A general approach to quantum integration of cross-sections in high-energy physics [arXiv:2502.14647]

EPS-HEP 2025: Marseille

Presented by:

Ifan Williams

11/07/25

ifan.williams@quantinuum.com

Introduction

Motivation: quantum computing applications in high-energy physics (HEP)

- Broadly expected to be two domains of application for quantum computing in HEP:

1. Theoretical modelling i.e., quantum simulation of intractable classical problems e.g., out-of-equilibrium and real-time dynamics, thermalisation and dynamics after quench in lattice gauge theories

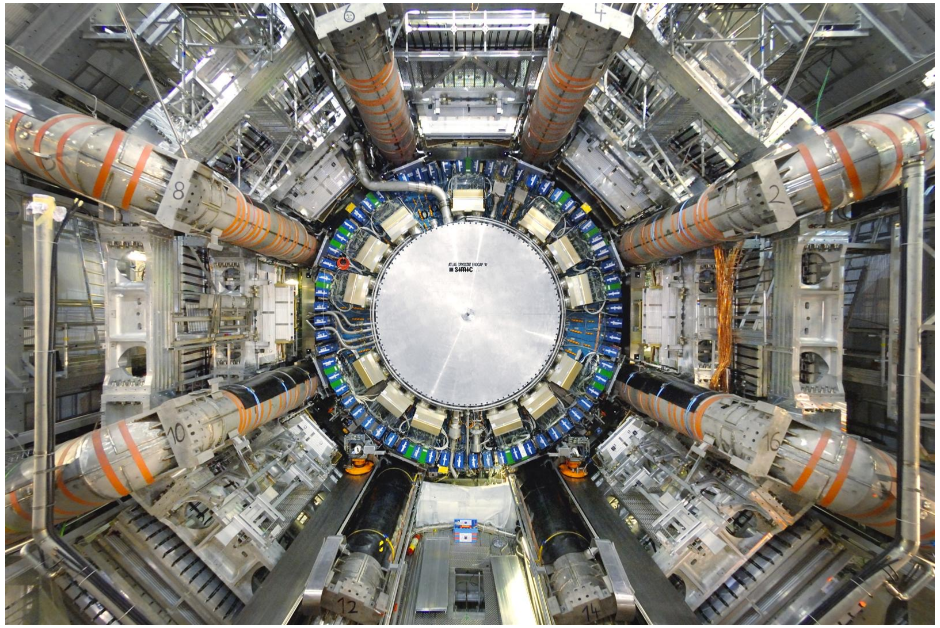
2. HEP experiments i.e., data analysis, data generation (simulation), detector algorithms, identification and reconstruction algorithms

Focus of this talk

Motivation: computational frontiers HEP

- Collider experiments such as the Large Hadron Collider (LHC) generate enormous volumes of data
- Precise theoretical predictions essential for comparison with experimental data
- Order billions of CPU hours per year consumed by the LHC experiments for generating simulated data

- Requires innovative technological solutions – theme of this talk!



Cross-section calculations

Cross sections

- Cross sections relate to the probability of a certain scattering process $a + b \rightarrow c + d + \cdots$ occurring in some collider experiment
- General form:

$$\sigma = \frac{1}{F} \int d\Phi \, |\mathcal{M}|^2 \, \Theta(C[\Phi] - C[\Phi_c])$$

- Specific form of matrix element integration in a cross-section calculation can be written

$$\sigma = \int \prod_{i=1}^{N_I} dx_i \frac{\sum_{S_k \in I} \alpha_k \prod_{j \in S_k} x_j^{n_j}}{\prod_{p=1}^{N_P} \left(x_p - M_{op}^2\right)^2 + M_{op}^2 \Gamma_{op}^2}$$

Multi-dimensional integral with highly peaked structure

- Monte Carlo integration

What is classical MCI (reminder)?

$$\mathrm{E}[f(X)] = \int f(x)p(x)dx$$

- Expectation of function of random variable can be estimated by averaging samples (let $X \sim p(x)$):

$$\mathrm{E}[f(X)] = \sum_{x} f(x)p(x) \approx \frac{1}{q}\sum_{j=1}^{q} f(X_j)$$

$f(x)$ - 'function applied'

$p(x)$ - 'probability distribution'

- Root mean-squared error (RMSE) scaling of MCI with number of samples q:

$$\mathrm{RMSE} \propto \mathcal{O}(1/\sqrt{q})$$

Reduce overall variance using techniques such as:

1. Importance sampling
2. Adaptive Monte Carlo

- Quantum Monte Carlo integration (QMCI)

What is QMCI?

$$\text{RMSE} \propto \mathcal{O}(1/q)$$

- Quantum analogue of MCI - starting point is circuit P preparing quantum state $|p\rangle$ encoding probability distribution $p(x)$ such that $|p\rangle = P|\mathbf{0}\rangle$:

$$|p\rangle = \sum_x \sqrt{p(x)}|x\rangle$$

- A circuit R that operates on $|p\rangle|0\rangle$:

$$R|p\rangle|0\rangle = \sum_x \sqrt{p(x)}\,|x\rangle\left(\sqrt{1-f(x)}|0\rangle + \sqrt{f(x)}|1\rangle\right)$$

- Probability of measuring 1 on final qubit:

$$\sum_x p(x)f(x)$$

Computed using Quantum Amplitude Estimation (QAE) algorithm - quadratic speedup in sample complexity [G. Brassard et al., arXiv:quant-ph/0005055 2000]

Fourier QMCI

- Extend function applied to the samples as a piecewise, periodic function, which can then be decomposed as a Fourier series:

