Fully Automated MC Tuning with Genetic Algorithms

Sami Kama

MC4LHC Readiness

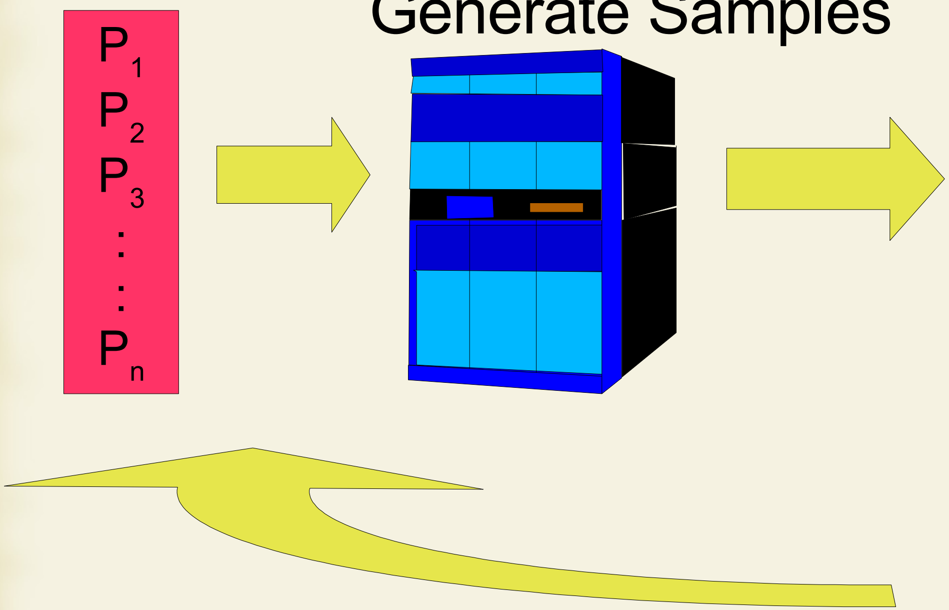
Outline

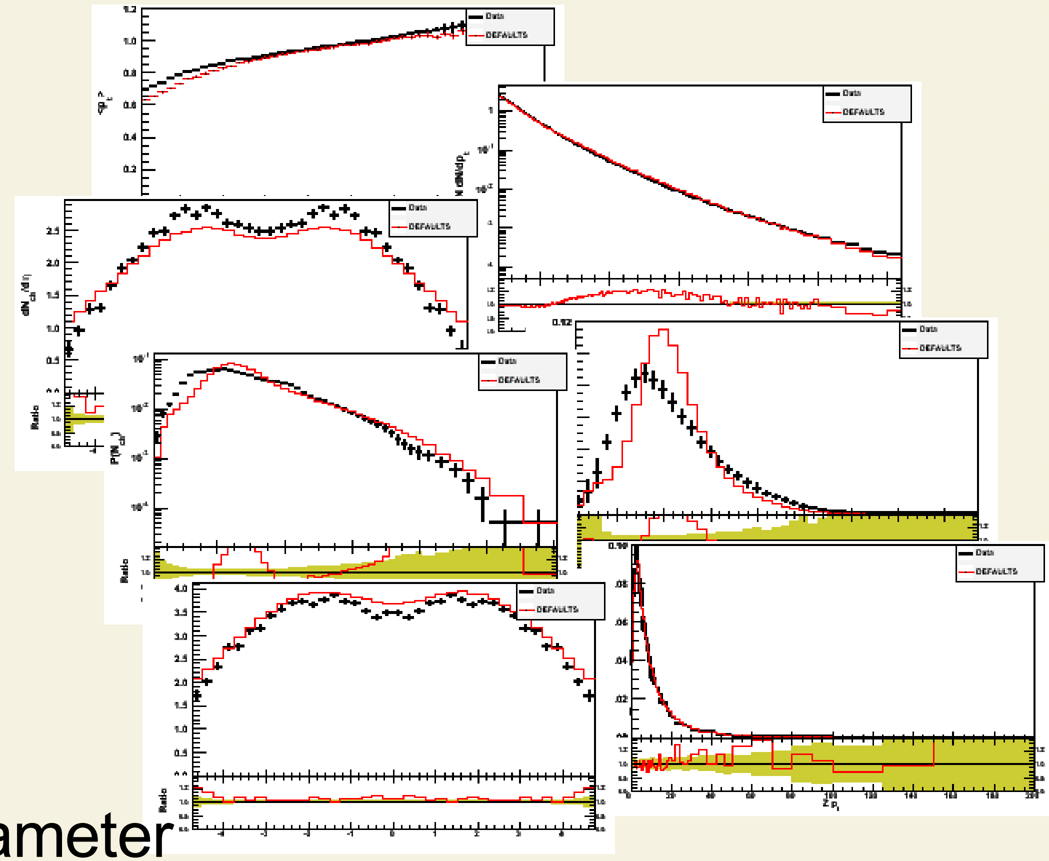
- Introduction to MC tuning
- Genetic Algorithms and their application to MC tuning
- PYTHIA tune for minimum bias events
- Current Status
- Conclusions

Introduction

Parameters

Generate Samples



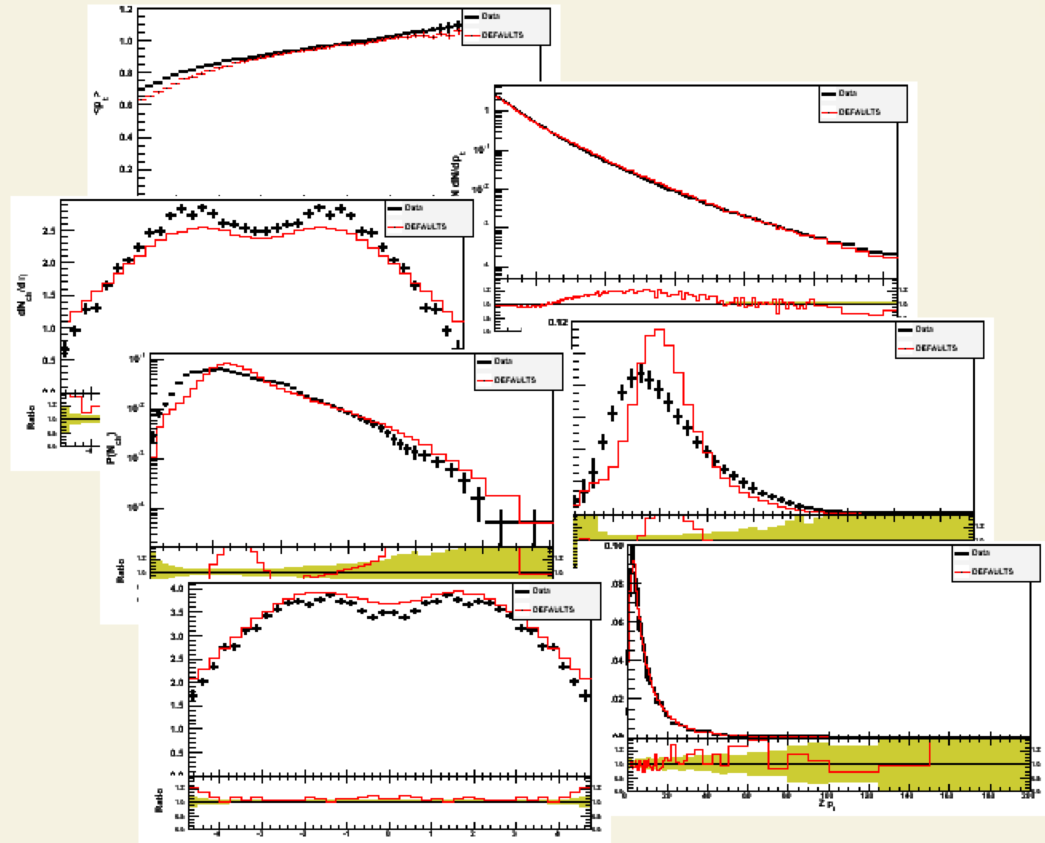


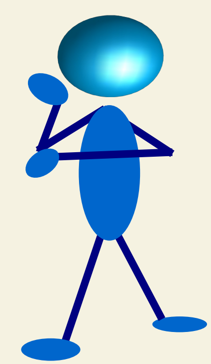
- Compare samples and data
- Select new parameters in parameter hypercube
- Repeat until satisfied with results

Ultimate goal is to have a perfect description at each bin of all data histograms

Manual Tuning

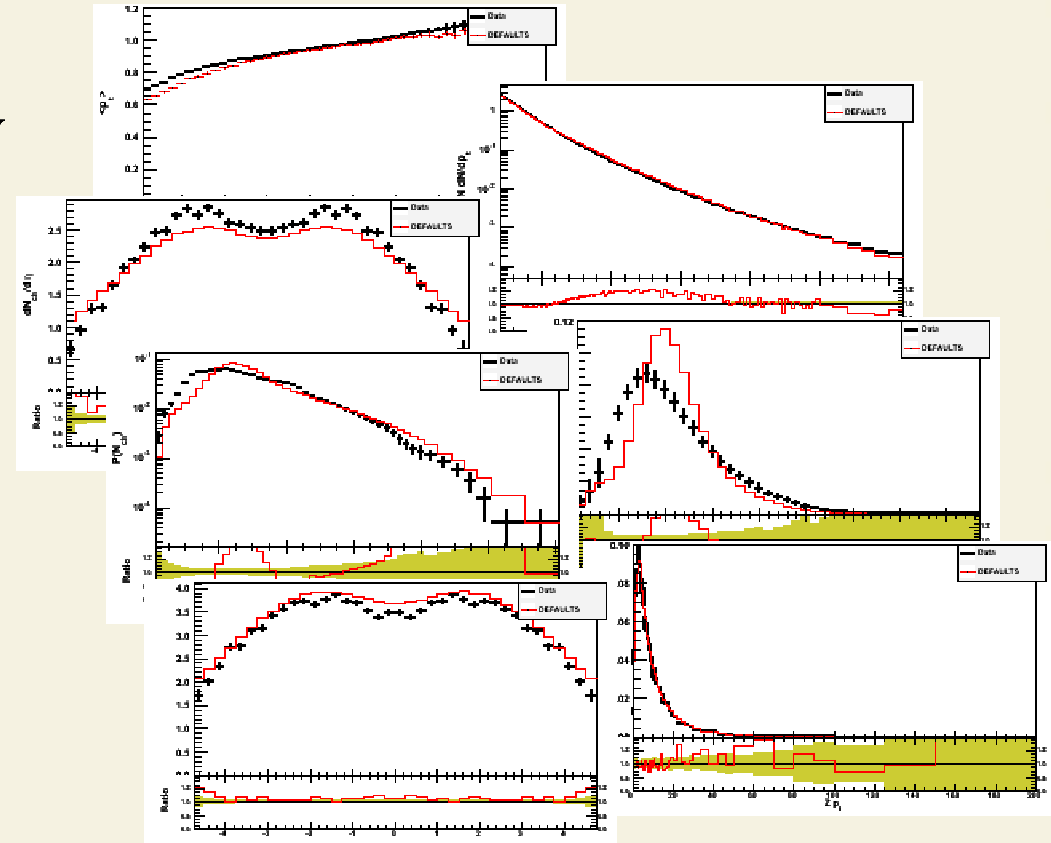
- Compare distributions by eye.

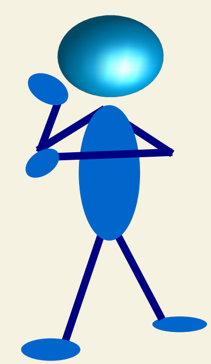




Manual Tuning

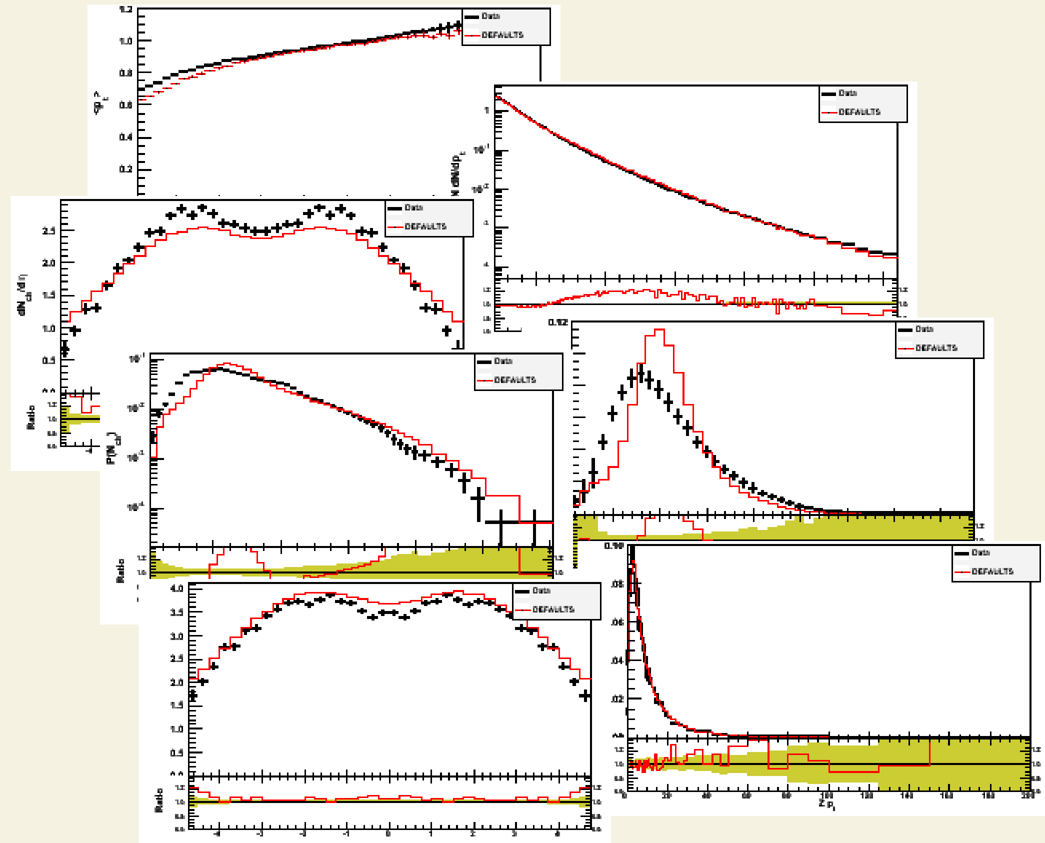
- Compare distributions by eye.
- Alter parameters by intuition and experience.

