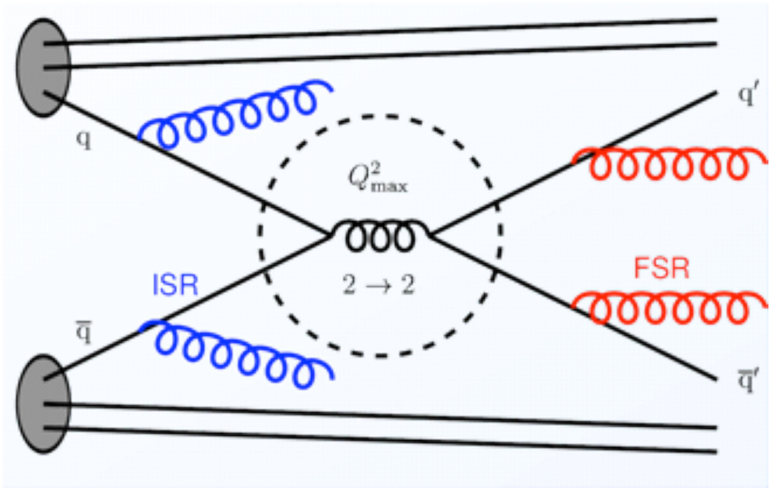
General-Purpose Event Generators



Calculate Everything ≈ solve QCD → requires compromise!

Improve lowest-order perturbation theory,
by including the 'most significant' corrections
→ complete events (can evaluate any observable you want)

The Workhorses

PYTHIA : Successor to JETSET (begun in 1978). Originated in hadronization studies: Lund String.
HERWIG : Successor to EARWIG (begun in 1984). Originated in coherence studies: angular ordering.
SHERPA : Begun in 2000. Originated in "matching" of matrix elements to showers: CKKW-L.
+ MORE SPECIALIZED: ALPGEN, MADGRAPH, HELAC, ARIADNE, VINCIA, WHIZARD, (a)MC@NLO, POWHEG, HEJ, PHOJET, EPOS, QGSJET, SIBYLL, DPMJET, LDCMC, DIPSY, HIJING, CASCADE, GOSAM, BLACKHAT, ...

(PYTHIA)

PYTHIA anno 1978

(then called JETSET)

LU TP 78-18
November, 1978

A Monte Carlo Program for Quark Jet Generation

T. Sjöstrand, B. Söderberg

A Monte Carlo computer program is presented, that simulates the fragmentation of a fast parton into a jet of mesons. It uses an iterative scaling scheme and is compatible with the jet model of Field and Feynman.

Note:
Field-Feynman was an early fragmentation model
Now superseded by the String (in PYTHIA) and Cluster (in HERWIG & SHERPA) models.

(PYTHIA)

PYTHIA anno 1978

(then called JETSET)

LU TP 78-18
November, 1978

A Monte Carlo Program for Quark Jet Generation

T. Sjöstrand, B. Söderberg

A Monte Carlo computer program is presented, that simulates the fragmentation of a fast parton into a jet of mesons. It uses an iterative scaling scheme and is compatible with the jet model of Field and Feynman.

Note:
Field-Feynman was an early fragmentation model
Now superseded by the String (in PYTHIA) and Cluster (in HERWIG & SHERPA) models.

```
      SUBROUTINE JETGEN(N)
      COMMON /JET/ K(100,2), P(100,5)
      COMMON /PAR/ PUD, PS1, SIGMA, CX2, EBEG, WFIN, IFLBEG
      COMMON /DATA1/ MESO(9,2), CMIX(6,2), PMAS(19)
      IFLSGN=(10-IFLBEG)/5
      W=2.*EBEG
      I=0
      IPD=0
C 1 FLAVOUR AND PT FOR FIRST QUARK
      IFL1=IABS(IFLBEG)
      PT1=SIGMA*SQRT(-ALOG(RANF(0)))
      PHI1=6.2832*RANF(0)
      PX1=PT1*COS(PHI1)
      PY1=PT1*SIN(PHI1)
  100 I=I+1
C 2 FLAVOUR AND PT FOR NEXT ANTIQUARK
      IFL2=1+INT(RANF(0)/PUD)
      PT2=SIGMA*SQRT(-ALOG(RANF(0)))
      PHI2=6.2832*RANF(0)
      PX2=PT2*COS(PHI2)
      PY2=PT2*SIN(PHI2)
C 3 MESON FORMED, SPIN ADDED AND FLAVOUR MIXED
      K(I,1)=MESO(3*(IFL1-1)+IFL2,IFLSGN)
      ISPIN=INT(PS1+RANF(0))
      K(I,2)=1+9*ISPIN+K(I,1)
      IF(K(I,1).LE.6) GOTO 110
      TMIX=RANF(0)
      KM=K(I,1)-6+3*ISPIN
      K(I,2)=8+9*ISPIN+INT(TMIX+CMIX(KM,1))+INT(TMIX+CMIX(KM,2))
C 4 MESON MASS FROM TABLE, PT FROM CONSTITUENTS
  110 P(I,5)=PMAS(K(I,2))
      P(I,1)=PX1+PX2
      P(I,2)=PY1+PY2
      PMTS=P(I,1)**2+P(I,2)**2+P(I,5)**2
C 5 RANDOM CHOICE OF X=(E+PZ)MESON/(E+PZ)AVAILABLE GIVES E AND PZ
      X=RANF(0)
      IF(RANF(0).LT.CX2) X=1.-X**(1./3.)
      P(I,3)=(X*W-PMTS/(X*W))/2.
      P(I,4)=(X*W+PMTS/(X*W))/2.
C 6 IF UNSTABLE, DECAY CHAIN INTO STABLE PARTICLES
  120 IPD=IPD+1
      IF(K(IPD,2).GE.8) CALL DECAY(IPD,I)
      IF(IPD.LT.I.AND.I.LE.96) GOTO 120
C 7 FLAVOUR AND PT OF QUARK FORMED IN PAIR WITH ANTIQUARK ABOVE
      IFL1=IFL2
      PX1=-PX2
      PY1=-PY2
C 8 IF ENOUGH E+PZ LEFT, GO TO 2
      W=(1.-X)*W
      IF(W.GT.WFIN.AND.I.LE.95) GOTO 100
      N=I
      RETURN
      END
```

(PYTHIA)

PYTHIA anno 2013

(now called PYTHIA 8)

~ 100,000 lines of C++

What a modern MC generator has inside:

LU TP 07-28 (CPC 178 (2008) 852)
October, 2007

A Brief Introduction to PYTHIA 8.1

T. Sjöstrand, S. Mrenna, P. Skands

The Pythia program is a standard tool for the generation of high-energy collisions, comprising a coherent set of physics models for the evolution from a few-body hard process to a complex multihadronic final state. It contains a library of hard processes and models for initial- and final-state parton showers, multiple parton-parton interactions, beam remnants, string fragmentation and particle decays. It also has a set of utilities and interfaces to external programs. […]

