Fermilab | U.S. DEPARTMENT OF ENERGY | Office of Science

GENIE development updates + input to “theory API” discussion

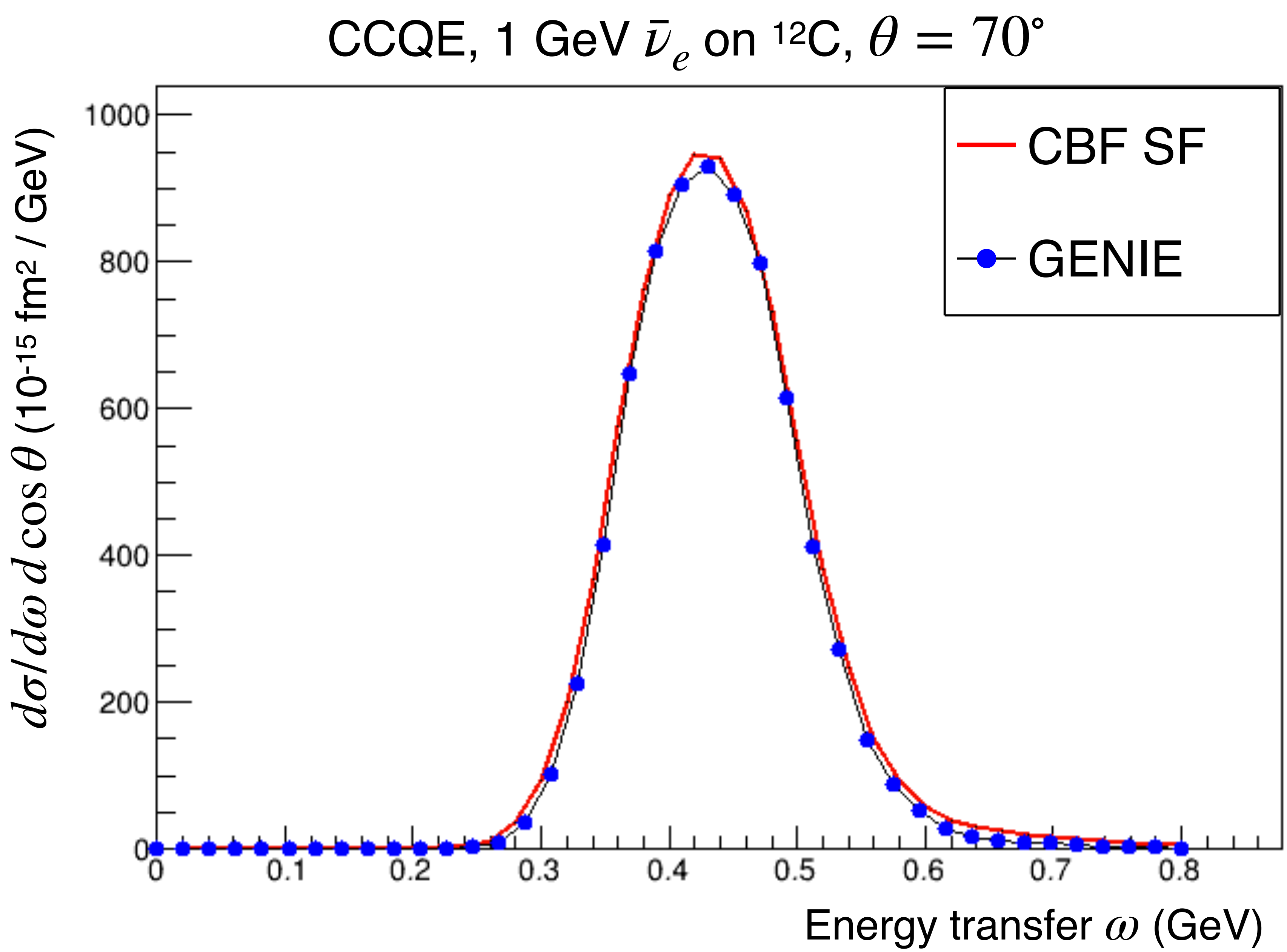
Steven Gardiner

Neutrino Theory/Experiment Joint Group Meeting

11 March 2021

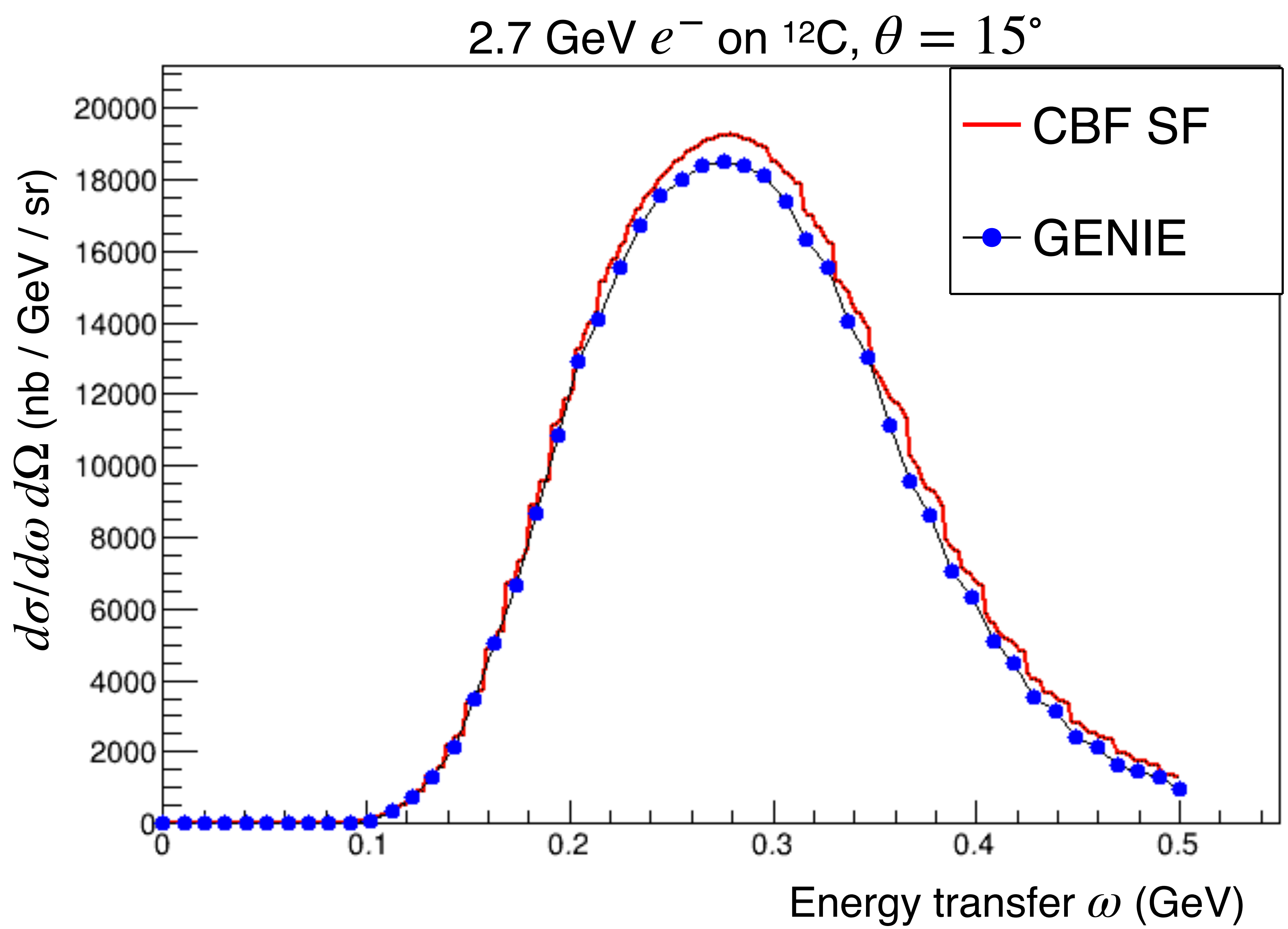
CBF Spectral Function

- Focus lately has been on validation of full nuclear cross sections
  - Low-level checks of tensors, free nucleon results all look good
- Noemi provided differential calculations for CC/NC/EM QE
- Most comparisons looked pretty reasonable



