**Baryon Block Formalism**

Once we have done the Wick contractions, we are left with a number of quark propagators whose indices need to be tied up appropriately.

- The proton-proton scattering by the $\Delta f = 2$ weak operator insertion has 4896 contractions. These reduce to 2208 contractions for degenerate up- and down-quark masses.

- Each contraction comes with a sign. These signs are
  - important for the physics: the signs control the interference between the diagrams.
  - tricky to keep track of by hand: they sensibly depend on operator ordering.

  - absolutely critical to get right: a single error casts the whole result into doubt.

- For partial-wave scattering, we store the full momentum dependence of the propagators. Each contraction corresponds to a Feynman diagram. This can be accomplished
  - Stochastically (allowing us to use arbitrary sinks), at the cost of increased noise.
  - Exactly (which requires a sum over the lattice volume). This is feasible with fixed sinks.

- For nucleon-nucleon matrix elements, this approach is prohibitively expensive, and so we fix our sinks in the same way we fix our sources.

**Introduction**

- A Wick contraction is a way to tie creation and annihilation operators into propagators. Each contraction corresponds to a Feynman diagram.

- The proton-proton scattering by the $\Delta f = 2$ weak operator insertion has 4896 contractions. These reduce to 2208 contractions for degenerate up- and down-quark masses.

- Each contraction comes with a sign. These signs are
  - important for the physics: the signs control the interference between the diagrams.

  - tricky to keep track of by hand: they sensibly depend on operator ordering.

  - absolutely critical to get right: a single error casts the whole result into doubt.

**Automatic Contraction Generation**

- It is impractical to write code for 2208 contractions by hand.

- If calculated by hand, the chance of 2208 correct signs would be low.

- We need software to help write the correct code.

- A good contraction generator would make this tedious task mindless.

  - Parse familiar physics expressions,
  - Be easy to extend,
  - Algorithmically generate all Wick contractions,
  - Generate code for a variety of lattice QCD software.

**Mathematics**

- Mathematics makes it easy to parse and manipulate symbolic expressions, and is interactive and fast enough to allow error correction and fast iteration.

- We have written an automatic contraction generator using Mathematica.

  - Proton two-point function has two Wick contractions
    - Begin with physics expression: $(p|x|p(x))$.
    - Translate into simple mathematica expression:
      
      ```mathematica
      pp = Proton[nuk][xf, x] -> bar[Proton[source][xi, y]] + SpinProjector[source, y, spin, p]]
      ```

  - Symbolically generate contractions:
    - Take generated contractions and generate code
      - Using GPD primitives
      - Symbolically generate contractions:
      ```mathematica
      Generate[GB][contractions] // Notation
      ```

  - for creating tensors for the baryon block formalism
    - Using CUDA
    - The two-baryon contraction problem can be viewed as a calculation of the dense-sparse-dense bilinear product
      ```mathematica
      BB(x + p) T BB(x)
      ```

- Using indirect addressing, we have to compute for all sites $x$ a product such as $v[I1[k]] * T[k] * w[I2[k]]$ and sum over $k$.

- compute $I1, I2$ once and store them along with $v, w$ and $\tau$ in global GPU memory

- read $v[I1[k]]$, $w[I2[k]]$ and $T[k]$ from global memory and compute the product, one $k$ per thread

- perform efficient binary reduction

- for each $x$, create separate CUDA stream to maximize concurrency

- host thread concurrently computes all FFTs and baryon blocks relevant for next step of the calculation

**Acknowledgements**

- Support for this work was provided through Scientific Discovery through Advanced Computing (SciDAC) program funded by U.S. Department of Energy, Office of Advanced Scientific Computing Research and Nuclear Physics.

- Prepared by LLNL under Contract DE-AC52-07NA27344.

- LLNL-POST-657635