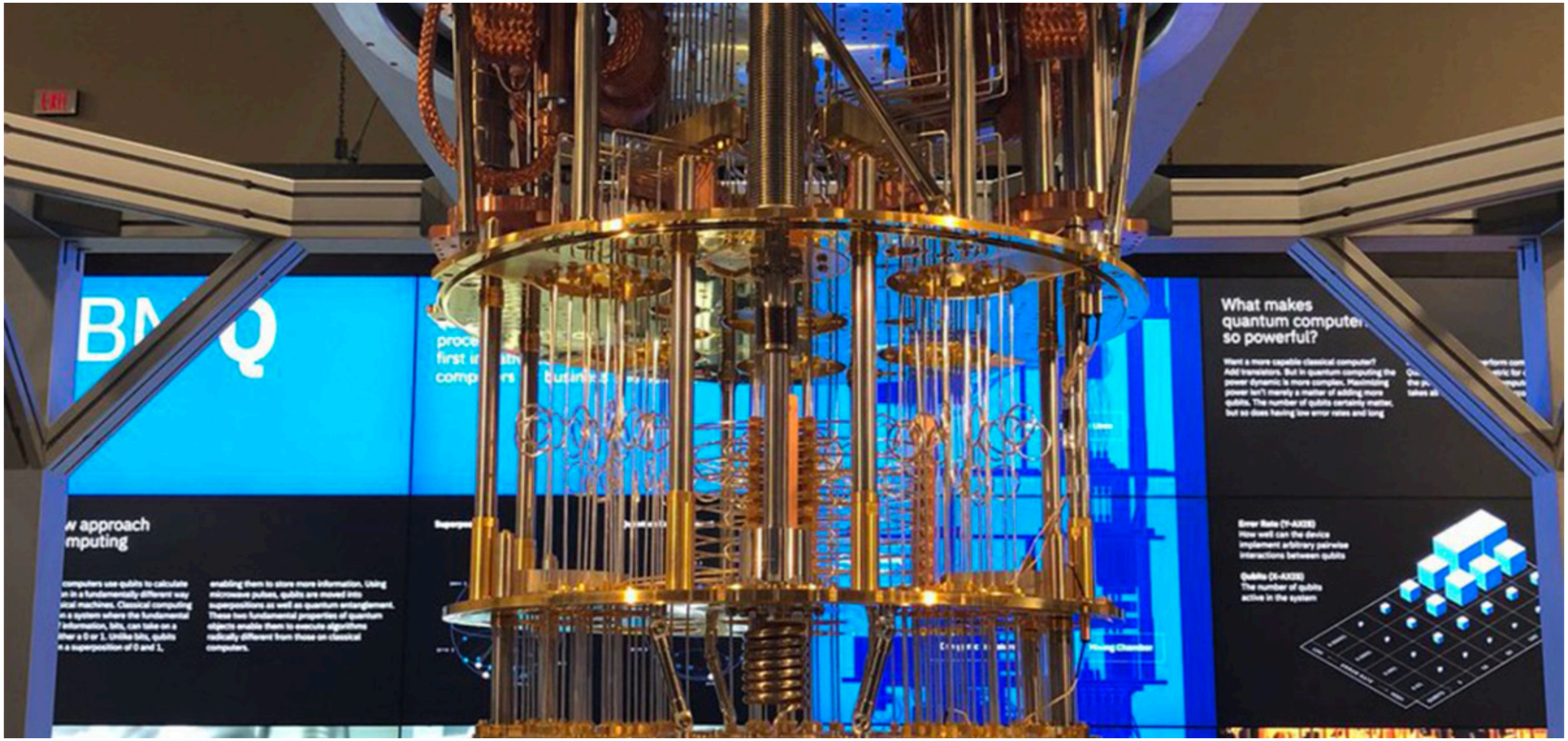


A quantum algorithm for high energy physics simulations

Christian Bauer, Wibe de Jong, Ben Nachman, Davide Provasoli

1904.03196

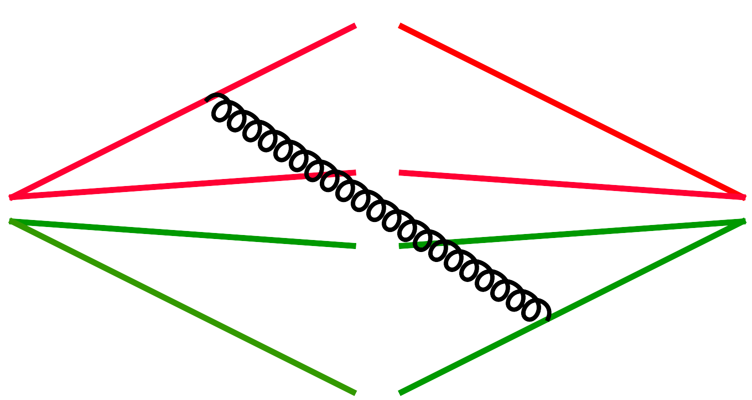
Interference effects are difficult to model in parton shower simulations

Parton showers are based on probabilistic MC approaches

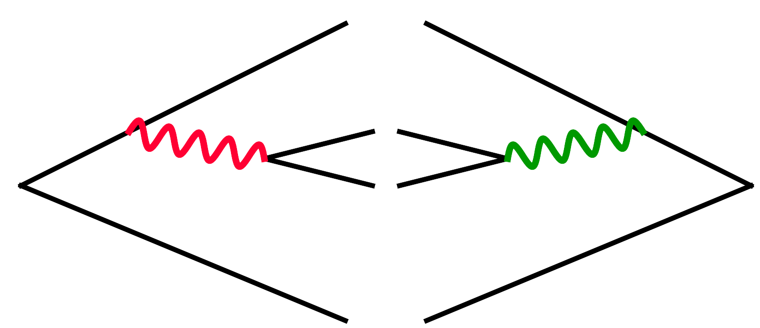
Quantum interference effects not easily included in formalism

