Introduction to QCD

1. Fundamentals of QCD
2. PDFs, Fixed-Order QCD, and Jet Algorithms
3. Parton Showers and Event Generators
4. QCD in the Infrared

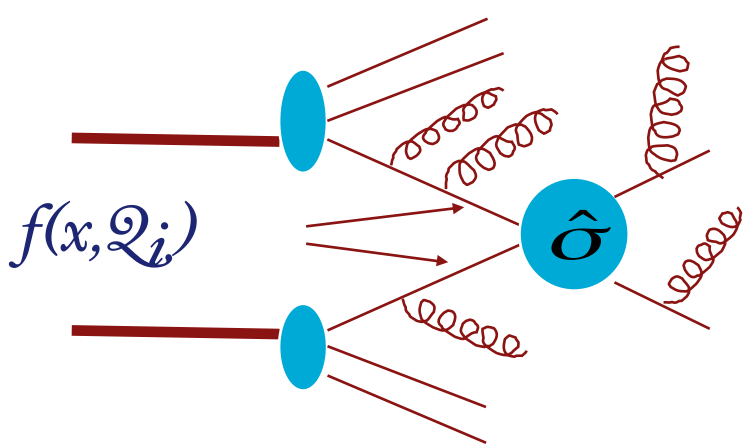
Slides posted at:
www.cern.ch/skands/slides

Lecture Notes:
[P. Skands, arXiv:1207.2389]

Factorization Summary

Factorization: expresses the independence of long-wavelength (soft) emission on the nature of the hard (short-distance) process.

$$\frac{\mathrm{d}\sigma}{\mathrm{d}X} = \sum_{a,b}\sum_{f}\int_{\hat{X}_f} \boxed{f_a(x_a,Q_i^2)f_b(x_b,Q_i^2)}\ \boxed{\frac{\mathrm{d}\hat{\sigma}_{ab\to f}(x_a,x_b,f,Q_i^2,Q_f^2)}{\mathrm{d}\hat{X}_f}}\ \boxed{D(\hat{X}_f \to X, Q_i^2, Q_f^2)}$$



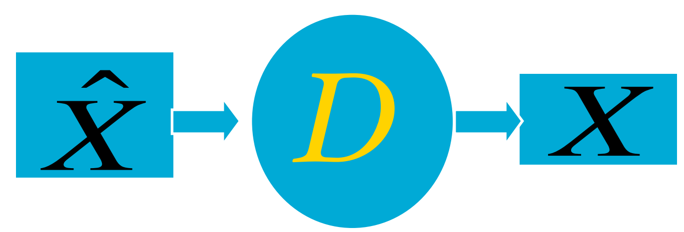


Illustration by M. Mangano

$$\vec{p}_j = x\,\vec{P}_{proton}$$

$$\boxed{f_a(x_a,Q_i^2)}$$

Parton distribution functions (PDF)

- sum over long-wavelength histories leading to a with x_a at the scale Q_i^2 (ISR)

$$\boxed{D(\hat{X}_f \to X, Q_i^2, Q_f^2)}$$

Fragmentation Function (FF)

- Sum over long-wavelength histories from $\hat{X}_f$ at Q_f^2 to X (FSR and Hadronization)

+ (At H.O. each of these defined in a specific scheme, usually $\overline{\mathrm{MS}}$)

Parton Densities

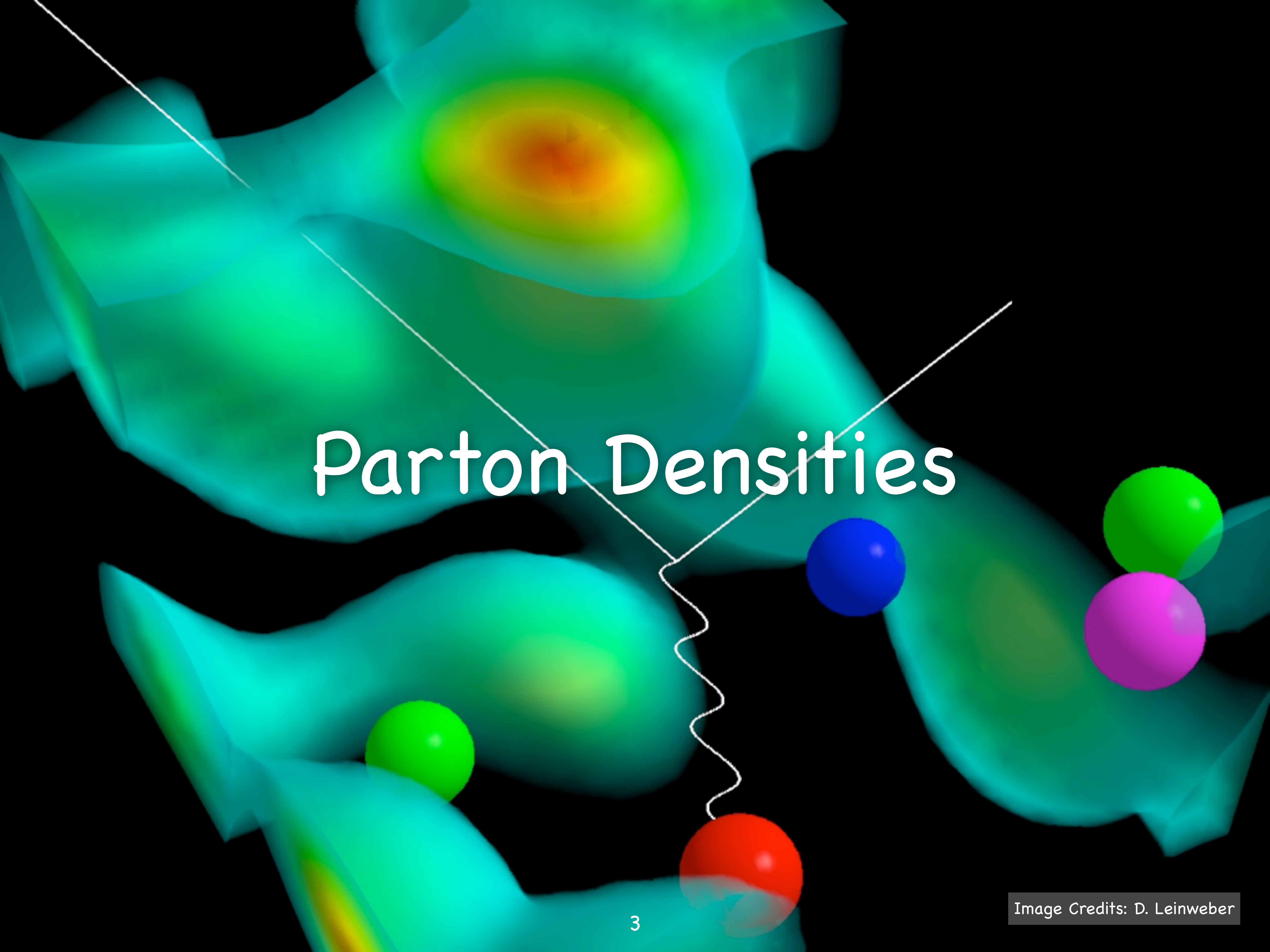


Image Credits: D. Leinweber

Parton Densities

$$\vec{p}_j = x\,\vec{P}_{proton}$$

$$\boxed{f_a(x_a, Q_i^2)}$$ Parton distribution functions (PDF)

- sum over long-wavelength histories leading to a with x_a at the scale Q_i^2 (ISR)

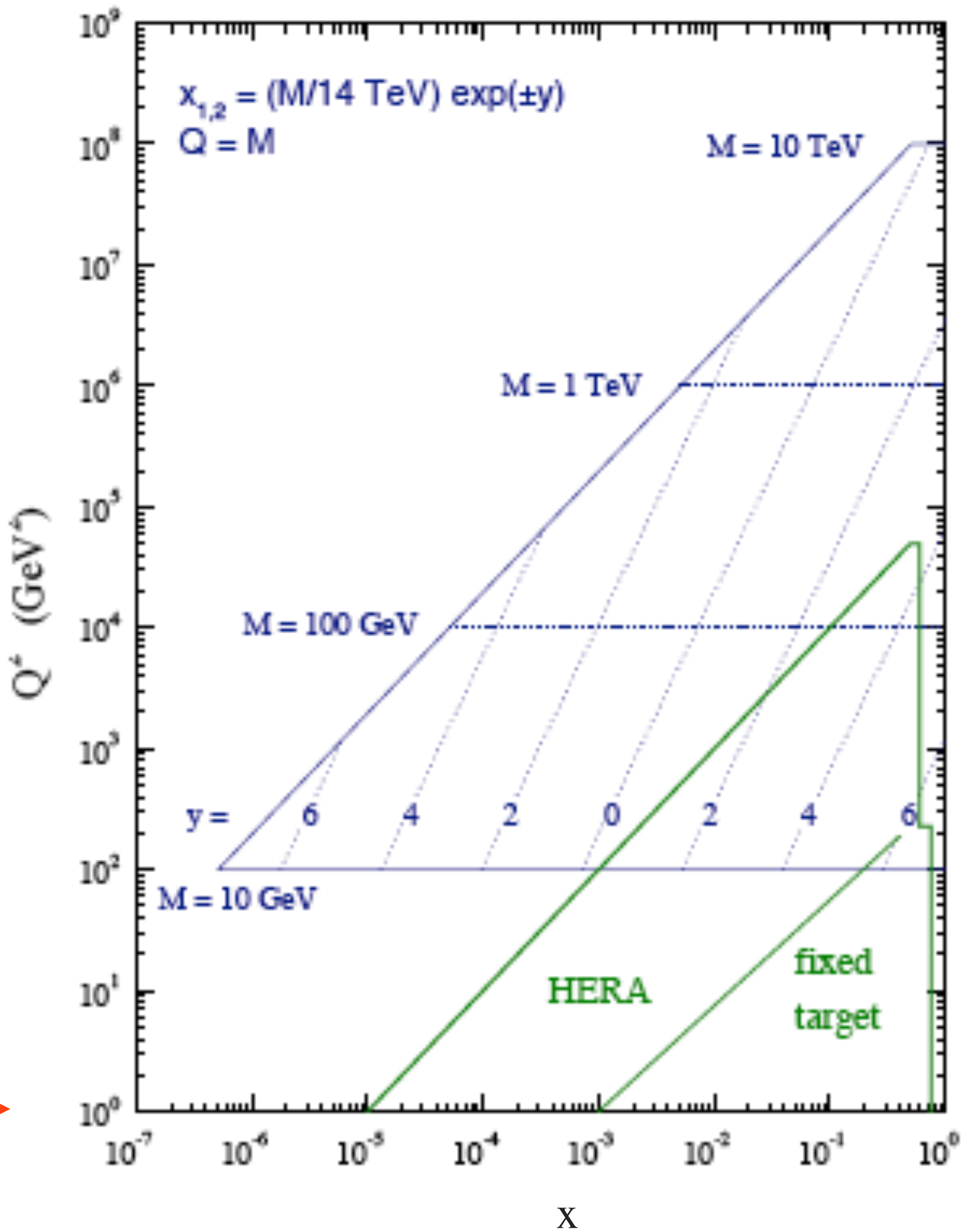
Shape of *f(x)* unknown
(non-perturbative)

Different groups (CTEQ, MSTW, NNPDF, etc) use different ansätze

→ fit to measurements

Evolve to fixed small reference scale $Q \approx m_{proton}$

LHC Coverage
x and Q^2



Evolution in Q^2 by DGLAP

(Dokshitzer-Gribov-Lipatov-Altarelli-Parisi)

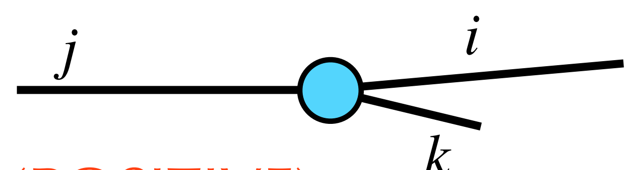
Changing Q^2 ~ changing the scale at which we look at the parton (zooming in/out on the fractal)

However, setting the factorisation scale $\mu = Q$ is **our choice**; unphysical
Require cross section independent of μ (at calculated order) → RGE

$$\frac{dq(x,\mu^2)}{d\ln\mu^2} = \frac{\alpha_s}{2\pi}\int_x^1 dz\, p_{qq}(z)\,\frac{q(x/z,\mu^2)}{z} - \frac{\alpha_s}{2\pi}\int_0^1 dz\, p_{qq}(z)\, q(x,\mu^2)$$

p_{qq} is real $q \leftarrow q$ **splitting kernel:** $p_{qq}(z) = C_F\frac{1+z^2}{1-z}$

A gain-loss equation



First term: some partons flow from higher x′=x/z to x (POSITIVE)
Second term: some partons at x flow to lower x′=zx (NEGATIVE)

$x_i = z\, x_j$
$x_k = (1-z)\, x_j$

Note: In this form, it looks pretty crazy for z → 1

PDF DGLAP : Details

Awkward to write real and virtual parts separately. Use more compact notation:

$$\frac{dq(x,\mu^2)}{d\ln\mu^2} = \frac{\alpha_s}{2\pi} \underbrace{\int_x^1 dz\, P_{qq}(z)\, \frac{q(x/z,\mu^2)}{z}}_{P_{qq}\otimes q}, \qquad P_{qq} = C_F \left(\frac{1+z^2}{1-z}\right)_+$$

This involves the *plus prescription:*

$$\int_0^1 dz\, [g(z)]_+\, f(z) = \int_0^1 dz\, g(z)\, f(z) - \int_0^1 dz\, g(z)\, f(1)$$

$z=1$ divergences of $g(z)$ cancelled if $f(z)$ sufficiently smooth at $z=1$

$$\frac{df_i(x_i,\mu_F^2)}{d\ln\mu_F^2} = \sum_j \int_{x_i}^1 \frac{dx_j}{x_j} f_j(x_j,\mu_F^2) \frac{\alpha_s}{2\pi} P_{j\to ik}\left(\frac{x_i}{x_j}\right)$$

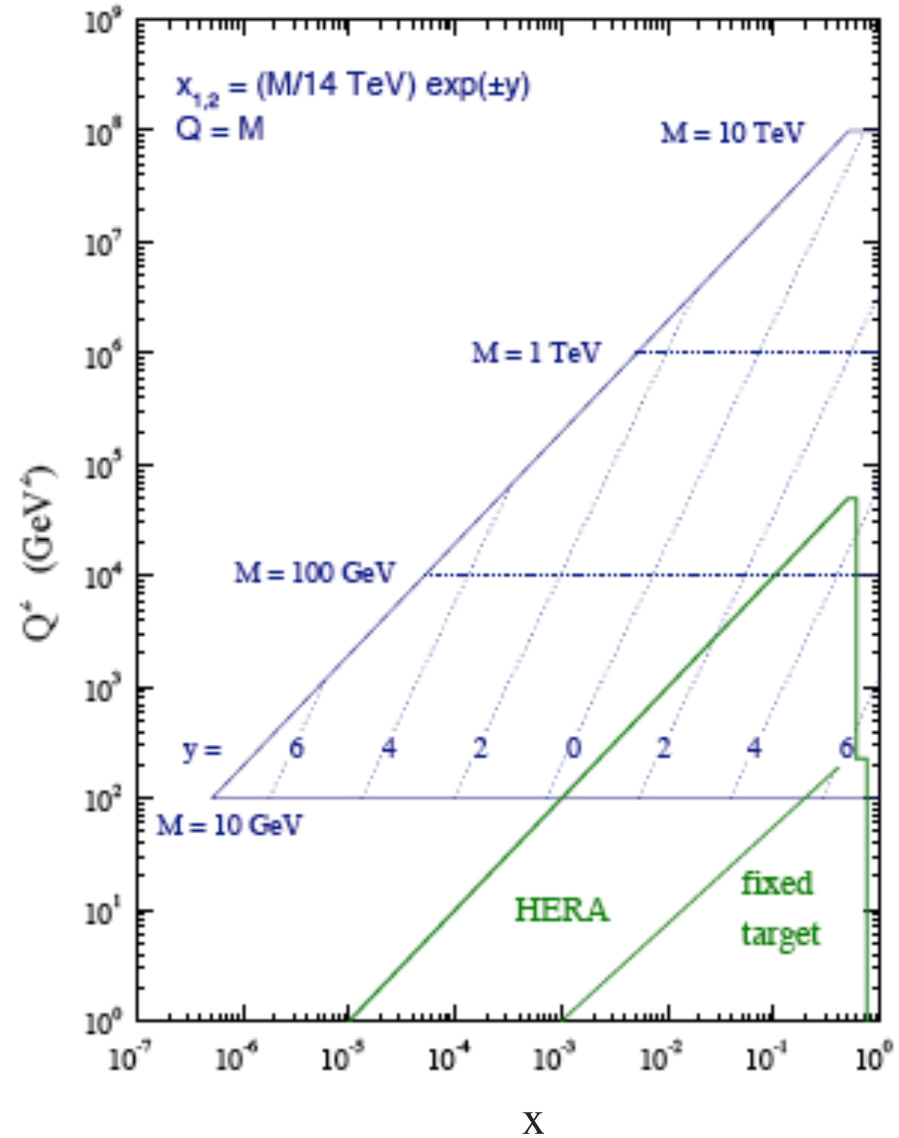
The (LO) DGLAP Evolution Kernels

$$\frac{df_i(x_i,\mu_F^2)}{d\ln\mu_F^2} = \sum_j \int_{x_i}^{1} \frac{dx_j}{x_j} f_j(x_j,\mu_F^2) \frac{\alpha_s}{2\pi} P_{j\to ik}\left(\frac{x_i}{x_j}\right)$$

Note: these are just the LO (one-loop) ones

$$\begin{aligned}
P_{\mathrm{q}\to\mathrm{qg}}(z) &= C_F \frac{1+z^2}{1-z} ,\\
P_{\mathrm{g}\to\mathrm{gg}}(z) &= N_C \frac{(1-z(1-z))^2}{z(1-z)} ,\\
P_{\mathrm{g}\to\mathrm{q}\bar{\mathrm{q}}}(z) &= T_R\,(z^2+(1-z)^2) ,\\
P_{\mathrm{q}\to\mathrm{q}\gamma}(z) &= e_{\mathrm{q}}^2 \frac{1+z^2}{1-z} ,\\
P_{\ell\to\ell\gamma}(z) &= e_\ell^2 \frac{1+z^2}{1-z} ,
\end{aligned}$$

LHC Coverage
x and Q^2



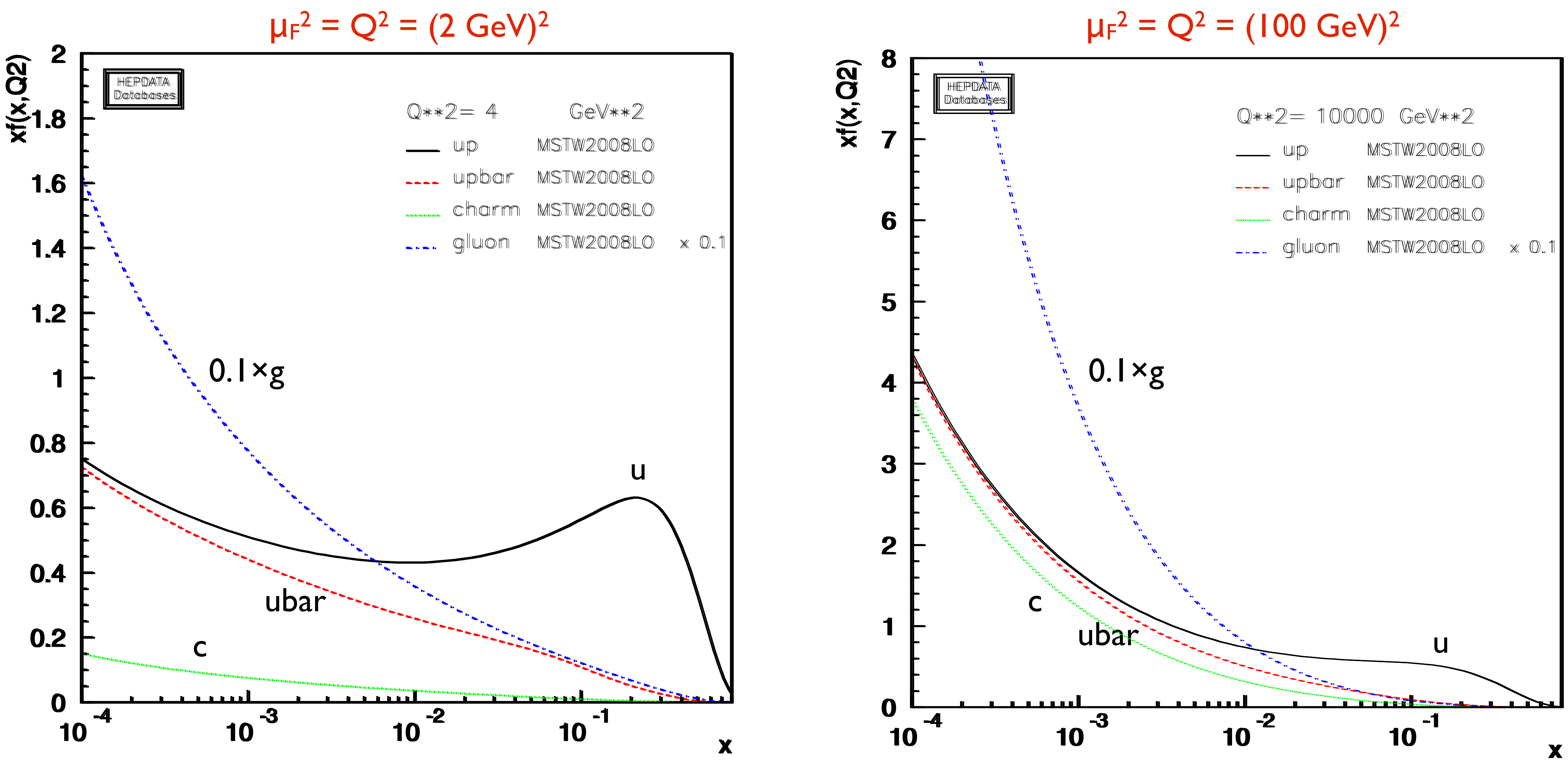
Relate measurements at different Q^2
Extrapolate to new energies (eg LHC)
(Note: extrapolation in x more tricky …)