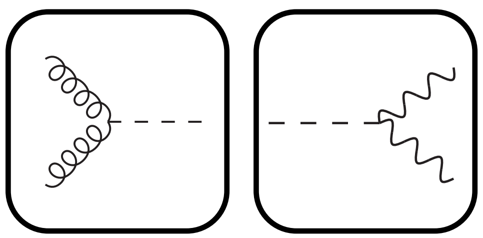
- Hard Processes (internal, interfaced, or via Les Houches events)
- BSM (internal or via interfaces)
- PDFs (internal or via interfaces)
- Showers (internal or inherited)
- Multiple parton interactions
- Beam Remnants
- String Fragmentation
- Decays (internal or via interfaces)
- Examples and Tutorial
- Online HTML / PHP Manual
- Utilities and interfaces to external programs

Divide and Conquer

Factorization → Split the problem into many (nested) pieces

+ Quantum mechanics → Probabilities → Random Numbers

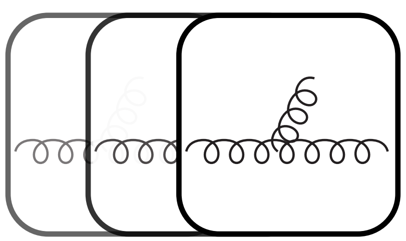
$$\mathcal{P}_{\rm event} = \mathcal{P}_{\rm hard} \otimes \mathcal{P}_{\rm dec} \otimes \mathcal{P}_{\rm ISR} \otimes \mathcal{P}_{\rm FSR} \otimes \mathcal{P}_{\rm MPI} \otimes \mathcal{P}_{\rm Had} \otimes \ldots$$



Hard Process & Decays:

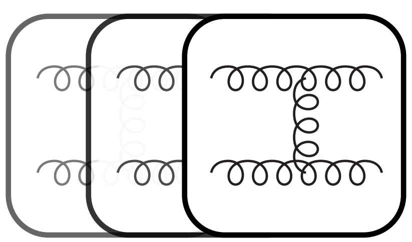
Use (N)LO matrix elements

→ Sets "hard" resolution scale for process: $Q_{\rm MAX}$



Initial- & Final-State Radiation (ISR & FSR):

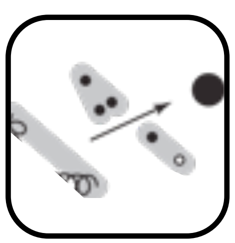
Altarelli-Parisi equations → differential evolution, dP/dQ^2, as function of resolution scale; run from $Q_{\rm MAX}$ to ~ 1 GeV (This Lecture)



MPI (Multi-Parton Interactions)

Additional (soft) parton-parton interactions: LO matrix elements

→ Additional (soft) "Underlying-Event" activity

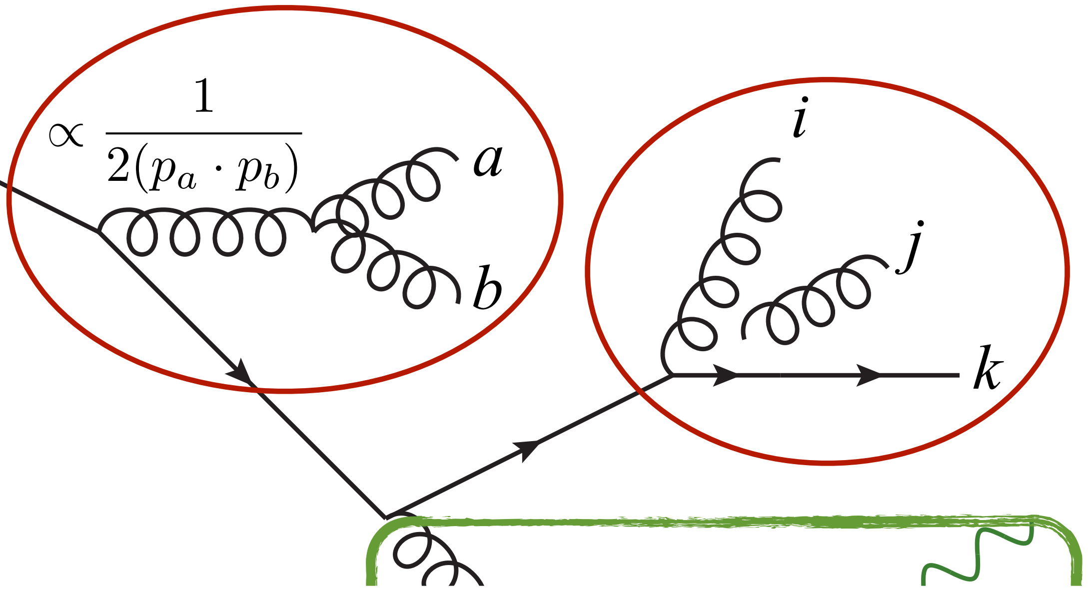


Hadronization

Non-perturbative model of color-singlet parton systems → hadrons

Recall: Jets ≈ Fractals

- **Most bremsstrahlung** is driven by divergent propagators → simple structure
- **Amplitudes factorize in singular limits** (→ universal "conformal" or "fractal" structure)



Partons ab → "collinear":

P(z) = DGLAP splitting kernels, with z = energy fraction = $E_a/(E_a+E_b)$

$$|\mathcal{M}_{F+1}(\ldots,a,b,\ldots)|^2 \stackrel{a||b}{\to} g_s^2 \mathcal{C} \frac{P(z)}{2(p_a \cdot p_b)} |\mathcal{M}_F(\ldots,a+b,\ldots)|^2$$

Gluon j → "soft":

Coherence → Parton j really emitted by (i,k) "colour antenna"

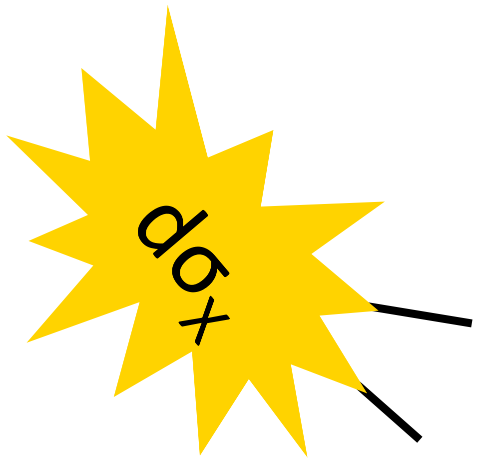
$$|\mathcal{M}_{F+1}(\ldots,i,j,k\ldots)|^2 \stackrel{j_g\to 0}{\to} g_s^2 \mathcal{C} \frac{(p_i \cdot p_k)}{(p_i \cdot p_j)(p_j \cdot p_k)} |\mathcal{M}_F(\ldots,i,k,\ldots)|^2$$