Several interference effects can appear

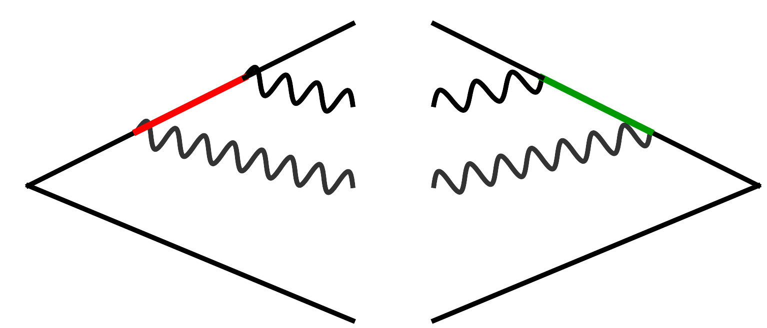
$1/N_C$ effects in dipole showers



γ/Z interference in EW showers



CKM interference in EW showers

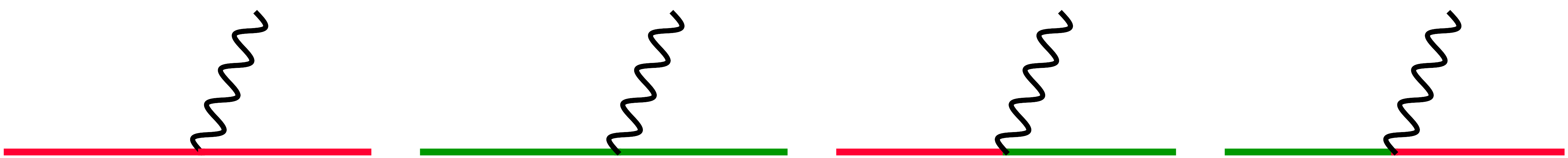


Consider a simpler toy model that exhibits interference effects similar to the CKM case

Yukawa theory with two types of fermions and mixing between them

$$\mathcal{L} = \bar{f}_1(i\partial\!\!\!/ + m_1)f_1 + \bar{f}_2(i\partial\!\!\!/ + m_2)f_2 + (\partial_\mu\phi)^2 + g_1\bar{f}_1 f_1\phi + g_2\bar{f}_2 f_2\phi + g_{12}\left[\bar{f}_1 f_2 + \bar{f}_2 f_1\right]\phi$$

Very simple Feynman rules



g_1 g_2 g_{12} g_{12}

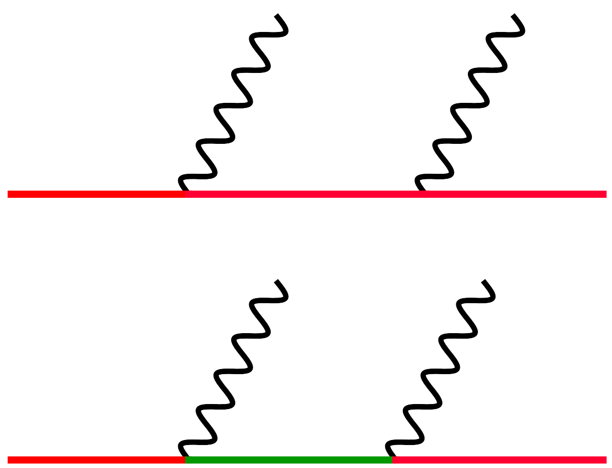
Many similarities with the CKM interference

Consider a simpler toy model that exhibits interference effects similar to the CKM case

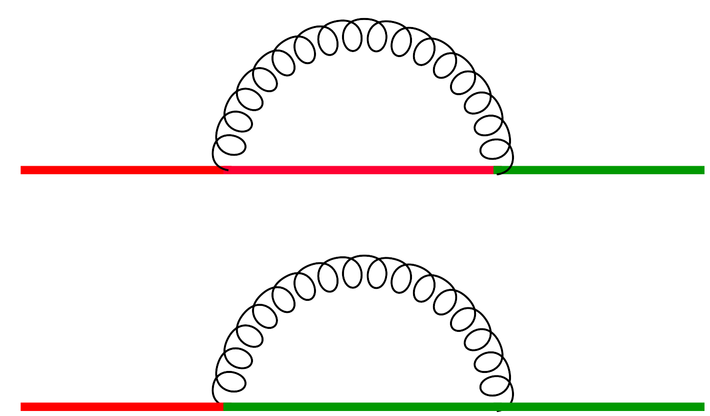
$$\mathcal{L} = \bar{f}_1(i\not{\partial} + m_1)f_1 + \bar{f}_2(i\not{\partial} + m_2)f_2 + (\partial_\mu\phi)^2 \\ + g_1\bar{f}_1 f_1\phi + g_2\bar{f}_2 f_2\phi + g_{12}\left[\bar{f}_1 f_2 + \bar{f}_2 f_1\right]\phi$$

The mixing g_{12} gives several interesting effects

Different real emission amplitudes give rise to interference



Virtual diagrams give rise to flavor change without radiation



Need to correct both real and virtual effects

Similar to including subleading color

In the high energy limit where masses can be ignored, can diagonalize this problem

Interaction can be written in matrix notation

$$(\bar{f}_1, \bar{f}_2) \begin{pmatrix} g_1 & g_{12} \\ g_{12} & g_2 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \phi$$

This can be diagonalized as

$$(\bar{f}_1, \bar{f}_2)\, U^\dagger \begin{pmatrix} g_1 & g_{12} \\ g_{12} & g_2 \end{pmatrix} U \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \phi \equiv (\bar{f}_a, \bar{f}_b) \begin{pmatrix} g_a & 0 \\ 0 & g_b \end{pmatrix} \begin{pmatrix} f_a \\ f_b \end{pmatrix} \phi$$

$$g_a = \frac{g_1 + g_2 - g'}{2}\,, \qquad g_b = \frac{g_1 + g_2 + g'}{2}\,, \qquad g' = \mathrm{sign}(g_2 - g_1)\sqrt{(g_1 - g_2)^2 + 4g_{12}^2}$$

