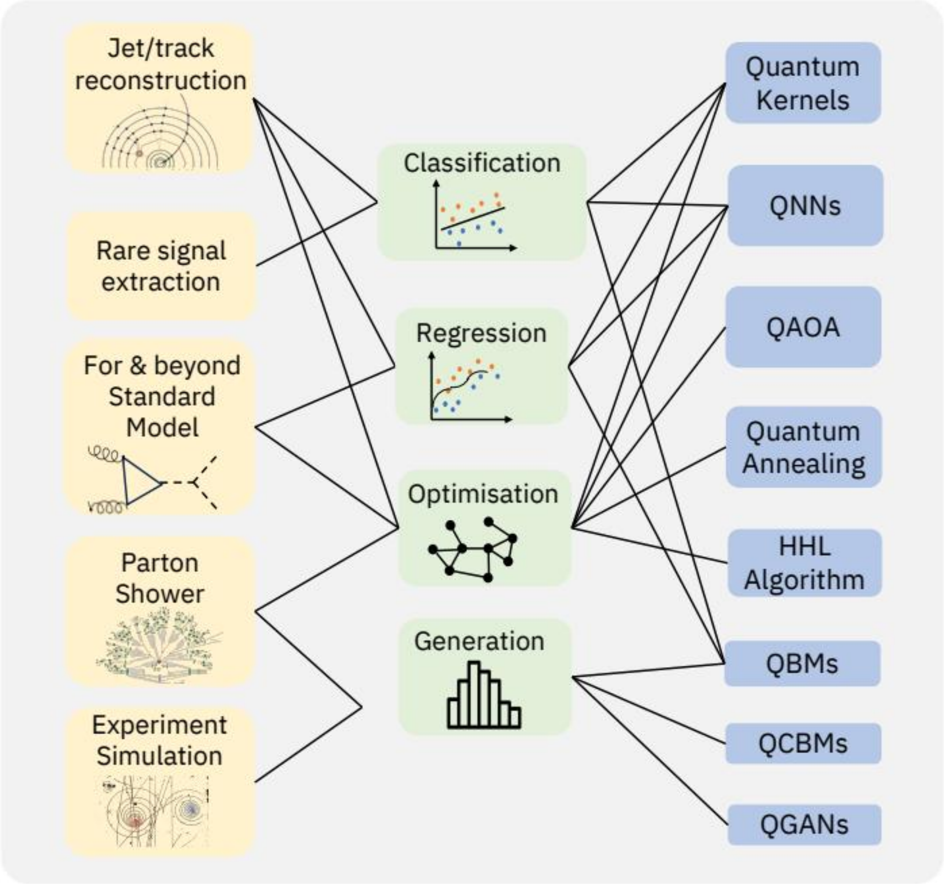
Machine learning for quantum data

Fully quantum models

L. Pire, P. Rebentrost

- Applications of quantum machine learning have been investigated for all components of experimental workflow with various quantum technologies (i.e. quantum gates, quantum annealing, quantum-inspired, etc.)

Experiment



Jet/track reconstruction

Rare signal extraction

For & beyond Standard Model

Parton Shower

Experiment Simulation

Classification

Regression

Optimisation

Generation

Quantum Kernels

QNNs

QAOA

Quantum Annealing

HHL Algorithm

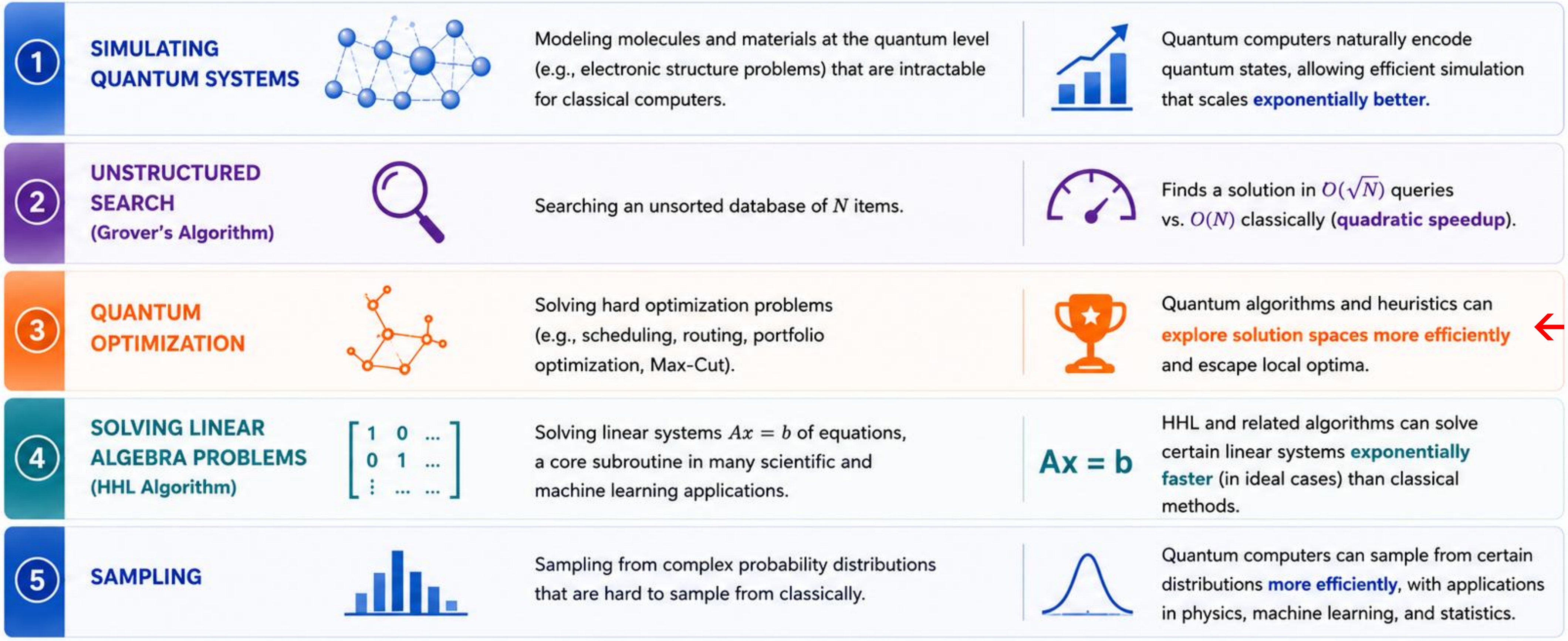
QBMs

QCBMs

QGANs

A. Di Meglio et al., PRX QUANTUM 5, 037001 (2024)

What is Quantum Computing Good At?



1	**SIMULATING QUANTUM SYSTEMS**	Modeling molecules and materials at the quantum level (e.g., electronic structure problems) that are intractable for classical computers.	Quantum computers naturally encode quantum states, allowing efficient simulation that scales **exponentially better.**
2	**UNSTRUCTURED SEARCH (Grover's Algorithm)**	Searching an unsorted database of N items.	Finds a solution in $O(\sqrt{N})$ queries vs. $O(N)$ classically (**quadratic speedup**).
3	**QUANTUM OPTIMIZATION**	Solving hard optimization problems (e.g., scheduling, routing, portfolio optimization, Max-Cut).	Quantum algorithms and heuristics can **explore solution spaces more efficiently** and escape local optima.
4	**SOLVING LINEAR ALGEBRA PROBLEMS (HHL Algorithm)**	Solving linear systems $Ax = b$ of equations, a core subroutine in many scientific and machine learning applications.	**Ax = b** HHL and related algorithms can solve certain linear systems **exponentially faster** (in ideal cases) than classical methods.
5	**SAMPLING**	Sampling from complex probability distributions that are hard to sample from classically.	Quantum computers can sample from certain distributions **more efficiently**, with applications in physics, machine learning, and statistics.

← **My Talk**

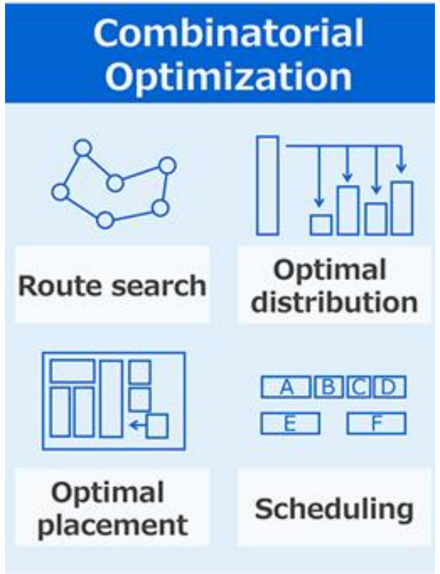
Excluding items irrelevant to HEP

- I will mostly be talking about applications using optimization algorithms.

Ising: ±1 spins, QUBO: 0,1 binaries

Optimization as Ising/QUBO Problems

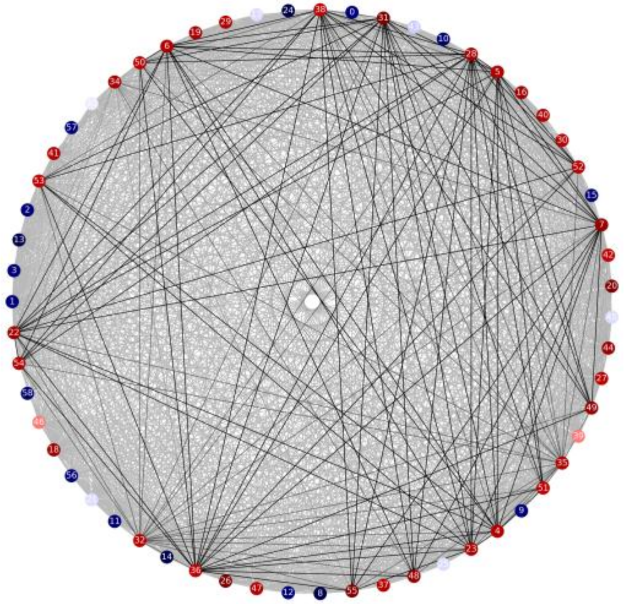
Specific Problem



©TOSHIBA

Map them to an Ising/QUBO Hamiltonian

N spins = N qubits

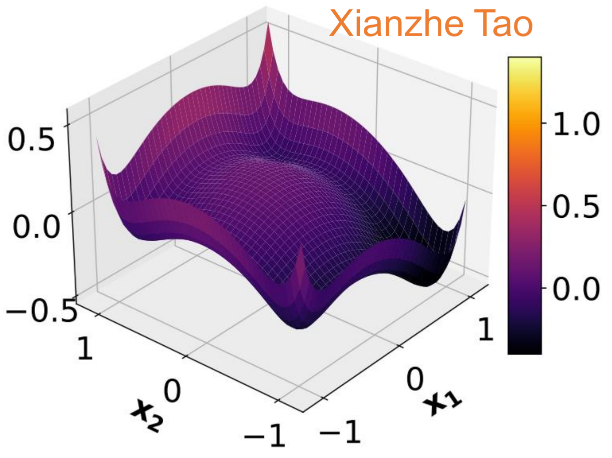


SA Abel, LA Nutricati, arXiv:2206.09956

Search the Ground State

The best spin configuration from 2^N options

Answer



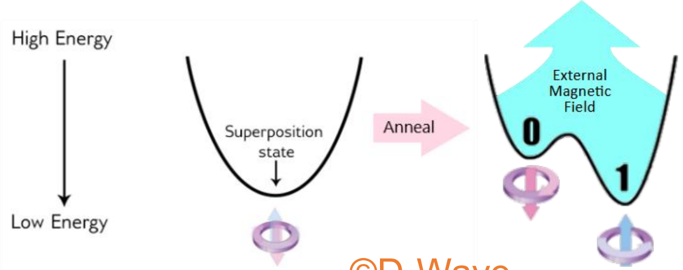
Xianzhe Tao

Schematic 2D Energy Landscape
→ practically N dimensional

- Many complicated problems can be formulated as Ising or QUBO Hamiltonian.
- **<u>We design the Ising/QUBO Hamiltonian so that the ground state corresponds to the answer.</u>**
- **Track & jet reconstruction can also be formulated as such problems (Typical numbers of qubits required: O($10^{5\text{-}6}$) [track], O($10^{2\text{-}3}$) [jet]).**

3 Classes of Ising Problem Solvers

Quantum Annealing (QA)

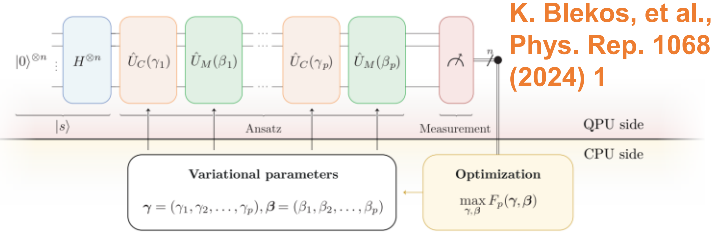


©D-Wave

$$H(s) = A(s)H_{\text{initial}} + B(s)H_{\text{target}}$$

- Hamiltonian is slowly modified from a symmetric initial form to the target Hamiltonian describing the Ising problem.
- Quantum adiabatic theorem supports the success of obtaining the ground state.
- **Quantum tunneling** helps avoid local minima.

Quantum Circuits



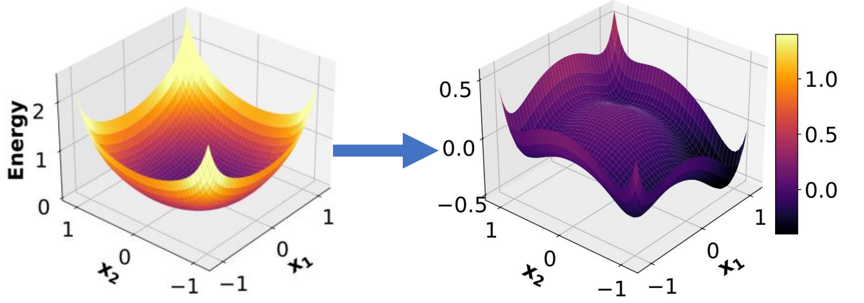
K. Blekos, et al., Phys. Rep. 1068 (2024) 1

$$U \approx \prod_{n=1}^{p} e^{-iH(n\Delta t)\Delta t} = \prod_{n=1}^{p} e^{-i(1-s(n\Delta t))H_{\text{initial}}\Delta t} e^{-is(n\Delta t)H_{\text{target}}\Delta t}$$
$$= \prod_{n=1}^{p} e^{-i\beta_n H_{\text{initial}}} e^{-i\gamma_n H_{\text{target}}} = \prod_{n=1}^{p} U_M(\beta_n)U_C(\gamma_n),$$

- **Quantum Approximate Optimization Algorithm (QAOA)** mimics quantum annealing with Trotterization.
- Imaginary Hamiltonian variational ansatz (iHVA) & Imaginary Time Evolution-Mimicking Circuit (ITEMC) overcome some known bottlenecks of QAOA.