Evolution in Q^2 by DGLAP

(Dokshitzer-Gribov-Lipatov-Altarelli-Parisi)

Require cross section independent of μ_F (at calculated order) → RGE

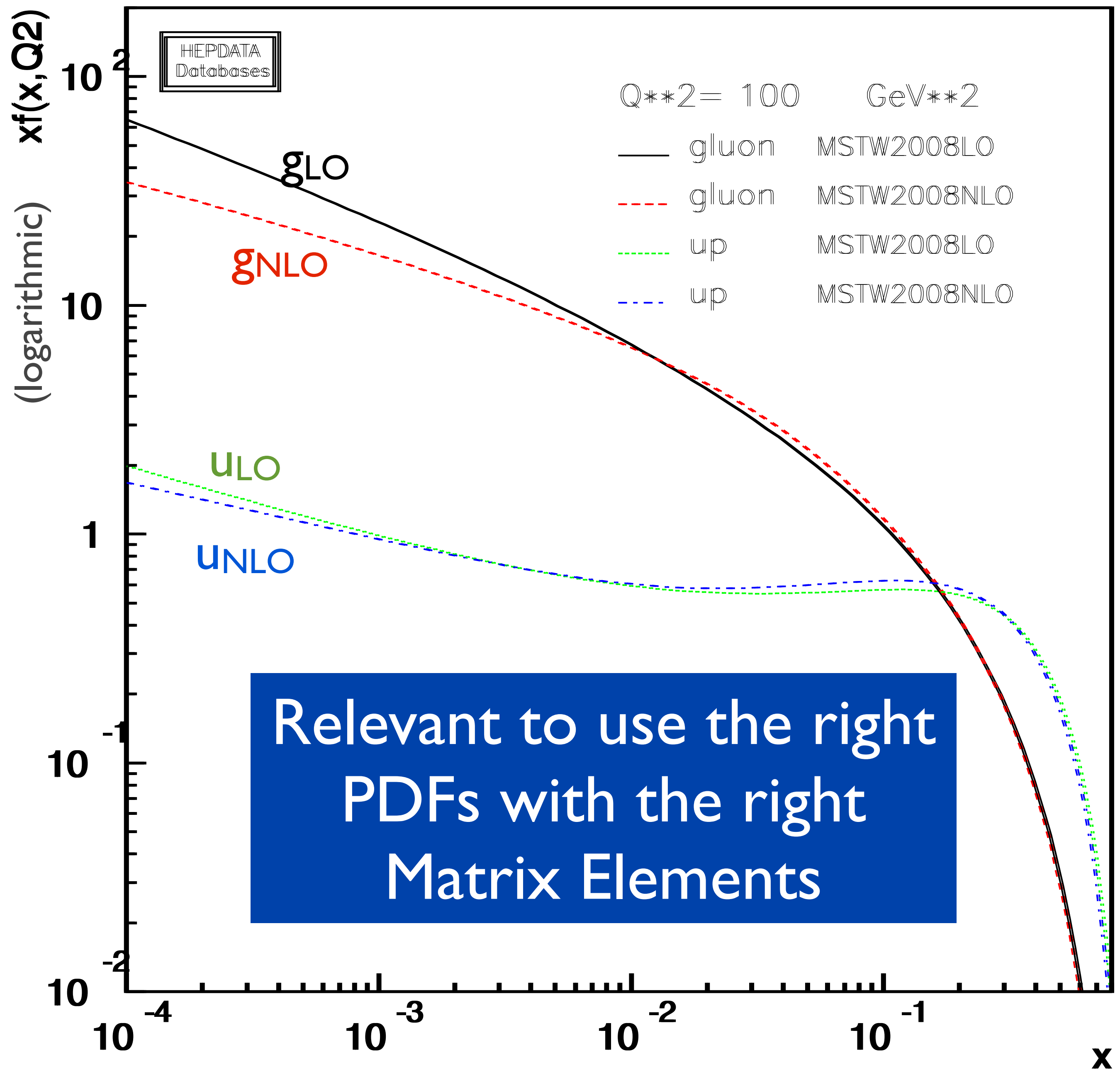
$$\frac{df_i(x_i,\mu_F^2)}{d\ln\mu_F^2} = \dots$$



LO vs NLO

$$\sigma^{\ell h} = \sum_i \sum_f \int dx_i \int d\Phi_f \, f_{i/h}(x_i, Q_F^2) \boxed{\frac{d\hat{\sigma}^{\ell i \to f}(x_i, \Phi_f, Q_F^2)}{dx_i \, d\Phi_f}}$$

$Q^2 = (10\ \text{GeV})^2$



The "best fit" depends on the matrix elements you use when doing the fit

NLO matrix elements

contain low-x enhancements (they are larger than LO×DGLAP)

→ need less low-x PDFs

(+ momentum conservation

→ more partons at high x

→ larger cross sections)

(Advanced) PDF Uncertainties

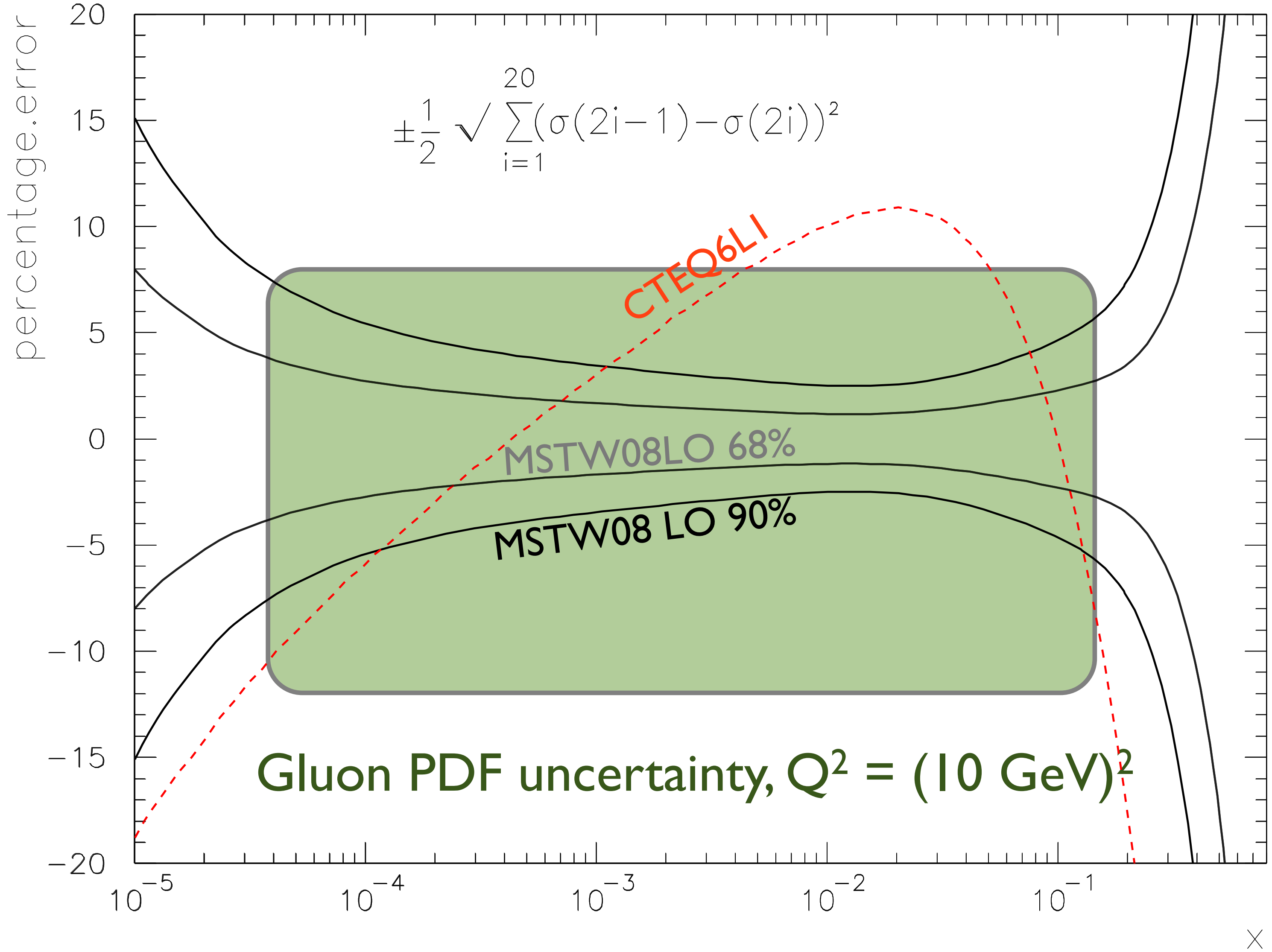
Much debate recently on PDF errors

Attempt to propagate experimental errors through PDF fits → 68% CL

But “tensions” between different data sets

→ 90%, or something else?

+ Different groups (CTEQ, MSTW, NNPDF, etc) use different ansätze for shape of f(x) at low-Q boundary



Still, good to ≈ 10% even for LO gluon in $10^{-4} < x < 10^{-1}$ (bigger errors at lower Q^2)

QCD at Fixed Order

Distribution of observable: O

In production of X + anything

Sum over identical amplitudes, then square

Momentum configuration

Fixed Order
(All Orders)

$$\left.\frac{d\sigma}{d\mathcal{O}}\right|_{\mathrm{ME}} = \sum_{k=0} \int \mathrm{d}\Phi_{X+k} \left|\sum_{\ell=0} M^{(\ell)}_{X+k}\right|^2 \delta\left(\mathcal{O} - \mathcal{O}(\{p\}_{X+k})\right)$$

Phase Space

Cross Section differentially in O

Sum over "anything" ≈ legs

Matrix Elements for X+k at (l) loops

Evaluate observable → differential in O

Truncate at $k = 0, \ell = 0$,
→ **Born Level = First Term**
Lowest order at which X happens

Loops and Legs

Another representation

Born

Max Born
(1882-1970)
Nobel Prize 1954

Loops					
	X	X+1	…		
	X	X+1	X+2	X+3	…
	Born	X+1	X+2	X+3	…

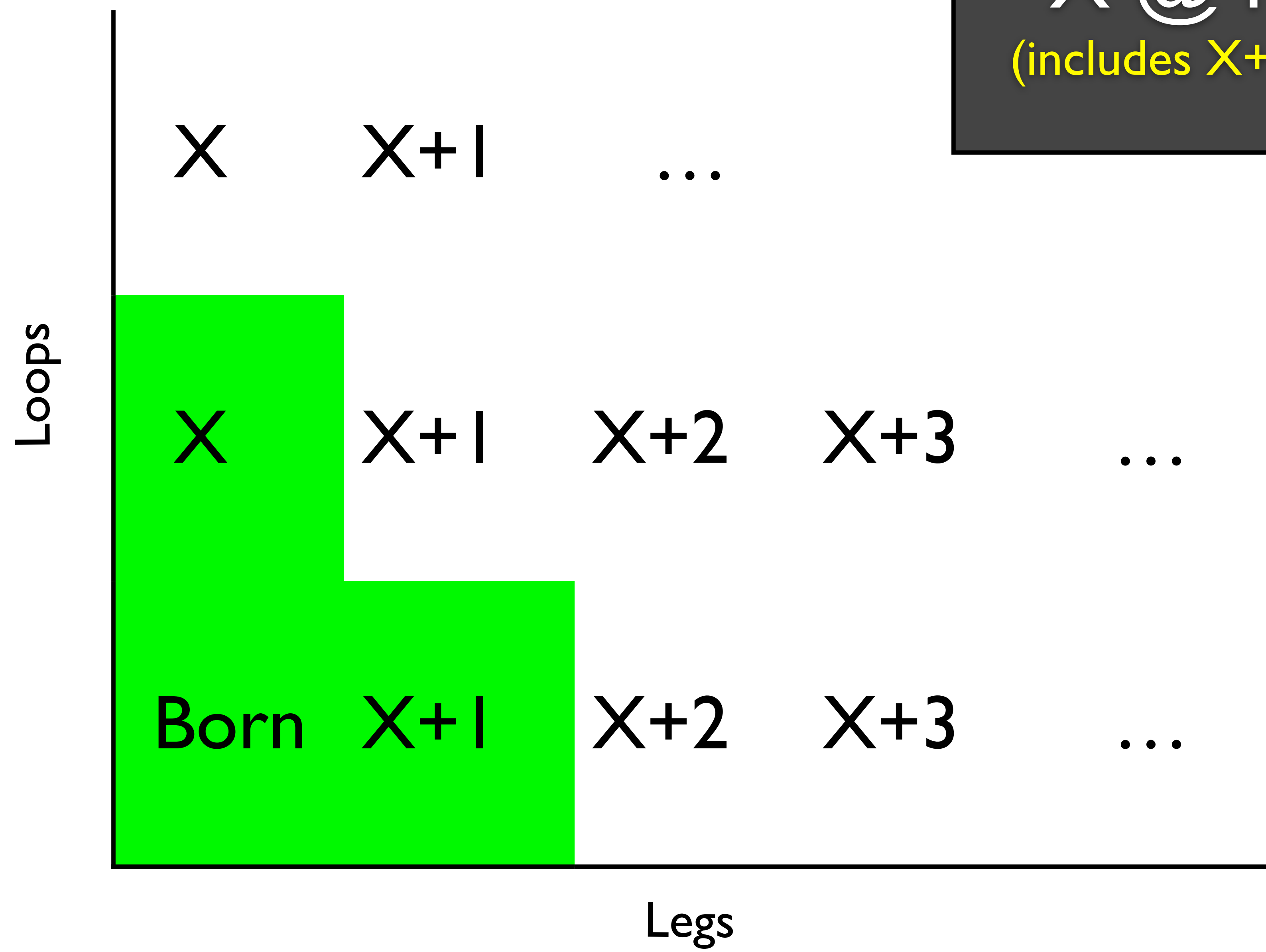
Legs

Loops and Legs

Another representation

X @ NLO

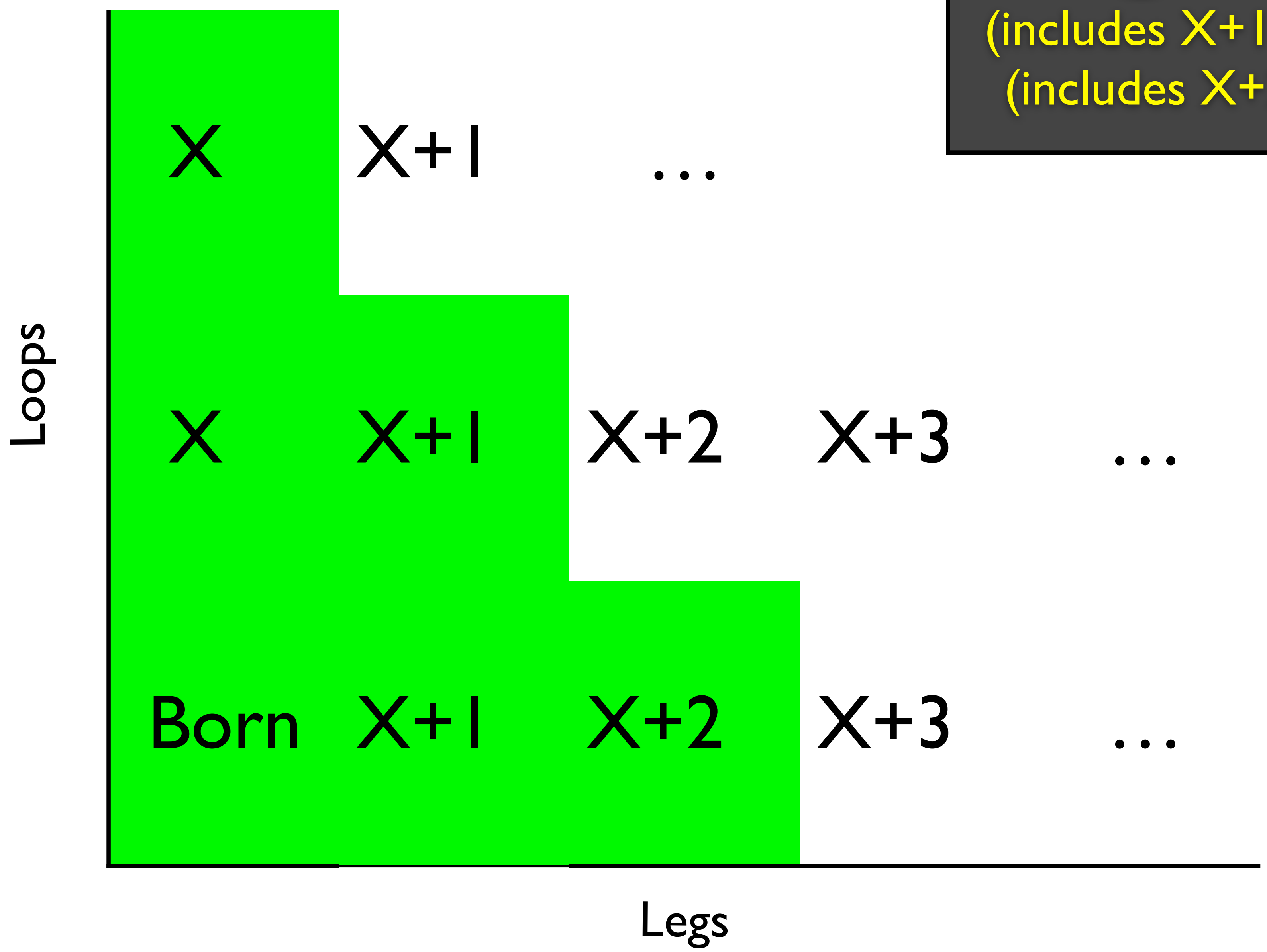
(includes X+1 @ LO)



Note: X+1 jet observables only correct at LO

Loops and Legs

Another representation



X @ NNLO
(includes X+1 @ NLO)
(includes X+2 @ LO)

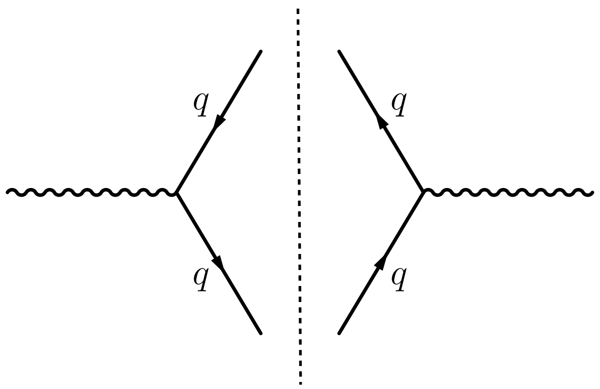
Note: X+1 jet observables only correct at NLO

Note: X+2 jet observables only correct at LO

Cross sections at LO

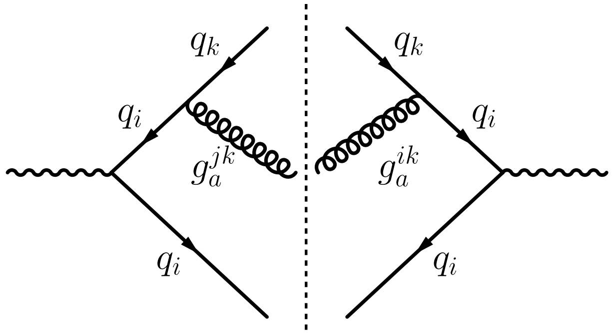
Born:

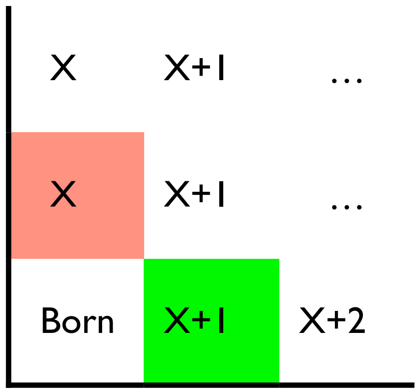
$$\sigma_{\mathrm{Born}} = \int |M_X^{(0)}|^2$$



