Imperial College London

Collider Events on a Quantum Computer

G. Gustafson[1], S. Prestel[1], M. Spannowsky[2] and **S. Williams**[3].

[1]Department of Astronomy and Theoretical Physics, Lund University, S-223 62 Lund, Sweden.
[2]Institute for Particle Physics Phenomenology, Department of Physics, Durham University.
[3]High Energy Physics Group, Blackett Laboratory, Imperial College, London.

Abstract

High-quality simulated data is crucial for particle physics discoveries. Therefore, parton shower algorithms are a major building block of the data synthesis in event generator programs.

With quantum computers' rapid and continuous development, dedicated algorithms are required to exploit the potential that quantum computers provide to address problems in high-energy physics.

This paper presents a novel approach to synthesising parton showers using the Discrete QCD method. The algorithm benefits from an elegant Quantum Walk implementation which can be embedded into the classical toolchain.

This is the first time a Noisy Intermediate-Scale Quantum (NISQ) device has been used to simulate realistic high-energy particle collision events.

Reinterpreting the Parton Shower

Parton shower processes evolve high-energy few-particle states to low-energy multi-particle states by successively decaying particles into lower-energy decay products.

Conventional, state-of-the-art parton shower algorithms generate physical multi-particle data through the decays of colour-anticolour dipoles.

This is achieved by implementing a single algorithm, the "veto algorithm", to numerically solve the parton shower master equation:

$$\mathcal{F}_n(\Phi_n, t_n, t_c; O) = \Delta(t_n, t_c)O(\Phi_n) + \int_{t_c}^{t_n} dt d\xi \frac{d\phi}{2\pi} C \frac{\alpha_s}{2\pi} \frac{2s_{ik}(t,\xi)}{s_{ij}(t,\xi)s_{jk}(t,\xi)} \Delta(t_n, t)\mathcal{F}_n(\Phi_{n+1}, t, t_c; O) \quad (1)$$

where C is the colour-charge factor, the two particle invariants $s_{ab} = 2p_a \dot{p}_b$, Δ is the no-branching probability, α_s is the strong coupling, and a particular choice of the functional form of t and ξ is called the phase space parameterisation. The parton shower master equation utilises the inclusive decay probability given by the eikonal interference pattern:

$$d\mathcal{P}\left(q(p_I)\bar{q}(p_K) \to q(p_i)g(p_j)\bar{q}(p_k)\right) \simeq \frac{ds_{ij}}{s_{IK}}\frac{ds_{jk}}{s_{IK}} C \frac{\alpha_s}{2\pi}\frac{2s_{IK}}{s_{ij}s_{jk}} \quad (2)$$

where the momentum conservation condition implies the relation $s_{IK} = s_{ik} + s_{ij} + s_{jk}$.

Current implementations of the veto-algorithm treat the variables t and ξ as continuous degrees of freedom. Therefore, such algorithms are not suited for current quantum computers. Consequently, other algorithmic solutions to solving Eq. 1 are required for the implementation of quantum parton shower algorithms.

Discrete QCD

As illustrated in Fig. (a) and (b), Discrete QCD abstracts the conventional approach to parton showers by:

1. Choosing a phase space parameterisation in terms of the gluon's transverse momentum, such that,

   $$k_\perp^2 = \frac{s_{ij}s_{jk}}{s_{IK}} \quad \text{and rapidity} \quad y = \frac{1}{2}\ln\left(\frac{s_{ij}}{s_{jk}}\right) \quad (3)$$

   leads to the inclusive decay probability becoming,

   $$d\mathcal{P}\left(q(p_I)\bar{q}(p_K) \to q(p_i)g(p_j)\bar{q}(p_k)\right) \simeq= \frac{C\alpha_s}{\pi}d\kappa dy \quad \text{with} \quad \kappa = \ln\left(\frac{k_\perp^2}{\Lambda^2}\right) \quad (4)$$

   where Λ is an arbitrary mass scale.

   Within this phase space parameterisation, allowed dipole decays are constrained in a triangular region of phase space with height $L = \ln(s_{IK}/\Lambda^2)$, as shown in Fig. (a).

   The emission of a gluon can be interpreted as "folding out" sampler triangles, as the allowed rapidity span for subsequent dipole decays is increased.