+ scaling violation: $g_s^2 \to 4\pi\alpha_s(Q^2)$

Can apply this many times
→ nested factorizations

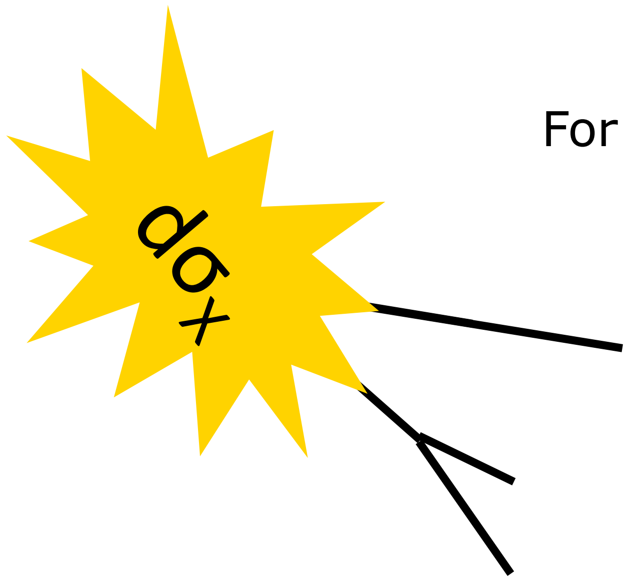
See: PS, *Introduction to QCD*, TASI 2012, arXiv:1207.2389

Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

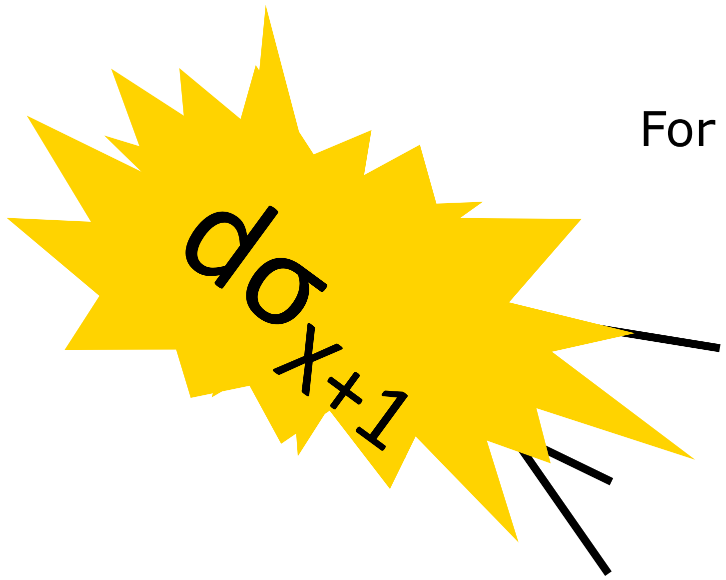
Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X$$

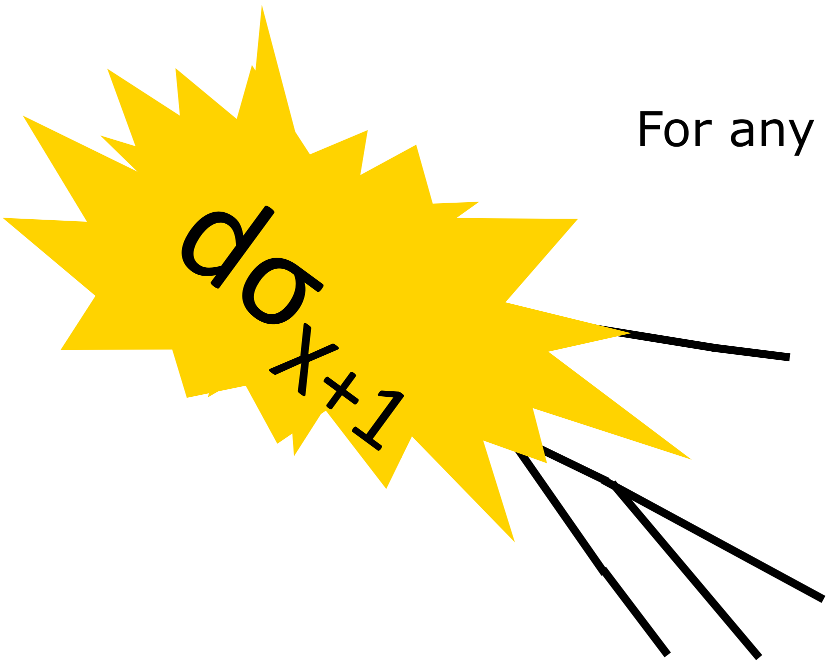
Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

Bremsstrahlung

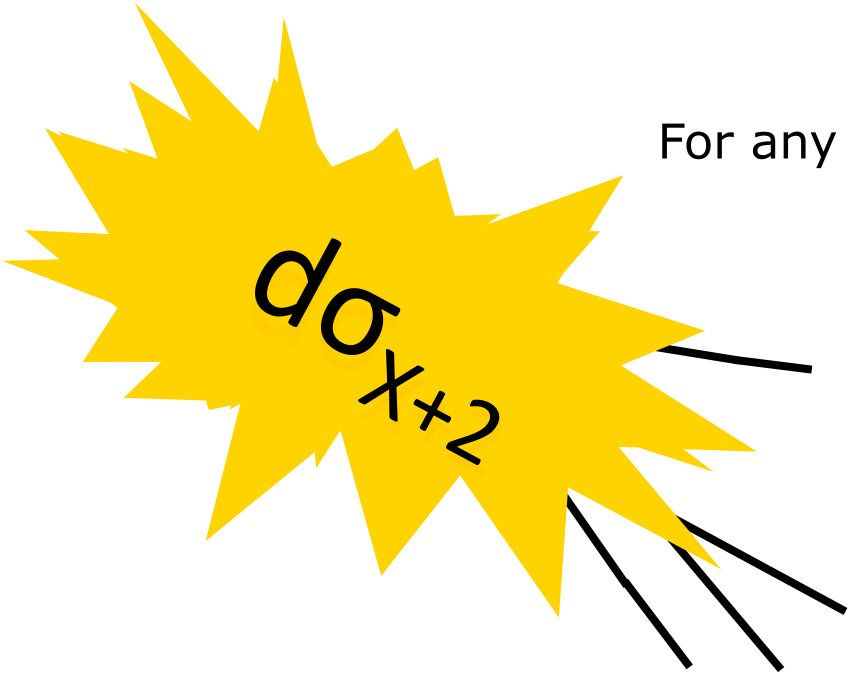


For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \qquad ✓$$

$$d\sigma_{X+2} \sim N_C 2 g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1}$$

