Parton Shower based on TMD parton distributions

Melanie Viola Schmitz [1], Francesco Hautmann [2], Hannes Jung [1], Sara Taheri Monfared [1]

[1]Deutsches Elektronen-Synchrotron (DESY)
[2]University of Antwerp

DIS 2019, Turin

Outline

1. The Parton Branching method
2. The TMD parton shower and its applications
3. Obtaining TMDs from two different parton showers
4. Detailed study of the TMD parton shower
5. Conclusion

The Parton Branching method

- The parton branching method is a **way to solve** the DGLAP equations

The Parton Branching method

- The parton branching method is a **way to solve** the DGLAP equations
- For this method the **Sudakov form factor** is defined:

$$\Delta_s(t) = \exp\left(-\int_x^{z_{max}} \mathrm{d}z \int_{t_0}^{t} \frac{\alpha_s}{2\pi} \frac{\mathrm{d}t'}{t'} \tilde{P}(z)\right)$$

The Parton Branching method

- The parton branching method is a **way to solve** the DGLAP equations
- For this method the **Sudakov form factor** is defined:

$$\Delta_s(t) = \exp\left(-\int_x^{z_{max}} \mathrm{d}z \int_{t_0}^{t} \frac{\alpha_s}{2\pi}\frac{\mathrm{d}t'}{t'}\tilde{P}(z)\right)$$

- with the Sudakov form factor, a general form of the evolution equation in a form using the Sudakov form factor can be derived $\left(t = \mu^2\right)$:

$$t\frac{\partial}{\partial t}\frac{f(x,t)}{\Delta_s} = \int \frac{\mathrm{d}z}{z}\frac{1}{\Delta_s}\frac{\alpha_s}{2\pi}P(z)f\left(\frac{x}{z},t\right)$$

The Parton Branching method

- The parton branching method is a **way to solve** the DGLAP equations
- For this method the **Sudakov form factor** is defined:

$$\Delta_s(t) = \exp\left(-\int_x^{z_{max}} \mathrm{d}z \int_{t_0}^{t} \frac{\alpha_s}{2\pi} \frac{\mathrm{d}t'}{t'} \tilde{P}(z)\right)$$

- with the Sudakov form factor, a general form of the evolution equation in a form using the Sudakov form factor can be derived $\left(t = \mu^2\right)$:

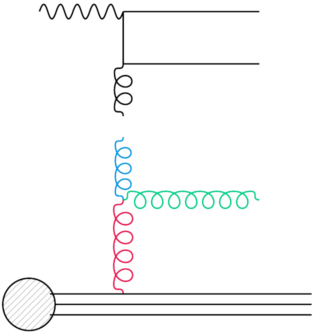
$$t \frac{\partial}{\partial t} \frac{f(x,t)}{\Delta_s} = \int \frac{\mathrm{d}z}{z} \frac{1}{\Delta_s} \frac{\alpha_s}{2\pi} P(z) f\left(\frac{x}{z}, t\right)$$

- Integration leads to:

$$f(x,t) = f(x,t_0)\Delta(t) + \int \frac{\mathrm{d}t'}{t'} \frac{\Delta(t)}{\Delta(t')} \frac{\alpha_s(t')}{2\pi} \int \frac{\mathrm{d}z}{z} P(z) f\left(\frac{x}{z}, t'\right)$$

- **solve it by iteration**

The Parton Branching method



$$f_0\left(x,t\right) = f\left(x,t_0\right)\Delta\left(t\right)$$

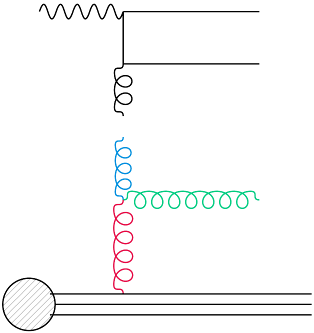
$$f_1\left(x,t\right) = f\left(x,t_0\right)\Delta\left(t\right) + \frac{\alpha_s}{2\pi}\int_{t_0}^{t}\frac{\mathrm{d}t'}{t'}\frac{\Delta\left(t\right)}{\Delta\left(t'\right)} \cdot \int_x^1 \frac{\mathrm{d}z}{z}\tilde{P}\left(z\right) f\left(x/z,t_0\right)\Delta\left(t'\right)$$