$$f(x) = c + \sum_{n=1}^{\infty} a_n \cos(n\omega x) + b_n \sin(n\omega x)$$

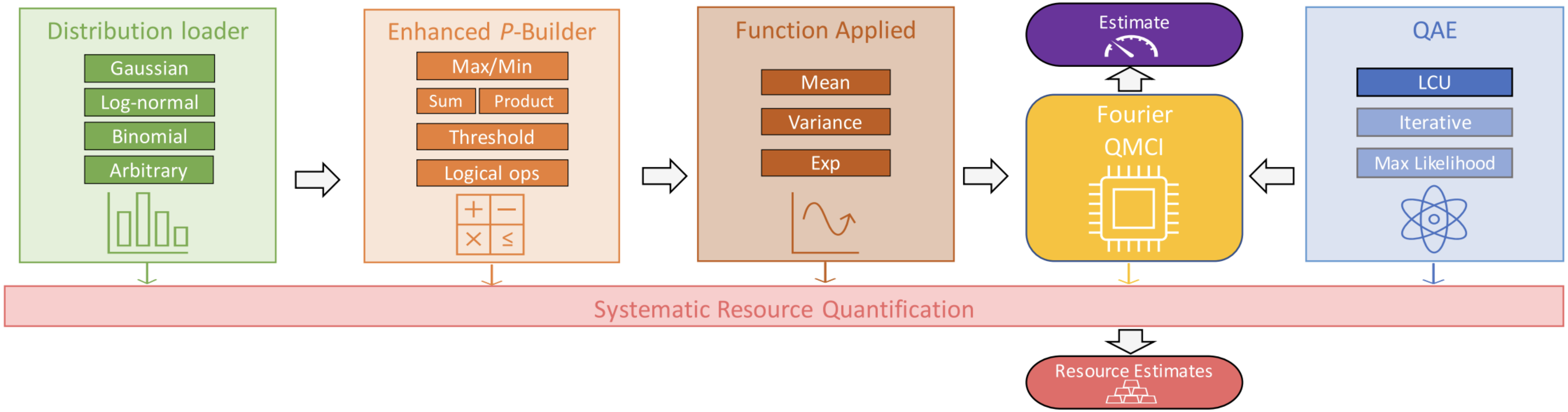
- Estimate for the expectation is easily estimated on a quantum computer:

$$\mathrm{E}[f(X)] = c + \sum_{n=1}^{\infty} a_n \left(\sum_x p(x) \cos(n\omega x) \right) + \sum_{n=1}^{\infty} b_n \left(\sum_x p(x) \sin(n\omega x) \right)$$

Each of the parenthesised sums can be efficiently estimated using QAE to provide low-depth quantum circuits – appealing for current noisy devices

Quantinuum's QMCI engine

- Quantinuum has developed a QMCI engine which has a theoretically guaranteed quadratic speed-up based on Fourier QMCI



- Engine architecture has modular framework

We are looking for beta users so please get in touch if interested!

Quantinuum’s QMCI engine

- Five key features for this talk:
  1. Agnostic to data-loading circuit P – any existing or future data-loading method can simply ’plug in’ to engine
  2. Enhanced P circuit builder – ability to automatically construct circuitry to condition the quantity being estimated on thresholds / maxima / minima / products / sums of random variables (implement cut functions)
  3. Fourier QMCI – decompose integral into minimal depth circuits for a variety of functions applied corresponding to moments or products of moments of random variables
  4. Specify target precision – perform integral calculations to a specified precision in terms of upper bound on RMSE of estimator
  5. Resource quantification – builds exact circuits corresponding to computations of interest and exactly quantify quantum resources required for running on given hardware for both noisy-intermediate (NISQ) and fault-tolerant (FT) eras

- QMCI for cross-section integration

Implementing a generic cross-section calculation

$$\sigma_1 \propto \int_0^s \frac{\mathrm{d}x\, x^n}{(x - M_o^2)^2 + M_o^2 \Gamma_o^2}$$

- Decompose integral in terms of 'building blocks'
- Treat propagator terms (denominator) as probability distribution - $p(x) = \frac{1}{\left(x - M_o^2\right)^2 + M_o^2 \Gamma_o^2}$
- Treat monomial terms (numerator) as function applied - $f(x) = x^n$

Implementing a generic cross-section calculation

$$\sigma_2 \propto \iint \frac{dxdy\, x^n y^m}{[(x - M_{o1}^2)^2 + M_{o1}^2\Gamma_{o1}^2][(y - M_{o2}^2)^2 + M_{o2}^2\Gamma_{o2}^2]}$$

$$f(x,y) = x^n\, y^m \qquad p(x,y) = \frac{1}{[(x - M_{o1}^2)^2 + M_{o1}^2\Gamma_{o1}^2][(y - M_{o2}^2)^2 + M_{o2}^2\Gamma_{o2}^2]}$$

- Probability distribution is a product of single-variable relativistic Breit-Wigner (BW) distributions
- Need efficient methods for preparing quantum states representing BW distributions for each of the relevant resonances in the Standard Model (W, Z, t, H)
- We propose and explore two different methods in our article (won't discuss here)

Implementing a generic cross-section calculation

$$\int_0^{c(x,y)} \ldots dxdy = \int \ldots \Theta\big(C(x,y)\big)\, dxdy$$

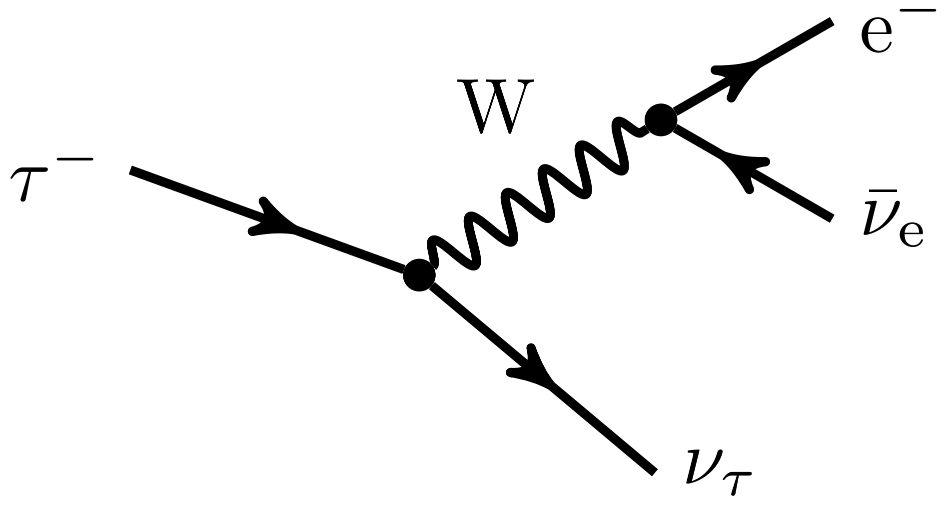
- Implement cuts via the QMCI engine's thresholding operations

- Example application

Tau decay example

$$\tau^- \rightarrow \nu_\tau e^- \overline{\nu_e}$$

$$p = k_1 + k_2 + k_3$$
$$s_1 = (p_1 + p_3)^2$$
$$s_2 = (p_2 + p_3)^2$$



$$|\mathcal{M}|^2 = -\frac{\alpha^2\pi^2}{\sin^4\theta_W}\frac{s_1^2 - M_\tau^2 s_1}{(s_2 - M_W^2)^2 + \Gamma_W M_W}$$

Tau decay example: methodology

- Example calculation (excluding integration over angles and constant factors)

$$\sigma \propto \int_0^s \int_0^{s-s_2} ds_1 ds_2 \frac{s_1^2 - s_1 M_\tau^2}{(s_2 - M_W^2)^2 + (M_W \Gamma_W)^2}$$