Bremsstrahlung

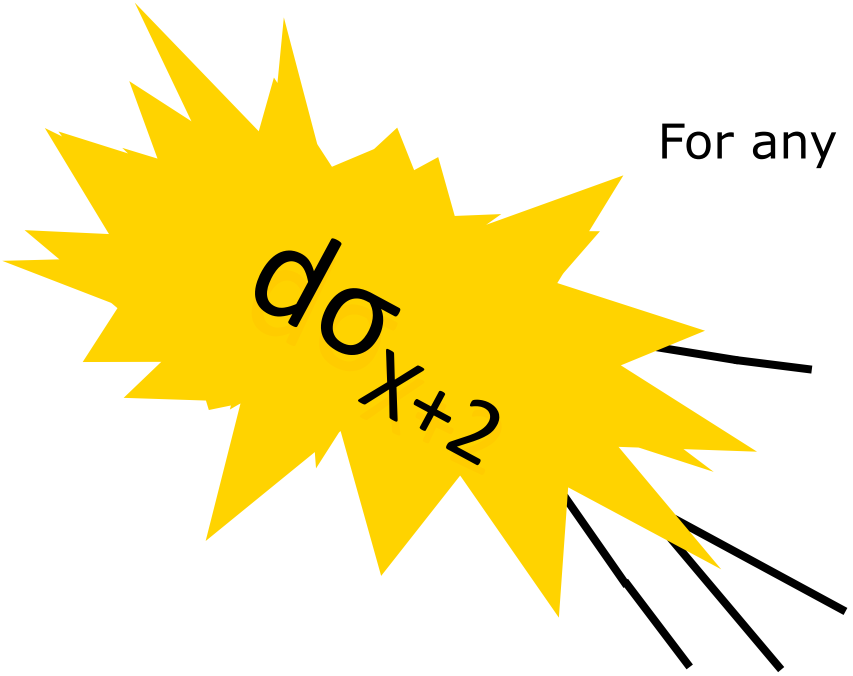


For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2 g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

Bremsstrahlung



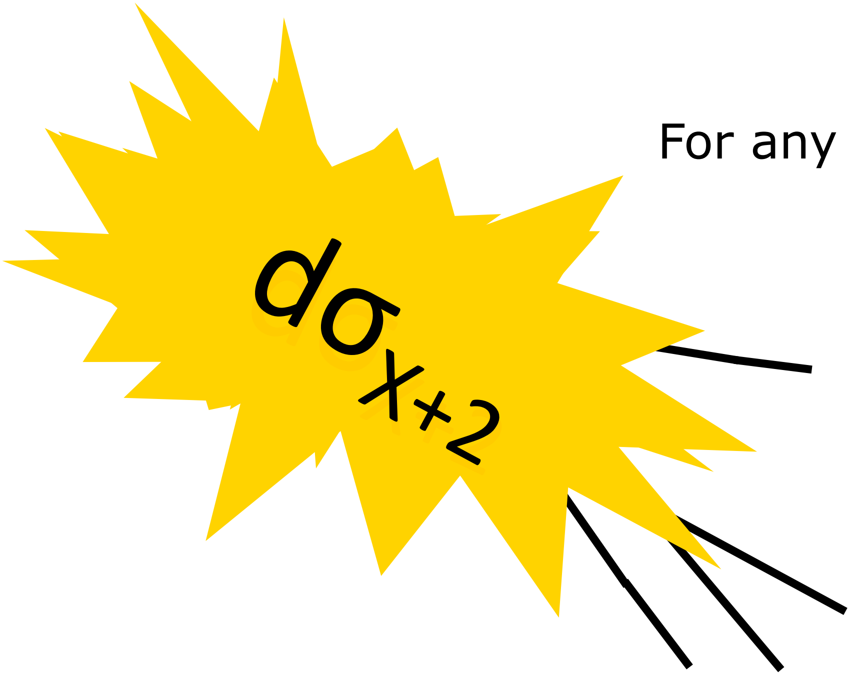
For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2 g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

$$d\sigma_{X+3} \sim N_C 2 g_s^2 \frac{ds_{i3}}{s_{i3}} \frac{ds_{3j}}{s_{3j}} d\sigma_{X+2} \quad \ldots$$

Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2 g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

$$d\sigma_{X+3} \sim N_C 2 g_s^2 \frac{ds_{i3}}{s_{i3}} \frac{ds_{3j}}{s_{3j}} d\sigma_{X+2} \quad ...$$

Factorization in Soft and Collinear Limits

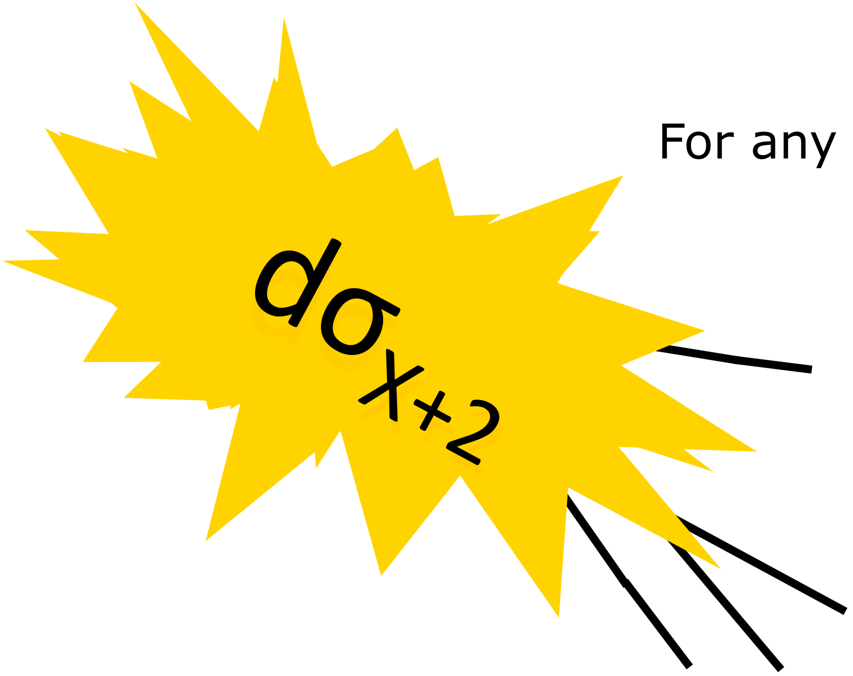
P(z) : "DGLAP Splitting Functions"

$$|M(\ldots,p_i,p_j\ldots)|^2 \stackrel{i||j}{\to} g_s^2 \mathcal{C} \frac{P(z)}{s_{ij}} |M(\ldots,p_i+p_j,\ldots)|^2$$

$$|M(\ldots,p_i,p_j,p_k\ldots)|^2 \stackrel{j_g\to 0}{\to} g_s^2 \mathcal{C} \frac{2s_{ik}}{s_{ij}s_{jk}} |M(\ldots,p_i,p_k,\ldots)|^2$$