$$= f\left(x,t_0\right)\Delta\left(t\right) + \log\tfrac{t}{t_0} A \otimes \Delta\left(t\right) f\left(x/z,t_0\right)$$

$$f\left(x,t\right) = \lim_{n\to\infty} f_n\left(x,t\right) = \lim_{n\to\infty}\sum_n \frac{1}{n!}\log^n\left(\frac{t}{t_0}\right) A^n \otimes \Delta\left(t\right) f\left(\frac{x}{z},t_0\right)$$

with $A = \int \frac{\mathrm{d}z}{z}\tilde{P}(z)$ $\rightarrow$ resummed to all orders in $\alpha_s \log t$

The Parton Branching method



$$f_0(x,t) = f(x,t_0)\,\Delta(t)$$

$$f_1(x,t) = f(x,t_0)\,\Delta(t) + \frac{\alpha_s}{2\pi}\int_{t_0}^{t}\frac{\mathrm{d}t'}{t'}\frac{\Delta(t)}{\Delta(t')} \cdot \int_x^1 \frac{\mathrm{d}z}{z}\tilde{P}(z)\,f(x/z,t_0)\,\Delta(t')$$

$$= f(x,t_0)\,\Delta(t) + \log\tfrac{t}{t_0}\,A\otimes\Delta(t)\,f(x/z,t_0)$$

$$f(x,t) = \lim_{n\to\infty} f_n(x,t) = \lim_{n\to\infty}\sum_n \frac{1}{n!}\log^n\left(\frac{t}{t_0}\right)A^n\otimes\Delta(t)\,f\left(\frac{x}{z},t_0\right)$$

with $A = \int \frac{\mathrm{d}z}{z}\tilde{P}(z)$ $\rightarrow$ resummed to all orders in $\alpha_s \log t$

$\rightarrow$ kinematics of **every single splitting process** can be treated exactly

The TMD parton shower and its applications

What is the gain of the parton branching method?

The TMD parton shower and its applications

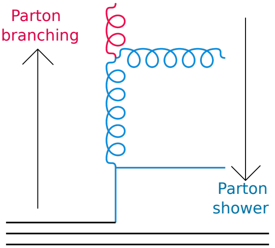
What is the gain of the parton branching method?

PB evolution

starts at hadron scale μ_0^2 and evolves from small to large μ^2

Parton shower

backward evolution from hard scale μ^2 to hadron scale μ_0^2



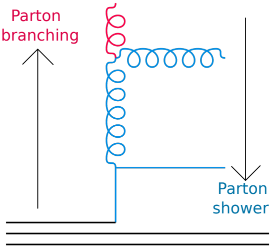
What is the gain of the parton branching method?

PB evolution

starts at hadron scale μ_0^2 and evolves from small to large μ^2

Parton shower

backward evolution from hard scale μ^2 to hadron scale μ_0^2



$\rightarrow$ PB method **allows to determine TMDs**

The TMD parton shower and its applications

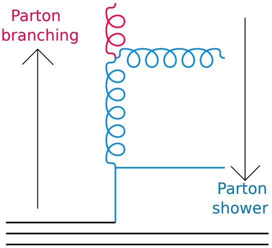
What is the gain of the parton branching method?

PB evolution

starts at hadron scale μ_0^2 and evolves from small to large μ^2

Parton shower

backward evolution from hard scale μ^2 to hadron scale μ_0^2



→ PB method **allows to determine TMDs**

→ PB method **allows to construct parton shower following exactly TMD**

The TMD parton shower and its applications

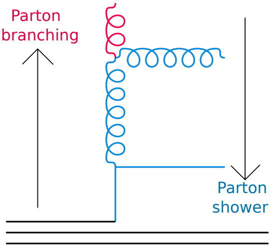
What is the gain of the parton branching method?

PB evolution

starts at hadron scale μ_0^2 and evolves from small to large μ^2

Parton shower

backward evolution from hard scale μ^2 to hadron scale μ_0^2



→ PB method **allows to determine TMDs**

→ PB method **allows to construct parton shower following exactly TMD**

TMD parton shower (implemented in CASCADE 3):

The TMD parton shower and its applications

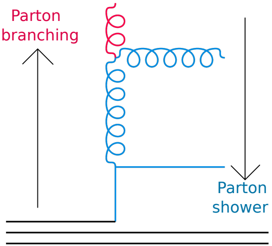
What is the gain of the parton branching method?

