Monte Carlo Tuning at Future Lepton Colliders

Leif Gellersen

Lund University

leif.gellersen@hep.lu.se

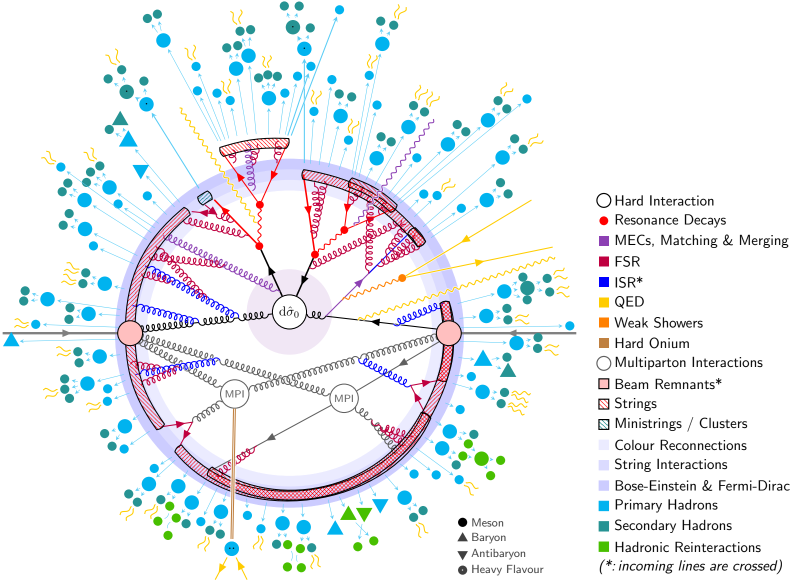
ECFA Higgs Factories: 2nd Topical Meeting on Generators
Brussels, June 21st to 22nd, 2023

Overview

- Datasets, observables & physics
- Status: Tools & methods
- Tuning in light of recent progress & future lepton colliders

Datasets, observables & physics

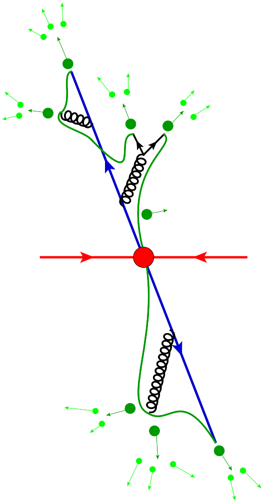
An event in PYTHIA 8 [Bierlich, Chakraborty, Desai, LG, et al. (2022)]



MC Event Generation: Lepton Collisions

- Cleaner environment as compared to hadronic collision
  - no MPI
  - much simpler PDF and ISR treatment
- Perturbative methods well known $\rightarrow$ work towards precision
  - Hard interaction: Matrix elements (LO/NLO)
  - Radiative Corrections: Parton shower in final state
- Non-perturbative models
  - Hadronization
  - Hadron decays

Models well motivated, but still many parameters, need optimization



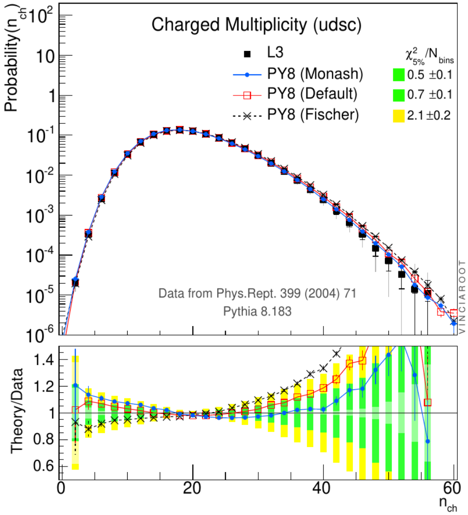
e^+e^- collision

Tuning: General Idea

- Optimize parameters based on well-measured data
- Factorize as much as possible (assuming universality)

FSR	e^+e^- data: LEP event shapes
Hadronization	Many parameters, model dependent. Use LEP identified particle spectra
ISR and UE	Use hadron collider data