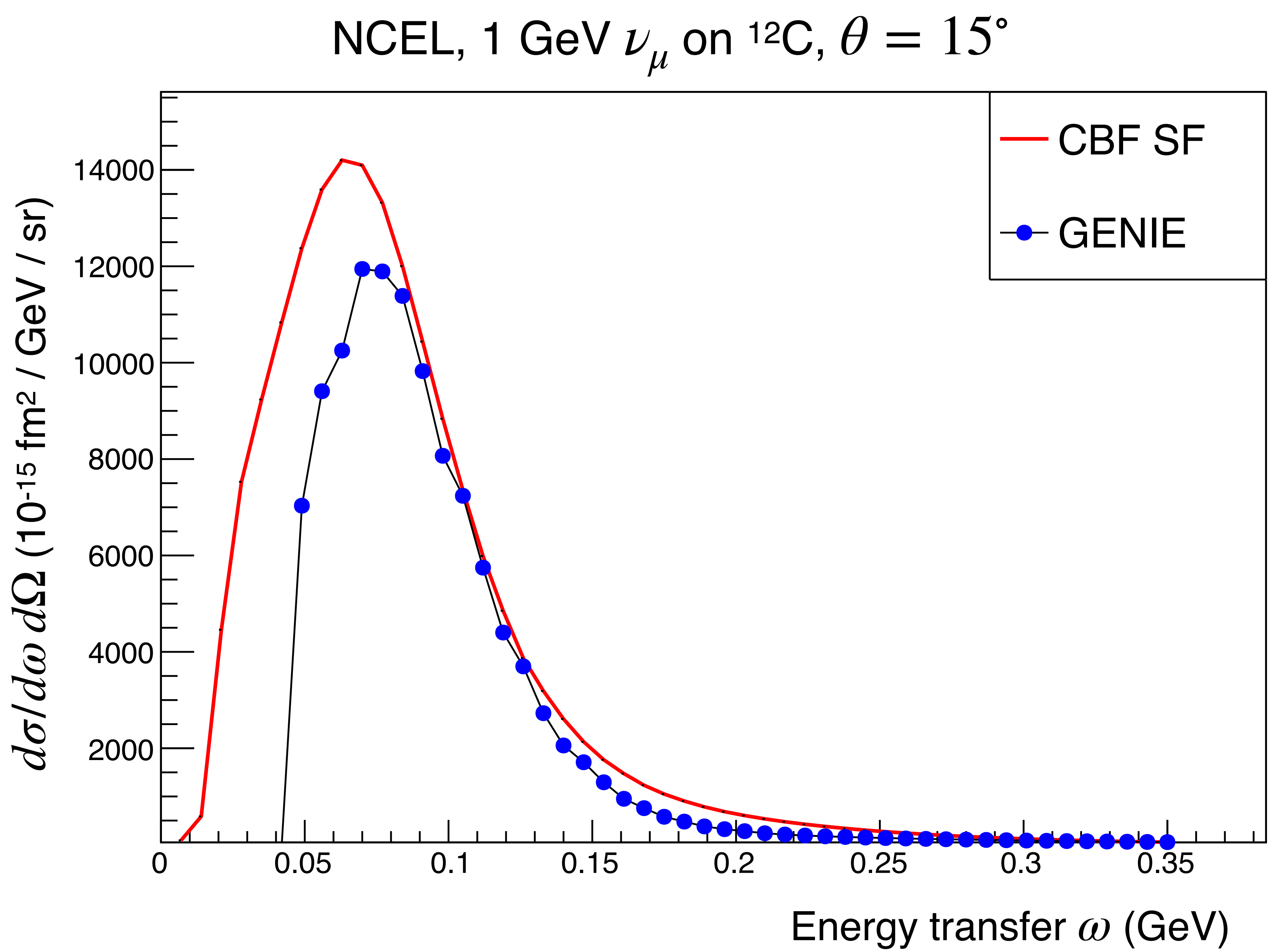
CBF Spectral Function

- Focus lately has been on validation of full nuclear cross sections
  - Low-level checks of tensors, free nucleon results all look good
- Noemi provided differential calculations for CC/NC/EM QE
- Most comparisons looked pretty reasonable