PB evolution

starts at hadron scale μ_0^2 and evolves from small to large μ^2

Parton shower

backward evolution from hard scale μ^2 to hadron scale μ_0^2



→ PB method **allows to determine TMDs**

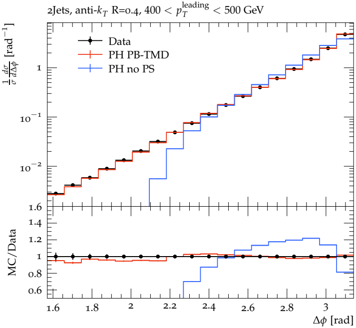
→ PB method **allows to construct parton shower following exactly TMD**

TMD parton shower (implemented in CASCADE 3):

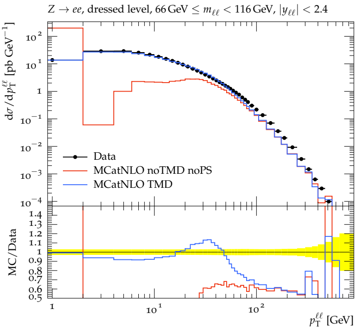
requires a TMD, depending on the transverse momentum of the propagator

How can this TMD parton shower be applied?

How can this TMD parton shower be applied?



$\Delta\Phi$ distribution for NLO dijets

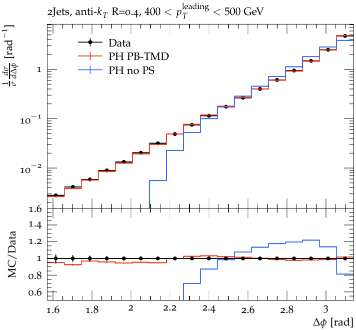


p_T distribution for NLO DY process

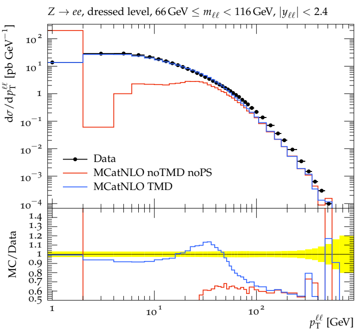
→ POWHEG+TMD+PS gives very good description & MC@NLO+TMD fits reasonable

The TMD parton shower and its applications

How can this TMD parton shower be applied?



$\Delta\Phi$ distribution for NLO dijets



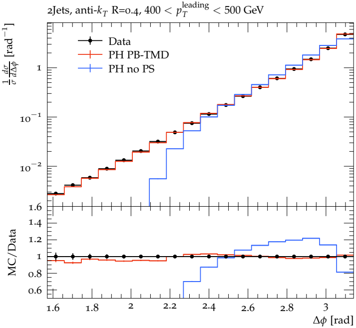
p_T distribution for NLO DY process

$\rightarrow$ POWHEG+TMD+PS gives very good description & MC@NLO+TMD fits reasonable

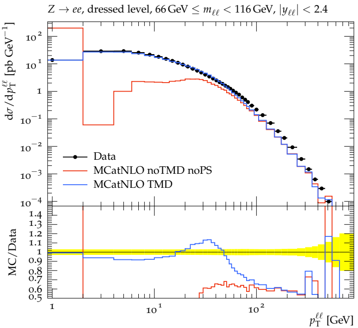
But how does it look like in detail?

The TMD parton shower and its applications

How can this TMD parton shower be applied?



$\Delta\Phi$ distribution for NLO dijets



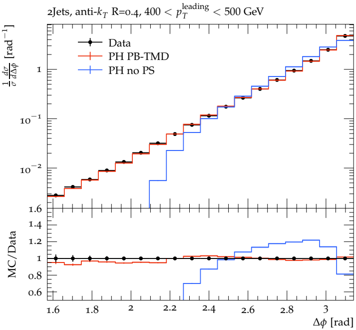
p_T distribution for NLO DY process

→ POWHEG+TMD+PS gives very good description & MC@NLO+TMD fits reasonable

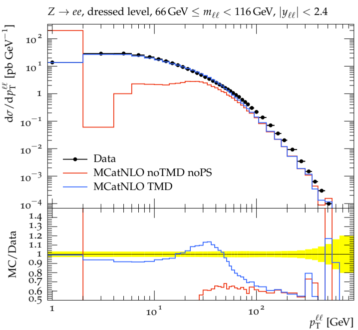
But how does it look like in detail?
How well does the parton shower really reproduce the TMDs?