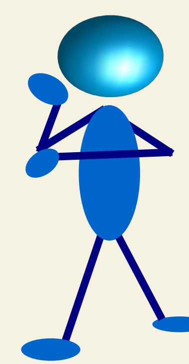




Manual Tuning

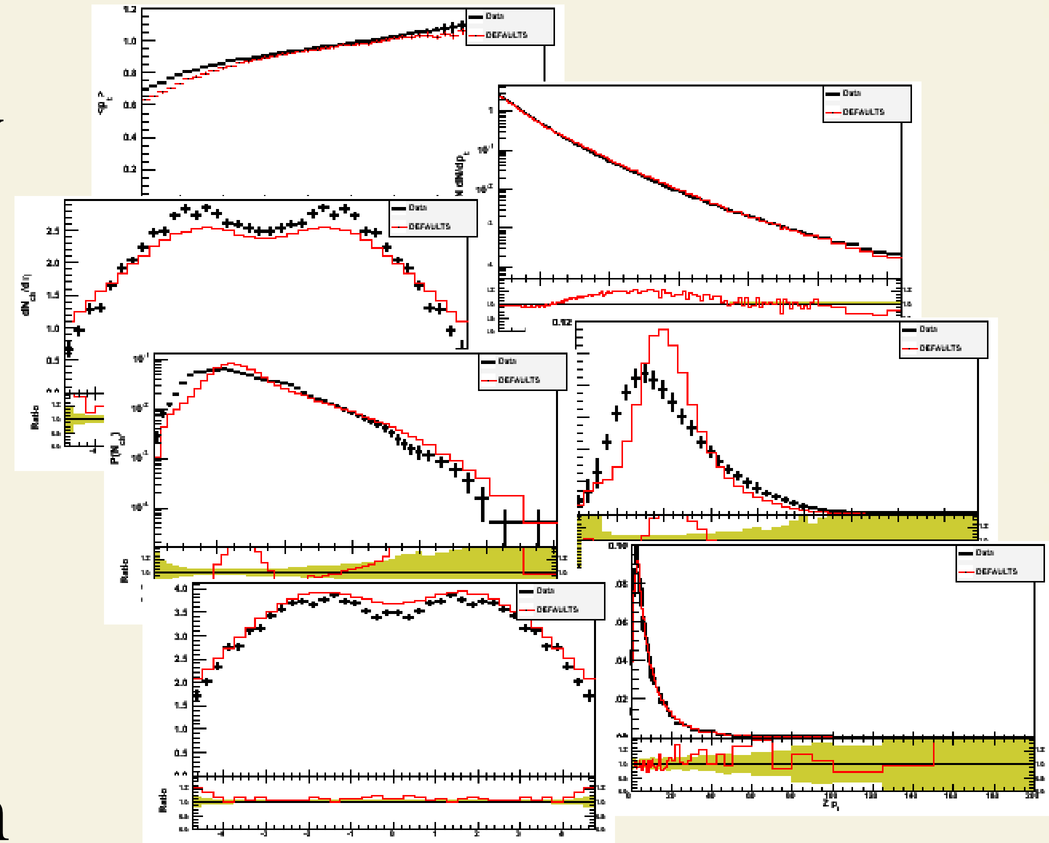
- Compare distributions by eye.
- Alter parameters by intuition and experience.
- Only real experts can produce good results.

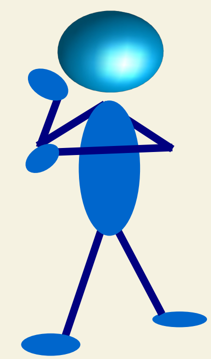




Manual Tuning

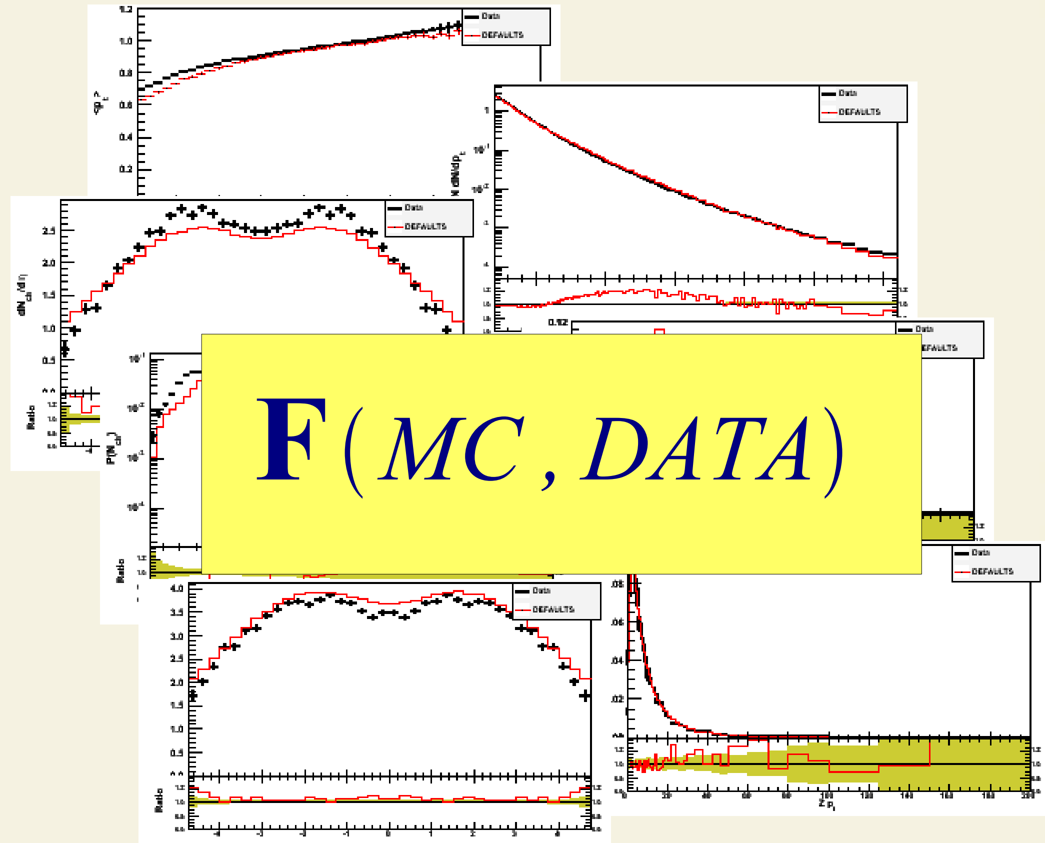
- Compare distributions by eye.
- Alter parameters by intuition and experience.
- Only real experts can produce good results.
- Experts might disagree on what is a good description.





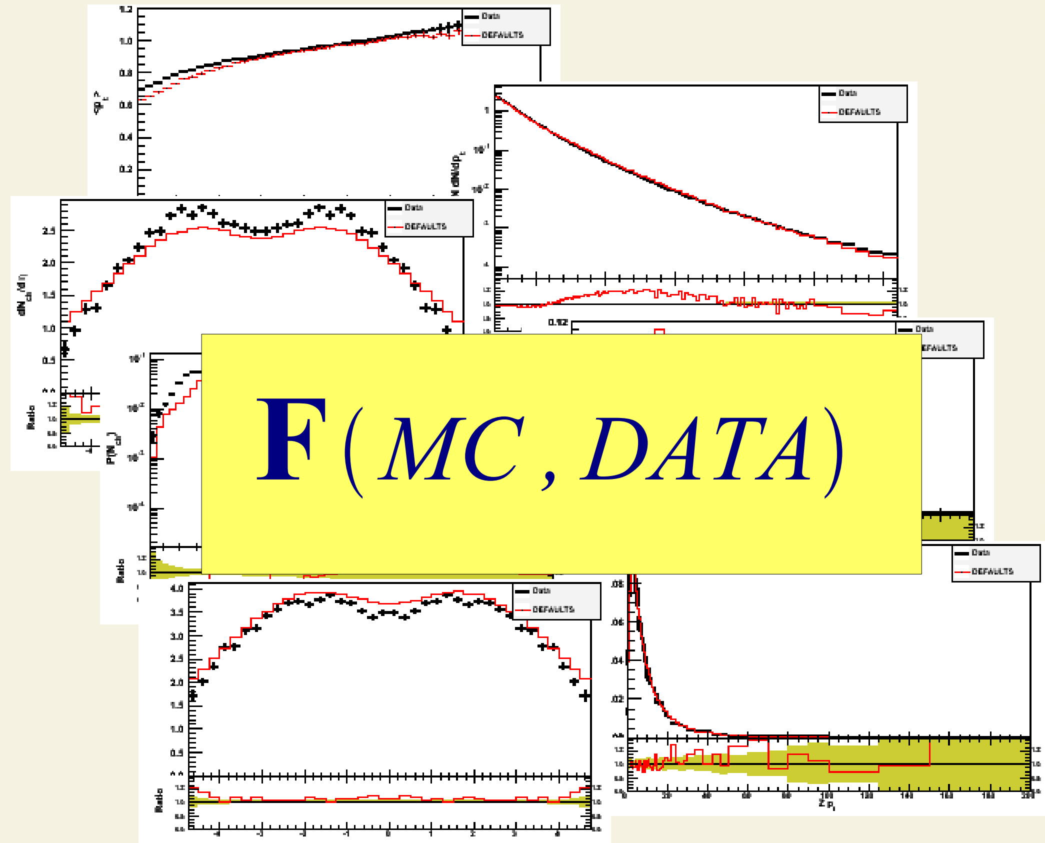
Automatic Tuning

- Define a measure (**F**) for goodness of distributions (typically chi-square).



Automatic Tuning

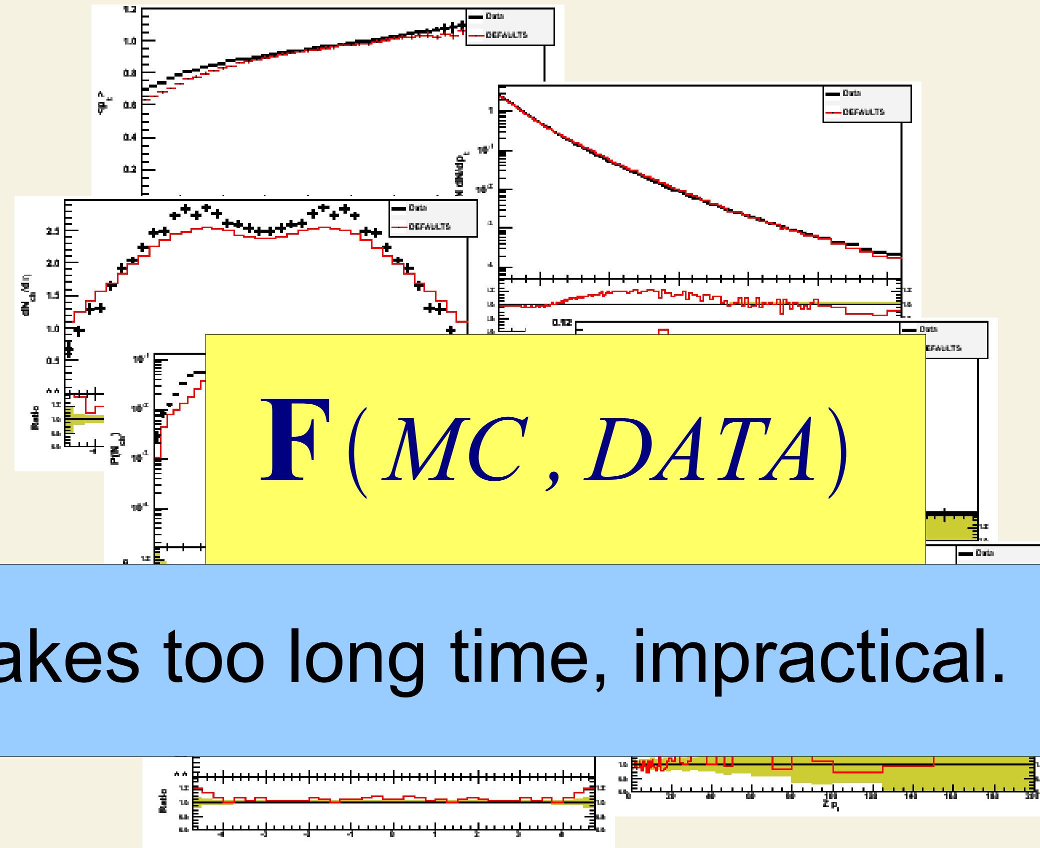
- Define a measure (**F**) for goodness of distributions (typically chi-square).
- Try to minimize **F()** by changing the parameters
  - Randomly
  - Using a minimization package



$$\mathbf{F}(MC, DATA)$$

Automatic Tuning

- Define a measure (**F**) for goodness of distributions (typically chi-square).
- Try to minimize **F()** by changing the parameters
  - Randomly
  - Using a minimization package

