Underlying event measurements and MC generator tuning with first LHC data

Holger Schulz, Heiko Lacker, Jan Eike von Seggern (HU Berlin),
Andy Buckley (Edinburgh), Hendrik Hoeth (Durham)

November 12, 2009
3rd Annual Workshop "Physics at the Terascale"

Introduction

- LHC is a QCD machine $\longrightarrow$ hard to find interesting signals
- QCD perturbatively calculable in hard processes
- Need $\boxed{\text{models for soft physics } (\alpha_s \not\ll 1)}$ to understand background
- Large background at LHC is Underlying Event (UE)
- UE $\approx$ everything except the hard scattering of interest
- Have different models/generators: Herwig, Pythia, Phojet, Sherpa ...
- LHC-predictions differ vastly
- $\longrightarrow$ need measurements to tune generators
- For early data: identify worthwile measurements first!

Outline

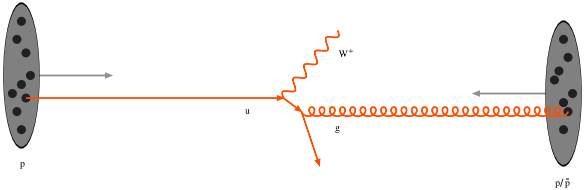
1. Underlying Event (UE)
2. Generator tuning with Professor
3. Tuning to early LHC data

Hadron collisions (borrowed from Leif Lönnblad)

Incoming beams, parton density functions (pdfs) & primordial $k_\perp$

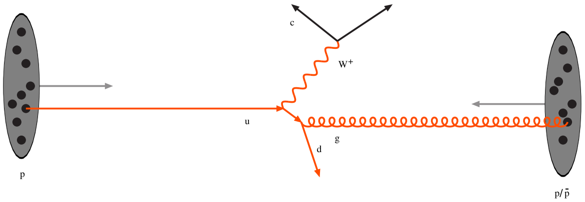
Hadron collisions (borrowed from Leif Lönnblad)

The hard sub-process, the matrix element

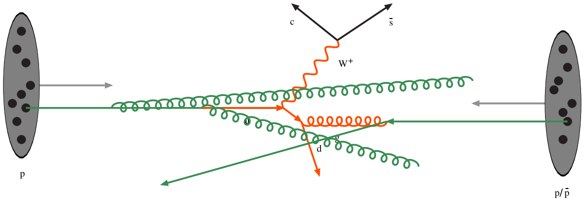


Hadron collisions (borrowed from Leif Lönnblad)

Resonance decays $\longrightarrow$ correlated with the hard sub-process

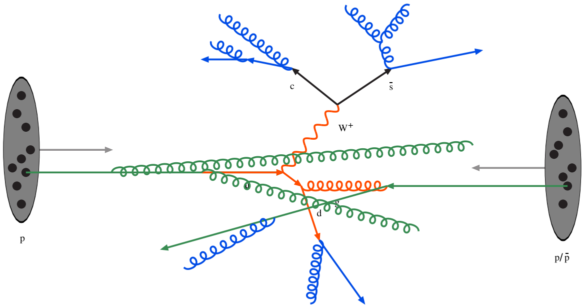


Initial-state radiation (ISR), parton shower (*backward* evolution)



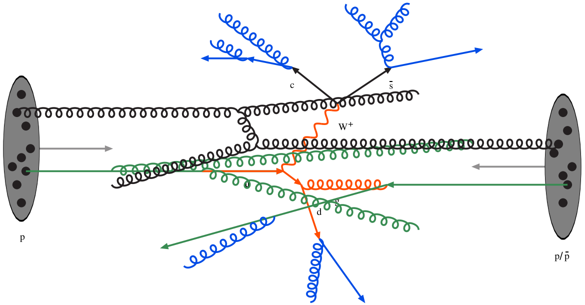
Hadron collisions (borrowed from Leif Lönnblad)

Final-state radiation (FSR), parton shower (*forward* evolution)



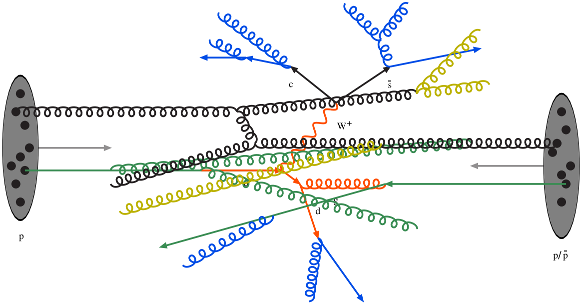
Hadron collisions (borrowed from Leif Lönnblad)

Multiple parton-parton interactions $\longrightarrow$ soft, semi-hard or hard scatterings



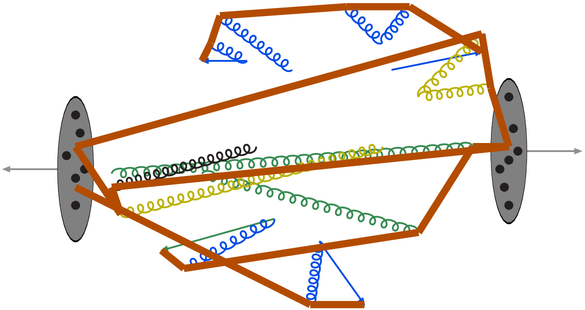
Hadron collisions (borrowed from Leif Lönnblad)

Initial-/Final state showers of ISR-particles

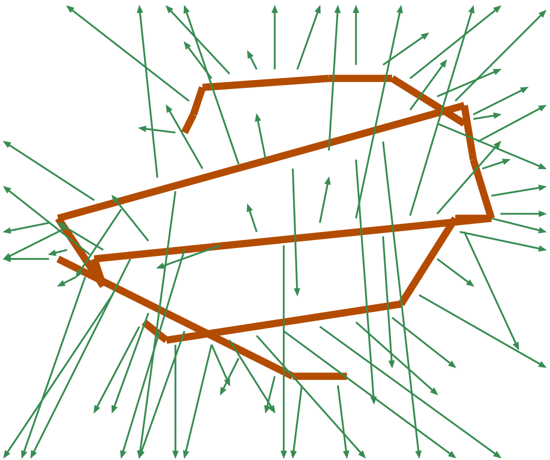


Hadron collisions (borrowed from Leif Lönnblad)