The TMD parton shower and its applications

How can this TMD parton shower be applied?



$\Delta\Phi$ distribution for NLO dijets



p_T distribution for NLO DY process

→ POWHEG+TMD+PS gives very good description & MC@NLO+TMD fits reasonable

But how does it look like in detail?
How well does the parton shower really reproduce the TMDs?

→ **Perform a parton shower study to investigate how well the shower follows the TMDs**

How to do the PS study?

How to do the PS study?

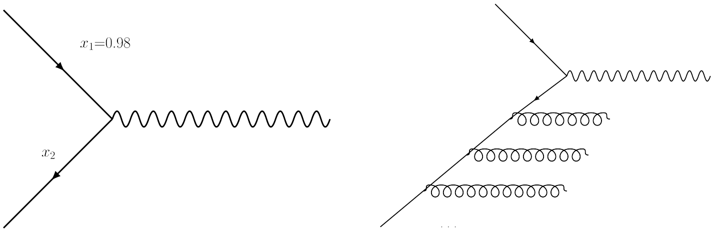
- define a **simple toy process** with a fixed x for one incoming parton, from which one can easily calculate kinematics of the other incoming parton, just using 4-vectors

How to do the PS study?

- define a **simple toy process** with a fixed x for one incoming parton, from which one can easily calculate kinematics of the other incoming parton, just using 4-vectors
- fix x_1=0.98 with no intrinsic $k_\perp$ and no PS from this incoming parton

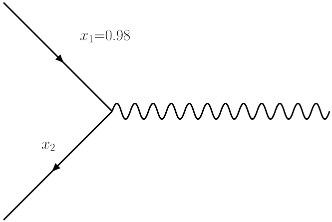
How to do the PS study?

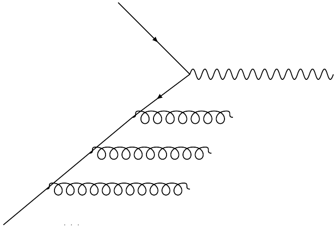
- define a **simple toy process** with a fixed x for one incoming parton, from which one can easily calculate kinematics of the other incoming parton, just using 4-vectors
- fix x_1=0.98 with no intrinsic $k_\perp$ and no PS from this incoming parton



How to do the PS study?

- define a **simple toy process** with a fixed x for one incoming parton, from which one can easily calculate kinematics of the other incoming parton, just using 4-vectors
- fix x_1=0.98 with no intrinsic $k_\perp$ and no PS from this incoming parton



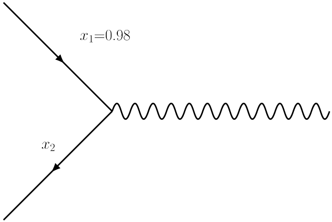


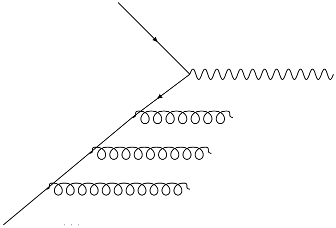
Technical details:

- generate hard process (PYTHIA) and "add" TMD and PS (CASCADE 3)

How to do the PS study?

- define a **simple toy process** with a fixed x for one incoming parton, from which one can easily calculate kinematics of the other incoming parton, just using 4-vectors
- fix x_1=0.98 with no intrinsic $k_\perp$ and no PS from this incoming parton





Technical details:

- generate hard process (PYTHIA) and "add" TMD and PS (CASCADE 3)
- analyse events, calculate kinematics, determine effective TMD

TMD from final state events

PDF set PYTHIA

hard process is generated with PYTHIA using:

- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set1
- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set2

TMD PDF set CASCADE

$k_\perp$ is "added" by CASCADE 3 using:

- PB-NLO-HERAI+II-2018-set1
- PB-NLO-HERAI+II-2018-set2

TMD from final state events

PDF set PYTHIA

hard process is generated with PYTHIA using:

- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set1
- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set2

TMD PDF set CASCADE

$k_\perp$ is "added" by CASCADE 3 using:

- PB-NLO-HERAI+II-2018-set1
- PB-NLO-HERAI+II-2018-set2

Difference between set1 & set2: set1 $\rightarrow$ $\alpha_s\left(\mu^2\right)$, set2 $\rightarrow$ $\alpha_s\left((1-z)^2\,\mu^2\right)$

TMD from final state events

PDF set PYTHIA

hard process is generated with PYTHIA using:

- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set1
- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set2

TMD PDF set CASCADE

$k_\perp$ is "added" by CASCADE 3 using:

- PB-NLO-HERAI+II-2018-set1
- PB-NLO-HERAI+II-2018-set2

Difference between set1 & set2: set1 $\rightarrow \ \alpha_s\left(\mu^2\right)$, set2 $\rightarrow \ \alpha_s\left(\left(1-z\right)^2\mu^2\right)$

$\rightarrow$ set1 integrated over $k_\perp$ is just collinear PDF - this is not the case for set2!

TMD from final state events

PDF set PYTHIA

hard process is generated with PYTHIA using:

- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set1
- LHAPDF6:PB-TMDNLO-HERAI+II-2018-set2

TMD PDF set CASCADE

$k_\perp$ is "added" by CASCADE 3 using:

- PB-NLO-HERAI+II-2018-set1
- PB-NLO-HERAI+II-2018-set2

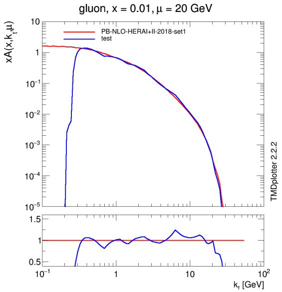
Difference between set1 & set2: set1 $\rightarrow$ $\alpha_s\left(\mu^2\right)$, set2 $\rightarrow$ $\alpha_s\left(\left(1-z\right)^2\mu^2\right)$

$\rightarrow$ set1 integrated over $k_\perp$ is just collinear PDF - this is not the case for set2!

$\rightarrow$ choose three different combinations of the collinear and TMD PDF sets to check consistency with the input TMDs

Consistency check: TMD vs. $k_\perp$

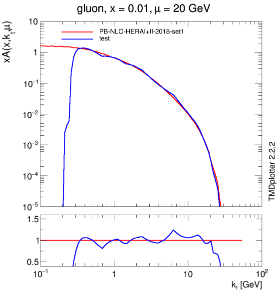
"test"=calculated distribution

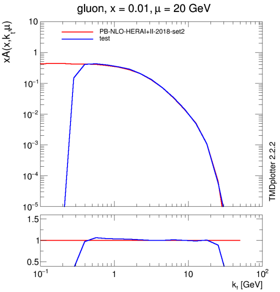


PYTHIA:
PB-TMDNLO-HERAI+II-2018-set1
CASCADE:
PB-NLO-HERAI+II-2018-set1

Consistency check: TMD vs. $k_\perp$

"test"=calculated distribution



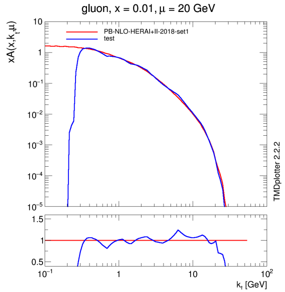


PYTHIA:
PB-TMDNLO-HERAI+II-2018-set1
CASCADE:
PB-NLO-HERAI+II-2018-set1

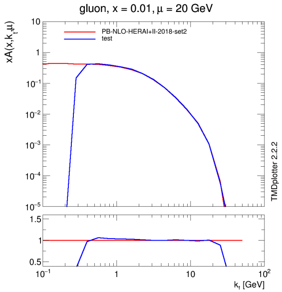
PYTHIA:
PB-TMDNLO-HERAI+II-2018-set2
CASCADE:
PB-NLO-HERAI+II-2018-set2