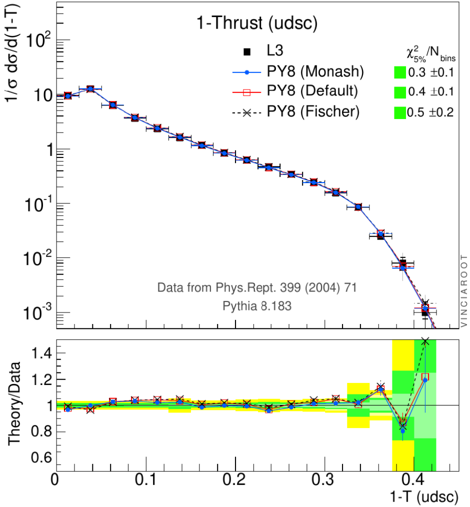
- PYTHIA's defaults based on Monash tune: explains correspondence between physics models, observables and data sets



arXiv:1404.5630, P. Skands et al., 2014

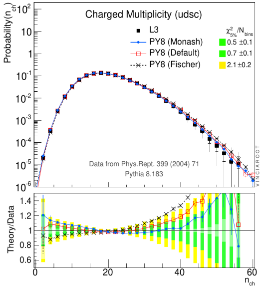
Final-State Radiation

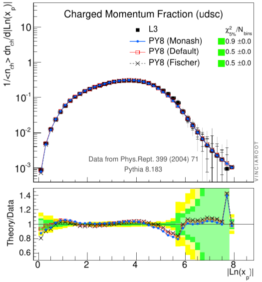
- Main parameter governing FSR: $\alpha_s(M_Z)$
- Best fitted to $e^+ + e^-$ event shapes (Thrust, C, D, B_W, B_T), light flavour tagged ($udsc$), from e.g. L3
- Further choices: running order (1), mimics NLO K factor for hard emissions
- IR cutoff $p_{\perp,\mathrm{min}}$ close to Λ_{QCD}, smooth transition to non-perturbative string breaks



Light-Flavour Fragmentation

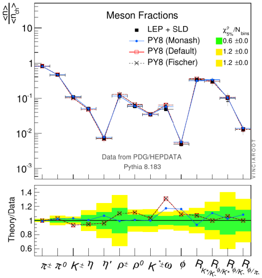
- Post-shower: non-perturbative Lund string fragmentation model converts partonic state to on-shell hadrons
- Main parameters:
  - $\sigma_\perp$ governs $p_\perp$ kicks form string breaks, determined through first bins of event shapes
  - a, b parameters govern longitudinal energy of hadrons through fragmentation function $f(z) \propto \frac{(1-z)^a}{z} \exp\left(\frac{-bm_\perp^2}{z}\right)$. a suppresses large hadron energy z, b suppresses small z
- Determine by simultaneously optimising inclusive charged-particle momentum and multiplicity spectra, anti-correlated

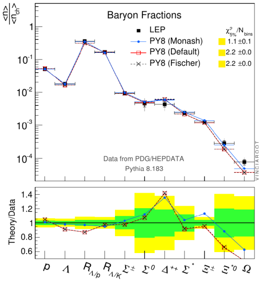




Identified Particles

- Flavour composition determined through light-flavour meson and baryon multiplicities, from PDG and LEP experiments
- Determines StringFlav parameter family in Pythia
- Some tension between PDG and values from respective experiments
- Similarly, use particle rates to determine relative rates of vector-mesons vs. pseudoscalars





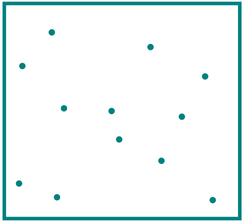
Tools & methods

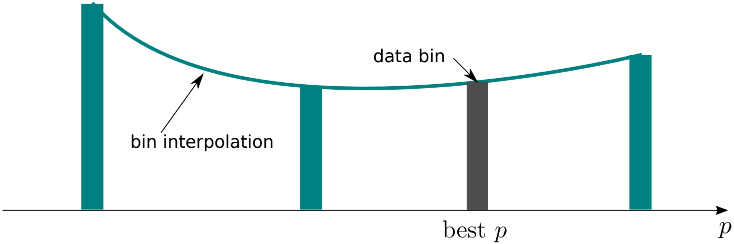
Tools & methods

- From manual expert tunes to automation
  - Manual expert tune: fit parameters one by one, iterate. Requires extensive knowledge, and much work.
  - Select parameters and corresponding observables carefully, check data for consistency (see universality)
  - BUT: make it reproducible, ideally in a mostly automated way. Allows for quick and easy retune for adapted models complying with exact same methodology.
  - Significant computing resources needed
- Available tools
  - Professor [Buckley, Hoeth et al (2010), arXiv:0907.2973]
  - Autotunes [Bellm, LG (2020), arXiv:1908.10811]
  - Event generator tuning using Bayesian optimization [Ilten, Williams, Yang (2017), arXiv:1610.08328]
  - Apprentice [Krishnamoorthy et al (2021), arXiv:2103.05748] [Wang, Krishnamoorthy et al (2022), arXiv:2103.05751]
  - ...
- General methodology: weights, uncertainties, universality