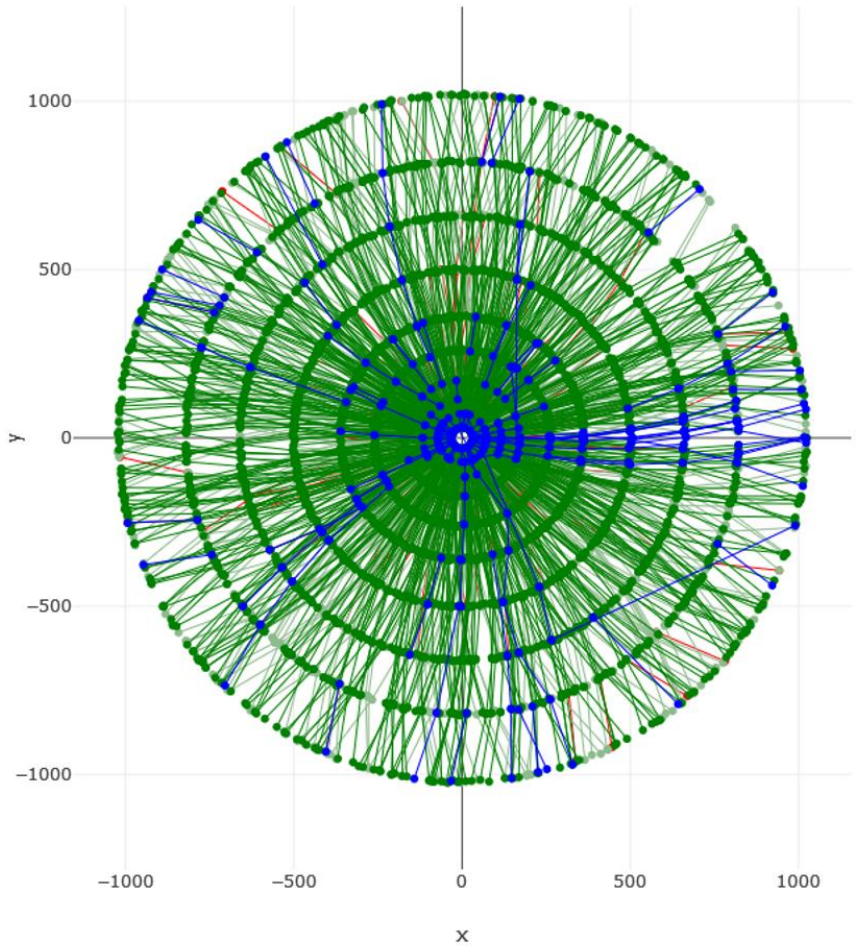
Quantum-Inspired

Xianzhe Tao



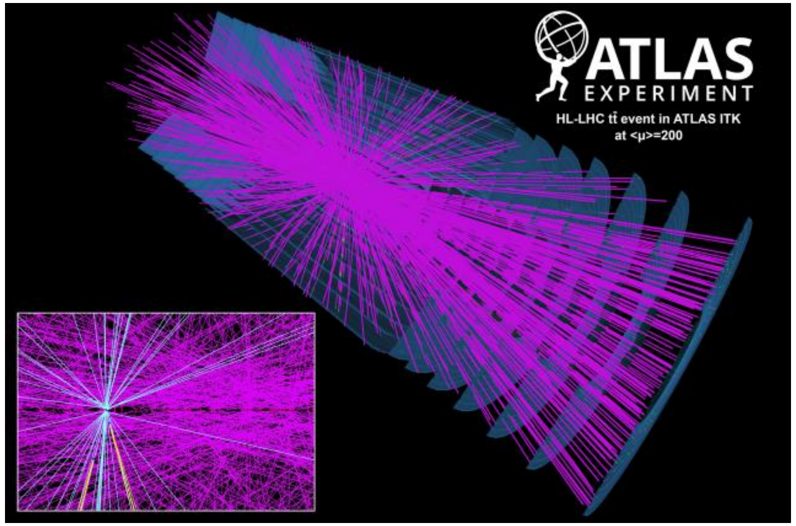
- Quantum-inspired runs on classical hardware with algorithms inspired by QC.
- **Simulated annealing (SA)** uses random moves in the solution space to search for the ground state.
- **Simulated bifurcation (SB) emulates quantum adiabatic evolution of Kerr-nonlinear parametric oscillators. It significantly outperforms SA.**

Track Reconstruction

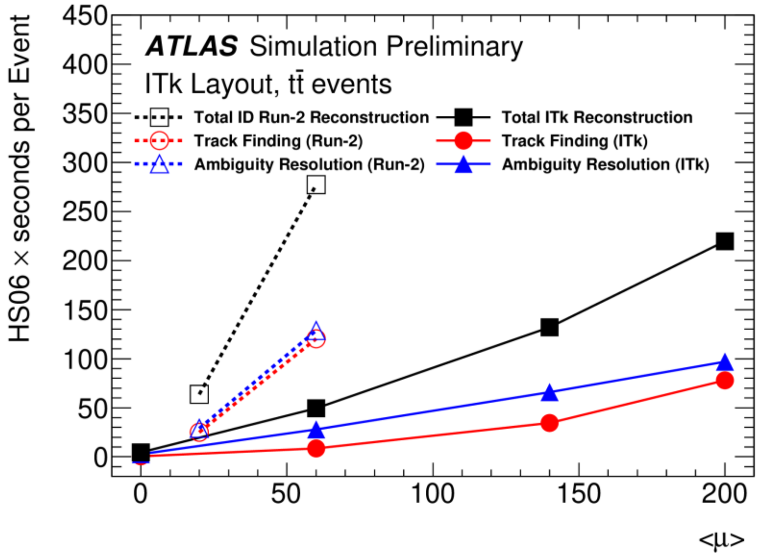


Reconstructed by a quantum-annealing-inspired algorithm

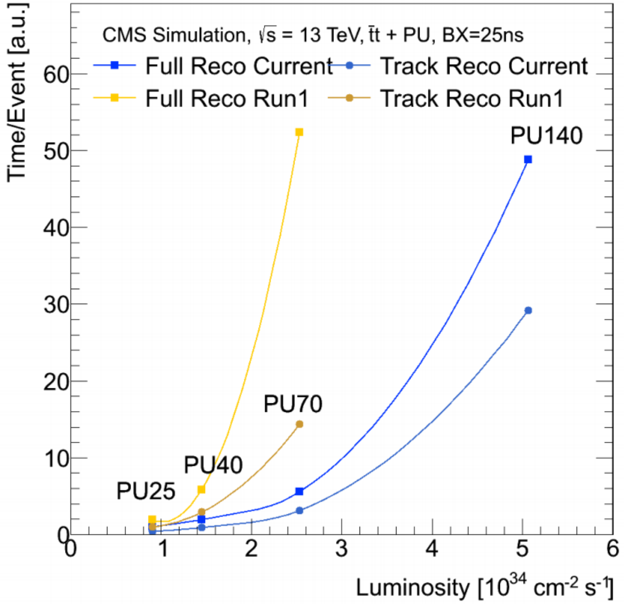
Track Reconstruction at LHC & HL-LHC



ATL-PHYS-PUB-2019-041



https://cds.cern.ch/record/1966040

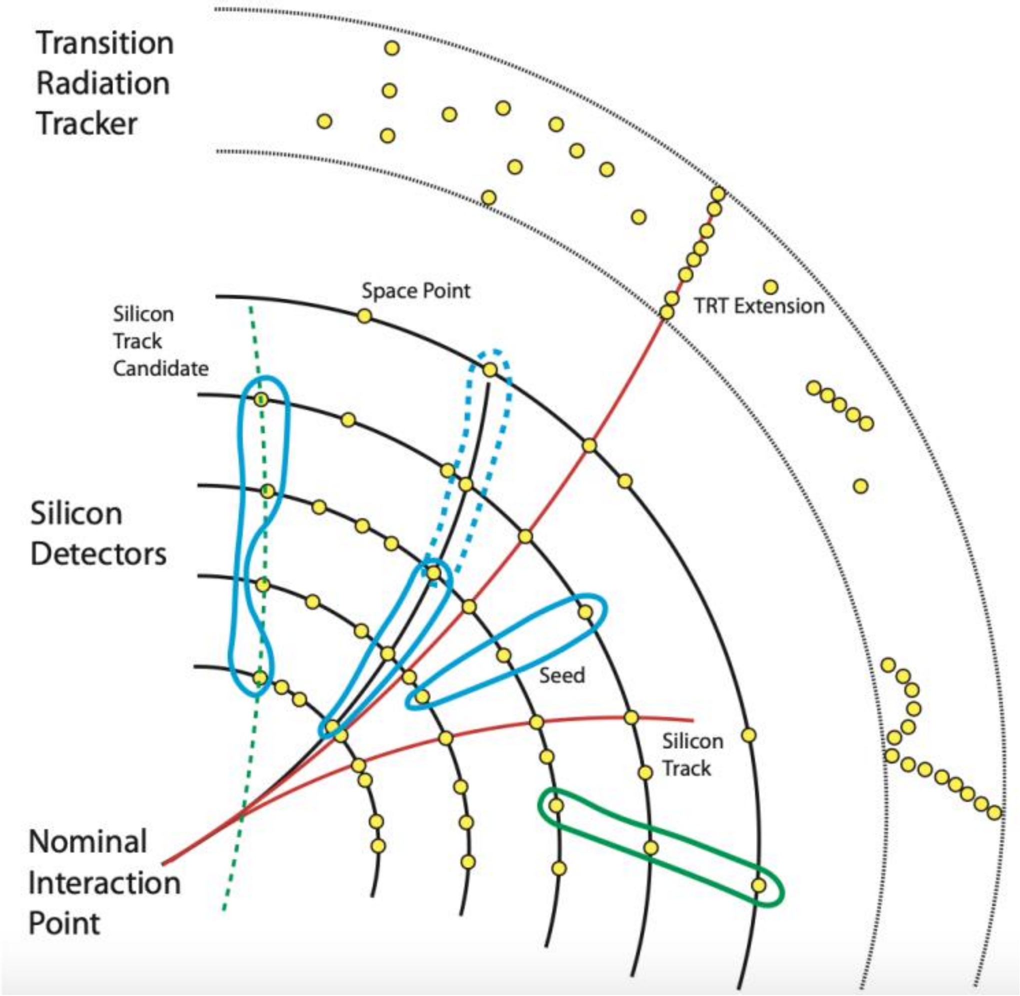


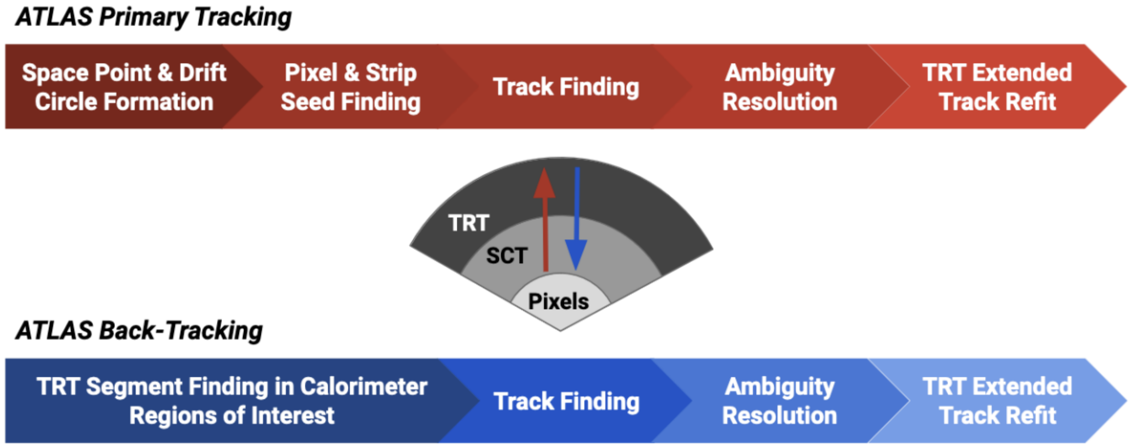
	Run 1	Run 2	HL-LHC
μ	21	40	150-200
Tracks	~280	~600	**~7-10k**

- At the HL-LHC, additional interactions per bunch crossing becomes exceedingly high & **CPU time increases exponentially with more pileup.**
- GPU & ML-based approaches are actively investigated, but quantum ML may play an important role.

Credits: Christian Grefe

Traditional Tracking in ATLAS





ATLAS Primary Tracking

Space Point & Drift Circle Formation → Pixel & Strip Seed Finding → Track Finding → Ambiguity Resolution → TRT Extended Track Refit

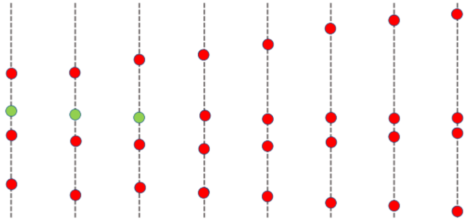
ATLAS Back-Tracking

TRT Segment Finding in Calorimeter Regions of Interest → Track Finding → Ambiguity Resolution → TRT Extended Track Refit

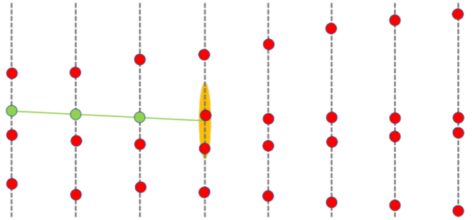
- **Find all valid combinations of segments with 3 silicon detector hits (triplets)**: p_T > 0.5 GeV, d_0 < 5 mm, z_0 < 200 mm
- Seeds are processed depending on their score: high-p_T, low-d_0 and a compatible hit in a 4th confirmation layer are preferred
- Combinatorial Kalman Filter is used to extend seeds inwards and outwards
- Confirmed Si-only tracks are extended into the TRT
- Global χ^2-fit is used to extract the final track parameters

D. Magano et al., PRD 105 (2022) 7, 076012
C Brown et al., Front. Artif. Intell. 7 (2024) 1339785

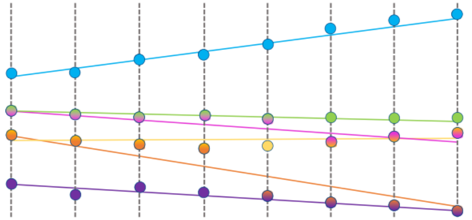
Track Reconstruction (Iterative)



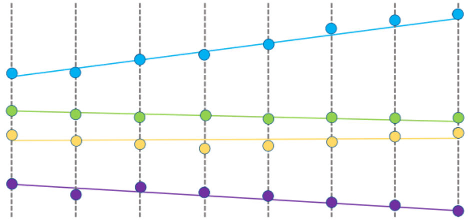
(a) Seeding



(b) Track building

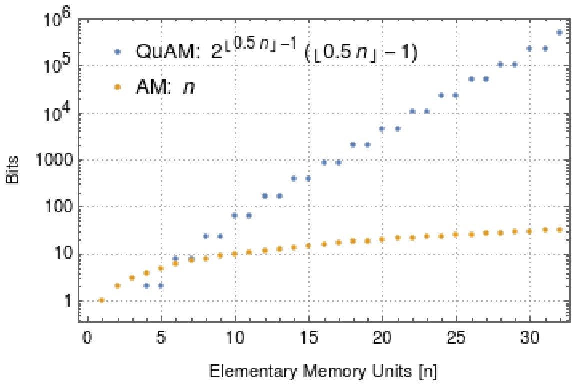


(c) Cleaning

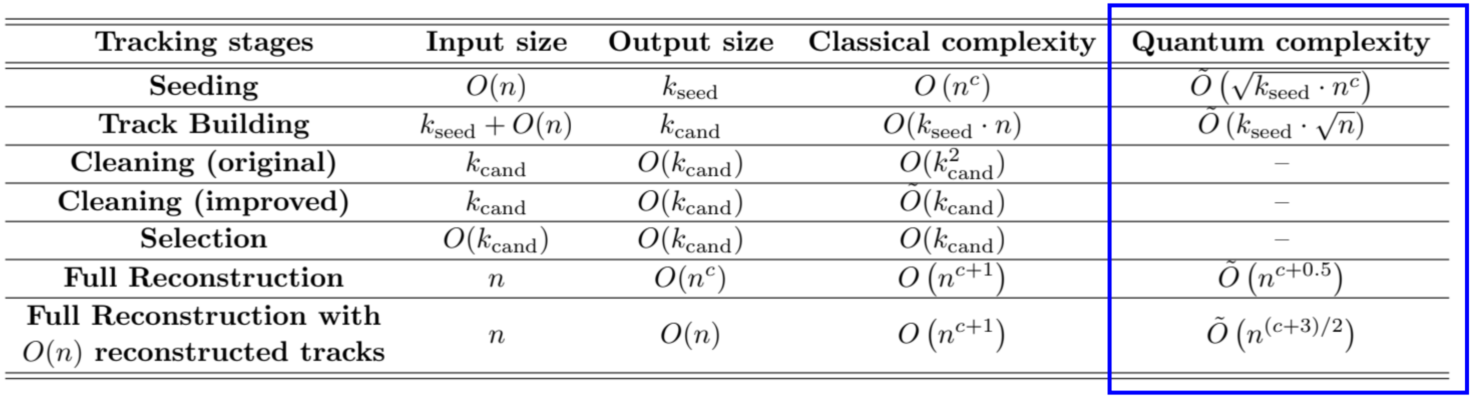


(d) Selection

- **Iterative reconstruction requires quantum associative memory (QuAM; not yet available)**
- Conceptual complexity analyses were performed for **4 stages of reconstruction**.
- **Grover search reduces complexity of seeding & track building in square root.**
- **However, expected overall improvement is rather mild.**