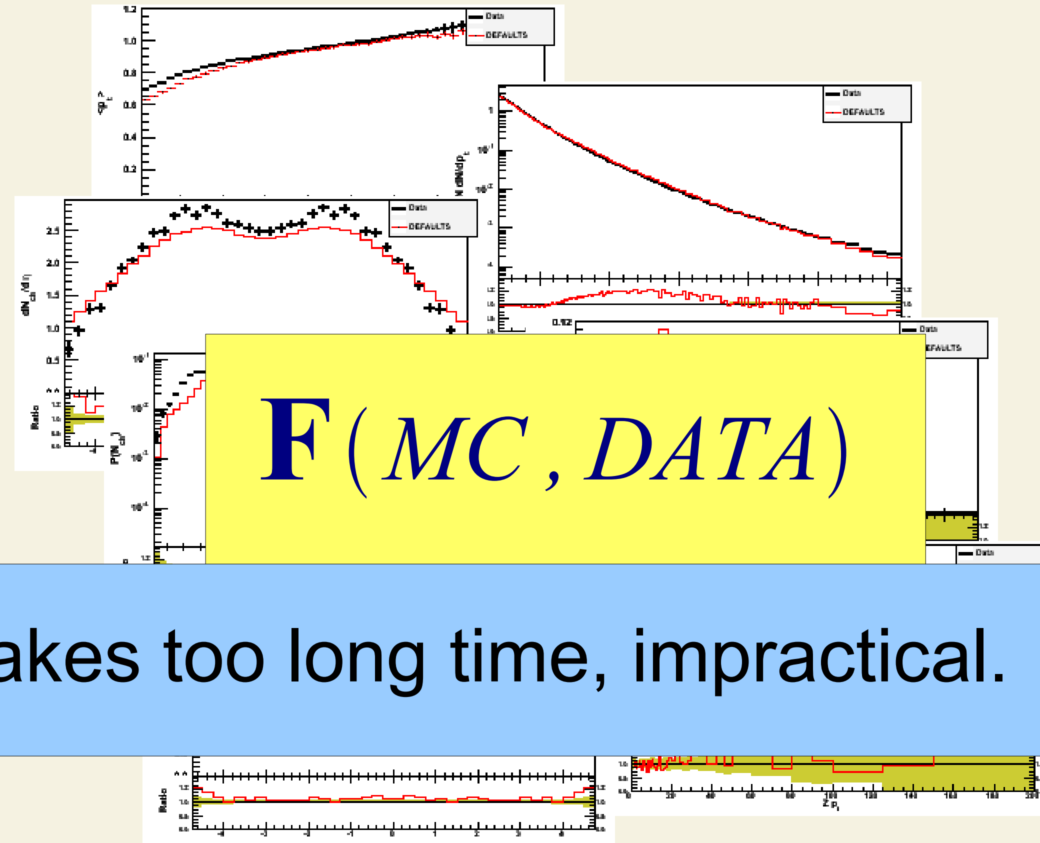


$$\mathbf{F}(MC, DATA)$$

Takes too long time, impractical.

Automatic Tuning

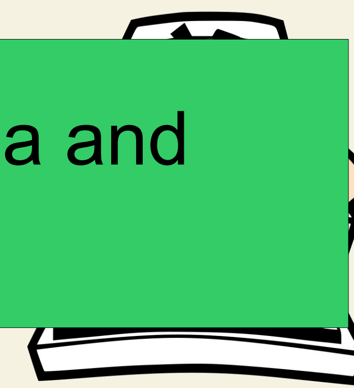
- Define a measure (**F**) for goodness of distributions (typically chi-square).
- Try to minimize **F()** by changing the parameters
  - Randomly
  - Using a minimization package



$$\mathbf{F}(MC, DATA)$$

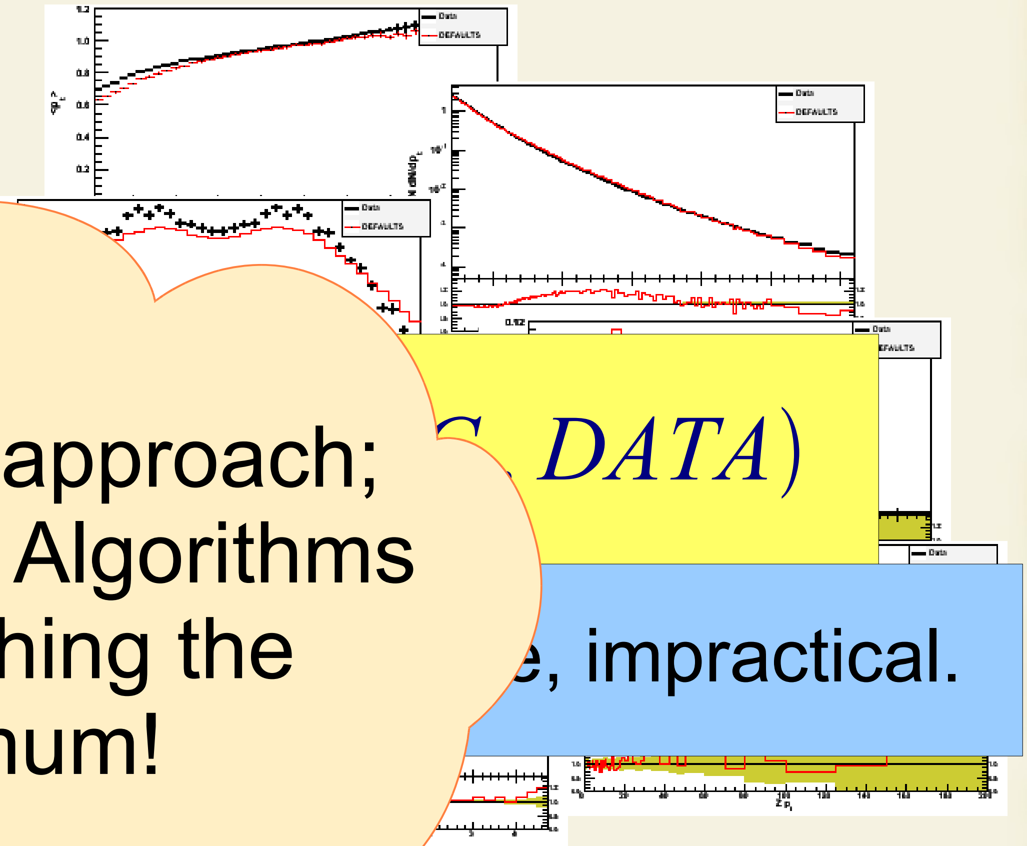
Takes too long time, impractical.

Prone to local minima and takes a long time.



Automatic Tuning

- Define a measure (**F**) for goodness of distributio
  (typically chi-s
- Try to minin
  changing th
  - Rando
  - Using a
    package

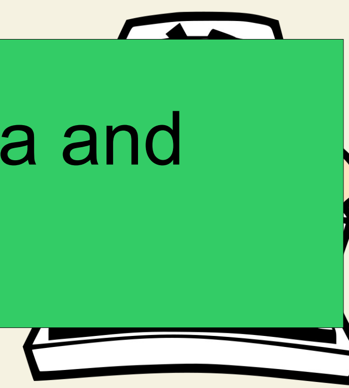


C, DATA)

e, impractical.

Alternative approach; use Genetic Algorithms for searching the minimum!

Prone to local minima and takes a long time.



Genetic Algorithms

Genetic Algorithms

- Genetic Algorithms are based on the evolutionary principles observed in nature.

Genetic Algorithms

- Genetic Algorithms are based on the evolutionary principles observed in nature.
- Potential solution candidates, called "individuals", are tested for *fitness*.

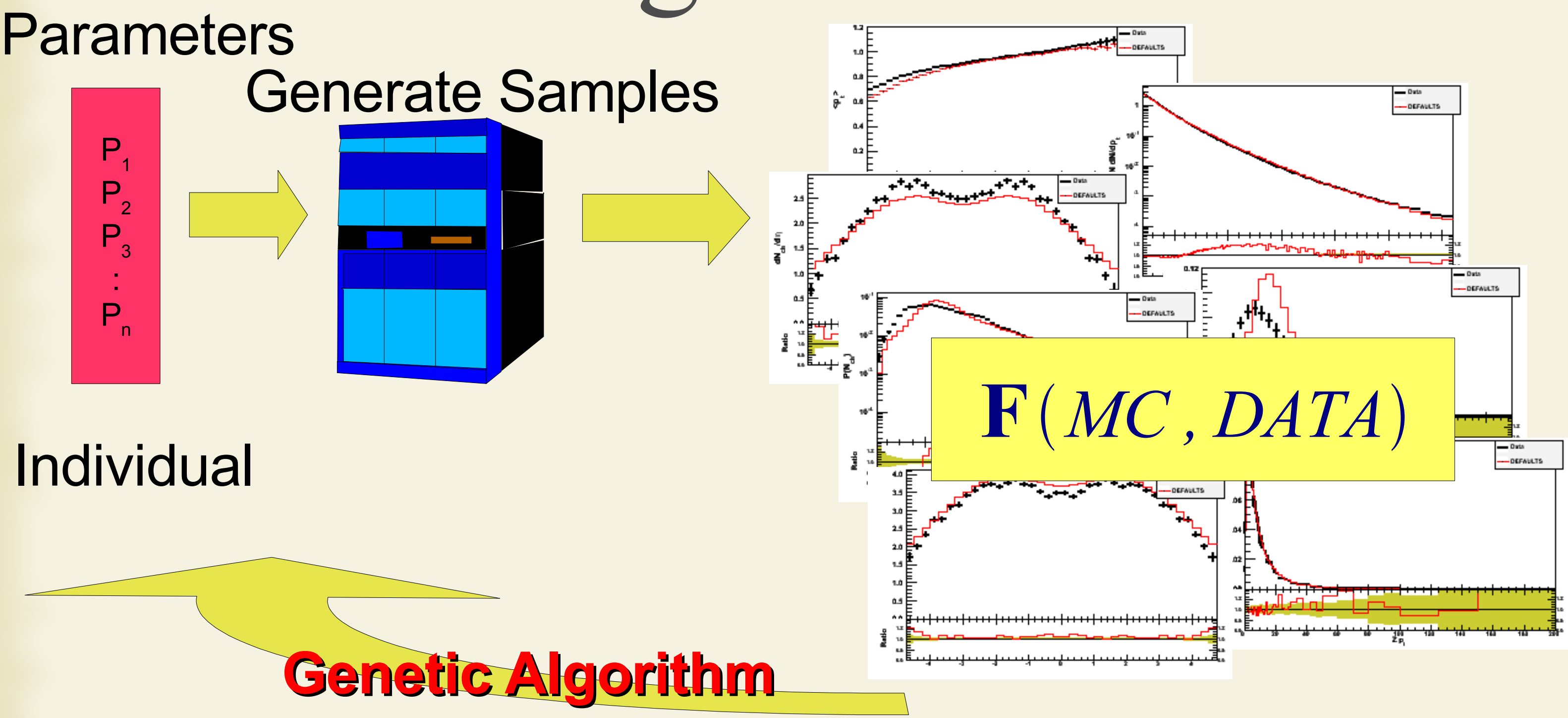
Genetic Algorithms

- Genetic Algorithms are based on the evolutionary principles observed in nature.
- Potential solution candidates, called "individuals", are tested for *fitness*.
- Fitter individuals procreate more, creating children containing their genes for the next generation.

Genetic Algorithms

- Genetic Algorithms are based on the evolutionary principles observed in nature.
- Potential solution candidates, called "individuals", are tested for *fitness*.
- Fitter individuals procreate more, creating children containing their genes for the next generation.
- Genetic Algorithms are very good at search and optimization problems.