Consistency check: TMD vs. $k_\perp$

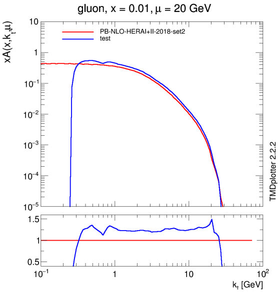
"test"=calculated distribution



PYTHIA:
PB-TMDNLO-HERAI+II-2018-set1
CASCADE:
PB-NLO-HERAI+II-2018-set1



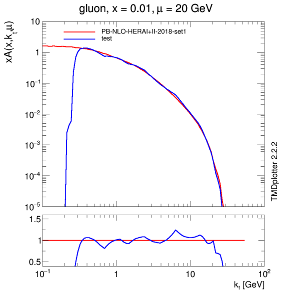
PYTHIA:
PB-TMDNLO-HERAI+II-2018-set2
CASCADE:
PB-NLO-HERAI+II-2018-set2



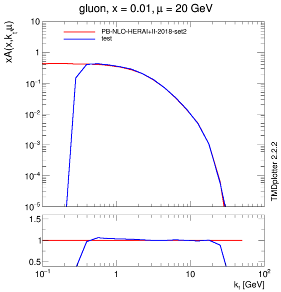
PYTHIA:
PB-TMDNLO-HERAI+II-2018-set1
CASCADE:
PB-NLO-HERAI+II-2018-set2

Consistency check: TMD vs. $k_\perp$

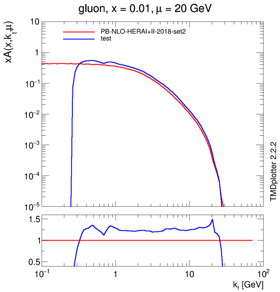
"test"=calculated distribution



PYTHIA:
PB-TMDNLO-HERAI+II-2018-set1
CASCADE:
PB-NLO-HERAI+II-2018-set1



PYTHIA:
PB-TMDNLO-HERAI+II-2018-set2
CASCADE:
PB-NLO-HERAI+II-2018-set2



PYTHIA:
PB-TMDNLO-HERAI+II-2018-set1
CASCADE:
PB-NLO-HERAI+II-2018-set2

→ reconstruction of differences between set1 & set2

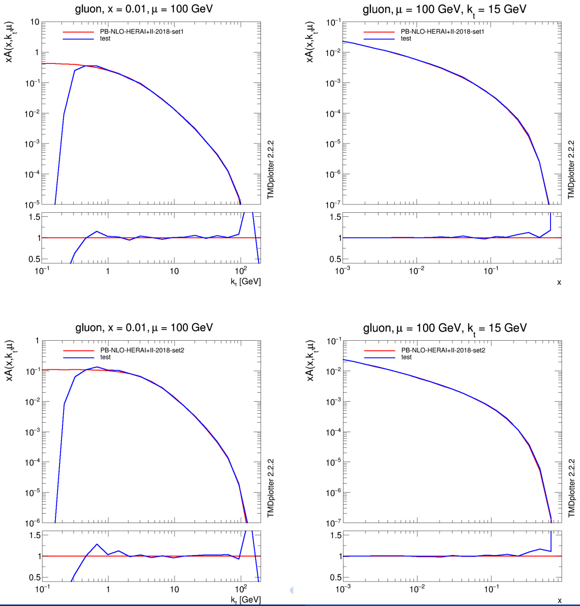
→ consistent use of collinear and TMD PDF set essential!

TMD parton shower: TMD vs. x and $k_\perp$

How does it look like when **PS is added**?
(set1 above, set2 below)

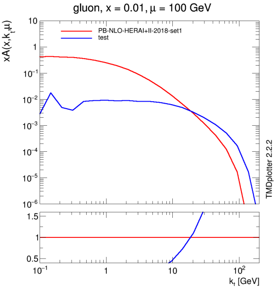
"test"=calculated distribution

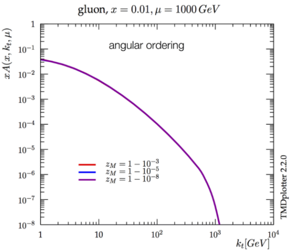
- **PS does not change the kinematics!**
- method can be used to determine TMDs from the shower



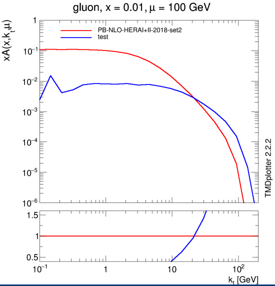
How does it look like for the **PYTHIA PS**?
(set1 above, set2 below)

