Simulating dark showers

Suchita Kulkarni (she/her)
Junior group leader

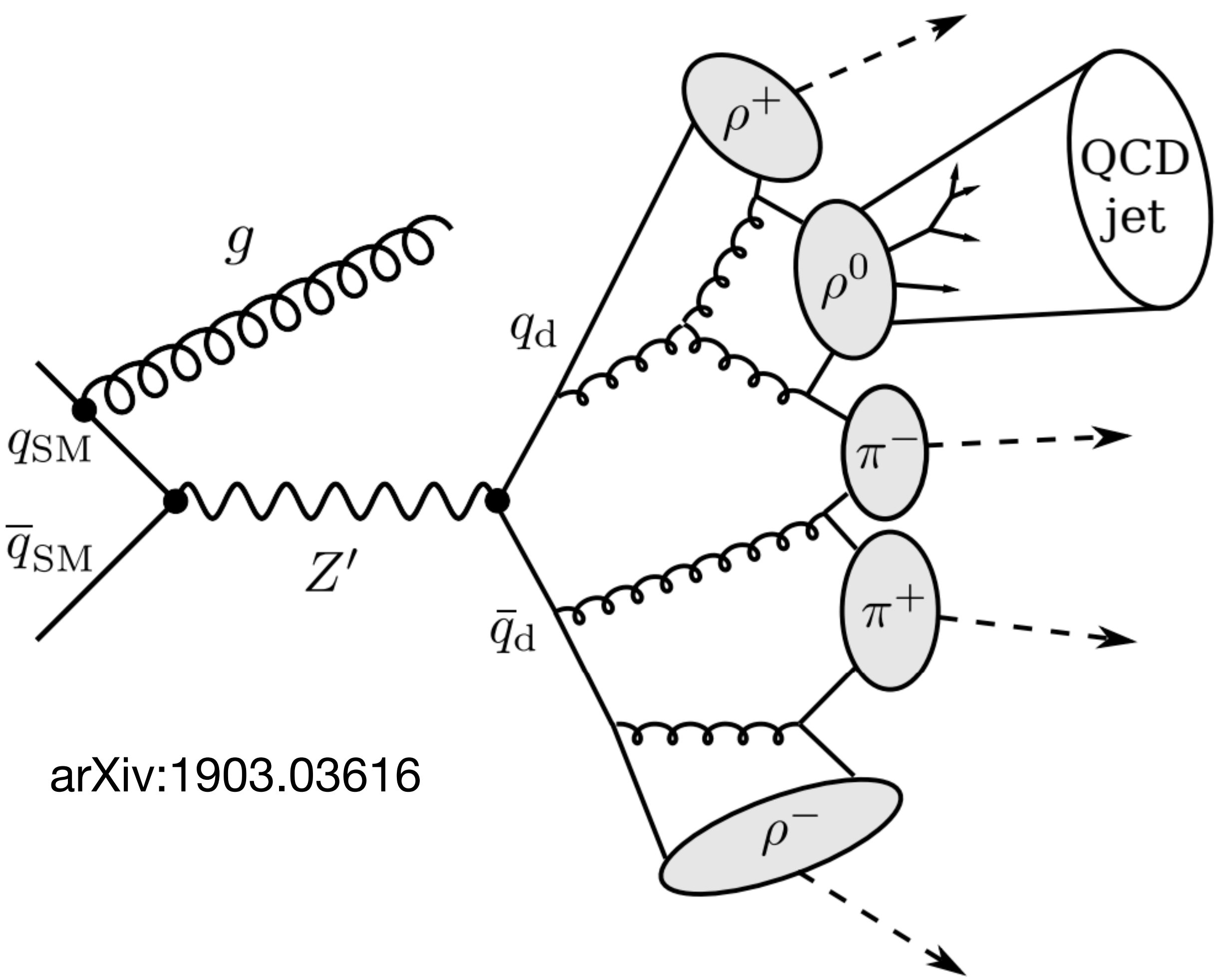
@suchi_kulkarni

Aims

1. The tutorial is designed as a hands on experience
2. The aim is to familiarise ourselves with the Hidden Valley module of Pythia and also with some of the other tools available to us
3. Plan of action
    1. Introduction to the semi-visible jets signature and a sample Pythia card
    2. Simulation and analysis of semi-visible jets signature at generator level
    3. Introduction to the dark showers tool
    4. Introduction to SUEPs and associated simulation technology
4. Disclaimer: The exact settings of your Pythia card will depend on details of your model, we provide an example, not the final setup.

