FCC analysis flow
Z'->ll example

Clement Helsens

Outline

- Delphes cards proposal
- FCCSW analysis flow
- Z'->ll full example

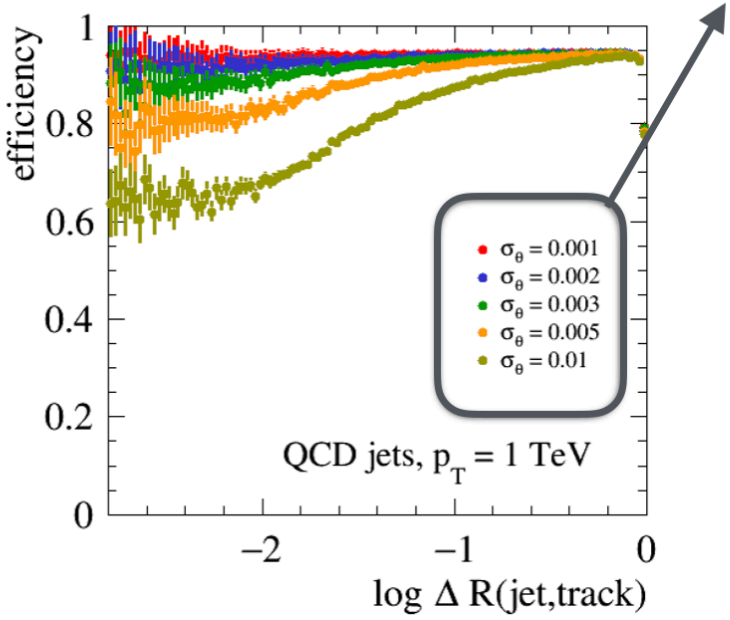
Delphes card proposal

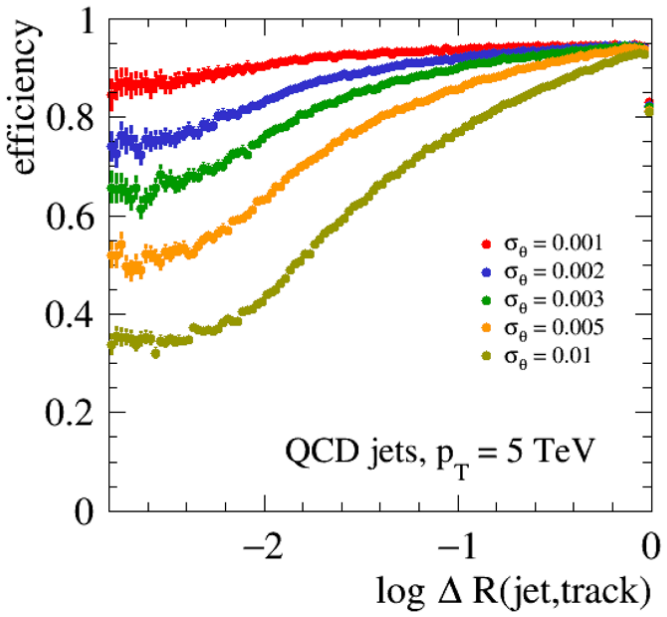
- As done for the ATLAS upgrade:
  - compare the nominal FCC Delphes parametrisation with a “better” and “worse” detector
- This will be very informative in terms of physics reach
  - different variations brackets the nominal
  - We will also present results using the CMS parametrisation
- Those variations (including CMS)
  - will be centrally supported
  - this does not prevent users to study more configurations if they want to, for example if only a specific aspect of the detector is relevant for an analysis