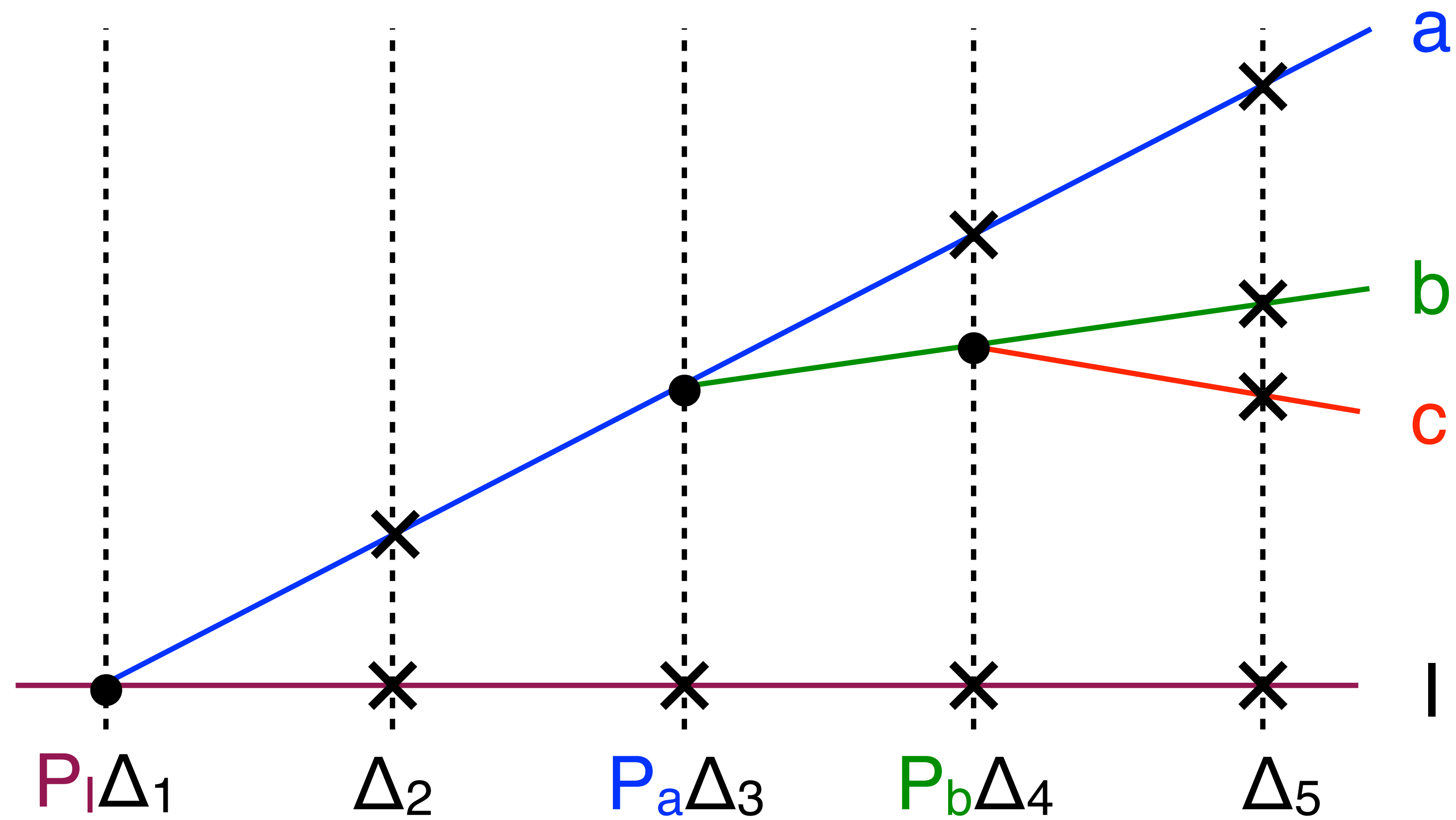
$$U = \begin{pmatrix} \sqrt{1-u^2} & u \\ -u & \sqrt{1-u^2} \end{pmatrix}\,, \qquad u = \sqrt{\frac{(g_1 - g_2 + g')}{2g'}}$$

Thus, the theory can be transformed into a system of non-interacting fermions

This allows to compute splitting amplitudes using insight from parton showers

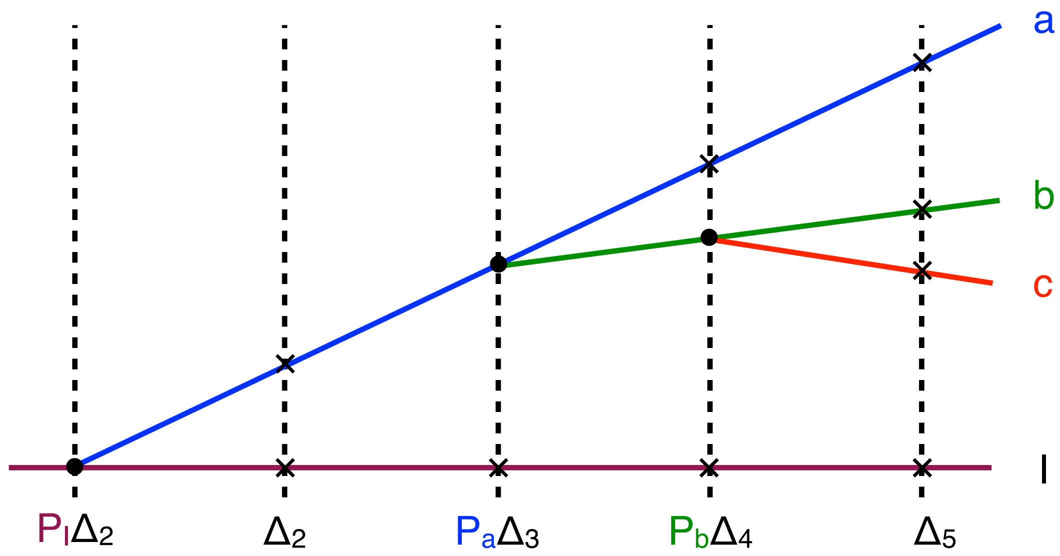
Parton showers are defined from splitting function and Sudakov factors and depend on an evolution variable

Discretize evolution variable and define (no-) emission probabilities (P) Δ



Emission depends on P of particle that emits and Δ of system at time t_i

However, computing the final result is exponentially hard in the number of final state particles



- Δ_i only depends on n_a, n_b, but different for each i
- P_α depends on flavor of each particle, but independent of i

There are two important facts to realize:

1. We need to rotate back to the f_1, f_2 basis in the end, so need to compute amplitudes, not probabilities
2. Need the results for all possible final state particles f_a, f_b

This means that for each shower history, need amplitudes for all possible flavors of fermions