Using GA for MC tuning



Fitness is determined by one of three different functions

$$\mathbf{F} = \sum_{i=FirstBin}^{LastBin} \frac{(Data_i - MC_i)^2}{Data_i^2}$$

Sensitive to tails

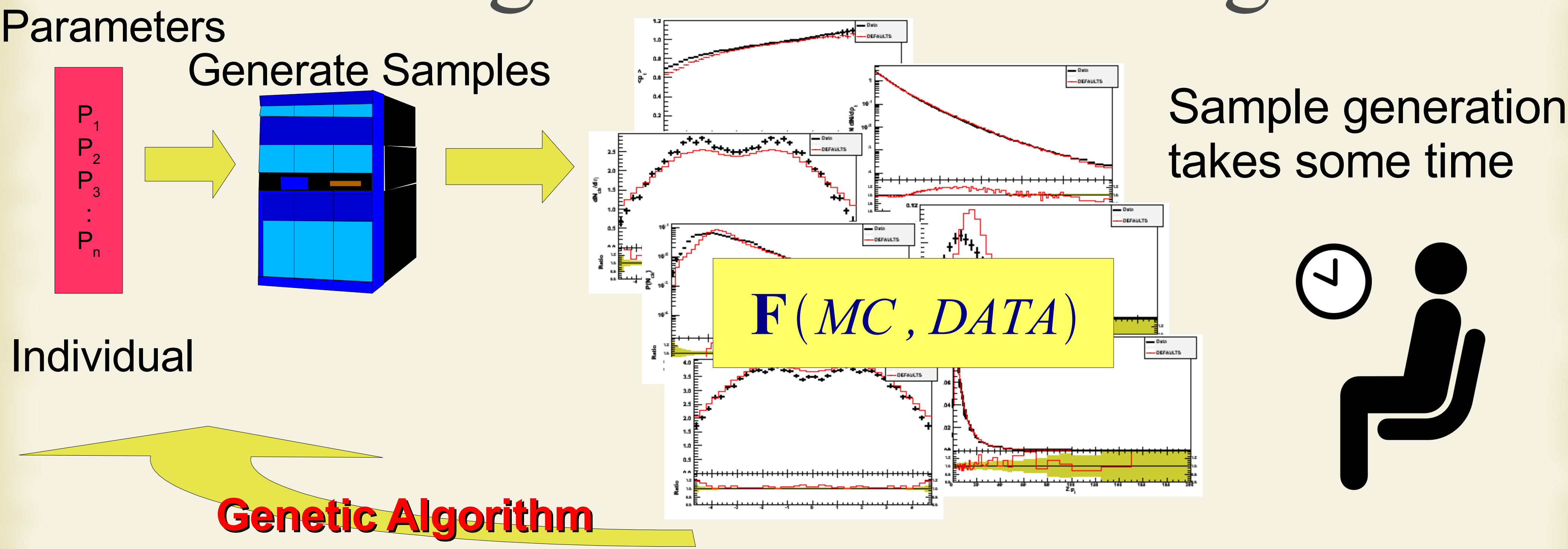
$$\mathbf{F} = \sum_{i=FirstBin}^{LastBin} \frac{(Data_i - MC_i)^2}{(\langle Data \rangle \times Data_i)}$$

Compromise between peaks and tails

$$\mathbf{F} = \sum_{i=FirstBin}^{LastBin} \frac{(Data_i - MC_i)^2}{\sigma_{Data}^2}$$

χ^2 *Function*

Using GA for MC tuning



Fitness is determined by one of three different functions

$$\mathbf{F} = \sum_{i=FirstBin}^{LastBin} \frac{(Data_i - MC_i)^2}{Data_i^2}$$

Sensitive to tails

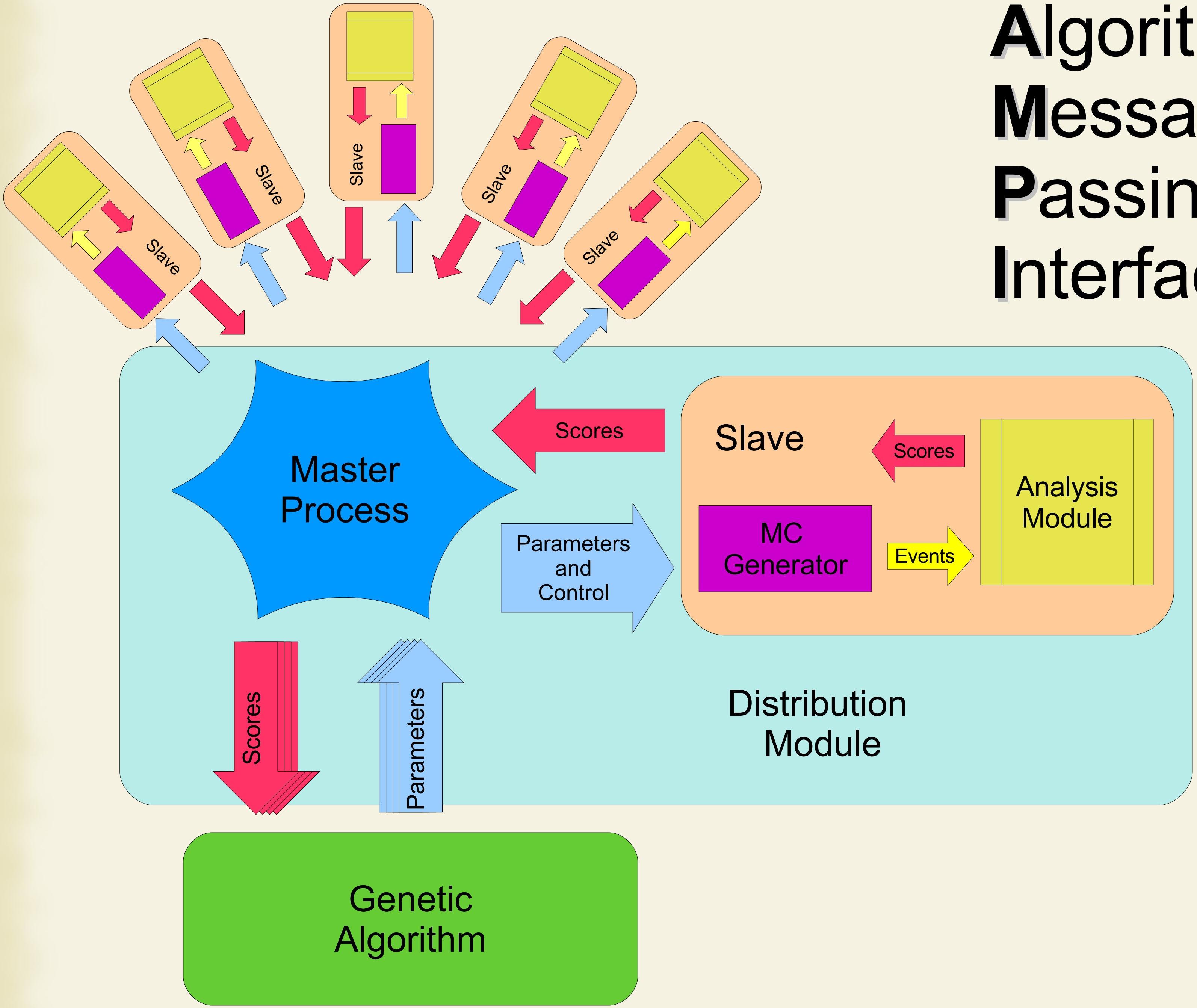
$$\mathbf{F} = \sum_{i=FirstBin}^{LastBin} \frac{(Data_i - MC_i)^2}{(\langle Data \rangle \times Data_i)}$$

Compromise between peaks and tails

$$\mathbf{F} = \sum_{i=FirstBin}^{LastBin} \frac{(Data_i - MC_i)^2}{\sigma_{Data}^2}$$