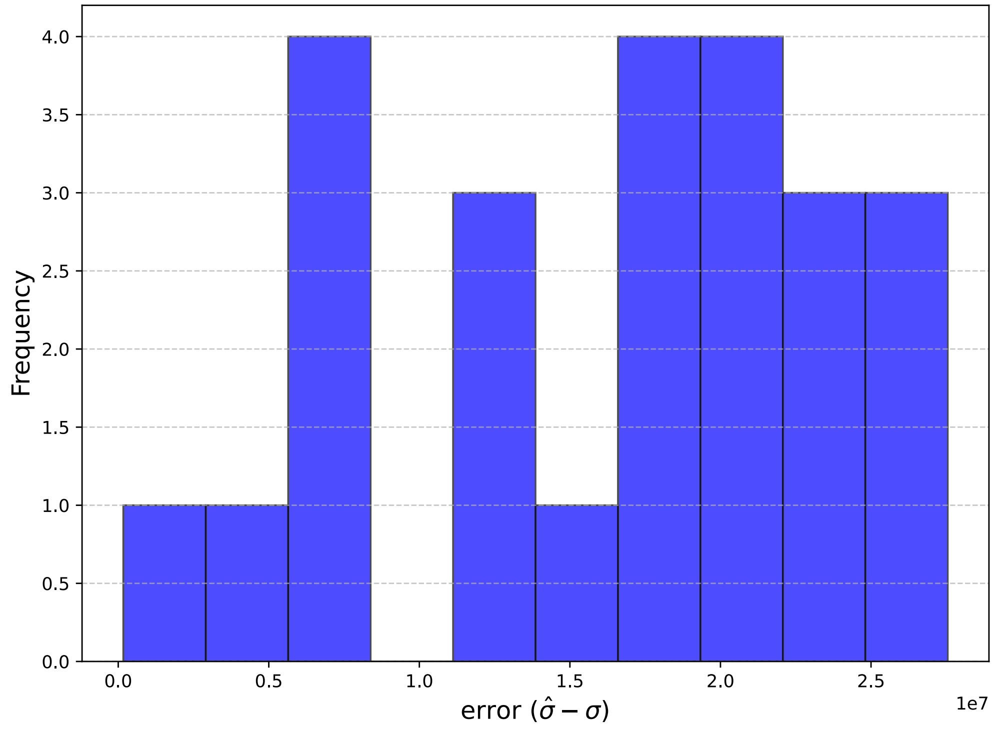
$\sqrt{s_{\max}} = 100\text{GeV}$

- Rewrite expression

$$\sigma \propto \int_0^s \int_0^s ds_1 ds_2 \underbrace{\frac{1}{(s_2 - M_W^2)^2 + (M_W \Gamma_W)^2}}_{p(s_1,s_2) = U(s_1)BW(s_2)} \underbrace{s_1^2}_{f(s_1,s_2) = s_1^2} C(s_1, s_2) - M_\tau^2 \int_0^s \int_0^s ds_1 ds_2 \underbrace{\frac{1}{(s_2 - M_W^2)^2 + (M_W \Gamma_W)^2}}_{p(s_1,s_2) = U(s_1)BW(s_2)} \underbrace{s_1}_{f(s_1,s_2) = s_1} C(s_1, s_2)$$

$C(s_1, s_2) = 1$ if $s_1 + s_2 < s$, else $= 0$

Tau decay example - results



$\sigma = 3.162 \times 10^8$

Compilation	Resource	Metric	Precision 10%	Precision 1%	Precision 0.1%
NISQ	Number of qubits	Largest across circuits	28	28	28
	CX gates	Total number across circuits	1.34×10^7	1.44×10^8	1.49×10^9
		Total depth across circuits	7.88×10^6	8.43×10^7	8.74×10^8
		Number in largest circuit	4.86×10^6	6.32×10^7	7.48×10^8
		Depth of largest circuit	2.85×10^6	3.71×10^7	4.39×10^8
	All gates	Total number across circuits	2.72×10^7	2.91×10^8	3.02×10^9
		Total depth across circuits	1.45×10^7	1.56×10^8	1.62×10^9
		Number in largest circuit	9.84×10^6	1.28×10^8	1.51×10^9
		Depth of largest circuit	5.27×10^6	6.85×10^7	8.11×10^8
Fault tolerant	Number of qubits	Largest across circuits	41	41	41
	T gates	Total number across circuits	5.37×10^8	6.97×10^9	8.23×10^{10}
		Total depth across circuits	5.21×10^8	6.75×10^9	7.98×10^{10}
		Number in largest circuit	2.18×10^8	3.08×10^9	4.21×10^{10}
		Depth of largest circuit	2.11×10^8	2.99×10^9	4.08×10^{10}

Conclusions

Conclusions

- HEP's massive computational demands make quantum technology a promising path to tackling classical bottlenecks.
- Developed a general quantum integration framework using Fourier QMCI via Quantinuum's engine — adaptable to any cross section with modular structure.
- FT hardware needs are high now but expected to drop as FT compilation techniques advance.
- Promising enhancements or efficiency improvements for HEP applications beyond integration — from resonance modelling to event sampling based on underlying distributions.

Joint work with Mathieu Pellen (mathieu.pellen@physik.uni-freiburg.de)

Article at arXiv:2502.14647

universität freiburg DFG Deutsche Forschungsgemeinschaft

QUANTINUUM

Backup

- Cross-section calculations

Cross sections

- Event selections in experimental analyses restrict the domain of integration to that physically accessible in the experiment
- Represented by a `cut function' C (which may not have a closed-form expression)

$$\sigma = \frac{1}{F} \int d\Phi \; |\mathcal{M}|^2 \, \Theta(C[\Phi] - C[\Phi_c])$$

Scalability

- Number of integration variables scales as $3n - 4$ for a $2 \rightarrow n$ scattering process
- Number of propagator terms depends on process (potentially all possible massive internal particles)