Born + n

$$\sigma^{\mathrm{LO}}_{\mathrm{X+1}}(R) = \int_R |M^{(0)}_{X+1}|^2$$





Infrared divergent (cf Lecture 1) → Must be regulated

R = some Infrared Safe phase space region

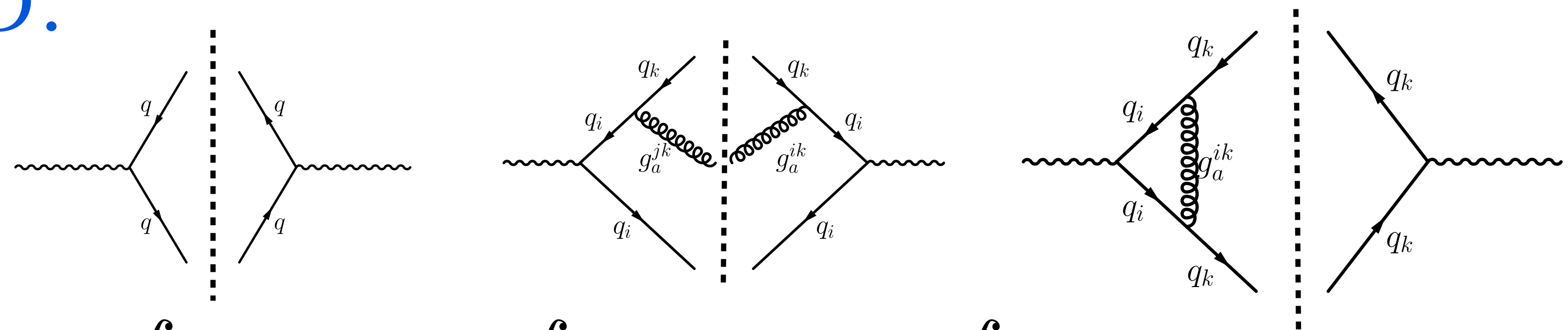
(Often a cut on $p_\perp > n$ GeV)

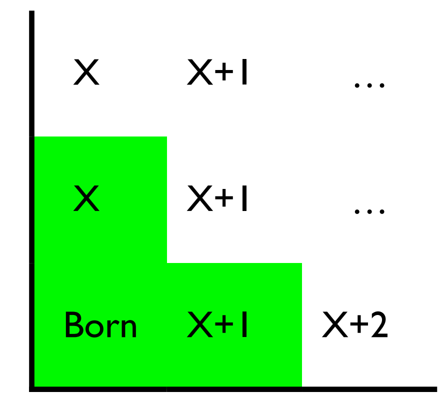
Careful not to take it too low!

if σ(X+n) ≈ σ(X) you got a problem
perturbative expansion not reliable
(see example on slide 23 of first lecture)

Cross sections at NLO

NLO:





$$\sigma_{\mathrm{X}}^{\mathrm{NLO}} = \int |M_X^{(0)}|^2 + \int |M_{X+1}^{(0)}|^2 + \int 2\mathrm{Re}[M_X^{(1)} M_X^{(0)*}]$$

(note: this is *not* the 1-loop diagram squared)

KLN Theorem (Kinoshita-Lee-Nauenberg)

Sum over 'degenerate quantum states' :

Singularities cancel at complete order (only finite terms left over)

$$= \sigma_{\mathrm{Born}} + \mathrm{Finite}\left\{\int |M_{X+1}^{(0)}|^2\right\} + \mathrm{Finite}\left\{\int 2\mathrm{Re}[M_X^{(1)} M_X^{(0)*}]\right\}$$

$$\boxed{\sigma_{\mathrm{NLO}}(e^+e^- \to q\bar{q}) = \sigma_{\mathrm{LO}}(e^+e^- \to q\bar{q}) \left(1 + \frac{\alpha_s(E_{\mathrm{CM}})}{\pi} + \mathcal{O}(\alpha_s^2)\right)}$$

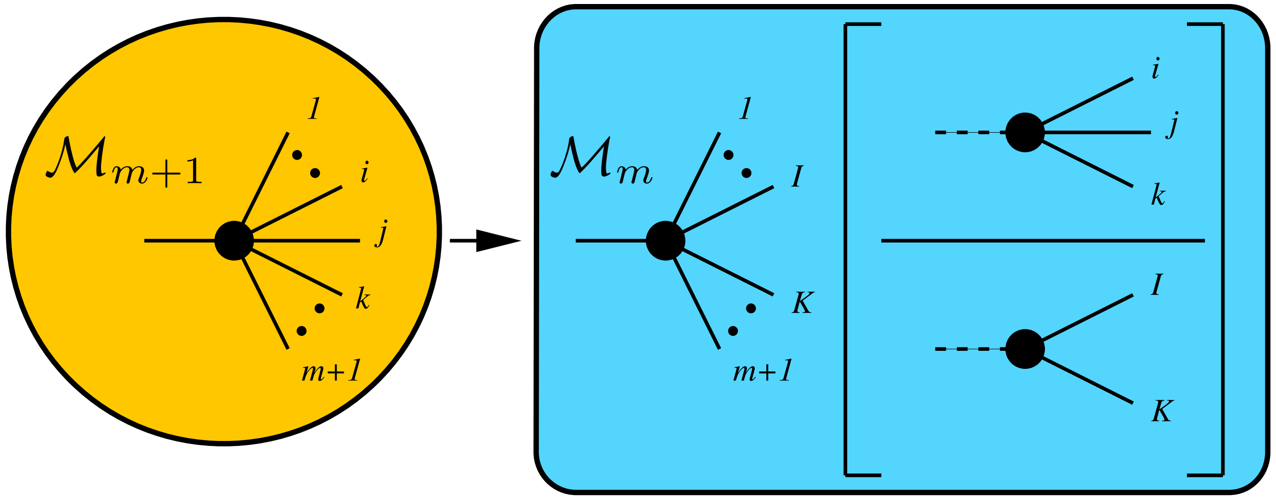
The Subtraction Idea

How do I get finite{Real} and finite{Virtual} ?

First step: classify IR singularities using universal functions

EXAMPLE: factorization of amplitudes in the **soft** limit

Soft Limit
($E_j \to 0$):



$$|\mathcal{M}_{n+1}(1,\cdots,i,j,k,\cdots,n+1)|^2 \xrightarrow{j_g\to 0} g_s^2 \mathcal{C}_{ijk}\, S_{ijk}\, |\mathcal{M}_n(1,\cdots,i,k,\cdots,n+1)|^2$$

Universal
"Soft Eikonal"

$$S_{ijk}(m_I,m_K) = \frac{2s_{ik}}{s_{ij}s_{jk}} - \frac{2m_I^2}{s_{ij}^2} - \frac{2m_K^2}{s_{jk}^2} \qquad s_{ij} \equiv 2p_i\cdot p_j$$

The Subtraction Idea

Add and subtract IR limits (SOFT and COLLINEAR)

$$\mathrm{d}\sigma_{NLO} = \underbrace{\int_{\mathrm{d}\Phi_{m+1}} \left(\mathrm{d}\sigma^R_{NLO} - \mathrm{d}\sigma^S_{NLO}\right)}_{\text{Finite by Universality}} + \underbrace{\left[\int_{\mathrm{d}\Phi_{m+1}} \mathrm{d}\sigma^S_{NLO} + \int_{\mathrm{d}\Phi_m} \mathrm{d}\sigma^V_{NLO}\right]}_{\text{Finite by KLN}}$$

Dipoles (Catani-Seymour)

Global Antennae (Gehrmann, Gehrmann-de Ridder, Glover)

Sector Antennae (Kosower)

…

Choice of subtraction terms:

Singularities mandated by gauge theory

Non-singular terms: up to you (added and subtracted, so vanish)

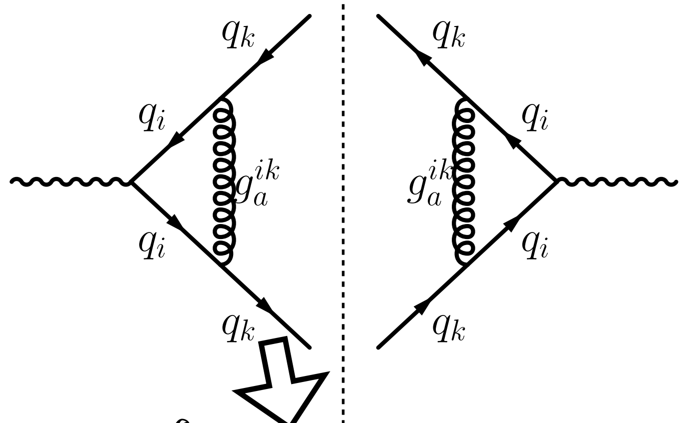
$$\frac{|\mathcal{M}(Z^0 \to q_i g_j \bar{q}_k)|^2}{|\mathcal{M}(Z^0 \to q_I \bar{q}_K)|^2} = g_s^2\, 2C_F \left[\overbrace{\frac{2s_{ik}}{s_{ij}s_{jk}}}^{\text{SOFT}} + \overbrace{\frac{1}{s_{IK}} \left(\frac{s_{ij}}{s_{jk}} + \frac{s_{jk}}{s_{ij}} \right)}^{\text{COLLINEAR}} \right]$$

$$\frac{|\mathcal{M}(H^0 \to q_i g_j \bar{q}_k)|^2}{|\mathcal{M}(H^0 \to q_I \bar{q}_K)|^2} = g_s^2\, 2C_F \left[\underbrace{\frac{2s_{ik}}{s_{ij}s_{jk}}}_{\text{SOFT}} + \underbrace{\frac{1}{s_{IK}} \left(\frac{s_{ij}}{s_{jk}} + \frac{s_{jk}}{s_{ij}} \right.}_{\text{COLLINEAR}} \underbrace{\left. + 2 \right)}_{\text{+F}} \right]$$

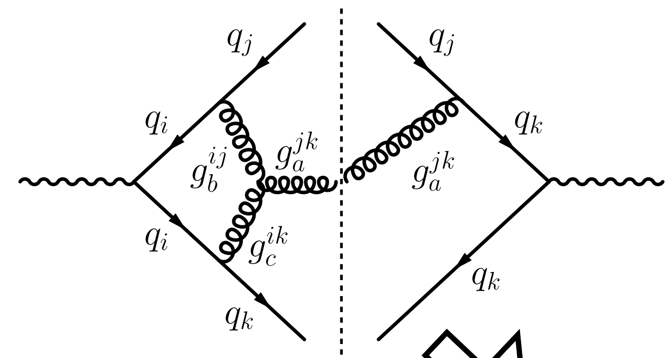
Structure of σ(NNLO)

NNLO

1-Loop × 1-Loop

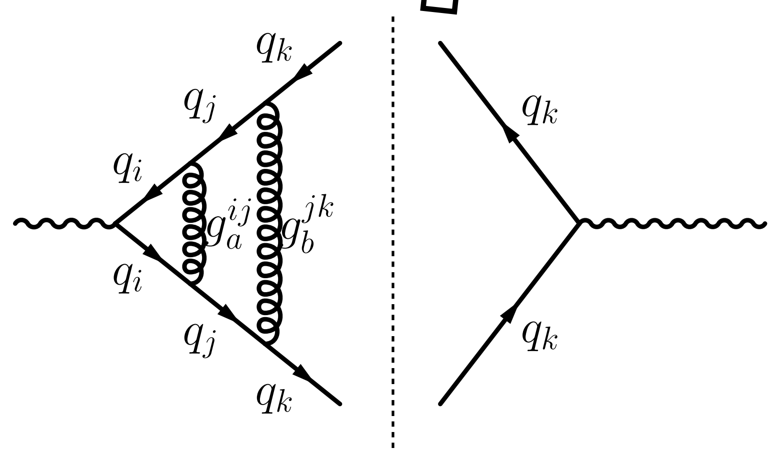


1-Loop × Real (X+1)

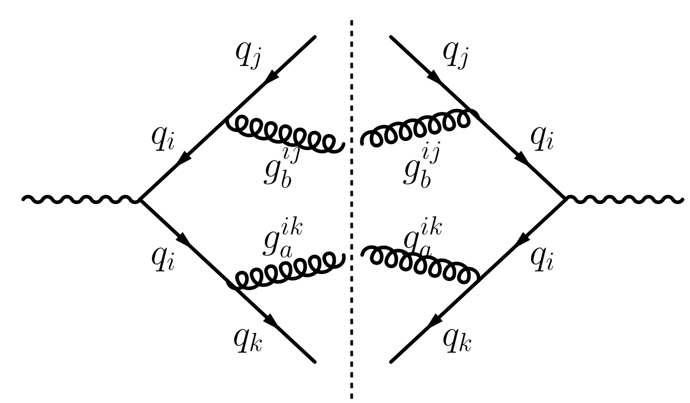


X	X+1	...
X	X+1	...
Born	X+1	X+2

$$\sigma_{\mathrm{X}}^{\mathrm{NNLO}} = \sigma_X^{\mathrm{NLO}} + \int \left(|M_X^{(1)}|^2 + 2\mathrm{Re}[M_X^{(2)} M_X^{(0)*}] \right) + \int 2\mathrm{Re}[M_{X+1}^{(1)} M_{X+1}^{(0)*}] + \int |M_{X+2}^{(0)}|^2$$



Two-Loop × Born Interference



Real × Real (X+2)

Why Go Numerical?

Part of Z → 4 jets …

5.3 Four-parton tree-level antenna functions

The tree-level four-parton quark-antiquark antenna contains three final states: quark-gluon-gluon-antiquark at leading and subleading colour, A_4^0 and $\tilde{A}_4^0$ and quark-antiquark-quark-antiquark for non-identical quark flavours B_4^0 as well as the identical-flavour-only contribution C_4^0. The quark-antiquark-quark-antiquark final state with identical quark flavours is thus described by the sum of antennae for non-identical flavour and identical-flavour-only. The antennae for the $qgg\bar{q}$ final state are:

$$A_4^0(1_q,3_g,4_g,2_{\bar{q}}) = a_4^0(1,3,4,2) + a_4^0(2,4,3,1)\,, \tag{5.27}$$

$$\tilde{A}_4^0(1_q,3_g,4_g,2_{\bar{q}}) = \tilde{a}_4^0(1,3,4,2) + \tilde{a}_4^0(2,4,3,1) + \tilde{a}_4^0(1,4,3,2) + \tilde{a}_4^0(2,3,4,1)\,, \tag{5.28}$$

$$\begin{aligned}
a_4^0(1,3,4,2) = \frac{1}{s_{1234}}\Bigg\{ & \frac{1}{2s_{13}s_{24}s_{34}}\left[2s_{12}s_{14}+2s_{12}s_{23}+2s_{12}^2+s_{14}^2+s_{23}^2\right] \\
&+\frac{1}{2s_{13}s_{24}s_{134}s_{234}}\left[3s_{12}s_{34}^2-4s_{12}^2s_{34}+2s_{12}^3-s_{34}^3\right] \\
&+\frac{1}{s_{13}s_{24}s_{134}}\left[3s_{12}s_{23}-3s_{12}s_{34}+4s_{12}^2-s_{23}s_{34}+s_{23}^2+s_{34}^2\right] \\
&+\frac{3}{2s_{13}s_{24}}\left[2s_{12}+s_{14}+s_{23}\right]+\frac{1}{s_{13}s_{34}}\left[4s_{12}+3s_{23}+2s_{24}\right] \\
&+\frac{1}{s_{13}s_{134}^2}\left[s_{12}s_{34}+s_{23}s_{34}+s_{24}s_{34}\right] \\
&+\frac{1}{s_{13}s_{134}s_{234}}\left[3s_{12}s_{24}+6s_{12}s_{34}-4s_{12}^2-3s_{24}s_{34}-s_{24}^2-3s_{34}^2\right] \\
&+\frac{1}{s_{13}s_{134}}\left[-6s_{12}-3s_{23}-s_{24}+2s_{34}\right] \\
&+\frac{1}{s_{24}s_{34}s_{134}}\left[2s_{12}s_{14}+2s_{12}s_{23}+2s_{12}^2+2s_{14}s_{23}+s_{14}^2+s_{23}^2\right] \\
&+\frac{1}{s_{24}s_{134}}\left[-4s_{12}-s_{14}-s_{23}+s_{34}\right]+\frac{1}{s_{34}^2}\left[s_{12}+2s_{13}-2s_{14}-s_{34}\right] \\
&+\frac{1}{s_{34}^2s_{134}^2}\left[2s_{12}s_{14}^2+2s_{14}^2s_{23}+2s_{14}^2s_{24}\right]-\frac{2s_{12}s_{14}s_{24}}{s_{34}^2s_{134}s_{234}} \\
&+\frac{1}{s_{34}^2s_{134}}\left[-2s_{12}s_{14}-4s_{14}s_{24}+2s_{14}^2\right] \\
&+\frac{1}{s_{34}s_{134}s_{234}}\left[-2s_{12}s_{14}-4s_{12}^2+2s_{14}s_{24}-s_{14}^2-s_{24}^2\right] \\
&+\frac{1}{s_{34}s_{134}}\left[-8s_{12}-2s_{23}-2s_{24}\right]+\frac{1}{s_{134}^2}\left[s_{12}+s_{23}+s_{24}\right] \\
&+\frac{3}{2s_{134}s_{234}}\left[2s_{12}+s_{14}-s_{24}-s_{34}\right]+\frac{1}{2s_{134}}+\mathcal{O}(\epsilon)\Bigg\}\,,
\end{aligned} \tag{5.29}$$

$$\begin{aligned}
\tilde{a}_4^0(1,3,4,2) = \frac{1}{s_{1234}}\Bigg\{ & \frac{1}{s_{13}s_{24}s_{134}s_{234}}\left[\frac{3}{2}s_{12}s_{34}^2-2s_{12}^2s_{34}+s_{12}^3-\frac{1}{2}s_{34}^3\right] \\
&+\frac{1}{s_{13}s_{24}s_{134}}\left[3s_{12}s_{23}-3s_{12}s_{34}+4s_{12}^2-s_{23}s_{34}+s_{23}^2+s_{34}^2\right]
\end{aligned}$$

This is one of the simplest processes … computed at tree level.

$$+\frac{1}{s_{13}s_{134}(s_{13}+s_{23})}\left[s_{12}s_{24}+s_{12}s_{34}+2s_{12}^2\right]$$

Now compute and add the quantum corrections …

$$+\frac{1}{s_{13}(s_{13}+s_{23})(s_{14}+s_{24})(s_{13}+s_{14})}$$

$$+\frac{1}{s_{13}(s_{13}+s_{23})(s_{13}+s_{14})}\left[s_{12}s_{24}+2s_{12}^2\right]$$

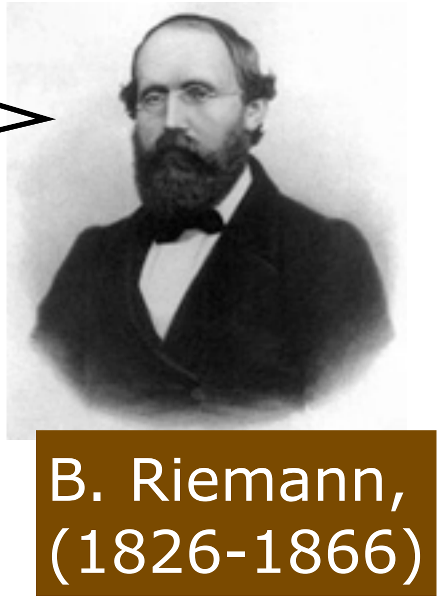
Then maybe worry about simulating the detector too …

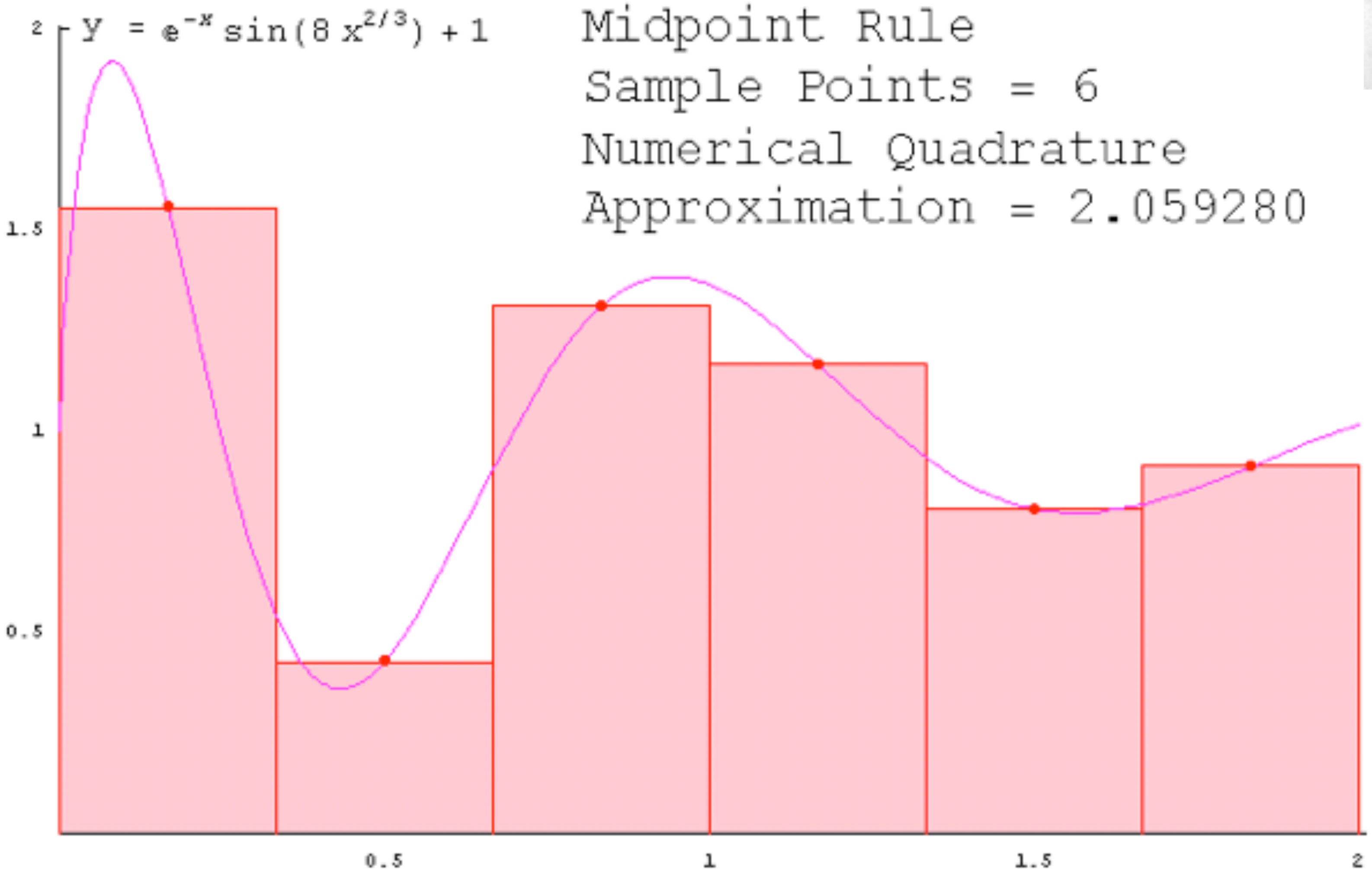
(5.30)

\+ Additional Subleading Terms …

Riemann Sums

$$\int_a^b f(x)\mathrm{d}x = \lim_{n\to\infty}\sum_{i=1}^{n} f(t_i)(x_{i+1}-x_i)$$

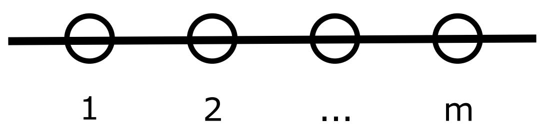




Higher Dimensions

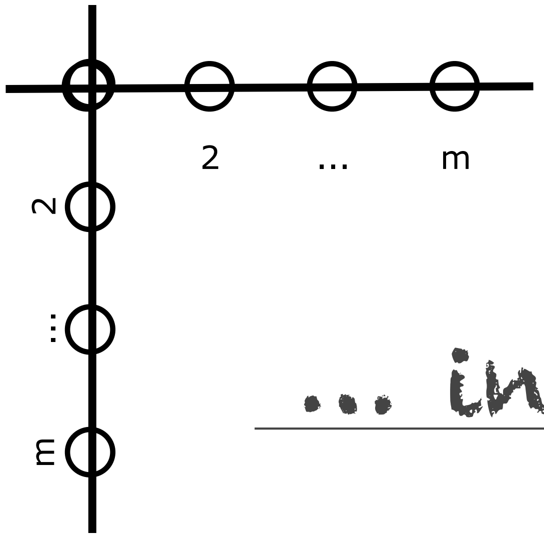
Fixed-Grid (Product) Rules scale exponentially with D

m-point rule in 1 dimension



→ m function evaluations per bin

… in 2 dimensions



→ m^2 evaluations per bin

… in D dimensions → m^D per bin

n-particle phase space grows like 3n-4
e.g. $D_3=5$ $D_4=8$ $D_5=11$

Numerical Precision

Convergence is slower in higher Dimensions!