Semi-visible jets

1. One of the many interesting signatures of strongly interacting dark matter
2. Features missing energy aligned in the direction of jets
3. Scenario we consider, SU(N_D) sector with N_f number of fermions in fundamental representation. These are uncharged under SM gauge groups. We connect SM and dark sector with a Z' gauge boson.



arXiv:1903.03616

Pythia card

1. We will generate the hard process p p > Z’ > qD qD using Pythia and let is shower and hadronize in SM and dark sector.
2. The Pythia card takes care of shower and hadronization parameters as well as the hard process

```
!contact: Suchita Kulkarni
!suchita.kulkarni@cern.ch; suchita.kulkarni@gmail.com
!This card generates semi-visible jets signature using pythia hidden valley module

! Settings that will be used in a main program.
Main:timesAllowErrors = 30          ! abort run after this many flawed events
! specify number of events
Main:numberOfEvents = 10000
Random:setSeed = on
Random:seed = 111

! Beam parameter settings. Values below agree with default ones.
Beams:idA = 2212                   ! first beam, p = 2212, pbar = -2212
Beams:idB = 2212                   ! second beam, p = 2212, pbar = -2212
Beams:eCM = 13000.                 ! CM energy of collision
! uncomment the following if you want to provide external LHE file
! Beams:frameType = 4                ! read info from a LHEF
! change input file name here
! Beams:LHEF = unweighted_events.lhe    ! the LHEF to read from

! Settings related to output in init(), next() and stat().
Init:showChangedSettings = on      ! list changed settings
Init:showAllSettings = off         ! list all settings
Init:showOneParticleData = 4900023  ! list changed particle data
Init:showAllParticleData = off     ! list all particle data
Next:numberCount = 1000            ! print message every n events
Next:numberShowLHA = 1             ! print LHA information n times
Next:numberShowInfo = 1            ! print event information n times
Next:numberShowProcess = 1         ! print process record n times
Next:numberShowEvent = 1           ! print event record n times
!Stat:showPartonLevel = on          ! additional statistics on MPI

! decouple the HV particles which are charged under both SM and HV charge
4900001:m0 = 5000
4900002:m0 = 5000
4900003:m0 = 5000
4900004:m0 = 5000
4900005:m0 = 5000
4900006:m0 = 5000
4900011:m0 = 5000
4900012:m0 = 5000
4900013:m0 = 5000
4900014:m0 = 5000
4900015:m0 = 5000
4900016:m0 = 5000

! Now we setup the process where we generate pp > Z' > qD qD
! Then we let qD hadronize
```

Docker commands

1. In order to use this card to produce events, you should have docker image on your machines

   `docker pull mgenest/dark-shower-tutorial`

2. Check if you have pulled the correct image

   `docker images`

```
(base) suchitakulkarni@Suchitas-MacBook-Pro ~ % docker images
REPOSITORY                     TAG       IMAGE ID       CREATED          SIZE
mgenest/dark-shower-tutorial   latest    e38dec7feaaa   22 hours ago     7.28GB
```

3. To get a an interactive shell on your command line

   `docker run -it mgenest/dark-shower-tutorial sh`

```
(base) suchitakulkarni@Suchitas-MacBook-Pro ~ % docker run -it mgenest/dark-shower-tutorial sh
sh-4.2#
```

Docker commands

1. `cd pythia8245/examples`
2. `make maindark`
3. `./maindark darkshower_semi_visible.cmnd`
4. Output of maindark

```
-rw-r--r-- 1 root root  3.7K May 27 10:31 mZp-1.dat
-rw-r--r-- 1 root root  3.7K May 27 10:31 mZp-0.dat
-rw-r--r-- 1 root root  3.7K May 27 10:31 pTmiss-0.dat
-rw-r--r-- 1 root root  3.7K May 27 10:31 pTjet-1.dat
-rw-r--r-- 1 root root  3.7K May 27 10:31 pTjet-0.dat
-rw-r--r-- 1 root root  3.7K May 27 10:31 dphiJetMET-1.dat
-rw-r--r-- 1 root root  3.7K May 27 10:31 dphiJetMET-0.dat
-rw-r--r-- 1 root root   894 May 27 10:31 dphiJetMET.py
-rw-r--r-- 1 root root   893 May 27 10:31 ZprimeMass.py
-rw-r--r-- 1 root root   634 May 27 10:31 MET.py
-rw-r--r-- 1 root root   877 May 27 10:31 JetpT.py
```