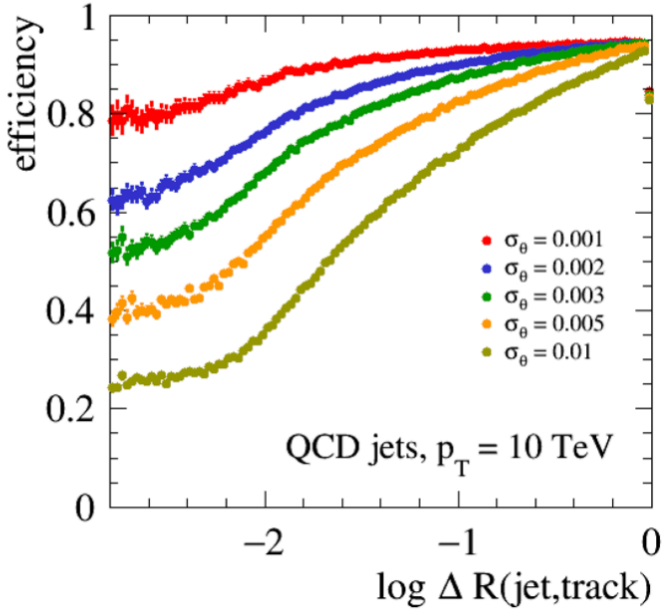
Track angular resolution

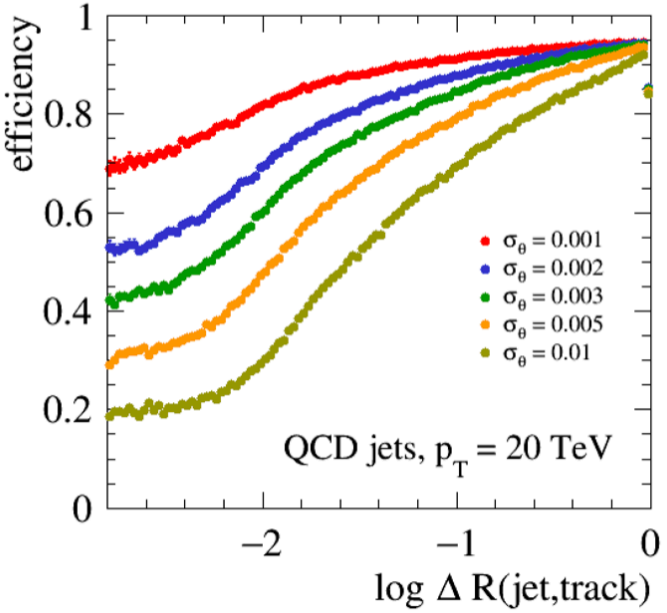
- In Delphes the track angular resolution is considered as perfect
  - This is a wrong assumption and it has been implemented in a new Delphes release (not yet propagated in FCCSW)
  - CMS value is consistent with σ_θ~0.003
  - Propose to choose 0.003 or even 0.002 as a conservative baseline
- Back of the envelop calculation with tracker experts
  - if we consider a track as merged if the 1st pixel layer is shared, lead to a σ_θ~0.004-0.006
  - should be able to distinguish 2 overlapping tracks even if they share the first 1-2-3 pixel layers?
  - Proposed number does not look completely crazy
- Of course a full simulation of the FCC tracker to study this track resolution would be more than welcome to validate the choice!

Instrinsic tracking angular resolution









Proposed variations

- All resolutions: x2, nominal, x0.5
- Tracking/ECAL/HCAL granularities: x2, nominal, x0.5

- Charged hadron efficiency: 90, 95, 99%
- Muon efficiency 95, 99, 100%
- Electron efficiency 90, 95, 99%
- Photon efficiency 85, 95, 100%

- B-tag efficiency (constant fake-rate): 75, 85, 95%
- C-tag efficiency (constant fake-rate): 35, 45, 55%

- After discussing with calorimeters colleagues, we might add 2 more variations keeping the granularity as nominal and vary the resolutions

FCCSW analysis flow

Fully integrated to the production system, links between

1. Generate events:
   - For the moment from MG5 gridpacks to produce LHE files

   Creates a database of LHE events available

2. Simulate events with FCCSW+Delphes
   - Directly from Pythia8
   - From the LHE
   - From HEPMC (previous talk)

   Creates a database of FCC EDM events available

3. Create flat ntuples with heppy
   - Using the events simulated in 2.

   Use the events in the database to produce analyses templates

4. Analyze the flat tree
   - Using events created in 3.
   - Create histograms, plots

   Use the events produced in heppy to create stacked plots

5. Limits significance
   - Using histograms created in 4.

   Get the cross sections

https://github.com/clementhelsens/EventProducer/tree/master

1. Generation

- Already explained a lot in the past
  - Start from Madgrap gridpacks
  - Create LHE files
  - List of available samples and statistics is available here: http://fcc-physics-events.web.cern.ch/fcc-physics-events/LHEevents.php
  - Most of the processes are binned in HT

https://github.com/clementhelsens/EventProducer/tree/master

2. Simulation (FCCSW Delphes)

- From the LHE files, create FCC EDM files with a given Delphes parametrisation
- Recently added the possibility to directly simulate events with Pythia8
  - Useful for bunch of signals

- Available samples:
- For FCC_v01 (nominal http://fcc-physics-events.web.cern.ch/fcc-physics-events/Delphesevents_fcc_v01.php

- For CMS http://fcc-physics-events.web.cern.ch/fcc-physics-events/Delphesevents_cms.php

https://github.com/HEP-FCC/heppy
https://github.com/clementhelsens/FCChhAnalyses/

3. Flat trees (heppy)

- Within heppy create an analysis directory that contains always the same files:
  - Analysis.py -> to define the list of modules to be run and the list of samples over which to run
    the inputs file lists and cross sections etc… is centrally defined and supported (from step 2.)

```
selectedComponents = [
    pp_ll012j_5f_HT_0_200,
    pp_ll012j_5f_HT_200_700,
    pp_ll012j_5f_HT_700_1500,
    pp_ll012j_5f_HT_1500_2700,
    pp_ll012j_5f_HT_2700_4200,
    pp_ll012j_5f_HT_4200_8000,
    pp_ll012j_5f_HT_8000_15000,
    pp_ll012j_5f_HT_15000_25000,
    pp_ll012j_5f_HT_25000_35000,
    pp_ll012j_5f_HT_35000_100000,
    pp_Zprime_5TeV_ll,
    pp_Zprime_10TeV_ll,
    pp_Zprime_15TeV_ll,
    pp_Zprime_20TeV_ll,
    pp_Zprime_25TeV_ll,
    pp_Zprime_30TeV_ll,
    pp_Zprime_35TeV_ll,
    pp_Zprime_40TeV_ll,
    pp_Zprime_45TeV_ll,
    pp_Zprime_50TeV_ll,
                        ]
```

https://github.com/HEP-FCC/heppy
https://github.com/clementhelsens/FCChhAnalyses/