"Soft Eikonal" : generalizes to Dipole/Antenna Functions (more later)

Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

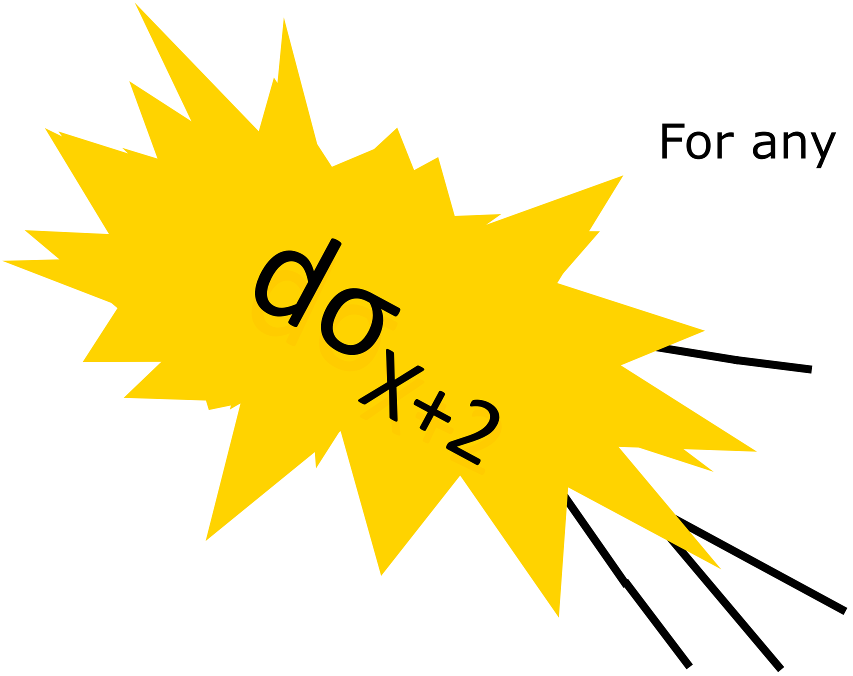
$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2 g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

$$d\sigma_{X+3} \sim N_C 2 g_s^2 \frac{ds_{i3}}{s_{i3}} \frac{ds_{3j}}{s_{3j}} d\sigma_{X+2} \quad \ldots$$

Singularities: mandated by gauge theory
Non-singular terms: process-dependent

Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

$$d\sigma_{X+3} \sim N_C 2g_s^2 \frac{ds_{i3}}{s_{i3}} \frac{ds_{3j}}{s_{3j}} d\sigma_{X+2} \quad \ldots$$

Singularities: mandated by gauge theory
Non-singular terms: process-dependent

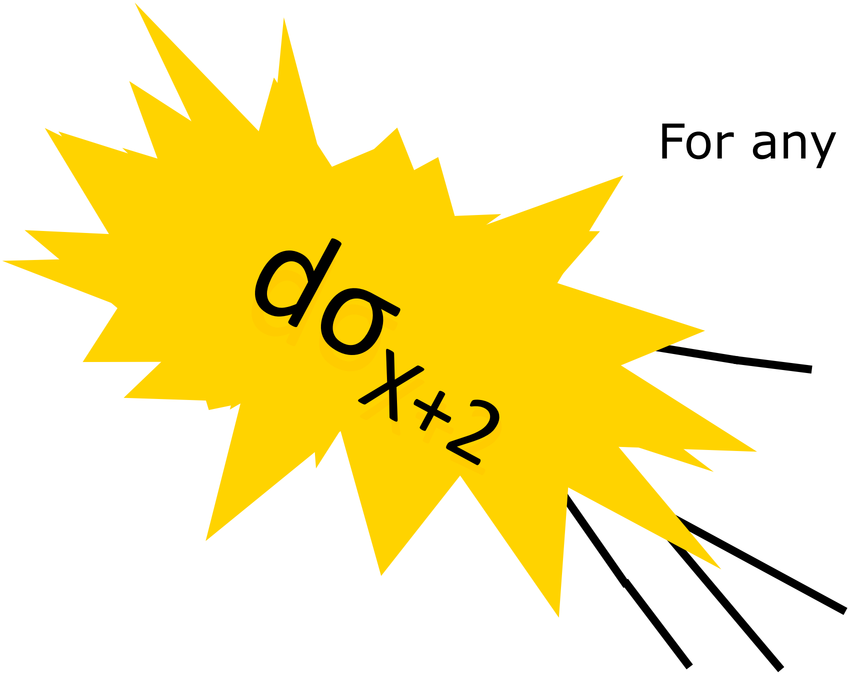
SOFT COLLINEAR

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

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

SOFT COLLINEAR+F

Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

$$d\sigma_{X+3} \sim N_C 2g_s^2 \frac{ds_{i3}}{s_{i3}} \frac{ds_{3j}}{s_{3j}} d\sigma_{X+2} \quad \ldots$$

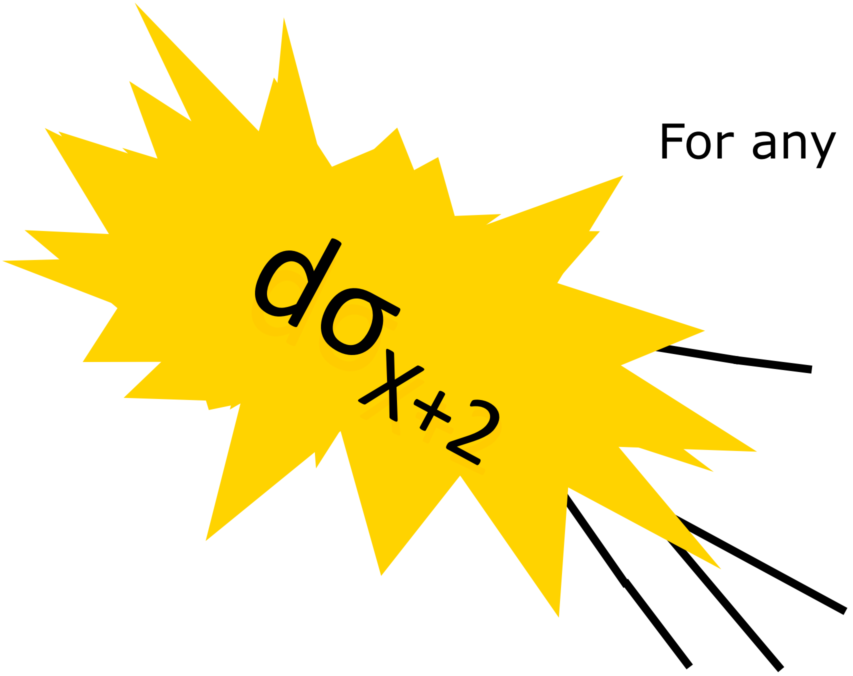
Iterated factorization

Gives us a universal approximation to ∞-order tree-level cross sections.