5. `python3.8 Zprimemass.py`

This will create a file called mZp.pdf containing zprime mass histogram

Copy file out of docker

1. To do this first find out the name of your docker container

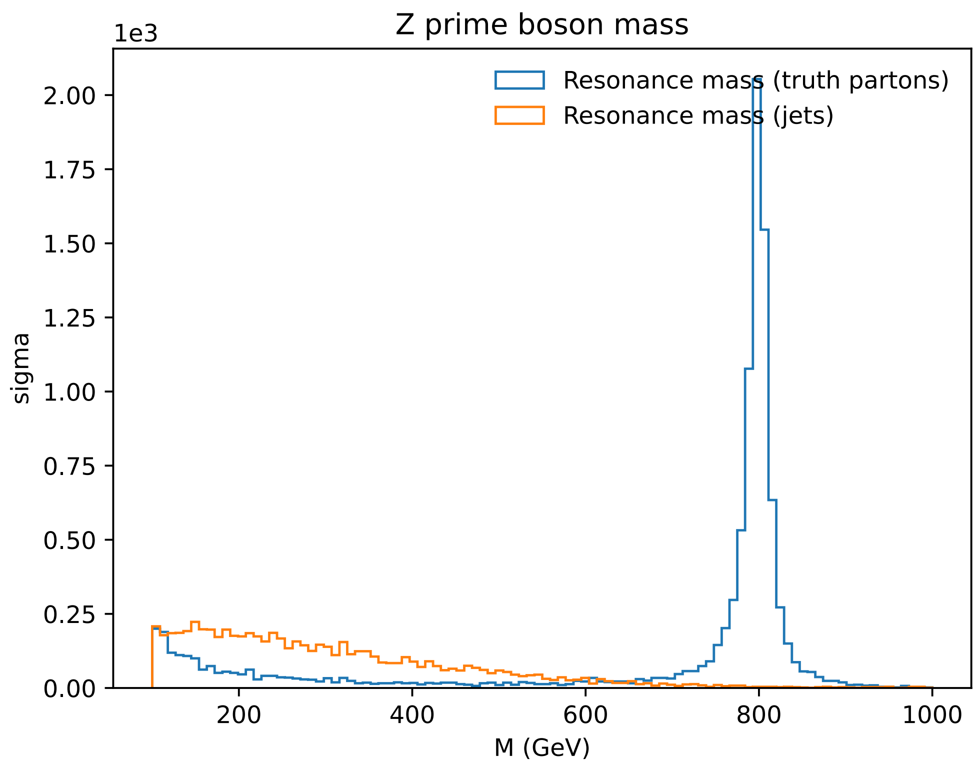
```
docker ps --format “{{.Names}}”
```

Name of my container `tender_babbage`, this is a random string and changes

2. Copy file out of docker to local path

```
docker cp <container name>:pythia8245/examples/mZp.pdf <local path>
```

```
(py27) suchitakulkarni@Suchitas-MacBook-Pro DS_tutorial % docker ps --format "{{.Names}}"
tender_babbage
(py27) suchitakulkarni@Suchitas-MacBook-Pro DS_tutorial % docker cp tender_babbage:pythia8245/examples/mZp.pdf ./
```

