Troubles / ~~xx~~ limitations of Parton Showers

~~Tha~~

The Parton Shower is based on the soft / collinear approximation ⇒ therefore they do not describe harder radiation correctly. The problem is that neither of the codes give a warning that ~~#~~ the generated events are not applicable.

IT'S UP TO THE USER TO DECIDE IF YOUR ARE USING THE SOFT-COLLINEAR APPROXIMATION OUTSIDE ITS RANGE OF VALIDITY.

It's a common mistake to abuse of this freedom

How to improve the Parton Shower.

To go beyond the collinear approximation

⇒ use exact matrix elements of higher than leading

two strategies:

1. Higher multiplicity matrix element corrections

→ tree level, CKKW, MLM

2. NLO ~~with~~ matched to the Parton Shower.

→ tree level + loop, MC@NLO, ~~[illegible]~~ POWHEG.

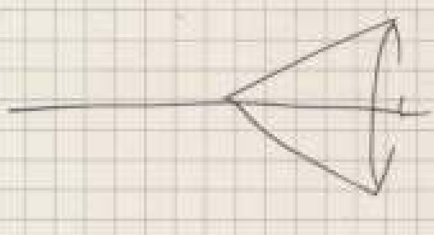
Higher Multiplicity Matrix Element corrections

Key Idea: Compute exactly higher multiplicity matrix elements and overcome <u>double counting</u> and <u>divergences</u>.

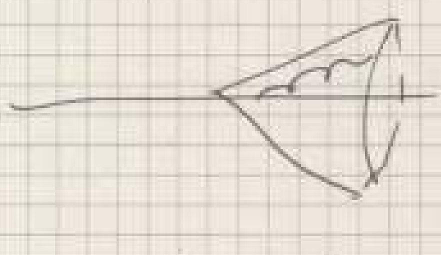
e.g. W-production:

Well-separated partons should evolve into ~~[illegible]~~ large p_T, well-separated jets.

A very <u>naive</u> solution would be to use some observable that decides if two partons are close or not. If they are not close: use matrix element. If they are close: use parton-shower.



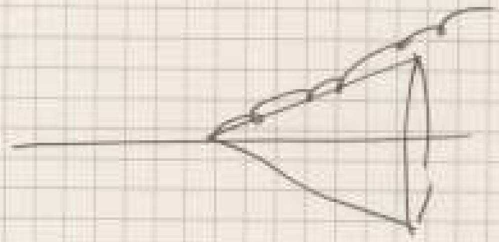
Simplest idea: Use a cone around a parton.



If gluon is inside the cone: Use parton shower to describe the radiation.

If radiation is outside the cone: Use the matrix element to

PROBLEMS WITH NAIVE ~~[illegible]~~ METHOD



How about this one?

1. Partons from far away can re-enter the cone due to more radiation

2. Relative weight of ME's with different multiplicities unspecified.

   * remember that the evolution from the hard-scale to the hadronization scale is unitary
     → P.S. do not change the normalization

   * Also higher ~~order~~ multiplicity ~~and~~ ME's ~~[illegible]~~ give a contribution to the total rate, strongly dependent on the size of the cone (in this example)

3. the final event sample should be <u>independent</u> of the cone size, jet algorithm, etc. How to be sure ~~th~~ to have that in this naive approach.

<u>Solutions</u>: two solutions (that are often used) exist:

CKKW (Catani, Krauss, Kuhn, Webber)

MLM (Mangano)

Definition of "parton distance"

First, instead of using cones, we need something more advanced as parton distance. Conveniently we use a jet algorithm (which has nothing to do with the analysis or jets as such). CKKW uses the k_T algorithm ~~with~~ where:

$d_i = p_{T_i}^2$ (parton - beam distance)

$d_{ij} = \min(p_{T_i}^2, p_{T_j}^2)\, R_{ij}^2$ (parton - parton distance)

$R_{ij}^2 = (\varphi_i - \varphi_j)^2 + (\eta_i - \eta_j)^2$

d_{stop} is the stopping scale (order 10 - 20 GeV or so).

⇒ Two partons are close two eachother if

$$d_{ij} < d_{stop}$$

or a parton is close to the beam if

$$d_i < d_{stop}$$

⇒ If partons are close they can be combined into a pseudo-parton

⇒ Iterate the process until all partons and pseudo-partons left are far apart.