Exact in singular (strongly ordered) limit.

Finite terms (non-universal) → Uncertainties for non-singular (hard) radiation

Bremsstrahlung



For any basic process $d\sigma_X$ = ✓ (calculated process by process)

$$d\sigma_{X+1} \sim N_C 2 g_s^2 \frac{ds_{i1}}{s_{i1}} \frac{ds_{1j}}{s_{1j}} d\sigma_X \quad ✓$$

$$d\sigma_{X+2} \sim N_C 2 g_s^2 \frac{ds_{i2}}{s_{i2}} \frac{ds_{2j}}{s_{2j}} d\sigma_{X+1} \quad ✓$$

$$d\sigma_{X+3} \sim N_C 2 g_s^2 \frac{ds_{i3}}{s_{i3}} \frac{ds_{3j}}{s_{3j}} d\sigma_{X+2} \quad \ldots$$

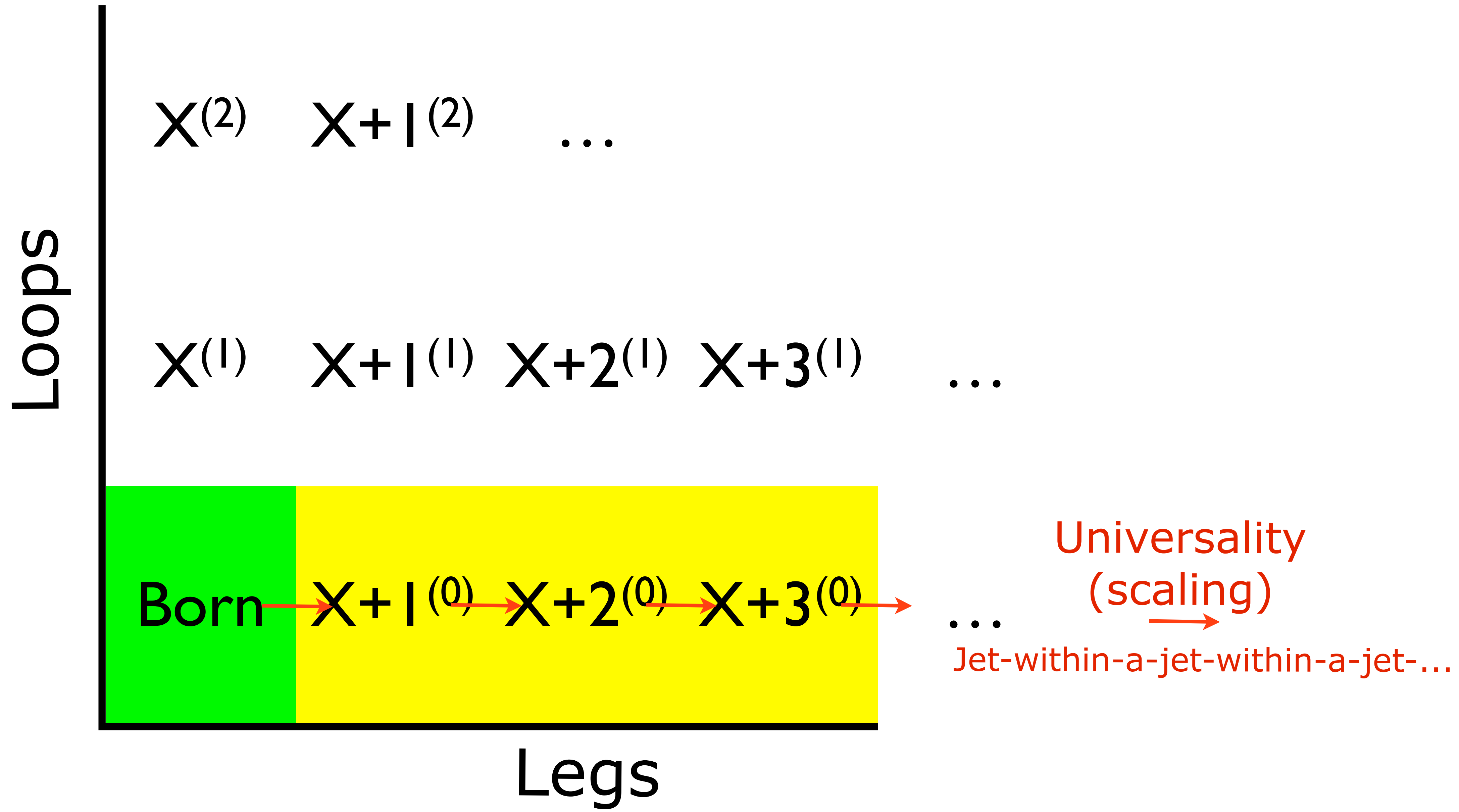
Iterated factorization

Gives us a universal approximation to ∞-order tree-level cross sections.
Exact in singular (strongly ordered) limit.
Finite terms (non-universal) → Uncertainties for non-singular (hard) radiation