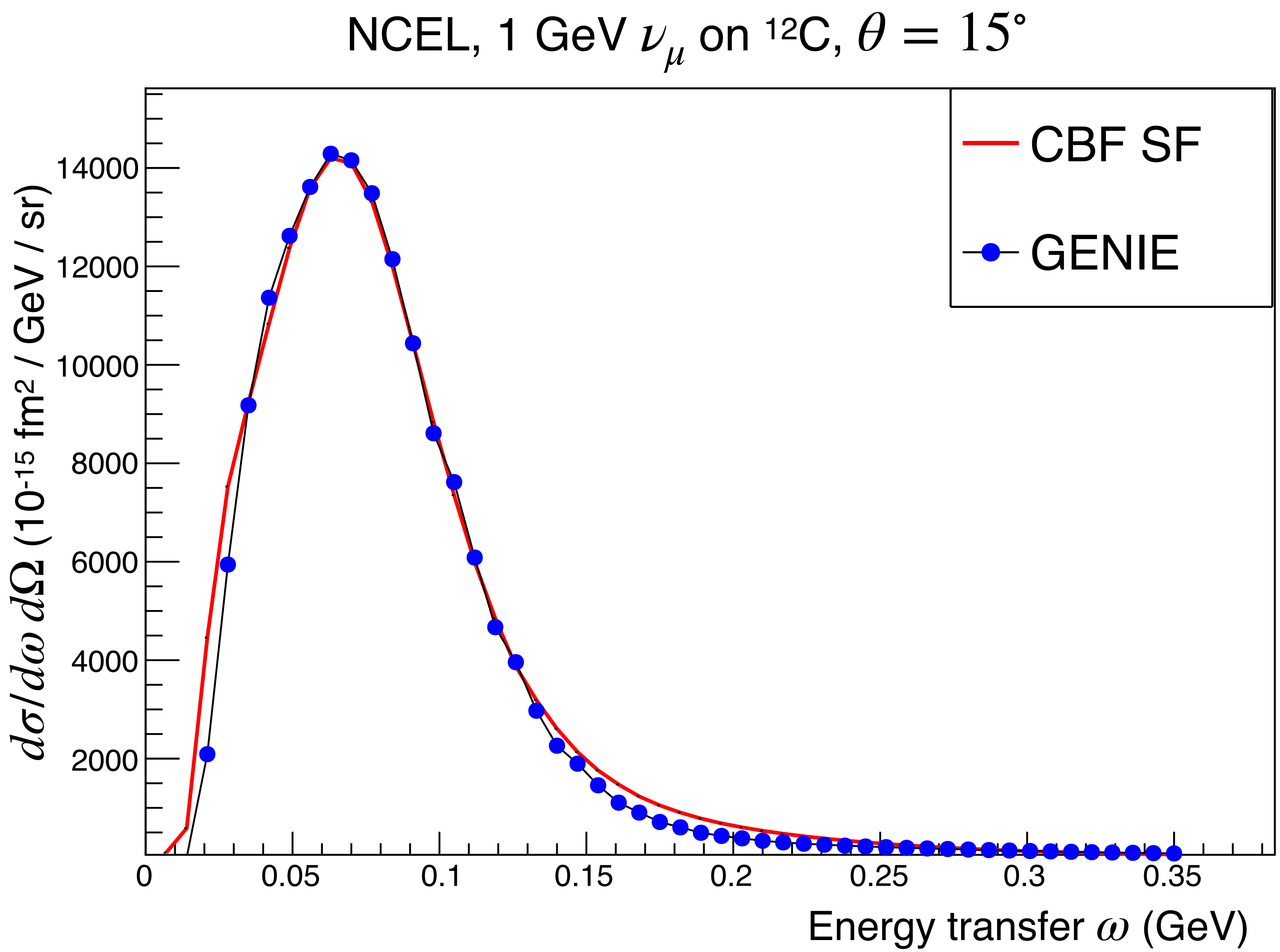
CBF Spectral Function

- Lots of trouble with the NC channel
- "Out of the box," I see obvious **missing strength** in certain regions of phase space
- Various checks were tried, but I didn't catch any major problem



