Parameter Optimisation in Monte Carlo Event Generators

Hendrik Hoeth
(University of Wuppertal)

Overview

Motivation – "why" and "what's the problem"

Strategy – "how to tune"

Summary – "what we learned" and a plea to experimentalists

Motivation

All generators are based on phenomenological models: dipole cascade, string fragmentation, cluster hadronisation, . . .

The models have free parameters which are a priori unknown: q_0^2, σ_q, Lund A and B, flavour ratios, . . .

We want the MC to describe the data the best possible way. So the parameters need to be tuned.

Even parameters like α_s need to be optimised!

Problems

The parameters are highly correlated
$\Rightarrow$ can't be tuned one after the other.

Many parameters to be tuned ($\mathcal{O}(10)$).

Tuning all parameters at the same time puts us into a high dimensional parameter space.

Brute force approaches don't work: Running the MC generator takes too long for every point in the parameter space (= setting of parameters).

We haven't the money, so we've got to think.

– Lord Rutherford

Divide and conquer:

Split the task into parts (parton shower, hadronisation)
$\Rightarrow$ cut down the number of parameters.

Be lazy:

Predict the MC output for any parameter set.

A strategy

1. Choose a tuning interval for the parameters, then pick random points in parameter space and run the generator with these settings.

2. Interpolate between points $\Rightarrow$ prediction of the MC output at any specific parameter setting.

3. Fit this prediction to data (minimal χ^2).

4. Repeat the fit for different combinations of observables.

5. Choose the nicest set of parameters.

1. Choosing parameters

Pick the parameters you want to tune:

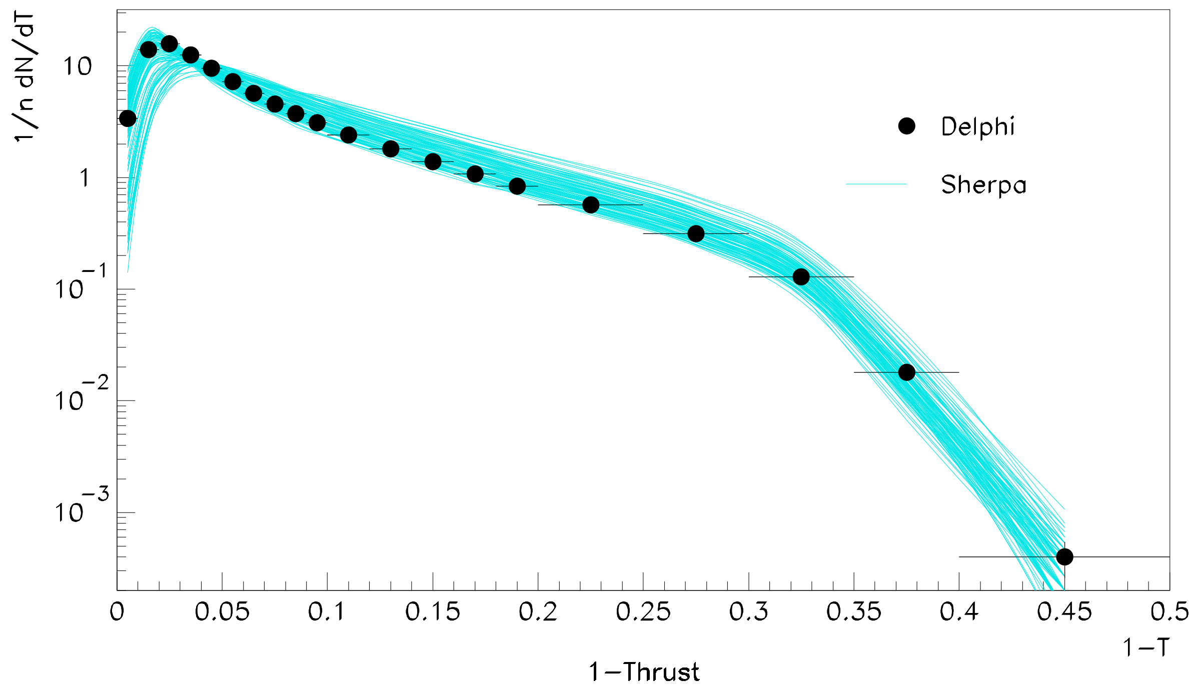
- Tune everything that is important.
- But remember: Each additional parameter adds one dimension to the parameter space.

Define parameter intervals:

- Make the interval large enough so that the result will not be outside.
- But remember: Cutting down 10 intervals by 10 % shrinks the volume of the parameter space by 2/3.

Now pick random points in parameter space and run the generator for each setting.

Calculating observables yields plots like this:



Every line corresponds to a certain setting.

2. Predict the Monte Carlo

Get a bin by bin prediction for the MC response as function of the parameter set $\vec{p} = (p_1, p_2, \ldots, p_n)$.