This grows like 2^N for N fermions

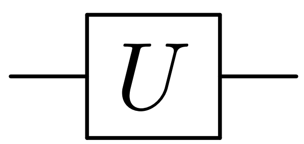
Since most of you have not thought about quantum computing, here is a very short primer

Every state is represented by quits. Examples are

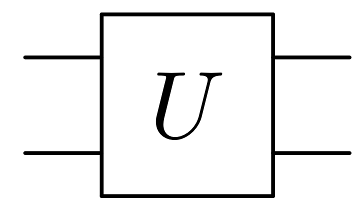
$$f_a = |0\rangle\,, f_b = |1\rangle\,, \qquad |5\rangle = |101\rangle\,, \qquad \text{true} = |0\rangle\,, \text{false} = |1\rangle$$

We operate on these states with unitary operations (matrices)

2x2 matrix U
on single qbit

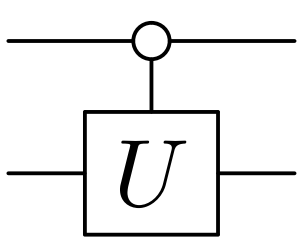


4x4 matrix U
on pair of qbits

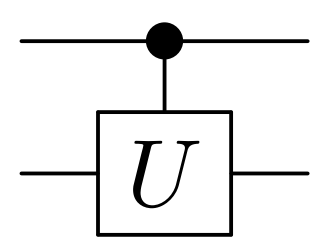


All Unitary operations can be built out of a very small set of basic operations

Can perform controlled operations



$$= U \otimes |0\rangle \langle 0|$$

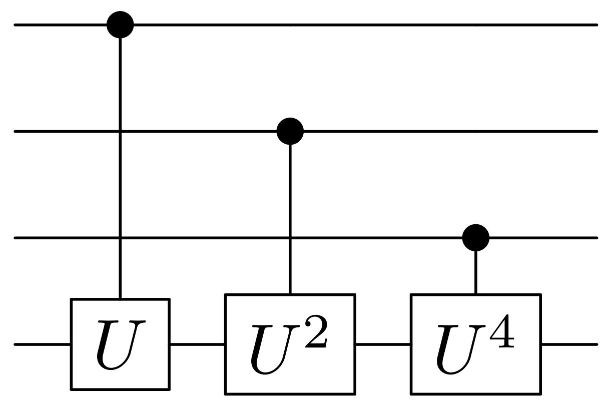


$$= U \otimes |1\rangle \langle 1|$$

Since most of you have not thought about quantum computing, here is a very short primer

On a quantum computer, can perform many calculations at once

Consider following circuit:



It performs operation

$$|i\rangle\,|q\rangle \to |i\rangle\,U^j\,|q\rangle$$

(think binary)

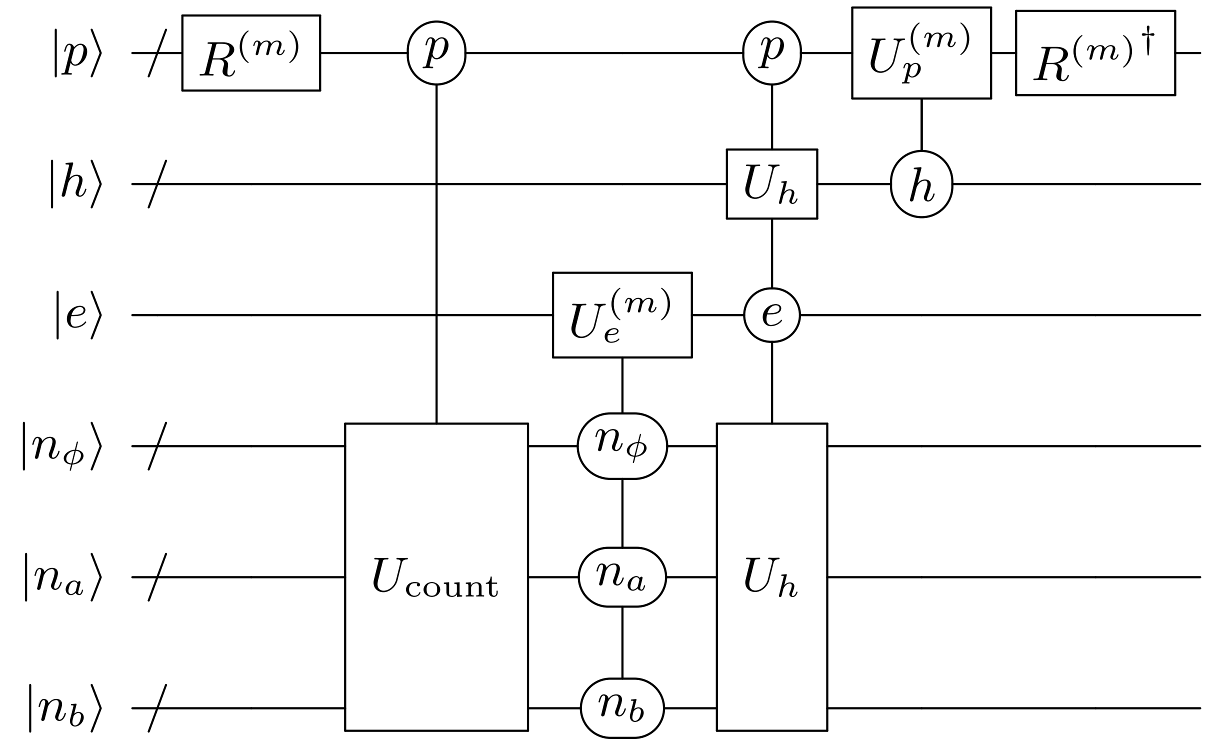
If start with superposition of all integers, perform 8 calculation with only 3 gates

$$\sum_i \alpha_i\,|i\rangle\,|q\rangle \to \sum_i \alpha_i\,|i\rangle\,U^j\,|q\rangle$$