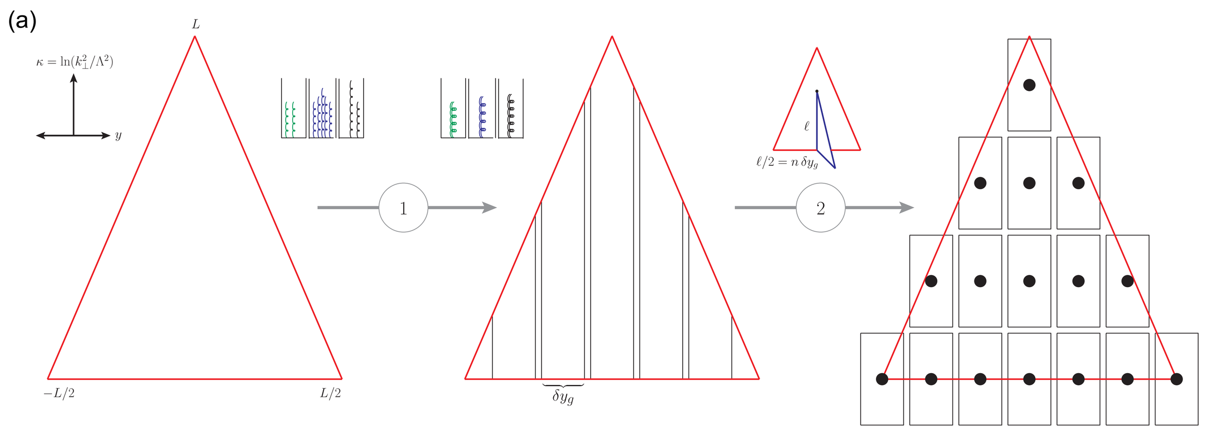
2. Neglecting $g \to q\bar{q}$ splittings, and examining the affect of a transverse-momentum-dependent running coupling,

   $$\alpha_s(k_\perp^2) = \frac{12\pi}{33 - 2n_f}\frac{1}{\ln(k_\perp^2/\Lambda^2_{\mathrm{QCD}})} \quad (5)$$

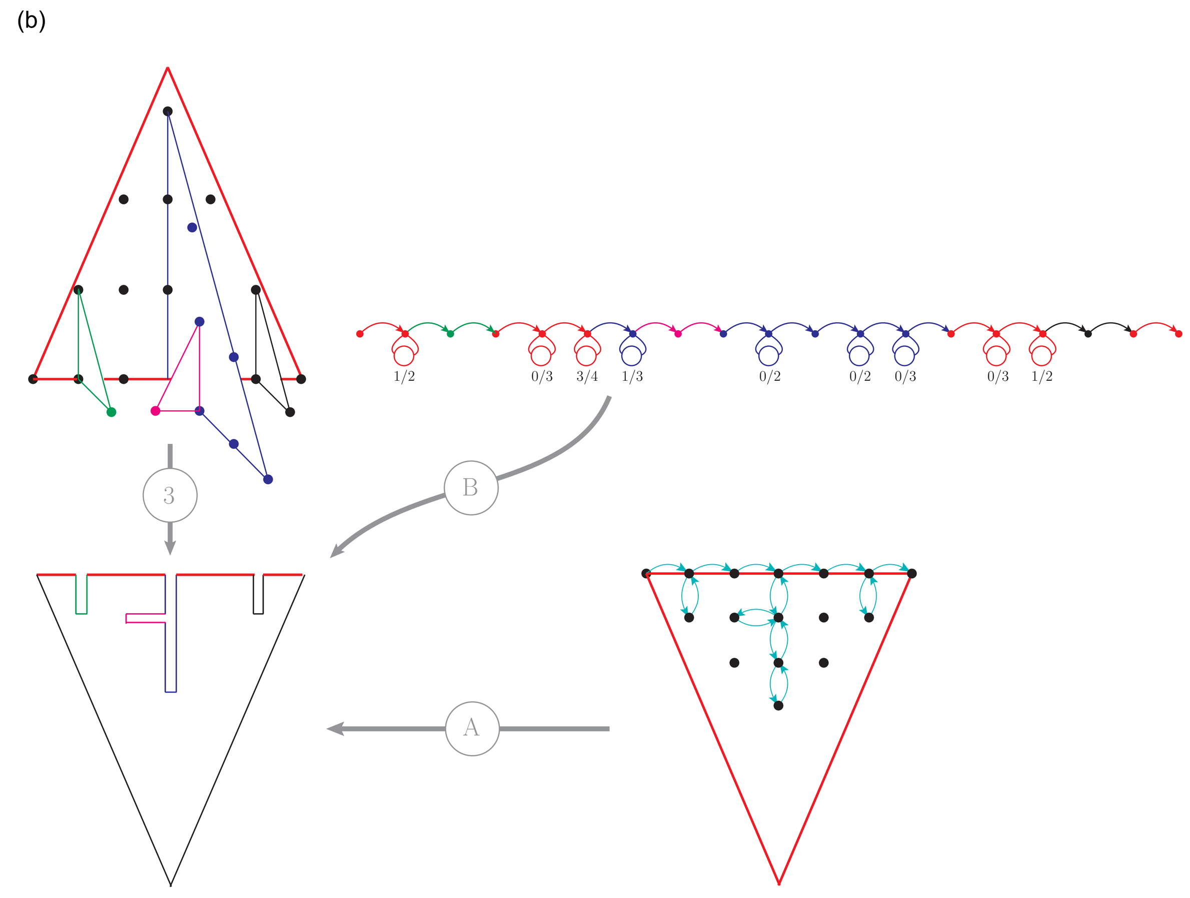
   leads to a simplified expression for the inclusive probability,