Formation of colour strings, outgoing partons & beam remnants

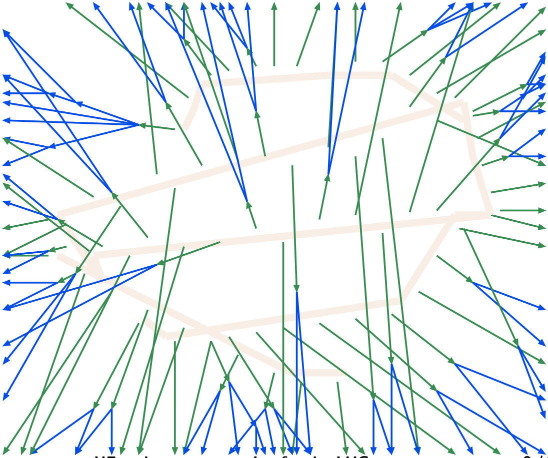


Hadronisation



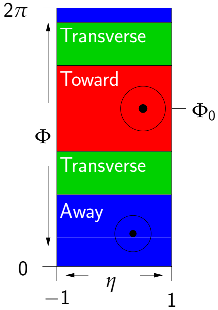
Hadron collisions (borrowed from Leif Lönnblad)

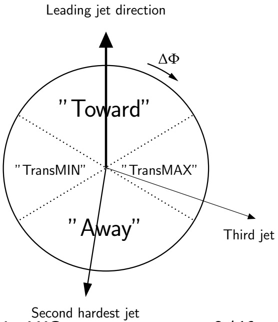
Decay of unstable particles, this is what hits the detector

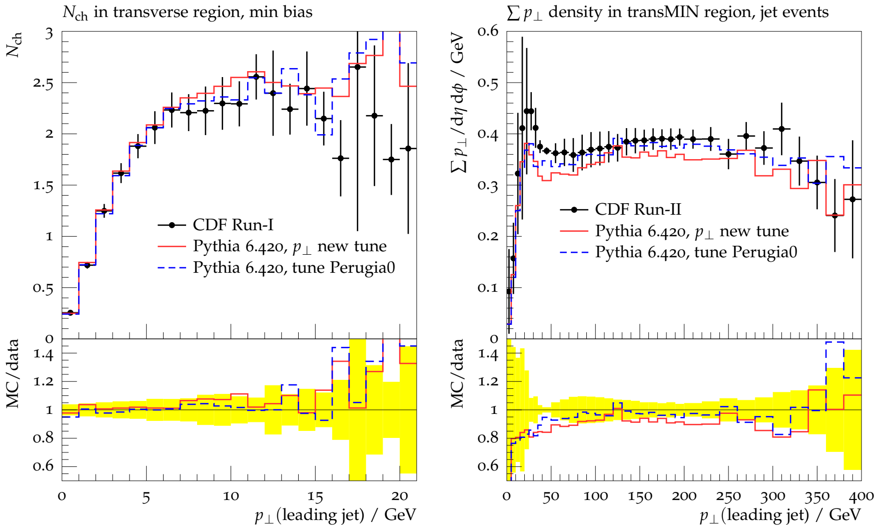


UE measurements at the Tevatron

- Z $p_\perp$ from $q\bar{q} \rightarrow$Z: α_S in ISR, primordial $k_\perp$
- Multiplicity distributions: number of particles produced
- $\langle p_\perp \rangle$ vs. N_{ch}: number and $p_\perp$ of particles produced
- Exploiting the event topology - $p_\perp^{\text{sum}}$, N_{ch} vs. $p_{\perp,\text{leading jet}}$ in jet events: almost everything



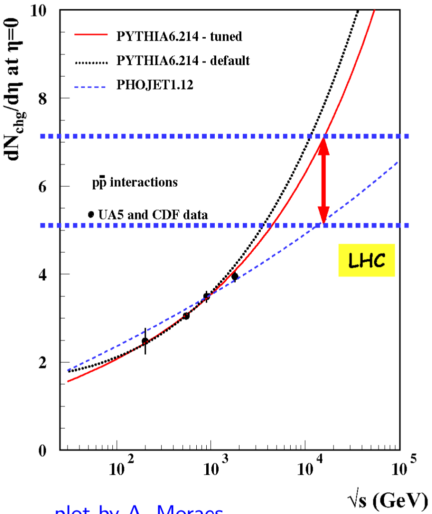




N_{ch} in transverse region, min bias
N_{ch}
MC/data
CDF Run-I
Pythia 6.420, $p_\perp$ new tune
Pythia 6.420, tune Perugia0
$p_\perp$(leading jet) / GeV
$\sum p_\perp$ density in transMIN region, jet events
$\sum p_\perp / d\eta\, d\phi$ / GeV
CDF Run-II
Pythia 6.420, $p_\perp$ new tune
Pythia 6.420, tune Perugia0
$p_\perp$(leading jet) / GeV

Extrapolations to the LHC

- Drastically different predictions for LHC
- Different UE energy-scaling:
  Phojet $\sim \ln s$
  Pythia $\sim \ln^2 s$
- Generators were tuned to data at different $\sqrt{s}$
- $\rightarrow$ Will need retuning of UE-parameters to LHC data



plot by A. Moraes

Outline

1. Underlying Event (UE)
2. Generator tuning with Professor
3. Tuning to early LHC data

A tool for systematic generator tuning

DELPHI 1995, Hamacher et al.: bin-wise interpolation of MC generator response and χ^2 minimization

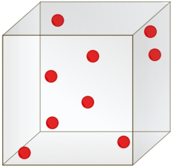
Professor (arXiv:0907.2973, arXiv:0906.0075, arXiv:0902.4403)

"PROcedure For EStimating Systematic errORrs"

- Pick up DELPHI idea
- Use flexible `python` interface
- Use quadratic or cubic interpolations
- Respond to new (LHC) data quickly
- Validation of results possible in many ways

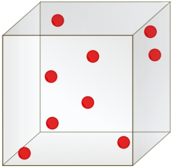
Tuning procedure in Professor (1D, 1Bin)

1. Random sampling: N parameter points in n-dimensional space



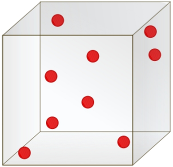
Tuning procedure in Professor (1D, 1Bin)