Example

$$pp \rightarrow \mu^- \overline{\nu_\mu} e^+ \nu_e \bar{b} b \bar{b} b$$

20 integration variables

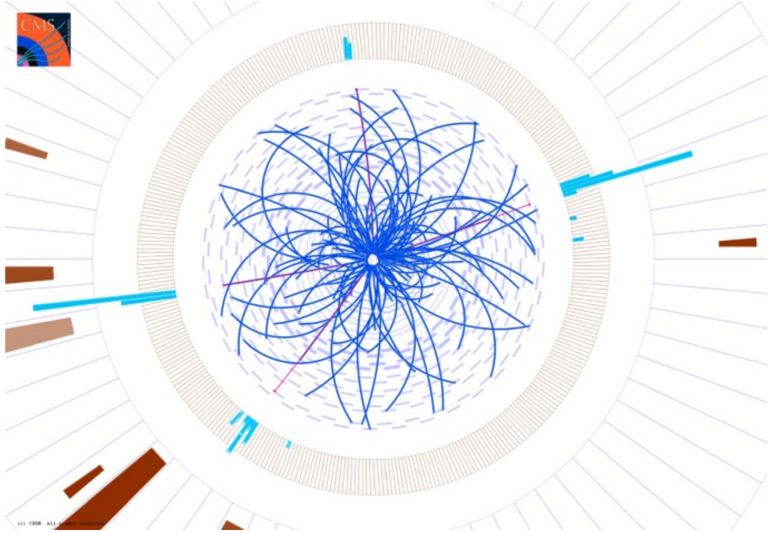
$\mathcal{O}(1000)$ propagators

- Monte Carlo integration

Monte Carlo integration (MCI) for HEP

- Numerical Monte Carlo techniques such as MCI 'workhorse' of theoretical HEP calculations
- Efficiently handle:

1. Cut functions without closed form
2. Intractability of analytical calculations at large multiplicities
3. Automation
4. Parton distribution functions defined on grid
5. Measurements of cross section as function of other observables



- Quantum Monte Carlo integration (QMCI)

Fourier QMCI

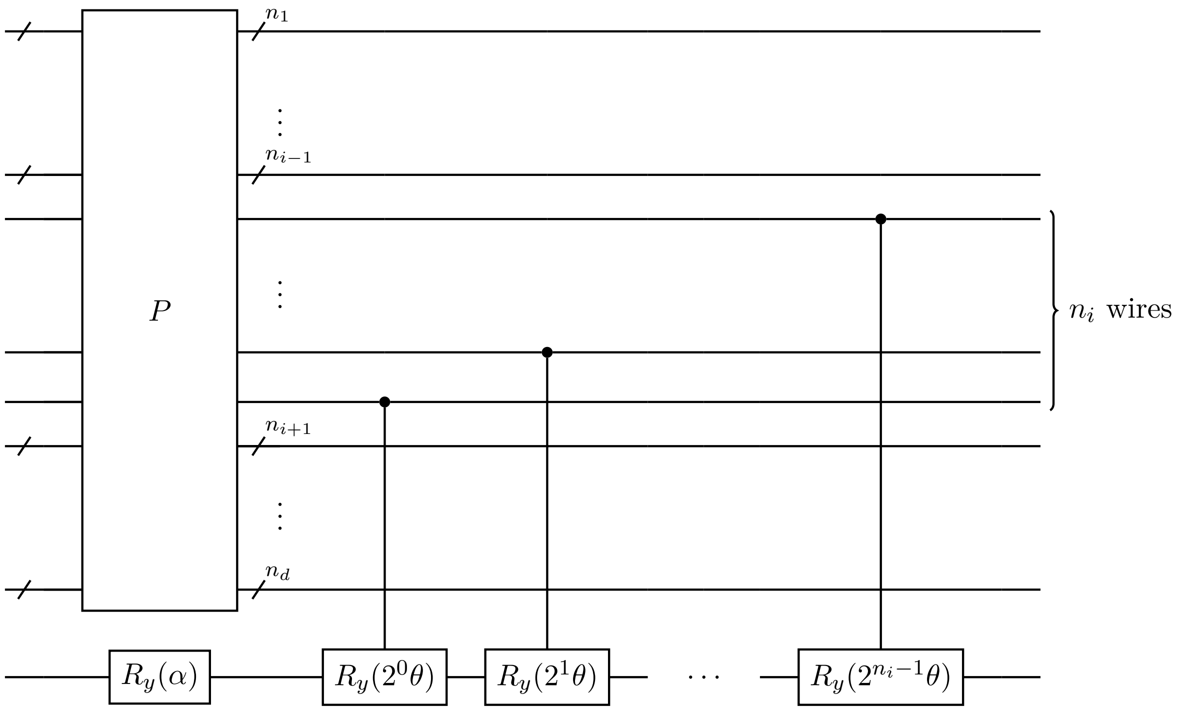
- The ‘natural’ quantity to estimate on a quantum computer is:

$$\mathrm{E}\left(\sin^2(mX + c)\right) = \sum_x p(x)\sin^2(mx + c)$$

- Can be achieved using a bank of $\mathrm{R_y}$ rotation gates

Fourier QMCI

- Costly quantum arithmetic operations replaced by bank of controlled rotation gates implementing trigonometric functions



Quantinuum's QMCI engine

- We have developed a QMCI engine which has a theoretically guaranteed quadratic speed-up based on the Fourier QMCI method

Method	Computes	MSE	Arithmetic
Classical MCI	$\mathbb{E}(f(X))$	$\Theta(q^{-1})$	Classical
Quantum MCI	$\mathbb{E}(f(X))$	$\Theta(q^{-2})$	Quantum & classical
Rescaled QMCI [1, 2]	$\mathbb{E}(X)$	$\Theta(q^{-4/3})$	Classical only
Fourier QMCI	$\mathbb{E}(f(X))$	$\Theta(q^{-2})$	Classical only

[1] arxiv.org:1806.06893
[2] arxiv.org:1905.02666

Quantum amplitude estimation convergence

- Characterise convergence of QAE in terms of mean-squared error (MSE) of estimate
- If q is either number of quantum queries or number of classical samples then MSE scaling is (up to)

QAE	$\mathcal{O}(q^{-2})$	**Quadratic advantage!**
Classical	$\mathcal{O}(q^{-1})$	

- State preparation of relativistic Breit-Wigner distributions

State preparation of probability distributions

- Preparing arbitrary probability distributions on a quantum computer thought to be computationally hard in general - no 'silver bullet' methodology
- Bottleneck for many quantum algorithms

Problem

- Load distribution into amplitude of n-qubit quantum state:

$$|p\rangle = \sum_x \sqrt{p(x)}|x\rangle$$

- Distribution discretised and truncated to $N = 2^n$ support points over $[a, b]$, with steps $\Delta = (b-a)/N$ and $x_0 = a + \frac{\Delta}{2}$, $x_i = x_0 + i\Delta$

Systematic errors

Discretisation error

1. $\epsilon_d = \left| \int_{x_l}^{x_u} f(x)p(x)\,dx - \sum_{i=0}^{N-1} f(x_i)p(x)(x_i)\Delta \right|$