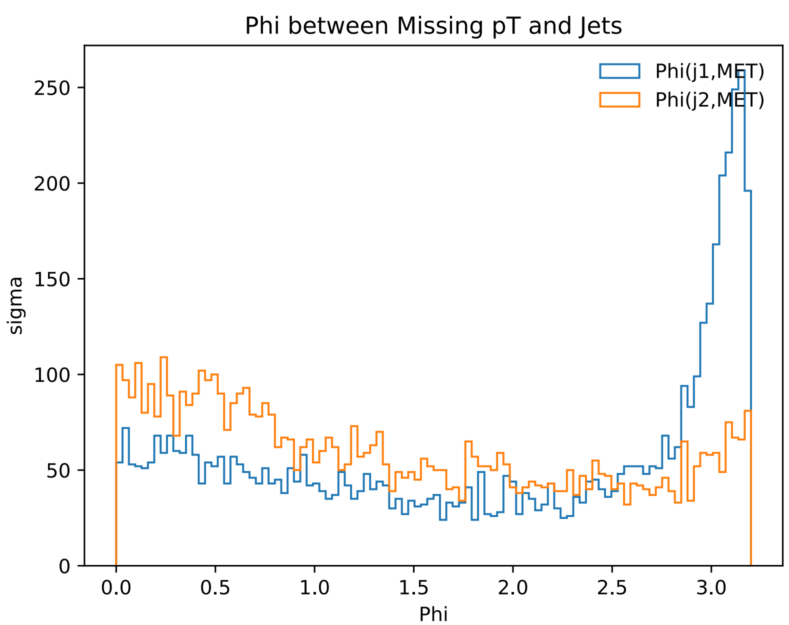


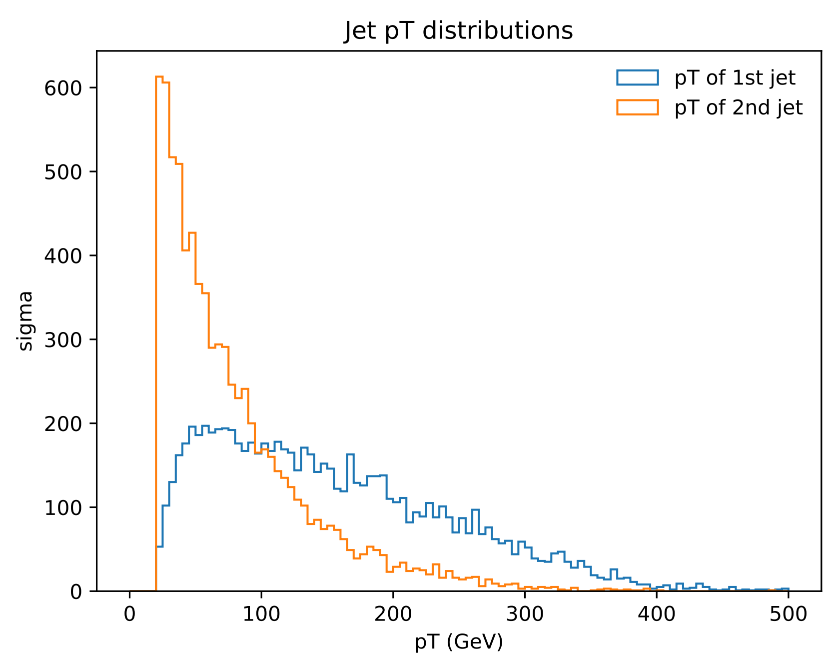
Z’ mass isn’t well reconstructed due to large missing energy component

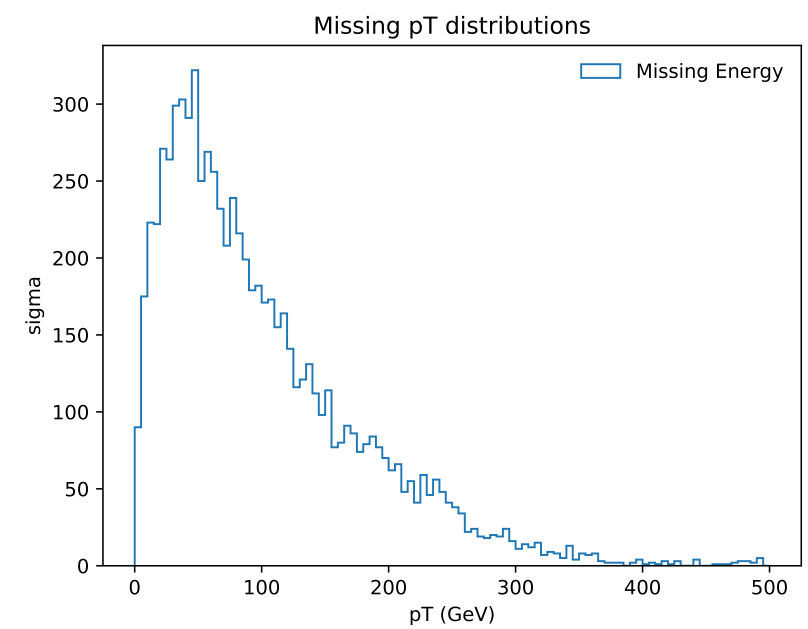
More distributions

You can also run other scripts to plot additional distributions

```
python3.8 MET.py
python3.8 JetpT.py
python3.8 dphiJetMET.py
```







Additional tasks

1. Current code uses slowjet clustering algorithm, the settings are done such that invisible particles are not counted in jet pT. If you want to check the effect of including invisible particle in kinematic distributions, change the slowjet parameters to

```
SlowJet slowJet( -1, 0.4, 20.0, 4.8, 2, 1);

SlowJet slowJet( -1, 0.4, 20.0, 4.8, 1, 2);
```

2. You can also change the mass of Z' to 2 TeV and redo the distributions to check the effect of higher scales. To do this, change range of histograms in `maindark.cc` and then change mass of Z' in the card.

```
Hist massPartonic("Resonance mass (truth partons)", 100, 100, 1000);

Hist massJets("Resonance mass (jets)", 100, 100, 1000);
```

3. Change endpoints in above definition to e.g. 3 TeV