1. Random sampling: N parameter points in n-dimensional space



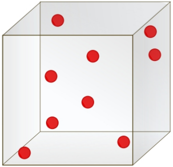
Tuning procedure in Professor (1D, 1Bin)

1. Random sampling: N parameter points in n-dimensional space



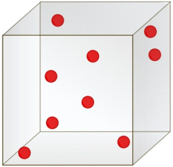
Tuning procedure in Professor (1D, 1Bin)

1. Random sampling: N parameter points in n-dimensional space



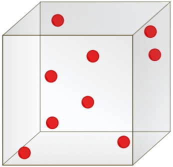
Tuning procedure in Professor (1D, 1Bin)

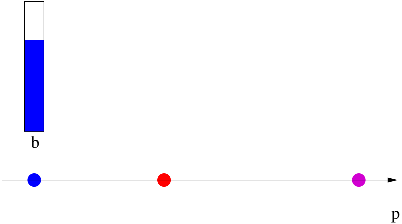
1. Random sampling: N parameter points in n-dimensional space



Tuning procedure in Professor (1D, 1Bin)

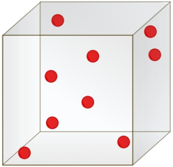
1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms

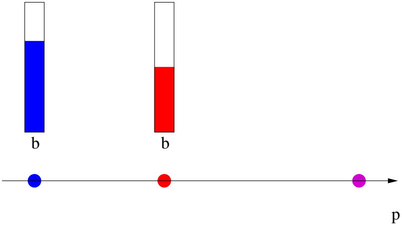




Tuning procedure in Professor (1D, 1Bin)

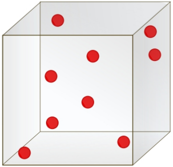
1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms

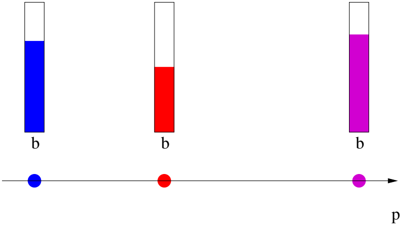




Tuning procedure in Professor (1D, 1Bin)

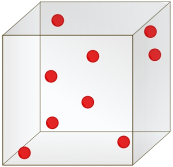
1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms

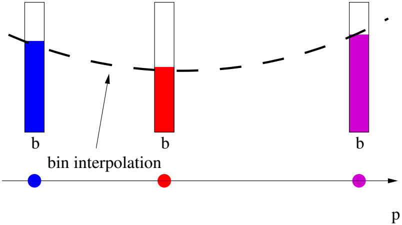




Tuning procedure in Professor (1D, 1Bin)

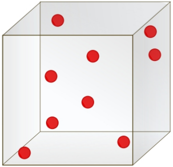
1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms
3. For each bin: use N points to fit interpolation (2nd or 3rd order polynomial)

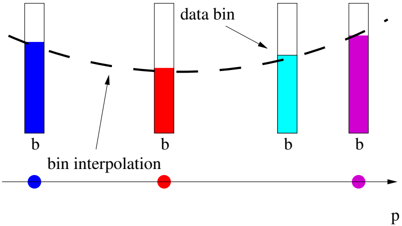




Tuning procedure in Professor (1D, 1Bin)

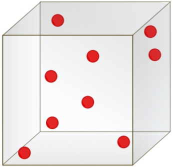
1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms
3. For each bin: use N points to fit interpolation (2^{nd} or 3^{rd} order polynomial)
4. Construct overall (now trivial) $\chi^2 = \sum_{bins} \frac{(interpolation - data)^2}{error^2}$

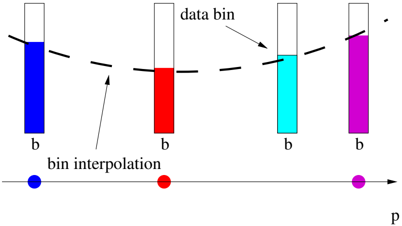




Tuning procedure in Professor (1D, 1Bin)

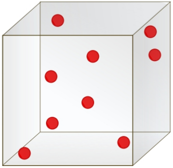
1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms
3. For each bin: use N points to fit interpolation (2nd or 3rd order polynomial)
4. Construct overall (now trivial) $\chi^2 = \sum_{bins} \frac{(interpolation - data)^2}{error^2}$
5. Numerically minimize using `pyMinuit`, `SciPy`

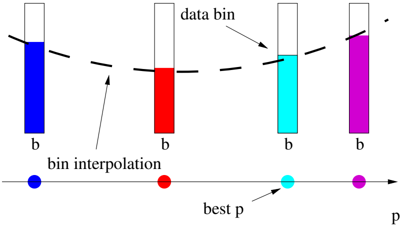




Tuning procedure in Professor (1D, 1Bin)

1. Random sampling: N parameter points in n-dimensional space
2. Run generator and fill histograms
3. For each bin: use N points to fit interpolation (2nd or 3rd order polynomial)
4. Construct overall (now trivial) $\chi^2 = \sum_{bins} \frac{(interpolation - data)^2}{error^2}$
5. Numerically minimize using `pyMinuit`, `SciPy`





Professor tunings

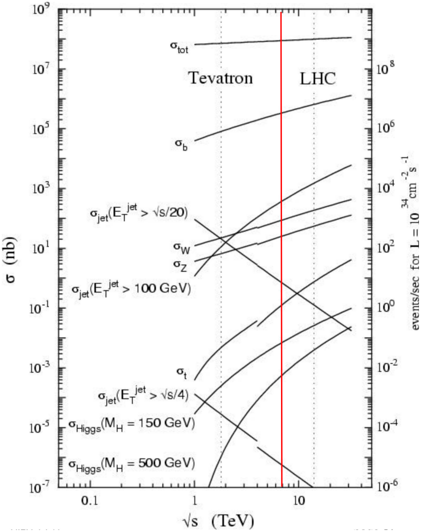
- Pythia6: simultaneous tuning of
  - 9 flavour parameters to LEP-data
  - 6 fragmentation parameters to LEP-data
  - 9 Underlying Event parameters to Tevatron-data
  - Repeated for several pdfs
- Pythia8
  - Repeated Pythia6-tune for flavour & fragmentation (new default)
  - Underlying Event broken/unusable in Pythia8
- Sherpa
  - Currently, shower (AHADIC) is being tuned to LEP-data