Normalisation error

2. $\epsilon_n = \sum_{i=0}^{N-1} \left| f(x_i)\big(p(x_i)\Delta - \tilde{p}(x_i)\big) \right|$

Thresholding error

3. $\epsilon_{th} = |E[X\,\Theta(X \geq V_{Th})] - E[X\,\Theta(X \geq x_i)]|$

State-preparation error

4. $\epsilon_s^{\mathrm{CMSE}} = \frac{1}{N}\sum_{i=1}^{N} \left(\tilde{P}(x_i) - P^s(x_i)\right)^2$

Relativistic Breit-Wigner distributions

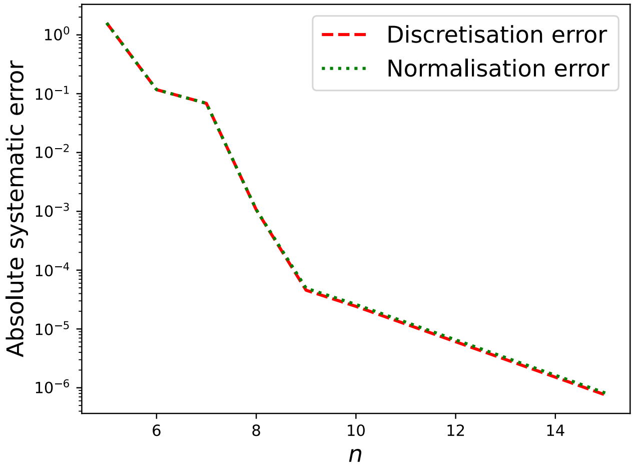
- For a given n, a sub-range of the full support of the BW distribution will exist where ϵ_d and ϵ_n are sufficiently minimised to give negligible impact
- Real calculation only performed for a given centre-of-mass (CoM) energy, $s_{\max}$, corresponding to integrating over a sub-range of the full support $[s_{\min} = 0 \text{ GeV}^2, s_{\max} = S \text{ GeV}^2]$

Strategy

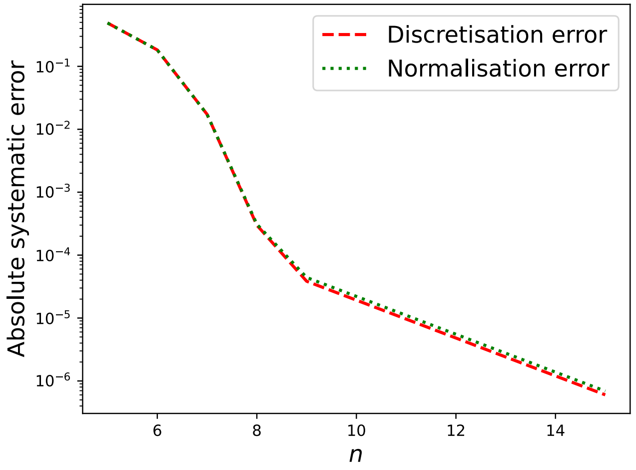
1. Generate circuits that prepare BW distributions for the resonances W, Z, t
2. Choose a range of supports spanning a range of different CoM energies
3. Set qubit numbers to sufficiently suppress systematic errors in each case