CBF Spectral Function

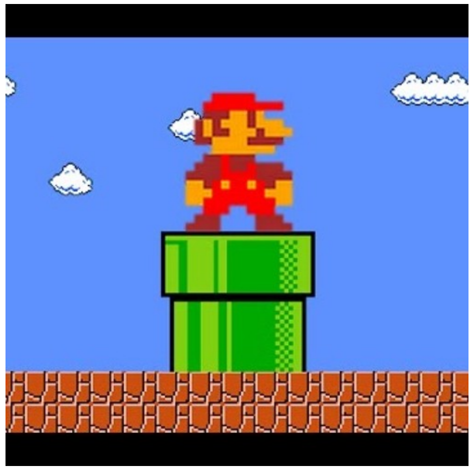
- As an experiment, I tried **turning off Pauli blocking.** Agreement very noticeably improves.
- Is it possible that the NCEL cross section tables were computed without Pauli blocking?
- I've been assuming k_F = 242 MeV/c to match Noemi's EMQE code for ^{12}C
- I will follow up to investigate further



Short-time approximation

- First paper about the inclusive implementation in GENIE was just published
  - J. L. Barrow et al., Phys. Rev. D 103, 052001 (2021)
  - Thanks very much to all who helped us get to this point!
- Next steps
  - GENIE: implement higher-dimensional interpolation of the response densities (toy example based on RooFit in hand).
    - Provides (ω, q)-dependent distributions of pre-FSI nucleon kinematics
  - Theory: extend calculations to higher-mass nuclei (e.g., ^{12}C)
- Active coding on hiatus due to other commitments (Josh graduating soon!). We hope to regain momentum before long.

PIPE: Primary Interaction Physics Engine



- Summary of my recent thoughts on what a “theory API” for neutrino generators might look like, possible “rebranding”

- **Primary Interaction**: limits the scope to what I suspect should be the focus
  - Things like mixing/matching FSIs are harder to universalize
  - Existing effort toward a common event format likely easier

- **Physics Engine**: envisioned as a core component that could in principle be shared by multiple generators
  - Cf. flux and geometry drivers

- Required interfaces
  - Generator: event record, pseudorandom number generator, physics parameters
  - Theory plug-ins: amplitude and/or pieces thereof ($L_{\mu\nu}$, etc.), physics parameters

What's different here from what GENIE already does? (1)

- **Traditional recipe**: point of contact with theory is $d^n\sigma/d\mathbf{x}$
- Many models with many different choices of working phase space $\mathbf{x}$
- Label each unique $\mathbf{x}$, then write custom code to
  1. Calculate $d^n\sigma/d\mathbf{x}$, possibly with transformations to other choices of phase space
  2. Sample elements of $\mathbf{x}$ (e.g., Q^2, W, ϕ_ℓ) from differential cross section
  3. Apply any needed kinematic cuts (e.g., low Q^2 for EM) and build full 4-momenta
  4. Integrate over $d\mathbf{x}$ to compute total cross section splines
  5. Compute likelihood ratios involving $d^n\sigma/d\mathbf{x}$ for reweighting
- Process can be very labor-intensive. For some choices of $\mathbf{x}$, steps 1-5 must be done from scratch

What's different here from what GENIE already does? (2)

- **Alternate strategy**: point of contact with theory is the (spin-summed) squared amplitude (or pieces thereof)

- Use the universal expression for the differential cross-section for the reaction a + b → n particles (in the style of LHC generators?)