3. Flat trees (heppy)

- Within heppy create an analysis directory that contains always the same files:
  - Analysis.py -> to define the list of modules to be run and the list of samples over which to run
  - Selection.py -> to define the list of pre-selections

```
def process(self, event):
    self.counters['cut_flow'].inc('All events')

    #select events with at least 2 leptons
    if len(event.selected_electrons)<2 and len(event.selected_muons)<2:
        return False
    self.counters['cut_flow'].inc('At least 2 same flavor leptons')

    return True
```

https://github.com/HEP-FCC/heppy
https://github.com/clementhelsens/FCChhAnalyses/

3. Flat trees (heppy)

- Within heppy create an analysis directory that contains always the same files:
  - Analysis.py -> to define the list of modules to be run and the list of samples over which to run
  - Selection.py -> to define the list of pre-selections
  - TreeProducer.py -> To define the variables to be stored in the output file
- The idea is to have all FCChh analyses defined in this form so that the analysis flow are fully reproducible, need the outputs to be stored on the eos/experiments/fcc/hh/analyses/
- List of analysis we have so far

RSGraviton_ww	Fix syntax errors
Zprime_ll	add 45 and 50 TeV Z' SSM to leptons
Zprime_tt	Fix syntax errors
h2l2v	added higgs studies configurations
h4l	added higgs studies configurations
haa	added higgs studies configurations
hmumu	added higgs studies configurations
hza	added higgs studies configurations

https://github.com/selvaggi/FlatTreeAnalyzer

4. Flat Tree analyser

- From the files produced in 3. produce stack plots and histograms for final analysis for different selections
- Selection based on variables available in the flatTree

```
# base pre-selections
selbase = 'lep1_pt > 200. && lep2_pt > 200. && abs(lep1_eta) < 4 && abs(lep2_eta) < 4 && zprime_ele_m>2000'
```

- Proposal is also to keep the analysis templates so that we can reproduce the results

- List of analyses so far:

zprime_ee.py	adding 45 and 50TeV mass points
zprime_mumu.py	adding 45 and 50TeV mass points
zprime_ttbar.py	first commit

https://github.com/clementhelsens/FCCFitter

5. Limit, Significance

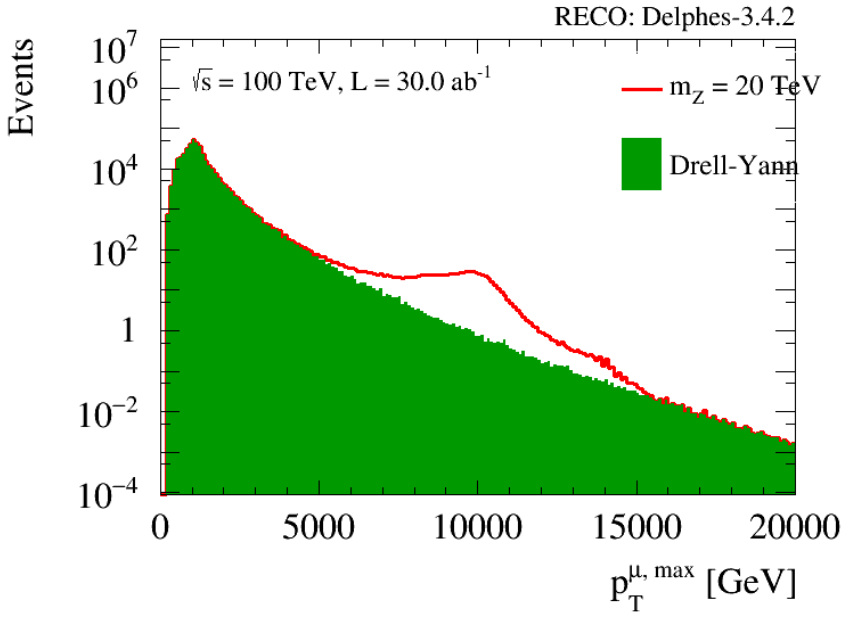
- Based on a tool heavily used in ATLAS
- ROOSTAT profile likelihood ratio
- Asymptotic limits
- Could add any kind of uncertainties (shape or normalizations)
- Combine channels and correlate nuisance parameters
- Full plotting available to produce:
  - Post-fit plots
  - Ranking plots
  - Correlation matrix
  - Pruning of systematics
- Other higher level macros to:
  - Produce confidence level exclusion plots versus mass
  - Scan the luminosity to get discovery potential