But something is not right … Total σ would be infinite …

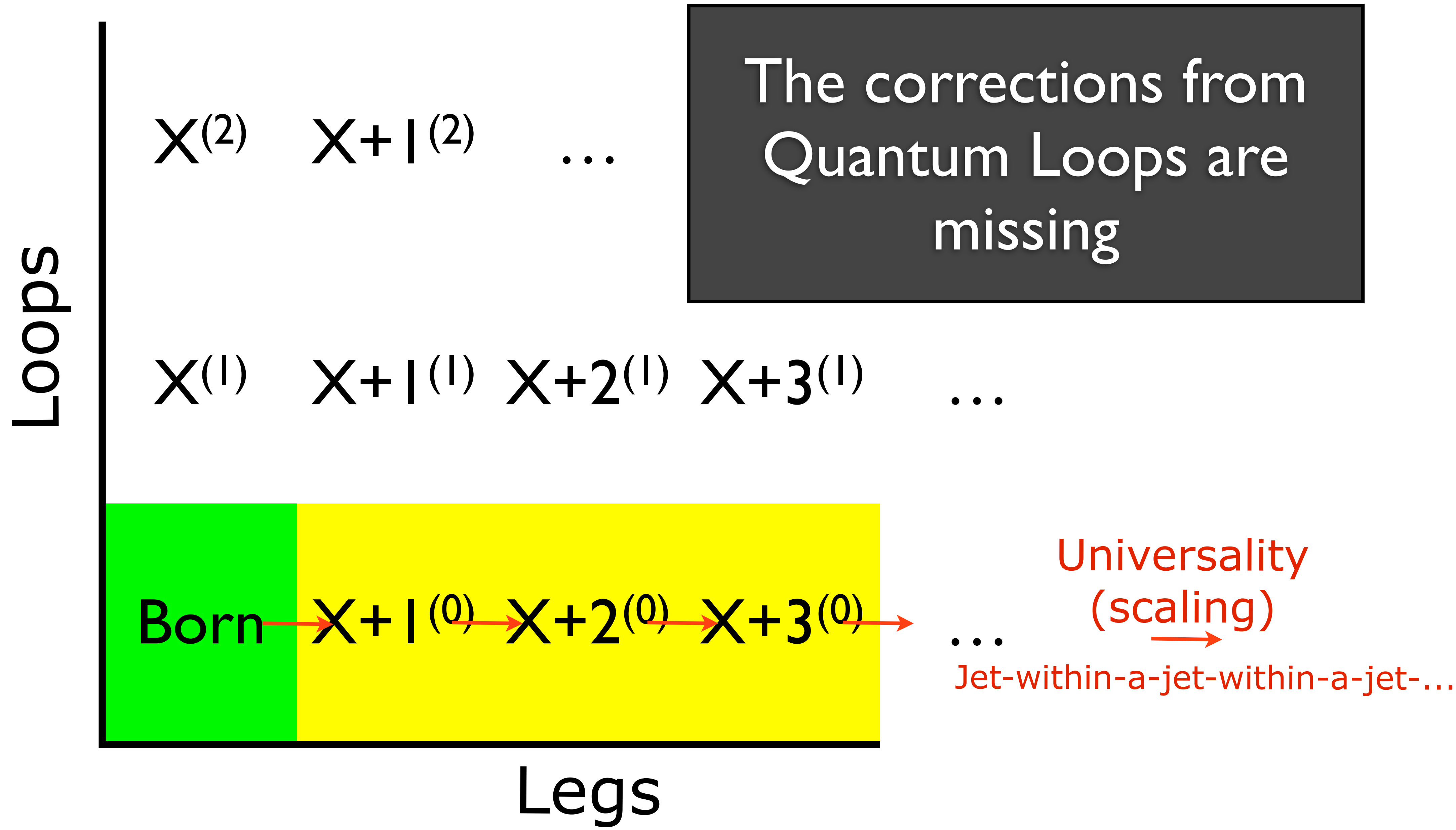
Loops and Legs

Coefficients of the Perturbative Series



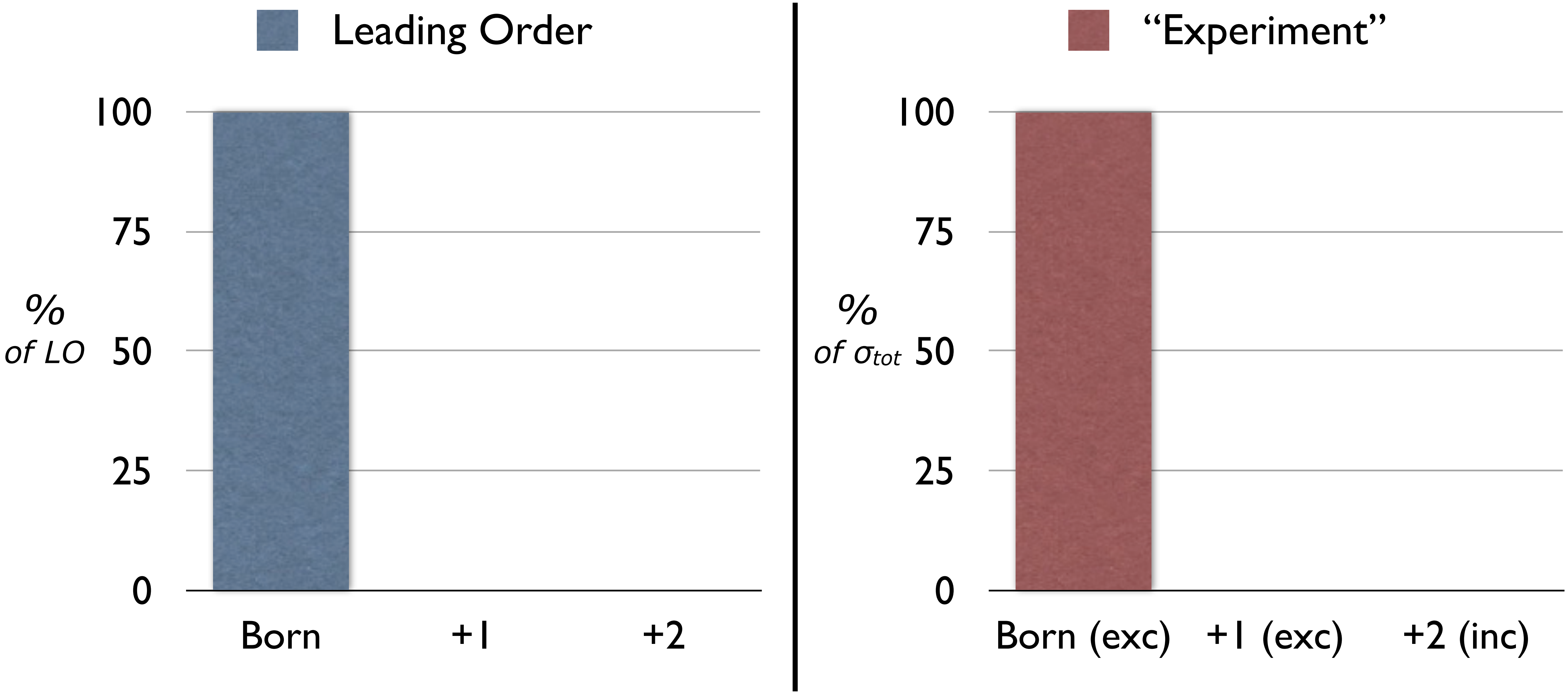
Loops and Legs

Coefficients of the Perturbative Series