Outline

1. Underlying Event (UE)
2. Generator tuning with Professor
3. Tuning to early LHC data

Disadvantages of first LHC data

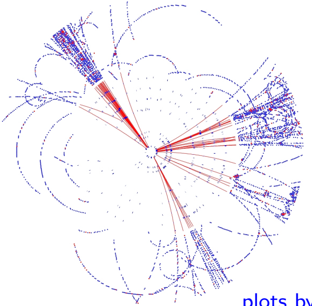
- Jet-energy calibration not very precise in the beginning
- $\rightarrow$ rather use tracks and lepton-ID
- Cross-section at $\sqrt{s} = 7$ TeV smaller than at 10 or 14 TeV
- Expect integrated luminosity of $\mathcal{O}(100\ \mathrm{pb}^{-1})$



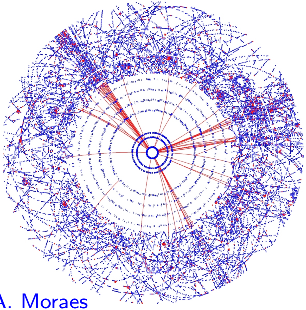
Advantages of first LHC data

- Measurements at $\sqrt{s} = 7$ TeV give another energy point for extrapolations to 10, 14 TeV
- Lower luminosity means reduced pile-up, e.g. $H \rightarrow b\bar{b}$:

$\mathcal{L} = 10^{33}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$



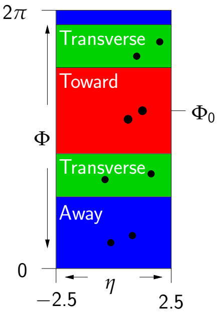
$\mathcal{L} = 10^{34}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$

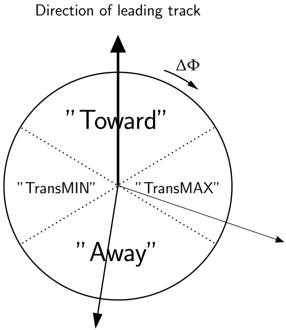


plots by A. Moraes

Leading track

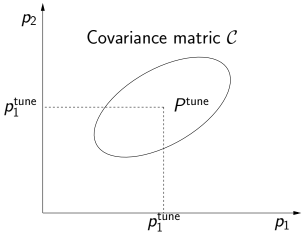
- Measure track-$p_\perp$ using only inner detector
- Identify leading track = largest $p_\perp$ in event $\rightarrow$ defines ϕ_0
- Define "transverse" region, measure N_{tracks}, scalar $p_\perp$-sum as function of $p_{\perp\text{, leading track}}$





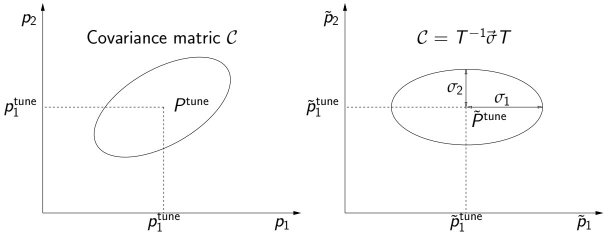
Tuning uncertainties

- get cov. matrix



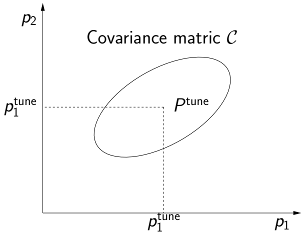
Tuning uncertainties

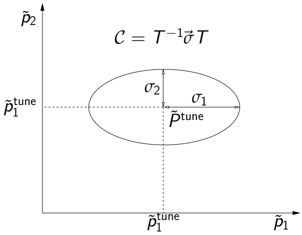
- get cov. matrix
- Eigendecomp.

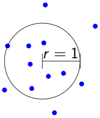


Tuning uncertainties

- get cov. matrix
- Eigendecomp.
- Sample from $\mathcal{G}^N$

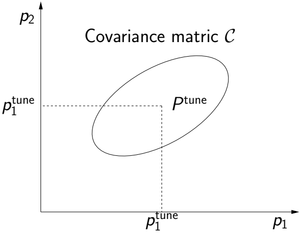


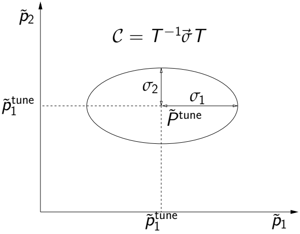


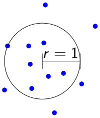


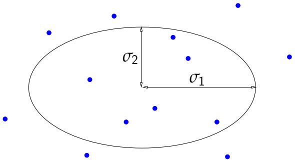
Tuning uncertainties

- get cov. matrix
- Eigendecomp.
- Sample from $\mathcal{G}^N$
- Stretch by σ_i



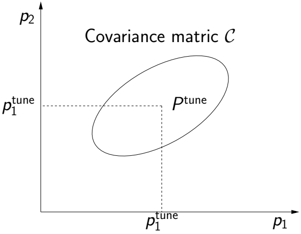


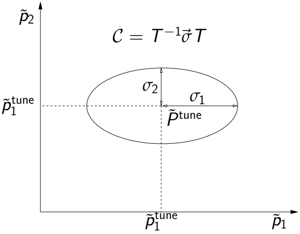


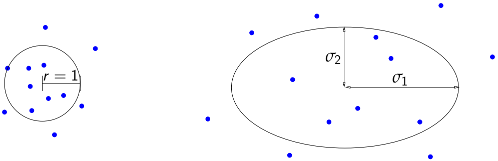


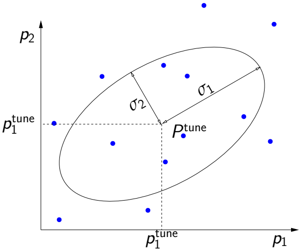
Tuning uncertainties

- get cov. matrix
- Eigendecomp.
- Sample from $\mathcal{G}^N$
- Stretch by σ_i
- Rotate back with T



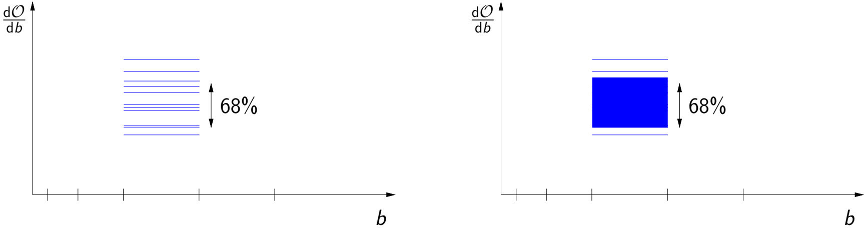






Confidence belt construction

1. Estimate tunig uncertainties: use points sampled from ellipse
2. Run generator or use parameterisation to get bin-content prediction
3. For each bin b and each observable $\mathcal{O}$: determine central 68, 95 pct.