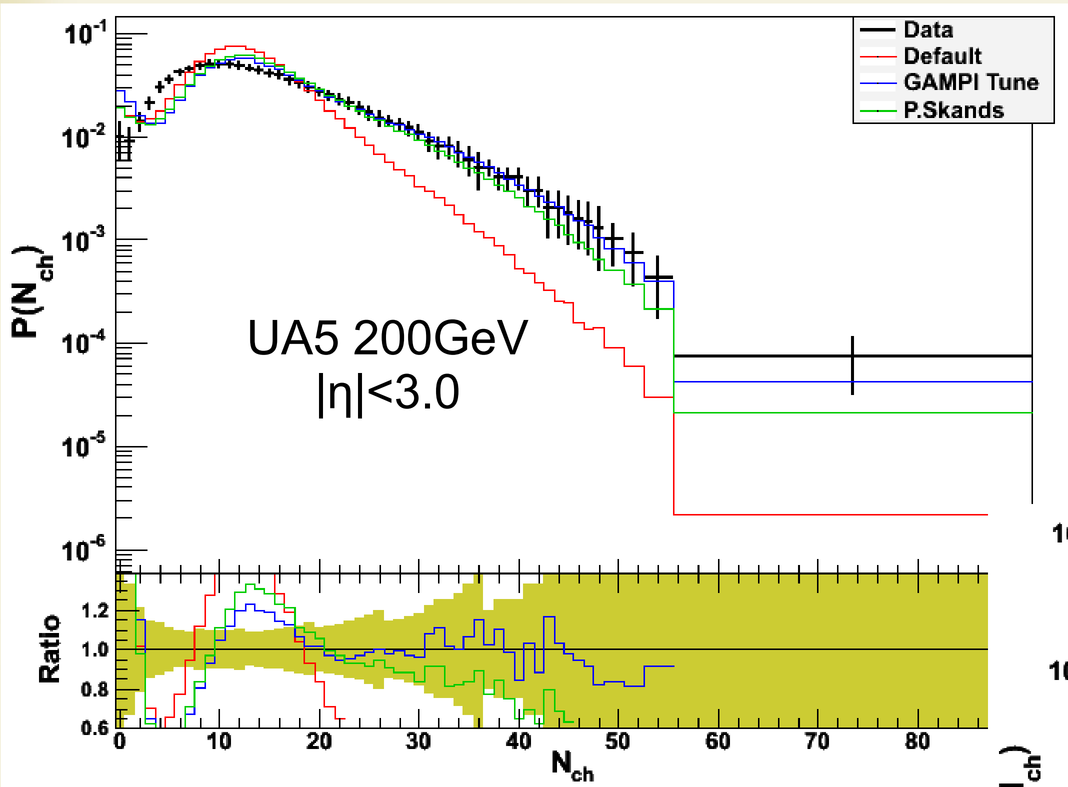
χ^2 *Function*

Genetic **A**lgorithms and **M**essage **P**assing **I**nterface

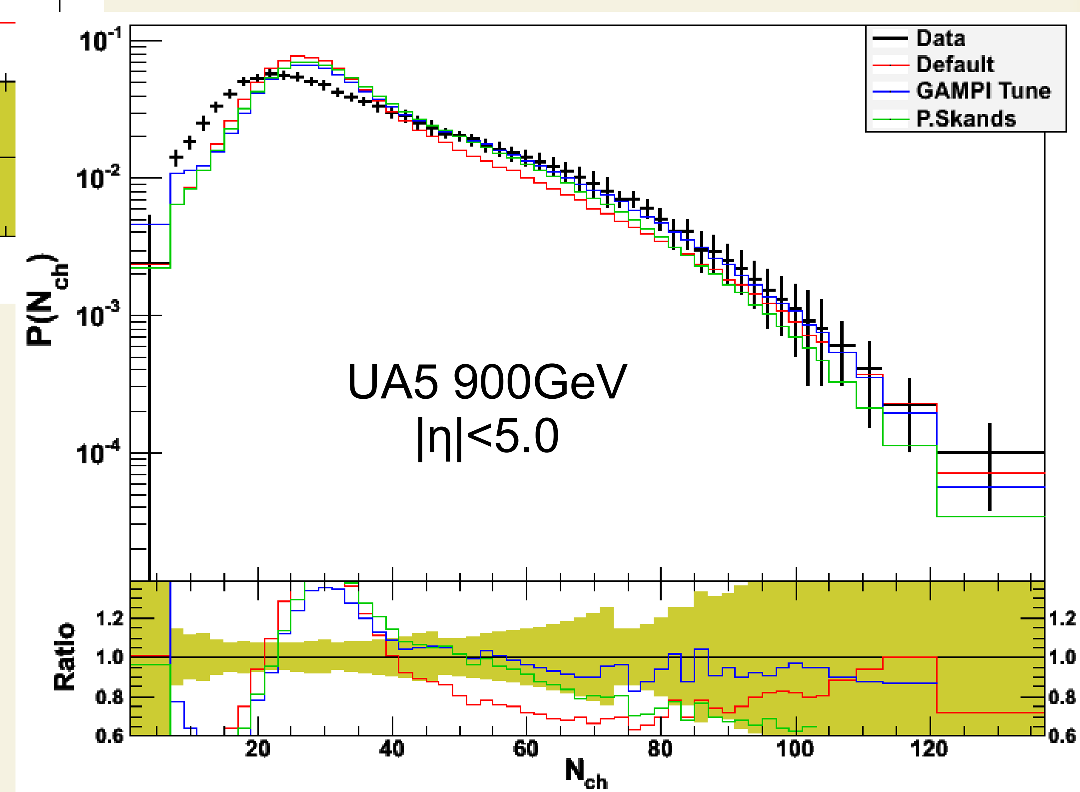


PYTHIA tune for Minimum Bias

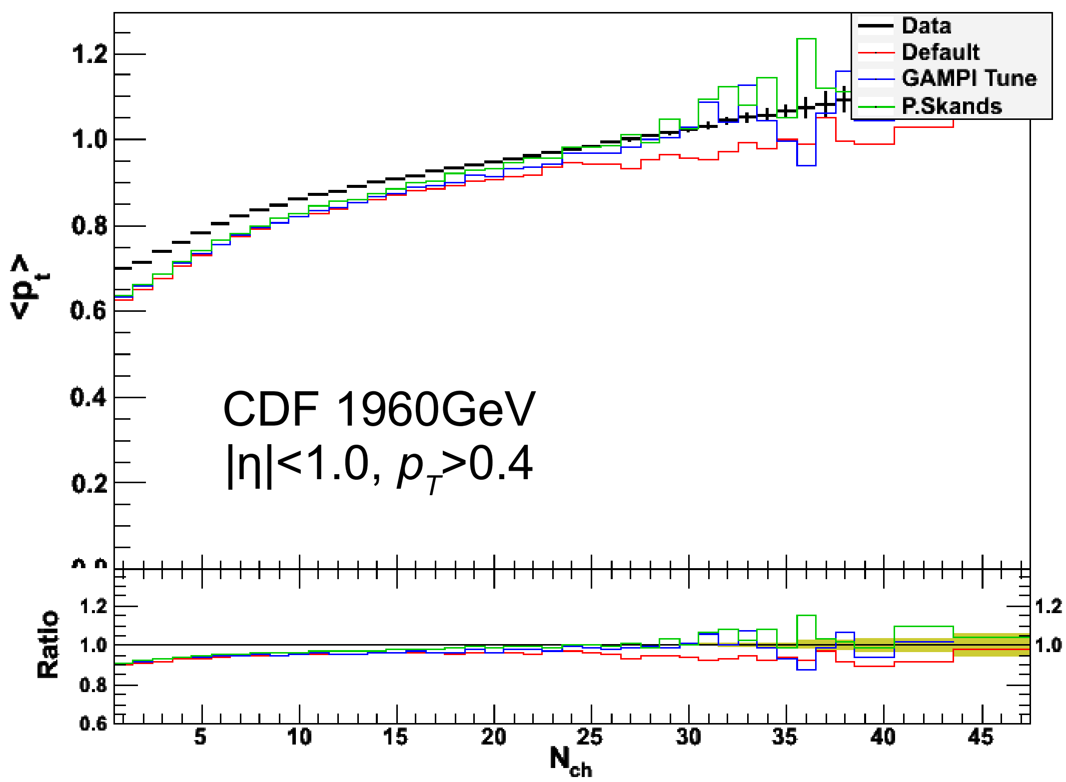
PYTHIA 8 Tune



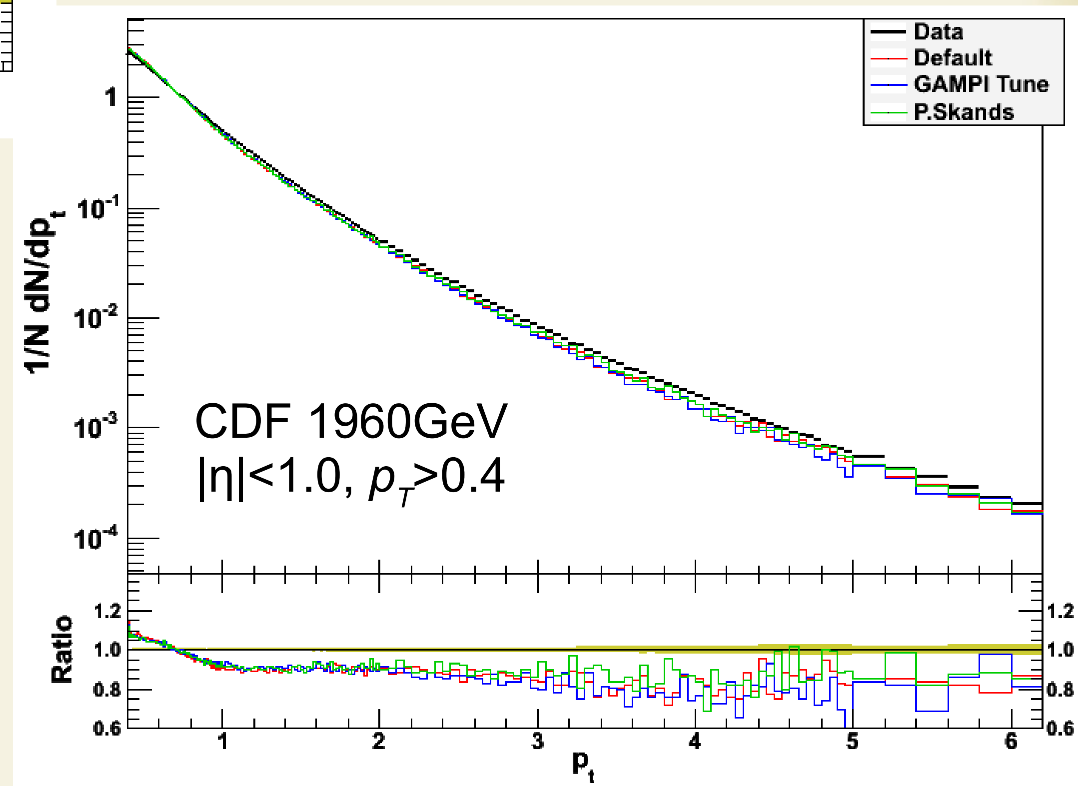
Charged Particle Multiplicity distribution is significantly improved. Low multiplicity region is not well described by the model.



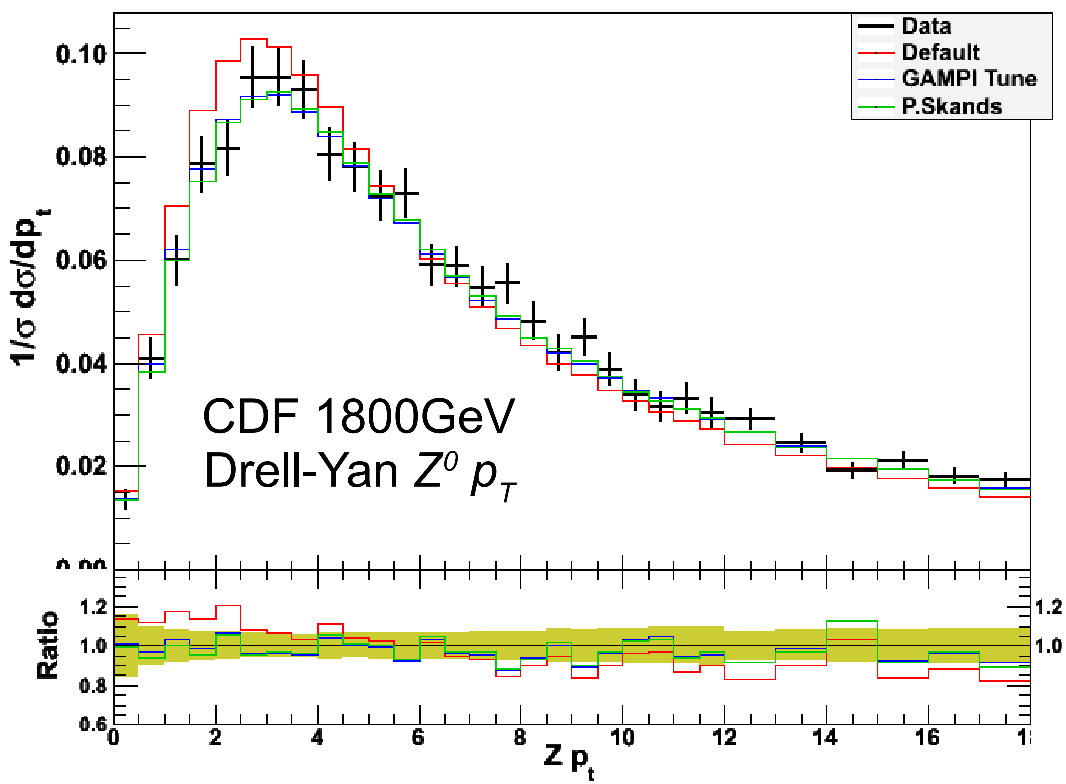
Genetic Algorithm tune(blue) has slightly better description than the tune from Peter Skands(green). Other distributions are similar.



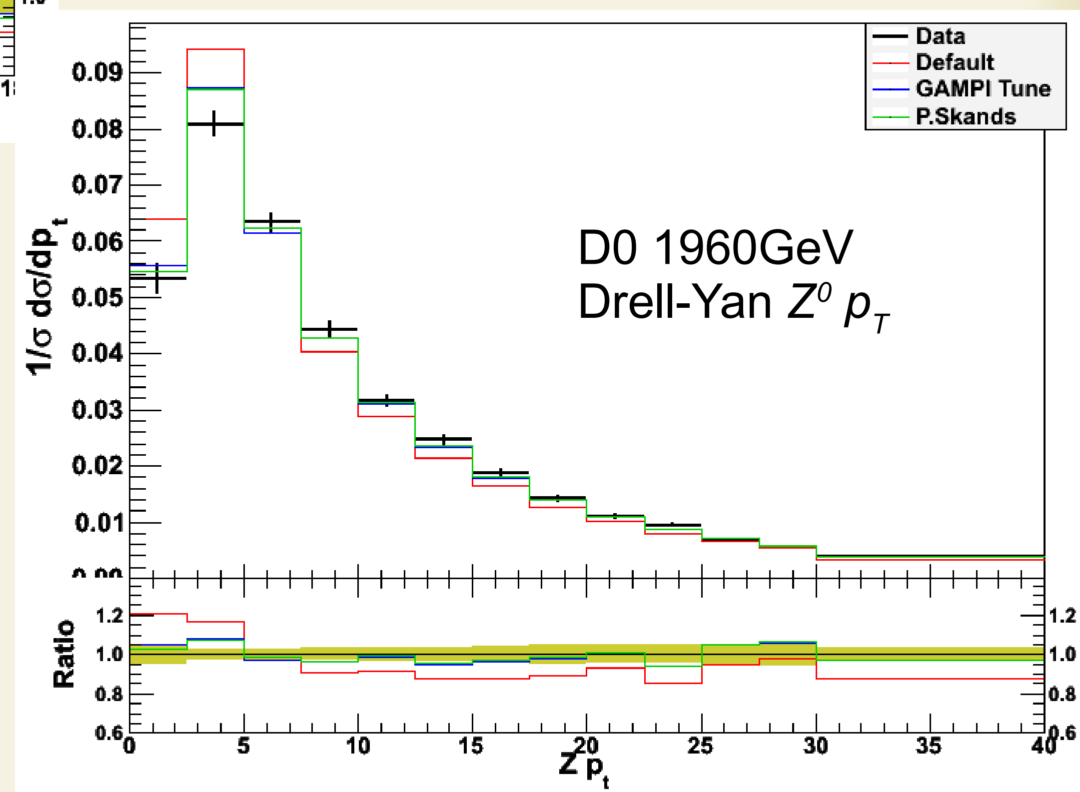
Mean p_T distribution agreement is also better than default. Low multiplicity region is slightly off from the measurement.



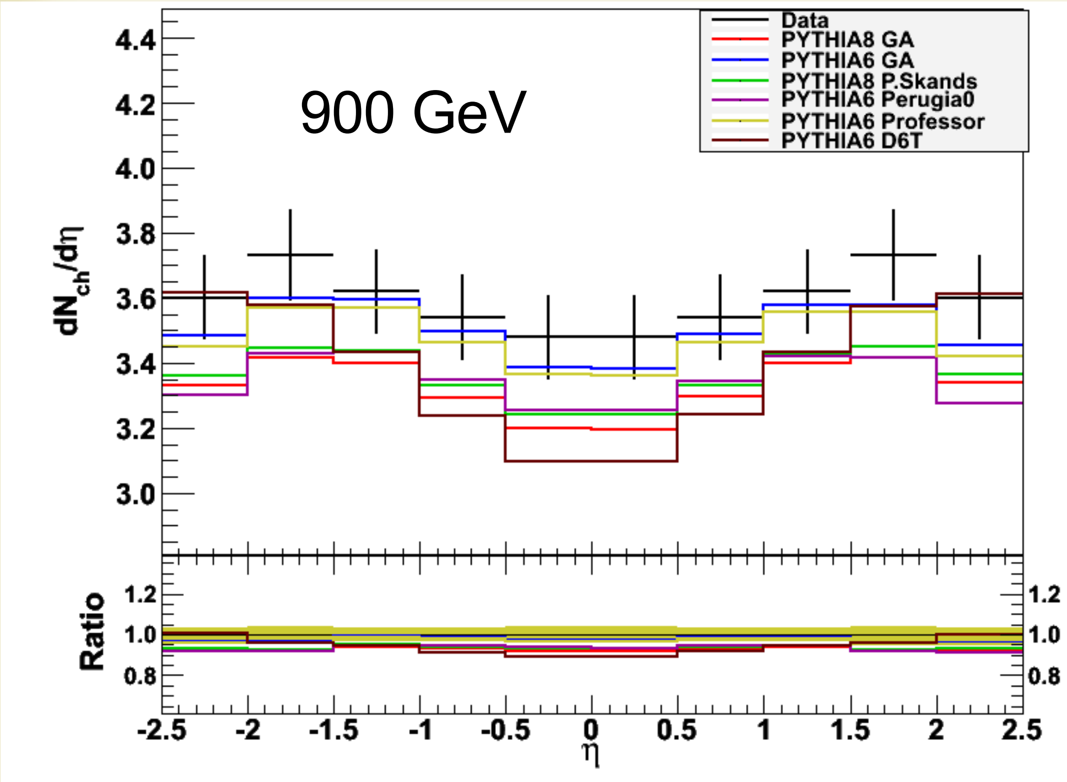
No significant difference between the tunes in p_T distribution