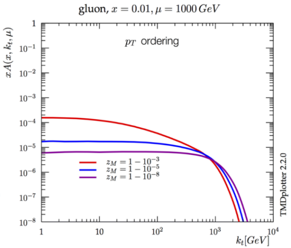
"test"=calculated distribution





- differences observed since P8 PS uses **p_T-ordering**
- TMDs can be determined from any parton shower



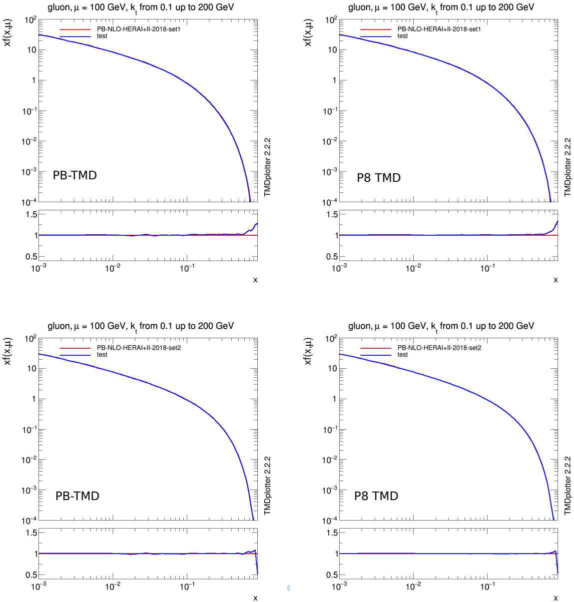


iTMD vs. x for both showers

How do the **integrated densities vs. x** look like for both showers?
(set1 above, set2 below)

"test"=calculated distribution

→ integration over $k_\perp$ gives back collinear PDF for both showers

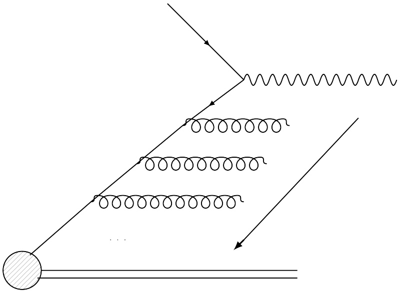


TMD parton shower: PS reconstruction

- the propagator partons in the PS are investigated

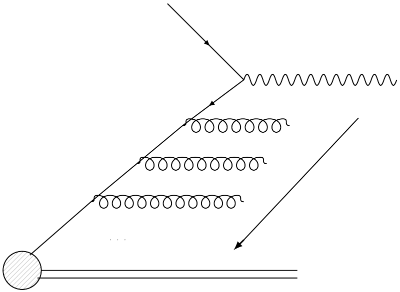
TMD parton shower: PS reconstruction

- the propagator partons in the PS are investigated
- the emitted partons of the IPS are ordered in rapidity to reconstruct the history of particle evolution (backward evolution)



TMD parton shower: PS reconstruction

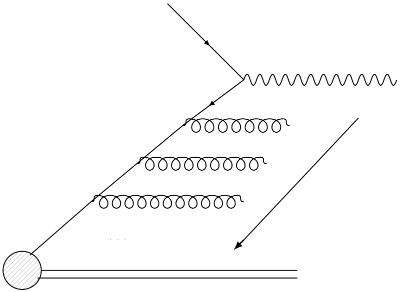
- the propagator partons in the PS are investigated
- the emitted partons of the IPS are ordered in rapidity to reconstruct the history of particle evolution (backward evolution)



- for every propagator the transverse momentum and the momentum fraction are calculated

TMD parton shower: PS reconstruction

- the propagator partons in the PS are investigated
- the emitted partons of the IPS are ordered in rapidity to reconstruct the history of particle evolution (backward evolution)

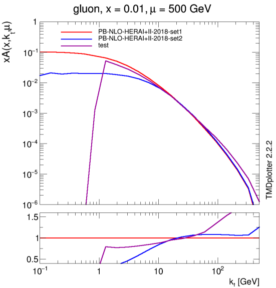


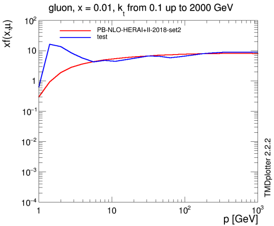
- for every propagator the transverse momentum and the momentum fraction are calculated
- the scales are equal to the rescaled transverse momenta of the emitted partons, $q_i = \frac{p_{ti}}{1-z_i}$