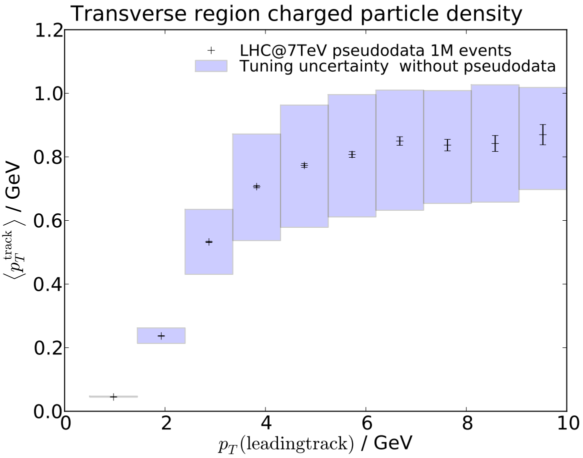


$\rightarrow$ Use confidence belts for early data sensitivity studies, i.e., if we add LHC (pseudo-) data to the existing tune, does the confidence belt shrink?

If so, consider corresponding measurement worthwile for early data.

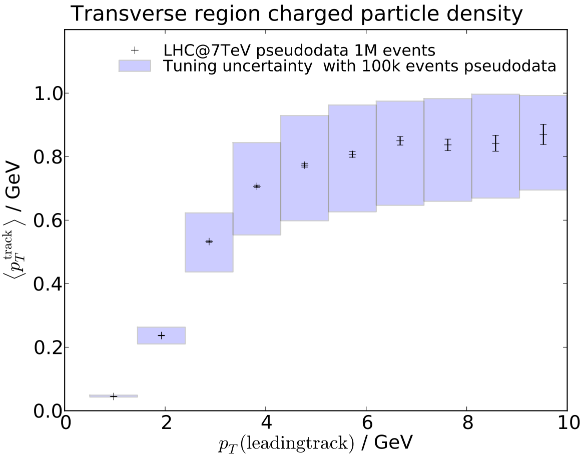
Leading track sensitivity study (generator level)

Pythia6 (tune 329) prediction for the LHC ($\sqrt{s}$ =7 TeV)



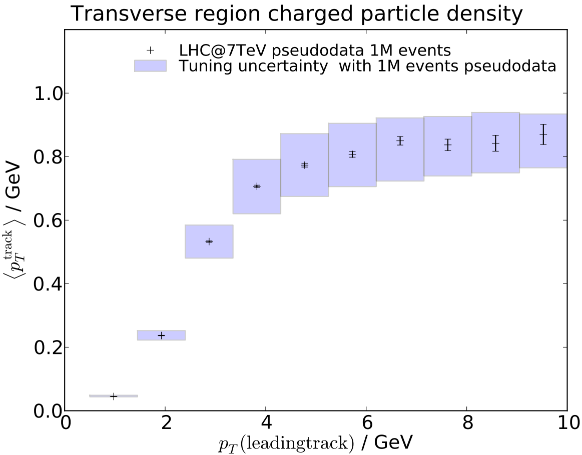
Leading track sensitivity study (generator level)

Pythia6 (tune 329) + 100k events of pseudo-data



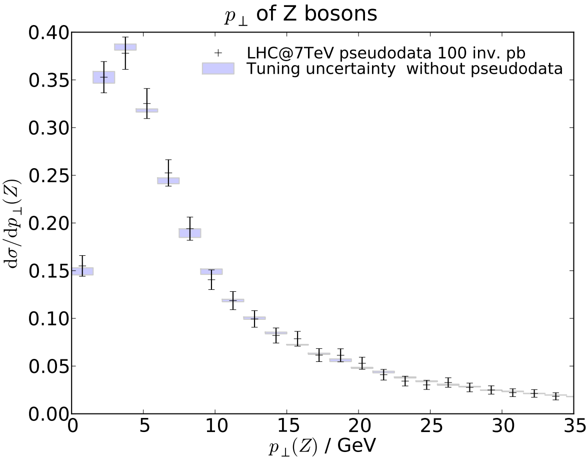
Leading track sensitivity study (generator level)

Pythia6 (tune 329) + 1M events of pseudo-data



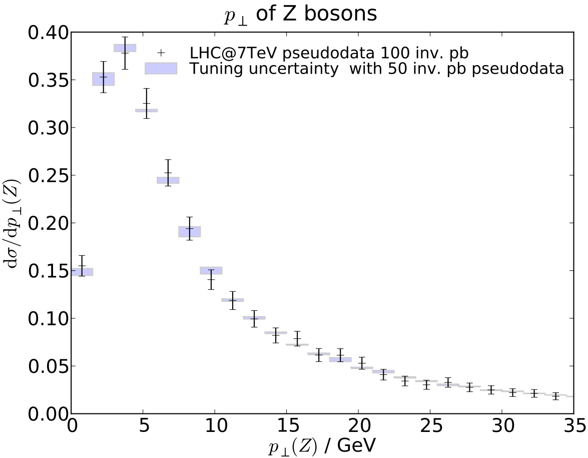
Z-$p_\perp$ SENSITIVITY STUDY

Pythia6 (tune 329) prediction for the LHC ($\sqrt{s}$ =7 TeV)