Professor

- PROFESSOR: Python package for MC tuning, highly automated, includes validation tools
  [Buckley, Hoeth et al (2010), arXiv:0907.2973]
- Generate MC pseudodata $f_i(\vec{p})$, compare to experimental data bin $\mathcal{R}_i$
- Iterative MC event generation slow $\rightarrow$ Use bin-wise parametrization of MC generator response





- Minimize $\chi^2(\vec{p}) = \sum_i w_i \frac{(f_i(\vec{p}) - \mathcal{R}_i)^2}{\Delta_{f_i}^2 + \Delta_{\mathcal{R}_i}^2}$, with data uncertainty Δ_i, bin weights w_i

AutoTunes

[Bellm, LG (2020), arXiv:1908.10811]

Problem

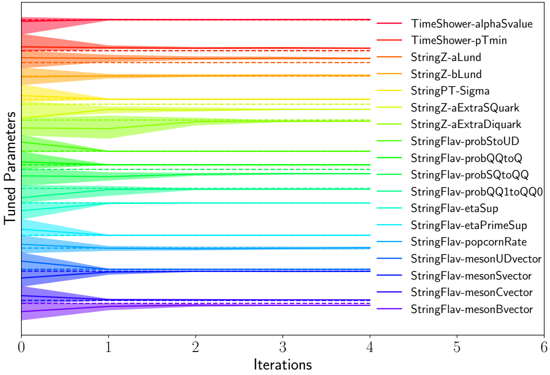
- Polynomial interpolation only possible for $\lesssim 10$ parameters
- Interpolation only good if ranges small enough
- χ^2 depends on weights $\rightarrow$ need to know data and generator

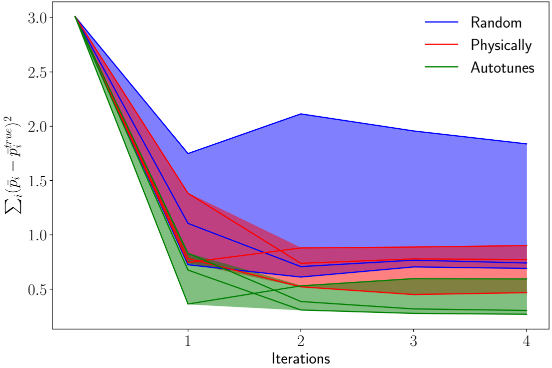
Goal

- Framework to reduce human interaction & make tune reproducible
- Tune many parameters at once: automatically divide into sub-tunes
- Set weights for observables automatically
- Allow for iterations with revised parameter ranges

AutoTunes: Iterative Pythia Tune to Pythia Pseudodata

Try to reproduce − − − values, ≈ 6000 DOF & 18 parameters





Bayesian Optimization

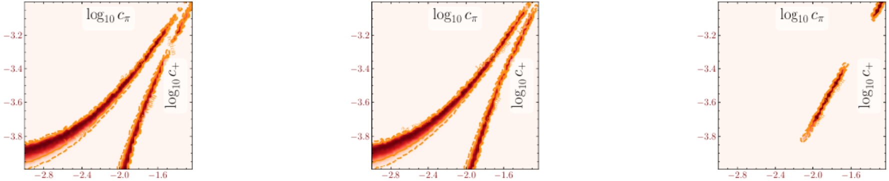
[Ilten, Williams, Yang (2017), arXiv:1610.08328]

- Study shows that tuning for lepton collider is possible using Bayesian Optimization, little expert-knowledge required
- Works without interpolation, by successive runs. All information used, not just local gradient
- Based on SPEARMINT software package
- Closure test: recover Monash tune
- Tune e^+e^-, 20 parameters, possible on laptop in few days
- $\Rightarrow$ could be interesting in lepton collider context

Apprentice

[Krishnamoorthy et al (2021), arXiv:2103.05748]

- Evolution of PROFESSOR framework introducing several improvements
- Polynomial fit not suitable for some observables ⇒ introduces rational approximation, more faithful, less limited in applicability range
- Automated weight assignment for each observable, based on different approaches
- Filtering: exclude data or observables the MC model cannot describe
- See [Wang, Krishnamoorthy et al (2022), arXiv:2103.05751] for detailed discussion of weight assignment