CKKW – an algorithm

Having this definition of distance and a way of combining partons we can have a look to the CKKW prescription –

- We are interested in ~~$pp \to$~~ $p_1 + p_2 \to X$ + many jets

1. Compute the probabilities : $P_n^{(0)} = \sigma_n^{(0)} \Big/ \sum_{i=1}^{N} \sigma_i^{(0)}$

   tree-level n-jet cross-section defined with a d_{stop}. (Use $\alpha_s = \alpha_s(d_{stop})$)

2. Choose a multiplicity $0 \leq \bar{n} \leq N$ with probability $P_{\bar{n}}^{(0)}$.

3. Use the matrix elements $M(p_1 + p_2 \to X + \bar{n}$ partons$)$ to generate an $X + \bar{n}$ partons kinematic configurations

   ⇒ We now have an $\bar{n}$ partons unweighted event

4. Cluster the $\bar{n}$ parton using the k_T-algorithm, and find the <u>nodal values</u> ~~α_s~~ $d_1 > d_2 > \dots d_{\bar{n}} > d_{stop}$, at which 1, 2, .., n jets are ~~just to~~ resolved

   ⇒ the $\bar{n}$-parton configuration can now be depicted as a branching tree, with successive ~~bar~~ branchings at scales d_i

5. Apply the reweighting factor : $\dfrac{\alpha_s(d_1)\,\alpha_s(d_2)\dots\alpha_s(d_{\bar{n}})}{(\alpha_s(d_{stop}))^{\bar{n}}} \leq 1$

   ⇒ Had we known the branching tree (for the parton shower) we should have computed with these couplings

6. Apply the Sudakov Form factor : $\dfrac{\Delta(d_{stop}, d_i)}{\Delta(d_{stop}, d_j)}$ to each line of the branching tree

7. Unweight again the hard configuration, (i.e. accept it if the product of coupling and Sudakov reweighting factors is larger than a random number. Otherwise start at 2)

8. The accepted configuration is the initial condition for the parton shower. Branchings $a \to bc$ in the shower <u>must be vetoed</u> if $d_{bc} > d_{stop}$

   ⇒ When an emission is ~~vetoed~~ vetoed it does not take place, but the shower scale for the next branching is ...

CKKW - in words

- A jet clustering algorithm is used to separate the ME-dominated from the PS-dominated phase-space regions

- In the ME-dominated regions the ME's are correct as if they were generated (kinematically) by the P.S.
  - ⇒ The Sudakov Factors make sure the the parton shower would not emit extra partons w.r.t. those entering in the M.E's

- In the PS dominated regions the P.S. does its job, but it's prevented (owing to the veto) from emitting large P_T, well-separated partons.

⇒ For IR-safe observables the final result should be (fairly) independent ~~of~~ of the jet clustering algorithm and of d_{stop}

Accuracy in CKKW

- ~~Only~~ A formal statement has only been give for jet ~~obs~~ observables in e^+e^- collisions, but it's generally assumed ~~that~~ to be correct also for hadronic observables.

⇒ The separation of the ME- and PS-dominated regions introduces a dependence on the d_{stop} in the n-jet cross-section:

$$\sigma_n \sim \alpha_s^{n-2} \sum_k a_k \alpha_s^k \log^{2k}\left(\frac{d_{stop}}{s}\right)$$

⇒ The CKKW procedure reduces this to

$$\sigma_n \sim \alpha_s^{n-2} \left(\left(\frac{d_{stop}}{s}\right)^a + \sum_k b_k \alpha_s^k \log^{2k-2} \frac{d_{stop}}{s} \right)$$

Ie. It's cancelled at NLL accuracy.

⇒ Is this good enough ?

→ <u>slides</u>

Concluding remarks on CKKW

CKKW is an interpolation procedure between a Parton shower and the matrix elements. It defines a framework, but there is a lot of freedom left

→ Can be used to reduce unphysical biases on observables

* Clustering algorithm and momentum-recombination scheme
* Sudakov definitions
* Scale choices
* Corrections due to $N < \infty$ (highest multiplicity M.E.)

! Never forget that the d_{stop} dependence can be reduced, but not ~~it~~ eliminated.

→ Make sure, before starting extensive physics searches, that d_{stop} is properly chosen and biases are small.

MLM matching

proposed by ~~Magano~~ available in Alpgen and MadGraph
(Mangano)

1. Generate hard unweighted events with the ME's, imposing

Alpgen: $E_T > E_T^{min}$, $R_{ij} > R_{min}$ (no jet algo needed at this step)
Madgraph: d_{stop}

2. Define a similar branching structure as in CKKW, but based on (leading) color flows from M.E. calculations.

3. reweight α_s at the nodal values found in 2),
(Do not reweight with Sudakov form factors)

4. Shower the event, without applying any veto.
When done, find all jets using a cone algorithm (Alpgen) or k_T algorithm (MadGraph) (with (E_T^{min} , R_{min}))
→ using d_{stop}

5. Require jets to be matched to hard partons

⇒ Events with more jets than hard partons are rejected (except for highest multiplicity M.E's)

<u>Note</u>: the jet-algorithm used here has nothing to do with the one used in the analysis ⇒ it's only used to generate a set of events that is consistent. ~~and~~

→ Shower-KT

Summary on Matrix Element Corrections

* Various approaches on the market that improve the Parton Shower using higher-multiplicity Matrix Elements.

* Existing approaches are robust and lead to tolerably small dependence on unphysical parameters ⇒ if they are chosen reasonably.

* There are small discrepancies among the different approaches

  ⇒ there is a lot of flexibility in implementation details

* Tuning to data is strongly recommended, and also necessary to figure out overall normalization

  ⇒ These are LO computations

* Matching parameter systematics must be assessed.

* CKKW is based on reweighting, MLM on rejection of events