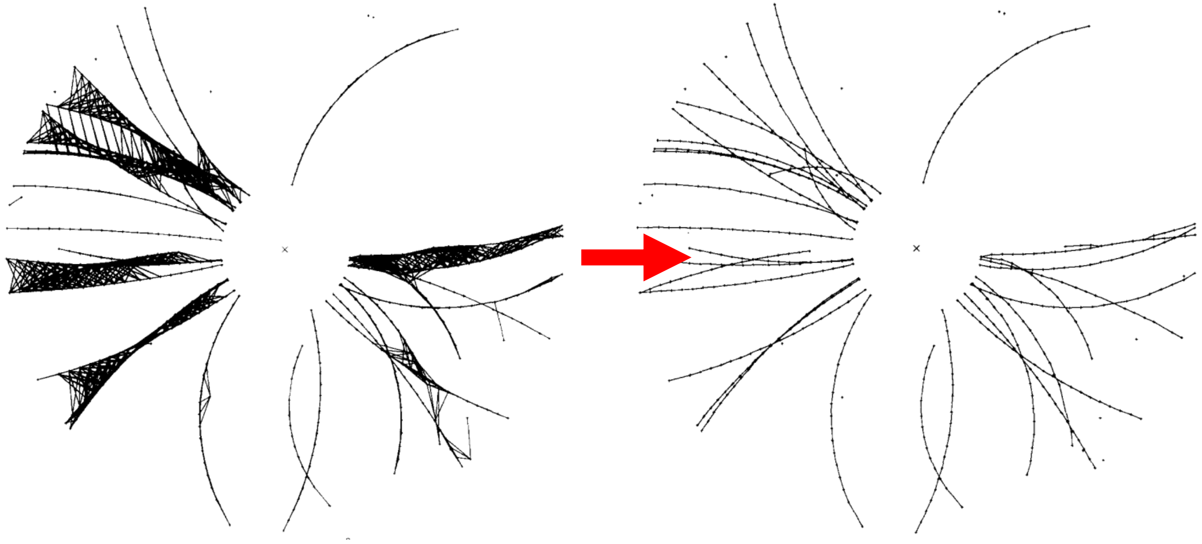
I. Shapoval, P. Calfiura, EPJ Web of Conf. 214, 01012 (2019)

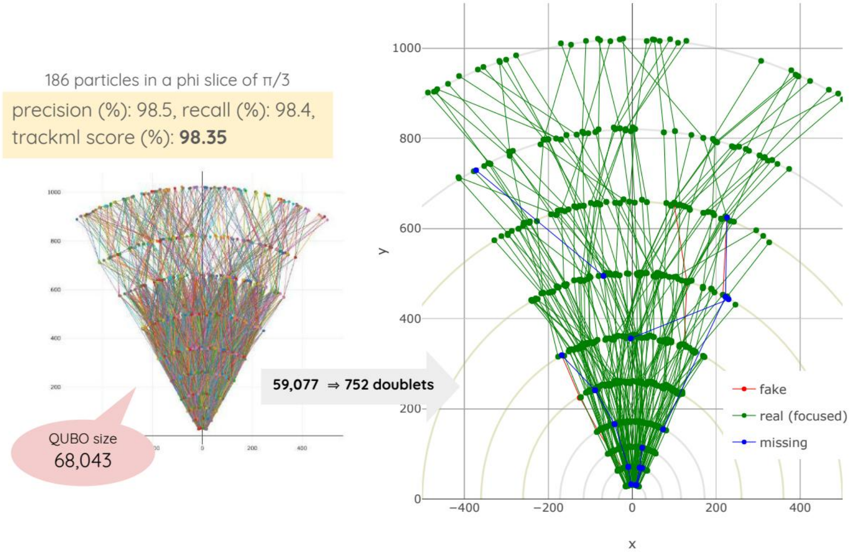


Tracking stages	Input size	Output size	Classical complexity	Quantum complexity
Seeding	$O(n)$	k_{seed}	$O\left(n^c\right)$	$\tilde{O}\left(\sqrt{k_{\text{seed}}} \cdot n^c\right)$
Track Building	$k_{\text{seed}} + O(n)$	k_{cand}	$O(k_{\text{seed}} \cdot n)$	$\tilde{O}\left(k_{\text{seed}} \cdot \sqrt{n}\right)$
Cleaning (original)	k_{cand}	$O(k_{\text{cand}})$	$O(k_{\text{cand}}^2)$	–
Cleaning (improved)	k_{cand}	$O(k_{\text{cand}})$	$\tilde{O}(k_{\text{cand}})$	–
Selection	$O(k_{\text{cand}})$	$O(k_{\text{cand}})$	$O(k_{\text{cand}})$	–
Full Reconstruction	n	$O(n^c)$	$O\left(n^{c+1}\right)$	$\tilde{O}\left(n^{c+0.5}\right)$
Full Reconstruction with $O(n)$ reconstructed tracks	n	$O(n)$	$O\left(n^{c+1}\right)$	$\tilde{O}\left(n^{(c+3)/2}\right)$

Tracking as Optimization Problem

- **Tracking as an optimization problem: a global approach to reconstruct tracks in one-go.** (↔iterative approach: Combined Kalman Filter)
- **Stimple-Abele & Garrido (1990):** generate all potential doublets with some cuts applied & pursue a binary classification task (i.e. solve an Ising/QUBO problem) to determine which ones should be kept.
- **Modern quantum computing versions:** quantum annealers w/ doublets (A. Zlokapa et al.) & triplet-based (F. Bapst et al.) approaches; quantum gate machines (L. Funcke et al., etc.; **H.Okawa**)





186 particles in a phi slice of π/3

precision (%): 98.5, recall (%): 98.4, trackml score (%): **98.35**

QUBO size 68,043

59,077 ⇒ 752 doublets

QUBO: 0,1 binaries
Ising: ±1 spins

QUBO Formulation w/ Triplets

- Tracks are formed by connecting silicon detector hits: e.g. triplets (segments w/ 3 hits).
- Doublets/triplets are connected to reconstruct tracks & it can be regarded as a **quadratic unconstrained binary optimization (QUBO)/Ising** problem.

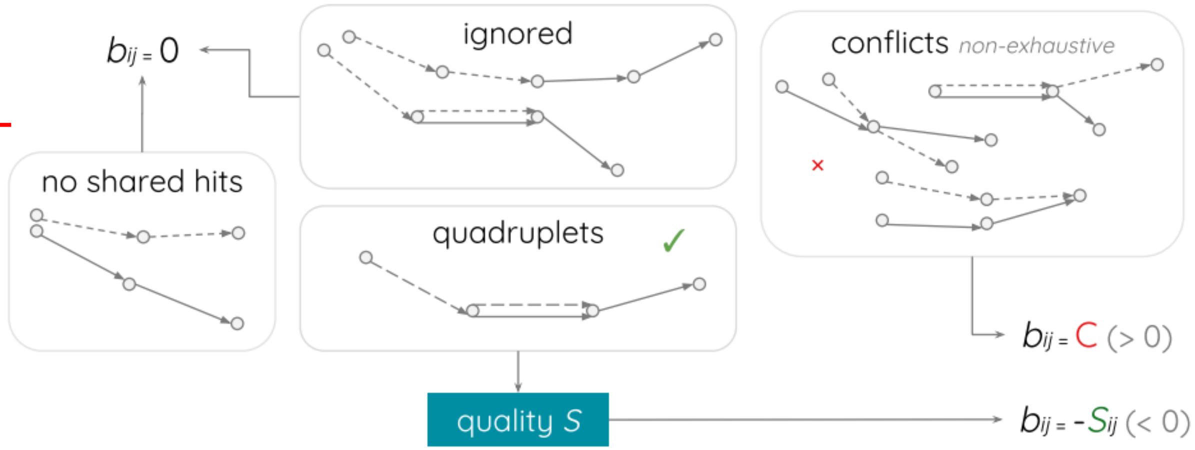
$$O(a,b,T) = \underbrace{\sum_{i=1}^{N} a_i T_i}_{\text{Quality of triplets}} + \underbrace{\sum_{i}^{N} \sum_{j<i}^{N} b_{ij} T_i T_j}_{\text{Compatibility b/w triplet pairs}}$$

$$a_i = \alpha\left(1 - e^{\frac{|d_0|}{\gamma}}\right) + \beta\left(1 - e^{\frac{|z_0|}{\lambda}}\right),$$

b_{ij} = 0 (if no shared hit), 1 (if conflict)
= $-S_{ij}$ (if two hits are shared)

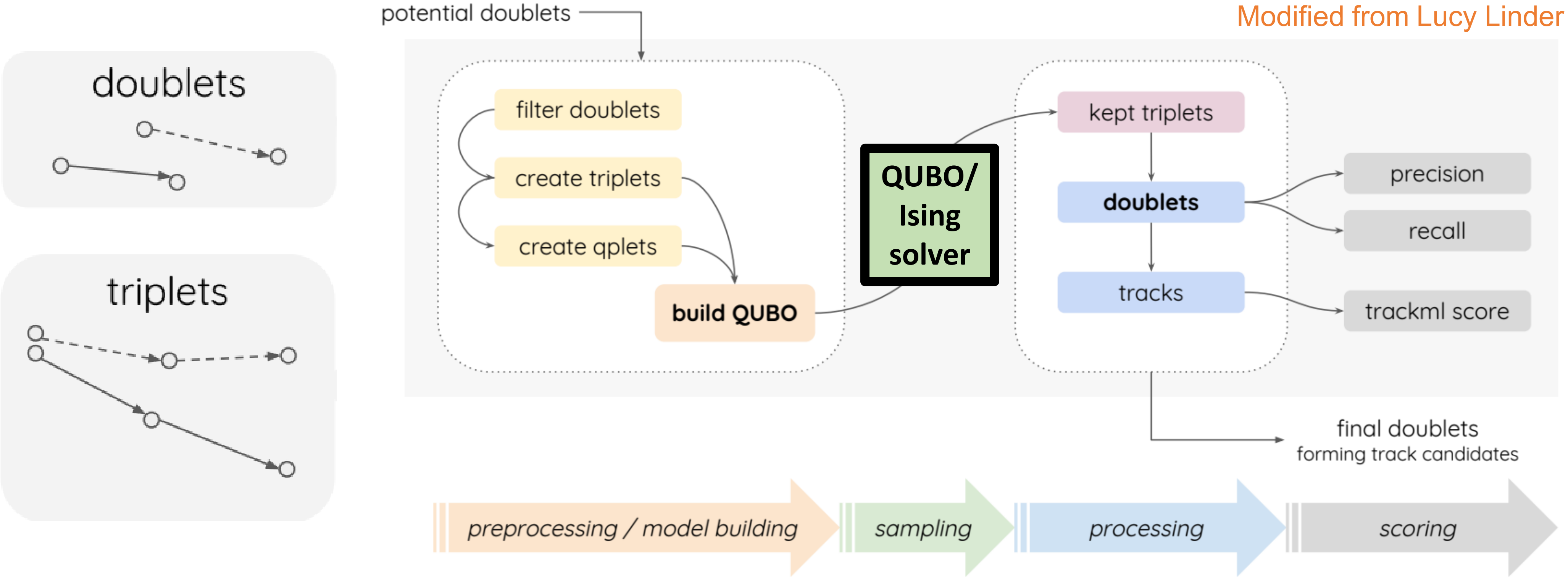
$$S_{ij} = \frac{1 - \frac{1}{2}(|\delta(q/p_{Ti}, q/p_{Tj})| + max(\delta\theta_i, \delta\theta_j))}{(1 + H_i + H_j)^2},$$

F. Bapst et al. Comp. Soft. Big Sci. 4 (2019) 1.



Minimizing QUBO is equivalent to searching for the ground state of the Hamiltonian.

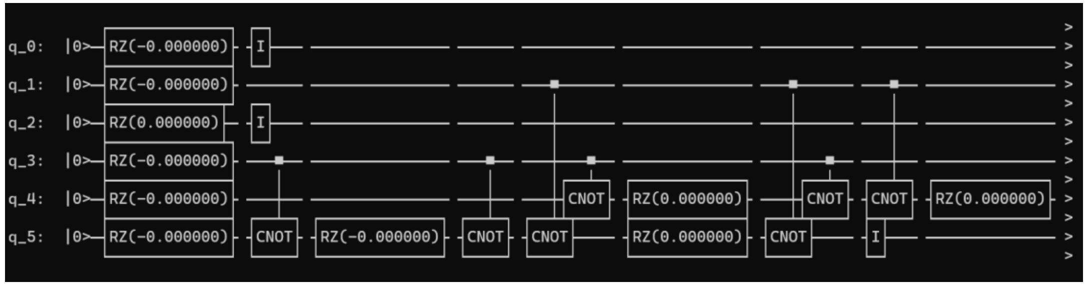
Algorithm Flow w/ QUBO



- We build QUBO on an event-by-event basis from the silicon detector hits.
- Predicted ground state will define which triplets should be kept (binary=1). Connecting the adopted triplets will give us the tracks.