Systematic uncertainty scaling

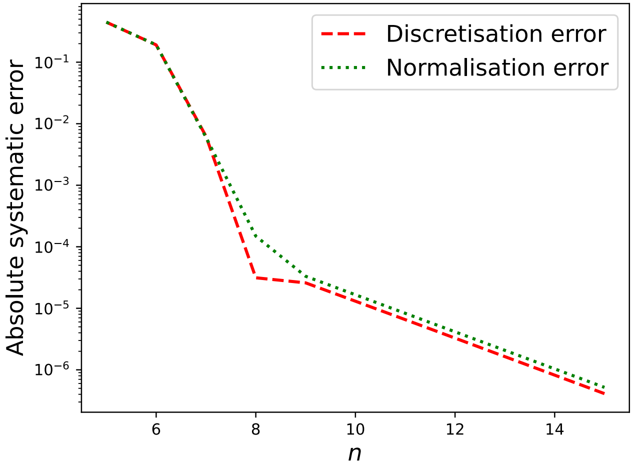
$s_{\max} = 200\ \mathrm{GeV}^2$



(a) W boson



(b) Z boson

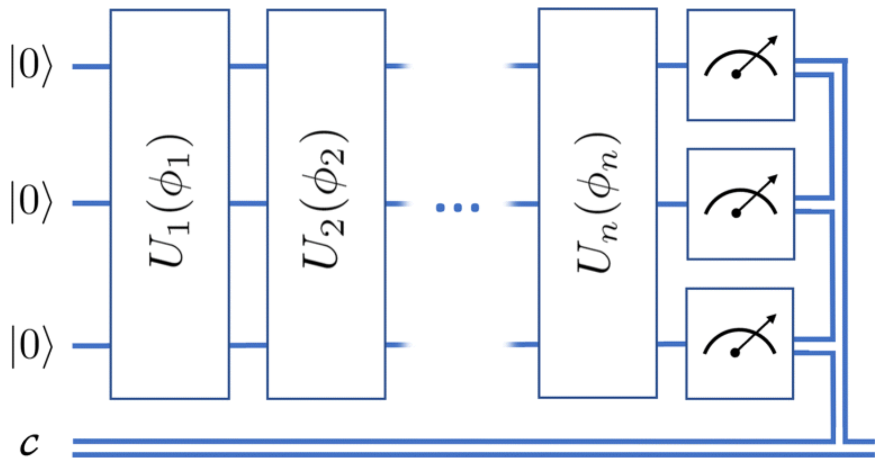


(c) t quark

State-preparation methods

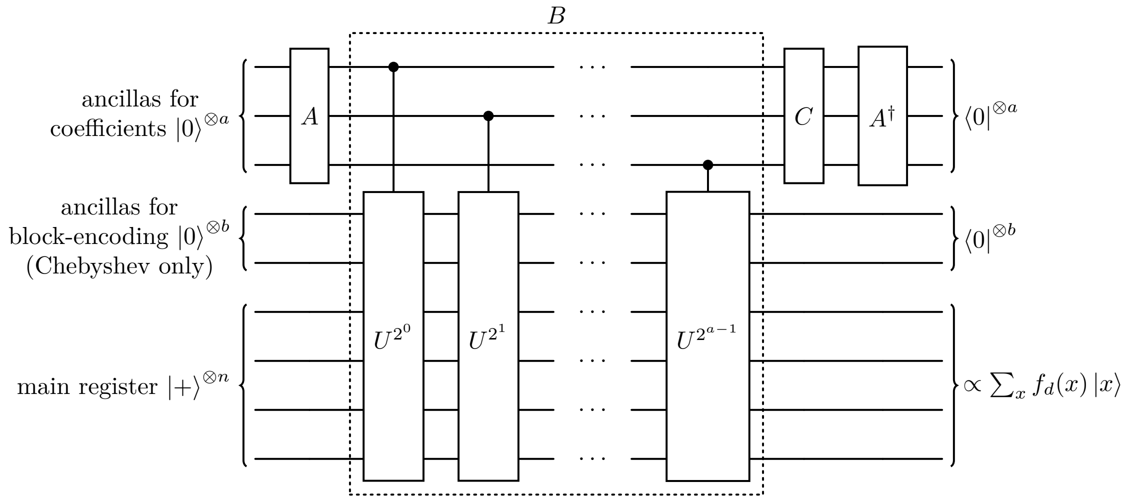
Variational

- Quantum machine-learning (QML) approach - train parameterised quantum circuit to generate target distribution
- Flexible, small circuits - fast training
- Limited scalability due to trainability issues in QML – works well for small-scale systems



Fourier expansion

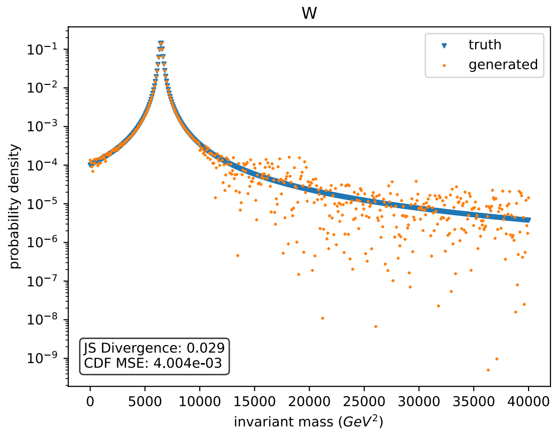
- Decompose distribution into Fourier series and use linear combination of unitary operations to form weighted Fourier sum
- Scalable method for larger systems - larger resource requirements
- Probabilistic preparation – requires post-selection