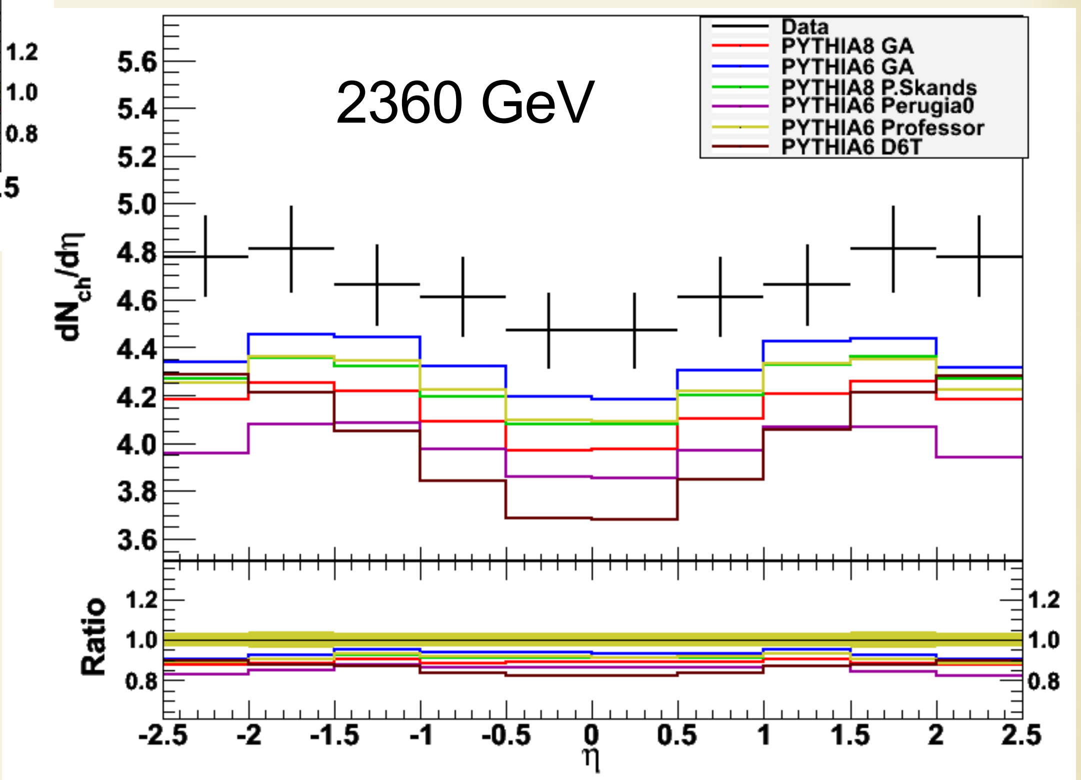
Slight improvement on the description of p_T of Z^0 in Drell-Yan events at 1800 and 1960 GeV



CMS dN/dη data



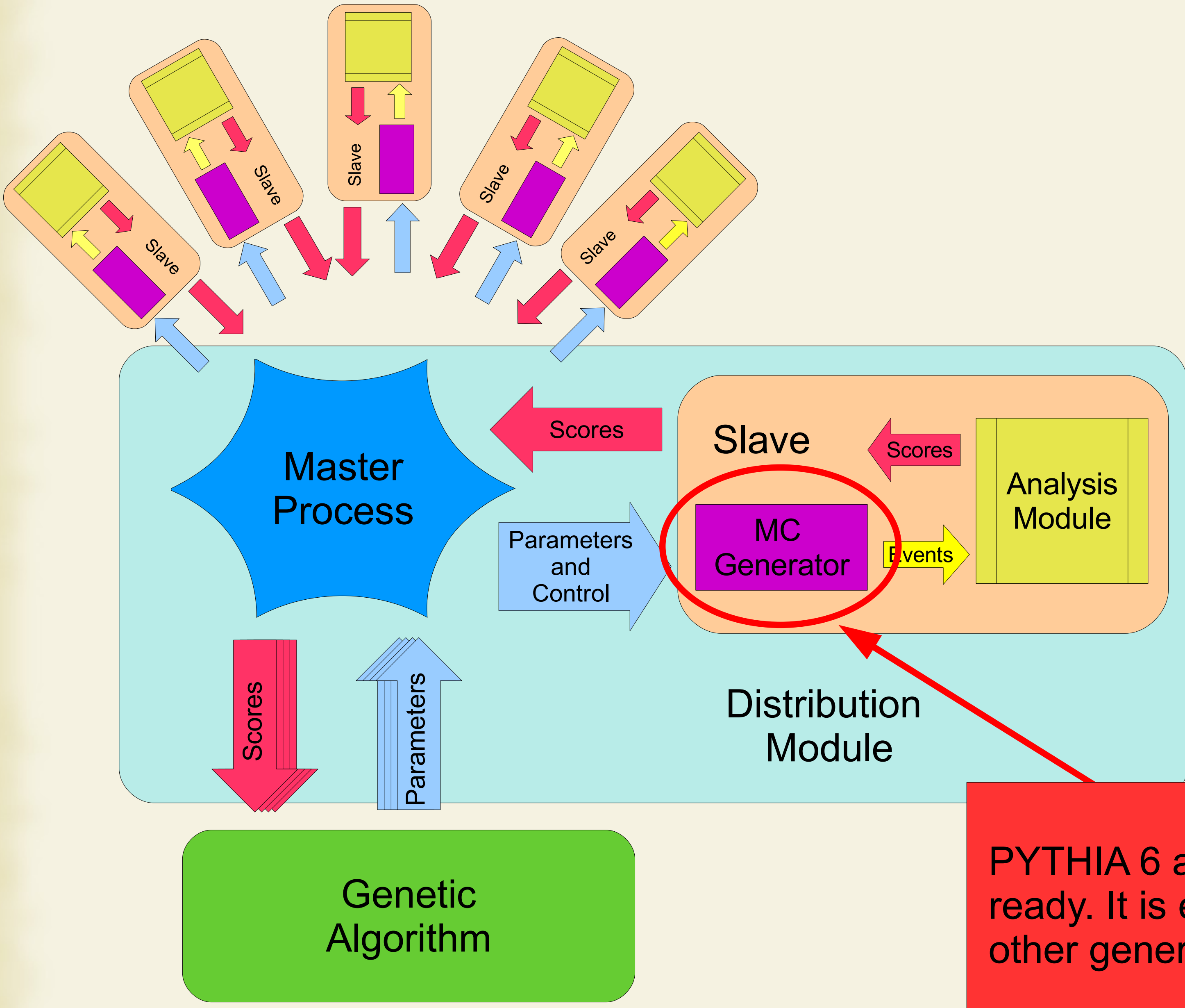
CMS (NSD-corrected) charged particle pseudorapidity distributions



Transverse-momentum and pseudorapidity distributions of charged hadrons in pp collisions at $\sqrt{s}=0.9$ and 2.36 TeV

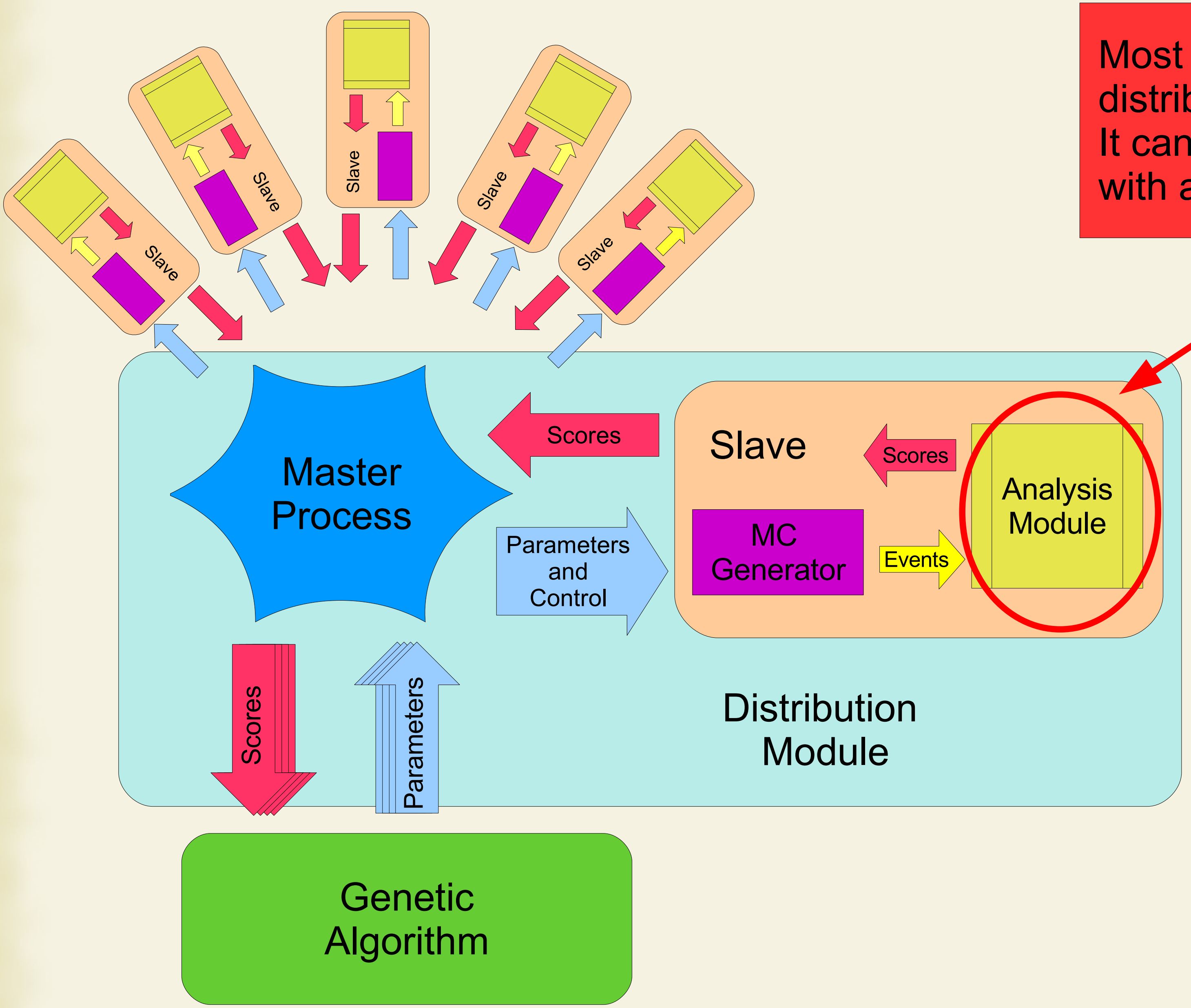
CMS Collaboration
arXiv:1002.0621v2

Current Status



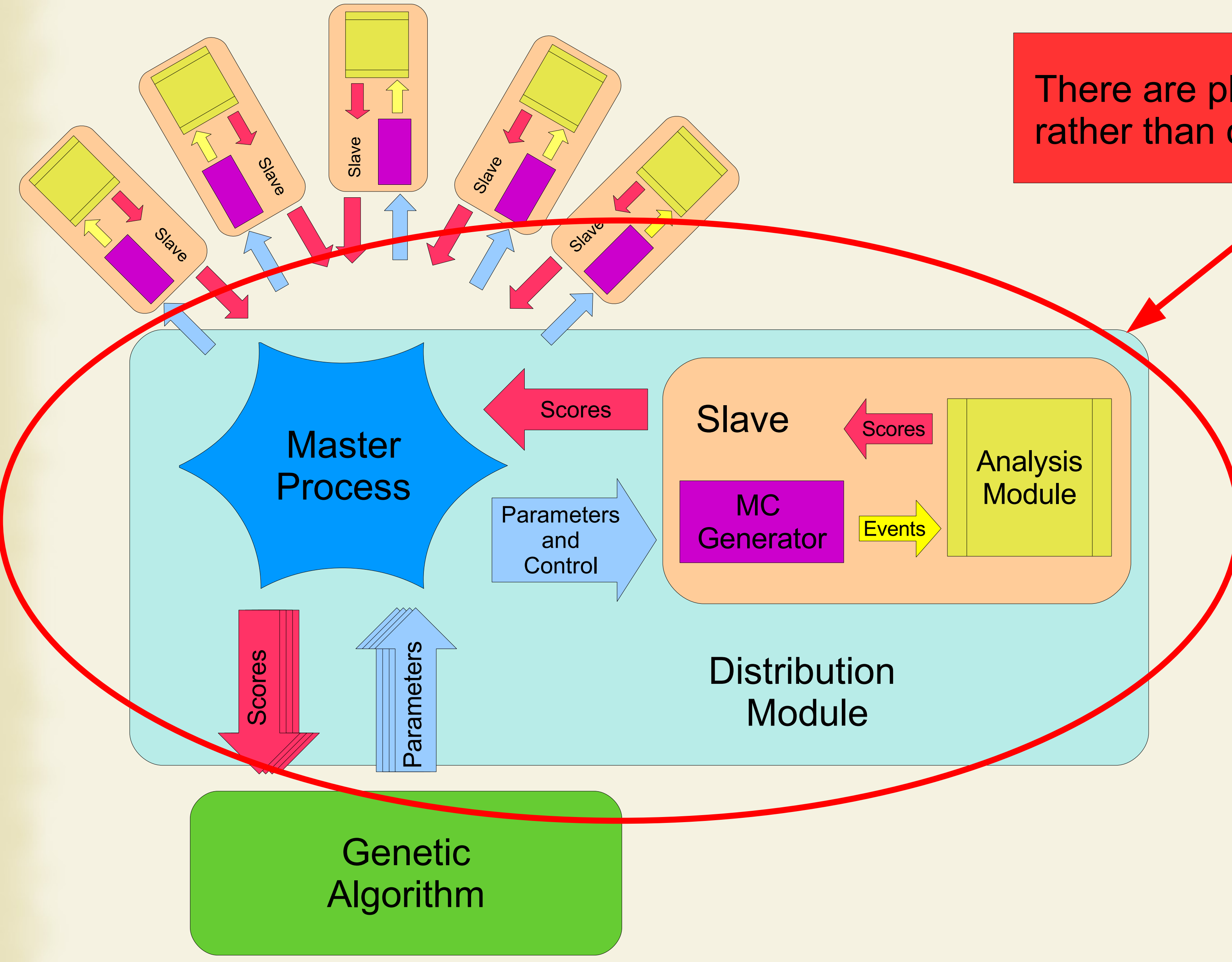
PYTHIA 6 and 8 interfaces are ready. It is easy to extend it to other generators.