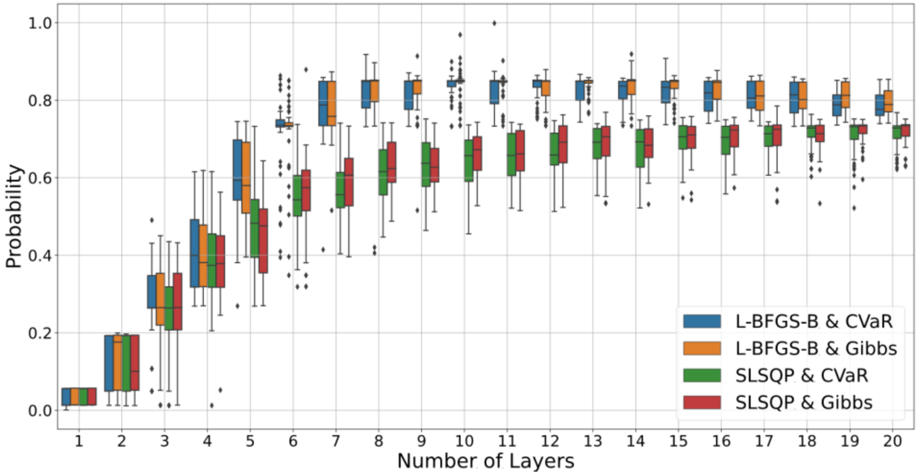
H. Okawa, Springer Communications in Computer and Information Science,
2036 (2024) 272–283, arXiv:2310.10255

1. Quantum Gate Approach

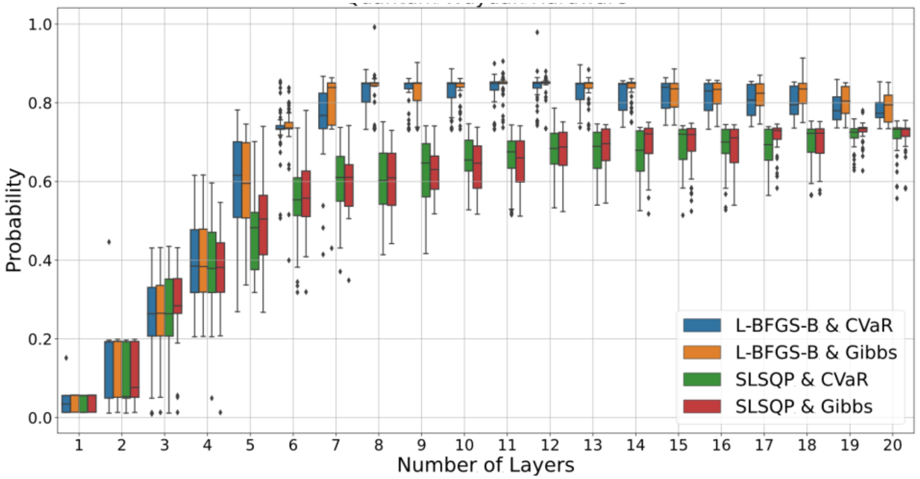


- Used QAOA in pyqpanda-algorithm by Origin Quantum. **QAOA is resilient to noise.**
- # of layers, loss function & optimizer are optimized.
- Used quantum simulator as well as 6-qubit Origin Quantum hardware (Wuyuan).
- Accuracy reaches ~100% for >=4 layers (as multiple measurements are pursued)
- No sign of performance degradation in hardware for <=20 layers

Simulator

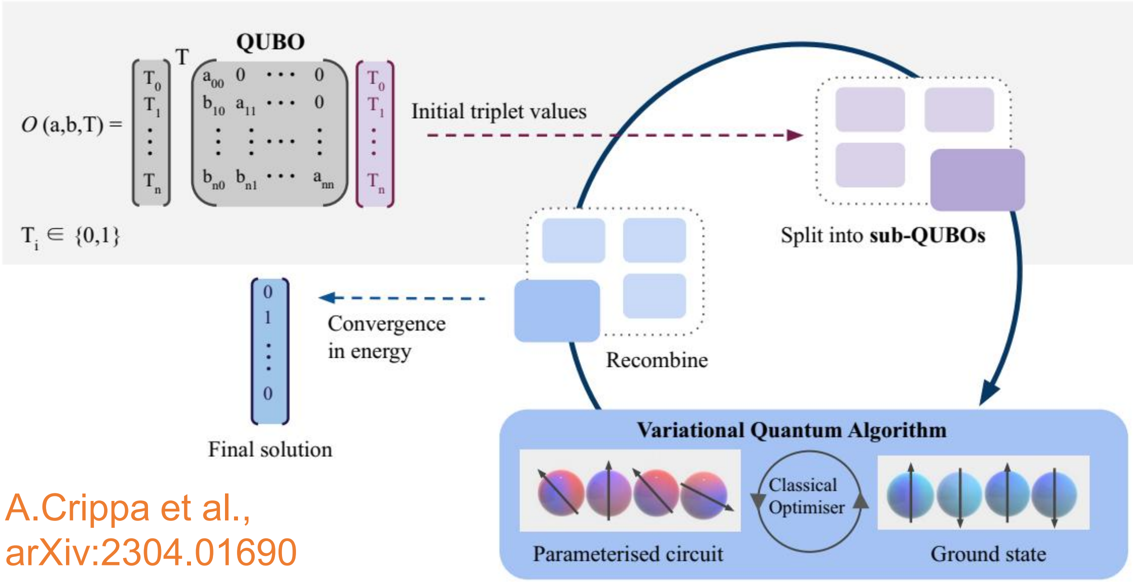


Wuyuan Hardware

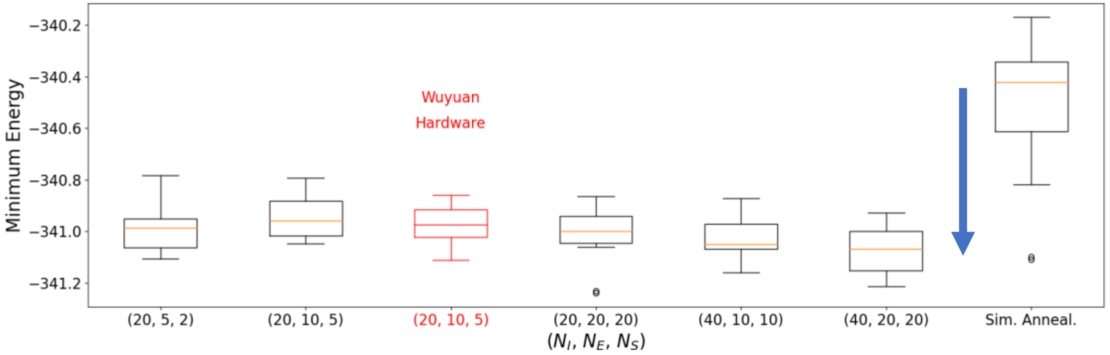


H. Okawa, Springer Communications in Computer and Information Science, 2036 (2024) 272–283, arXiv:2310.10255

Sub-QUBO Method

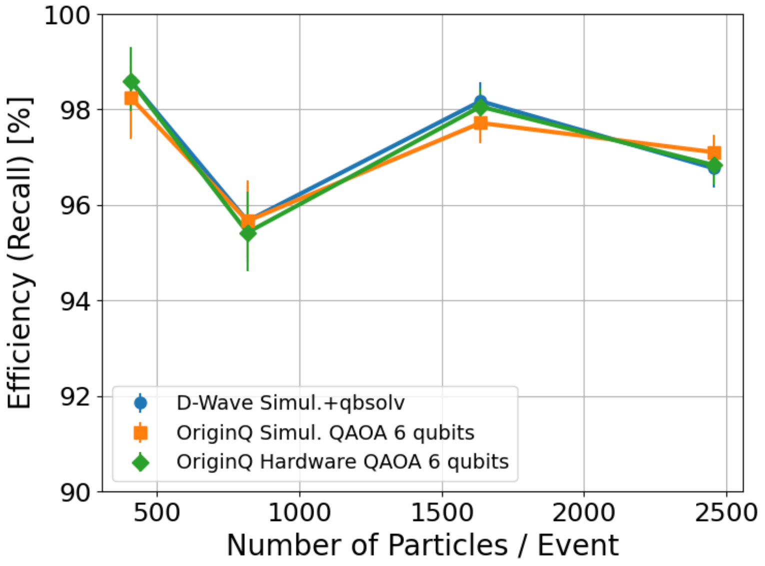


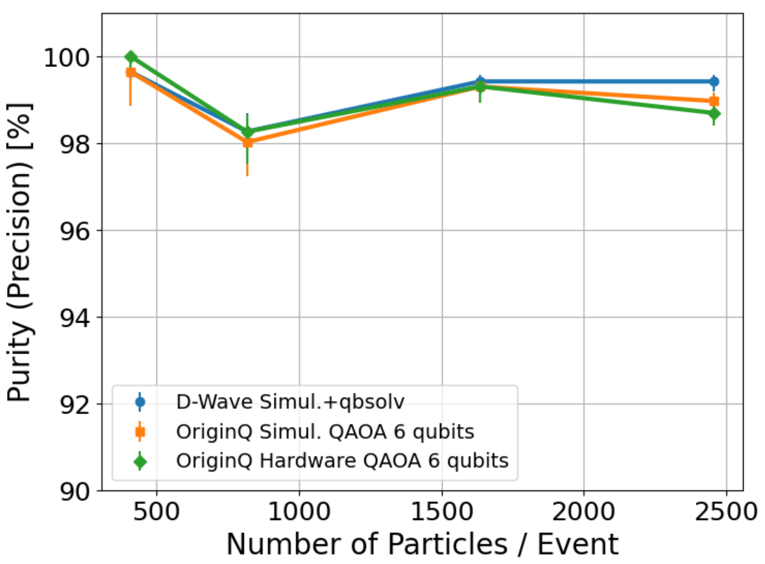
A.Crippa et al., arXiv:2304.01690

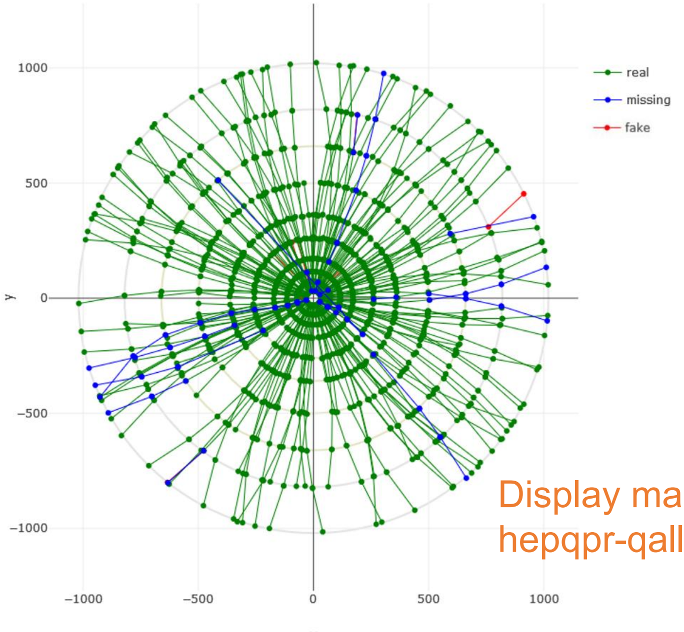


- Number of triplet candidates determines number of qubits required → **~O(10^5x10^5)**
  HL-LHC conditions do not fit into the current scale of quantum annealing & gate computers
- QUBO is split into sub-QUBOs of size 6x6 to match with Origin Quantum hardware.
- **Theoretically-robust sub-QUBO method using multiple solution instances** is adopted (Y. Atobe, M. Tawada, N. Togawa, IEEE Trans. Comp. 71, 10 (2022) 2606) → known to outperform empirical methods like qbsolv

WIP: Triplet Efficiency & Purity







Display made w/ hepqpr-qallse

$$\text{Efficiency} = \frac{TP}{TP+FN} = \frac{\#\text{ of matched reconstructed doublets}}{\#\text{ of true doublets}},$$

$$\text{Purity} = \frac{TP}{TP+FP} = \frac{\#\text{ of matched reconstructed doublets}}{\#\text{ of all reconstructed doublets}},$$

- QAOA+sub-QUBO provides **compatible performance as the previous quantum annealing studies.**
- **No sign of degradation in the real hardware**
- **This is the 1st tracking w/ QAOA, theoretically robust sub-QUBO & Chinese quantum computer**
- However, usage of sub-QUBO degrades computing speed by a few orders of magnitude.

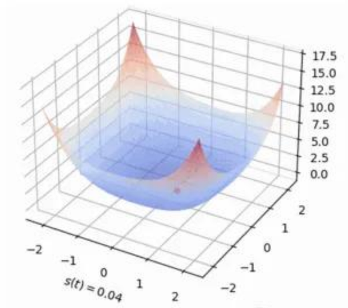
Goto et al., Sci. Adv. 2019; 5: eaav2372
Goto et al., Sci. Adv. 2021; 7: eabe7953

Simulated Bifurcation (SB)

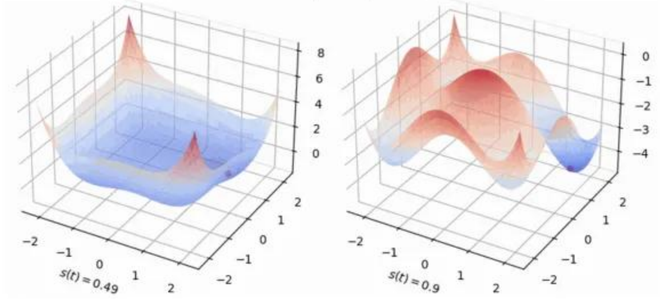
$$H_{SB}(\mathbf{x},\mathbf{y},t) = \sum_{i=1}^{N} \frac{\Delta}{2} y_i^2 + \sum_{i=1}^{N} [\frac{K}{4} x_i^4 + \frac{\Delta - p(t)}{2} x_i^2] - \frac{\xi_0}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij} x_i x_j$$

$$\dot{x}_i = \frac{\partial H_{\mathrm{SB}}}{\partial y_i} = \Delta y_i$$

$$\dot{y}_i = -\frac{\partial H_{\mathrm{SB}}}{\partial x_i} = -[K x_i^2 - p(t) + \Delta] x_i + \xi_0 \sum_{j=1}^{N} J_{i,j} x_j$$



陶贤哲



- “Quantum-annealing-inspired” algorithms search for ground state through the **classical time evolution of differential equations.**
- **Simulated bifurcation (SB) emulates quantum adiabatic evolution of Kerr-nonlinear parametric oscillators, exhibiting bifurcation phenomena.**
- Three variants exist depending on how one handles the continuous treatment of the spins (x_i): aSB, bSB, dSB

$$V_{\mathrm{aSB}} = \sum_{i=1}^{N} \left(\frac{x_i^4}{4} + \frac{a_0 - a(t)}{2} x_i^2 \right) - \frac{c_0}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij} x_i x_j$$

adiabatic SB (aSB; original)

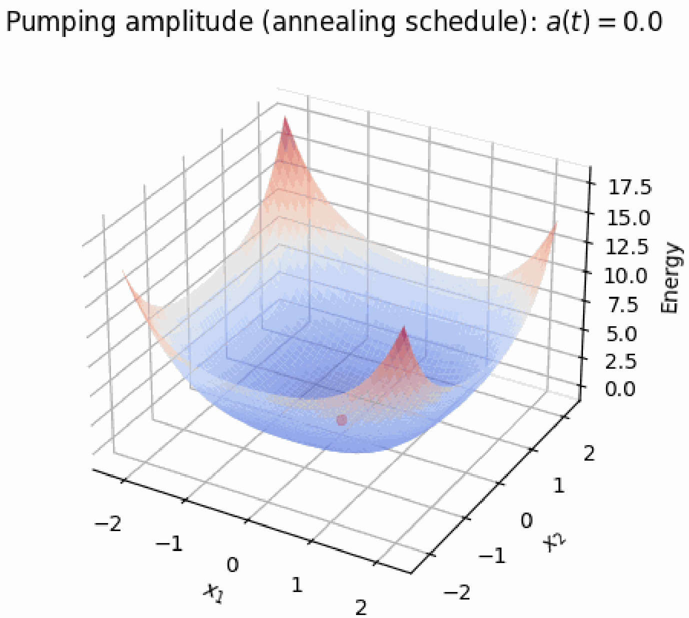
$$V_{\mathrm{bSB}} = \begin{cases} \frac{a_0 - a(t)}{2} \sum_{i=1}^{N} x_i^2 - \frac{c_0}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij} x_i x_j, & \text{if } |x_i| \le 1, \forall i \\ \infty, & \text{otherwise.} \end{cases}$$