Using a second order polynomial takes the correlations between the parameters into account:

$$X_{\mathrm{MC}}(p_1, p_2, \ldots, p_n) =$$
$$A_0 + \sum_{i=1}^{n} B_i p_i + \sum_{i=1}^{n} C_i p_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} D_{ij} p_i p_j + \cdots$$

3. Fit the prediction to data

Having A_0, B_i, C_i and D_{ij} we can predict the MC response for any set of parameters very fast. This prediction can be fitted to data, minimising the χ^2:

$$\chi^2(\vec{p}) = \sum_{\text{observables}} \sum_{\text{bins}} \left(\frac{X_{\text{data}} - X_{\text{MC}}(\vec{p})}{\sigma_{\text{data}}} \right)^2$$

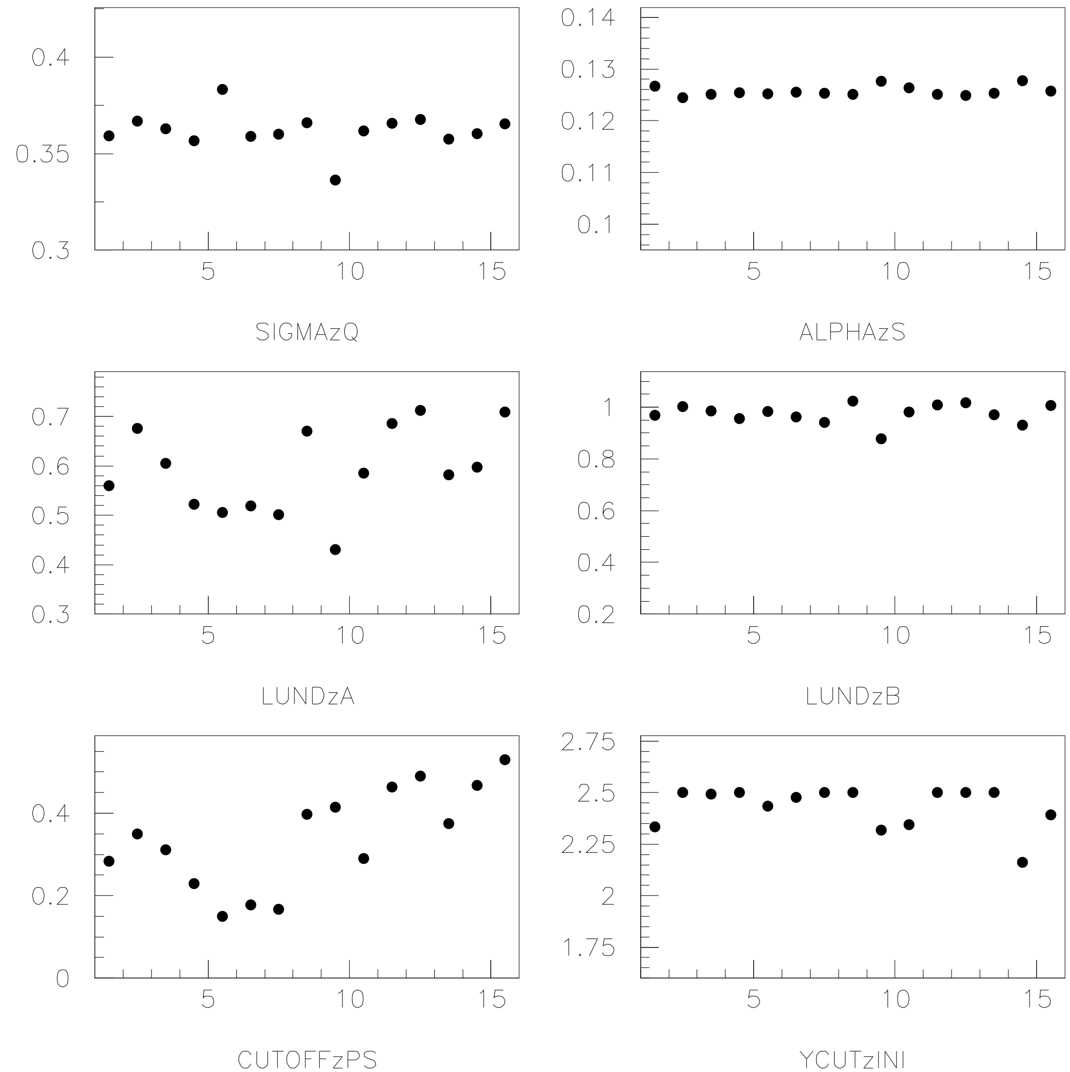
Include all the relevant data distributions in the fit!

This fit only takes seconds (as compared to days or weeks for a brute force approach).

4. Use different data sets

Using different combinations of observables yield different optimisation results.

Learn something about correlations and stability of the tuning.



What data should we use ...?

It depends . . .

In general: Use observables which are physically related to the parameters you want to tune!

Examples:

For the parton shower use event shapes like Thrust, Sphericity, Planarity, Major, Minor, (differential) jet rates, p_t spectra, N_{ch}, . . .

For hadronisation use identified particle spectra, multiplicities . . .

... and what data is available?

- LEP has published *lots* of high precision data.
- There is good data from SLD and the DESY experiments.
- There is very little useful data from the Tevatron!

To compare data to MC either the data needs to be acceptance corrected or the MC needs to be folded with the detector response. *Most of the published Tevatron data satisfies neither condition and can't be compared to anything!*

Summary

- Monte Carlo event generators are based on phenomenological models.
- The model parameters need tuning to describe the data.
- Parameters are correlated and have to be tuned simultaneously.
- Creating predictive functions for the MC response helps to fit the model parameters to data.
- Good data is crucial for a good tuning.

Plea to the experimentalists

If you want good Monte Carlo, please provide the MC authors with good data to tune their generators