→ More points for less precision

Uncertainty (after n evaluations)	N	Approx Conv. Rate (in D dim)
Trapezoidal Rule (2-point)	2	1/N
Simpson's Rule (3-point)	3	1/N
… m-point (Gauss rule)	m	1/N

See, e.g., "Numerical Recipes"

See, e.g., F. James, "Monte Carlo Theory and Practice"

→ Monte Carlo

A Monte Carlo technique: is any technique making use of random numbers to solve a problem

Convergence:

Calculus: {A} converges to B
if an n exists for which
|A

Monte Carlo: {A} converges to B
if n exists for which
the probability for
|A
is > P, for any P[0<P<1]

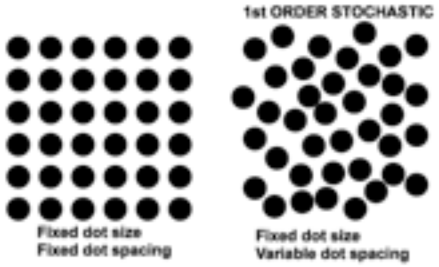
"This risk, that convergence is only given with a certain probability, is inherent in Monte Carlo calculations and is the reason why this technique was named after the world's most famous gambling casino. Indeed, the name is doubly appropriate because the style of gambling in the Monte Carlo casino, not to be confused with the noisy and tasteless gambling houses of Las Vegas and Reno, is serious and sophisticated."

F. James, "Monte Carlo theory and practice", Rept. Prog. Phys. 43 (1980) 1145

Numerical Precision

MC convergence is Stochastic!

$\frac{1}{\sqrt{n}}$ in <u>any</u> dimension



Uncertainty (after n function evaluations)	n	Approx Conv. Rate (in 1D)	Approx Conv. Rate (in D dim)
Trapezoidal Rule (2-point)	2	1/N	1/N
Simpson's Rule (3-point)	3	1/N	1/N
… m-point (Gauss rule)	m	1/N	1/N
Monte Carlo	1	1/N	1/N

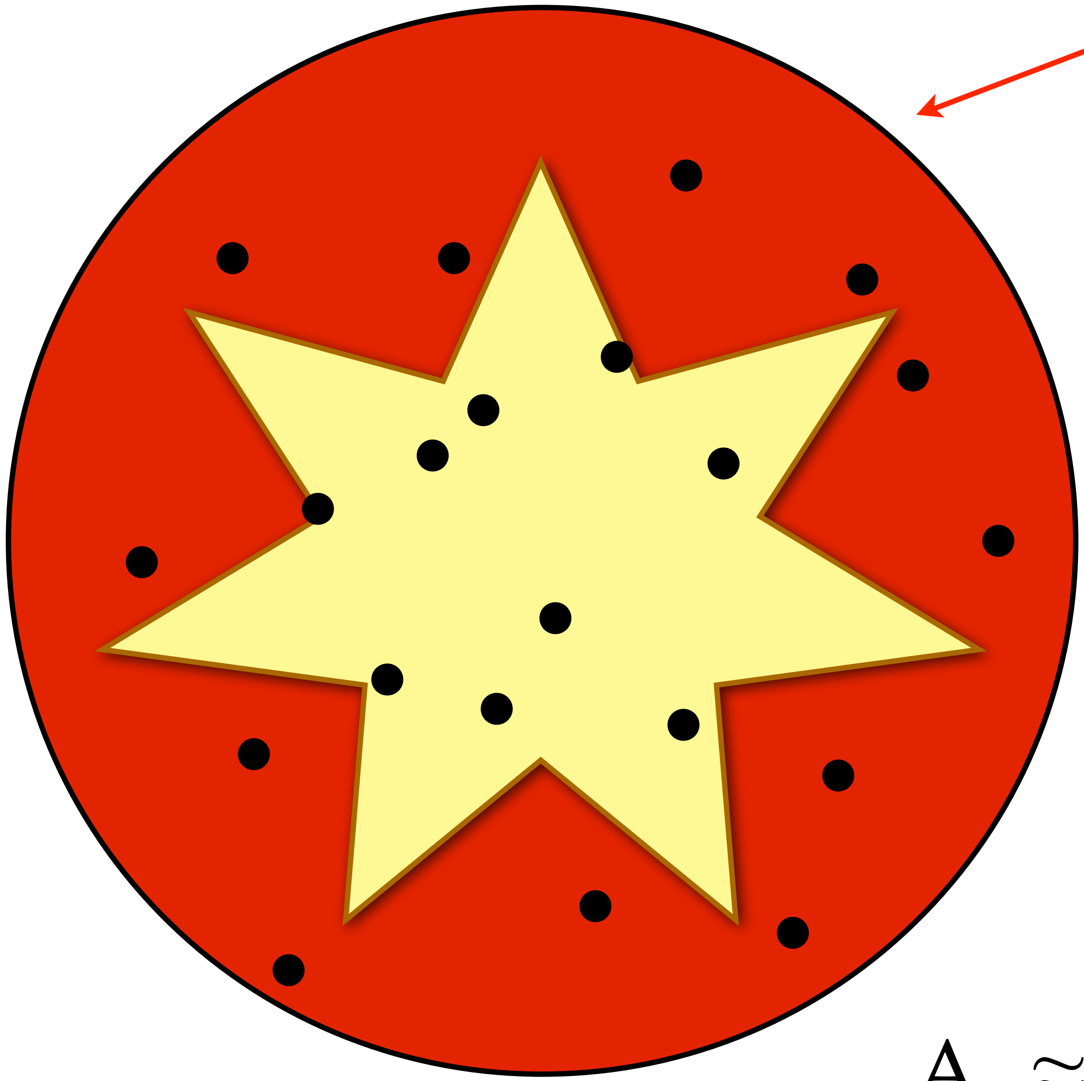
+ many ways to optimize: stratification, adaptation, …
+ gives "events" → iterative solutions (but **note:** not the only reason)
+ interfaces to detector simulation & propagation codes

MC Integration

You want: to know the area of this shape:

Now get a few friends, some balls, and throw random shots inside the circle
(but be careful to make your shots truly random)

Count how many shots hit the shape inside and how many miss



Assume you know the area of this shape:
πR^2
(an overestimate)

Earliest Example of MC calculation: Buffon's Needle (1777) to calculate π

G. Leclerc, Comte de Buffon (1707-1788)

$$A_{\star} \approx N_{\mathrm{hit}}/N_{\mathrm{throws}} \times \pi R^2$$

+ I'll stop talking about it now. More in next Lecture

Random Numbers

I will not tell you how to *write* a Random-number generator (interesting topic & history in its own right)

Instead, if you want to play with one, link to a random-number generator, from a library

E.g., ROOT includes one that you can use if you like.

PYTHIA also includes one

From the PYTHIA 8 HTML documentation, under "Random Numbers":

Random numbers `R` uniformly distributed in `0 < R < 1` are obtained with

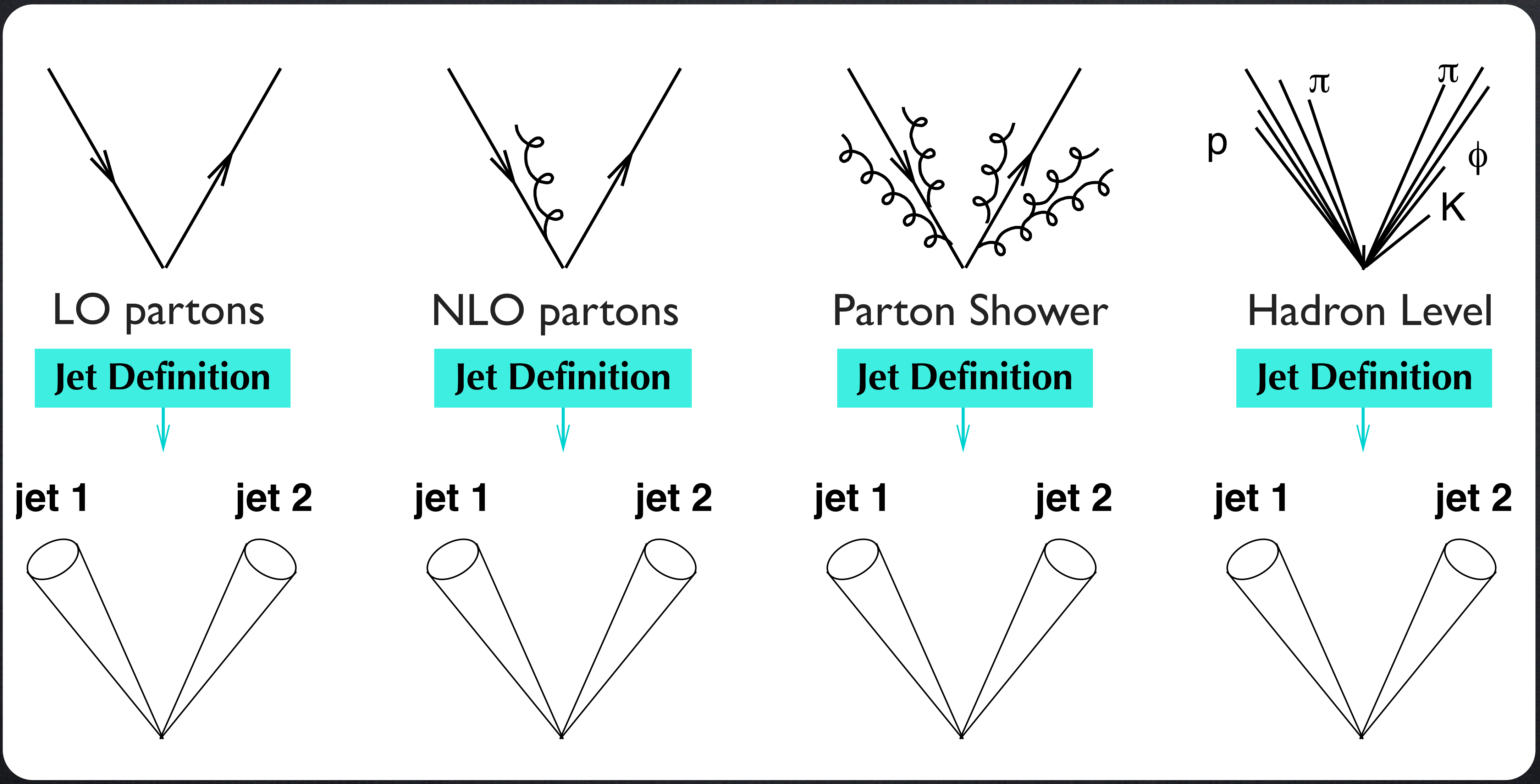
```
Pythia8::Rndm::flat();
```

+ Other methods for exp, x*exp, 1D Gauss, 2D Gauss.

Jets

& Infrared Safety

Jets as Projections



Projections to jets provides a universal view of event

Illustrations by G. Salam

There is no unique or “best” jet definition

YOU decide how to look at event

The construction of jets is inherently ambiguous

1. Which particles get grouped together?

   JET ALGORITHM (+ parameters)

2. How will you combine their momenta?

   RECOMBINATION SCHEME

   (e.g., ‘E’ scheme: add 4-momenta)

Jet Definition

Ambiguity complicates life, but gives flexibility in one’s view of events → Jets non-trivial!

Types of Algorithms

1. Sequential Recombination

Take your 4-vectors. Combine the ones that have the lowest 'distance measure'

Different names for different distance measures

Durham k_T : $\Delta R_{ij}^2 \times \min(k_{Ti}^2, k_{Tj}^2)$

Cambridge/Aachen : ΔR_{ij}^2

Anti-k_T : $\Delta R_{ij}^2 / \max(k_{Ti}^2, k_{Tj}^2)$

ArClus $(3\to2)$: $p_\perp^2 = s_{ij} s_{jk} / s_{ijk}$

$$k_{Ti}^2 = E_i^2 (1 - \cos\theta_{ij})$$

$$\Delta R_{ij}^2 = (\eta_i - \eta_j)^2 + \Delta\phi_{ij}^2$$

+ Prescription for how to combine 2 momenta into 1

(or 3 momenta into 2)

→ New set of (n-1) 4-vectors

Iterate until A or B (you choose which):

A: all distance measures larger than something

B: you reach a specified number of jets

Look at event at:

specific resolution

specific n_{jets}

Why k_T (or p_T or ΔR)?

Attempt to (approximately) capture universal jet-within-jet-witin-jet… behavior

Approximate full matrix element

"Eikonal"
(universal, always there)

$$\frac{|M^{(0)}_{X+1}(s_{i1}, s_{1k}, s)|^2}{|M^{(0)}_X(s)|^2} = 4\pi\alpha_s C_F \left(\frac{2s_{ik}}{s_{i1}s_{1k}} + \ldots \right)$$

by Leading-Log limit of QCD → universal dominant terms

$$\frac{\mathrm{d}s_{i1}\mathrm{d}s_{1k}}{s_{i1}s_{1k}} \longrightarrow \frac{\mathrm{d}p_\perp^2}{p_\perp^2}\frac{\mathrm{d}z}{z(1-z)} \longrightarrow \frac{\mathrm{d}E_1}{\min(E_i, E_1)}\frac{\mathrm{d}\theta_{i1}}{\theta_{i1}} \quad (E_1 \ll E_i,\ \theta_{i1} \ll 1) \qquad ,\ldots$$

Rewritings in soft/collinear limits

"smallest" k_T (or p_T or θ_{ij}, or …) → largest Eikonal

Types of Algorithms

2. "Cone" type

Take your 4-vectors. Select a procedure for which "test cones" to draw

- Different names for different procedures
  - Seeded : start from hardest 4-vectors (and possibly combinations thereof, e.g., CDF midpoint algorithm) = "seeds"
  - Unseeded : smoothly scan over entire event, trying everything
- Sum momenta inside test cone → new test cone direction
- Iterate until stable (test cone direction = momentum sum direction)

Warning: seeded algorithms are INFRARED UNSAFE

Infrared Safety

Definition

An observable is infrared safe if it is <u>in</u>sensitive to

<u>SOFT radiation:</u>

Adding any number of infinitely soft particles (zero-energy) should not change the value of the observable

<u>COLLINEAR radiation:</u>

Splitting an existing particle up into two comoving particles (conserving the total momentum and energy) should not change the value of the observable

Safe vs Unsafe Jets

May look pretty similar in experimental environment …

But it’s not nice to your theory friends …

Unsafe: badly divergent in pQCD → large IR corrections:

$$\text{IR Sensitive Corrections} \;\propto\; \alpha_s^n \log^m\left(\frac{Q^2_{\mathrm{UV}}}{Q^2_{\mathrm{IR}}}\right) \quad , \quad m \le 2n$$

Even if we have a hadronization model with which to compute these corrections, the dependence on it → larger uncertainty

Safe → IR corrections power suppressed:

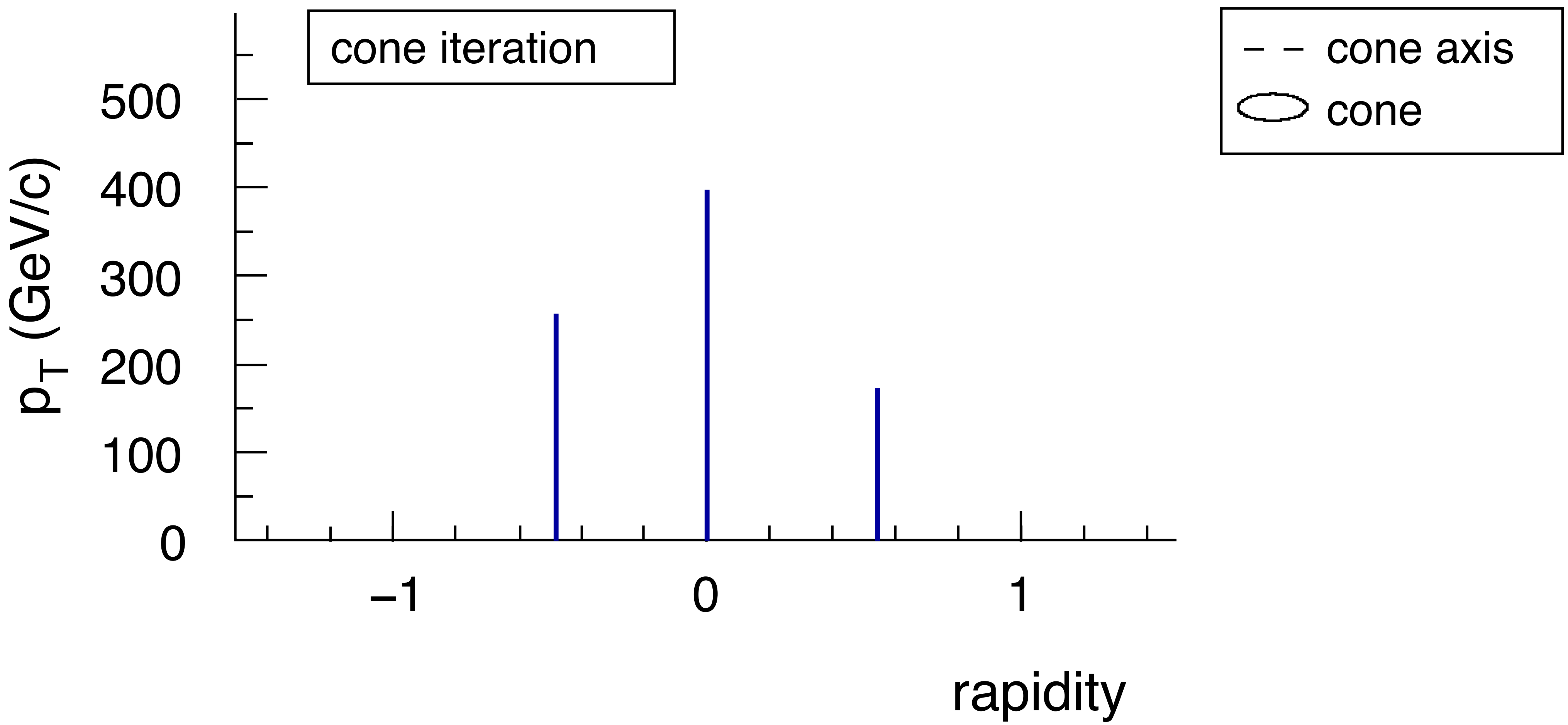
$$\text{IR Safe Corrections} \;\propto\; \frac{Q^2_{\mathrm{IR}}}{Q^2_{\mathrm{UV}}}$$

Can still be computed (MC) but can also be neglected (pure pQCD)