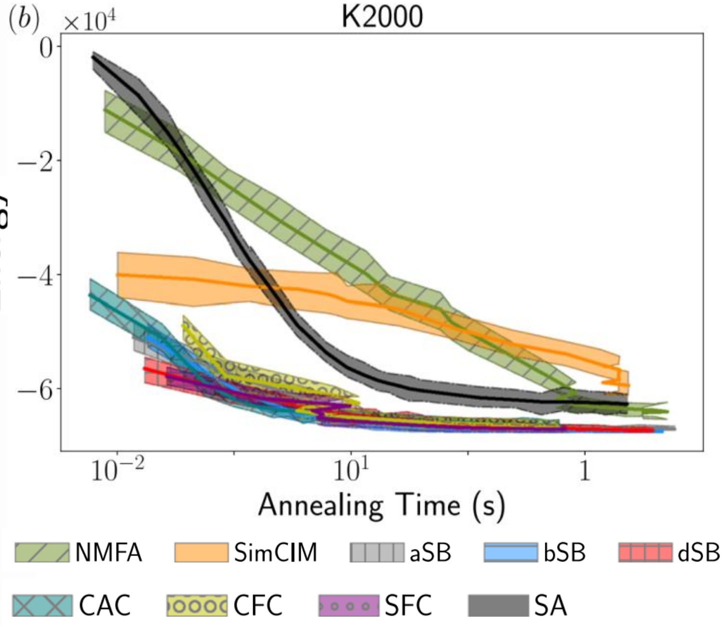
ballistic SB (bSB)

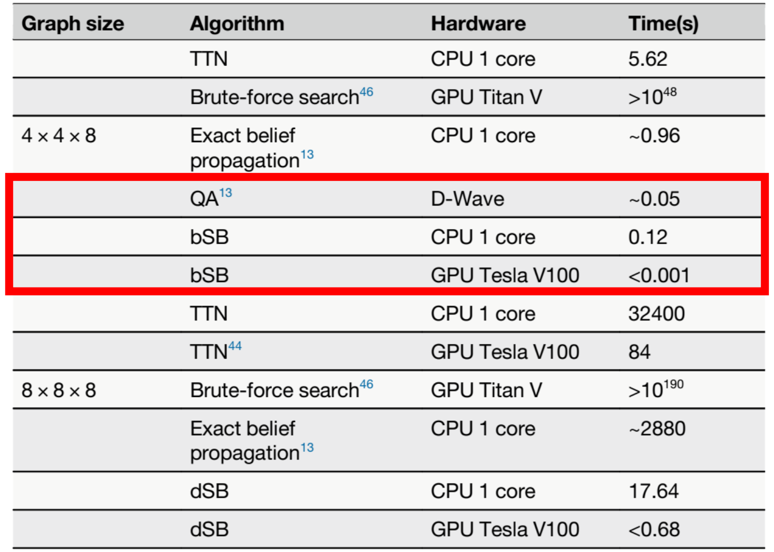
$$V_{\mathrm{dSB}} = \begin{cases} \frac{a_0 - a(t)}{2} \sum_{i=1}^{N} x_i^2 - \frac{c_0}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij} x_i \operatorname{sgn}(x_j), & \text{if } |x_i| \le 1, \forall i \\ \infty, & \text{otherwise.} \end{cases}$$

discrete SB (dSB)

Simulated Bifurcation (SB)





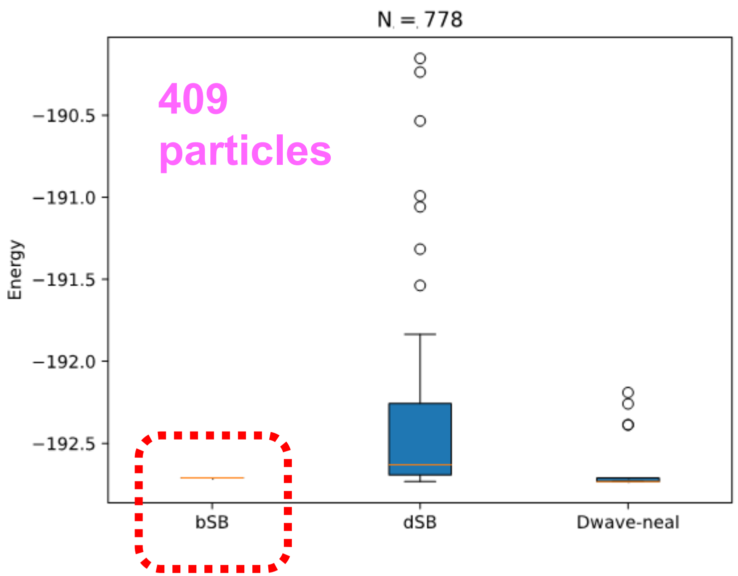


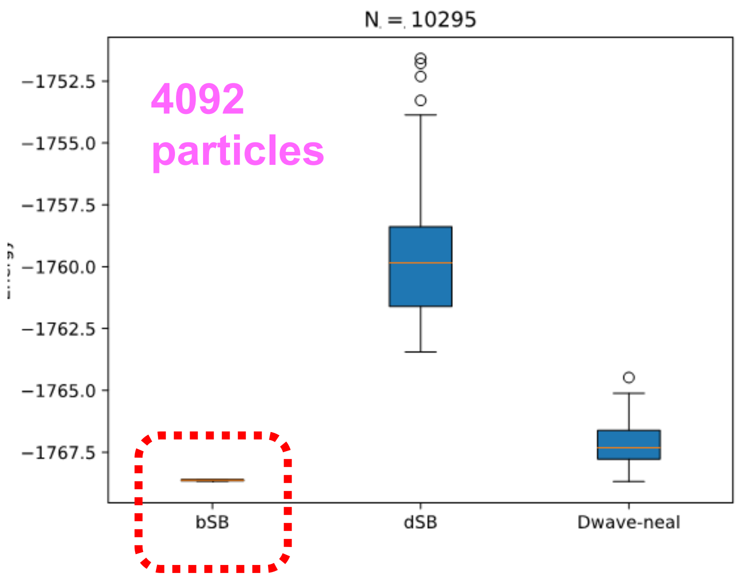
Graph size	Algorithm	Hardware	Time(s)
4 × 4 × 8	TTN	CPU 1 core	5.62
	Brute-force search[46]	GPU Titan V	>10^{48}
	Exact belief propagation[13]	CPU 1 core	~0.96
	QA[13]	D-Wave	~0.05
	bSB	CPU 1 core	0.12
	bSB	GPU Tesla V100	<0.001
8 × 8 × 8	TTN	CPU 1 core	32400
	TTN[44]	GPU Tesla V100	84
	Brute-force search[46]	GPU Titan V	>10^{190}
	Exact belief propagation[13]	CPU 1 core	~2880
	dSB	CPU 1 core	17.64
	dSB	GPU Tesla V100	<0.68

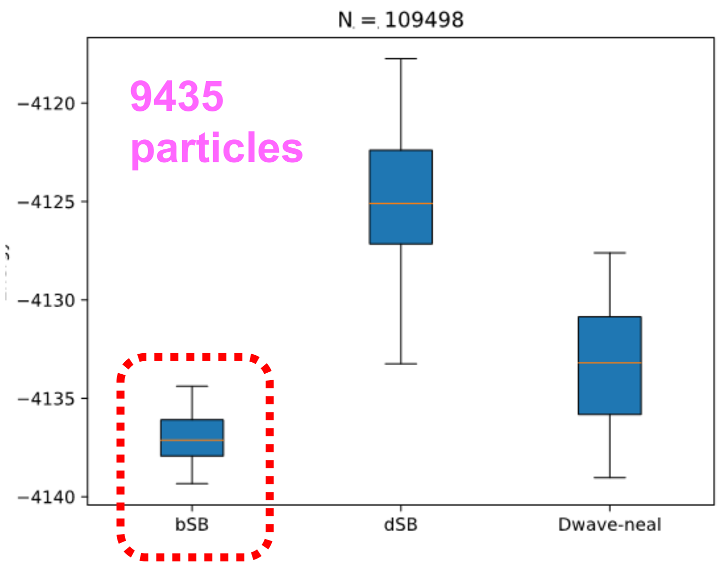
Q.G. Zeng et al., Comm. Phys. (2024) 7:249

- **Simulated bifurcation (SB) is a powerful quantum-inspired algorithm.**
- **It can run in parallel** unlike simulated annealing. It also benefits from cutting-edge resources such as **GPUs and FPGAs**.
- **It is known to outperform other classical algorithms as well as quantum annealing (QA) for some problems.**

Minimum Ising Energy Prediction



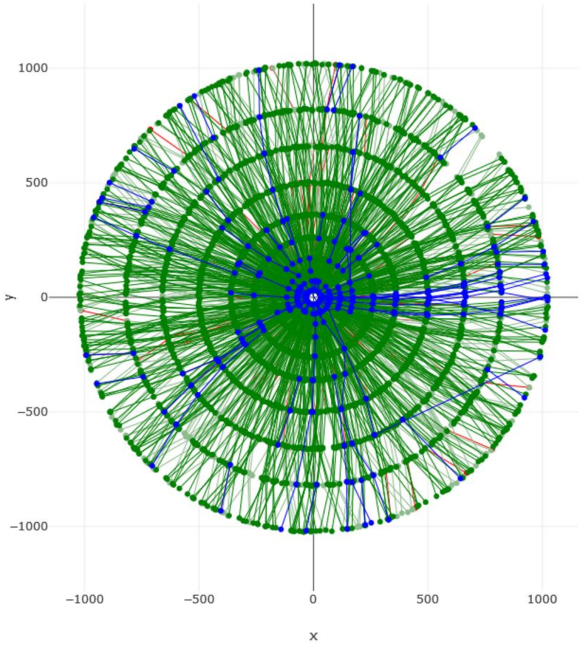




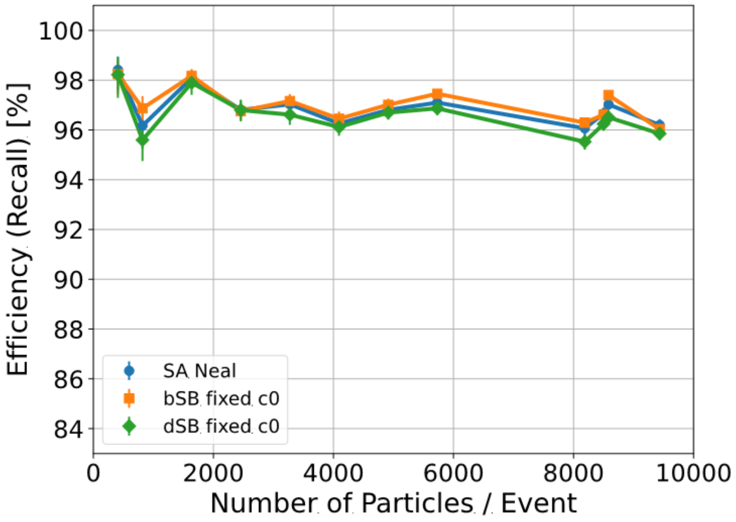
- **Ballistic simulated bifurcation (bSB) provides the best prediction of minimum energy with the least fluctuation for the tracking QUBO**.
- Discrete simulated bifurcation (dSB) is not as good as the other two, but the impact on the reconstruction performance is not significant (next slide)

H. Okawa, Q.G. Zeng, X.Z. Tao, M.H. Yung, Comput. Softw. Big Sci. 8, 16 (2024)

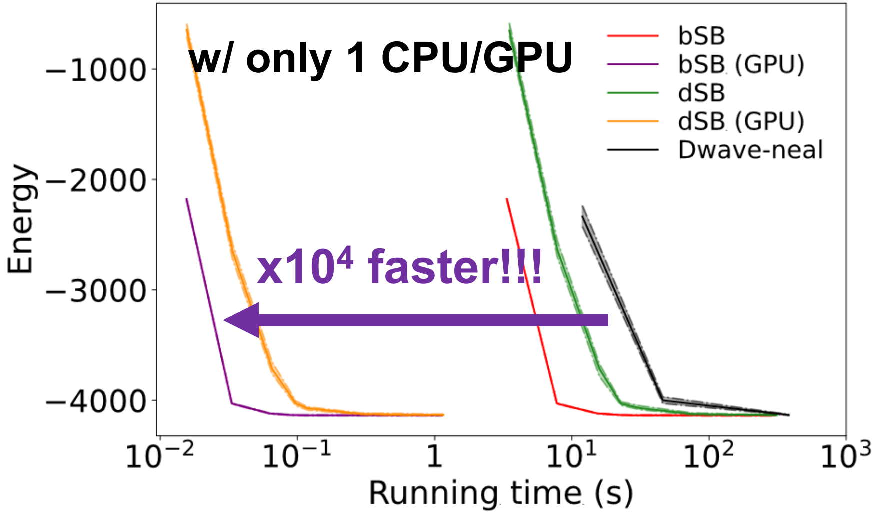
Global Track Reconstruction - Triplets



Efficiency



Ising Energy vs computing time

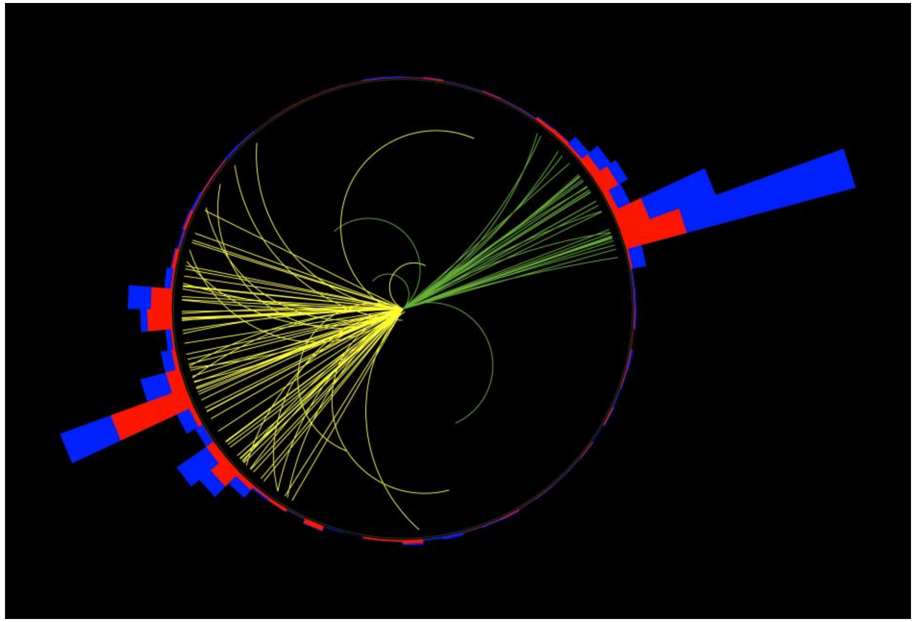


w/ only 1 CPU/GPU

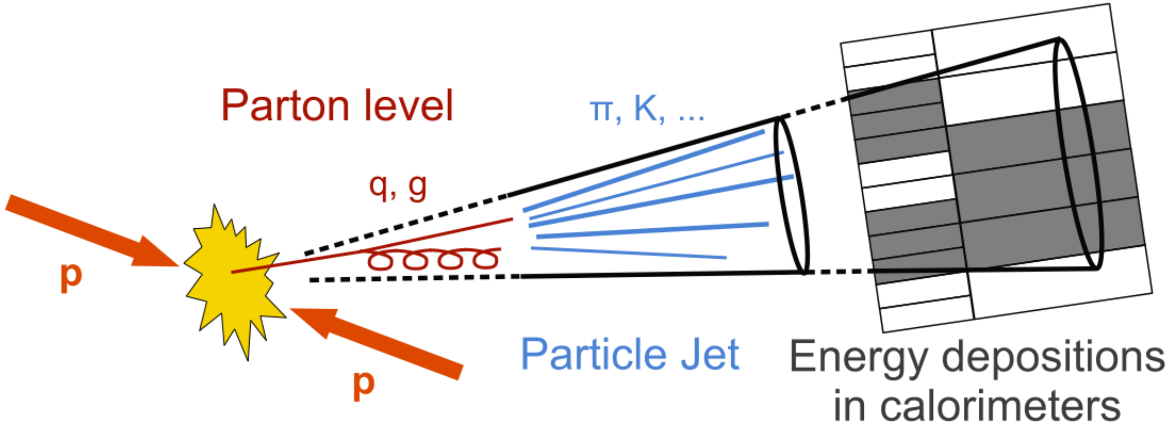
x10^4 faster!!!