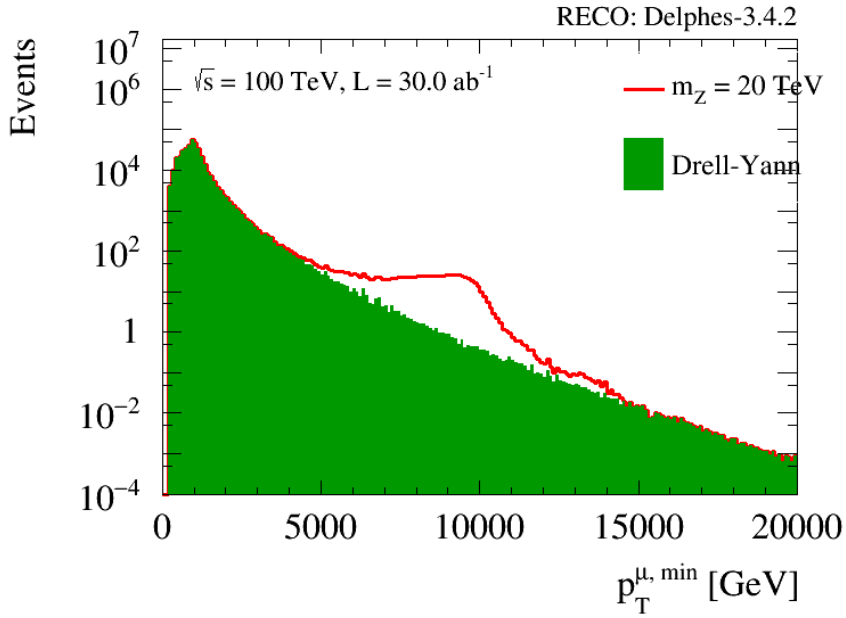
Z'->ll

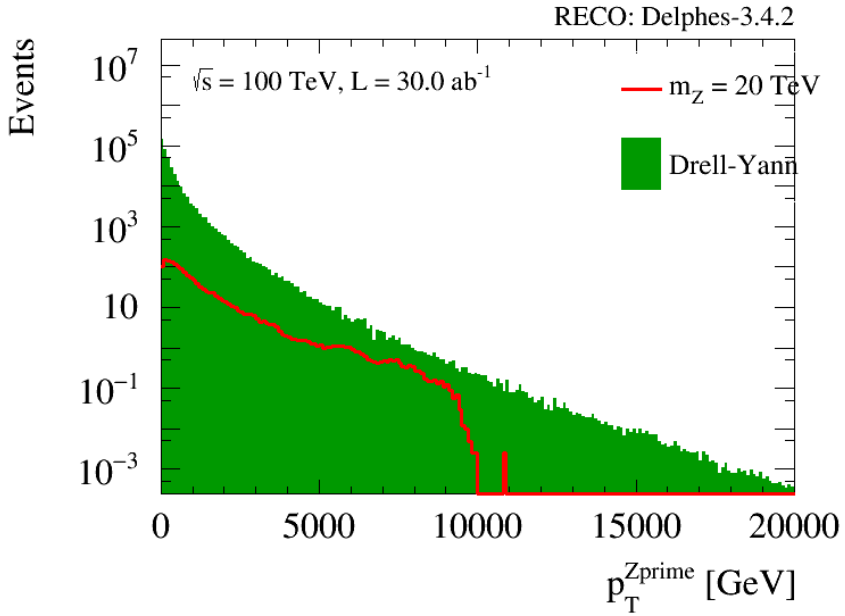
- Following the previous discussions
  - Re-did the discovery potential of Z'->ll
  - Using FCC (nominal only) and CMS parametrisations

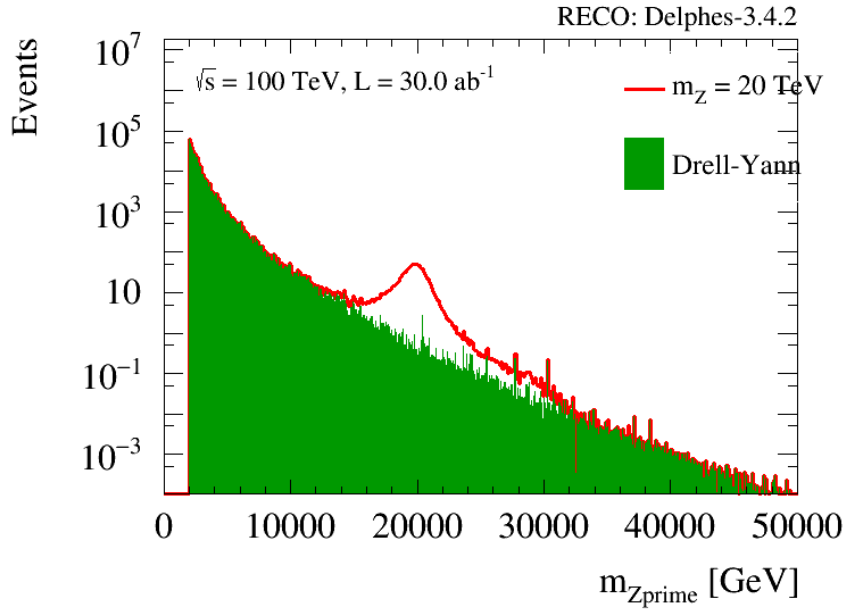
- Selection
  - 2 same flavor lepton, p_T>200GeV, M_{ll}>2TeV
  - Use full M_{ll} shape as discriminant
  - Asymptotic limits
  - 50% uncertainty on Drell-Yann
  - 30ab^{-1} luminosity for exclusion plots

