Modeling Hadronization using Machine Learning

QCD@LHC 2022

Tony Menzo

PhD candidate, University of Cincinnati

In collaboration with:

Phil Ilten, Stephen Mrenna, Manuel Szewc, Michael Wilkinson, Ahmed Youssef, and Jure Zupan

Based upon work done in 2203.04983
Goals and outlook

The main motivation is to create a better simulation of collider events.

But also to promote a paradigm shift in the modeling of non-perturbative physics.

**Present:** In [2203.04983](https://arxiv.org/abs/2203.04983) we showed that machine learning techniques can be used to implement a model of hadronization using (artificial) training data.

**Near Future:** Implement a machine learning-improved (i.e. data-improved) model of hadronization.

**Far future:** Take what we’ve learned and develop BETTER theoretical models.
Event Generators

1. Hard process

2. Parton Showers

3. Hadronization

4. Unstable particle decay

Early 80s brought many non-perturbative models: Cluster, percolation, ...

Lund String Model
(currently implemented in Pythia)
Stringy Hadronization

The momentum fraction $z$ of each fragmenting hadron is sampled according to the 

**Lund fragmentation function**

$$f(z) \propto \frac{(1 - z)^a}{z} \exp \left( \frac{-bm^2}{z} \right)$$

$$z = \frac{p_z + E_h}{2E}$$
Troubles in fit paradise

\[ f(z) \propto \frac{(1 - z)^a}{z} \exp \left( -\frac{bm_{\perp}^2}{z} \right) \]

The \( a \) parameter

\[ a = 0.9 \quad a = 0.1 \]

The \( b \) parameter

\[ b = 0.5 \quad b = 2.0 \]

Cannot be improved by retuning

Monash Tune \((a,b,\sigma_{\text{perp}})\)
Some ~new disagreements for high multiplicity events...

Similar properties to heavy ion collisions:

- “The ridge” i.e. enhanced particle production around the azimuthal angle of a trigger jet (CMS)
- Strangeness production increases as a function of event multiplicity (ALICE)
How to improve the generator: two* approaches

• Improve model
  - MPIs, rope hadronization, transverse mass suppression, flavor asymmetries, hadronic rescattering, multiscale models (string → hydrodynamical), flavor selector, etc.
  - Utilize techniques from gauge-gravity duality
    Hard to come up with mathematically precise model without established calculational techniques

• Data-driven generator
  - Sample directly from global distributions
    Non-universal and extremely difficult to convert into representative particle flow data

* or a combination of both (our approach)
Hybrid approach

Hadronization models already do really well!

Model + Experimental data

↓

Complete (or at the very least better) phenomenological model of hadronization

For example, modify the fragmentation function $f(z)$...

\[ b = 1 \text{ GeV}^{-2}, \quad m_\perp = 1 \text{ GeV} \]

\[ a = 0.9, \quad a = 0.1 \]

\[ b = 0.5, \quad b = 2.0 \]

\[ a = 0.5, \quad m_\perp = 1 \text{ GeV} \]
Why machine learning?

To make any headway we need a tool which will allow us to efficiently sample probability distributions whose analytic form is unknown.

Generative machine learning algorithms are the perfect tool!
Consider Pythia output as ‘experimental data’ and try to reproduce hadronization observables by training on single emission kinematics (~learn the fragmentation function \( f(z) \)).

Start from simplest hadronizing system:

1. \( q\bar{q} \rightarrow \pi's \)
2. Assume no correlations between emissions
3. \( E_{\text{cut}} \sim 5 \text{ GeV} \) (To avoid termination effects)

Train on \( p_z \) and \( p_T \) distributions of 1st emitted \( \pi \)
Training Results (cSWAE)

![Graphs showing training results for different models.](image-url)
Training Results
(cSWAE with labels and boundaries)

*Preliminary*
Hadronization (kinematics + flavor selector)

The trained model distributions now need to be integrated into a chain of fragmentations.

Stopping condition: $E_i < E_{\text{cut}}$
Global observable (Hadron multiplicity cSWAE)

![Graph showing the distribution of length of fragmentation chain for different models, with labels for PYTHIA (Avg. = 9.06 ± 0.08) and MLHAD (Avg. = 9.15 ± 0.06).]
Global scaling (Hadron multiplicity vs string energy cSWAE)
Error estimation \textit{(BNN)}

Incorportating (theoretical/experimental) errors from training dataset errors into the hadronization simulation

\*Preliminary
Conclusion

Model + machine learning methods CAN be used to implement hadronization.

What’s next:

• ML-improved (data-improved) model of hadronization
• ML flavor selector
• Error estimation
• Much more

Check out our repo!
https://gitlab.com/uchep/mlhad

Check out our paper!
arXiV: 2203.04983
Back-up
Architectures

Conditional sliced-Wasserstein Autoencoder (cSWAE)

Conditional normalizing flow (cNF)
Training Results
(cNF with labels)

*Preliminary

PDF
Target, \( m = 0.2 \) GeV
\( E = 100.0 \) GeV
\( E = 400.0 \) GeV
\( E = 700.0 \) GeV
\( E = 1000.0 \) GeV

PDF
Target, \( m = 0.2 \) GeV
\( E = 100.0 \) GeV
\( E = 400.0 \) GeV
\( E = 700.0 \) GeV
\( E = 1000.0 \) GeV

Ratio
\( p_z \)
\( p_T \)