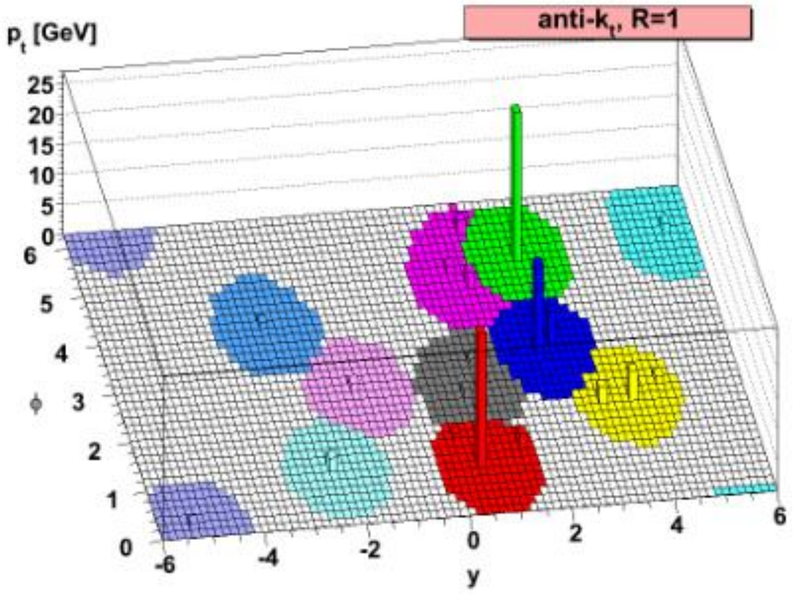
- **Quantum-inspired overcomes both hardware limitation & computing speed bottleneck.**
- **We introduced simulated bifurcation (SB) to HEP for the first time.** It provides comparable or slightly better efficiency & purity than D-Wave Neal SA.
- A SB variant, bSB provides **4 orders of magnitude speed-up (23min → 0.14s) from D-Wave Neal SA** **(cf. D-Wave hardware w/ sub-QUBO is ~2 orders of magnitude slower than Neal).**
- **SB can effectively run with multiple processing, GPU & FPGA → Perfect match with HEP!**

Jet Reconstruction



Jet Reconstruction

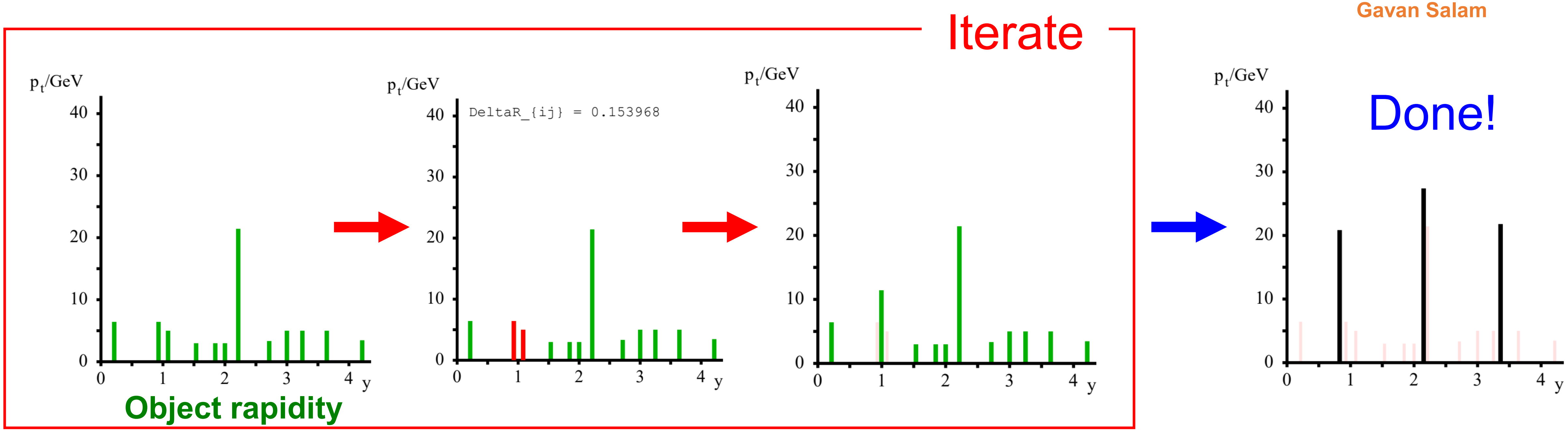




- Gluons & quarks spray collimated arrays of particles. They are detected as tracks & energy deposits in the calorimeter.
- Clustering particles as jets provides important proxies to understand the original parton kinematics but is a non-trivial & CPU-consuming task.
- There have been numerous studies on this topic with various algorithms proposed. **Many of them are implemented in FastJet.**

Traditional Approach

Modified figures from Gavan Salam



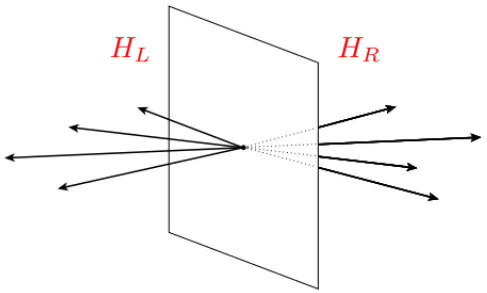
- Repeatedly recombine closest pair of objects (tracks, calorimeter clusters or particle flows etc.):
  - Terminate by a **user-defined distance R [inclusive clustering]**
  - Terminate by a **user-defined jet multiplicity [exclusive clustering]**
- **Users also define the distance**; i.e. how they call objects as “close” → $\Delta R_{ij}^2 = \Delta y_{ij}^2 + \Delta \phi_{ij}^2$

e.g. Distance adopted in Cambridge-Aachen jet algorithm

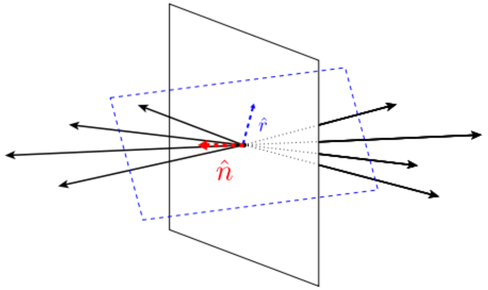
Jet Reconstruction (Iterative)

Iterative jet reconstruction also requires QuAM, so studies are mostly on complexity analyses.

Thrust as partitioning (QA)



Thrust as axis finding (Grover search)

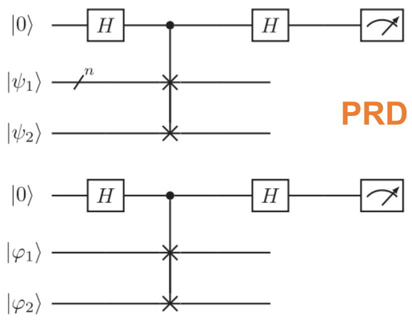


- Authors improved classical algorithm inspired by SISCone.
- **Quantum & classical algorithms scale the same, but for different reasons.**

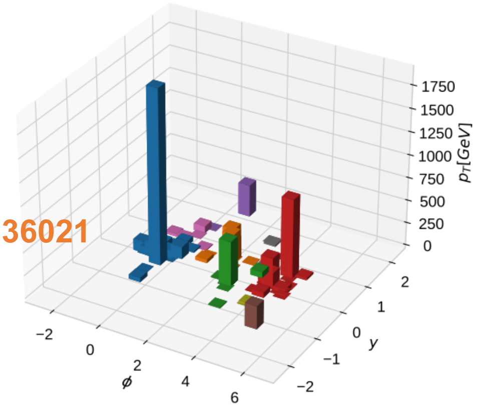
Implementation	Time Usage	Qubit Usage
Classical[124]	$O(N^3)$	—
Classical with sort inspired by SISCone[125]	$O(N^2 \log N)$	—
Classical with parallel sort	$O(N \log N)$	—
Quantum annealing	Gap dependent	$O(N)$
Quantum search: sequential model	$O(N^2)$	$O(\log N)$
Quantum search: parallel model	$O(N \log N)$	$O(N \log N)$

A. Wei, P. Naik, A.W. Harrow, J. Thaler, PRD 101, 094015 (2020),

Quantum subroutine to compute Minkowski-based distance



PRD 106 036021



(b) Quantum anti-k_T, $p = -1$, $R = 1$, $\epsilon_c = 0.99$.

- **Quantum & classical algorithms [FastJet] scales the same for sequential clustering.**
- IBM quantum circuit simulator was also used to actually perform clustering..

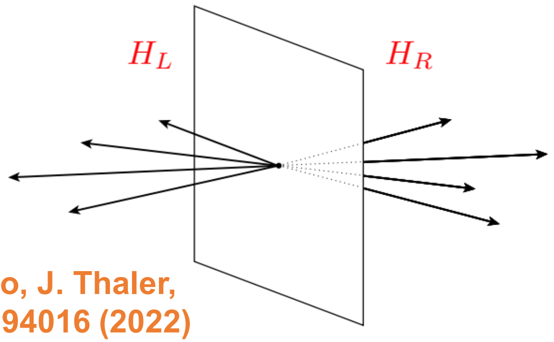
Jet algorithms	Classical	Quantum
`K-means`	$O(NKD)$	$O(NK \log D)$,[116] $O(N \log K \log(D-1))$[127]
AP	$O(N^2TD)$	$O(N^2T \log(D-1))$[127]
k_t, C/A, anti-k_t	$O(N^3)$ [suboptimal]	$O(N^2 \log N)$[127]
	$O(N \log N)$ [`FastJet`][132, 133]	$O(N \log N)$[127]

[116] D. Pires, P. Bargassa, J. Seixas, Y. Omar, arXiv:2101.05618 (2021)

[127] J.J. Martinez de Lejarza, L. Cieri, G. Rodrigo, PRD 106 036021 (2022)

Dijet Reconstruction (Global)

Quantum Annealing (Thrust)



A. Delgado, J. Thaler, PRD 106, 094016 (2022)

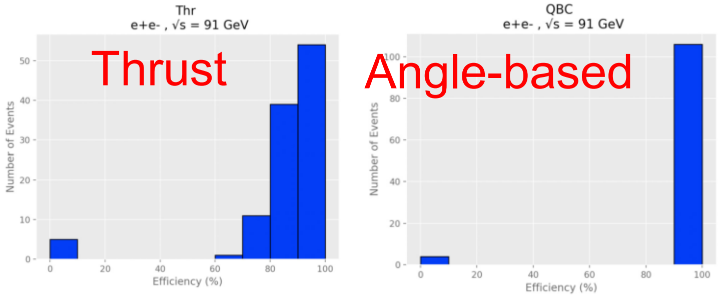
$$O_{\text{QUBO}}(\{x_i\}) = \left(\sum_{i=1}^{N} |\vec{p}_i|\right)^2 T(\{x_i\})^2$$

$$T(\{x_i\}) = 2\frac{\left|\sum_{i=1}^{N} x_i \vec{p}_i\right|}{\sum_{i=1}^{N} |\vec{p}_i|}$$

- Quantum annealing with tuned annealing parameters & hybrid quantum/classical approach without tuned parameters exhibit similar performance to exact classical approaches.

D. Pires, Y. Omar, J. Seixas, PLB 843 (2023) 138000

Quantum Annealing (Angle)

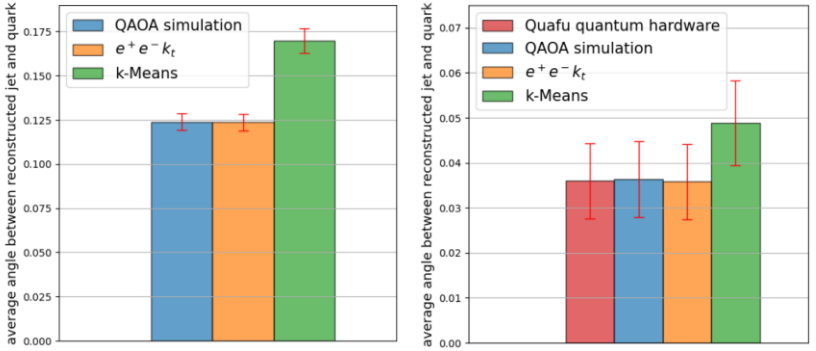


$$H = \frac{1}{2}\sum_{i,j=1}^{N} -\cos\left[\theta(\vec{p}_i, \vec{p}_j)\right] s_i s_j$$

- Angle-based approach shows more compatible clustering as ee-k_t for dijet events.
- However, **the same approach did not work for multijet events**. → Usage of multiple qubits for one hot encoding is prone to errors.

Y. Zhu et al., Sci. Bul. 70 (2025) 460

QAOA (Angle) or K-means


30-particle data ($e^+e^- \rightarrow ZH \rightarrow \nu\nu ss$)

6-particle data ($e^+e^- \rightarrow ZH \rightarrow \nu\nu ss$)

$$\widehat{H}_C = \frac{1}{2}\sum_{(i,j)\in E} W_{ij}\left(I - \sigma_i^z \sigma_j^z\right)$$

W_{ij}: angle b/w particles i & j

- Used simplified/small-sized (6- or 30-particle) dijet events.
- Evaluated average angle w/ QAOA or K-means with Quafu hardware/simulator.