Let’s look at a specific example …

ICPR iteration issue

Iterative Cone Progressive Removal

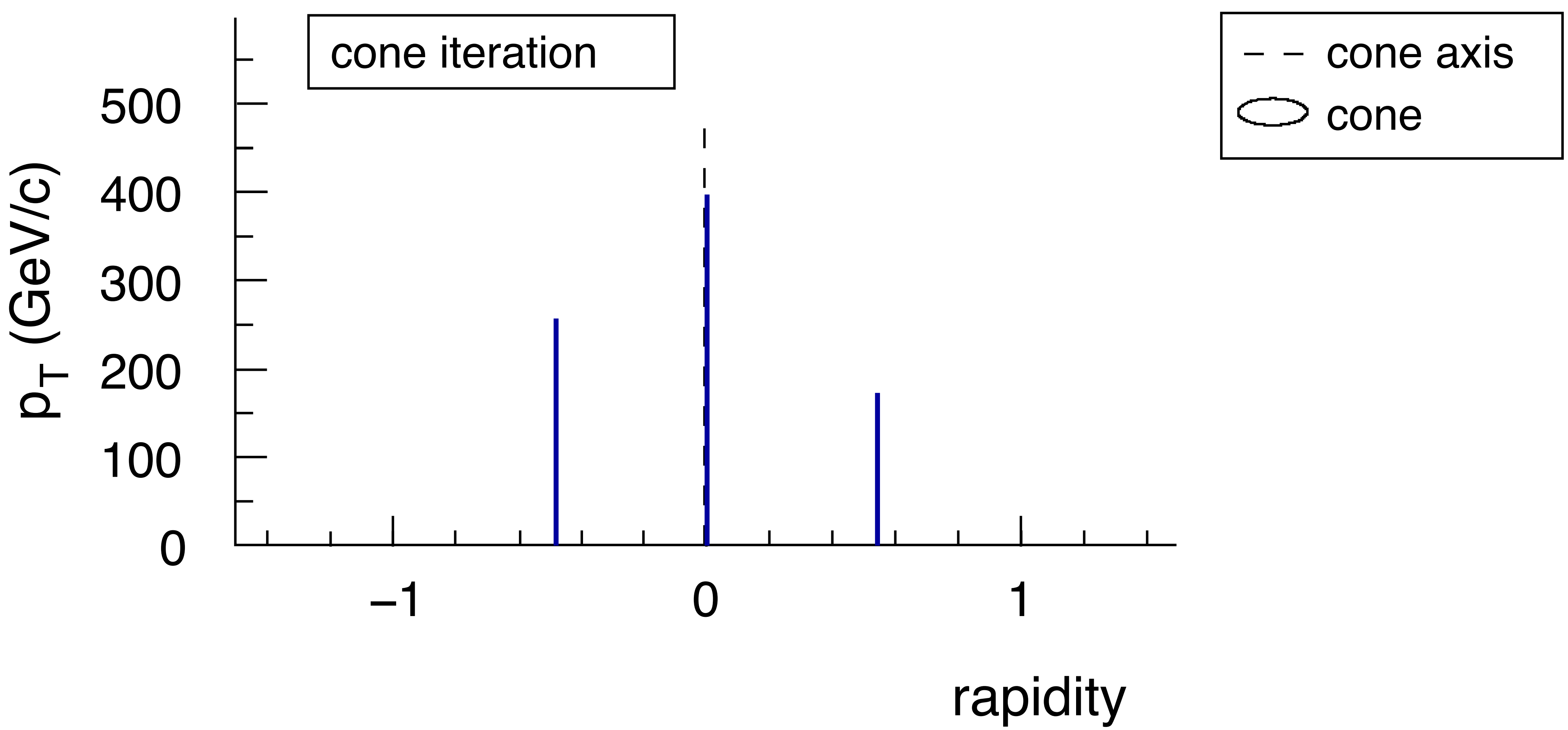


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

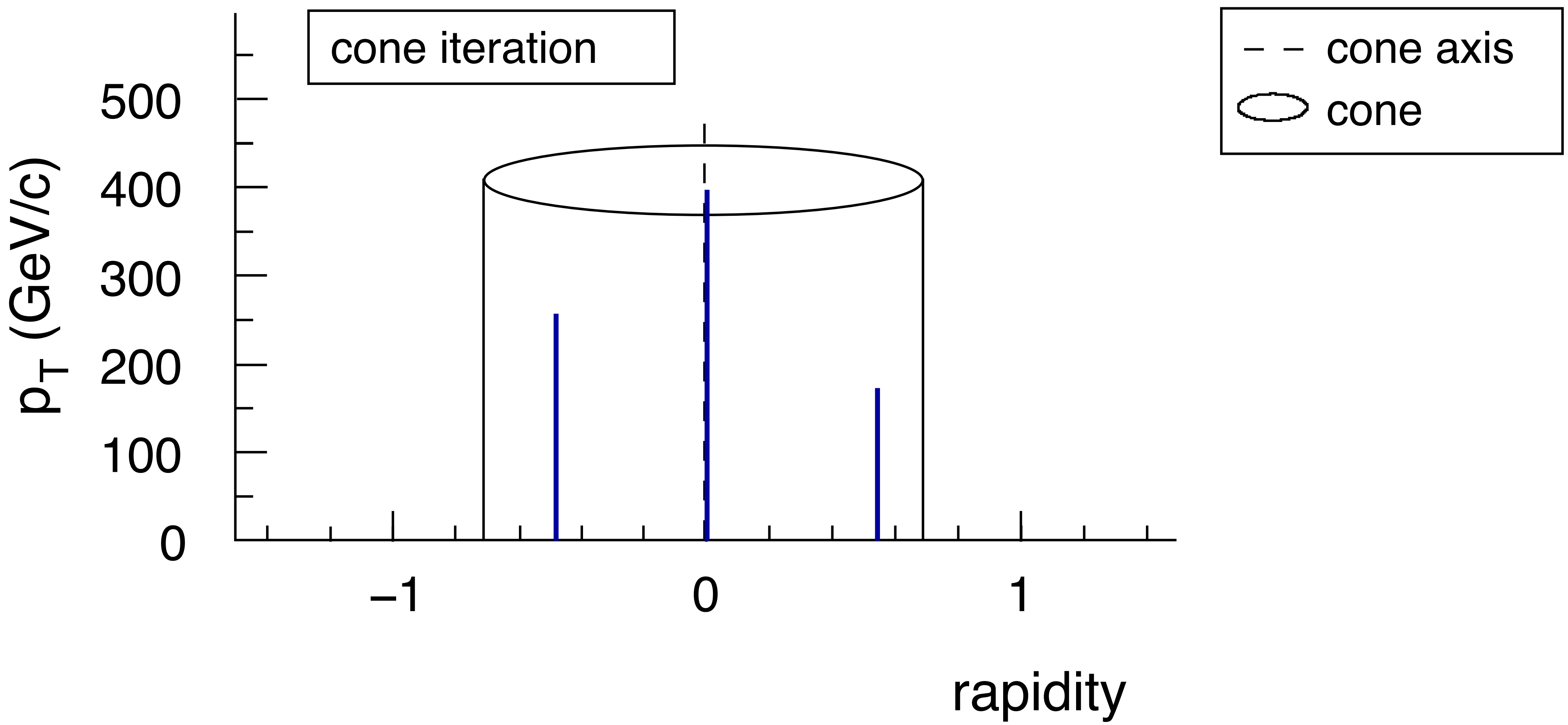


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

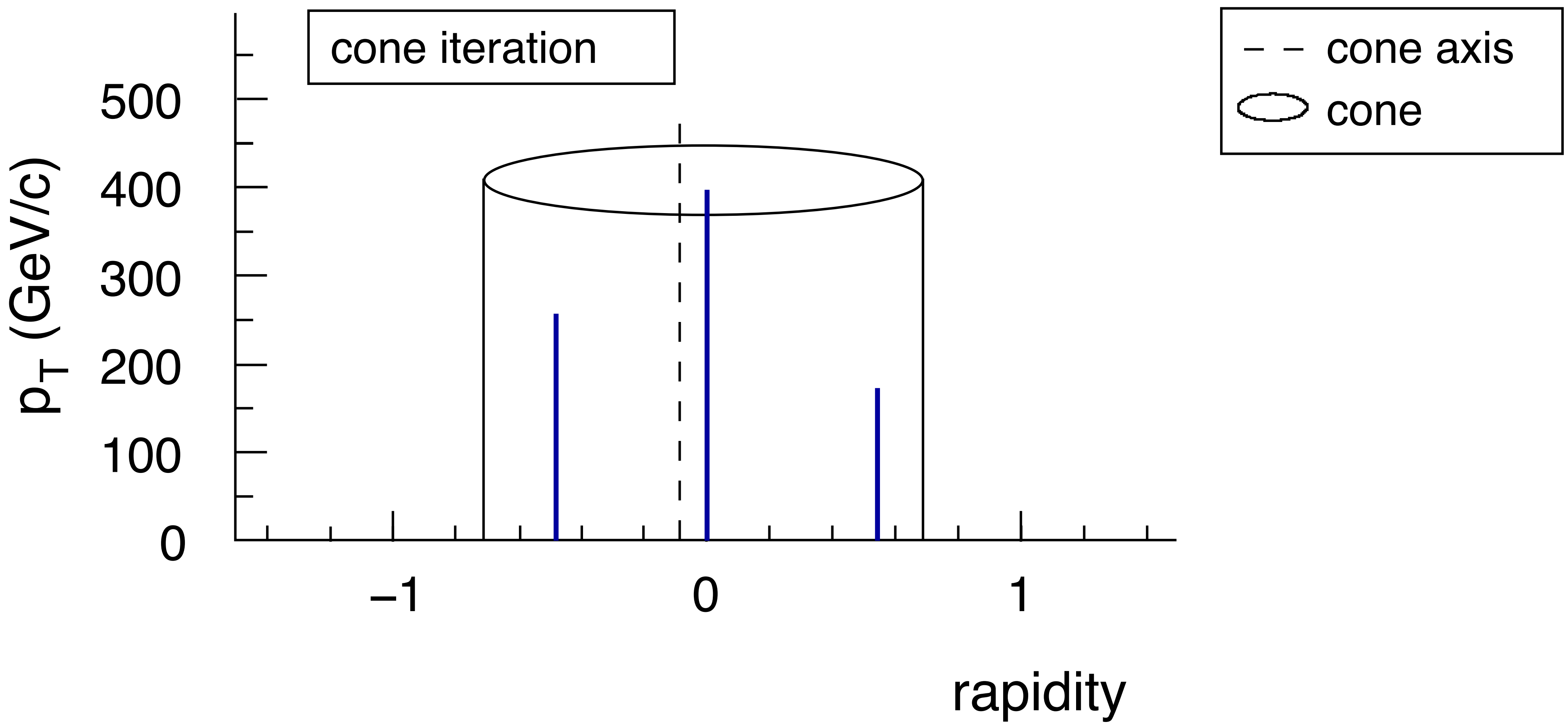
ICPR iteration issue

Iterative Cone Progressive Removal



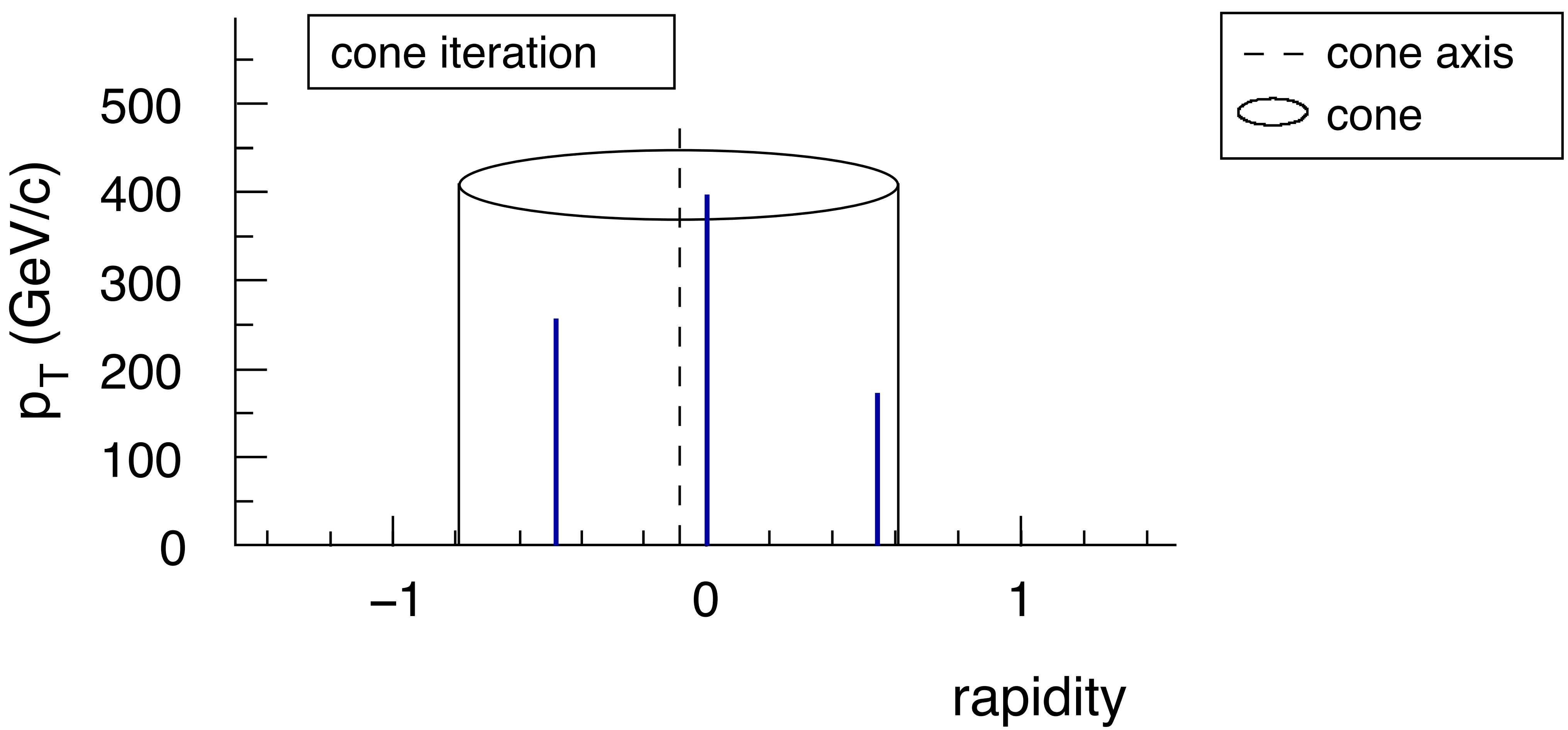
Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam



Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

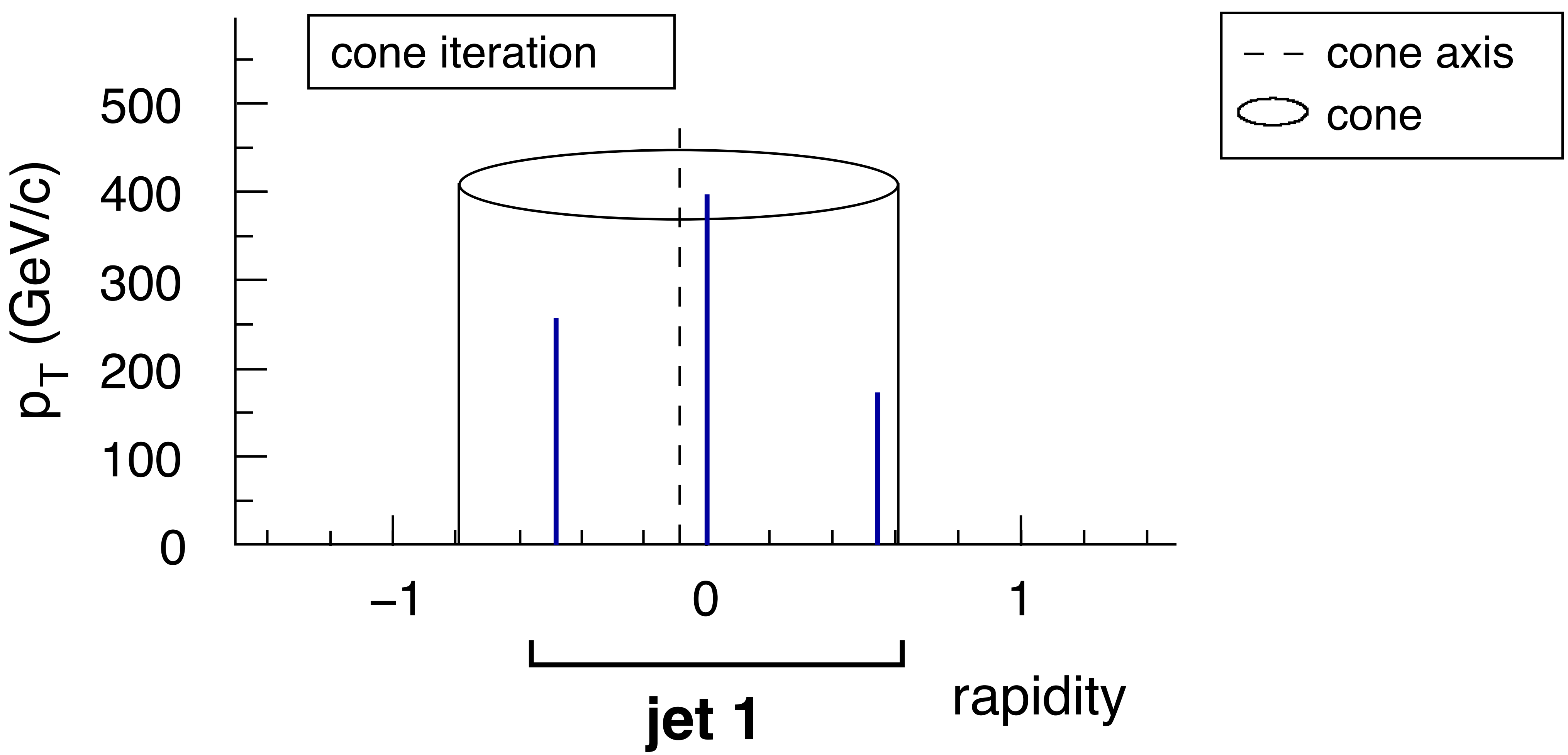


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

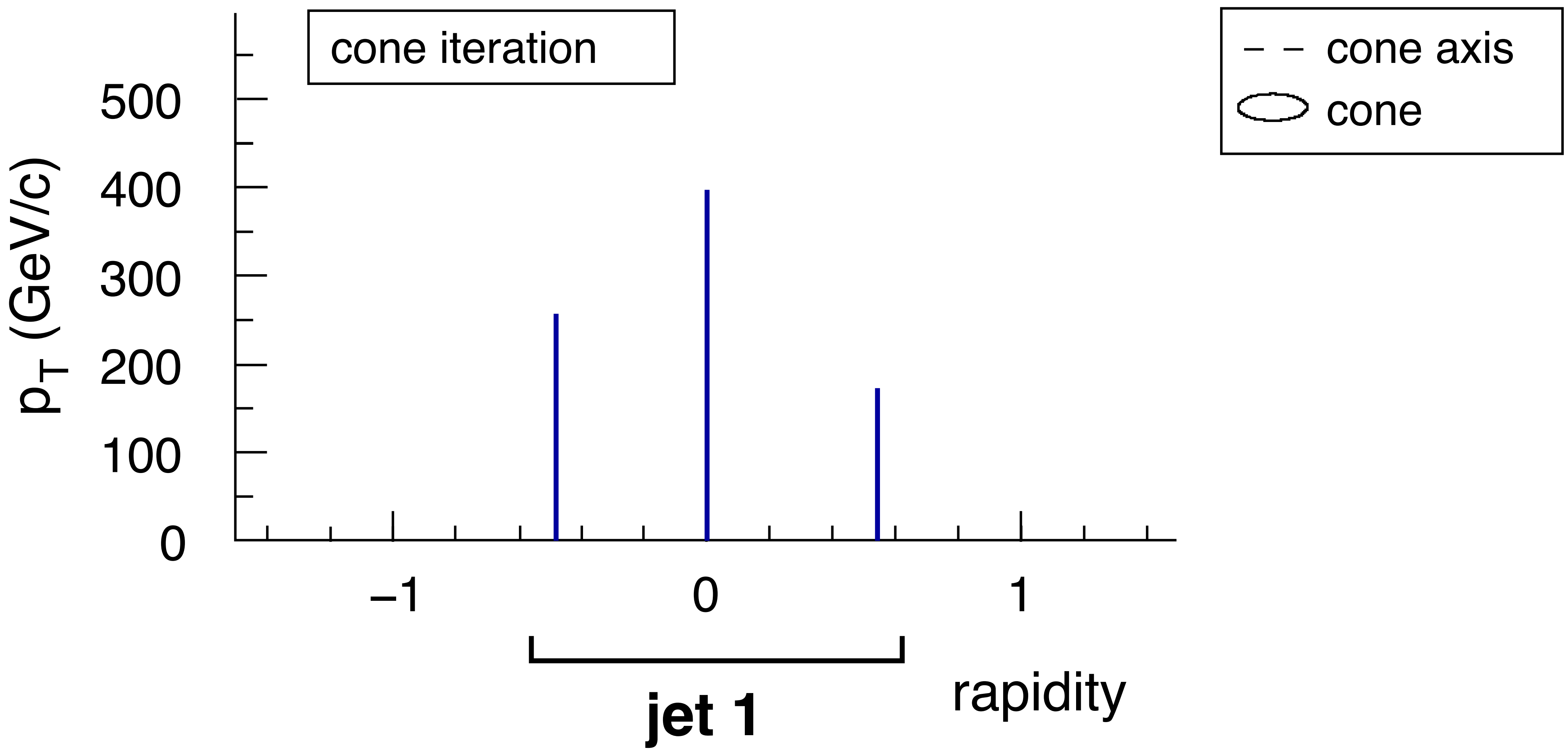


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

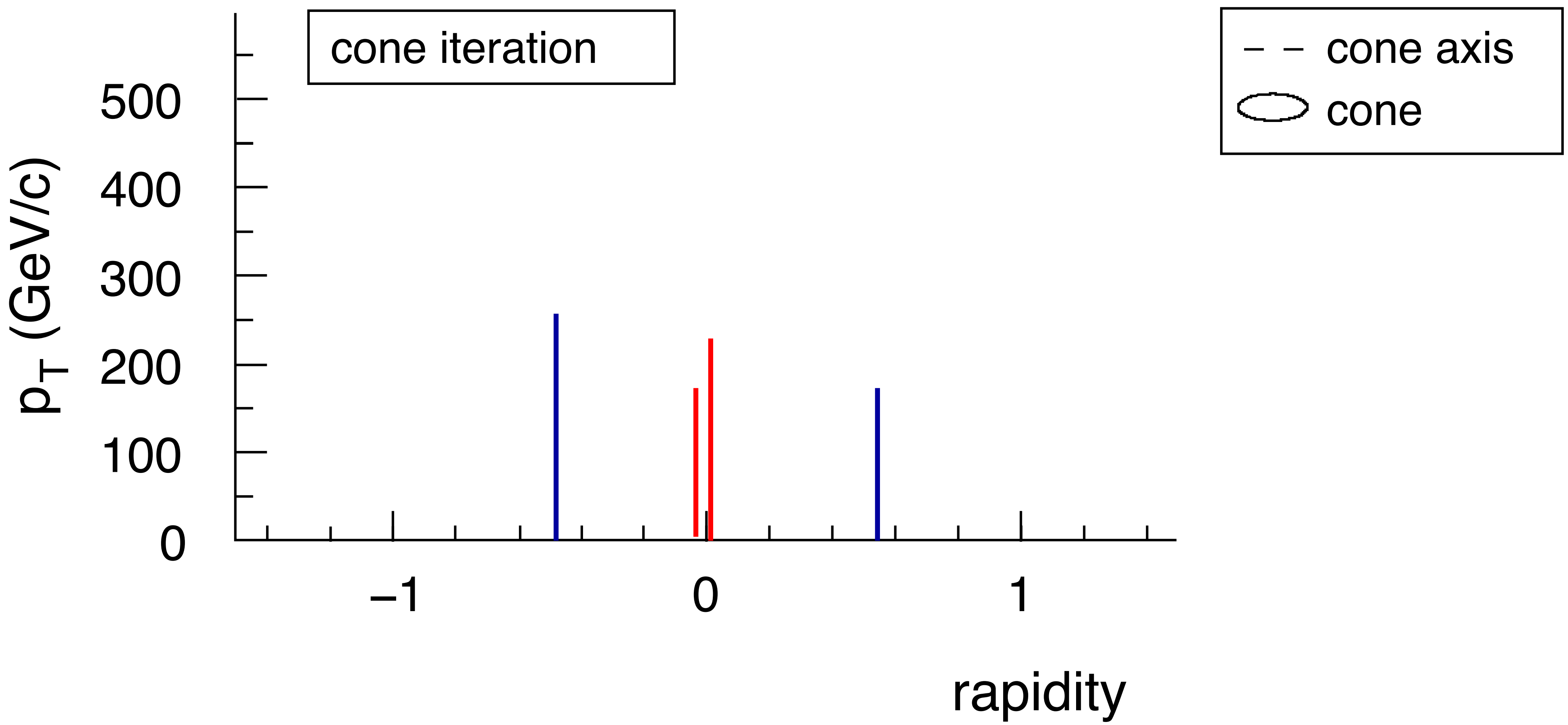


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

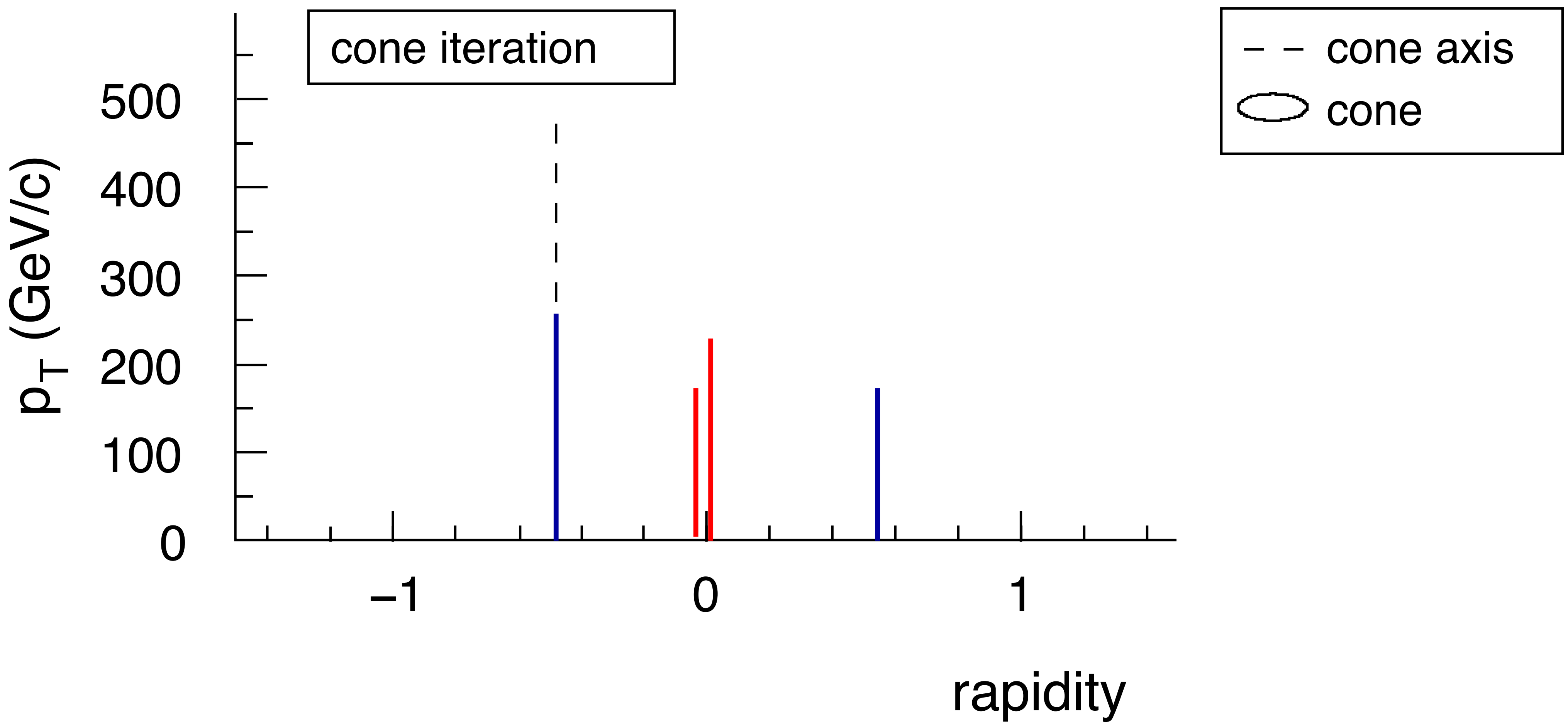
ICPR iteration issue

Iterative Cone Progressive Removal



Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

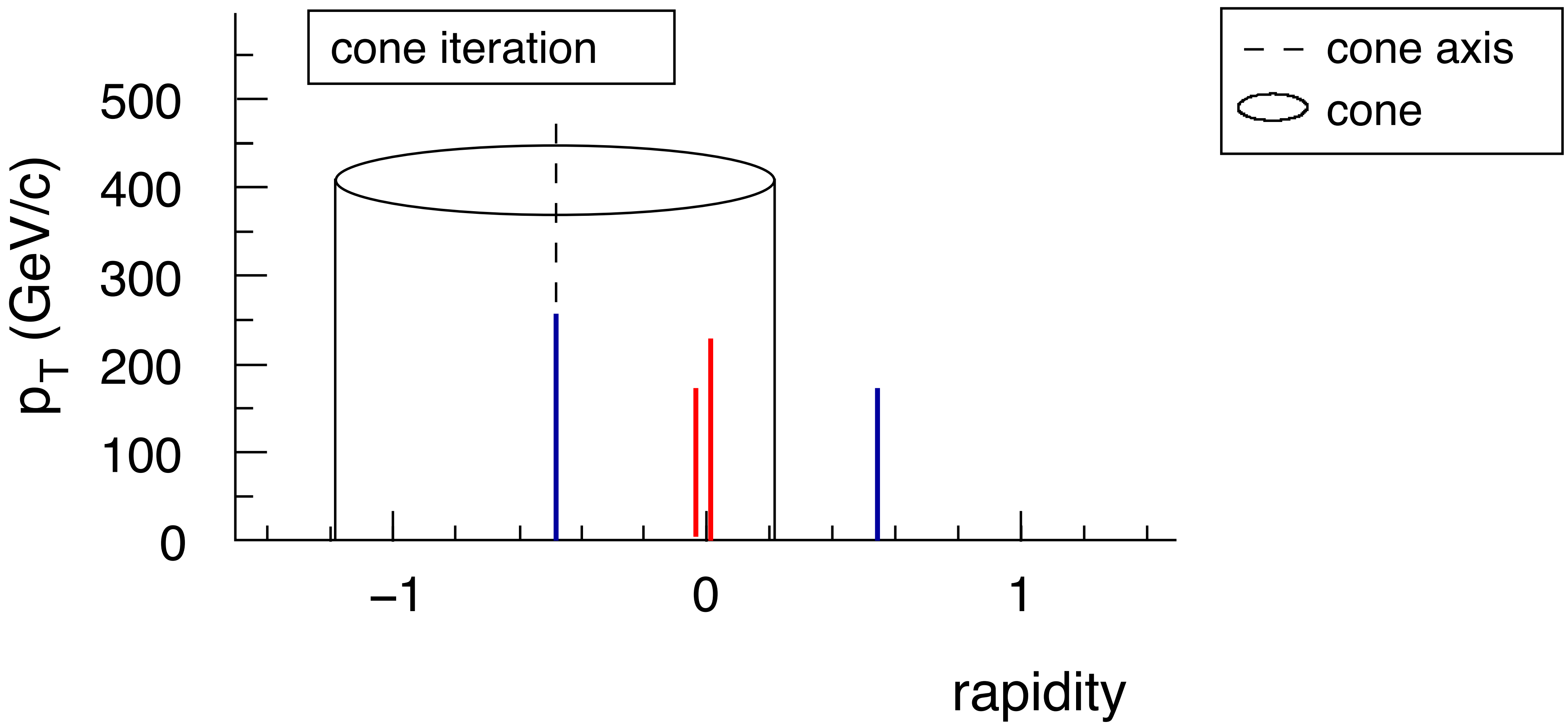


Collinear splitting can modify the hard jets: ICPR algorithms are **collinear unsafe** $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

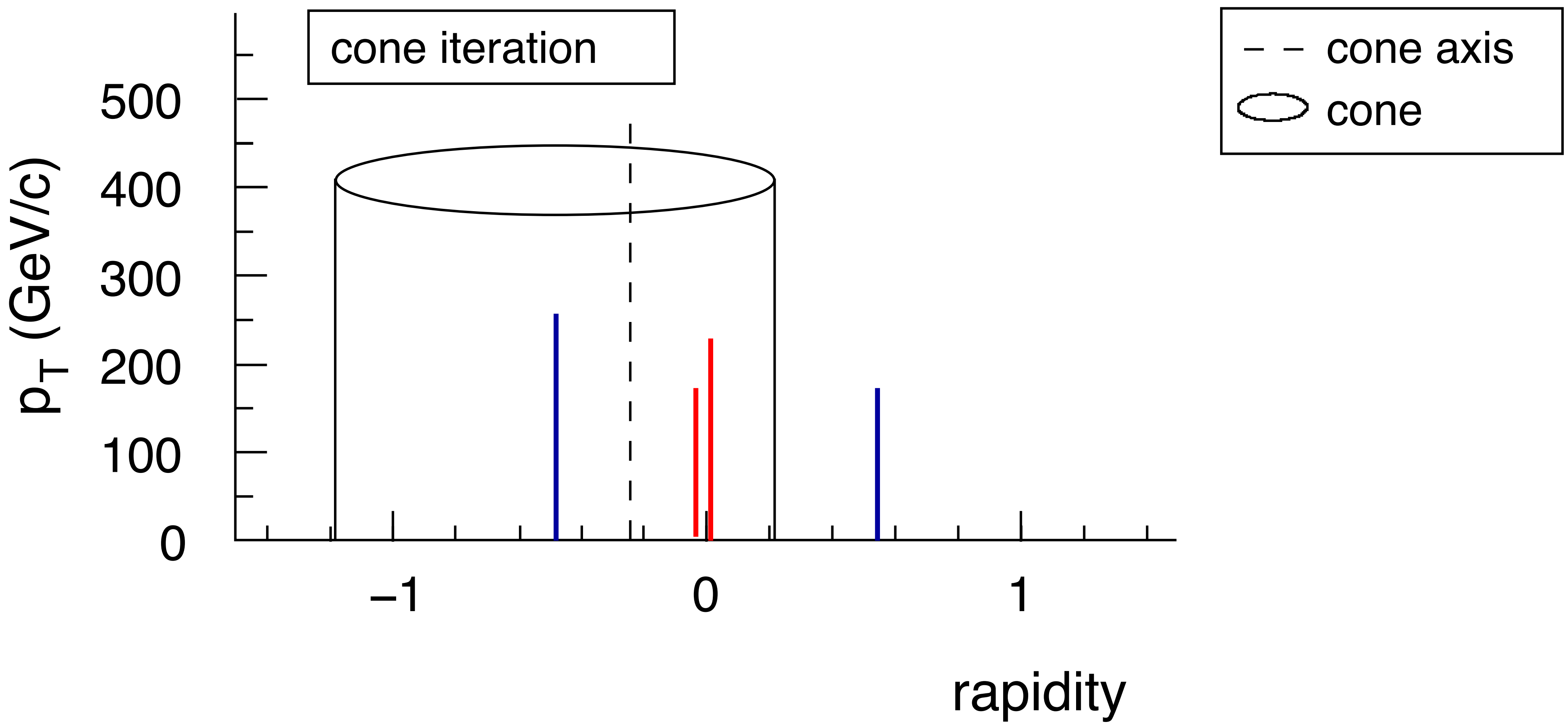


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

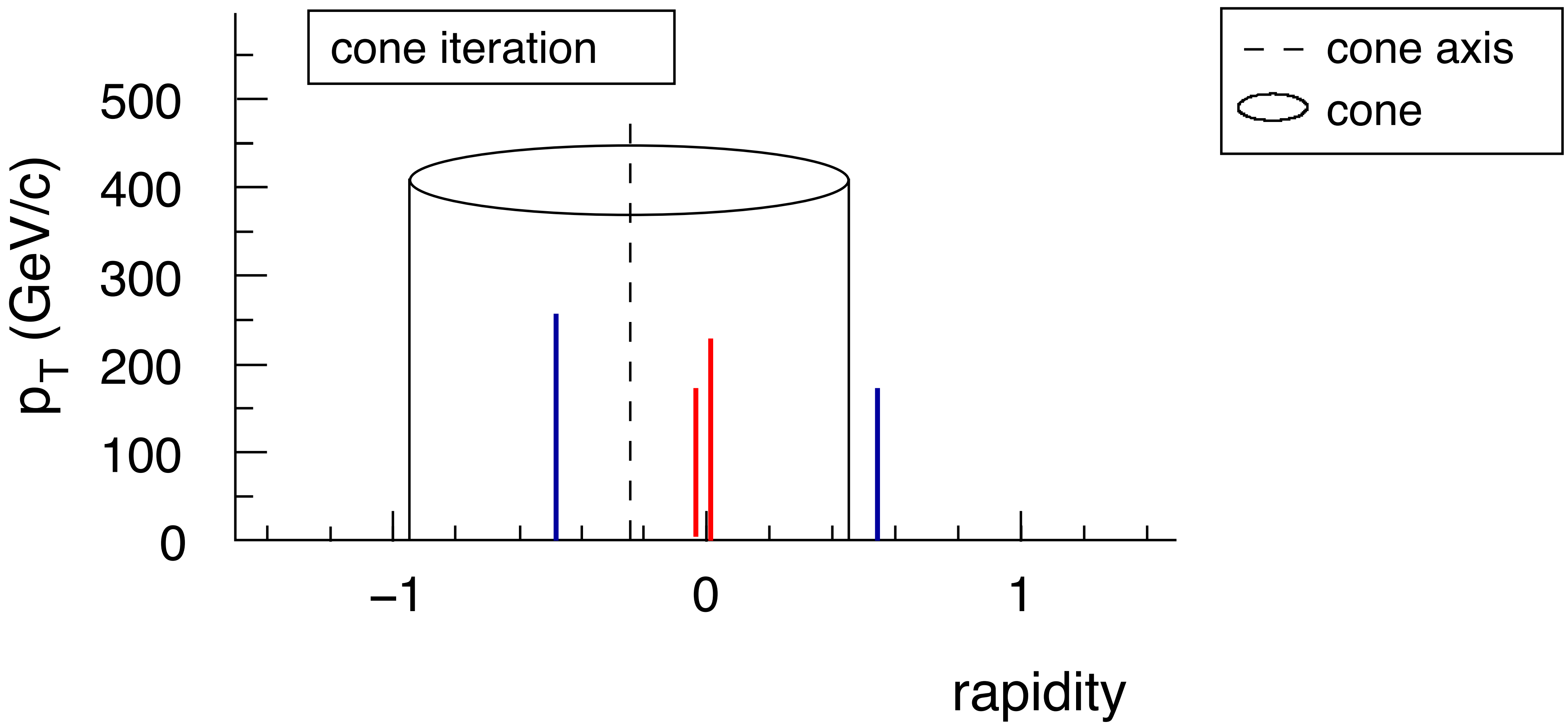


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

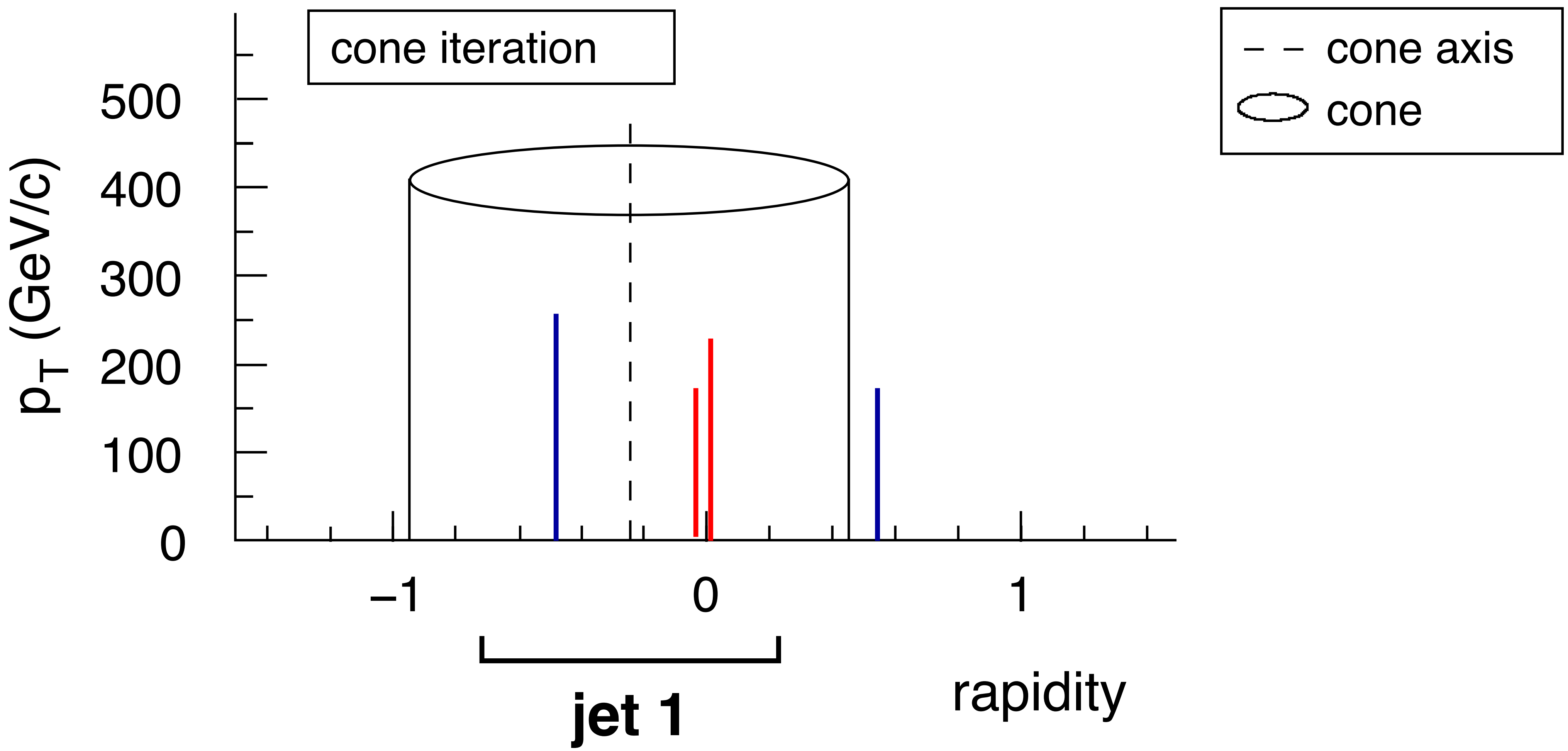


Collinear splitting can modify the hard jets: ICPR algorithms are **collinear unsafe** $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