FCC

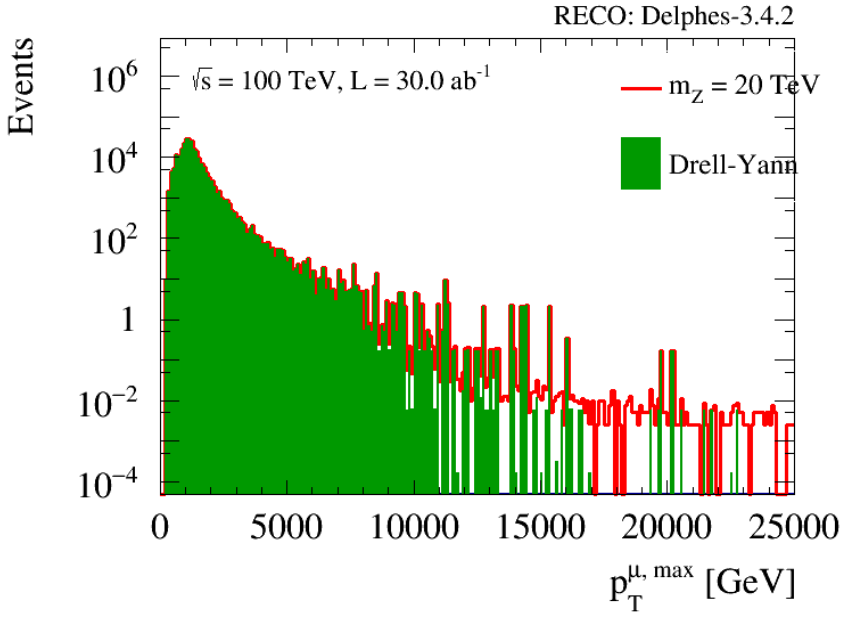

RECO: Delphes-3.4.2
Events
$\sqrt{s}$ = 100 TeV, L = 30.0 ab^{-1}
m_Z = 20 TeV
Drell-Yann
$p_T^{\mu,\,max}$ [GeV]

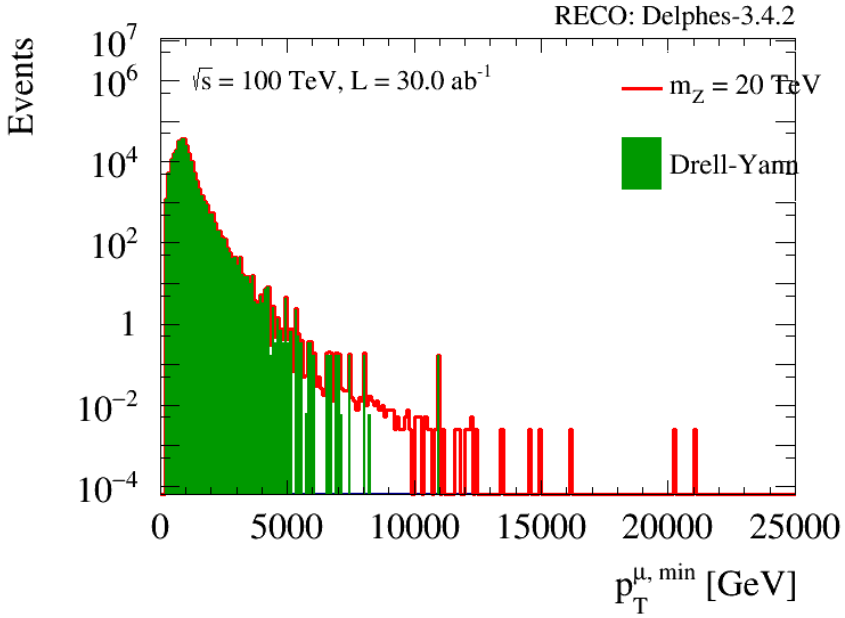

RECO: Delphes-3.4.2
Events
$\sqrt{s}$ = 100 TeV, L = 30.0 ab^{-1}
m_Z = 20 TeV
Drell-Yann
$p_T^{\mu,\,min}$ [GeV]