Multiple Jet Clustering as Optimization

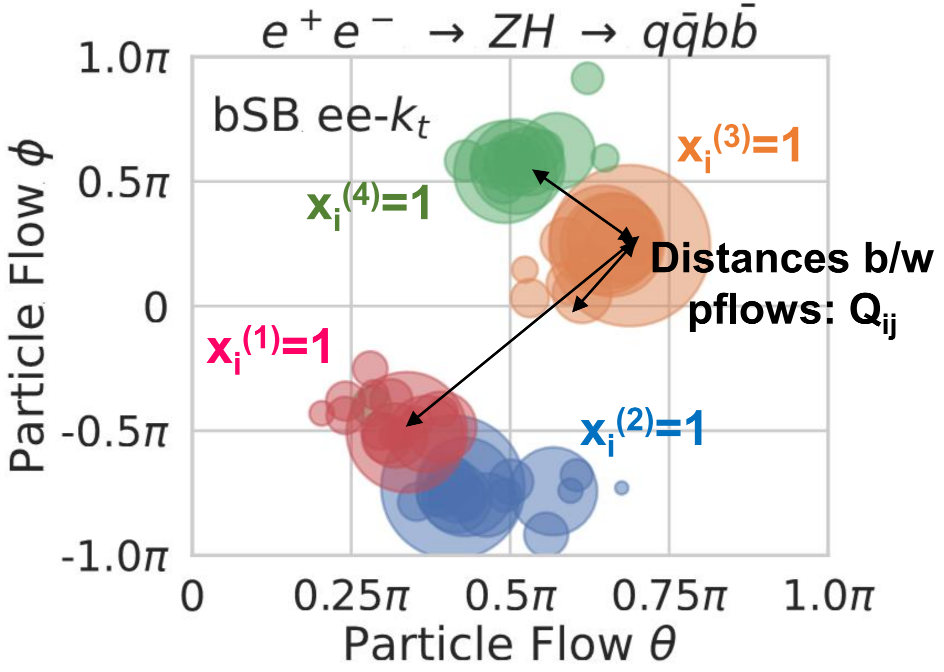
QUBO Formulation

$$O_{\mathrm{QUBO}}^{\mathrm{multijet}}(x_i) = \underbrace{\sum_{n=1}^{n_{\mathrm{jet}}} \sum_{i,j=1}^{N_{\mathrm{input}}} Q_{ij} x_i^{(n)} x_j^{(n)}}_{\text{Defines distances b/w particle flow candidates}} + \lambda \underbrace{\sum_{i=1}^{N_{\mathrm{input}}} \left(1 - \sum_{n=1}^{n_{\mathrm{jet}}} x_i^{(n)}\right)^2}_{\text{Avoids double/none-assignment of particle flow candidates}},$$

$$Q_{ij} = 2\mathrm{min}(E_i^2, E_j^2)(1 - \cos\theta_{ij}).$$ **[ee-k_t distance]**

$$Q_{ij} = -\frac{1}{2}\cos\theta_{ij}$$ **[angle-based]**

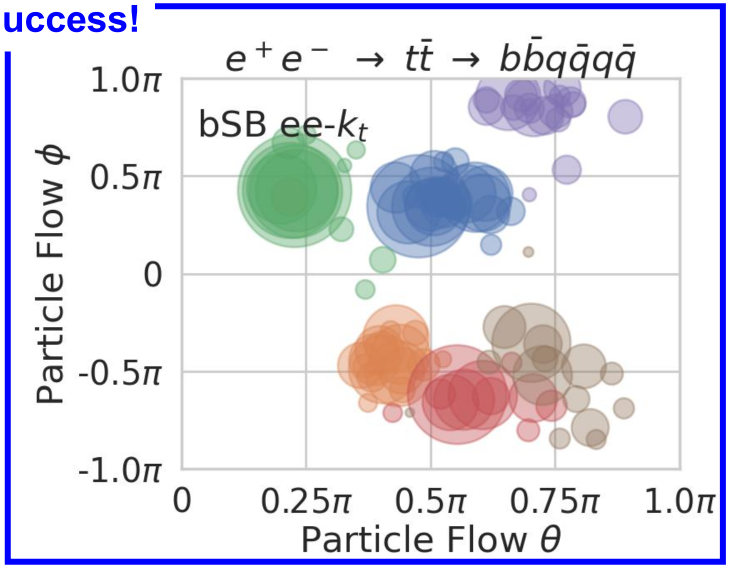
D. Pires, Y. Omar, J. Seixas, PLB 843 (2023) 138000



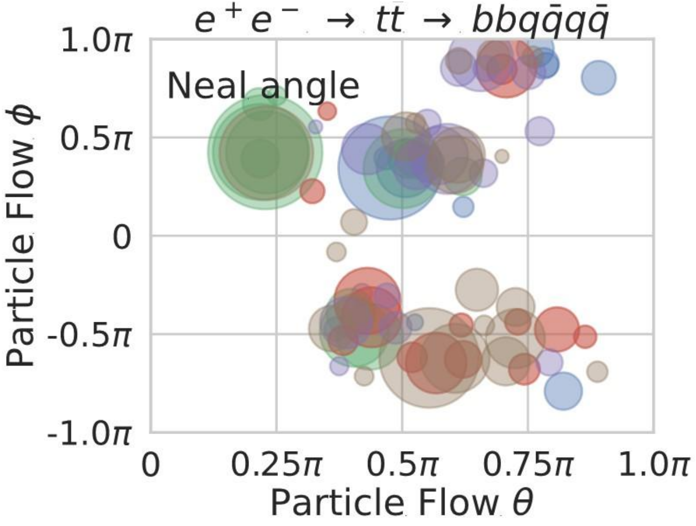
- **In this study, we consider exclusive jet finding (i.e. fixed number of jets):** the baseline at CEPC & other e+e- future Higgs factories.
- **QUBO is designed for multijet beyond dijet with the ee-k_t distance.** $x_i^{(n)}$=1 means i-th particle flow belongs to n-th jet.
  → Performance is compared to the angle-based distance method (used in D. Pires et al.).

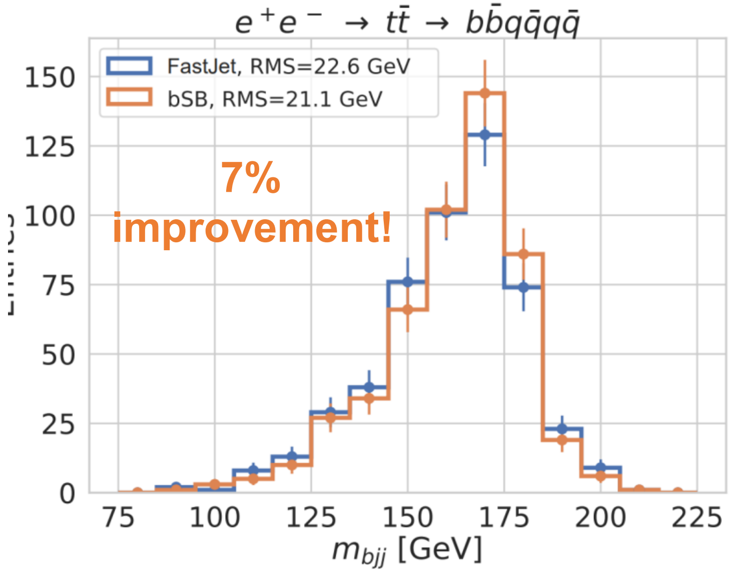
Event Displays ($t\bar{t}$)

Success!



Previous method



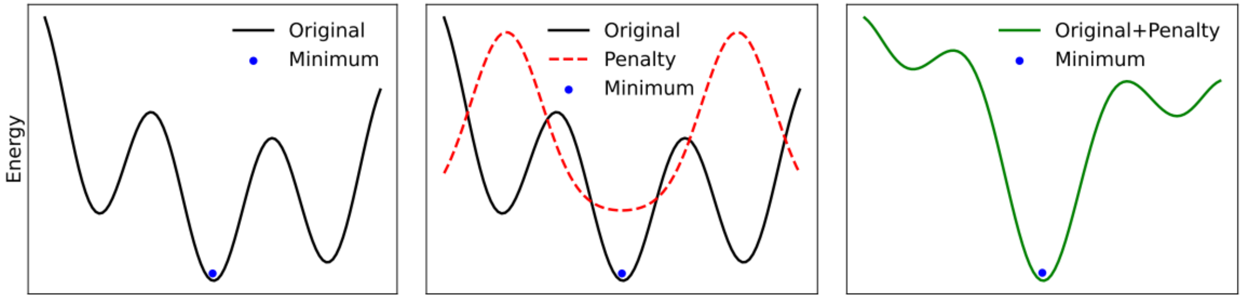


7% improvement!

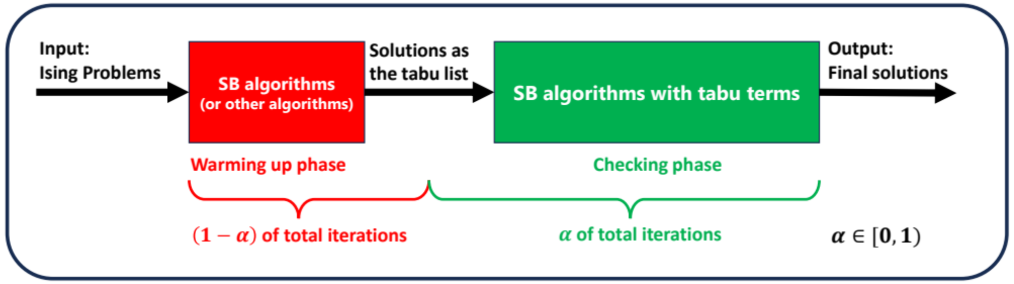
- Only bSB quantum-inspired w/ ee-k_t QUBO model can reasonably reconstruct all jets.
- Angle QUBO model & other quantum-inspired algorithms cannot handle multijet.
- bSB improve mass resolution for multijet! (6% for H→bb, 7% for top quark→bqq)
  → the virtue of global jet reconstruction
- **<u>Key takeaway: performance of Ising problem solver & QUBO modeling are both crucial.</u>**

XZ Tao, QG Zeng, ZJ Huang, BW Zuo, J Zhuang, H Okawa,
MH Yung, *Commun. Phys.* 9 (2026) 1, 100

New Quantum-Inspired Algorithm

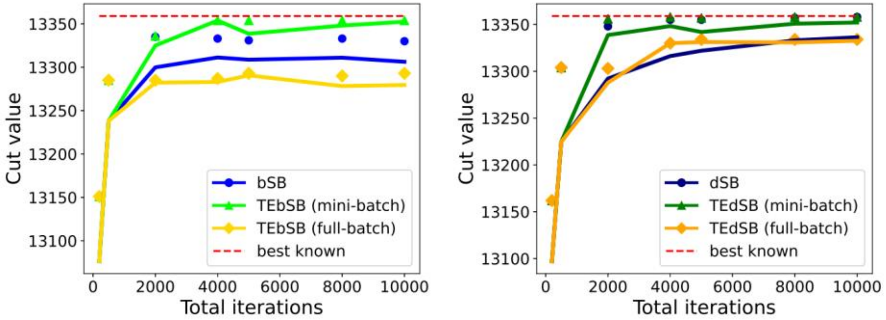


Tabu-Enhanced Simulated Bifurcation



- We have developed a new variant of SB: Tabu-enhanced simulated bifurcation.
- **A penalty is applied to fill the extracted local minima during the warm-up phase.**
- **Visible improvement in <u>both minimum energy prediction & computing time</u>** for graph & TrackML benchmark datasets.
- Applications to other HEP tasks are under way.

Max-cut values from G22 instance



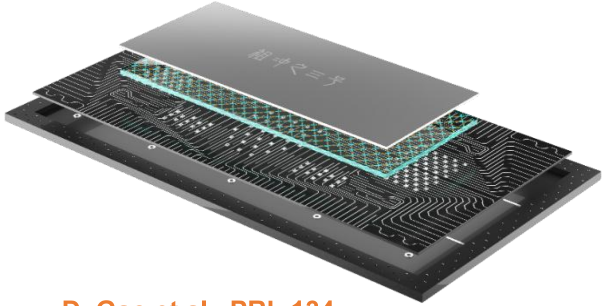
Minimum energy predictions from TrackML datasets

	bSB		TEbSB		dSB		TEdSB	
	Time (s)	Energy (a.u.)	Time (s)	Energy (a.u.)	Time (s)	Energy (a.u.)	Time (s)	Energy (a.u.)
ev1004 (N=109498)	8.67	-448998	7.25	**-449363**	9.02	-447488	7.43	-449349
ev1014 (N=78812)	5.06	-263353	4.27	**-263650**	5.24	-261860	4.33	-263641
ev1023 (N=80113)	5.33	-261244	4.42	-261345	5.48	-260928	4.80	**-261362**

Prospects

- Quantum circuits in principle should accelerate some components of reconstruction but require QuAM architecture, error correction & increased # of qubits.
- For quantum annealing, increasing the connectivity will be mandatory.
- It is also to be seen what quantum-centric supercomputing can bring to HEP.
- In any case, for very large-scale experimental tasks [using $O(10^6)$ qubits], quantum-inspired techniques will likely stay valuable.

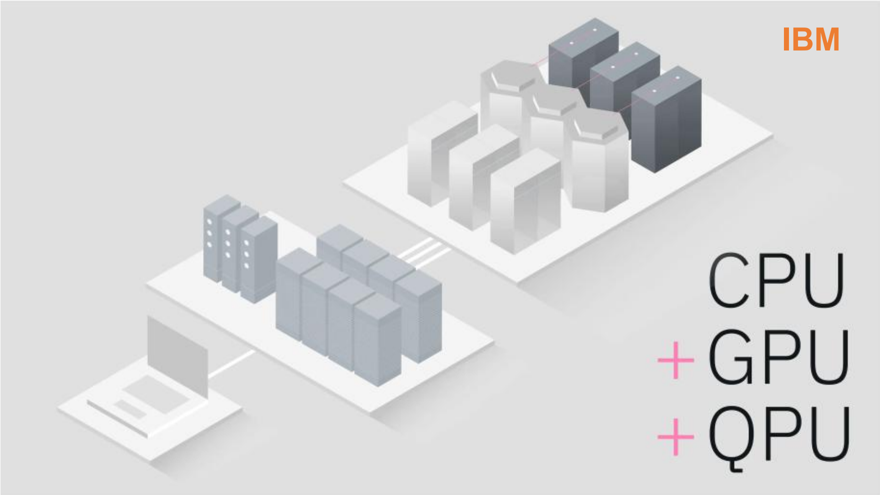
USTC Zuchongzhi-3



D. Gao et al., PRL 134 (2025) 090601

D-Wave Advantage2

Quantum-centric supercomputing

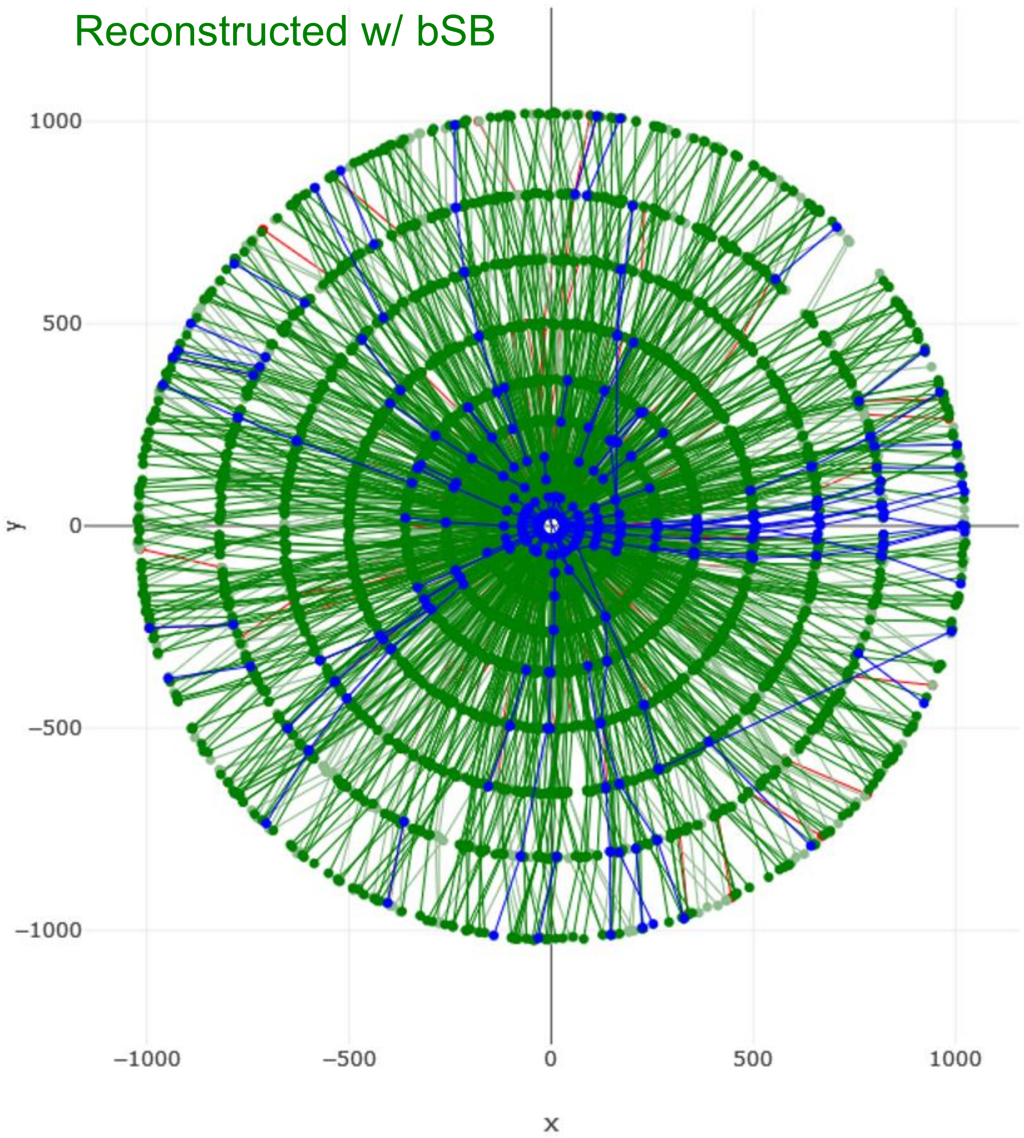


References:

- H. Okawa, CCIS 2036 (2024) 272, arXiv:2310.10255
- H. Okawa, et al., Comput. Softw. Big Sci. 8, 16 (2024)
- H. Okawa, et al., Phys. Lett. B 864 (2025) 139393
- XZ Tao et al., *Commun. Phys.* 9 (2026) 1, 100
- H. Okawa, arXiv:2511.16713 (invited review)

Summary

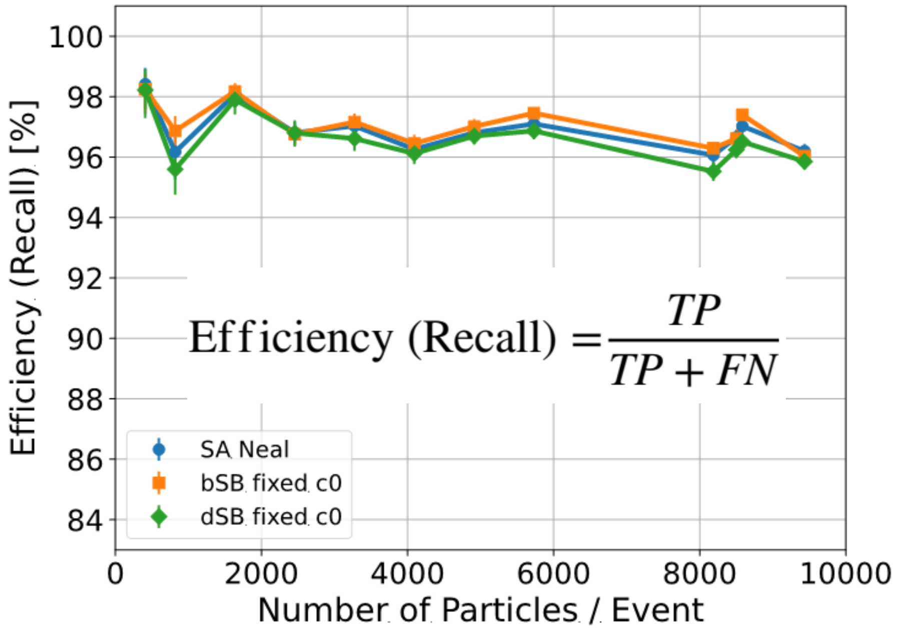
- Various experimental tasks, e.g. reconstruction, can be formulated as optimization problems.
- Each experimental task has differences in the number of qubits required, the level of connectivity/sparsity of QUBO, the demand of computing speed, etc.
    - We should keep investigating in detail what kind of approaches are valuable for practical implementations.
- Meanwhile, a quantum-inspired approach using simulated bifurcation demonstrated a scalable approach to large-scale track reconstruction.
- We have also succeeded in improving SB quantum-inspired algorithms further. Applications to specific high-energy physics problems are under way.
- I focused mostly on optimization & quantum-inspired studies in this talk, but other QML studies are also under way.

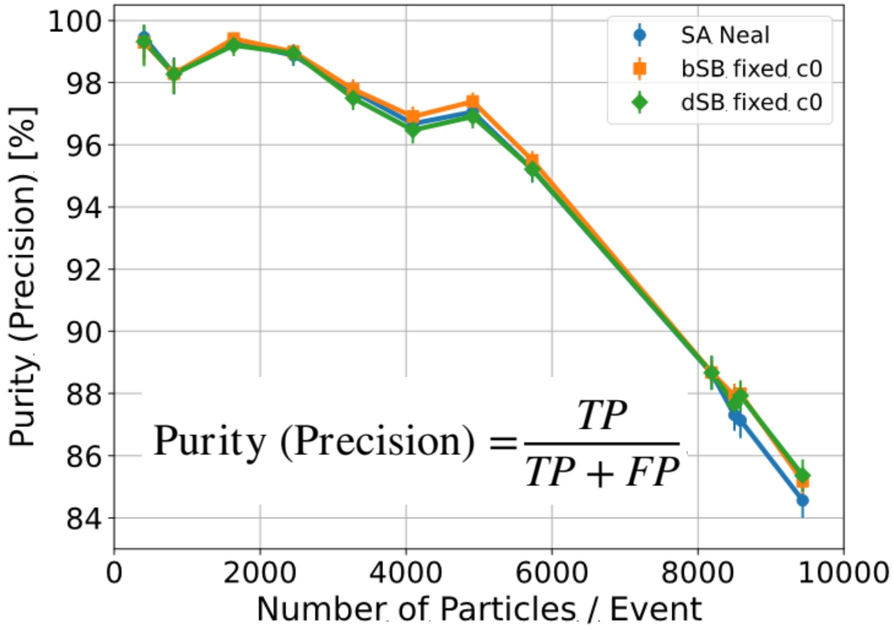


Thank you for listening!

H. Okawa, Q.-G. Zeng, X.-Z. Tao, M.-H. Yung, Springer Comput Softw Big Sci 8, 16 (2024).

Track Efficiency & Purity w/ QAIA

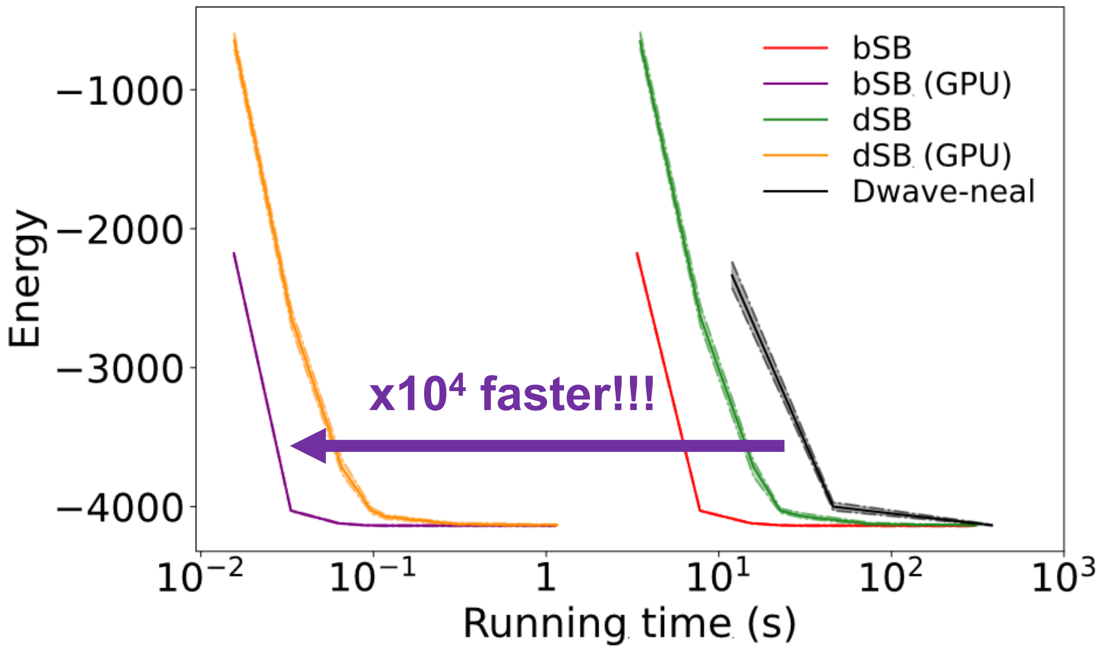




- bSB & dSB provides **compatible or slightly better performance than D-Wave Neal.**
- **Track efficiency stays over 95%** for all dataset up to the highest HL-LHC conditions
- Purity degrades with track multiplicity but **>90% for <6000 particles, >84% even for ~10000 particles**. → **Very promising for the HL-LHC!**

H. Okawa, Q.-G. Zeng, X.-Z. Tao, M.-H. Yung, Springer Comput Softw Big Sci 8, 16 (2024).

Computation Speed



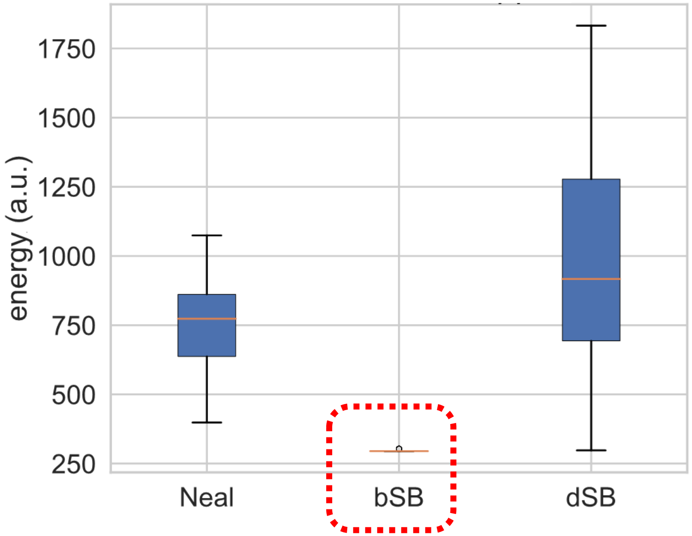
Only 1 CPU or GPU used

Data Information		Time to target [s]				
# of particles	QUBO size	bSB	bSB (GPU)	dSB	dSB (GPU)	D-Wave Neal
409	778	0.007	0.021	0.032	0.092	0.060
818	1431	0.012	0.019	0.293	0.478	0.169
1637	2904	0.012	0.019	0.293	0.478	0.169
2456	4675	0.014	0.017	–	–	0.479
3274	6945	0.032	0.022	–	–	1.229
4092	10295	0.005	0.022	0.015	0.065	0.030
4912	14855	0.027	0.016	–	–	2.165
5730	22022	0.109	0.042	–	–	3.853
8187	67570	0.488	0.028	–	–	404.297
8500	78812	1.899	0.108	–	–	785.732
8583	80113	1.321	0.067	–	–	93.782
9435	109498	3.884	0.140	–	–	1366.808

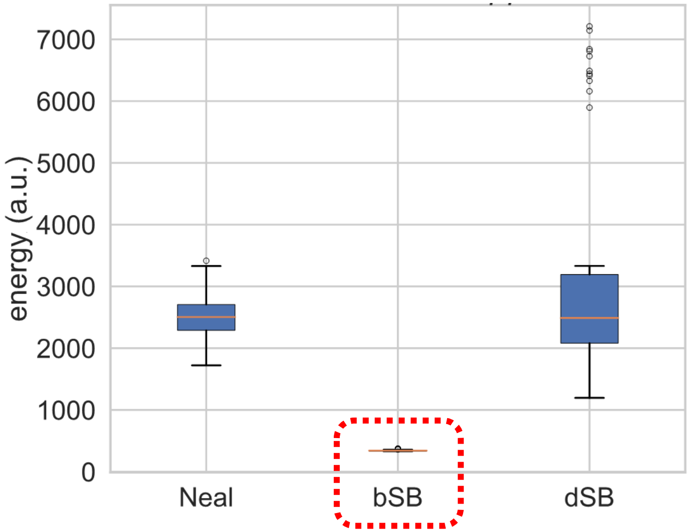
- bSB provides 4 orders of magnitude speed-up (23min → 0.14s) from D-Wave Neal at most **(D-Wave quantum annealing + qbsolv is even 2 orders of magnitude slower than Neal)**. **bSB expects even more speed-up with larger data size.**
- Unlike D-Wave Neal, **simulated bifurcation can effectively run w/ multiple processing, GPU & FPGA → Perfect match with HEP computing environment!!**

H Okawa, XZ Tao, QG Zeng, MH Yung, arXiv:2410.14233 (2024)

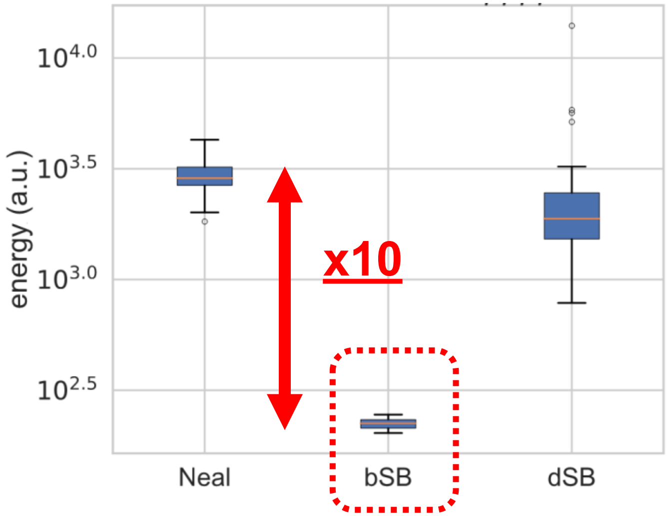
Ising Energy Prediction (Jet Reco)



$e^+e^- \rightarrow Z \rightarrow q\bar{q}$



$e^+e^- \rightarrow ZH \rightarrow q\bar{q}b\bar{b}$



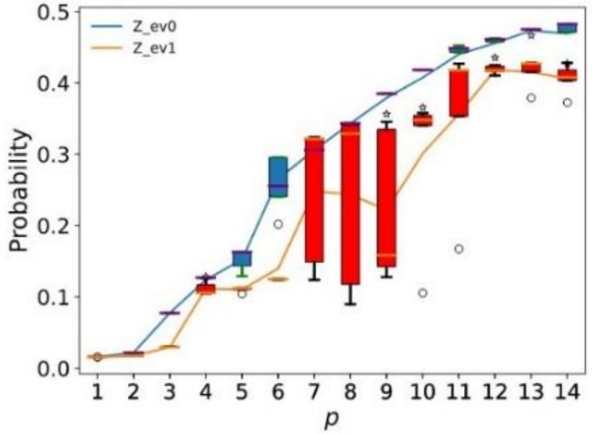
$e^+e^- \rightarrow t\bar{t} \rightarrow b\bar{b}q\bar{q}q\bar{q}$

- **Fully-connected QUBOs are difficult to solve**; it is known that quantum annealing hardware is not good at solving them so far.
  - This is in contrast to track reconstruction, in which the QUBOs are largely sparse.
- **Ballistic SB (bSB) predicts lowest energy with the smallest fluctuation.**
- **Performance is especially outstanding for 6-jet QUBOs → bSB can find x10 lower minimum energy for the all-hadronic $t\bar{t}$ events!**

Comparison w/ Quantum Hardware

- Performance was compared for two simplified jet datasets (12 qubits) to quantum annealing and QAOA using simulator.
  - QuantumAnnealing.jl package is used to evaluate D-Wave 2000Q performance (6-way connectivity).
- Even for such small datasets, **bSB exceeds the speed of quantum annealing by about two orders of magnitude & even more for QAOA** (w/ a caveat that QAOA should run faster on real quantum hardware).

QAOA Performance



Time-to-solution for D-Wave 2000Q estimated by simulation, bSB, dSB, and QAOA on a quantum circuit simulator for two simplified $Z \to q\bar{q}$ events.

Event	D-Wave [s]	bSB [s]	dSB [s]	QAOA [s]
0	21.29	0.35	0.79	1.07×10^3
1	20.52	0.36	0.89	3.36×10^3