Flavoured jet algorithms at the LHC

Ludovic Scyboz

Royal Society Research Grant (RP/R1/180112)

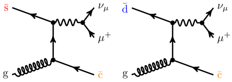
Durham, UK
September 7$^{\text{th}}$ 2023

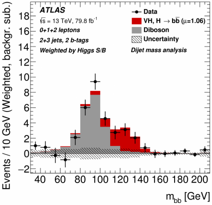
- Jets are a crucial component of analyses at the LHC:
  - *Precision measurements*: fits of α_s, jet substructure, ...
  - *New phenomena*: heavy-particle decays (e.g. boosted topologies)
  - *"Background"*: QCD multijets, underlying event, ...

$\rightsquigarrow$ one is often interested in the kinematics of the jets (e.g. the **radiation intensity** in phase-space) see Mathieu's talk on Wed

- Jets are a crucial component of analyses at the LHC:
  - *Precision measurements*: fits of α_s, jet substructure, ...
  - *New phenomena*: heavy-particle decays (e.g. boosted topologies)
  - *"Background"*: QCD multijets, underlying event, ...

  $\rightsquigarrow$ one is often interested in the kinematics of the jets (e.g. the **radiation intensity** in phase-space)
- In many cases, one also wants to know/discriminate the **flavour** of the originating parton
  - *Higgs*: Yukawa couplings, hadronic decays
  - *PDF's*: W/Z + heavy flavour
  - *Top physics*
  - *Jet substructure*





[ATLAS 1808.08238]

Jet definition

A jet is defined through the application of an **algorithm** to a list of input particles:

Jet definition

A jet is defined through the application of an **algorithm** to a list of input particles:

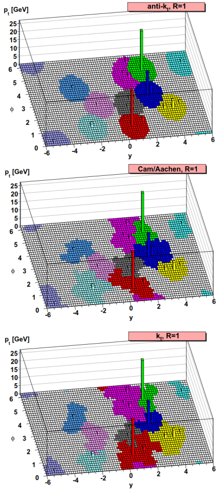
Generalised-k_t family
($p = -1$: anti-k_t, $p = 0$: C/A, $p = 1$: k_t)

- Find the smallest distance among:

$$d_{ij} = \min(p_{t,i}^{2p}, p_{t,j}^{2p}) \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{i,B} = p_{t,i}^{2p}$$

- If d_{ij}, recombine i and j into a single pseudojet
- If $d_{i,B}$, pseudojet i is declared a jet and removed from the list

Repeat until the list is empty & accept jets above $p_{t,\min}$

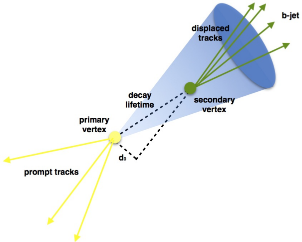


Experimental definition of flavour in jets

- Construct standard anti-k_t jets (without any flavour input)
- A jet j is b-tagged if there is at least one ghost-associated B-hadron with

(for ATLAS/LHCb) $$p_{t,B} > p_{T,\text{cut}}$$

$$\Delta R(j,B) < R_{\text{cut}}$$

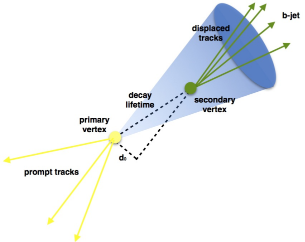


Experimental definition of flavour in jets

- Construct standard anti-k_t jets (without any flavour input)
- A jet j is b-tagged if there is at least one ghost-associated B-hadron with

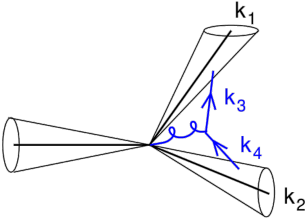
(for ATLAS/LHCb) $p_{t,B} > p_{T,\mathrm{cut}}$

$$\Delta R(j, B) < R_{\mathrm{cut}}$$



this definition is infrared and collinear (IRC) unsafe
(divergent for $m_b = 0$, or logarithms $\ln \frac{m_b}{p_t}$)

Flavour IRC (un-)safety at NNLO



Generalised-k_t + flavour recombination
$(p_i, f_i) + (p_j, f_j) \to (p_i + p_j, f_i + f_j)$

- At NLO: ✓
- At NNLO: IR unsafe, terms $\alpha_s^2 \ln \frac{m_b}{p_t}$

From 2006, 2022–2023: many proposals for flavoured jet algorithms:

- Banfi, Salam, Zanderighi '06 .... flavour-k_t
- Caletti, Larkoski, Marzani, Reichelt '22 .... SoftDrop [up to NNLO]
- Czakon, Mitov, Poncelet '22 .... flavour anti-k_t
- Huss, Gauld, Stagnitto '22 .... flavour dressing
- Caola, Grabarczyk, Hutt, Salam, Scyboz, Thaler '23 .... IFN

Infrared safe definition of jet flavor

Andrea Banfi (Cambridge U., DAMTP and Cambridge U. and Milan Bicocca U. and INFN, Milan), Gavin P. Salam (Paris, LPTHE), Giulia Zanderighi (Fermilab and CERN)
Jan, 2006

- Modification of k_t $(p = 1)$ clustering distance for flavoured pseudojets

$$d_{ij} = \frac{\Delta R_{ij}^2}{R^2} \times \begin{cases} \max(p_{t,i}, p_{t,j})^{\alpha} \min(p_{t,i}, p_{t,j})^{2-\alpha}\,, & \text{softer of } i, j \text{ flavoured} \\ \min(p_{t,i}^2, p_{t,j}^2)\,, & \text{else} \end{cases}$$

(and $d_{iB} \to d_{iB}(\eta_i)$ for hadron collisions)

Note:

- Seemed to solve the IRC unsafety from a theoretical viewpoint
- Not overly popular with experimentalists

Caveat on IRC safety

$d_{ij} \to d_{ij,\Omega}$

The flavour anti-k_t algorithm

Infrared-safe flavoured anti-k_T jets

Michal Czakon (Aachen, Tech. Hochsch.), Alexander Mitov (Cambridge U.), Rene Poncelet (Cambridge U.)
May 24, 2022

- How to get kinematics that are closer to anti-k_t (and IRC-safe flavour)?
- Similarly modify the anti-k_t ($p = -1$) clustering distance for flavoured pseudojets

$$d_{ij} \to d_{ij}^{\text{akt}} \times \begin{cases} \mathcal{S}_{ij}\,, & i,j \text{ oppositely flavoured} \\ 1\,, & \text{else} \end{cases}$$

with $\mathcal{S}_{ij} = 1 - \Theta(1-\kappa_{ij})\cos\left(\frac{\pi}{2}\kappa_{ij}\right)$, and $\kappa_{ij} = \frac{1}{a}\frac{p_{t,i}^2+p_{t,j}^2}{2p_{t,\max}^2}$

Note:

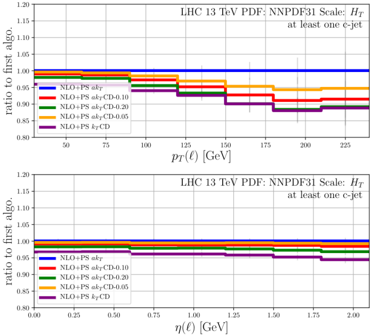
- if one of i or j is hard, κ_{ij} is large
- if both are soft, $\mathcal{S}_{ij} \propto \kappa_{ij}^2$ is suppressed

Caveat on IRC safety

$\mathcal{S}_{ij} \to \mathcal{S}_{ij,\Omega}$

Flavour anti-k_t in practice

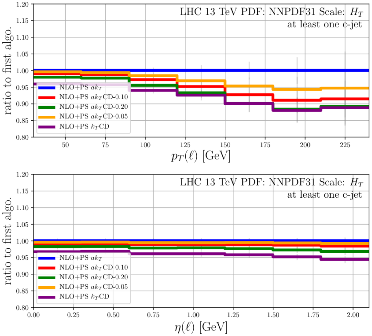
$W + c$ [2212.00467]



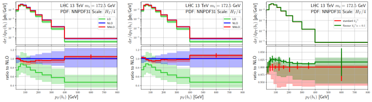
- kinematics close to anti-k_t (e.g. 5% in $p_{t,\ell}$ for $a = 0.05$)

Flavour anti-k_t in practice

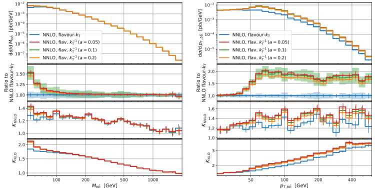
$W + c$ [2212.00467]



$t\bar{t}$ (and also $Z + b$) [2205.11879]



$Wb\bar{b}$ [2209.03280]



- kinematics close to anti-k_t (e.g. 5% in $p_{t,\ell}$ for $a = 0.05$)

see Bayu's talk on Wed

The flavour dressing algorithm

Flavor Identification of Reconstructed Hadronic Jets

Rhorry Gauld (U. Bonn, Phys. Inst., BCTP and Munich, Max Planck Inst.), Alexander Huss (CERN), Giovanni Stagnitto (Zurich U.)
Aug 23, 2022