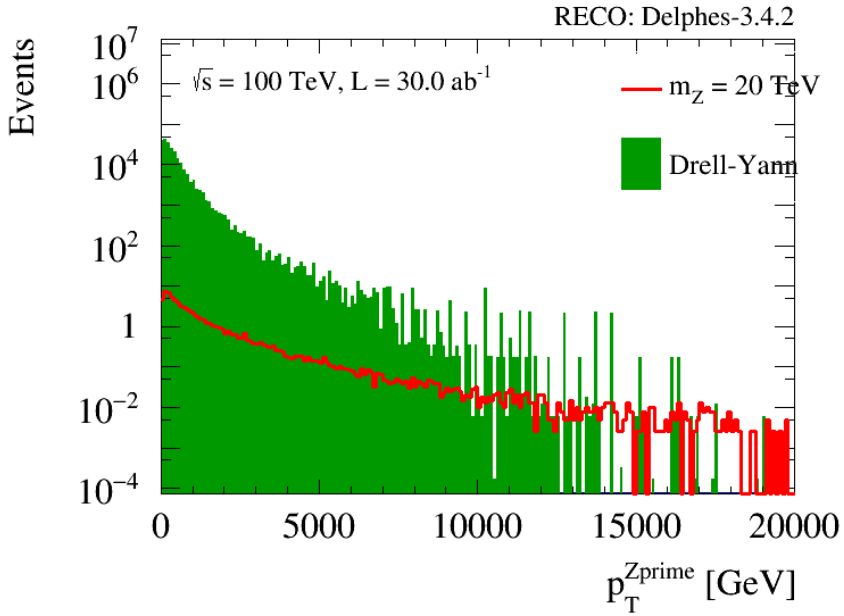

RECO: Delphes-3.4.2
Events
$\sqrt{s}$ = 100 TeV, L = 30.0 ab^{-1}
m_Z = 20 TeV
Drell-Yann
p_T^{Zprime} [GeV]

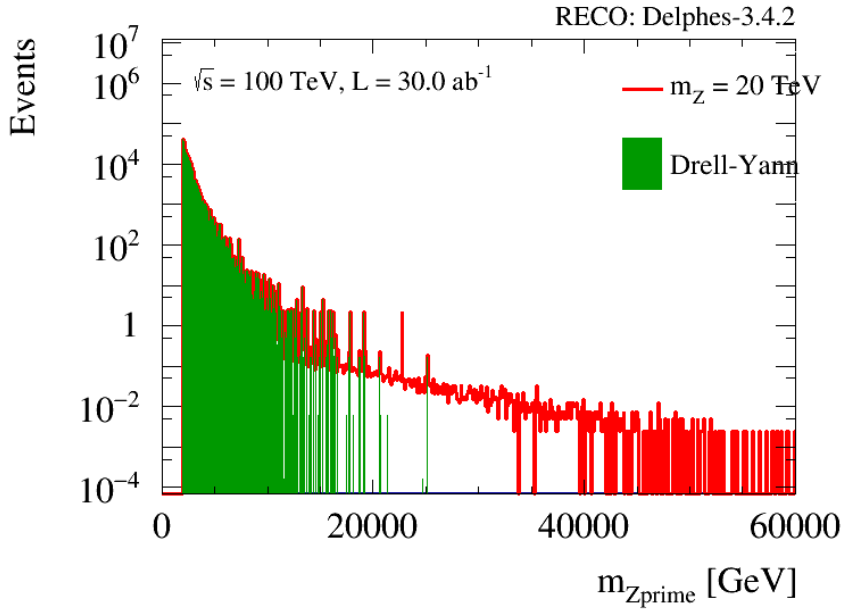

RECO: Delphes-3.4.2
Events
$\sqrt{s}$ = 100 TeV, L = 30.0 ab^{-1}
m_Z = 20 TeV
Drell-Yann
m_{Zprime} [GeV]

CMS

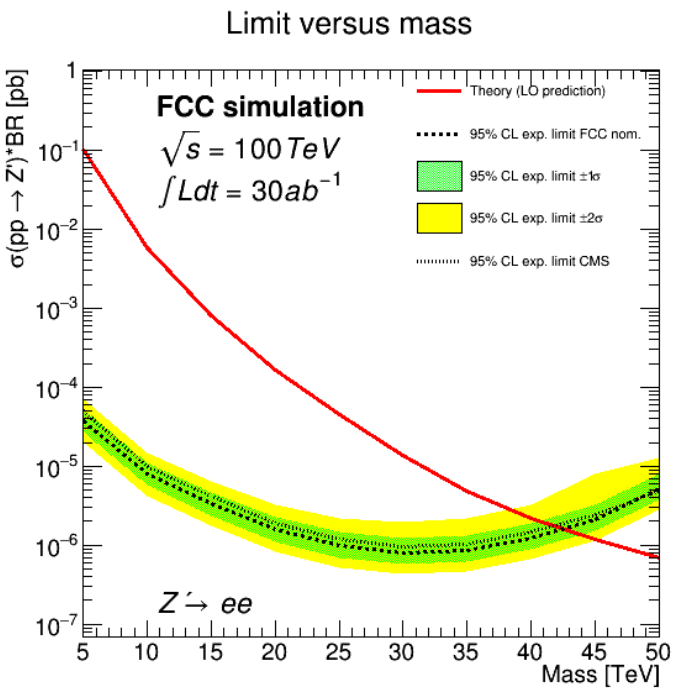

RECO: Delphes-3.4.2
√s = 100 TeV, L = 30.0 ab-1
mZ = 20 TeV
Drell-Yann
Events
p_T^μ, max [GeV]