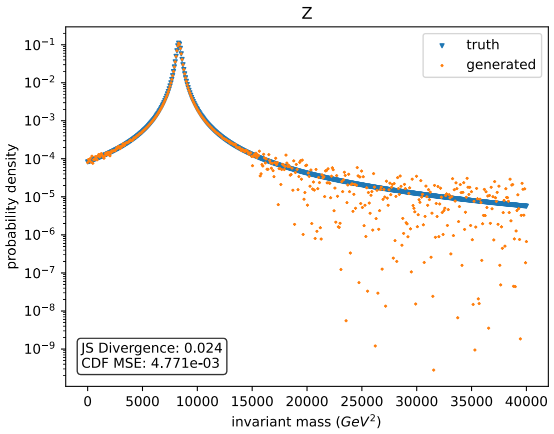
Performance comparison

$s_{\max} = 200\ \mathrm{GeV}^2$

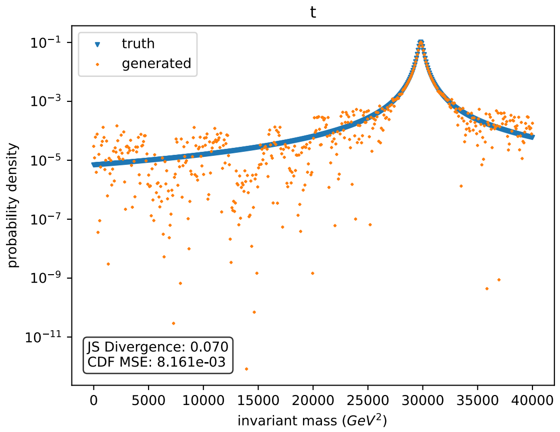
Variational



(a) W boson



(b) Z boson

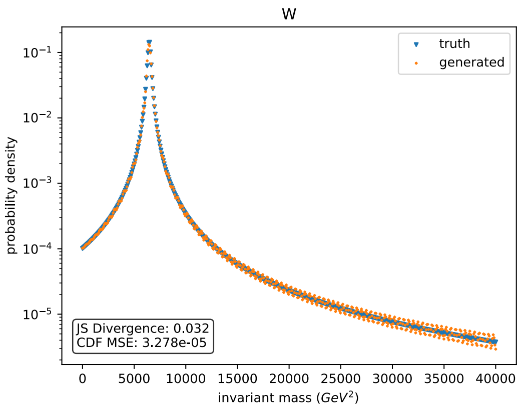


(c) t quark

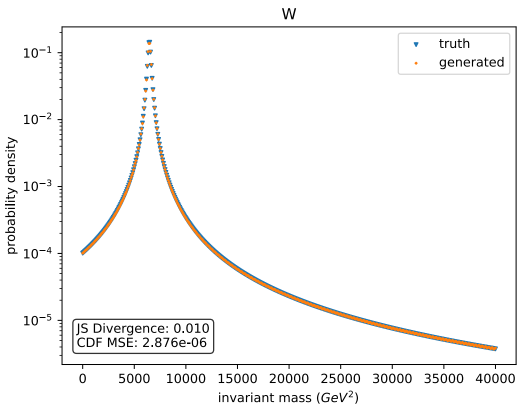
Performance comparison

$s_{\max} = 200\ \mathrm{GeV}^2$

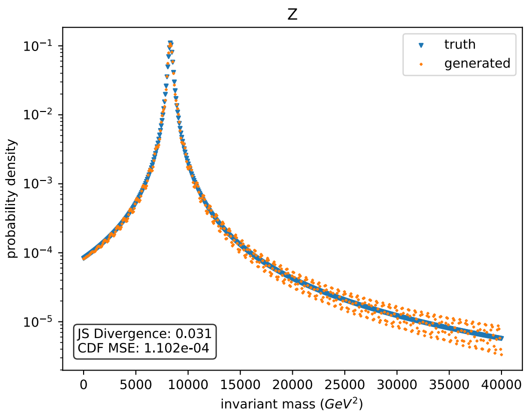
Fourier expansion



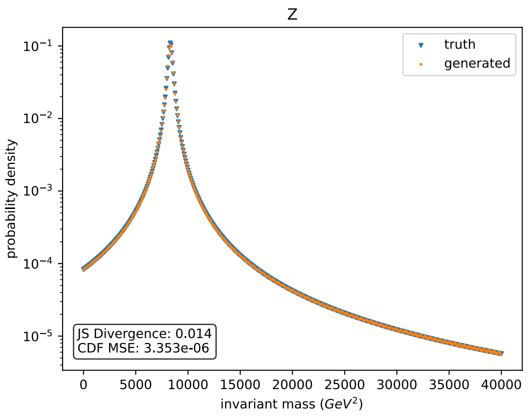
(a) $d = 175$



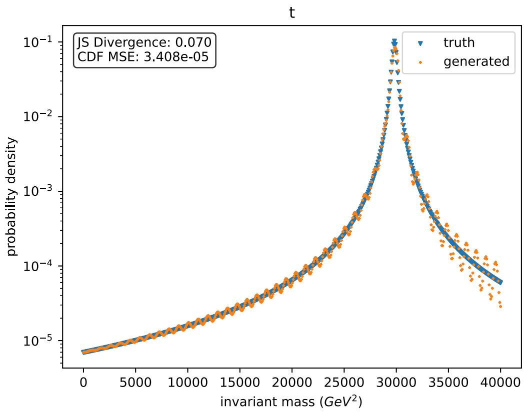
(b) $d = 250$



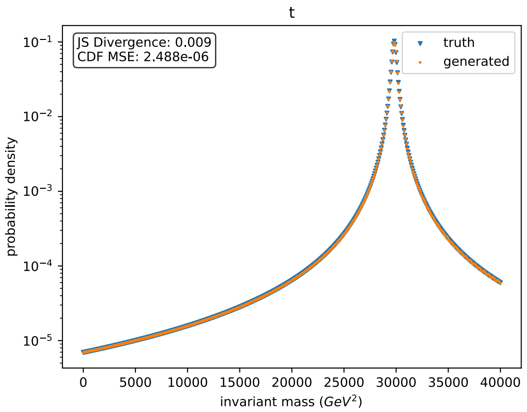
(a) $d = 130$



(b) $d = 170$



(a) $d = 85$



(b) $d = 180$

Performance comparison

Circuit used for example applications

$\sqrt{s_{max}}$	**Method**	**Res**	**Accuracy**	n	g_{1q}	g_{2q}	$\epsilon_s^{\mathbf{CMSE}}$	**JSD**	p_{success}
100 GeV	Variational	W	Optimised	6	186	150	1.22×10^{-4}	3.31×10^{-6}	N/A
	Fourier	W	Matched ($d = 250$)	15	1592	1638	1.22×10^{-4}	6.73×10^{-5}	3.12%
200 GeV	Variational	W	Optimised	9	180	152	4.01×10^{-4}	0.029	N/A
		Z	Optimised	9	234	200	4.77×10^{-3}	0.024	N/A
		t	Optimised	9	126	104	8.16×10^{-3}	0.070	N/A
	Fourier	W	Matched ($d = 175$)	18	1576	1684	3.28×10^{-5}	0.032	0.9%
			More ($d = 250$)	18	1602	1686	2.13×10^{-6}	0.003	0.9%
		Z	Matched ($d = 130$)	18	1591	1692	3.28×10^{-5}	0.031	1.1%
			More ($d = 170$)	18	1601	1686	2.58×10^{-6}	0.014	1.2%
		t	Matched ($d = 85$)	17	825	906	3.41×10^{-5}	0.070	1.4%
			More ($d = 180$)	18	1593	1692	2.45×10^{-6}	0.010	1.2%

- Example application

Simplified 1D integration

- Simplified example, considering only numerator of the matrix element (excluding phase-space integral)

$$\sigma \propto \int_0^s ds_2 \int_0^{s-s_2} ds_1 (s_1^2 - s_1 M_\tau^2)$$

$\sqrt{s_{\max}} = M_\tau = 1.776\text{GeV}$

- Rewrite expression (pre-calculating trivial integration over s_2) as

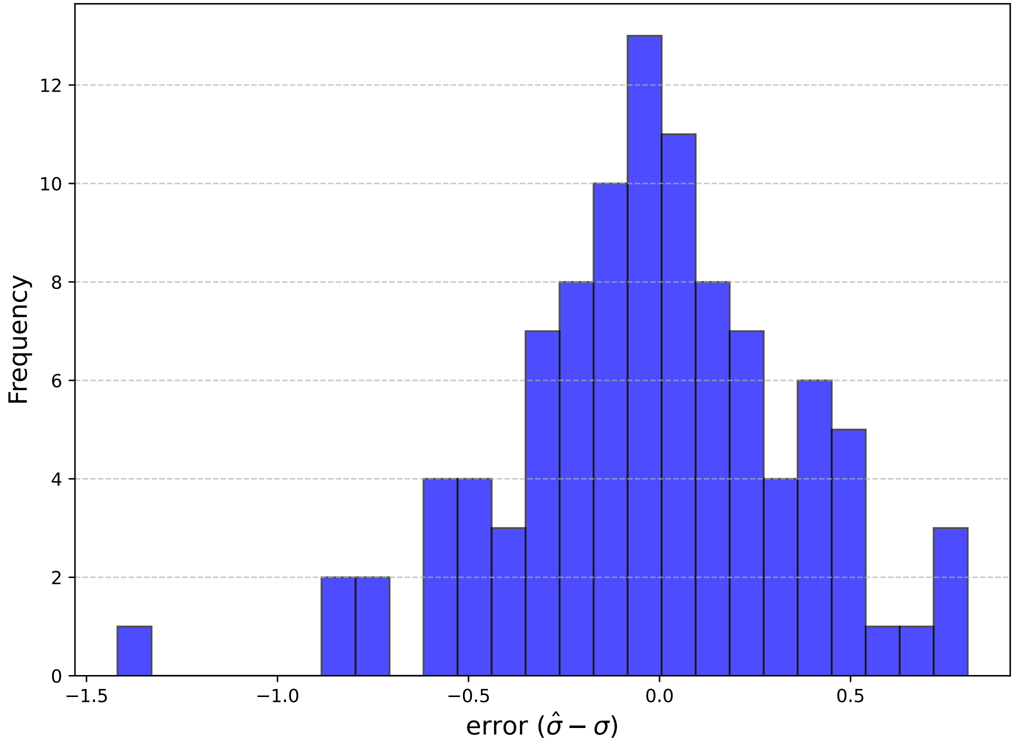
$C(s_1, s_2) = 1 \text{ if } s_1 + s_2 < s,$ else $= 0$; $C(s_1, s_2) = 1 \text{ if } s_1 + s_2 < s,$ else $= 0$

$$\sigma \propto s\left(\int_0^s ds_1 s_1^2 C(s_1, s_2) - M_\tau^2 \int_0^s ds_1 s_1 C(s_1, s_2)\right)$$