Current Status



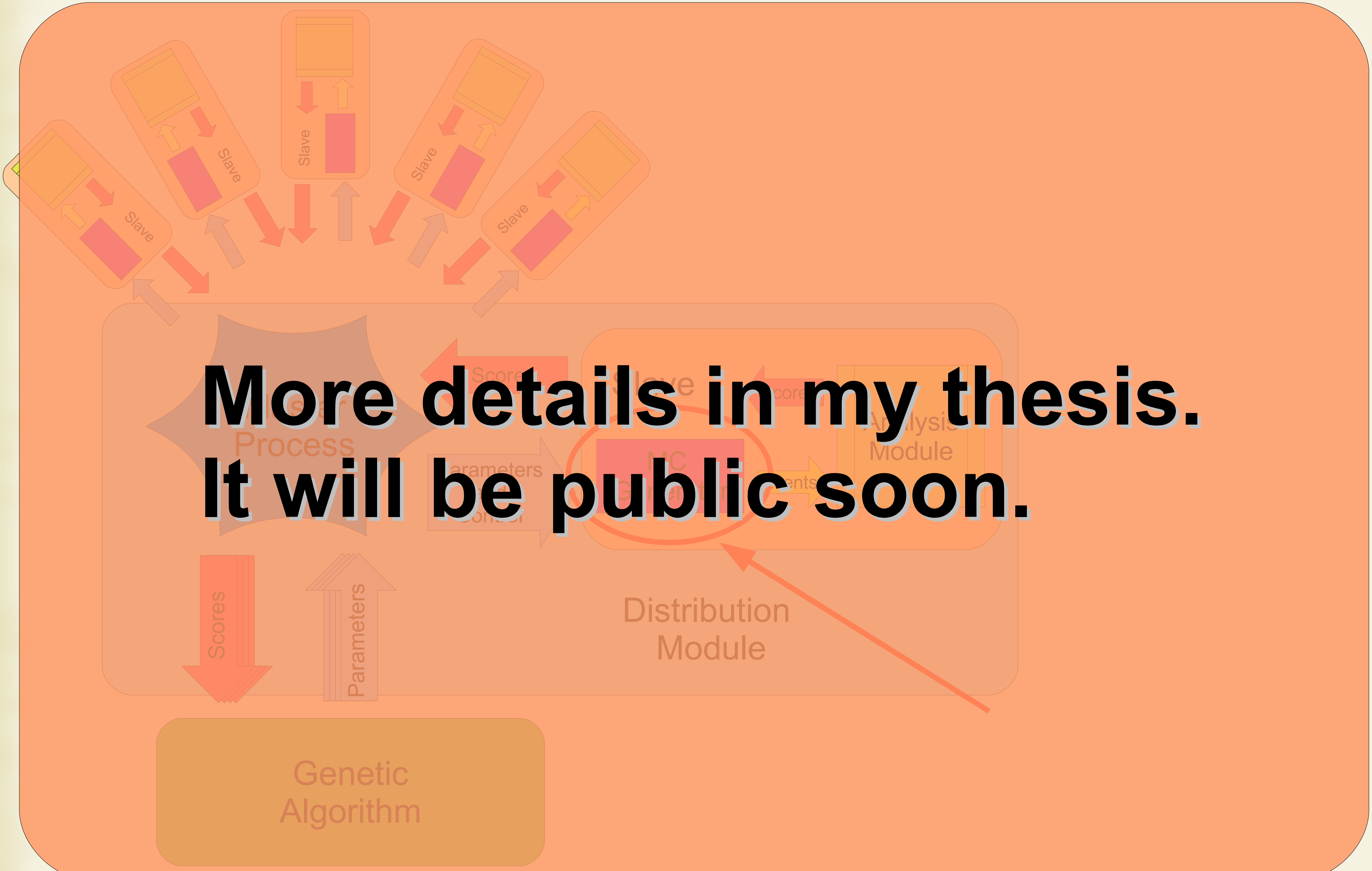
Most minimum bias distributions are implemented. It can be extended or replaced with another library.

Current Status



There are plans to use grid rather than clusters

Current Status



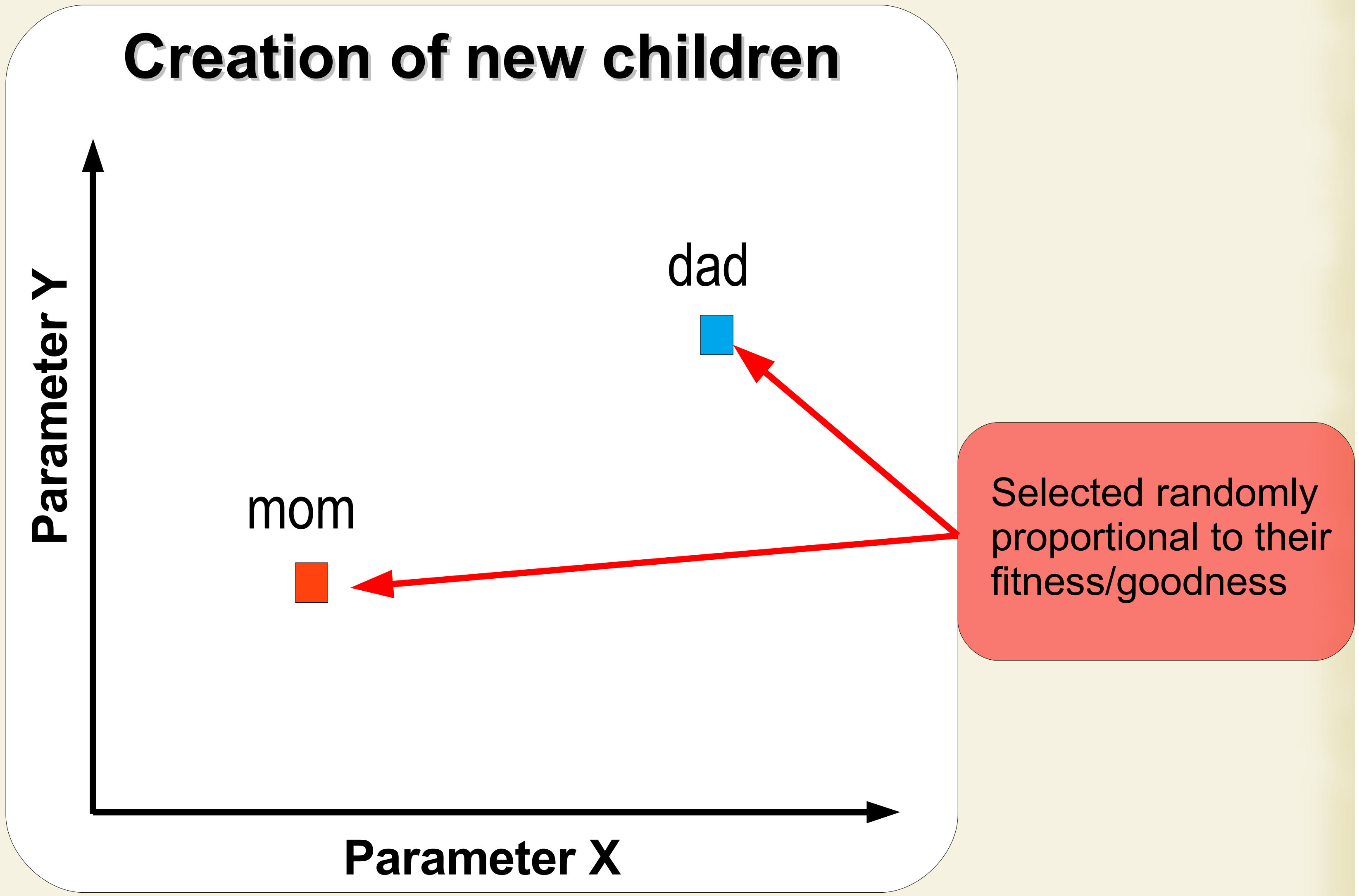
More details in my thesis.
It will be public soon.

Conclusions

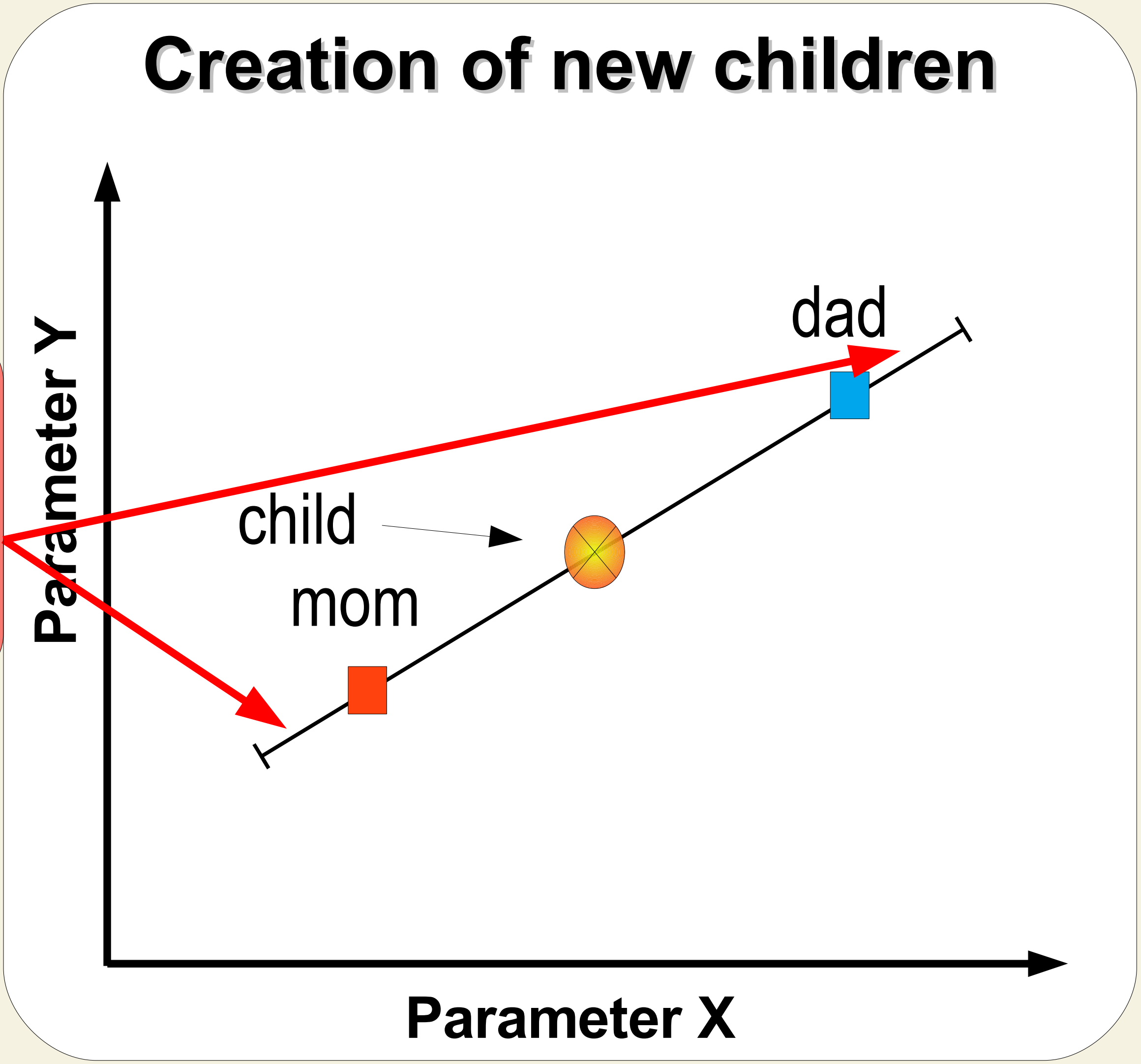
- Genetic Algorithms can be used in MC tuning and GAMPI provides a way of doing it automatically.
- It uses exact generator response; no systematic errors from the method, no approximations.
- Shape of the fitness (hyper-)surface is not an issue.
- Modular approach makes it easy to adapt to other generators and analyses.
- Applied for a repository in HepForge, waiting for it to make the code public.

Thank you!