1. Cluster particles with anti-k_t $\rightarrow$ jets $\{j_1, \ldots, j_n\}$
2. From input particles $\{f_1, \ldots, f_k\}$, create a list of flavoured "clusters" $\{\hat{f}_1, \ldots, \hat{f}_m\}$
   - by dressing flavoured particles with unflavoured collinear radiation

$$\frac{\min(p_{t,i}, p_{t,j})}{p_{t,i} + p_{t,j}} > z_{\text{cut}} \left(\frac{\Delta R_{ij}}{R_{\text{cut}}} \right)^{\beta}$$

3. Assign the identified clusters $\{\hat{f}_1, \ldots, \hat{f}_m\}$ to jets $\{j_1, \ldots, j_n\}$

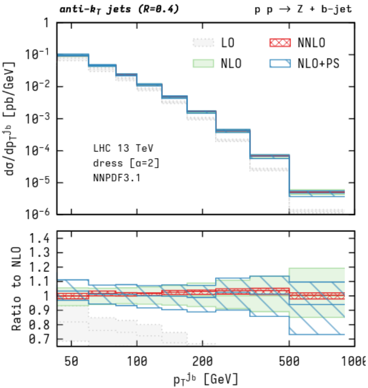
$$d^{\text{flav-}k_t}_{\hat{f}_i \hat{f}_k}, \qquad d^{\text{flav-}k_t}_{\hat{f}_i j_k}, \qquad d^{\text{flav-}k_t}_{\hat{f}_i B_\pm}$$

   - Originally: $d^{\text{flav-}k_t}_{\hat{f}_i \hat{f}_k}$: annihilate $\rightarrow$ recombine
   - Similar to flav-k_t caveat $\rightarrow$ flav-$k_{t,\Omega}$ $(\alpha\beta < 2)$, or Jade-like $(\beta < 2)$

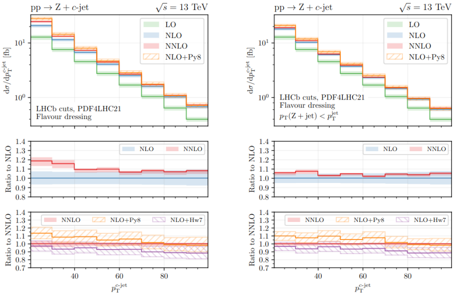
Caveat on IRC safety

Flavour dressing in practice

$Z + b$ [2208.11138]



$Z + c$ [2302.12844]



. . .

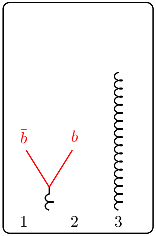
Interleaved Flavour Neutralisation (IFN)

Flavoured jets with exact anti-k_t kinematics and tests of infrared and collinear safety

Fabrizio Caola (Oxford U., Theor. Phys.), Radoslaw Grabarczyk (Oxford U., Theor. Phys.), Maxwell L. Hutt (Oxford U., Theor. Phys. and Imperial Coll., London), Gavin P. Salam (Oxford U., Theor. Phys. and Oxford U.), Ludovic Scyboz (Oxford U., Theor. Phys.), Jesse Thaler (MIT, Cambridge, CTP)
Jun 12, 2023

- Cluster particles with a generalised-k_t algorithm

$$d_{ij} = \min\left(p_{ti}^{2p}, p_{tj}^{2p}\right)\frac{\Delta R_{ij}^2}{R^2} \qquad d_{iB} = p_{ti}^{2p}$$



Interleaved Flavour Neutralisation (IFN)

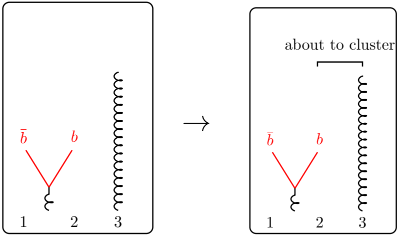
Flavoured jets with exact anti-k_t kinematics and tests of infrared and collinear safety

Fabrizio Caola (Oxford U., Theor. Phys.), Radoslaw Grabarczyk (Oxford U., Theor. Phys.), Maxwell L. Hutt (Oxford U., Theor. Phys. and Imperial Coll., London), Gavin P. Salam (Oxford U., Theor. Phys. and Oxford U.), Ludovic Scyboz (Oxford U., Theor. Phys.), Jesse Thaler (MIT, Cambridge, CTP)

Jun 12, 2023

- Cluster particles with a generalised-k_t algorithm

$$d_{ij} = \min\left(p_{ti}^{2p}, p_{tj}^{2p}\right)\frac{\Delta R_{ij}^2}{R^2} \qquad d_{iB} = p_{ti}^{2p}$$



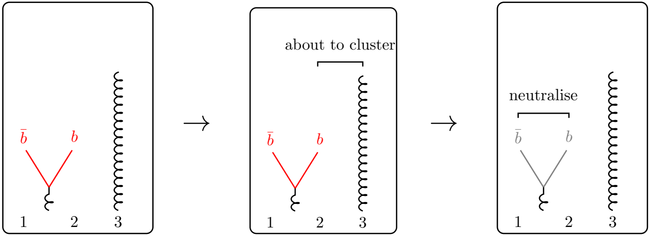
Interleaved Flavour Neutralisation (IFN)

Flavoured jets with exact anti-k_t kinematics and tests of infrared and collinear safety

Fabrizio Caola (Oxford U., Theor. Phys.), Radoslaw Grabarczyk (Oxford U., Theor. Phys.), Maxwell L. Hutt (Oxford U., Theor. Phys. and Imperial Coll., London), Gavin P. Salam (Oxford U., Theor. Phys. and Oxford U.), Ludovic Scyboz (Oxford U., Theor. Phys.), Jesse Thaler (MIT, Cambridge, CTP)
Jun 12, 2023

- Cluster particles with a generalised-k_t algorithm

$$d_{ij} = \min\left(p_{ti}^{2p}, p_{tj}^{2p}\right)\frac{\Delta R_{ij}^2}{R^2} \qquad d_{iB} = p_{ti}^{2p}$$



neutralise $\equiv$ remove the (opposite) flavour of both 1 & 2 while maintaining kinematics

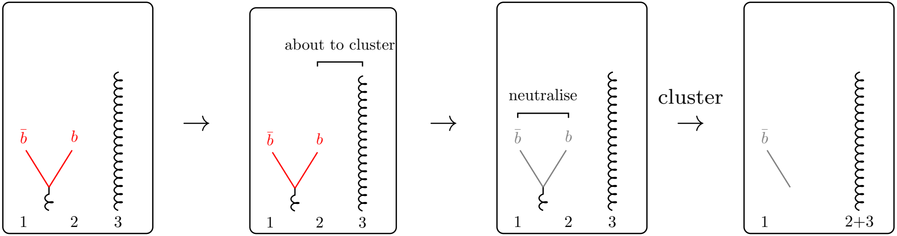
Flavoured jets with exact anti-k_t kinematics and tests of infrared and collinear safety

Fabrizio Caola (Oxford U., Theor. Phys.), Radoslaw Grabarczyk (Oxford U., Theor. Phys.), Maxwell L. Hutt (Oxford U., Theor. Phys. and Imperial Coll., London), Gavin P. Salam (Oxford U., Theor. Phys. and Oxford U.), Ludovic Scyboz (Oxford U., Theor. Phys.), Jesse Thaler (MIT, Cambridge, CTP)

Jun 12, 2023

- Cluster particles with a generalised-k_t algorithm

$$d_{ij} = \min\left(p_{ti}^{2p}, p_{tj}^{2p}\right)\frac{\Delta R_{ij}^2}{R^2} \qquad d_{iB} = p_{ti}^{2p}$$



neutralise $\equiv$ remove the (opposite) flavour of both 1 & 2 while maintaining kinematics

Interleaved Flavour Neutralisation (IFN)

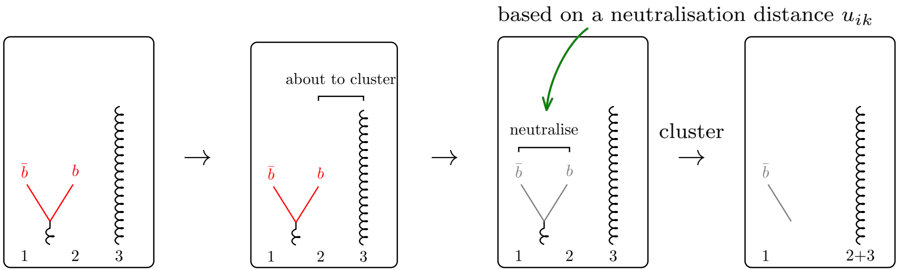
Flavoured jets with exact anti-k_t kinematics and tests of infrared and collinear safety

Fabrizio Caola (Oxford U., Theor. Phys.), Radoslaw Grabarczyk (Oxford U., Theor. Phys.), Maxwell L. Hutt (Oxford U., Theor. Phys. and Imperial Coll., London), Gavin P. Salam (Oxford U., Theor. Phys. and Oxford U.), Ludovic Scyboz (Oxford U., Theor. Phys.), Jesse Thaler (MIT, Cambridge, CTP)

Jun 12, 2023

- Cluster particles with a generalised-k_t algorithm

$$d_{ij} = \min\left(p_{ti}^{2p}, p_{tj}^{2p}\right) \frac{\Delta R_{ij}^2}{R^2} \qquad d_{iB} = p_{ti}^{2p}$$



neutralise ≡ remove the (opposite) flavour of both 1 & 2 while maintaining kinematics

Interleaved Flavour Neutralisation (IFN)

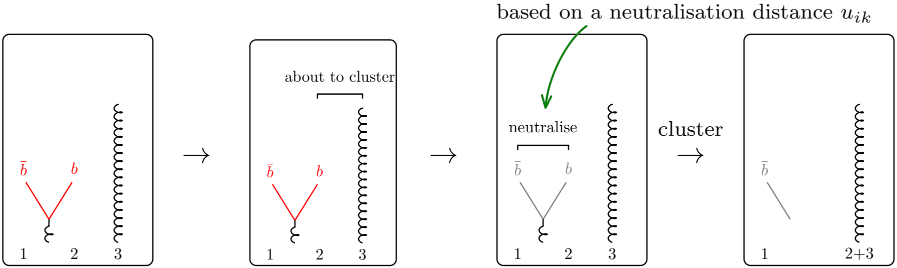
Flavoured jets with exact anti-k_t kinematics and tests of infrared and collinear safety

Fabrizio Caola (Oxford U., Theor. Phys.), Radoslaw Grabarczyk (Oxford U., Theor. Phys.), Maxwell L. Hutt (Oxford U., Theor. Phys. and Imperial Coll., London), Gavin P. Salam (Oxford U., Theor. Phys. and Oxford U.), Ludovic Scyboz (Oxford U., Theor. Phys.), Jesse Thaler (MIT, Cambridge, CTP)

Jun 12, 2023

- Cluster particles with a generalised-k_t algorithm

$$d_{ij} = \min\left(p_{ti}^{2p}, p_{tj}^{2p}\right)\frac{\Delta R_{ij}^2}{R^2} \qquad d_{iB} = p_{ti}^{2p}$$



neutralise ≡ remove the (opposite) flavour of both 1 & 2 while maintaining kinematics

need to apply this recursively

The choice of the neutralisation distance u_{ik}

- Generic form (with parameters α and ω):

$$u_{ik} = \max\left(p_{ti}, p_{tk}\right)^{\alpha} \min\left(p_{ti}, p_{tk}\right)^{2-\alpha} \cdot \Omega_{ik}^2$$

$$\Omega_{ik}^2 = 2\left[\frac{1}{\omega^2}\left(\cosh(\omega\Delta y_{ik}) - 1\right) - \left(\cos\Delta\phi_{ik} - 1\right)\right]$$

- Generic form (with parameters α and ω):

$$u_{ik} = \max\left(p_{ti}, p_{tk}\right)^{\alpha} \min\left(p_{ti}, p_{tk}\right)^{2-\alpha} \cdot \Omega_{ik}^2$$

$$\Omega_{ik}^2 = 2\left[\frac{1}{\omega^2}\left(\cosh(\omega \Delta y_{ik}) - 1\right) - \left(\cos \Delta\phi_{ik} - 1\right)\right]$$

(similar to alternative proposal for ΔR^2 by [Catani et al. '93]!)

- Identical to flavour-k_t distance, except for angular part:
  - $\rightarrow \Delta R_{ik}^2$ for any ω when $\Delta R_{ik} \rightarrow 0$
  - $\rightarrow \exp\left(\omega \Delta y_{ik}\right)$ for $\Delta y_{ik} \gg 1$

eliminates divergence from interplay between ISR collinear and soft, large-angle flavour

The choice of the neutralisation distance u_{ik}

- Generic form (with parameters α and ω):

$$u_{ik} = \max\left(p_{ti}, p_{tk}\right)^{\alpha} \min\left(p_{ti}, p_{tk}\right)^{2-\alpha} \cdot \Omega_{ik}^2$$

$$\Omega_{ik}^2 = 2\left[\frac{1}{\omega^2}\left(\cosh(\omega \Delta y_{ik}) - 1\right) - \left(\cos \Delta\phi_{ik} - 1\right)\right]$$

(similar to alternative proposal for ΔR^2 by [Catani et al. '93]!)

- Identical to flavour-k_t distance, except for angular part:
  - $\to \Delta R_{ik}^2$ for any ω when $\Delta R_{ik} \to 0$
  - $\to \exp\left(\omega \Delta y_{ik}\right)$ for $\Delta y_{ik} \gg 1$

- In the following:
  - $\alpha = 1,\ \omega = 2$
  - $\alpha = 2,\ \omega = 1$

eliminates divergence from interplay between ISR collinear and soft, large-angle flavour

The choice of the neutralisation distance u_{ik}

- Generic form (with parameters α and ω):

$$u_{ik} = \max\left(p_{ti}, p_{tk}\right)^{\alpha} \min\left(p_{ti}, p_{tk}\right)^{2-\alpha} \cdot \Omega_{ik}^2$$

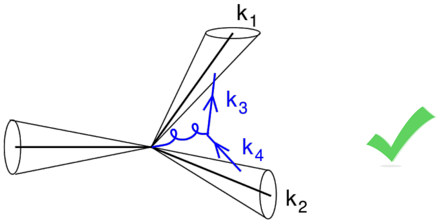
$$\Omega_{ik}^2 = 2\left[\frac{1}{\omega^2}\left(\cosh(\omega \Delta y_{ik}) - 1\right) - \left(\cos\Delta\phi_{ik} - 1\right)\right]$$

(similar to alternative proposal for ΔR^2 by [Catani et al. '93]!)

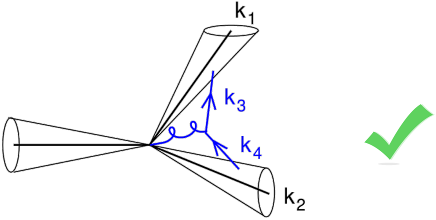
- Identical to flavour-k_t distance, except for angular part:
  - $\to \Delta R_{ik}^2$ for any ω when $\Delta R_{ik} \to 0$
  - $\to \exp\left(\omega \Delta y_{ik}\right)$ for $\Delta y_{ik} \gg 1$

- In the following:
  - $\alpha = 1,\ \omega = 2$
  - $\alpha = 2,\ \omega = 1$

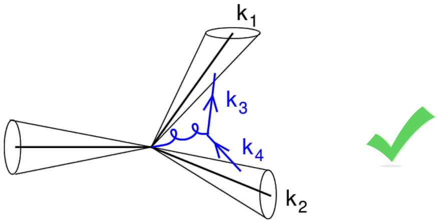
Need $\alpha + \omega > 2$ from IRC safety too



- All of the above algorithms take care of the “original” issue with soft, large-angle gluon splittings at NNLO



- All of the above algorithms take care of the “original” issue with soft, large-angle gluon splittings at NNLO
- Safety **at any order**, for generic configurations?



- All of the above algorithms take care of the “original” issue with soft, large-angle gluon splittings at NNLO
- Safety **at any order**, for generic configurations?
- Complicated (nested) structure of soft/collinear divergences → need a systematic framework to check for bad behaviour

IRC-safety tests: results

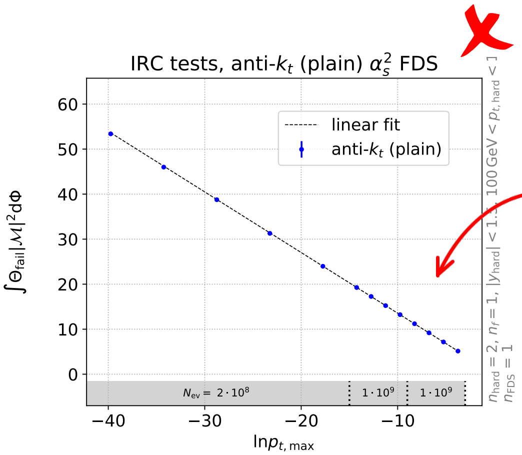
	anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
FDS						
IDS						

Abbr.

FDS = final-state double-soft
IDS = initial-state double-soft
FC = final-state hard-collinear
IC = initial-state hard-collinear

IRC-safety tests: results

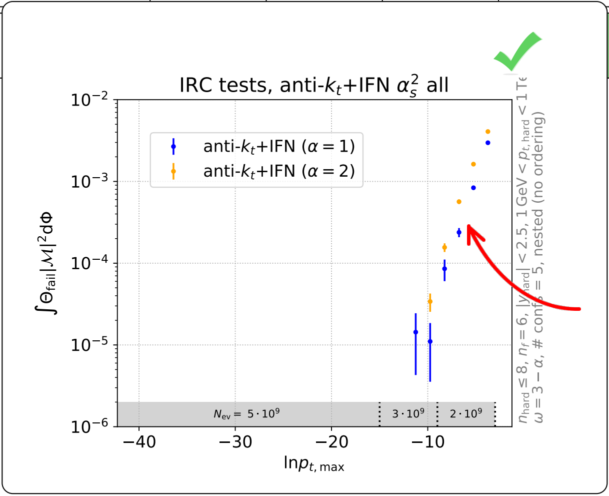
	anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)



$\mathcal{O}(\alpha_s^2 L)$ divergence

IRC-safety tests: results

	anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)



$\rightarrow$ 0 as a power law
$(\sim |p_{t,\max}|^n)$

IRC-safety tests: results

		anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
α_s^2	FDS						
	IDS						
	FC×IC						
	FC×FC						
	IC×IC						

IRC-safety tests: results

		anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
α_s^2	FDS						
	IDS						
	FC×IC						
	FC×FC						
	IC×IC						
α_s^3			marginal	+IC×FDS			

IRC-safety tests: results

		anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
α_s^2	FDS						
	IDS						
	FC×IC						
	FC×FC						
	IC×IC						
α_s^3			marginal	+IC×FDS			
α_s^4					+IDS×FDS		

Fix $d_{ij}^{\text{flav}-k_t} \to d_{ij,\Omega}^{\text{flav}-k_t} = d_{ij}^{\text{flav}-k_t} \frac{\Omega_{ij}^2}{\Delta R_{ij}^2}$

IRC-safety tests: results

		anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
α_s^2	FDS						
	IDS						
	FC×IC						
	FC×FC						
	IC×IC						
α_s^3			marginal	+IC×FDS			
α_s^4					+IDS×FDS		
α_s^5							
α_s^6							

Fix by replacing angular factor (for flavoured pairs)

$$\mathcal{S}_{ij} \to \bar{\mathcal{S}}_{ij} = \mathcal{S}_{ij} \frac{\Omega_{ij}^2}{\Delta R_{ij}^2} \quad \text{with } \omega > 1$$

IRC-safety tests: results

		anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
α_s^2	FDS						
	IDS						
	FC×IC						
	FC×FC						
	IC×IC						
α_s^3			marginal	+IC×FDS			
α_s^4					+IDS×FDS		
α_s^5							
α_s^6							

Flav-k_t fix ($\alpha\beta < 2$) / Jade ($\beta < 2$), dressing modification

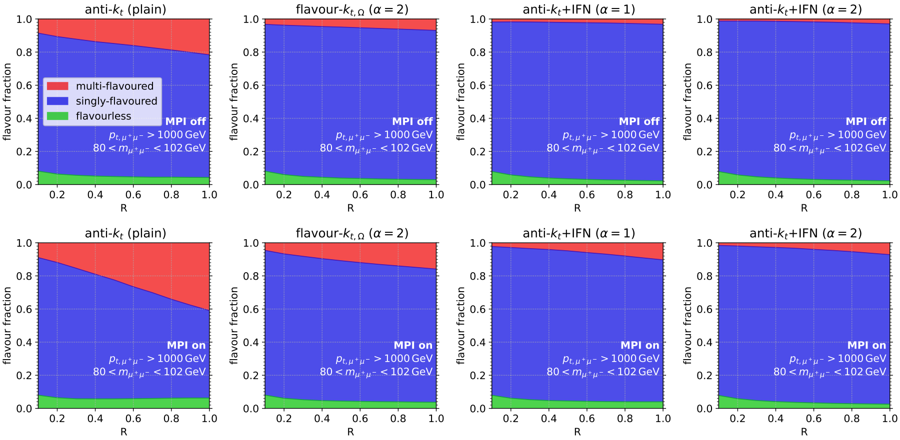
IRC-safety tests: results

		anti-k_t	flav-k_t	flav-akt	flav-dress	IFN(2,1)	IFN(1,2)
α_s^2	FDS						
	IDS						
	FC×IC						
	FC×FC						
	IC×IC						
α_s^3							
α_s^4							
α_s^5							
α_s^6							
					?		

$\rightarrow$ potentially all IRC-safe

IFN in practice: $pp \to Z + q$

- "Mis"-tag rate (as a function of ΔR)



Flavour wish-list

We'd like to have an algorithm that:

1. is **IRC-safe** at all orders
   - Flavour-k_t attempt
   - Flav-$k_{t,\Omega}$, flav-a$k_{t,\Omega}$, flav-dress$_\Omega$ + collinear fix, IFN: ✓

We'd like to have an algorithm that:

1. is **IRC-safe** at all orders
   - Flavour-k_t attempt
   - Flav-$k_{t,\Omega}$, flav-a$k_{t,\Omega}$, flav-dress$_\Omega$ + collinear fix, IFN: ✓
2. reproduces anti-k_t kinematics
   - flav-$k_{t,\Omega}$: ✗, flav-a$k_{t,\Omega}$ $\sim$, flav-dress$_\Omega$, IFN: ✓

We'd like to have an algorithm that:

1. is **IRC-safe** at all orders
   - Flavour-k_t attempt
   - Flav-$k_{t,\Omega}$, flav-a$k_{t,\Omega}$, flav-dress$_\Omega$ + collinear fix, IFN: ✓
2. reproduces anti-k_t kinematics
   - flav-$k_{t,\Omega}$: ✗, flav-a$k_{t,\Omega}$ $\sim$, flav-dress$_\Omega$, IFN: ✓
3. may make flavour information accessible for jet substructure (tracking flavour along the cluster sequence)

- IRC safety with flavour is a difficult problem!
- Multiple recent attempts to define a IRC-safe flavour algorithm with kinematics identical (or close) to anti-k_t
- As the IFN project evolved, it became clear that a set of systematic tests for divergences was needed
- Release of all algorithms (`SDFlavPlugin`, `CMPPlugin`, `GHSAlgo` and `IFNPlugin`) in a common repository:

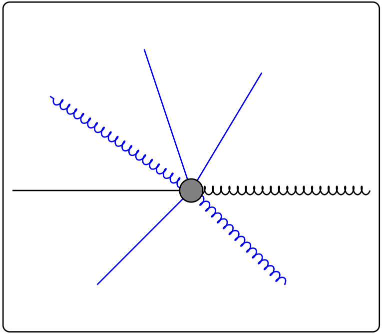
  `https://github.com/jetflav`

- Algorithms may have different properties (various sets of parameters, robustness w.r.t. extra radiation, etc.)
- First comparisons (fixed-order and showered) at Les Houches 2023: on the way

Backup

Systematic IRC-safety tests

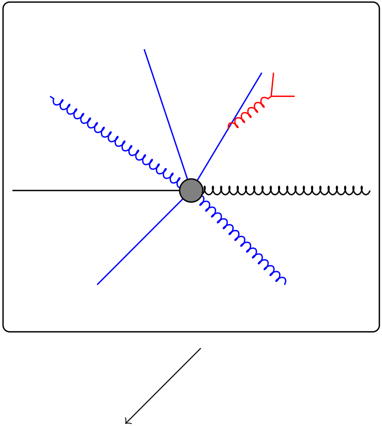
- Implemented such a fixed-order framework:



Cluster "hard" event

Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

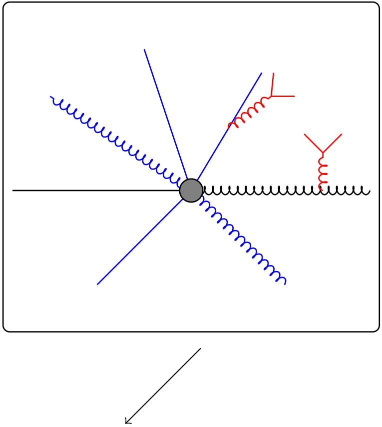
- Implemented such a fixed-order framework:



FDS = FS double-soft

Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

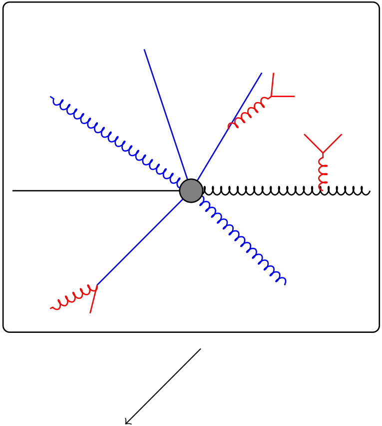
- Implemented such a fixed-order framework:



FDS = FS double-soft
IDS = IS double-soft

Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

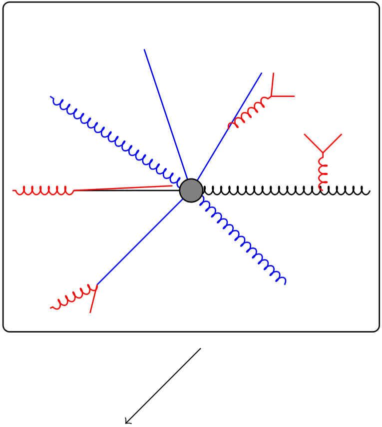
- Implemented such a fixed-order framework:



FDS = FS double-soft
IDS = IS double-soft
FC = final hard-collinear

Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

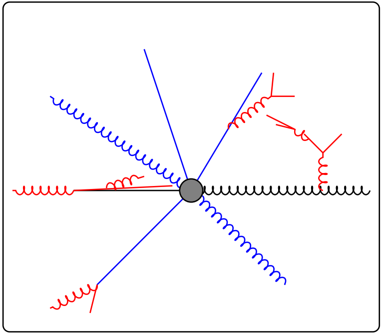
- Implemented such a fixed-order framework:



FDS = FS double-soft
IDS = IS double-soft
FC = final hard-collinear
IC = IS hard-collinear

Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

- Implemented such a fixed-order framework:

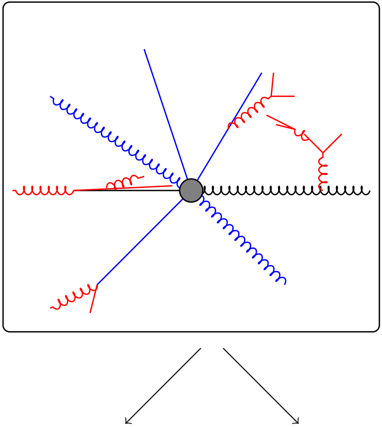


FDS = FS double-soft
IDS = IS double-soft
FC = final hard-collinear
IC = IS hard-collinear

possibly nested

Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

- Implemented such a fixed-order framework:



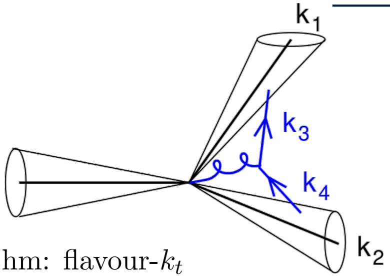
Set of hard jets
$\mathcal{J}_{\text{hard}} = \{(p_1, f_1), ...\}$

$\stackrel{!}{=}$

Set of hard+IRC jets
$\mathcal{J}_{\text{hard+IRC}} = \{(\tilde{p}_1, \tilde{f}_1), ...\}$

First appearance at order $\mathcal{O}(\alpha_s^2)$

- Soft, large-angle gluon splitting to $q\bar{q}$
  → flavour *pollution*



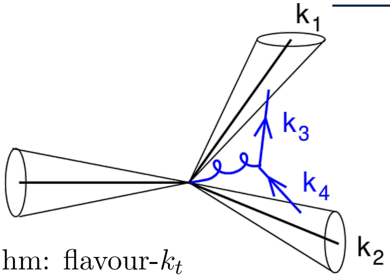
- Proposal for IRC-safe algorithm: flavour-k_t
  [Banfi,Salam,Zanderighi '06]
- More recently: multiple alternative proposals
  - "Practical jet flavour through NNLO"
    [Caletti,Larkoski,Marzani,Reichelt '22]
  - "Infrared-safe flavoured anti-k_t jets"
    [Czakon,Mitov,Poncelet '22]
  - "A dress of flavour to suit any jet"
    [Gauld,Huss,Stagnitto '22]

Different kinematics than anti-k_t

The problem of jet flavour & IRC safety

First appearance at order $\mathcal{O}(\alpha_s^2)$

- Soft, large-angle gluon splitting to $q\bar{q}$
  → flavour *pollution*



- Proposal for IRC-safe algorithm: flavour-k_t
  [Banfi,Salam,Zanderighi '06]
- More recently: multiple alternative proposals
  - "Practical jet flavour through NNLO"
    [Caletti,Larkoski,Marzani,Reichelt '22]
  - "Infrared-safe flavoured anti-k_t jets"
    [Czakon,Mitov,Poncelet '22] "CMP"
  - "A dress of flavour to suit any jet"
    [Gauld,Huss,Stagnitto '22] "GHS"

Different kinematics than anti-k_t

Flavour-k_t

New distances

$$d_{ij} = \max(p_{ti}, p_{tj})^{\alpha} \cdot \min(p_{ti}, p_{tj})^{2-\alpha} \frac{\Delta R_{ij}^2}{R^2}$$

if the softer of i and j is flavoured

... and some new beam distances d_{iB}

CMP

New distances

$$d_{ij} = d_{ij}^{\text{anti}-k_t} \times \mathcal{S}_{ij}$$

$$\mathcal{S}_{ij} = 1 - \Theta(1-\kappa)\cos\left(\frac{\pi}{2}\kappa\right)$$

$$\kappa = \frac{1}{a}\frac{p_{ti}^2 + p_{tj}^2}{2p_{t,\max}^2}$$

if i and j are oppositely flavoured

GHS

1. Cluster with anti-k_t
2. "Accumulation" step with C/A + SoftDrop
3. Flavour "dressing" with flav-k_t distances

$$d_{\hat{f}_i \hat{f}_j},\ d_{\hat{f}_i j_k},\ d_{\hat{f}_i B}$$

... assigns flavour $\hat{f}_i$ to jet j_k if $d_{\hat{f}_i j_k}$ is smaller

Flavour dressing: accumulation

Input particles $f \longrightarrow$ flavoured "clusters" $\{\hat{f}_1, \ldots, \hat{f}_m\}$

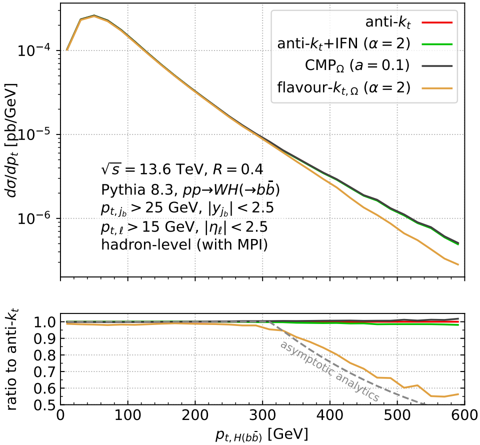
- For distances $d_{ij} = \Delta R_{ij}^2$, repeat (as long as $\min d_{ij} > \Delta R_{\text{cut}}^2$)
  1. if i and j are flavourless, recombine $i + j$
  2. if i and j are flavoured: accumulation is complete and $\hat{f}_i$, $\hat{f}_j$ are added to the list
  3. else (if i or j is flavoured), test a SoftDrop criterion:

$$\frac{\min(p_{t,i}, p_{t,j})}{p_{t,i} + p_{t,j}} > z_{\text{cut}} \left(\frac{\Delta R_{ij}}{R_{\text{cut}}} \right)^{\beta}$$

  - If it passes, recombine $i + j$
  - Otherwise, remove the unflavoured one from the list

$$|\eta_\ell| < 2.5\,, \quad p_{t\ell} > 15\,\text{GeV}$$
$$|\eta_{j_b}| < 2.5\,, \quad p_{tj_b} > 25\,\text{GeV}$$

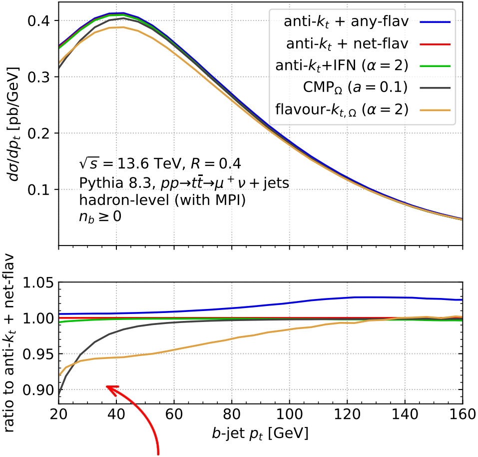
- Standard anti-k_t: one needs $m_b > 0$ to regulate $g \to b\bar{b}$ divergence
- Large differences between anti-k_t and flavour-k_t in e.g. the tail of $p_{t,H}$ (at NNLO: [Behring et al. '20])



Semi-leptonic $t\bar{t}$: b-jet p_t distribution

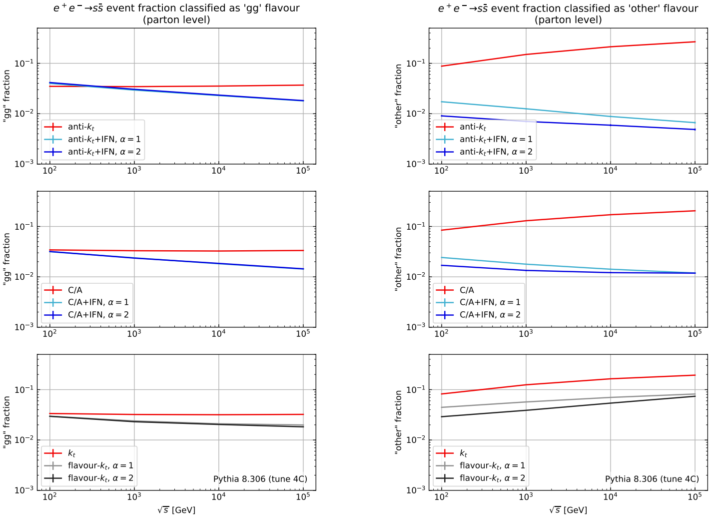
$$p_{t\mu} > 30\,\mathrm{GeV}\,, \quad |\eta_\mu| < 2.4$$
$$p_{t,j_b} > 20\,\mathrm{GeV}\,, \quad p_{t,\mathrm{miss}} > 30\,\mathrm{GeV}$$

- CMP (with $a = 0.1$, corrected) can still differ from exact anti-k_t kinematics
- FN reproduces anti-k_t kinematics instead (as does GHS by construction)



Flavour-k_t, CMP: $\mathcal{O}(10\%)$ disagreement with anti-k_t

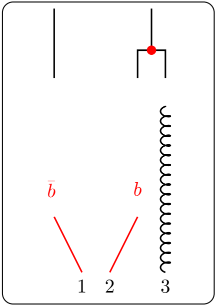
$e^+e^- \to q\bar{q}$ scaling



$i = 2, j = 3$

When pseudojets i and j recombine (where $p_{ti} < p_{tj}$):

1. If i is flavourless, combine i and j and apply the usual flavour summation.



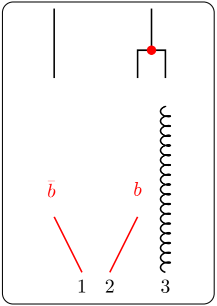
Flavour neutralisation: the "simple" algorithm

When pseudojets i and j recombine (where $p_{ti} < p_{tj}$):

1. If i is flavourless, combine i and j and apply the usual flavour summation.
2. If i is flavoured, identify all pseudojets carrying flavour at this stage (including jets that were declared as *beam jets* earlier). $\rightarrow$ list K.

$i = 2, j = 3$

$K = \{1\}$

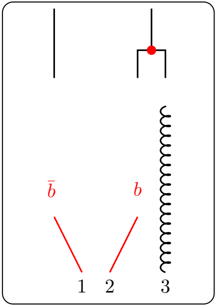


Flavour neutralisation: the "simple" algorithm

$i = 2, j = 3$

$K = \{1\}$

$E = \{2, 3\}$



When pseudojets i and j recombine (where $p_{ti} < p_{tj}$):

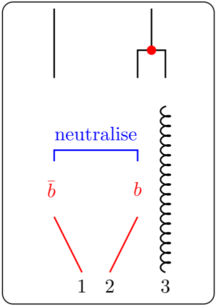
1. If i is flavourless, combine i and j and apply the usual flavour summation.
2. If i is flavoured, identify all pseudojets carrying flavour at this stage (including jets that were declared as *beam jets* earlier). → list K.
   Initialise a set of particles to be excluded from consideration, $E = \{i, j\}$.
3. **Neutralisation step**: $\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{ij}$, neutralise as much flavour in i from k as possible.

Flavour neutralisation: the "simple" algorithm

When pseudojets i and j recombine (where $p_{ti} < p_{tj}$):

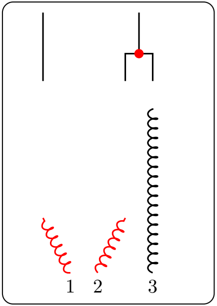
1. If i is flavourless, combine i and j and apply the usual flavour summation.
2. If i is flavoured, identify all pseudojets carrying flavour at this stage (including jets that were declared as *beam jets* earlier). → list K.
   Initialise a set of particles to be excluded from consideration, $E = \{i, j\}$.
3. **Neutralisation step**: $\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{ij}$, neutralise as much flavour in i from k as possible.

$i = 2, j = 3$
$K = \{1\}$
$E = \{2, 3\}$



Flavour neutralisation: the "simple" algorithm

$i = 2, j = 3$
$K = \{1\}$
$E = \{2, 3\}$



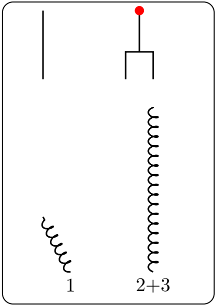
When pseudojets i and j recombine (where $p_{ti} < p_{tj}$):

1. If i is flavourless, combine i and j and apply the usual flavour summation.
2. If i is flavoured, identify all pseudojets carrying flavour at this stage (including jets that were declared as *beam jets* earlier). $\rightarrow$ list K.
   Initialise a set of particles to be excluded from consideration, $E = \{i, j\}$.
3. **Neutralisation step**: $\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{ij}$, neutralise as much flavour in i from k as possible.

$i = 2, j = 3$

$K = \{1\}$

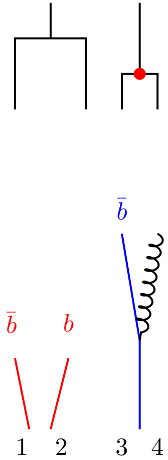
$E = \{2, 3\}$



When pseudojets i and j recombine (where $p_{ti} < p_{tj}$):

1. If i is flavourless, combine i and j and apply the usual flavour summation.
2. If i is flavoured, identify all pseudojets carrying flavour at this stage (including jets that were declared as *beam jets* earlier). → list K.
   Initialise a set of particles to be excluded from consideration, $E = \{i, j\}$.
3. **Neutralisation step**: $\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{ij}$, neutralise as much flavour in i from k as possible.
4. Move on to the next kinematic clustering step.

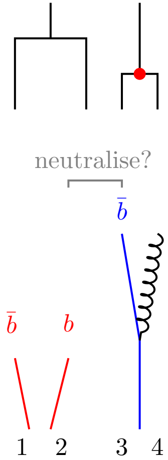
- Recursive version (needed for IRC safety!)



- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

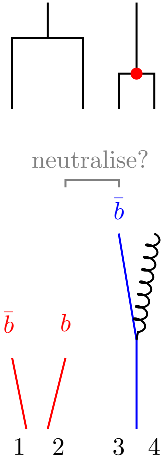
1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.



- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

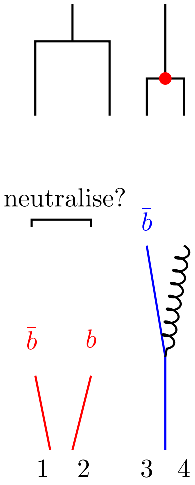
1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.



- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

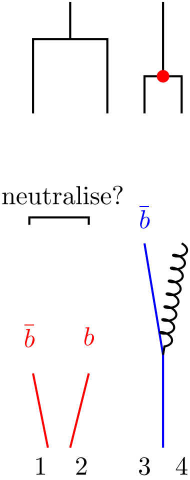
1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.



- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

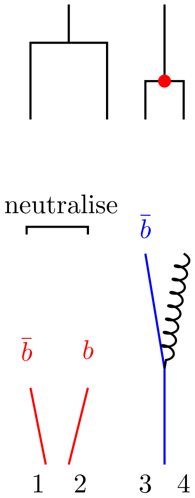
1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.



- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

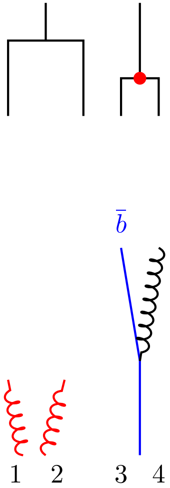
1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.



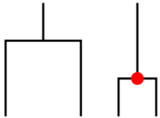
- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.



Flavour neutralisation: the “recursive” algorithm

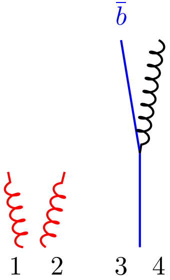


- Recursive version (needed for IRC safety!)

$N(K, E, i, u_{max})$:

$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

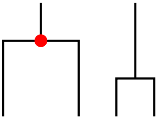
1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.

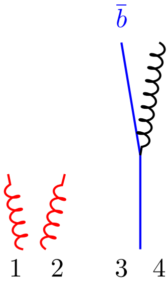


- Recursive version (needed for IRC safety!)

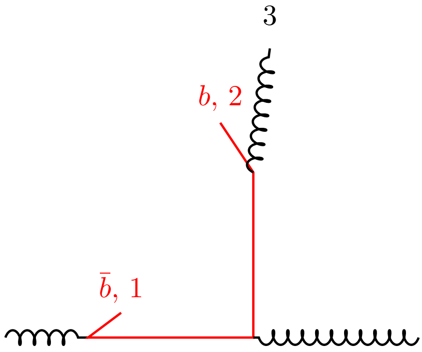
$N(K, E, i, u_{max})$:
$\forall k \in K \setminus E$, in order of decreasing $u_{ik} < u_{\max}$,

1. If k has no flavour that can neutralise i, continue
2. If it does, call $N(K, E \cup \{k\}, k, u_{ik})$
3. Neutralise as much flavour in i as one can with k.
4. If i is now flavourless, stop.
5. Otherwise continue to the next k, or exit if there aren't any left.





IFN: angular parameter $\omega + \alpha > 2$



$$e^{y_1} \sim p_{t,1} \ll p_{t,2} \lesssim p_{t,3} \sim 1$$

- We need to ensure $u_{12} > u_{23}$

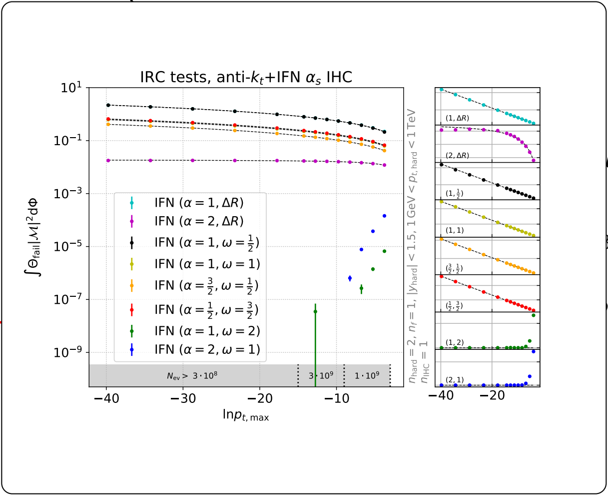
$$p_{t2}^{\alpha} p_{t1}^{2-\alpha} (\frac{p_{t3}}{p_{t1}})^{\omega} > p_{t3}^{\alpha} p_{t2}^{2-\alpha} \Delta R_{23}^2$$

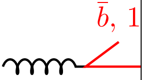
$$\leftrightarrow p_{t3}^{\alpha+\omega} p_{t1}^{2-\alpha-\omega} > p_{t3}^{\alpha} p_{t2}^{2-\alpha} \Delta R_{23}^2$$

$$\leftrightarrow \alpha + \omega > 2$$

IFN: angular parameter $\omega + \alpha > 2$

3





$u_{12} > u_{23}$

$_3 p_{t2}^{2-\alpha} \Delta R_{23}^2$

$_{t2}^{2-\alpha} \Delta R_{23}^2$

$e^{y_1} \sim p_t,$

- Czakon, Mitov & Poncelet (CMP): modification of anti-k_t distance for flavoured pairs

$$d_{ij} = \min\left(p_{ti}^{-2}, p_{tj}^{-2}\right) \frac{\Delta R_{ij}^2}{R^2} \times \mathcal{S}_{ij} \qquad d_{iB} = p_{ti}^{-2}$$

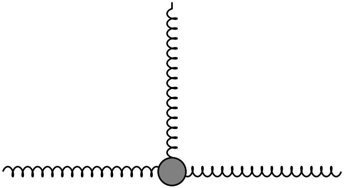
$$\mathcal{S}_{ij} = 1 - \Theta(1-\kappa)\cos\left(\frac{\pi}{2}\kappa\right) \qquad \kappa = \frac{1}{2a}\frac{p_{ti}^2 + p_{tj}^2}{p_{t,\max}^2}$$

Examples of pitfalls / I

- Czakon, Mitov & Poncelet (CMP): modification of anti-k_t distance for flavoured pairs

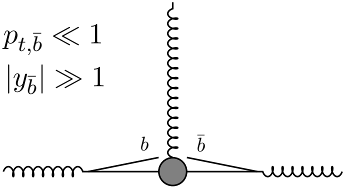
$$d_{ij} = \min\left(p_{ti}^{-2}, p_{tj}^{-2}\right)\frac{\Delta R_{ij}^2}{R^2} \times \mathcal{S}_{ij} \qquad d_{iB} = p_{ti}^{-2}$$

$$\mathcal{S}_{ij} = 1 - \Theta(1-\kappa)\cos\left(\frac{\pi}{2}\kappa\right) \qquad \kappa = \frac{1}{2a}\frac{p_{ti}^2 + p_{tj}^2}{p_{t,\max}^2}$$



$p_{t,b},\, p_{t,\bar{b}} \ll 1$

$|y_b|,\, |y_{\bar{b}}| \gg 1$

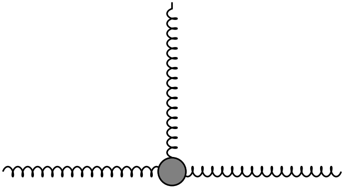


Examples of pitfalls / I

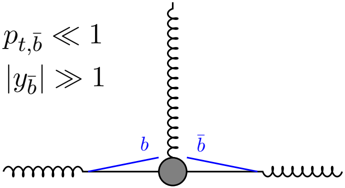
- Czakon, Mitov & Poncelet (CMP): modification of anti-k_t distance for flavoured pairs

$$d_{ij} = \min\left(p_{ti}^{-2}, p_{tj}^{-2}\right)\frac{\Delta R_{ij}^2}{R^2} \times \mathcal{S}_{ij} \qquad d_{iB} = p_{ti}^{-2}$$

$$\mathcal{S}_{ij} = 1 - \Theta(1-\kappa)\cos\left(\frac{\pi}{2}\kappa\right) \qquad \kappa = \frac{1}{2a}\frac{p_{ti}^2 + p_{tj}^2}{p_{t,\max}^2}$$



$$p_{t,b},\, p_{t,\bar{b}} \ll 1$$
$$|y_b|,\, |y_{\bar{b}}| \gg 1$$

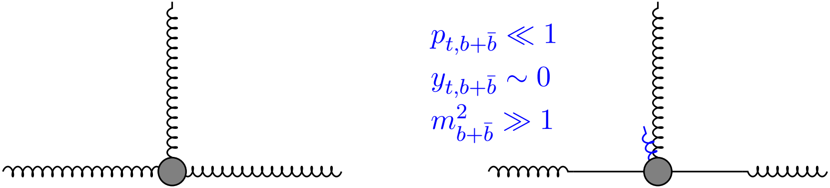


Examples of pitfalls / I

- Czakon, Mitov & Poncelet (CMP): modification of anti-k_t distance for flavoured pairs

$$d_{ij} = \min\left(p_{ti}^{-2}, p_{tj}^{-2}\right) \frac{\Delta R_{ij}^2}{R^2} \times \mathcal{S}_{ij} \qquad d_{iB} = p_{ti}^{-2}$$

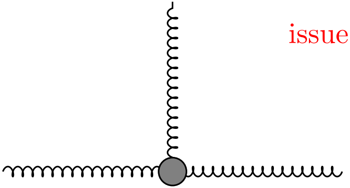
$$\mathcal{S}_{ij} = 1 - \Theta(1-\kappa)\cos\left(\frac{\pi}{2}\kappa\right) \qquad \kappa = \frac{1}{2a}\frac{p_{ti}^2 + p_{tj}^2}{p_{t,\max}^2}$$



- Czakon, Mitov & Poncelet (CMP): modification of anti-k_t distance for flavoured pairs

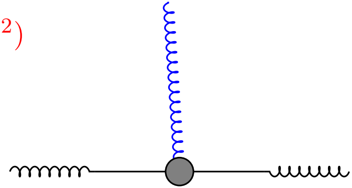
$$d_{ij} = \min\left(p_{ti}^{-2}, p_{tj}^{-2}\right) \frac{\Delta R_{ij}^2}{R^2} \times \mathcal{S}_{ij} \qquad d_{iB} = p_{ti}^{-2}$$

$$\mathcal{S}_{ij} = 1 - \Theta(1-\kappa)\cos\left(\frac{\pi}{2}\kappa\right) \qquad \kappa = \frac{1}{2a}\frac{p_{ti}^2 + p_{tj}^2}{p_{t,\max}^2}$$



issue at $\mathcal{O}(\alpha_s^2 L^2)$

$\neq$

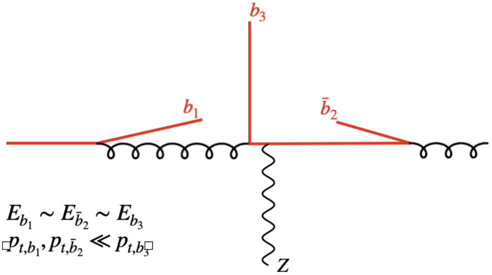


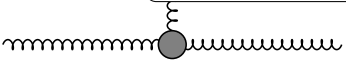
- Czakon ... ti-k_t
distanc

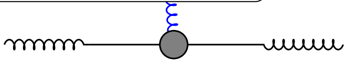
d_{ij}

$\mathcal{S}_{ij}$

Could also be $Z + b$ at NNLO:



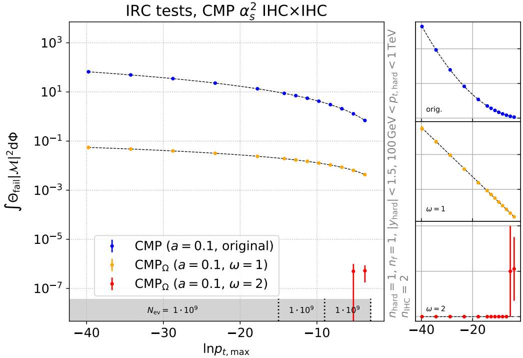


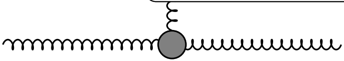


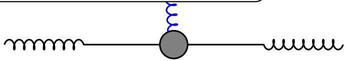
- Czakon … ti-k_t
  distanc…

d_{ij}

$\mathcal{S}_{ij}$





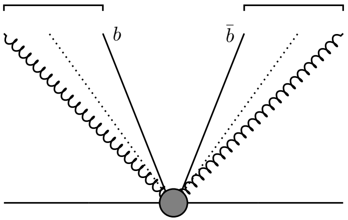


- Gauld, Huss, Stagnitto (GHS): set of jet-cluster and cluster-cluster distances

$$d_{\hat{f}_i,\hat{f}_j}, \quad d_{\hat{f}_i,j_k}, \quad d_{\hat{f}_i,B_\pm}$$

- Gauld, Huss, Stagnitto (GHS): set of jet-cluster and cluster-cluster distances

$$d_{\hat{f}_i,\hat{f}_j}, \quad d_{\hat{f}_i,j_k}, \quad d_{\hat{f}_i,B_\pm}$$



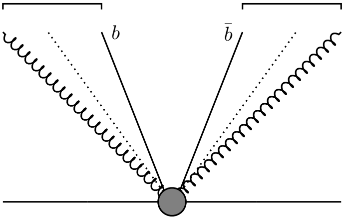
Hard event: 2 flavoured jets

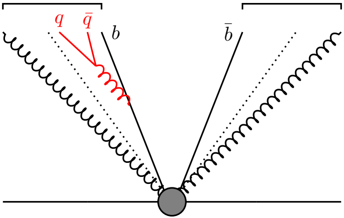
- Gauld, Huss, Stagnitto (GHS): set of jet-cluster and cluster-cluster distances