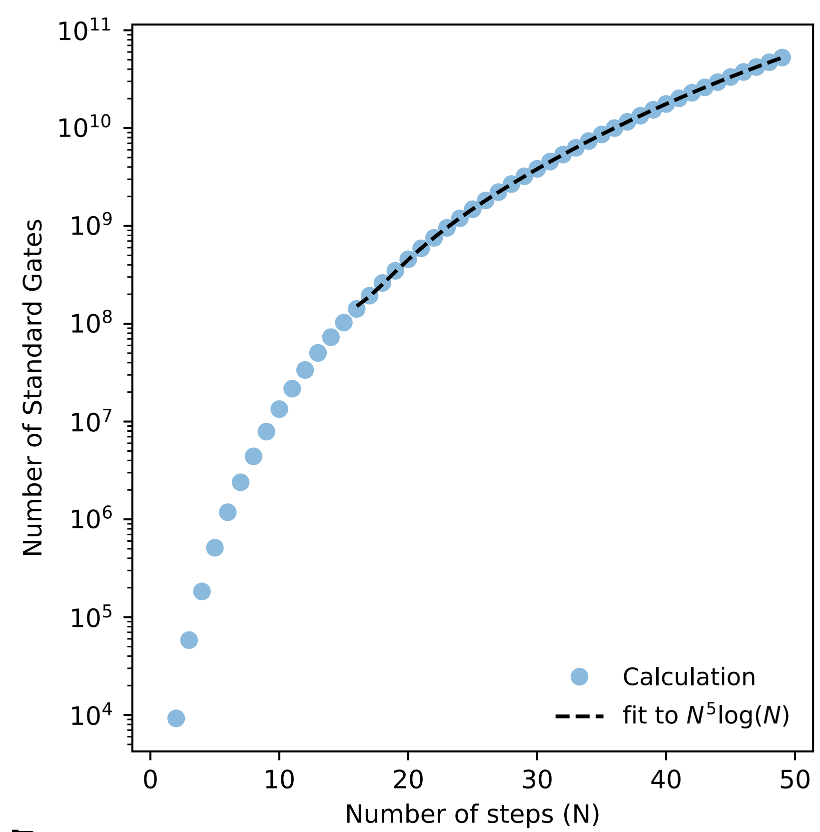
However, in the end can only measure one of the 8 possible states, so need to think carefully how to use this parallelism

A quantum computer can compute the 2^N amplitudes using polynomial number of operators

At each discreet time interval, algorithm rotates from f_1, f_2 basis to f_a, f_b basis, performs shower in 4 separate steps, and rotates back to f_1, f_2 basis



Operation	Scaling
count particles U_{count}	N lnN
decide emission U_e	N^4 lnN
create history U_h	N^5 lnN
adjust particles U_p	N^2 lnN



Christian Bauer

A quantum algorithm for high energy physics simulations

The basic idea of the 4 steps can be understood quite easily from parton shower intuition

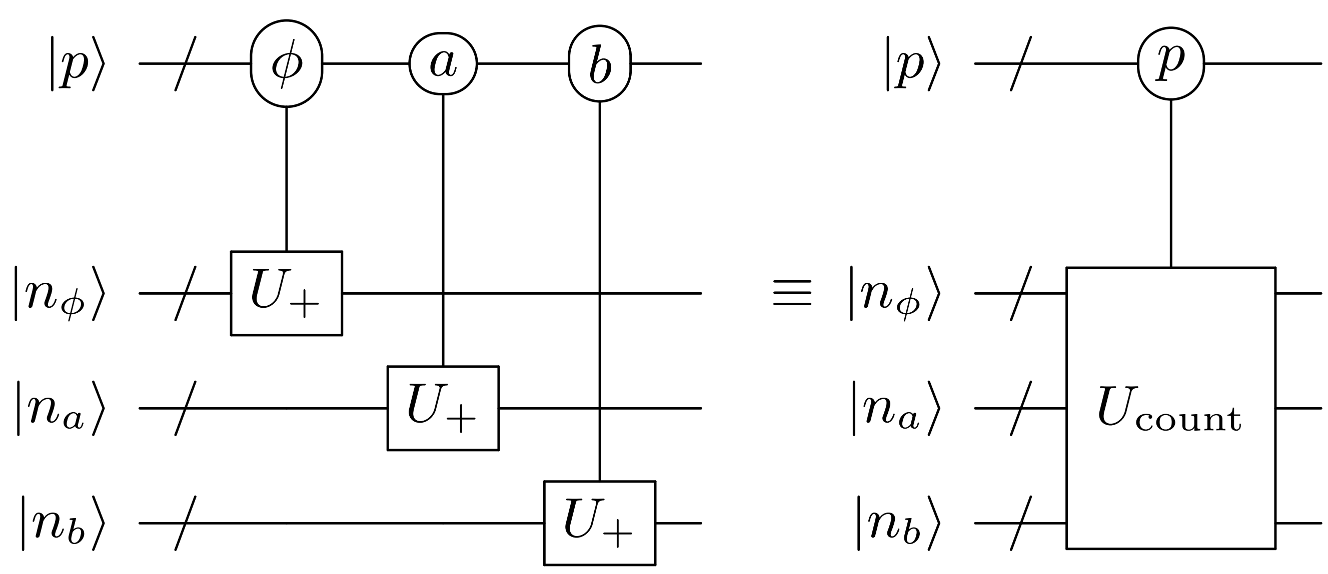
Operation	Scaling
count particles U_{count}	N lnN
decide emission U_e	N^4 lnN
create history U_h	N^5 lnN
adjust particles U_p	N^2 lnN

Count the particles

Total emission probability is product of Sudakov factor for each particle

$$\Delta^{(m)} = [\Delta_a(\theta_m)]^{n_a} [\Delta_b(\theta_m)]^{n_b} \Delta_\phi(\theta_m)]^{n_\phi}$$

Need information about the total number of particles of each type



Can show that this counting step is linear in n[qbits] = lnN

Total scaling is N lnN

The basic idea of the 4 steps can be understood quite easily from parton shower intuition

Operation	Scaling
count particles U_{count}	N lnN
decide emission U_e	N^4 lnN
create history U_h	N^5 lnN
adjust particles U_p	N^2 lnN

Decide emission (did it happen or not?)