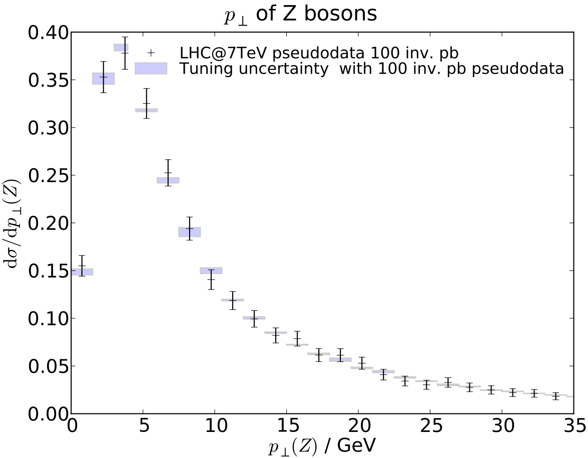
Z-$p_\perp$ SENSITIVITY STUDY

Pythia6 (tune 329) + 50 pb^{-1} of pseudo-data



Z-$p_\perp$ SENSITIVITY STUDY

Pythia6 (tune 329) + 100 pb^{-1} of pseudo-data



Summary and Outlook

- UE measurements at LHC essential for understanding high $p_\perp$ physics
- UE measurements important for generator re-tuning
- Professor excellent tool for quick turn-around tunings, sensitivity estimation

---

- UE as function of leading track $p_\perp$ looks promising
- Z-boson $p_\perp$ important cross-check (as well as 900 GeV LHC data)
- Investigate also sensitivity to UE activity as function of Z-$p_\perp$, W-$p_\perp$
- Can quantify tuning uncertainty, include systematics of Professor method soon

For Further Reading

- A. Buckley et al., 2009

  Systematic event generator tuning for the LHC

  arXiv:0907.2973, accepted for publication in EPJC

- A. Buckley et al., 2008

  Monte Carlo tuning and generator validation

  arXiv:0906.0075

- A. Buckley et al., 2008

  Monte Carlo event generator validation and tuning for the LHC

  arxiv:0902.4403

- Website of the Professor project (plots, howtos, exercises, ...)

  `http://projects.hepforge.org/professor`

  If you are interested, please join the fun!

Backup

Professor tunes in Pythia 6:

```
6.4.20 : 20 February 2009
  - Comprehensive updates to PYTUNE, with the addition of the
    "Perugia" and "Pro" tunes, following the MPI workshop in Perugia
    in October 2008. The older tunes remain unaltered. The new
    available tunes in PYTUNE are:
    ===== Old UE, Q2-ordered showers ====================================
    --- Professor Tunes : 110+ (= 100+ with Professor's tune to LEP) ----
     110   A-Pro : Tune A, with LEP tune from Professor        (Oct 2008)
     111  AW-Pro : Tune AW, -"-                                (Oct 2008)
     112  BW-Pro : Tune BW, -"-                                (Oct 2008)
     113  DW-Pro : Tune DW, -"-                                (Oct 2008)
     114 DWT-Pro : Tune DWT, -"-                               (Oct 2008)
     115  QW-Pro : Tune QW, -"-                                (Oct 2008)
     116 ATLAS-DC2-Pro: ATLAS-DC2 / Rome, -"-                  (Oct 2008)
     117 ACR-Pro : Tune ACR, -"-                               (Oct 2008)
     118  D6-Pro : Tune D6, -"-                                (Oct 2008)
     119 D6T-Pro : Tune D6T, -"-                               (Oct 2008)
    --- Professor's Q2-ordered Perugia Tune : 129 -----------------------
     129 Pro-Q20 : Professor Q2-ordered tune                   (Feb 2009)
    ===== Intermediate and Hybrid Models ================================
     211 APT-Pro : Tune APT, with LEP tune from Professor      (Oct 2008)
    . . .
     ===== New UE, interleaved pT-ordered showers, annealing CR ==========
    --- Professor Tunes : 310+ (= 300+ with Professor's tune to LEP)
     310   S0-Pro : S0 with updated LEP pars from Professor    (Oct 2008)
     311   S1-Pro : S1 -"-                                     (Oct 2008)
     312   S2-Pro : S2 -"-                                     (Oct 2008)
     313  S0A-Pro : S0A -"-                                    (Oct 2008)
     314 NOCR-Pro : NOCR -"-                                   (Oct 2008)
     315  Old-Pro : Old -"-                                    (Oct 2008)
     . . .
    --- Professor's pT-ordered Perugia Tune : 329 -----------------------
     329 Pro-pT0   : Professor pT-ordered tune w. S0 CR model  (Feb 2009)
    =====================================================================
```

2nd order polynomial includes lowest-order correlations between parameters

$$MC_b(\vec{p}\,) \approx f^{(b)}(\vec{p}\,) = \alpha_0^{(b)} + \sum_i \beta_i^{(b)}\, p'_i + \sum_{i\leq j} \gamma_{ij}^{(b)}\, p'_i\, p'_j$$

Now use N generator runs, i.e. N different parameter sets x,y:

$$\underbrace{\begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{pmatrix}}_{\vec{v}\text{ (N values, i.e. N bin contents)}} = \underbrace{\begin{pmatrix} 1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 \\ 1 & x_2 & y_2 & x_2^2 & x_2 y_2 & y_2^2 \\ & & & \vdots & & \\ 1 & x_N & y_N & x_N^2 & x_N y_N & y_N^2 \end{pmatrix}}_{\tilde{\mathbf{P}}\text{ (N sampled parameter sets)}} \underbrace{\begin{pmatrix} \alpha_0 \\ \beta_x \\ \beta_y \\ \gamma_{xx} \\ \gamma_{xy} \\ \gamma_{yy} \end{pmatrix}}_{\vec{c}\text{ (coeffs)}}$$

Therefore: $\boxed{\vec{c}_b = \tilde{\mathcal{I}}[\tilde{\mathbf{P}}]\vec{v}}$ where $\tilde{\mathcal{I}}$ is the pseudoinverse operator.