Procreation

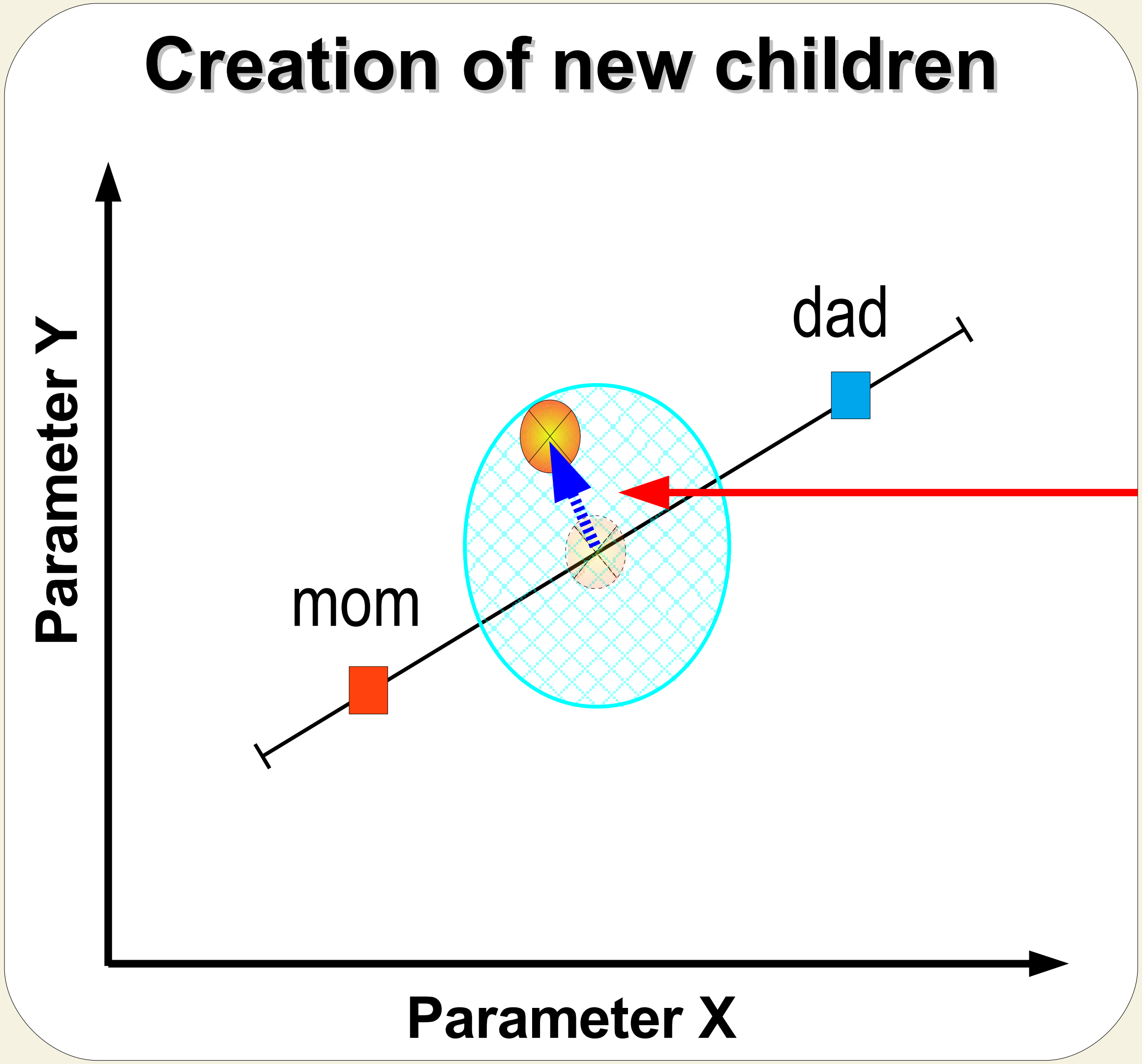


Procreation



Child is generated at a random point on the line segment connecting parents with some extra in both ends.

Procreation

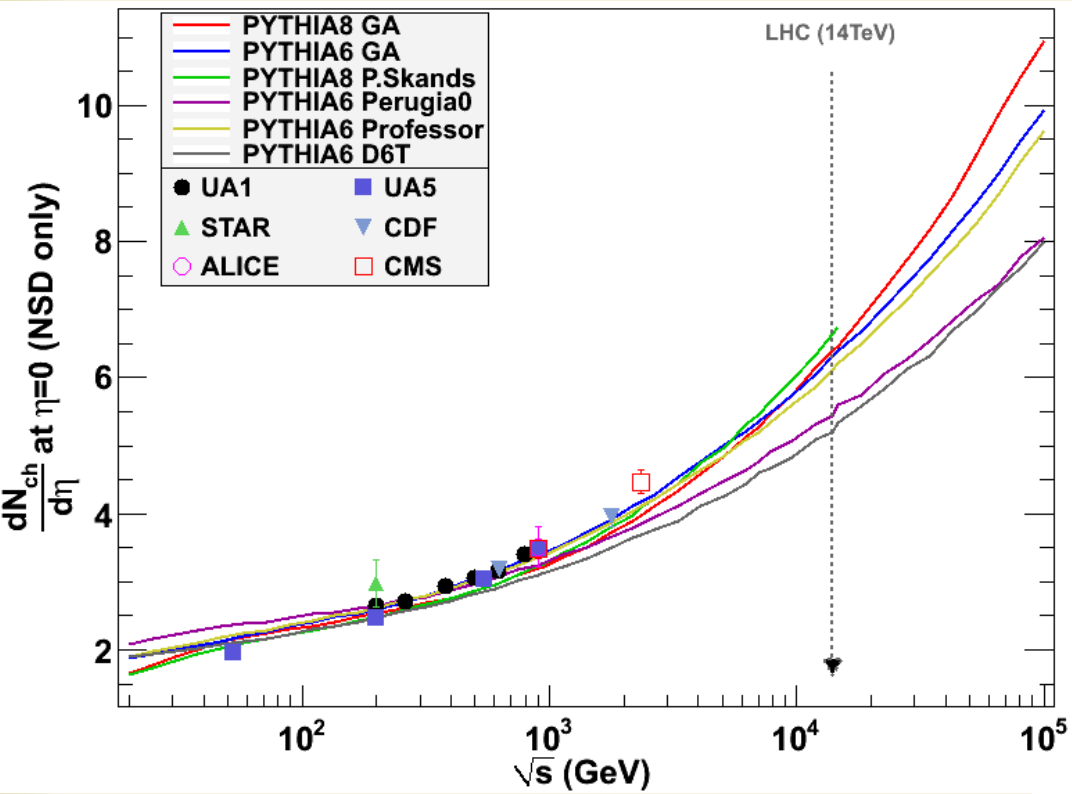


Child is randomly shifted with a given probability within a circle of predefined radius

Data Sets

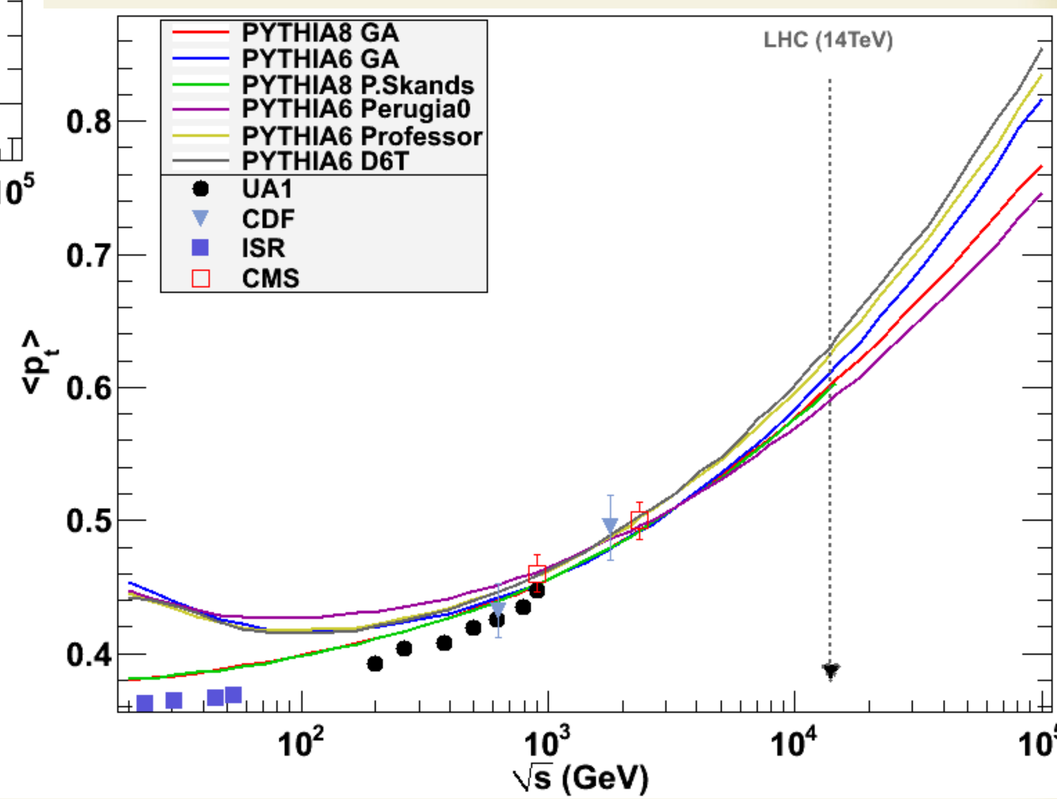
Coll.	$\sqrt{s}$	Data Set
CDF	1960 GeV	$< p_T >$
		dN_{ch}/dp_T
		UE Trans-Min $\Sigma\, p_T$ density in Drell-Yan
		UE Trans-Max $\Sigma\, p_T$ density in Drell-Yan
		UE Trans-Min $\Sigma\, N_{ch}$ density in Drell-Yan
		UE Trans-Max $\Sigma\, N_{ch}$ density in Drell-Yan
	1800 GeV	dN_{ch}/dp_T $1/\sigma d\sigma/dp_T$ (Drell-Yan)
	630 GeV	dN_{ch}/dp_T
D0	1960 GeV	$d\sigma/dp_T$ Drell Yan
		$d\sigma/dp_T$, $y > 2$ $p_T < 30$ GeV
UA5	900 GeV	$dN_{ch}/d\eta$
		N_{ch}
		N_{ch}, $\|\eta\| < 0.5$
		N_{ch}, $\|\eta\| < 1.5$
		N_{ch}, $\|\eta\| < 3.0$
		N_{ch}, $\|\eta\| < 5.0$
	546 GeV	N_{ch}
	200 GeV	$dN_{ch}/d\eta$
		N_{ch}
		N_{ch}, $\|\eta\| < 0.5$
		N_{ch}, $\|\eta\| < 1.5$
		N_{ch}, $\|\eta\| < 3.0$

Center-of-Mass evolution

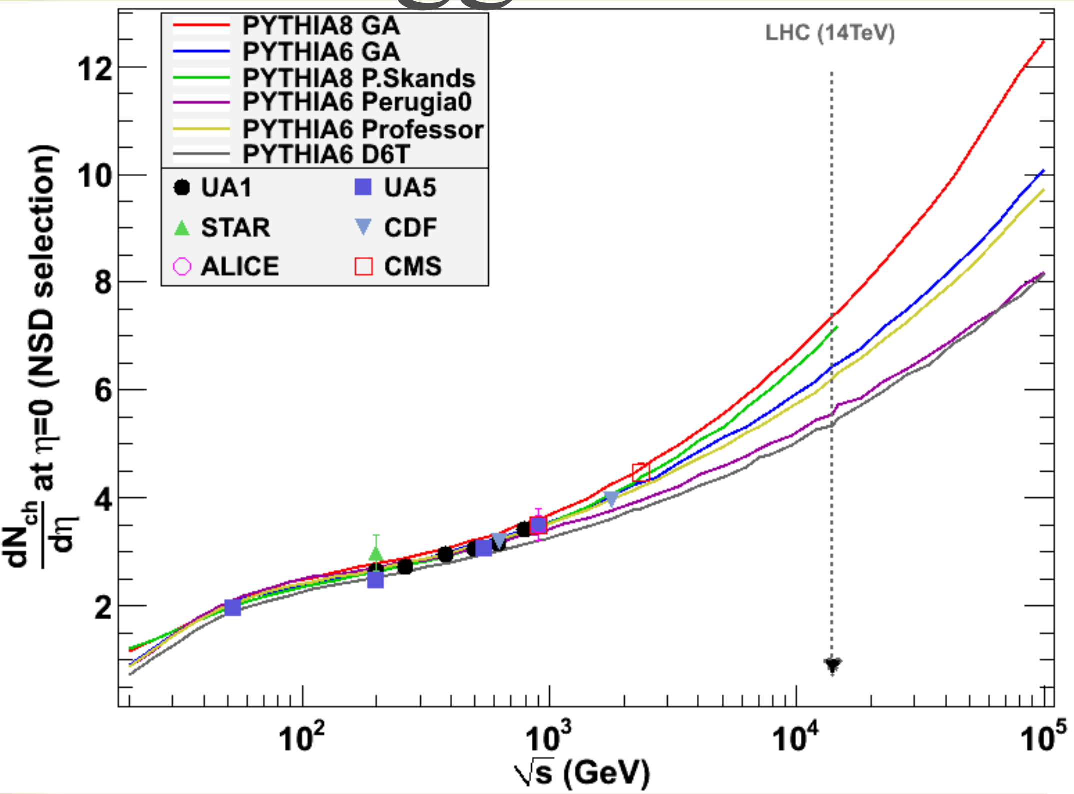


Central charged particle pseudorapidity

Mean p_t



Trigger selected dN/dη evolution



UA1 Selection,
A charged particle in 1.5<|η|<5.5, both hemispheres

CMS selection
A particle with E>3Gev in 2.9<|η|<5.2, both hemispheres