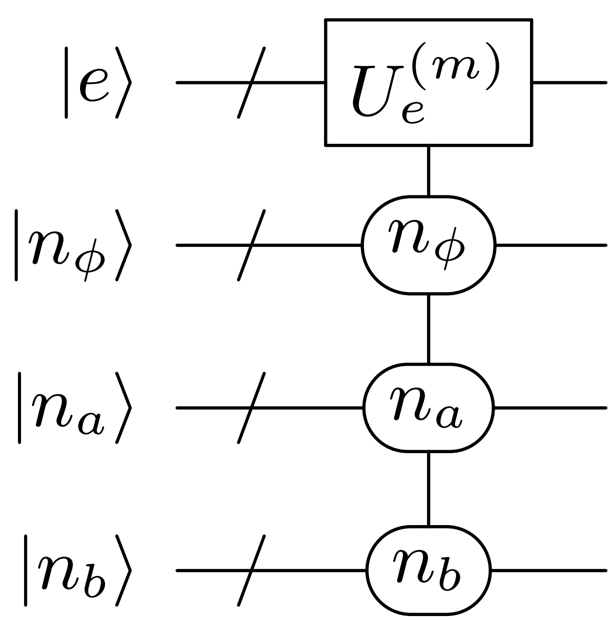
Emission can be written as simple Unitary transformation

$$U_e^{(m)} = \begin{pmatrix} \sqrt{\Delta^{(m)}(\theta_m)} & -\sqrt{1-\Delta^{(m)}(\theta_m)} \\ \sqrt{1-\Delta^{(m)}(\theta_m)} & \sqrt{\Delta^{(m)}(\theta_m)} \end{pmatrix}$$

where from before Sudakov factor depends on n_i

$$\Delta^{(m)} = [\Delta_a(\theta_m)]^{n_a} [\Delta_b(\theta_m)]^{n_b} \Delta_\phi(\theta_m)]^{n_\phi}$$

This can be implemented using a simple quantum circuit



Performing the controls on the n_i values is somewhat expensive (N^3 operations)

Scaling is N^4 lnN

The basic idea of the 4 steps can be understood quite easily from parton shower intuition

Operation	Scaling
count particles U_{count}	N lnN
decide emission U_e	N^4 lnN
create history U_h	N^5 lnN
adjust particles U_p	N^2 lnN

Create history (which particle emitted?)

Once we know an emission happened, decide which particle did emission

Relative probability given by

$$\frac{P_k(\theta_m)}{\sum_k P_k(\theta_m)}$$

Complicated operation, which depends on type of all particles available $\Longrightarrow$ exponentially possible combinations

Does this imply exponential number of operations?

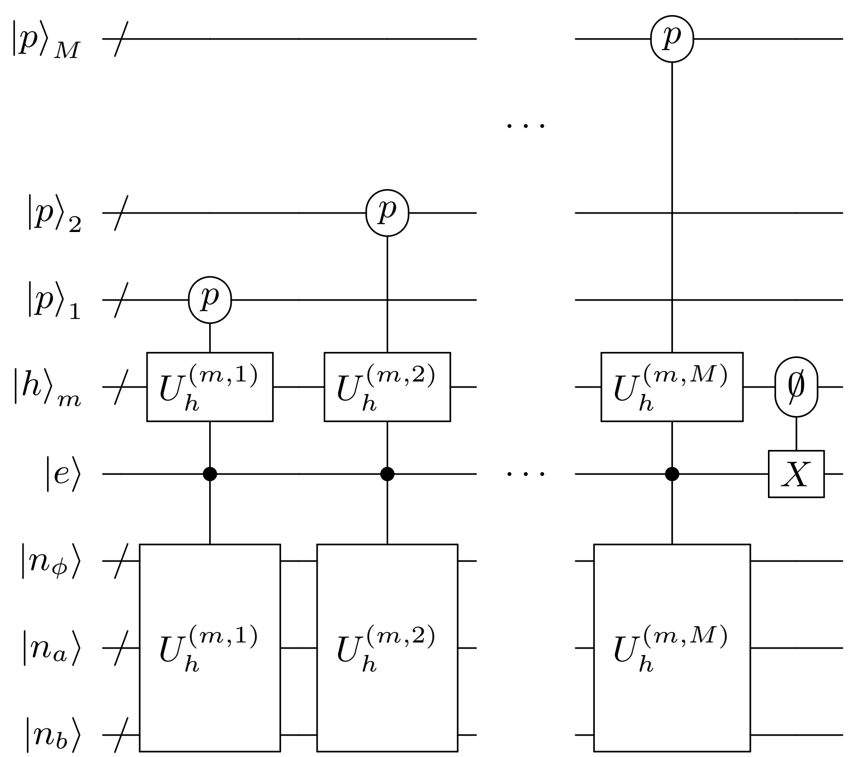
One can find an operation that only grows polynomially with N

The basic idea of the 4 steps can be understood quite easily from parton shower intuition

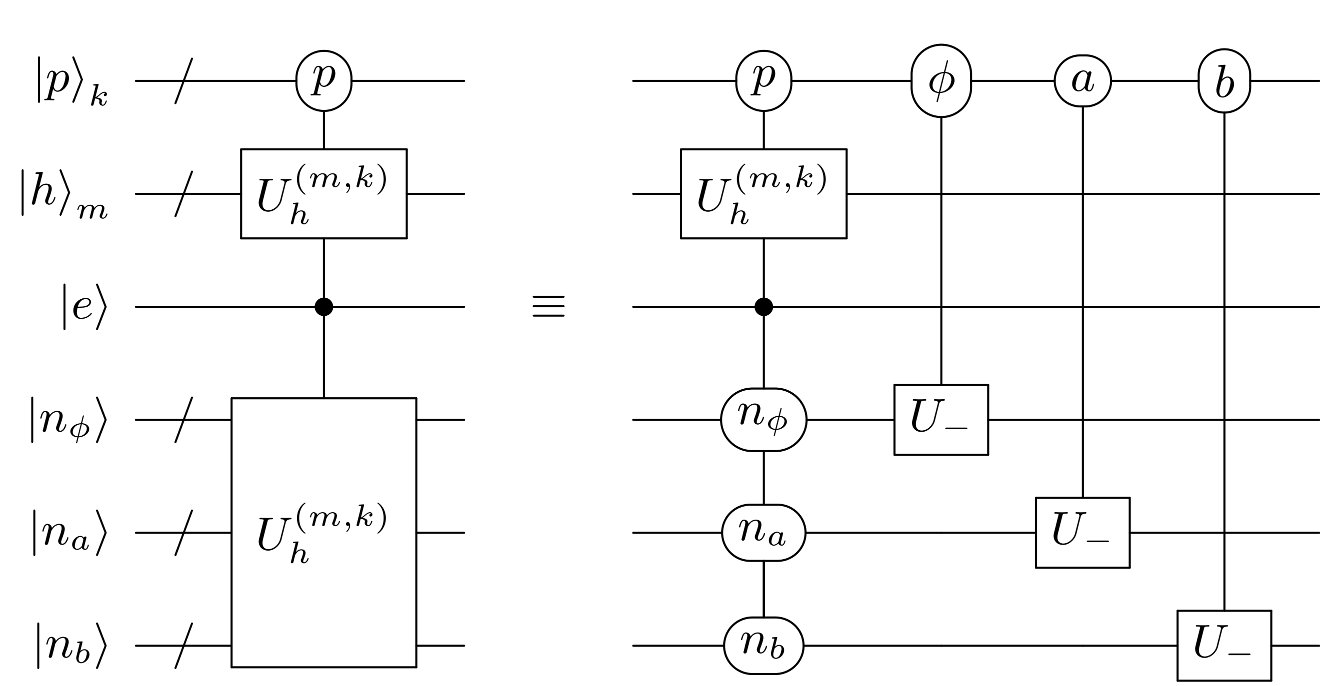
Operation	Scaling
count particles U_{count}	N lnN
decide emission U_e	N^4 lnN
create history U_h	N^5 lnN
adjust particles U_p	N^2 lnN