$$\boxed{\vec{c}_b = \tilde{\mathcal{I}}[\tilde{\mathbf{P}}]\vec{v}}$$

- Use Singular Value Decomposition (SVD), a general diagonalisation for all normal matrices M: $M = U\Sigma V^*$
- Method available in SciPy.linalg
- Minimal number of runs = number of coefficients in $\vec{c}_b$: $N_{\text{min}}^{(n)} = 1 + n + n(n+1)/2$

$$\boxed{\vec{c}_b = \tilde{\mathcal{I}}[\tilde{\mathbf{P}}]\vec{v}}$$

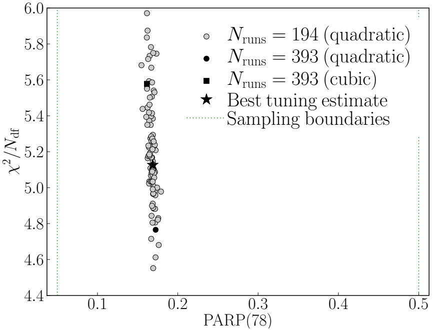
- Use Singular Value Decomposition (SVD), a general diagonalisation for all normal matrices M:$M = U\Sigma V^*$
- Method available in SciPy.linalg
- Minimal number of runs = number of coefficients in $\vec{c}_b$:
  $N_{\text{min}}^{(n)} = 1 + n + n(n+1)/2 + \underbrace{(n+1)(n+2)/6}_{\text{cubic only}}$

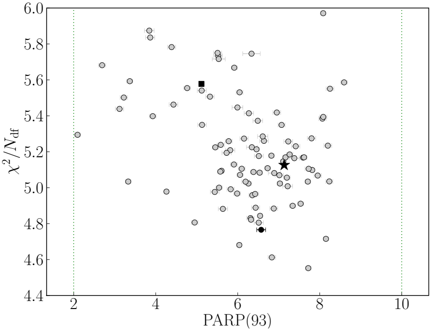
$$\boxed{\vec{c}_b = \tilde{\mathcal{I}}[\tilde{\mathbf{P}}]\vec{v}}$$

- Use Singular Value Decomposition (SVD), a general diagonalisation for all normal matrices M:$M = U\Sigma V^*$
- Method available in SciPy.linalg
- Minimal number of runs = number of coefficients in $\vec{c}_b$: $N_{\text{min}}^{(n)} = 1 + n + n(n+1)/2 + \underbrace{(n+1)(n+2)/6}_{\text{cubic only}}$
- Oversampling by a factor of three has proven to be much better

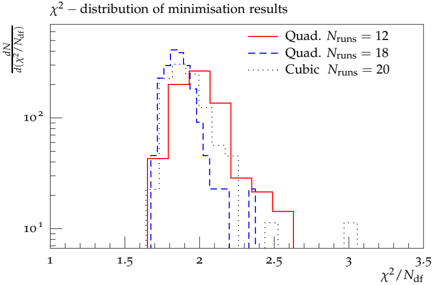
Num params, P	$N_2^{(P)}$ (2nd order)	$N_3^{(P)}$ (3rd order)
1	3	4
2	6	10
4	15	35
6	28	84
8	45	165
9	55	220

- If we have N generator runs: choose combinations of k ($k < N$) runs
- So far: $k \approx 3 \cdot N_{\min}$ and $k/N \approx 0.66$
- Each combination $\rightarrow$ different parameterisation $\rightarrow$ (slightly) different minimisation result
- Investigate spread parameter-wise