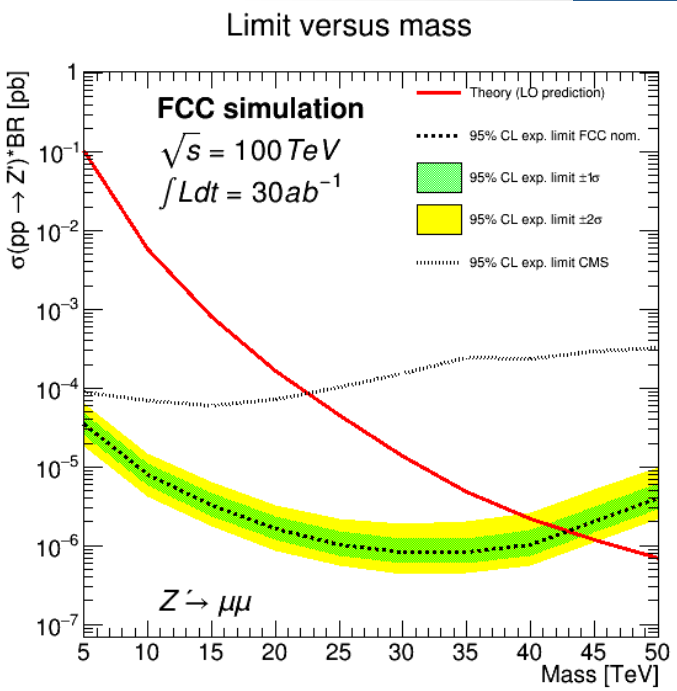

RECO: Delphes-3.4.2
√s = 100 TeV, L = 30.0 ab-1
mZ = 20 TeV
Drell-Yann
Events
p_T^μ, min [GeV]


RECO: Delphes-3.4.2
√s = 100 TeV, L = 30.0 ab-1
mZ = 20 TeV
Drell-Yann
Events
p_T^Zprime [GeV]


RECO: Delphes-3.4.2
√s = 100 TeV, L = 30.0 ab-1
mZ = 20 TeV
Drell-Yann
Events
m_Zprime [GeV]