Create history (which particle emitted?)

Defining $\sum_k P_k(\theta_m) = P(n_a, n_b, n_\phi)$, the following algorithm does the correct thing



with



$$U_h^{(m,k)} = \begin{pmatrix} \sqrt{1 - \frac{P_{p_k}(\theta_m)}{P(n_\phi, n_a, n_b)}} & -\sqrt{\frac{P_{p_k}(\theta_m)}{P(n_\phi, n_a, n_b)}} \\ \sqrt{\frac{P_{p_k}(\theta_m)}{P(n_\phi, n_a, n_b)}} & \sqrt{1 - \frac{P_{p_k}(\theta_m)}{P(n_\phi, n_a, n_b)}} \end{pmatrix}$$

At each step, assigns the correct amplitude to integer k (particle p_k) and keeps unassigned in state l0>

Scaling is N^5 lnN

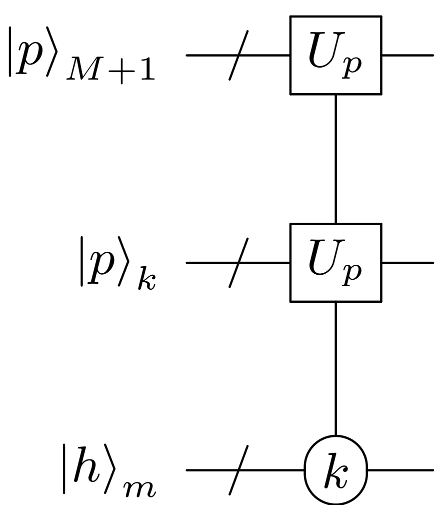
The basic idea of the 4 steps can be understood quite easily from parton shower intuition

Operation	Scaling
count particles U_{count}	N lnN
decide emission U_e	N^4 lnN
create history U_h	N^5 lnN
adjust particles U_p	N^2 lnN

Adjust particles (adjust the flavors based on the emission)

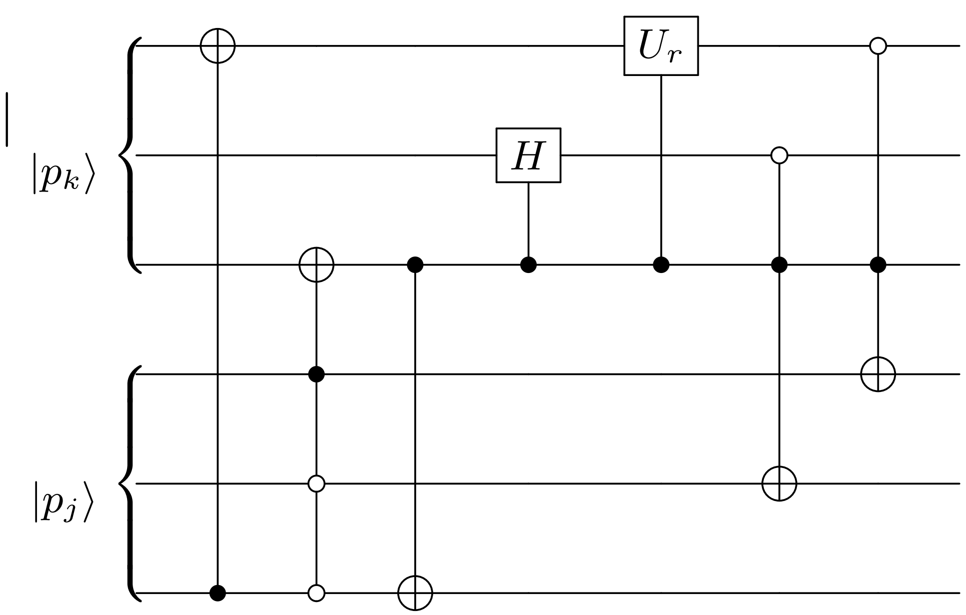
If the particle that emitted is a fermion, add a boson. If emitted particle is boson, add fermion and change boson to fermion

Can be implemented rather easily



$$U_p = \sum_{i=a,b} |f_i\rangle |\phi\rangle \langle f_i| \langle 0| + \sum_{i=1,b} |\bar{f}_i\rangle |\phi\rangle \langle \bar{f}_i| \langle 0|$$
$$+ \sum_{i=a,b} \hat{g}_i \left(|f_i\rangle |\bar{f}_i\rangle + |\bar{f}_i\rangle |f_i\rangle \right) \langle \phi| \langle 0|$$