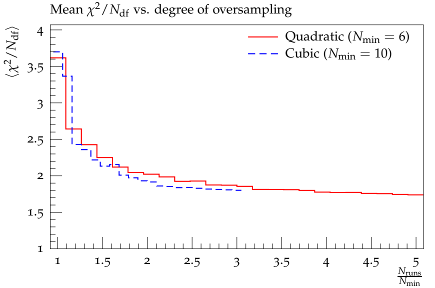


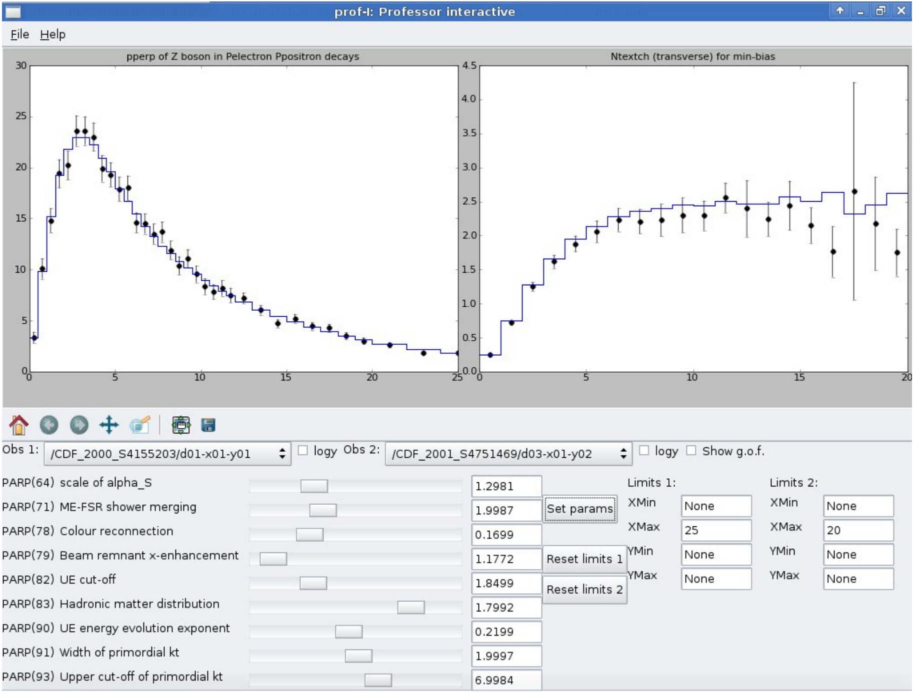


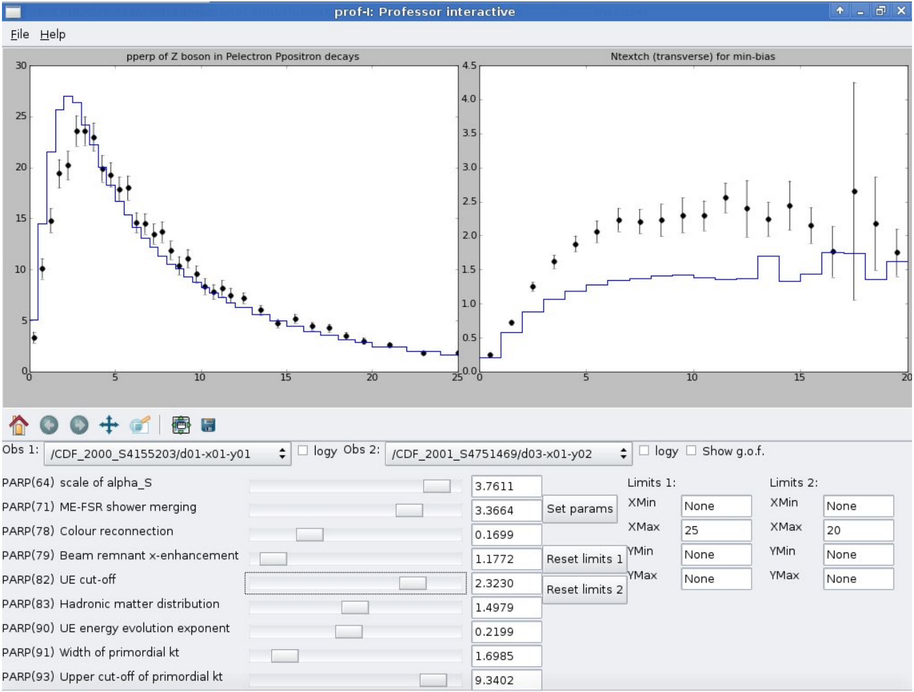
Observe lower χ^2/N_{df}-boundary:

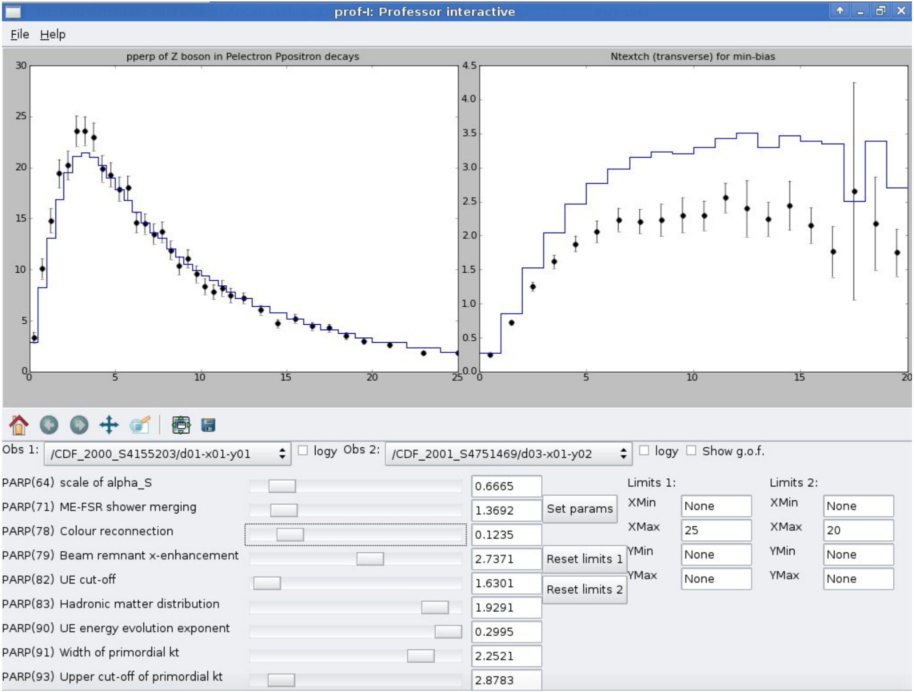


Oversampling is neccesary (at least 2 to 3 times N_{min}):




prof-I: Professor interactive
File Help
pperp of Z boson in Pelectron Ppositron decays
Ntextch (transverse) for min-bias
Obs 1: /CDF_2000_S4155203/d01-x01-y01
logy
Obs 2: /CDF_2001_S4751469/d03-x01-y02
logy
Show g.o.f.
PARP(64) scale of alpha_S
1.2981
PARP(71) ME-FSR shower merging
1.9987
PARP(78) Colour reconnection
0.1699
PARP(79) Beam remnant x-enhancement
1.1772
PARP(82) UE cut-off
1.8499
PARP(83) Hadronic matter distribution
1.7992
PARP(90) UE energy evolution exponent
0.2199
PARP(91) Width of primordial kt
1.9997
PARP(93) Upper cut-off of primordial kt
6.9984
Set params
Reset limits 1
Reset limits 2
Limits 1:
XMin None
XMax 25
YMin None
YMax None
Limits 2:
XMin None
XMax 20
YMin None
YMax None


prof-I: Professor interactive
File
Help
pperp of Z boson in Pelectron Ppositron decays
Ntextch (transverse) for min-bias
Obs 1:
/CDF_2000_S4155203/d01-x01-y01
logy
Obs 2:
/CDF_2001_S4751469/d03-x01-y02
logy
Show g.o.f.
PARP(64) scale of alpha_S
3.7611
PARP(71) ME-FSR shower merging
3.3664
Set params
PARP(78) Colour reconnection
0.1699
PARP(79) Beam remnant x-enhancement
1.1772
Reset limits 1
PARP(82) UE cut-off
2.3230
Reset limits 2
PARP(83) Hadronic matter distribution
1.4979
PARP(90) UE energy evolution exponent
0.2199
PARP(91) Width of primordial kt
1.6985
PARP(93) Upper cut-off of primordial kt
9.3402
Limits 1:
XMin None
XMax 25
YMin None
YMax None
Limits 2:
XMin None
XMax 20
YMin None
YMax None


prof-I: Professor interactive
File Help
pperp of Z boson in Pelectron Ppositron decays
Ntextch (transverse) for min-bias
Obs 1: /CDF_2000_S4155203/d01-x01-y01
logy
Obs 2: /CDF_2001_S4751469/d03-x01-y02
logy
Show g.o.f.
PARP(64) scale of alpha_S
0.6665
PARP(71) ME-FSR shower merging
1.3692
Set params
PARP(78) Colour reconnection
0.1235
PARP(79) Beam remnant x-enhancement
2.7371
Reset limits 1
PARP(82) UE cut-off
1.6301
Reset limits 2
PARP(83) Hadronic matter distribution
1.9291
PARP(90) UE energy evolution exponent
0.2995
PARP(91) Width of primordial kt
2.2521
PARP(93) Upper cut-off of primordial kt
2.8783
Limits 1:
XMin None
XMax 25
YMin None
YMax None
Limits 2:
XMin None
XMax 20
YMin None
YMax None