(a) Full simulation using MC generator

(b) Pole-free rational approximation

(c) Polynomial approximation

Role of Weights

- Problem with weights: threat to reproducibility and robustness of method if chosen by hand
- Different purposes of weights:
    - Favour reliable, high-quality data
    - Focus on relevant data (extreme: can regard weigths as selection of observable, all other zero)
    - Potentially: take into account correlations in parameters
    - Potentially: take into account correlations in data: Avoid over-representation of very well measured observables
- Treatment of each data bin as independent problematic, not solved by mentioned methods

Uncertainties

- From different data on same observable $\Rightarrow$ need careful pre-selection, or rely on outlier detection
- Assume baseline uncertainty on MC prediction to avoid unreasonable fine-tuning to data with small uncertainties
- Correlated parameters $\Rightarrow$ eigentunes. Don't miss correlations if tuning successively!
- Systematic tune uncertainties should go beyond data constraints (eigentunes), combine with model variations (see e.g. Les Houches 2017 SM report p. 224 for cross talk of parameter optimization and perturbative variations)

Universality

Independent tunes for different...

- ... CM energies
- ... processes
- ... experiments
- ... observables
- ... ?

Shows what a model can/cannot describe → results allow us to learn about physics models

- automated tuning greatly simplifies such studies. Examples:
- minimum-bias tunes at different energies: Good universality except for CR strength [Schulz, Skands (2011), arXiv:1103.3649]
- hadronization parameters at LEP, different experiments, different observables. Gives envelope of uncertainties [Amoroso, Caron et al (2019), arXiv:1812.07424]

Tuning in light of recent progress & future lepton colliders

Tuning in light of recent progress

- Shower developments to take into account for precision tunes
  - Matrix-element corrections
  - N^kLO matched predictions
  - Multi-jet merged predictions
  - Improved logarithmic accuracy
  - NLO showers
  - Subleading color corrections
  - QED & EW showers
- What this means for tuning
  - Ideally: more universal tunes, due to less freedom in perturbative input
  - Or: find discrepancies that allow to refine models

Past and future lepton colliders

- Lepton collider data very valuable for factorized tuning approaches
- Reanalysis of LEP data might give more consistent results across experiments $\rightarrow$ stronger constraints on leptonic tunes
- Large statistics from FCC-ee promises unprecedented baseline for precision tunes
  - For shower α_{s}, and shower modeling in general
  - For fragmentation parameters (both light and heavy flavours)
  - For identified particles
- For future efforts: focus on reproducibility, assessment of universality

Summary & Outlook

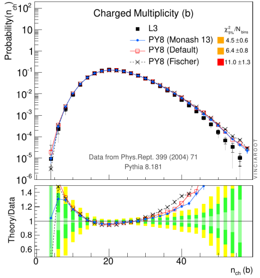
Summary & Outlook

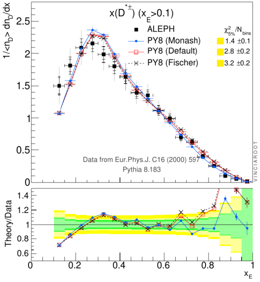
- Robust tunes based on carefully chosen data essential for reliable predictions
- Automated tools available, with automated weight setting, focus on reproducibility
- Recent progress on generators might give more universal tunes, or point to model shortcomings
- Lepton collider data very valuable for tuning

Backup

Heavy-Quark Fragmentation

- Lund fragmentation function modified for heavy quarks:
  $$f_{\text{massive}}(z, m_Q) \propto \frac{f(z)}{z^{b r_Q m_Q^2}}$$
- Captures effect of massive string endpoints, expect $r_Q \simeq 1$
- Suppresses $z \to 1$ region
- Determine for c and b independently
- For b: use b-tagged event shapes & multiplicities, scaled momentum of B hadrons
- For c: use D meson momentum spectra, c-tagged event shapes desirable but not available at the time





AutoTunes: The Idea

- Normalize each bin f_i and each parameter p^α to $[0, 1]$
- Find slopes $\mathcal{S}_i^\alpha$
- $\vec{\mathcal{S}}_i$ vectors in parameter space
- Normalize: $\mathcal{N}_i^\alpha = \frac{\mathcal{S}_i^\alpha}{\sum_i \mathcal{S}_i^\alpha}$
- Find $\vec{\mathcal{J}} = (1, 0, 0, 1, 0, \ldots, 1)$ that maximizes $\mathcal{M} = \sum_i (\vec{\mathcal{N}}_i \cdot \vec{\mathcal{J}})^2$
  $\rightarrow$ "Most correlated" subset of parameters: tune in one step
- Use weights $w_i = \frac{(\vec{\mathcal{N}}_i \cdot \vec{\mathcal{J}})^2}{\sum_\alpha \mathcal{N}_i^\alpha}$, emphasizes relevant data bins