Differential cross section

$$d\sigma = F^{-1}\,\overline{|M|}^2\,(2\pi)^{4-3n}\,dV_n$$

Møller flux factor

$$F = 4\sqrt{(p_a \cdot p_b)^2 - m_a^2 m_b^2}$$

n-body phase space element

$$dV_n = \delta^{(4)}\left(p_a + p_b - \sum_{f=1}^{n} p_f\right) \prod_{f=1}^{n} \delta(p_f^2 - m_f^2)\,\theta(p_f^0)\,d^4p_f$$

How does this help?

- Clever methods exist for sampling V_n values, i.e., **complete sets of final 4-momenta** which respect the cuts imposed by the delta functions
  - Conservation of 4-momentum, positive on-shell energies
  - Required inputs are just $\sqrt{s}$ and the final particle masses
- ROOT's TGenPhaseSpace implements a simple approach
  - Tools like RAMBO provide more sophisticated techniques
- Leveraging this could simplify the work needed to add a new primary interaction model

Adding a new primary interaction model becomes simpler

1. Calculate $d^n\sigma/d\mathbf{x}$, possibly with transformations to other choices of phase space
   - Pass 4-momenta to theory code, get $\overline{|M|}^2$ back (the rest is trivial)
2. Sample elements of $\mathbf{x}$ (e.g., Q^2, W, ϕ_ℓ) from differential cross section
   - TGenPhaseSpace (or equivalent) handles this in all cases
   - Fermi motion, W dependence introduce a few minor complications
3. Apply any needed kinematic cuts (e.g., low Q^2 for EM) and build full 4-momenta
   - Trivial, apply any needed cuts to 4-momenta
4. Integrate over $d\mathbf{x}$ to compute total cross section splines
   - Use MC integration via TGenPhaseSpace. “Naive” sampling works, could apply something more sophisticated (VEGAS, etc.) for better performance
   - Can also make a spline of the maximum of $d^n\sigma/dV_n$ for easy rejection sampling during event generation
5. Compute likelihood ratios involving $d^n\sigma/d\mathbf{x}$ for reweighting
   - Reduces to ratio of matrix elements $\overline{|M|}^2$

Fermi motion

- In principle, this approach already works for Fermi motion
  - Include spectral function $P(\mathbf{p}, E)$ in the amplitude
  - Determines 4-momentum of final-state spectator nucleus
- In practice, this can be very inefficient for sampling
- Alternative recipe
  - Throw $\mathbf{p}$ and E from spectral function, build 4-momentum of spectator
  - Use $\sqrt{s}$ for neutrino-nucleon system, reduce n by one
- Seems similar to handling of parton distribution functions in LHC generators

W dependence

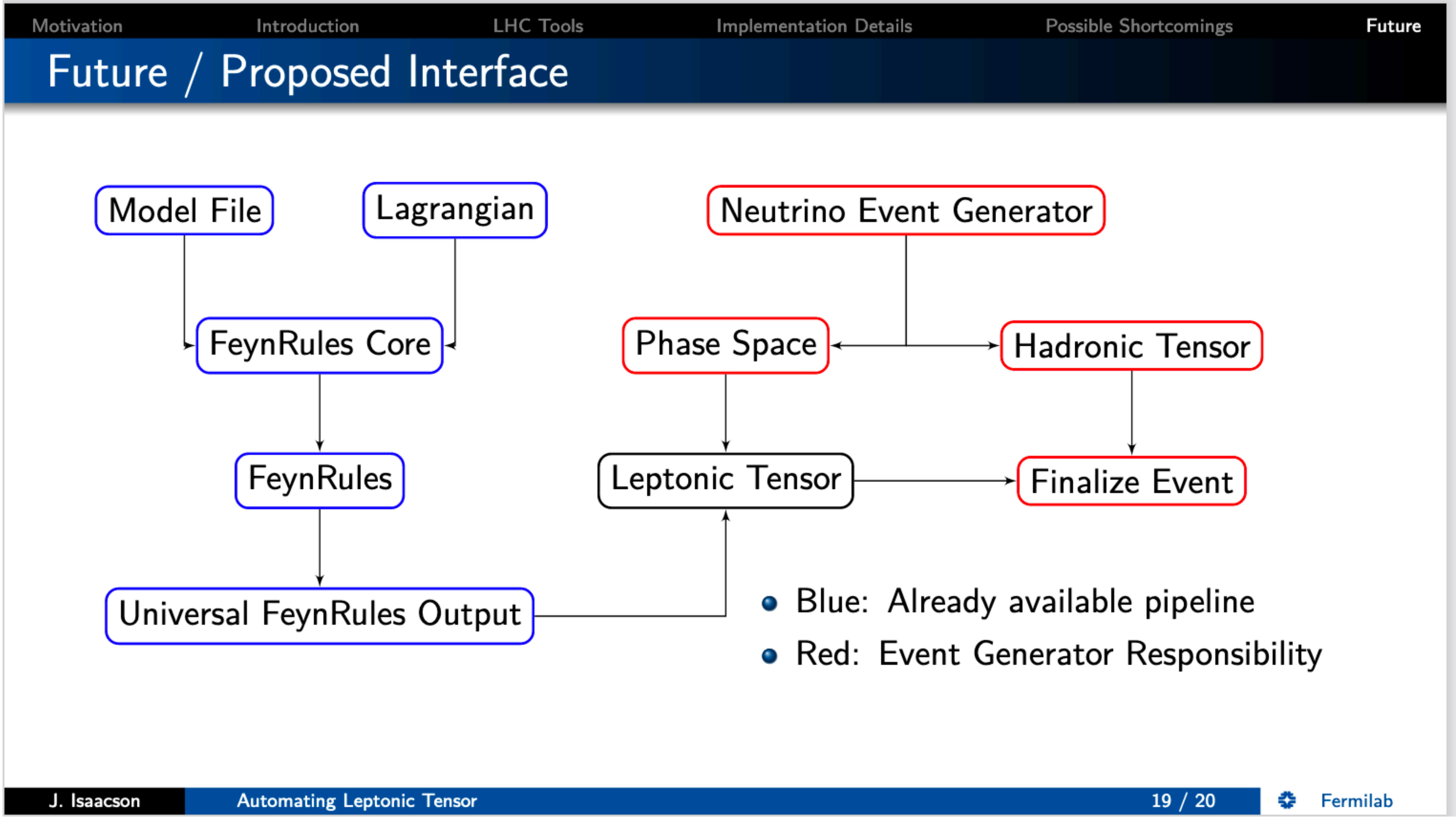
- Phase space sampling requires knowledge of final particle masses
- Brute-force solution: throw W uniformly between known limits, then choose a phase-space point V_n
  - Likely inefficient, but seems workable if W limits are known
- For GENIE RES models, likely better to "pull out" a Breit-Wigner distribution from $\overline{|M|}^2$
  - Throw W according to this distribution, then throw V_n according to phase space
  - Accept/reject V_n using $d^n\sigma/dV_n$ with B-W removed

Configuration of model parameters

- To ensure consistency of physics parameters, we need a means of passing them between the generator and external theory code
- Possible option: **JSON objects**
  - Hierarchical data structure based on nested key-value pairs
  - Can represent arbitrary groupings of numbers, strings, arrays, boolean values
  - Good inter-language support (APIs for C++, Fortran, Python, etc.)
- Each theory code could specify the names and types (e.g., integer) of the parameters needed from the generator
- The generator would then supply this information via a JSON interface

Discussion

- I believe that the vision that I've outlined here is compatible with Josh's
- Do you think this path forward is workable? What problems can you see with it?



Backup

Snowmass workshop on neutrino generators (postponed)

- Together with the NF06 conveners, I was working on an agenda late last year for a generator-focused workshop for Snowmass (involving NF06/CompF2/TF11)

- I'm still interested in making this happen, but it will be important to respect the "Snowmass pause"

- Scope will be similar to the one from January 2020 (see **https://arxiv.org/abs/2008.06566** for a summary)
  - Share new developments in this area
  - Provide concrete input (in the form of a white paper) to the Snowmass process