with $$\hat{g}_i \equiv \frac{g_i}{\sqrt{2(g_a^2 + g_b^2)}}$$



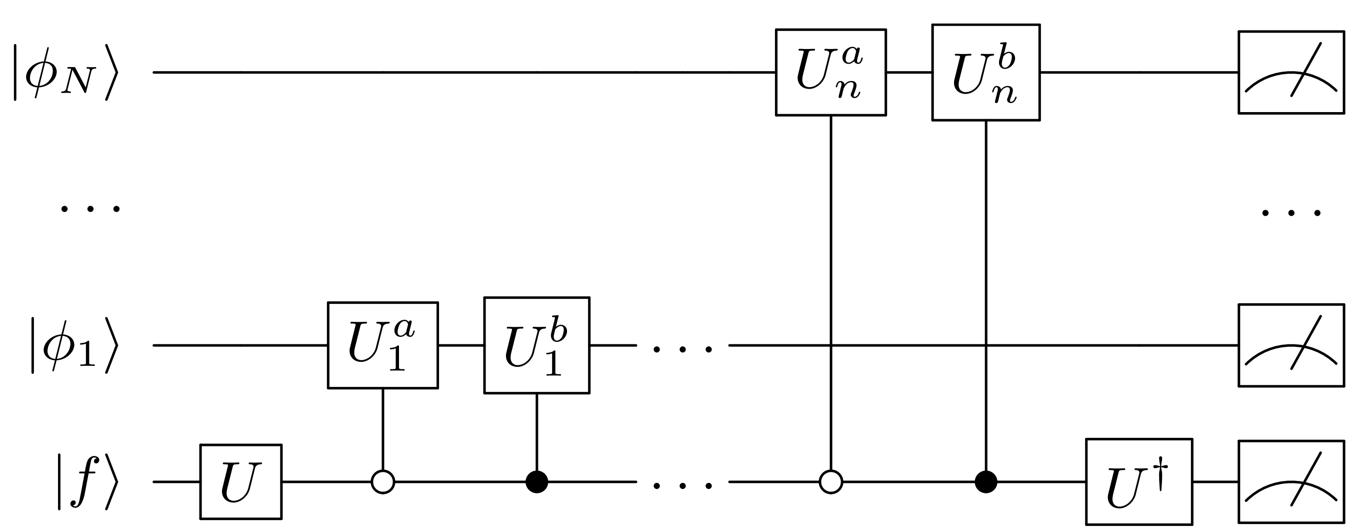
Scaling is N^2 lnN

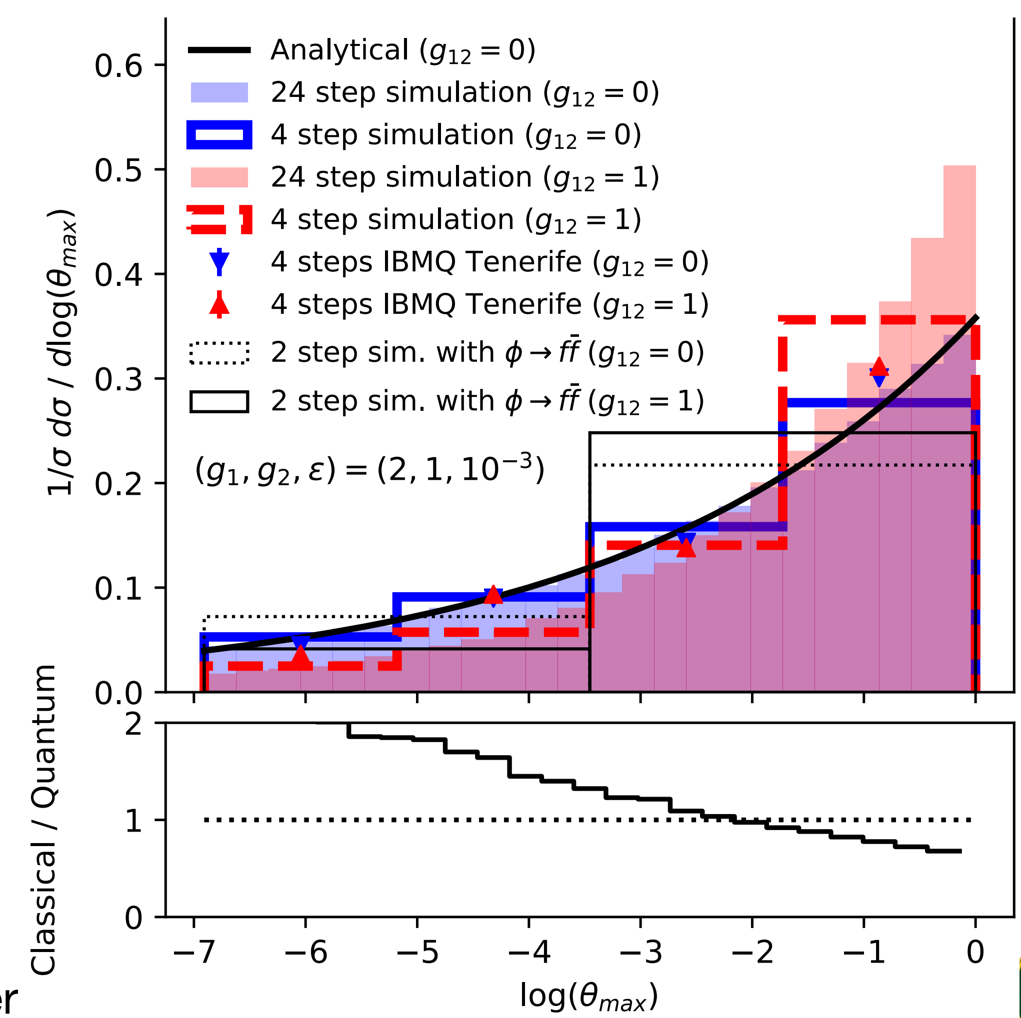
Number of qbits still larger than what is available on current hardware. But we can simulate something simpler...

Removing ϕ → ff splittings and starting from single fermion simplifies things a lot, since only one fermion throughout whole evolution process

1. No need to keep track of numbers n_a, n_b, n_ϕ
2. All emission probabilities equal (determined by single fermion)
3. Since only the one fermion can split, no need to track history
4. Every emission simply adds one fermion

Much simpler algorithm





There are many more things that can be done using quantum computers, and it is fun to explore possibilities

1. There are many basic operations that are know how to do on a QC
   - Fourier Transform
   - Measuring phase of an eigenstate
   - Time evolution of a quantum mechanical system
   - ...
2. In fact, it was shown how one can simulate an entire QFT on a quantum computer
   - Shown for both scalars and fermions
   - Works at least in principle for strongly interacting theories
   - Resource requirements prohibitively expensive

Very interesting to figure out what is possible with this completely different way to perform computations