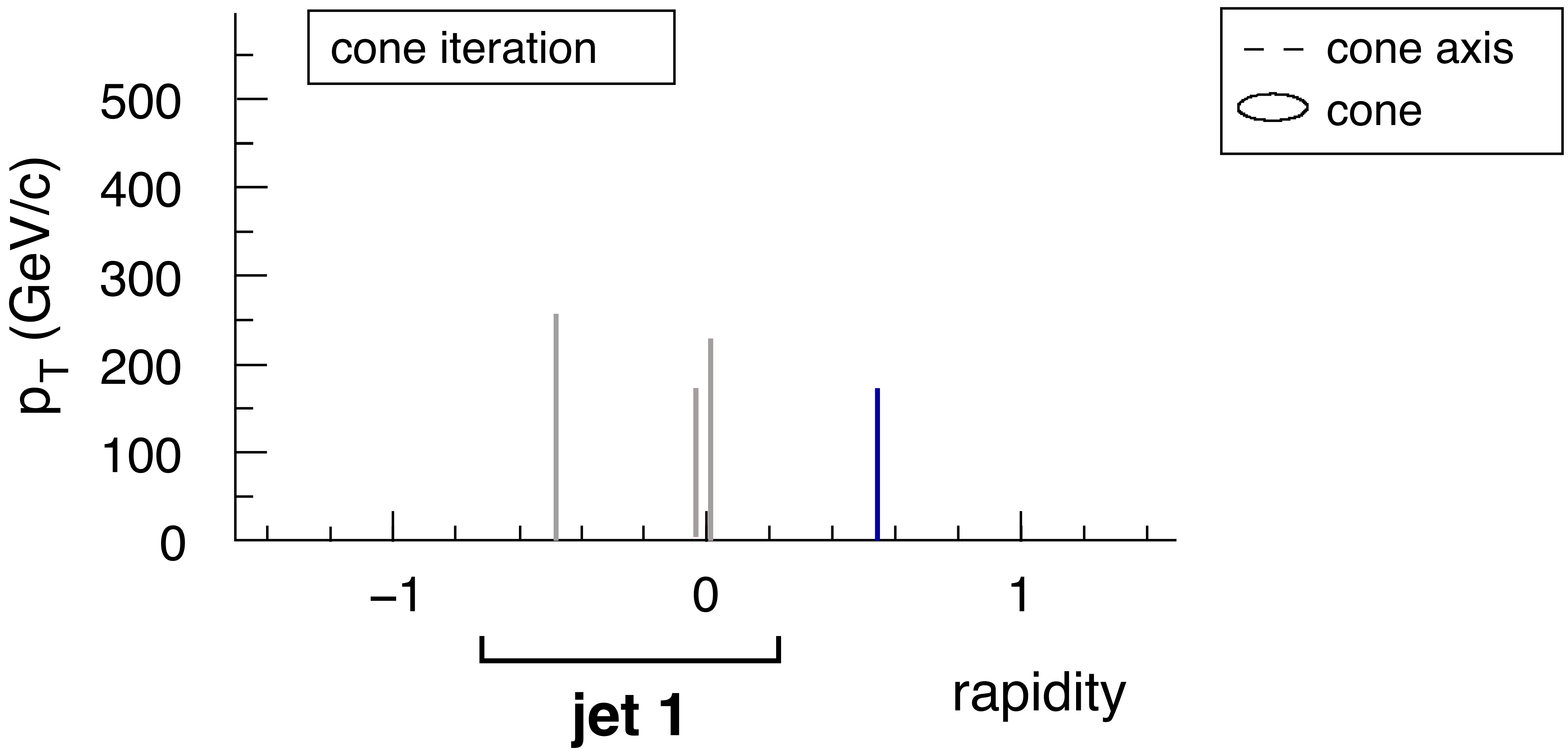


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

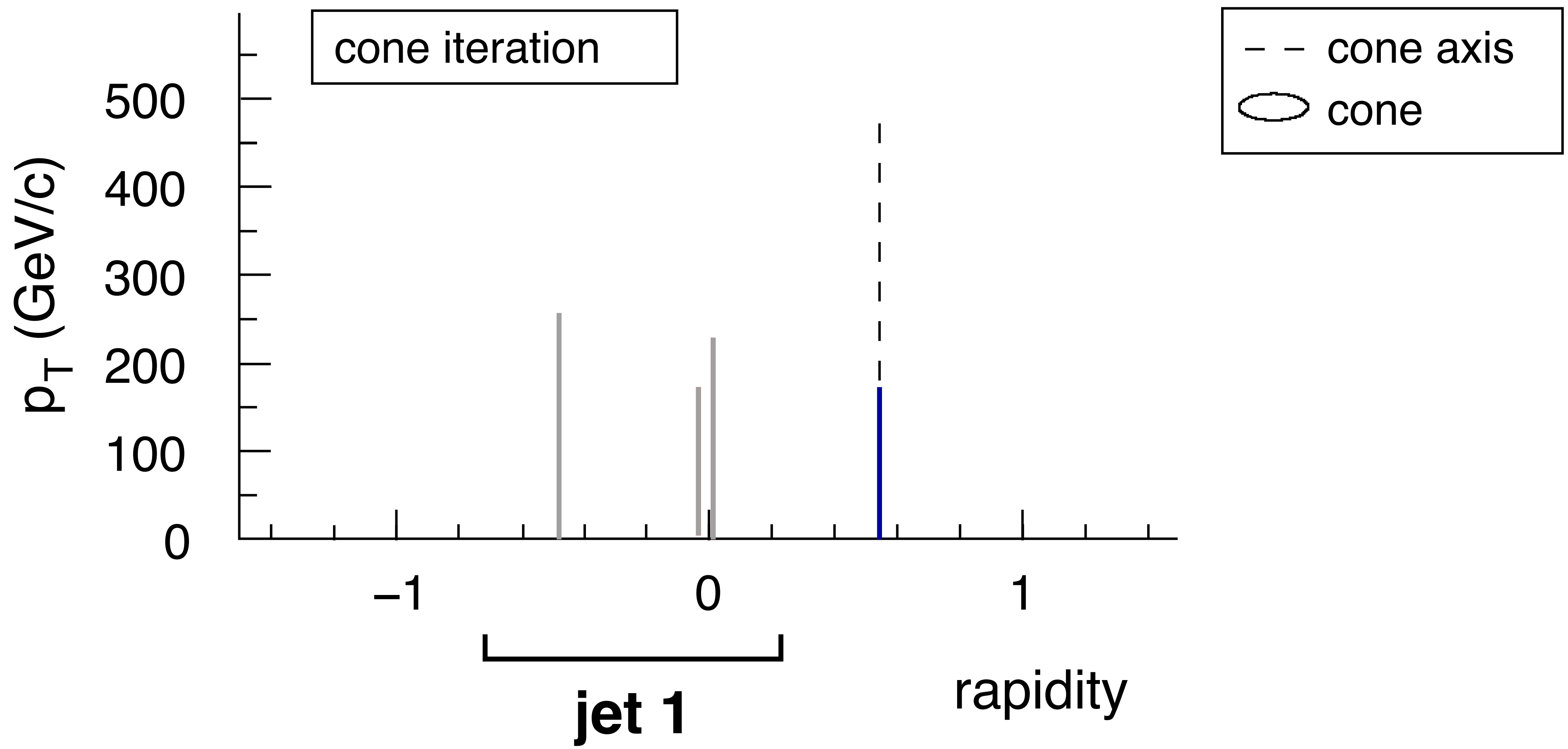
Iterative Cone Progressive Removal



Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue
Iterative Cone Progressive Removal

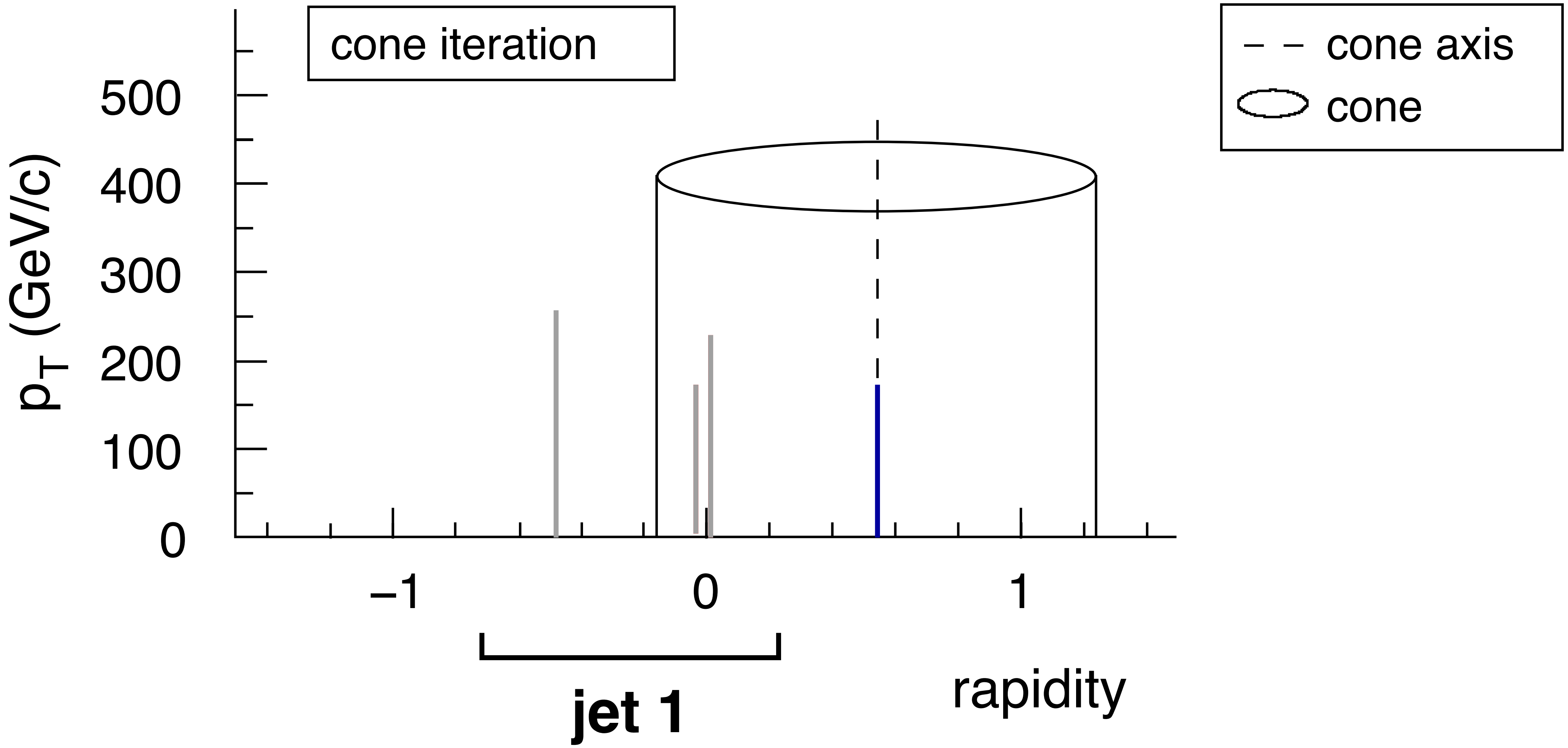


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

ICPR iteration issue

Iterative Cone Progressive Removal

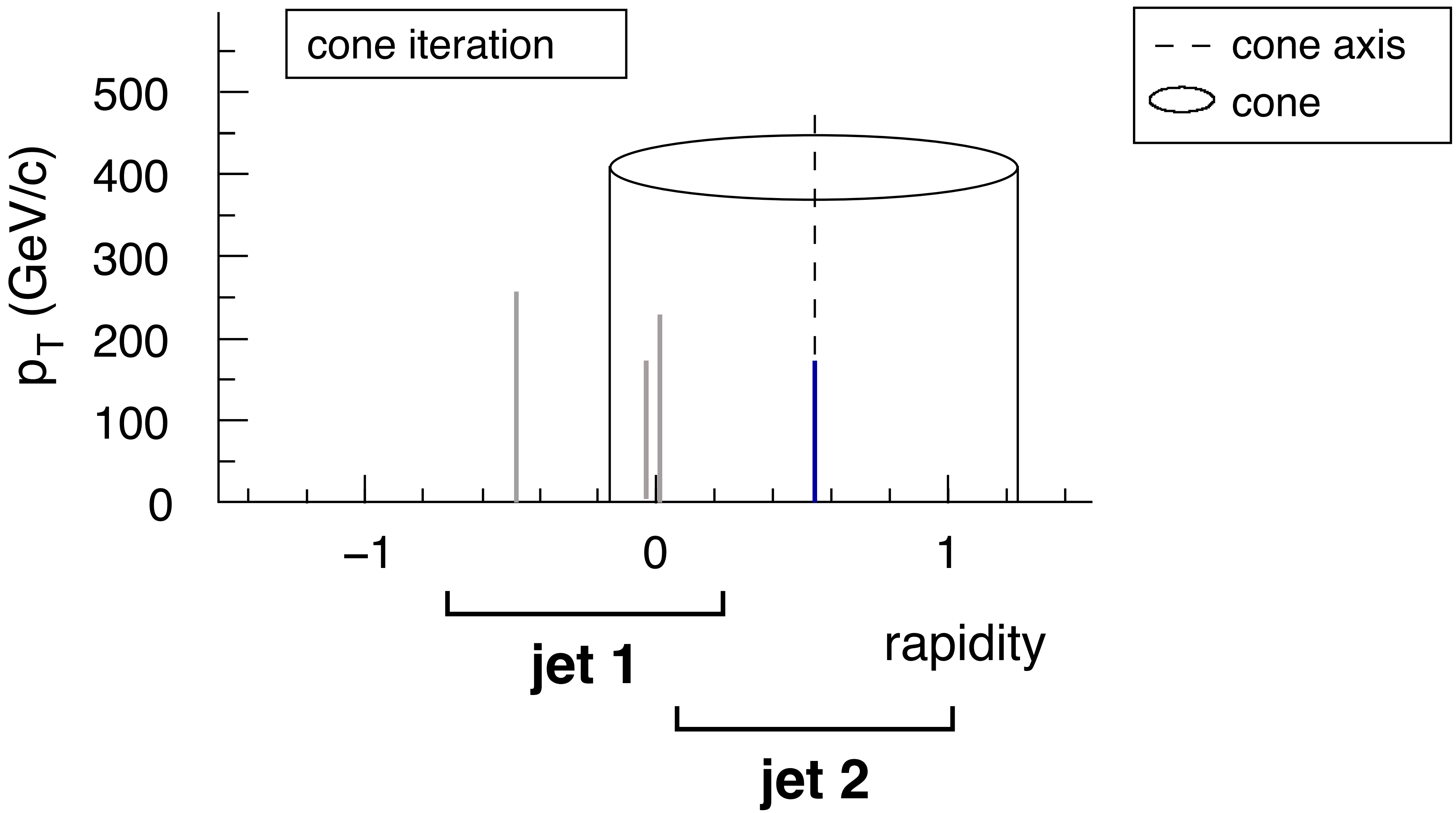


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

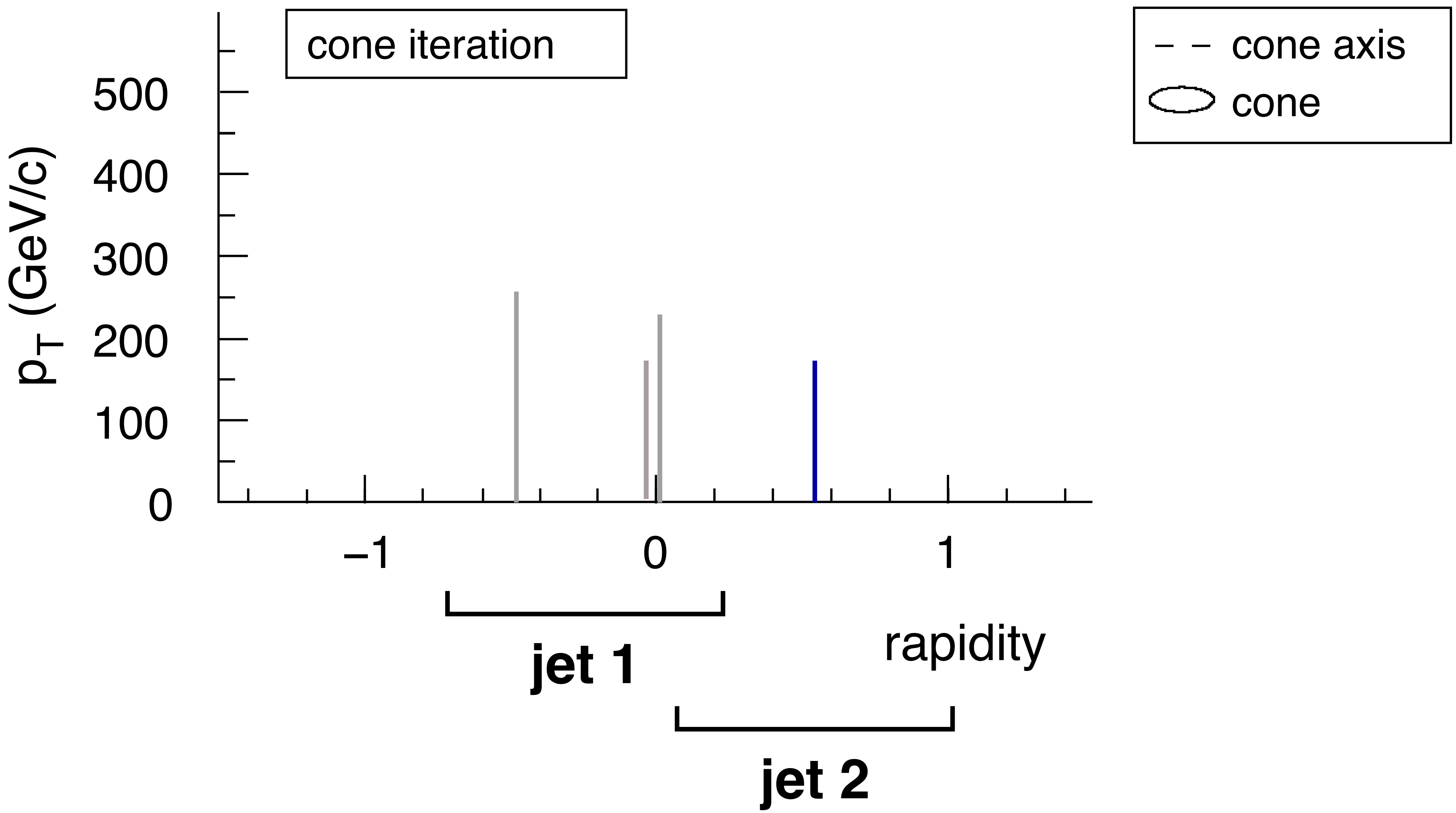
ICPR iteration issue

Iterative Cone Progressive Removal



Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

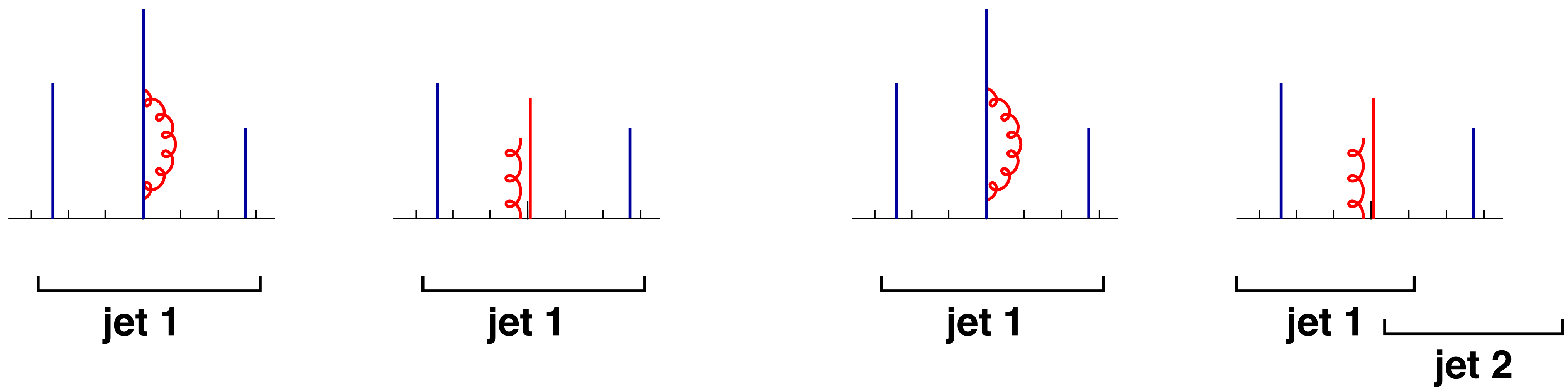


Collinear splitting can modify the hard jets: ICPR algorithms are collinear unsafe $\Longrightarrow$ perturbative calculations give ∞

Slides from G. Salam

Consequences of Collinear Unsafety

Collinear Safe | **Collinear Unsafe**



$\alpha_s^n \times (-\infty)$ $\alpha_s^n \times (+\infty)$ $\alpha_s^n \times (-\infty)$ $\alpha_s^n \times (+\infty)$

Infinities cancel | **Infinities do not cancel**

Invalidates perturbation theory

Real life does not have infinities, but pert. infinity leaves a real-life trace

$$\alpha_s^2 + \alpha_s^3 + \alpha_s^4 \times \infty \to \alpha_s^2 + \alpha_s^3 + \alpha_s^4 \times \ln p_t/\Lambda \to \alpha_s^2 + \underbrace{\alpha_s^3 + \alpha_s^3}_{\text{BOTH WASTED}}$$

Stereo Vision

Use IR Safe algorithms

http://www.fastjet.fr/

To study short-distance physics

These days, ≈ as fast as IR unsafe algos and widely implemented (e.g., FASTJET), including

"Cone-like": SiSCone, Anti-k_T, …

"Recombination-like": k_T,Cambridge/Aachen,Anti-k_T…


Image Credits: Richard Seaman

Then use IR Sensitive observables

E.g., number of tracks, identified particles, …

To explicitly check hadronization and models of IR physics

More about IR in lecture on soft QCD …

Introduction to QCD

1. Fundamentals of QCD
2. PDFs, Fixed-Order QCD, and Jet Algorithms
3. Parton Showers and Event Generators
4. QCD in the Infrared

Slides posted at:
www.cern.ch/skands/slides

Lecture Notes:
P. Skands, arXiv:1207.2389

Supplementary Slides

Uncalculated Orders

Naively O(α_s) - True in e^+e^- !

$$\sigma_{\mathrm{NLO}}(e^+e^- \to q\bar{q}) = \sigma_{\mathrm{LO}}(e^+e^- \to q\bar{q}) \left(1 + \frac{\alpha_s(E_{\mathrm{CM}})}{\pi} + \mathcal{O}(\alpha_s^2)\right)$$

Generally larger in hadron collisions

Typical "K" factor in pp (= $\sigma_{\mathrm{NLO}}/\sigma_{\mathrm{LO}}$) ≈ 1.5 ± 0.5

Why is this? Many pseudoscientific explanations

Explosion of # of diagrams ($n_{\mathrm{Diagrams}} \approx n!$)
New initial states contributing at higher orders (E.g., gq → Zq)
Inclusion of low-x (non-DGLAP) enhancements
Bad (high) scale choices at Lower Orders, …

Theirs not to reason why // Theirs but to do and die

Tennyson, The Charge of the Light Brigade

Changing the scale(s)

Why scale variation ~ uncertainty?