   $$d\mathcal{P}\left(q(p_I)\bar{q}(p_K) \to q(p_i)g(p_j)\bar{q}(p_k)\right) \simeq= \frac{d\kappa}{\kappa}\frac{dy}{\delta y_g} \quad \text{with} \quad \delta y_g = \frac{11}{6} \quad (6)$$

   highlighting an important result of [1]: interpreting the running-coupling renormalisation group equation as gain-loss equation means that gluons within a rapidity range δy_g act coherently as one "effective-gluon".



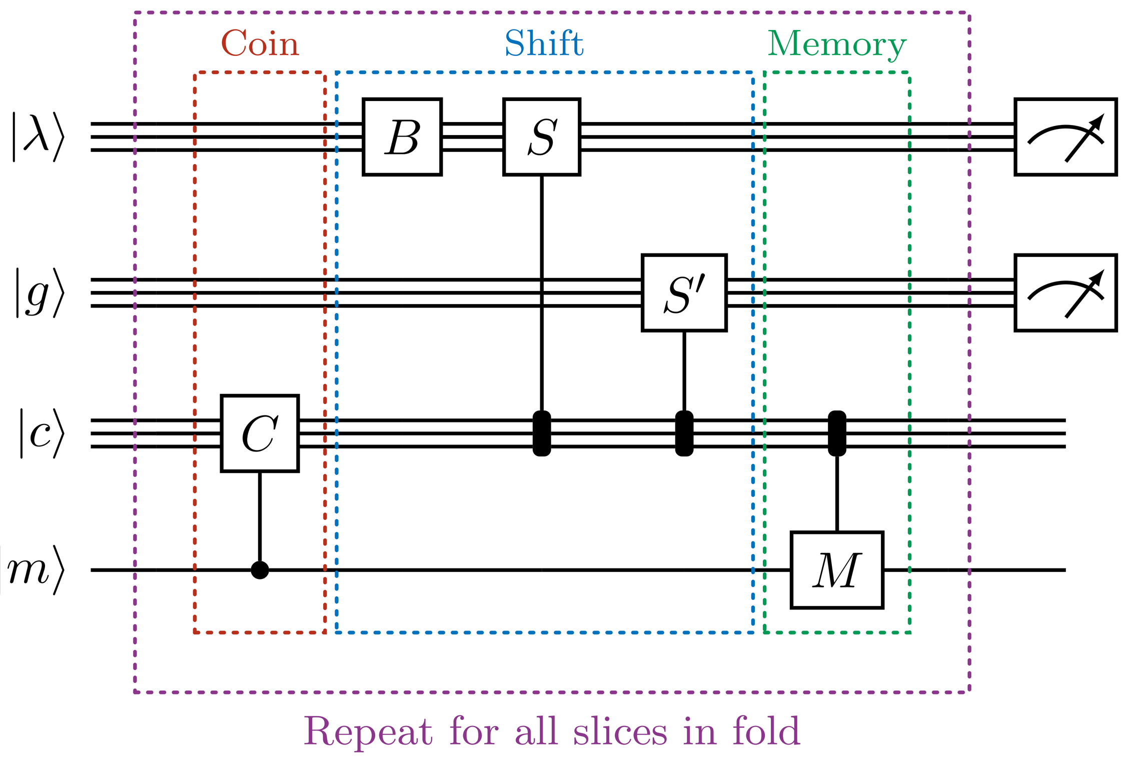
(a) The phase space of effective gluon emission is discrete, since ① gluons within a rapidity region δy_g act coherently due to running-coupling effects. The κ (or equivalently the $k_\perp^2$) dimension is also quantised, since ② additional phase space folds opening due to gluon emission are quantised in units of δy_g.



(b) The Discrete QCD parton shower algorithm can be re-interpreted as a one-dimensional random walk, since ③ the baseline of the folded structure carries all necessary information. The "grove-like" baseline structure can Ⓐ be generated by a heavily constrained two-dimensional random walk. Due to the low fractal dimension of the grove structure, a one-dimensional random-walk algorithm Ⓑ is equally viable. For Ⓑ, the notation n/n_{max} indicates that option n was picked out of n_{max} choices. The two-particle invariants $(\ln(s_{ij}))$ can be read off by following the path from particle i to particle j, and skipping segments whose colour was created and reabsorbed along the way.

The quantisation of the phase space into multiples of δy_g means the baseline of an additional triangle extends to positive y by $l/2$, the height l at which the effect gluon is emitted is quantised into multiples of $2\delta y_g$.

Thus, we may model the parton shower by generating effective gluons at the centre of discrete tiles covering the phase-space triangle. These realisations form the basis of the "Discrete QCD algorithm" of [1].



(c) Schematic of the quantum Discrete QCD parton shower algorithm circuit. The algorithm is a quantum walk with memory, constructed from maximum five operations per step: the coin operation C, the baseline shift B, the λ shift S, the gluon shift S', and the memory operation M.

Grove Generation on a Quantum Device

The grove structures shown in Fig. (b) can be elegantly generated on a quantum device using the Quantum Walk with Memory framework.

The walker moves in the total Hilbert space, constructed from tensor product of the coin space $\mathscr{H}_C$, the gluon space $\mathscr{H}_g$ and the baseline λ-space $\mathscr{H}_\lambda$. A single step of the algorithm is schematically shown in Fig. (c) and is constructed from a maximum of five operations:

1. The coin, C, constructs an equal state on the coin register, such that the outcome of the coin gives an equal probability of selecting a tile in the slice
2. The Baseline operation, B, shifts the walker along the base of the triangle at every step, as shown in Fig. (b).
3. The λ-shift operation, S, controls from the coin and moves the walker correctly in $\mathscr{H}_\lambda$ to represent the "folding out" of a new fold.
4. The gluon-shift operation, S', controls from the coin and updates the shower content accordingly, if a gluon has been emitted.
5. The memory operation, M, records the coin output and allows for conditional coin operations if needed.

This step is then repeated for all slices in the parent fold, and then for all subsequent folds produced. At the end of the algorithm the gluon- and λ-registers are measured. From these, the grove structure can be reconstructed, and the baseline obtained to construct the kinematics.

Generating scattering events from groves

Once the grove structure has been selected, event data can then be synthesised iterative from the baseline of the grove structure, starting from the highest-κ gluons first:

1. For each effective gluon j that has been emitted from a dipole IK, we read off the values of s_{ij}, s_{jk} and s_{IK} from the grove.
2. Generate a uniformly distributed azimuthal decay angle ϕ, and then employ the momentum mapping from [2] to produce post-branching momenta

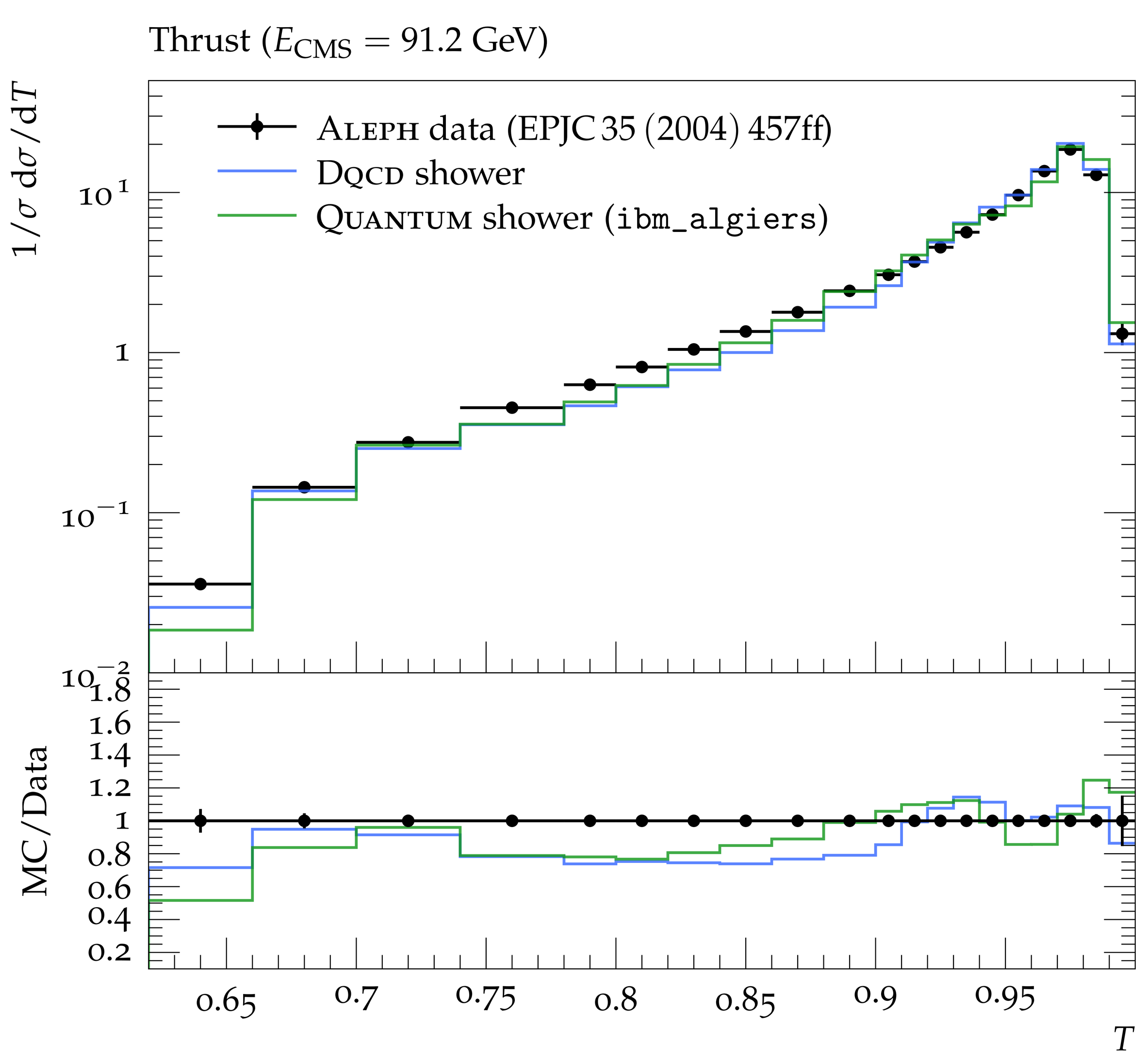
Each point in the tile (y, κ) is equally likely, allowing for some degree of freedom for the second step above. We follow the method discussed in [1] and distribute the y and κ-values of the effective gluon in the highest and second-highest κ tiles uniformly within the tile. This ensures a good model for the highest-κ gluons.

Results and Summary

The algorithm has been run on the ibm_algiers for 20,000 shots using the ibm_cloud. The results are shown in Fig. (c) and show good agreement with the classical Discrete QCD algorithm and the real life LEP data. Despite considerable amounts of noise in the uncorrected output from the quantum computer, the event generation is remarkably robust to quantum errors.

We have synthesised realistic particle collision events by sampling parton shower configurations on a quantum device. As an application, we have compared the simulated data to data recorded at experiments at the Large Electron-Positron (LEP) collider, finding favourable agreement.

This is the first time that the result of a quantum algorithm has been compared to "real-life" particle physics data. The quantum algorithm is constructed using the quantum walk framework and, consequently, is a compact algorithm with a short circuit depth, an important aspect to consider to obtain practical results from NISQ devices.



(d) Sample comparison of the Discrete QCD model and the Quantum Parton Shower to data taken at the LEP collider. The Quantum Parton Shower results are not corrected for errors in the qubit evaluations

This poster summarises the work presented in: G. Gustafson, S. Prestel, M. Spannowsky and S. Williams, *Collider Events on a Quantum Computer*, arXiv:2207.10694 (2022)
[1] B. Andersson, G. Gustafson, and J. Samuelsson, Nucl. Phys. B 463, 217 (1996).
[2] A. Gehrmann-De Ridder, M. Ritzmann, and P. Z. Skands, Phys. Rev. D 85, 014013 (2012), 1108.6172.

Contact details: Simon Williams, Imperial College London
s.williams19@imperial.ac.uk