$f(s_1, s_2) = s_1^2$; $f(s_1, s_2) = s_1$

$p(s_1, s_2) = U(s_1)U(s_2)$; $p(s_1, s_2) = U(s_1)U(s_2)$

Simplified 1D integration



$\sigma = -8.248$

Compilation	Resource	Metric	Precision 10%	Precision 1%	Precision 0.1%
NISQ	Number of qubits	Largest across circuits	24	24	24
	CX gates	Total number across circuits	3.99×10^6	4.68×10^7	6.26×10^8
		Total depth across circuits	2.52×10^6	2.94×10^7	3.93×10^8
		Number in largest circuit	1.80×10^6	1.14×10^7	1.69×10^8
		Depth of largest circuit	1.13×10^6	7.18×10^6	1.06×10^8
	All gates	Total number across circuits	7.97×10^6	9.34×10^7	1.25×10^9
		Total depth across circuits	4.71×10^6	5.51×10^7	7.37×10^8
		Number in largest circuit	3.59×10^6	2.28×10^7	3.37×10^8
		Depth of largest circuit	2.12×10^6	1.35×10^7	1.99×10^8
Fault tolerant	Number of qubits	Largest across circuits	35	35	35
	T gates	Total number across circuits	8.62×10^6	3.49×10^8	5.85×10^9
		Total depth across circuits	7.21×10^6	2.92×10^8	4.89×10^9
		Number in largest circuit	3.83×10^6	9.62×10^7	1.71×10^9
		Depth of largest circuit	3.21×10^6	8.06×10^7	1.43×10^9

Non-separable 2D integration

- Extension increasing complexity of the problem and mimicing more general case of multivariate integration

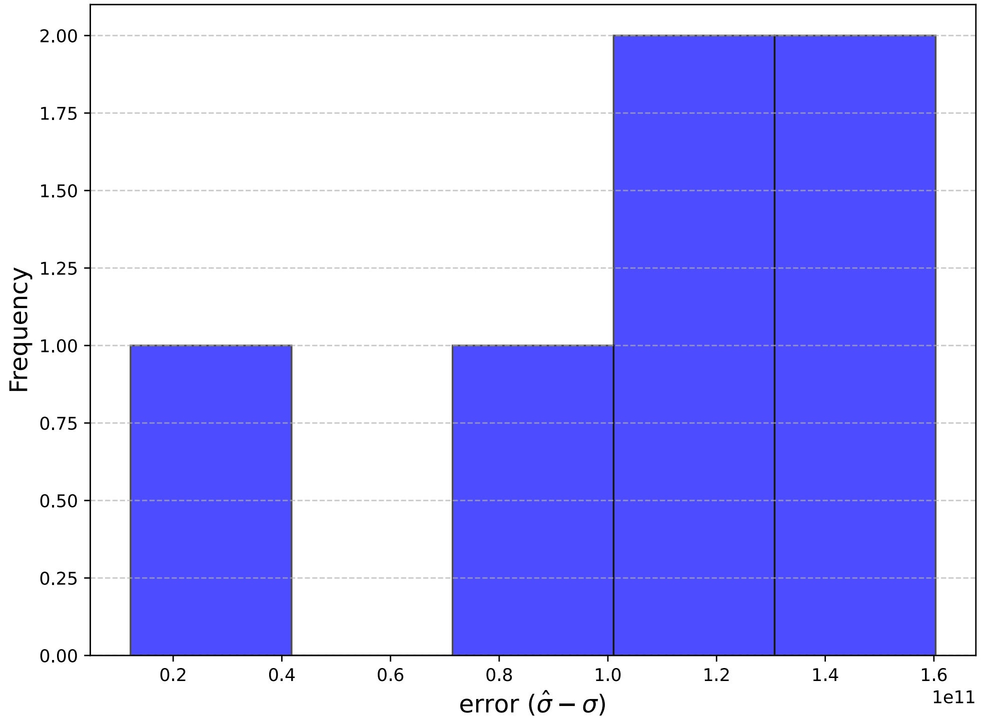
$$\sigma \propto \int_0^{s} \int_0^{s-s_2} ds_1 ds_2 \frac{s_1^2 s - s_1 M_\tau^2 s + s_1 M_\tau^2 s_2}{(s_2 - M_W^2)^2 + (M_W \Gamma_W)^2}$$

$\sqrt{s_{\max}} = 100\mathrm{GeV}$

- Amounts to computing additional multivariate term

$$I_1 = \int_0^{s} \int_0^{s-s_2} ds_1 ds_2 \frac{s_1 M_\tau^2 s_2}{(s_2 - M_W^2)^2 + (M_W \Gamma_W)^2}$$

Non-separable 2D integration



$\sigma = 3.179 \times 10^{12}$

Compilation	Resource	Metric	Precision 10%	Precision 1%	Precision 0.1%
NISQ	Number of qubits	Largest across circuits	28	28	28
	CX gates	Total number across circuits	7.39×10^7	6.15×10^8	5.09×10^9
		Total depth across circuits	4.34×10^7	3.61×10^8	2.99×10^9
		Number in largest circuit	3.39×10^7	2.71×10^8	1.20×10^9
		Depth of largest circuit	1.99×10^7	1.59×10^8	7.03×10^8
	All gates	Total number across circuits	1.50×10^8	1.24×10^9	1.03×10^{10}
		Total depth across circuits	8.02×10^7	6.67×10^8	5.52×10^9
		Number in largest circuit	6.86×10^7	5.49×10^8	2.42×10^9
		Depth of largest circuit	3.68×10^7	2.94×10^8	1.30×10^9
Fault tolerant	Number of qubits	Largest across circuits	41	41	41
	T gates	Total number across circuits	3.39×10^9	4.08×10^{10}	2.72×10^{11}
		Total depth across circuits	3.29×10^9	3.95×10^{10}	2.63×10^{11}
		Number in largest circuit	1.65×10^9	2.14×10^{10}	6.97×10^{10}
		Depth of largest circuit	1.60×10^9	2.07×10^{10}	6.75×10^{10}

Conclusions

Current limitations

- Currently practical only for 2D integrals — scalability to higher dimensions remains a major hurdle (based on current FQMCI methods)
- Uniform spacing for representing underlying probability distributions with qubits limits flexibility and efficiency
- Complex kinematic cuts and realistic phase spaces remain largely unexplored
- Approach restricted to specific cross-section integrand forms — generalisation is still lacking
- Accuracy bottlenecked by state-preparation fidelity for Breit-Wigner distributions — tailored methods are needed
- Systematic errors in state preparation not yet fully understood — a barrier to precision
- Resource estimates indicate that even simple tree-level cases require FT hardware — impractical for near-term quantum devices, especially for high-dimensional problems encountered in state-of-the-art classical calculations