$$d_{\hat{f}_i,\hat{f}_j}, \quad d_{\hat{f}_i,j_k}, \quad d_{\hat{f}_i,B_\pm}$$

$d_{\bar{b},j_{\bar{b}}} < d_{b,\bar{b}}$

$p_{t,g} \sim z p_{t,b}$, $z \to 1$



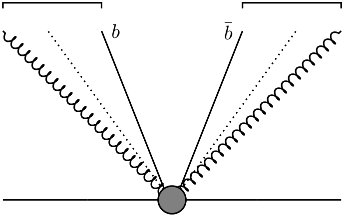


Examples of pitfalls / II

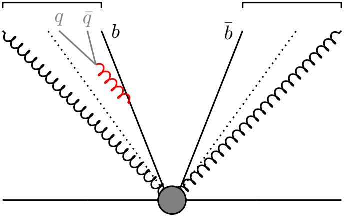
- Gauld, Huss, Stagnitto (GHS): set of jet-cluster and cluster-cluster distances

$$d_{\hat{f}_i,\hat{f}_j}, \quad d_{\hat{f}_i,j_k}, \quad d_{\hat{f}_i,B_\pm}$$

$d_{\bar{b},j_{\bar{b}}} < d_{b,\bar{b}}$



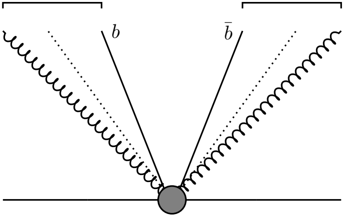
$d_{b\bar{b}} \downarrow$



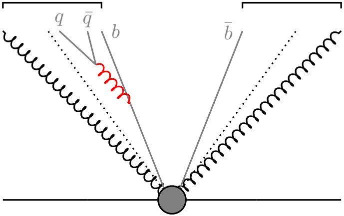
- Gauld, Huss, Stagnitto (GHS): set of jet-cluster and cluster-cluster distances

$$d_{\hat{f}_i,\hat{f}_j}, \quad d_{\hat{f}_i,j_k}, \quad d_{\hat{f}_i,B_\pm}$$

$d_{\bar{b},j_{\bar{b}}} < d_{b,\bar{b}}$

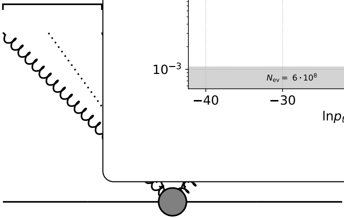


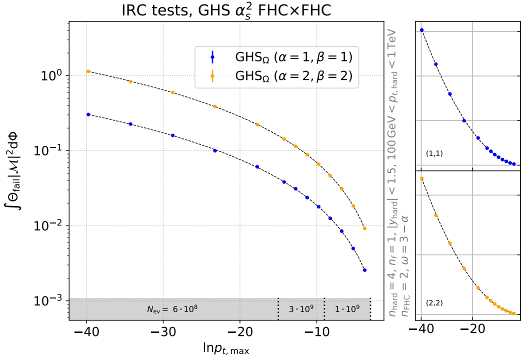
$\neq$

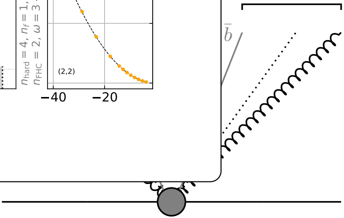


- Ga… d
  clus…

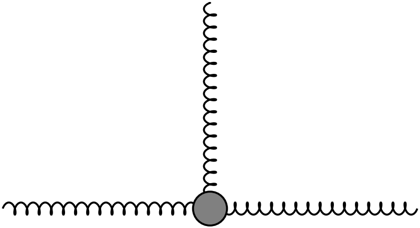
$d_{\bar{b},j_{\bar{b}}} < d_{b,\bar{b}}$ … avourless jets



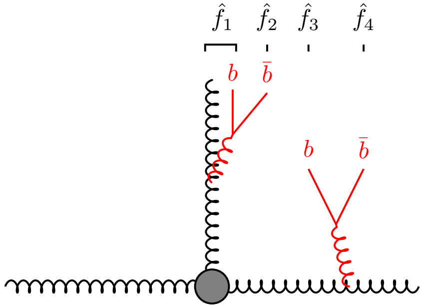




(Could also be $t\bar{t}$ boosted, or $t\bar{t} + 1$ jet)



Hard event:
→ 1 flavourless jet



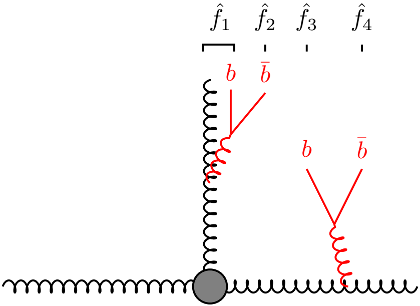
Hard event:
→ 1 flavourless jet

hard+IRC event:

1(b) accumulated into hard g, but not 2($\bar{b}$)

$\hat{f}_2$ and $\hat{f}_3$ annihilate, but $\hat{f}_1$ and $\hat{f}_4$ do not

→ 1 b-jet (+ 1 $\bar{b}$ beam jet)



Hard event:

→ 1 flavourless jet

hard+IRC event:

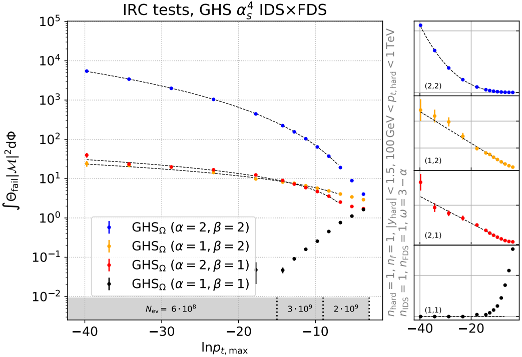
1(b) accumulated into hard g, but not 2($\bar{b}$)

$\hat{f}_2$ and $\hat{f}_3$ annihilate, but $\hat{f}_1$ and $\hat{f}_4$ do not

→ 1 b-jet (+ 1 $\bar{b}$ beam jet)

- Some analytic/numerical understanding of the complicated interplay between distances (as a function of α and β)

  → suggests $\alpha \cdot \beta < 2$ is fine for this configuration



into hard g,

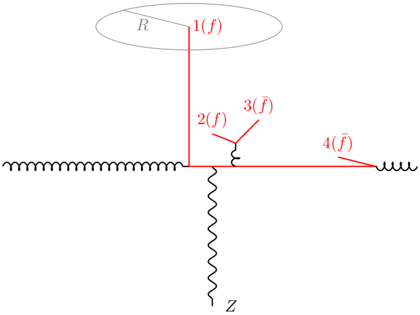
te,

not

am jet)

- Some … cated interpl…)

$\rightarrow$ suggests $\alpha \cdot \beta < 2$ is fine for this configuration



Hard event:
$\rightarrow$ 1 flavoured (f) jet

Hard+(1 IDS pair) event:

$$p_{t,2} = z p_{t,g}$$
$$E_4 \sim E_1,\ y_4 \sim \log \frac{E_4}{p_{t,4}}$$

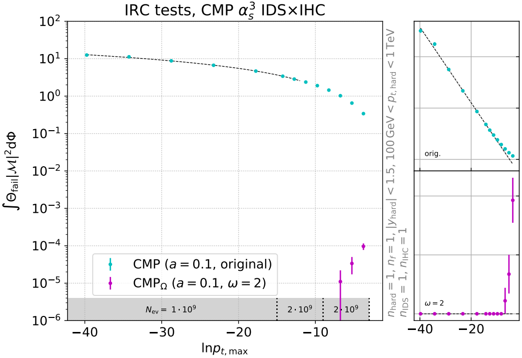
$$d_{23} \sim \frac{p_{t,2}^2}{z^2},\ d_{24} \sim p_{t,2}^2 y_4^2$$

CMP failure rate: $\mathcal{N} \sim \alpha_s^3 \int_0^{p_{t,1}} \frac{dp_{t,2}}{p_{t,2}} \int_0^{p_{t,1}} \frac{dp_{t,4}}{p_{t,4}} \int_0^1 dz \Theta(d_{24} < d_{23})$

$$\sim \alpha_s^3 \int_0^L d\ell_{14} \int_0^{l_{14}} d\ell_{24} \int_0^{1/\ell_{14}} dz$$

$$\sim \alpha_s^3 L$$

Examples of pitfalls / IV



et

event:

$\frac{E_4}{p_{t,4}}$