Scale dependence of calculated orders must be canceled by contribution from uncalculated ones (+ non-pert)

$$\alpha_s(Q^2) = \alpha_s(m_Z^2)\frac{1}{1 + b_0\ \alpha_s(m_Z)\ln\frac{Q^2}{m_Z^2} + \mathcal{O}(\alpha_s^2)}$$

$$b_0 = \frac{11N_C - 2n_f}{12\pi}$$

→ $$\left(\alpha_s(Q'^2) - \alpha_s(Q^2)\right)|M|^2 = \alpha_s^2(Q^2)|M|^2 + \ldots$$

→ Generates terms of higher order, but proportional to what you already have ($|M|^2$)→ a first naive* way to estimate uncertainty

*warning: some theorists believe it is the only way … but be agnostic! There are other things than scale dependence …

(Factorization: Caveats)

1. The proof only includes the first term in an operator product expansion in "twist" = mass dimension - spin

→ Strictly speaking, only valid for $Q^2 \to \infty$. Neglects corrections of order

$$\text{Higher Twist :} \quad \frac{\left[\ln\left(Q^2/\Lambda^2\right)\right]^{m<2n}}{Q^{2n}} \qquad \text{(n=2 for DIS)}$$

2. The proof only applies to inclusive cross sections

In e^+e^- , in DIS, and in Drell-Yan. For everything else: factorization *ansatz*

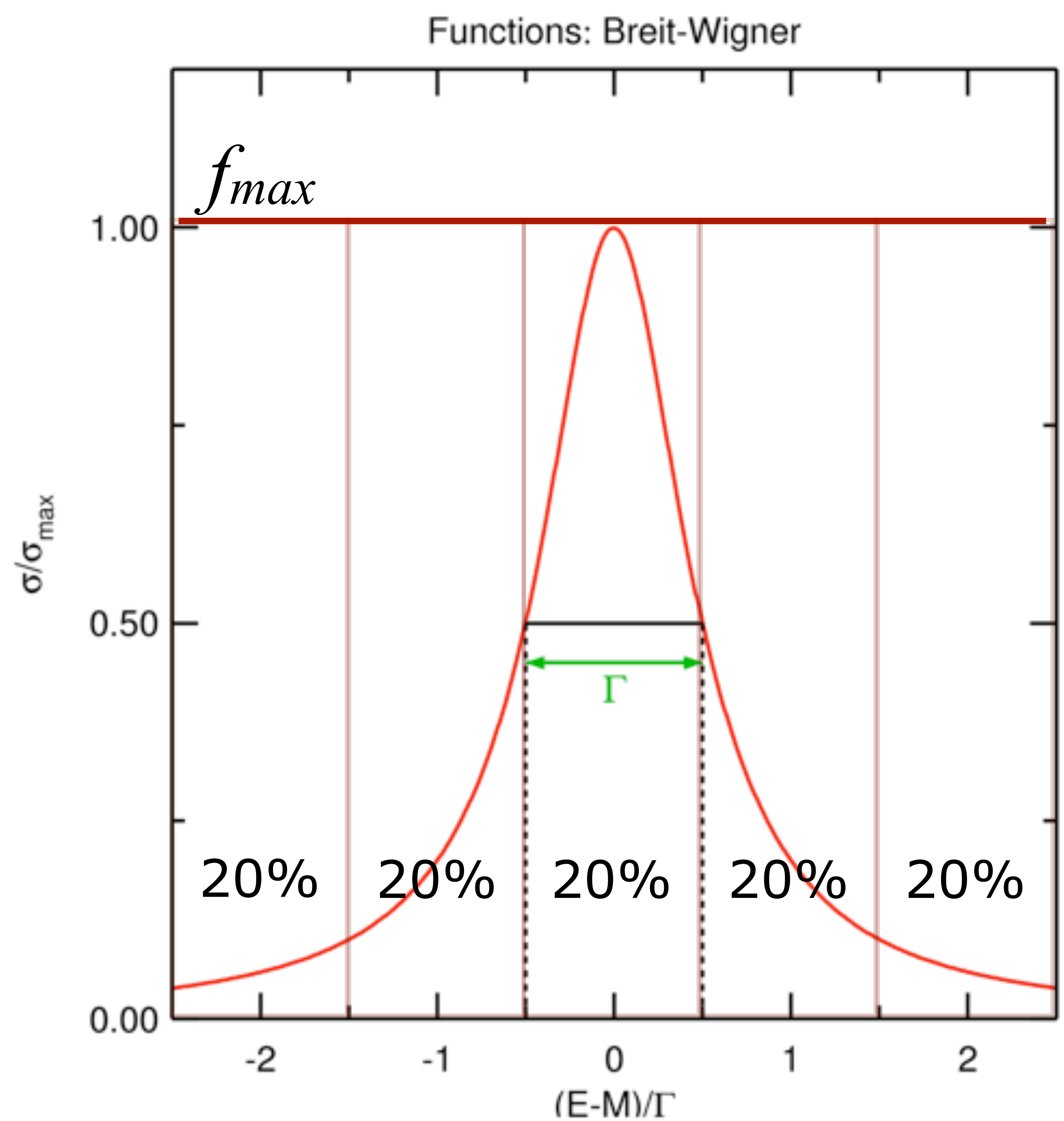
3. Scheme dependence

In practice limited to MSbar + variations of Q_F

4. Interpretation of PDFs as parton number densities

Is only valid at Leading Order

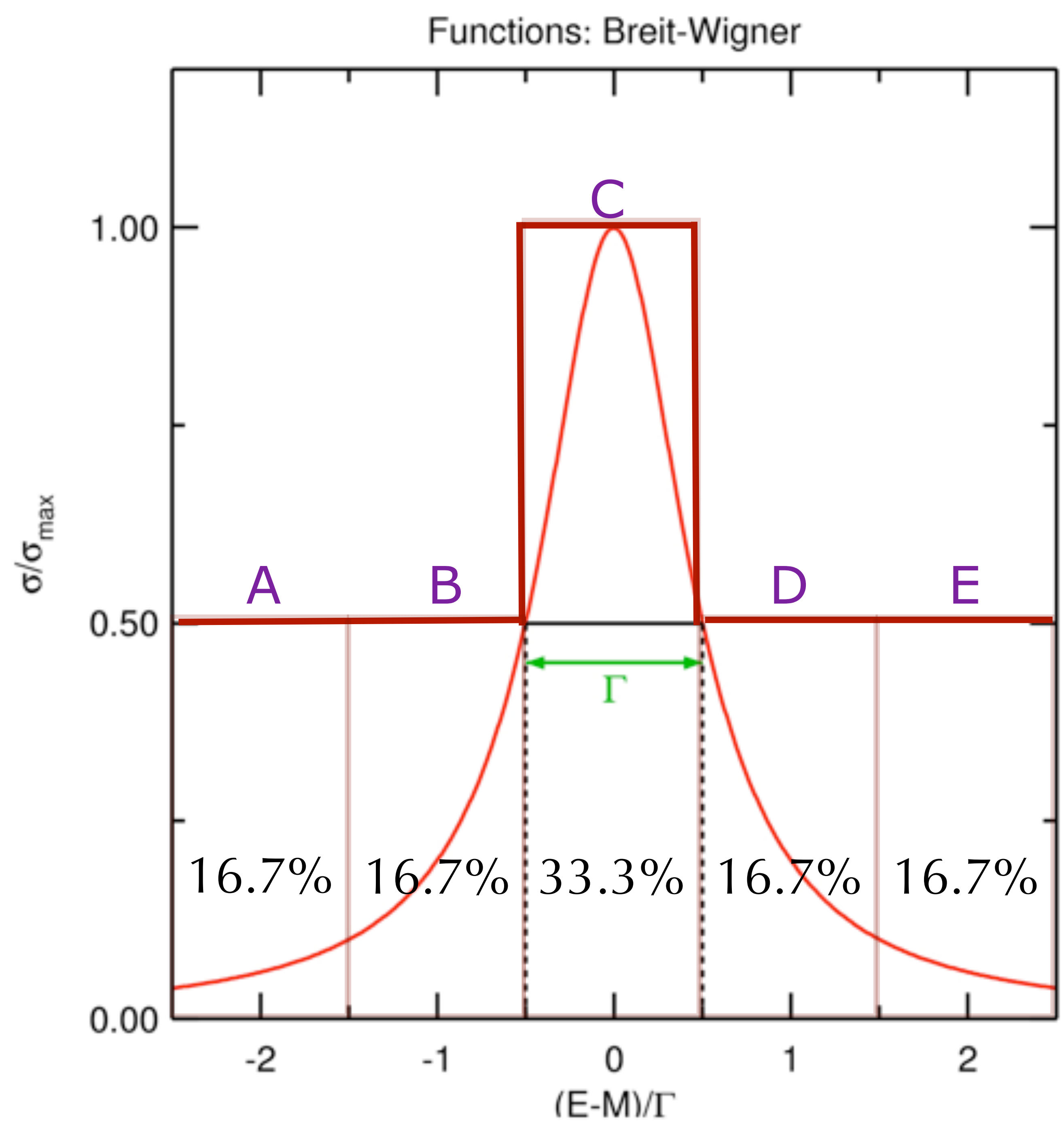
Peaked Functions



Precision on integral dominated by the points with f ≈ f_{max} *(i.e., peak regions)*

→ slow convergence if high, narrow peaks

Stratified Sampling



→ Make it twice as likely to throw points in the peak

Choose:

For: $6 \cdot R_1 \in$

[0,1] → Region A

[1,2] → Region B

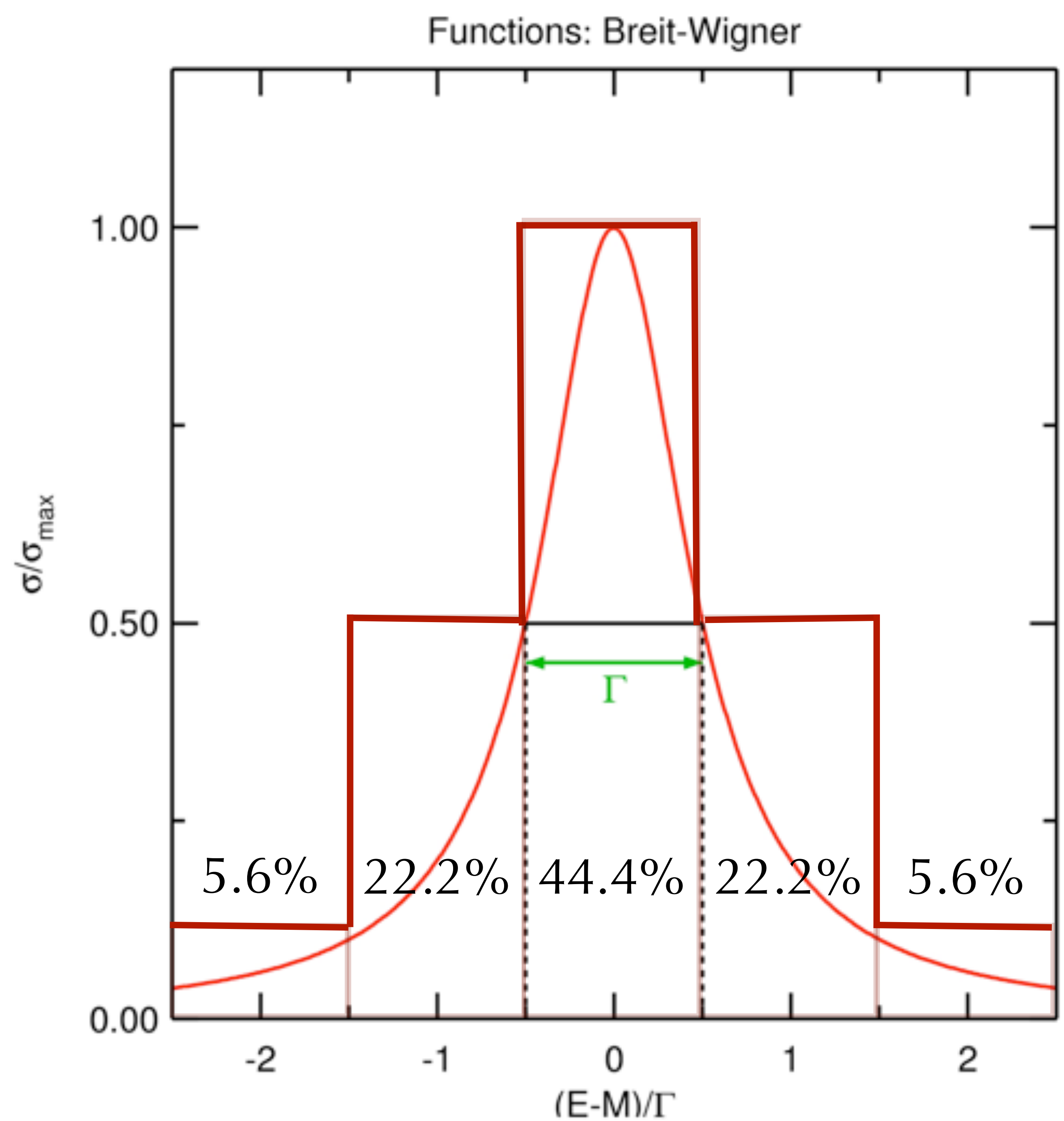
[2,4] → Region C

[4,5] → Region D

[5,6] → Region E

→ faster convergence for same number of function evaluations

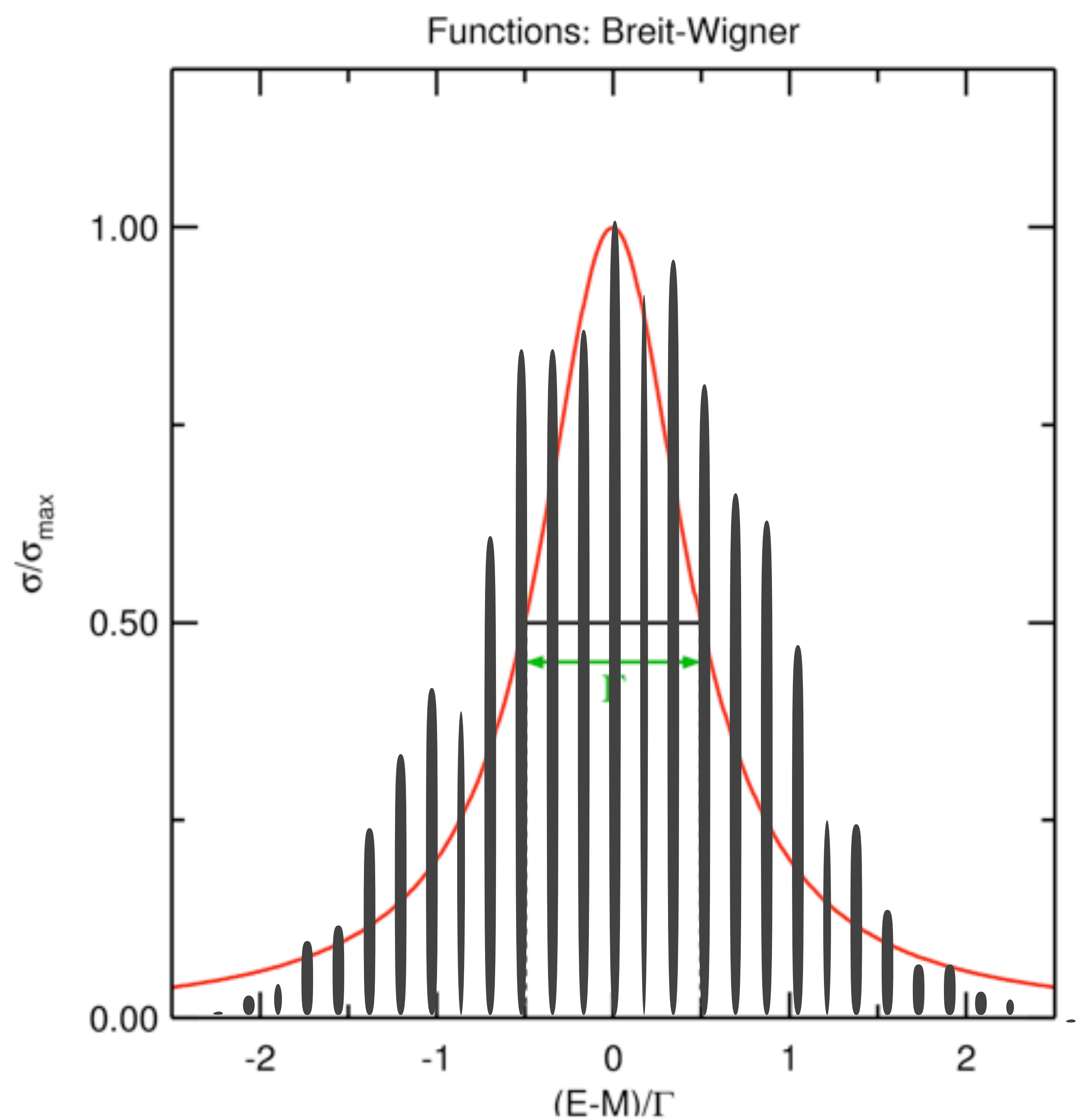
Adaptive Sampling



→ Can even design algorithms to do this automatically as they run (not covered here)

→ Adaptive sampling

Importance Sampling



→ or throw points according to some smooth peaked function for which you have, or can construct, a random number generator (here: Gauss)

E.g., VEGAS algorithm, by G. Lepage

Why does this work?

1) You are inputting knowledge: obviously need to know where the peaks are to begin with … (say you know, e.g., the location and width of a resonance)

2) Stratified sampling increases efficiency by combining fixed-grid methods with the MC method, with further gains from adaptation

3) Importance sampling:

$$\int_a^b f(x)\mathrm{d}x = \int_a^b \frac{f(x)}{g(x)}\mathrm{d}G(x)$$

Effectively does flat MC with changed integration variables

Fast convergence if $f(x)/g(x) \approx 1$

How we do Monte Carlo

Take your system

- Set of radioactive nuclei
- Set of hard scattering processes
- Set of resonances that are going to decay
- Set of particles coming into your detector
- Set of cosmic photons traveling across the galaxy
- Set of molecules
- …

How we do Monte Carlo

Take your system

Generate a "trial" (event/decay/interaction/...)

Not easy to generate random numbers distributed according to exactly the right distribution?

May have complicated dynamics, interactions ...

→ use a simpler "trial" distribution

Flat with some stratification

Or importance sample with simple overestimating function (for which you can generate random #s)

How we do Monte Carlo

Take your system

Generate a "trial" (event/decay/interaction/...)

Accept trial with probability f(x)/g(x)

f(x) contains all the complicated dynamics

g(x) is the simple trial function

If accept: replace with new system state

If reject: keep previous system state

no dependence on g in final result - only affects convergence rate

And keep going: generate next trial ...

How we do Monte Carlo

Take your system

Generate a “trial” (event/decay/in

Accept trial with probability f(x)/g(

f(x) contains all the complicated

g(x) is the simple trial function

If accept: replace with new system

If reject: keep previous system stat

no dependence on g ir
result - only affec
convergence rate

Sounds deceptively simple, but …

with it, you can integrate arbitrarily complicated functions *(in particular chains of nested functions)*, over arbitrarily complicated regions, in arbitrarily many dimensions …

And keep going: