The TMDGen Monte Carlo Generator

S. Gliske

Argonne National Lab
(Formerly of the U. of Michigan)
HERMES and STAR Collaborations

TMD Monte Carlo Workshop
Frascati, Italy
November 7th and 8th, 2011

Guiding Principles

- Depend on as few other libraries as possible
- Allow for future expansion and modification (C++ polymorphism)
- Allow for a variety of basic physics
  - I.e. allow for *pp*, *ep* collider, *ep* fixed target, all polarizations, etc.
  - Also allow for SIDIS single hadron, SIDIS dihadron, exclusive production, etc.
- Allow for different end users
  - Output must have interface to HERMES software framework
  - Allow expansion for other experimental software frameworks.
  - Must be able usable for theorists with no experimental software framework
- Allow for any model for the cross section
  - For factorized cross section, allow any model for distribution and fragmentation functions
  - Also can allow non-factorized models, i.e. models for the structure functions instead.
- Allow full modeling of intrinsic k_T and p_T distributions:
  - Every event has a given value of k_T and p_T

General Structure

- Basic algorithm is
  1. Throw cross section variables
  2. Evaluate the cross section
  3. Apply acceptance/rejection
     - Can also just use the computed weight of the event.
  4. Convert cross section variables to lab variables
  5. Save the output (in some fashion)
- Note: TMDGen began as a port of GMC_Trans from FORTRAN to C++
- Although TMDGen has outgrown its beginings, many thanks to Gunar Schnell, Dehlia Hasch, and others who worked on GMC_Trans

What is currently tested in TMDGen?

- All input options given via a human readable text file
  - Reminiscent of a KUMAC, but it is not
- Basic physics:
  - SIDIS single hadron, unpolarized and transverse target moments
  - SIDIS dihadrons, unpolarized and transverse target moments
- Output
  - HERMES DAD/ADAMO tables
  - Root TFile
- Variety of models (listed later)

How would one add...

- Other SIDIS polarization states
  - Just need to finish testing the cross section terms
- A new model for distribution or fragmentation functions
  - Create a new child of the right abstract base class
  - Construct the object in the right place
- A new term in the cross section
  - Create a new child of the right abstract base class
  - Construct the object in the right place
- A new cross section (such as exclusive production or *pp*)
  - Define the cross section term classes and follow the pattern already existent
- Collider *ep* instead of fixed target *ep*
  - Add a new conversion function from cross section to lab variables
- A target other than a proton
  - Ensure the distribution functions support the desired target
  - Add possibly missing distribution functions

Intrinsic Transverse Momentum

- Every event has a value of intrinsic k_T and p_T.
- Not all models necessary use the values given
- The observable $P_{h\perp}$ and the quark distribution intrinsic transverse momentum k_T are chosen as the independent variables.
- The fragmentation p_T is then set by momentum conservation (satisfying the usual delta function).
- TMDGen allows for addition TMD functions to multiply models of distribution and fragmentation functions without explicit TMD.
- One can also create any arbitrary TMD function for distribution and fragmentation functions.
- TMDGen is a full test ground for modeling intrinsic TMD of models!

Implemented Distribution Functions

Distribution Functions	**Model**
f_1	CTEQ
f_1	LHAPDF
f_1	BCR08
f_1	GRV98
g_1	GRSV2000
$f_{1T}, h_{1T}^{\perp}, h_1$	Torino Group
$f_1, g_1, g_{1L}, g_{1T}, f_{1T}, h_1, h_1^{\perp}, h_{1T}^{\perp}$	Pavia Spectator Model

Implemented Fragmentation Functions

Frag. Functions	Final State	Model Identifier
D_1	pseudo-scalar	fDSS
D_1	pseudo-scalar	Kretzer
$D_1, H_1^{\perp}$	dihadron	TMD Spectator Model
$D_1, H_1^{\perp}$	dihadron	Set given partial wave proportional to any other partial wave

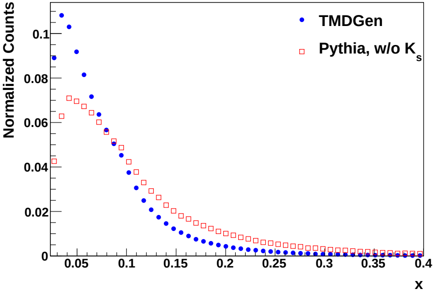
- Dihadron fragmentation functions use the Gliske-partial wave expansion
- The above dihadron D_1 and $H_1^{\perp}$ represent all the partial waves, including the interference fragmentation function $H_{1,sp}^{\sphericalangle}$
- Not all partial waves are necessarily non-zero.

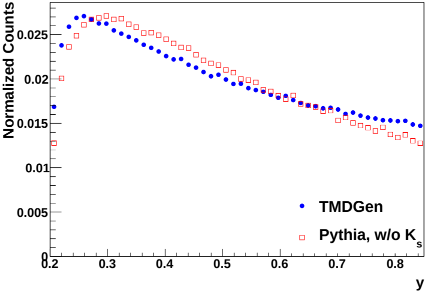
TMDGen Example: SIDIS $\pi^+\pi^0$ production

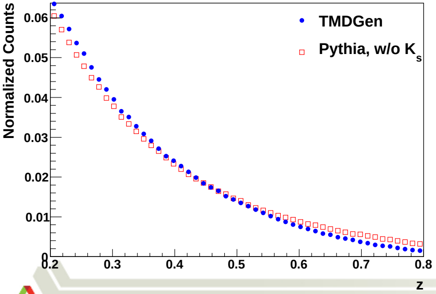
- Dihadron $\pi^+\pi^0$ production includes ρ^+ production
- Final state is $\pi^+\gamma\gamma$
- Distribution function model: GRV98.
- Fragmentation model: my TMD generalization of Bacchetta, *et. al* PRD 74:11 (2006)
- Full details of these plots are in my thesis

  http://www-hermes.desy.de/notes/pub/11-LIB/sgliske.11-003.thesis.pdf
- Following slides will compare data from PYTHIA (tuned to HERMES kinematics) with output of TMDGen.

$\pi^+\pi^0$ Kinematic Distributions, p.1

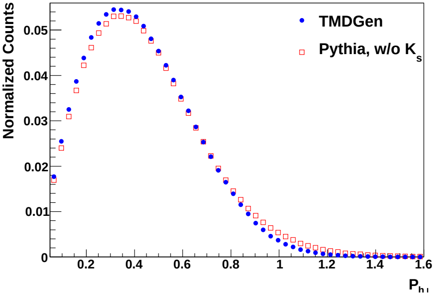






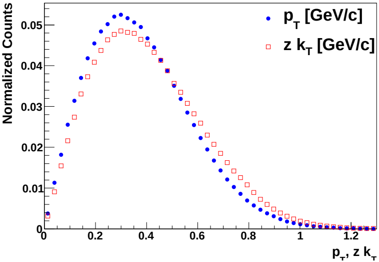
- Close agreement for x, y, z distributions.
- Main discrepancy in x—may be due to imbalance in the flavor contributions, or Q^2 effects.
- Similar results for other $\pi\pi$ and KK dihadrons.

$\pi^+\pi^0$ Kinematic Distributions, p.2



- Fairly good agreement in both $P_{h\perp}$ and M_h distributions.
- Note: some discrepancies in full 5*D* kinematic, but PYTHIA also doesn't match data in full 5*D*

$\pi^+\pi^0$ Kinematic Distributions, Intrinsic Transverse Momentum



- Partonic transverse momentum denoted p_T
- The fragmenting quark's transverse momentum is zk_T
- Model requires $p_T \approx zk_T$ in order to get narrow $P_{h\perp}$ peak
- Model does not require any flavor dependence to k^2, k_T^2 cut-offs
- However, model poorly constrains RMS values $\langle p_T^2 \rangle$, $\langle k_T^2 \rangle$
- No other generator can show p_T, k_T distributions

Conclusion/Outlook

- TMDGen is a flexible framework for all TMD cross sections
- So far, TMDGen tested and used for HERMES Dihadron results
- Many more possible uses for TMDGen
- Main constraint is time/manpower to continue development of TMDGen
- I will gladly share the source code
  - I just need to prepare some installation and use instructions