Limits

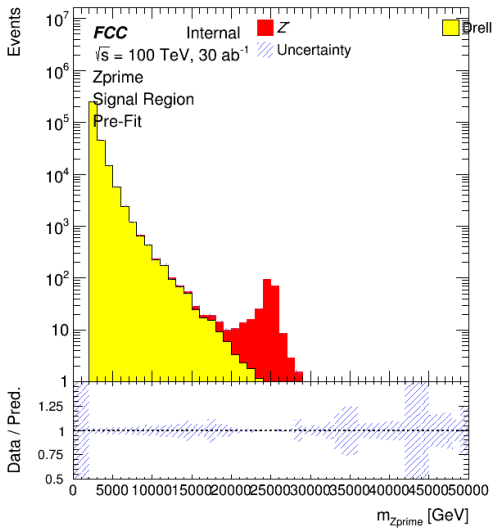


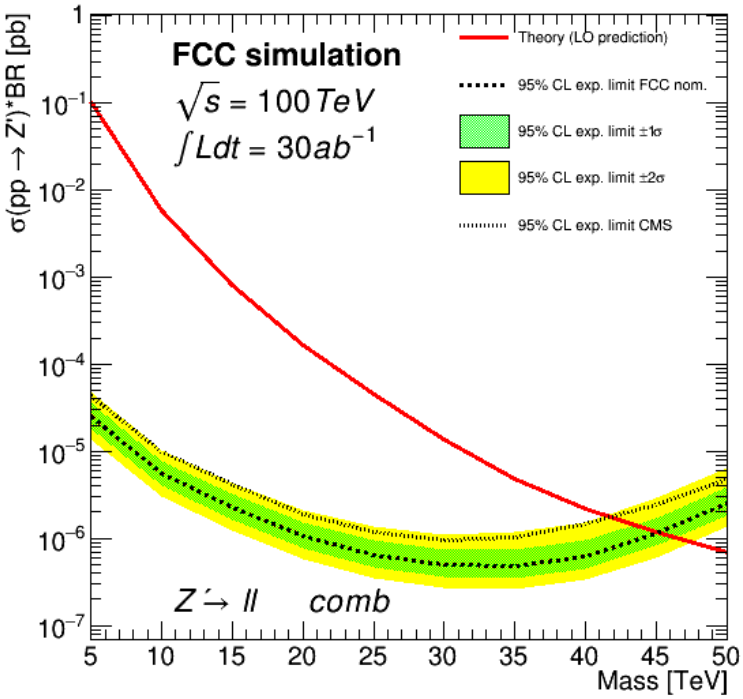


Similar reach for electrons

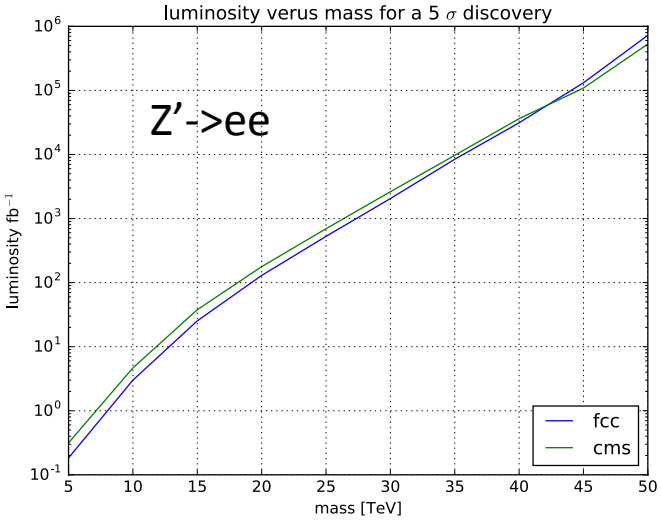
For muons of course things are different…

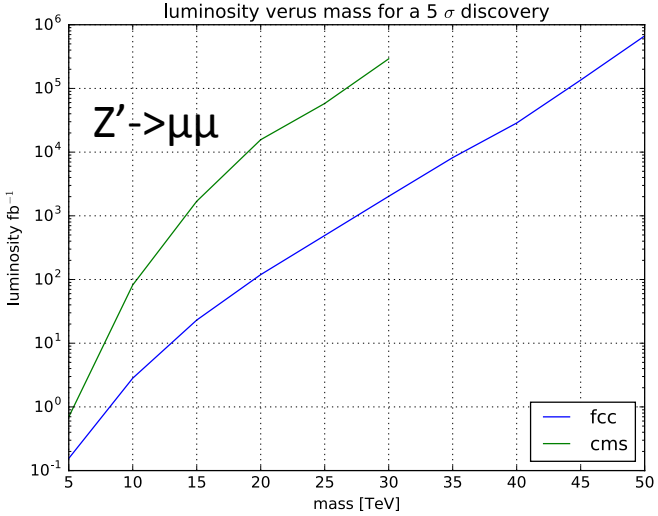
Expectation 45(42)TeV for FCC(CMS)



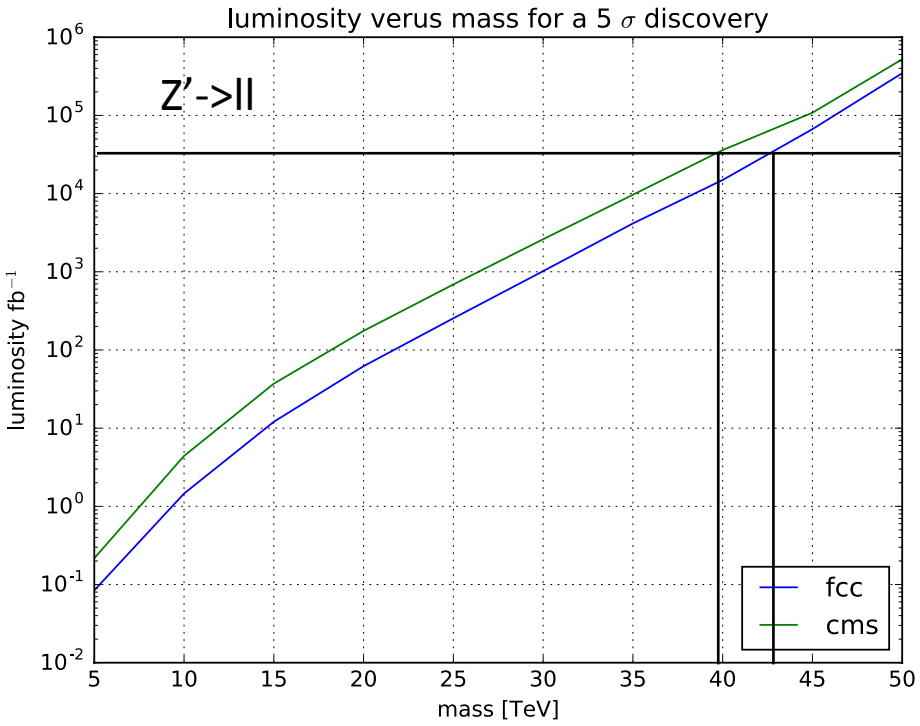


Significance





With 30ab^{-1} can discover up to:
43TeV for FCC
40TeV for CMS



Summary

- Delphes
  - card variations proposed to have possibly results that “brakets” the nominal detector parametrisation
  - Also support CMS card
- FCCSW
  - New analysis flow defined and fully operational
  - Make it as efficient as possible
  - Try to be as reproducible as possible
- Z’->ll
  - analysis within FCCSW framework being finalized
  - documentation in preparation together with other heavy resonances