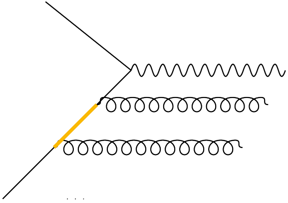
TMD parton shower: TMD vs. $k_\perp$ and iTMDs

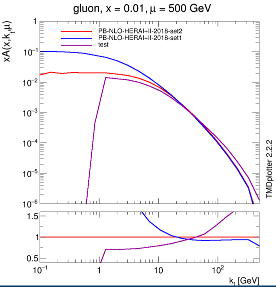
How does it look like for the **first propagator parton** of the PS?
(set1 above, set2 below)

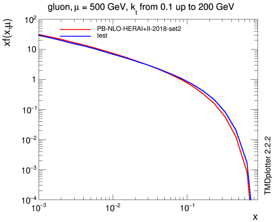
"test"=calculated distribution











Conclusion

- method described to determine TMDs from final state
  $\rightarrow$ prove of concept using TMDs
  $\rightarrow$ distributions exactly reproduced

Conclusion

- method described to determine TMDs from final state
  $\rightarrow$ prove of concept using TMDs
  $\rightarrow$ distributions exactly reproduced
- advantage of concept of TMD: **PS does not change kinematics**

Conclusion

- method described to determine TMDs from final state
  $\rightarrow$ prove of concept using TMDs
  $\rightarrow$ distributions exactly reproduced
- advantage of concept of TMD: **PS does not change kinematics**
- method applied to standard PS MC
  $\rightarrow$ PYTHIA
  $\rightarrow$ differences in TMD distributions to PB observed

Conclusion

- method described to determine TMDs from final state

  $\rightarrow$ prove of concept using TMDs

  $\rightarrow$ distributions exactly reproduced
- advantage of concept of TMD: **PS does not change kinematics**
- method applied to standard PS MC

  $\rightarrow$ PYTHIA

  $\rightarrow$ differences in TMD distributions to PB observed
- method applied to first propagator in the TMD PS

  $\rightarrow$ TMDs can be obtained

Conclusion

- method described to determine TMDs from final state

  $\rightarrow$ prove of concept using TMDs

  $\rightarrow$ distributions exactly reproduced
- advantage of concept of TMD: **PS does not change kinematics**
- method applied to standard PS MC

  $\rightarrow$ PYTHIA

  $\rightarrow$ differences in TMD distributions to PB observed
- method applied to first propagator in the TMD PS

  $\rightarrow$ TMDs can be obtained

This is the first time that parton densities are reconstructed from the shower history!

Thank you for your attention

BACKUP

The MC Event Generator CASCADE

- full hadron level Monte Carlo event generator for ep, γp and $p\bar{p}$ and pp processes
- previously intended for small-x processes, now extended for all x and all $k_\perp$
- uses TMD ... and TMD shower
- hadronisation is performed using the Lund string fragmentation implemented in PYTHIA

More information about TMDs

- two main sets of data provide the experimental information on TMDs: DIS at high energy, and low-q_T Drell Yan and semi-inclusive DIS
- for collinear and TMD distributions the factorization theorems are different and hence also the evolution equations
- TMDs can be obtained for example from the parton branching method with CCFM or DGLAP evolution equation
- some observables can not be predicted with collinear factorization $\rightarrow$ example given by Drell-Yan Z-boson production

References

Hannes Jung et. al.(2010)
The CCFM Monte Carlo generator CASCADE 2.2.0
Eur.Phys.J.C70:1237-1249, 2010

Marcin Bury et. al.(2017)
Calculations with off-shell matrix elements, TMD parton densities and TMD parton showers
arXiv:1712.05932 [hep-ph], 2017

F. Hautmann, H. Jung, A. Lelek, V. Radescu, R. Zlebcik (2017)
Collinear and TMD Quark and Gluon Densities from Parton Branching Solution of QCD Evolution Equations
arXiv:1708.03279 [hep-ph], 2017