Evolution

$$Q \sim Q_X$$

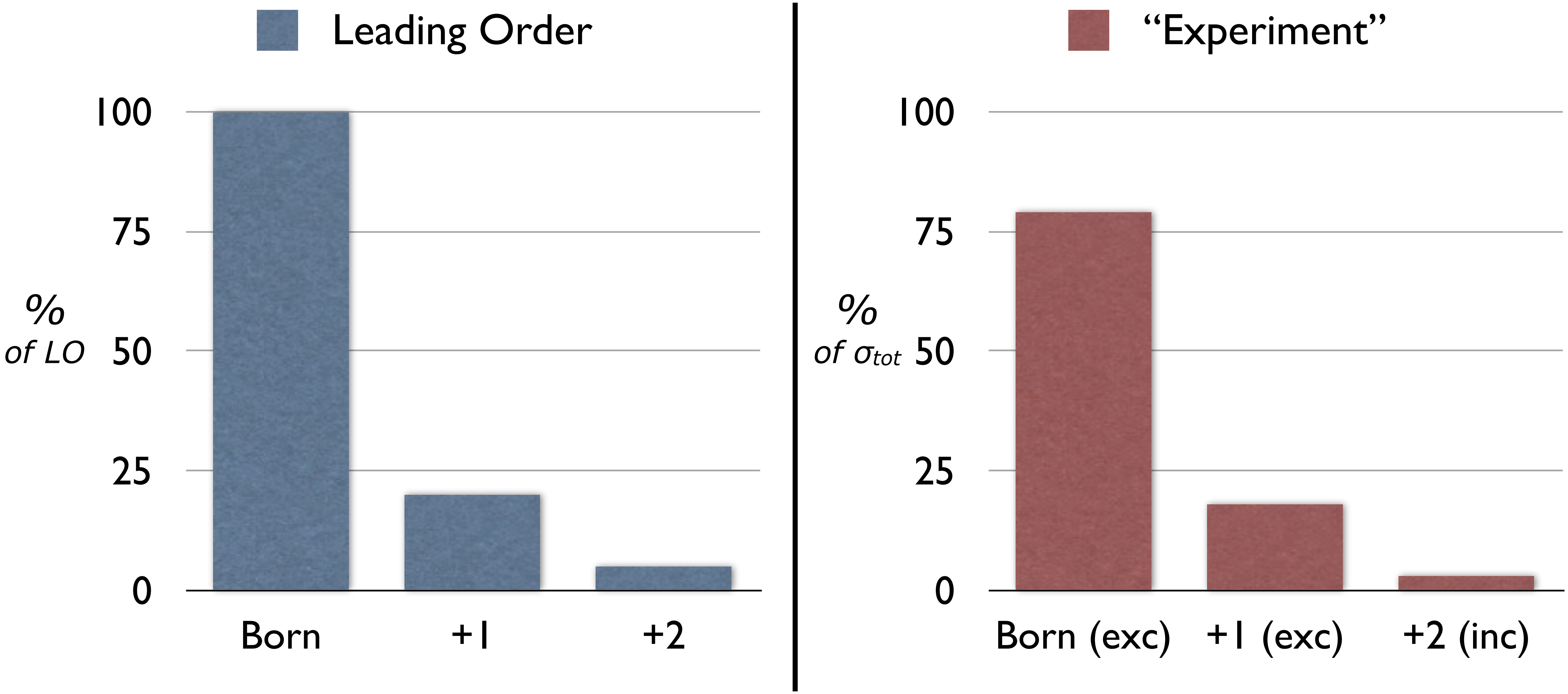


Exclusive = n and only n jets

Inclusive = n or more jets

Evolution

$$Q \sim \frac{Q_X}{\text{“A few”}}$$

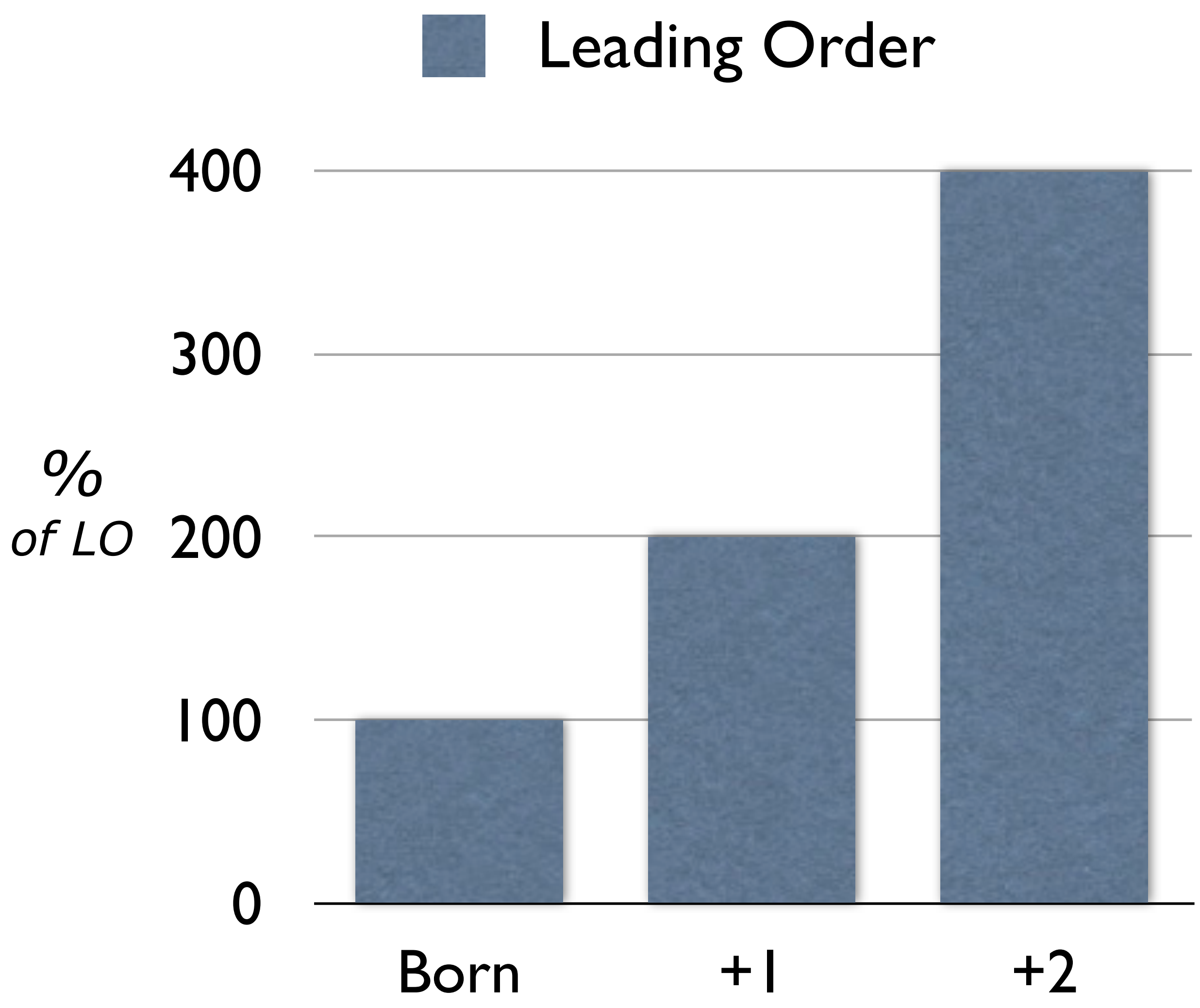


Exclusive = n and only n jets

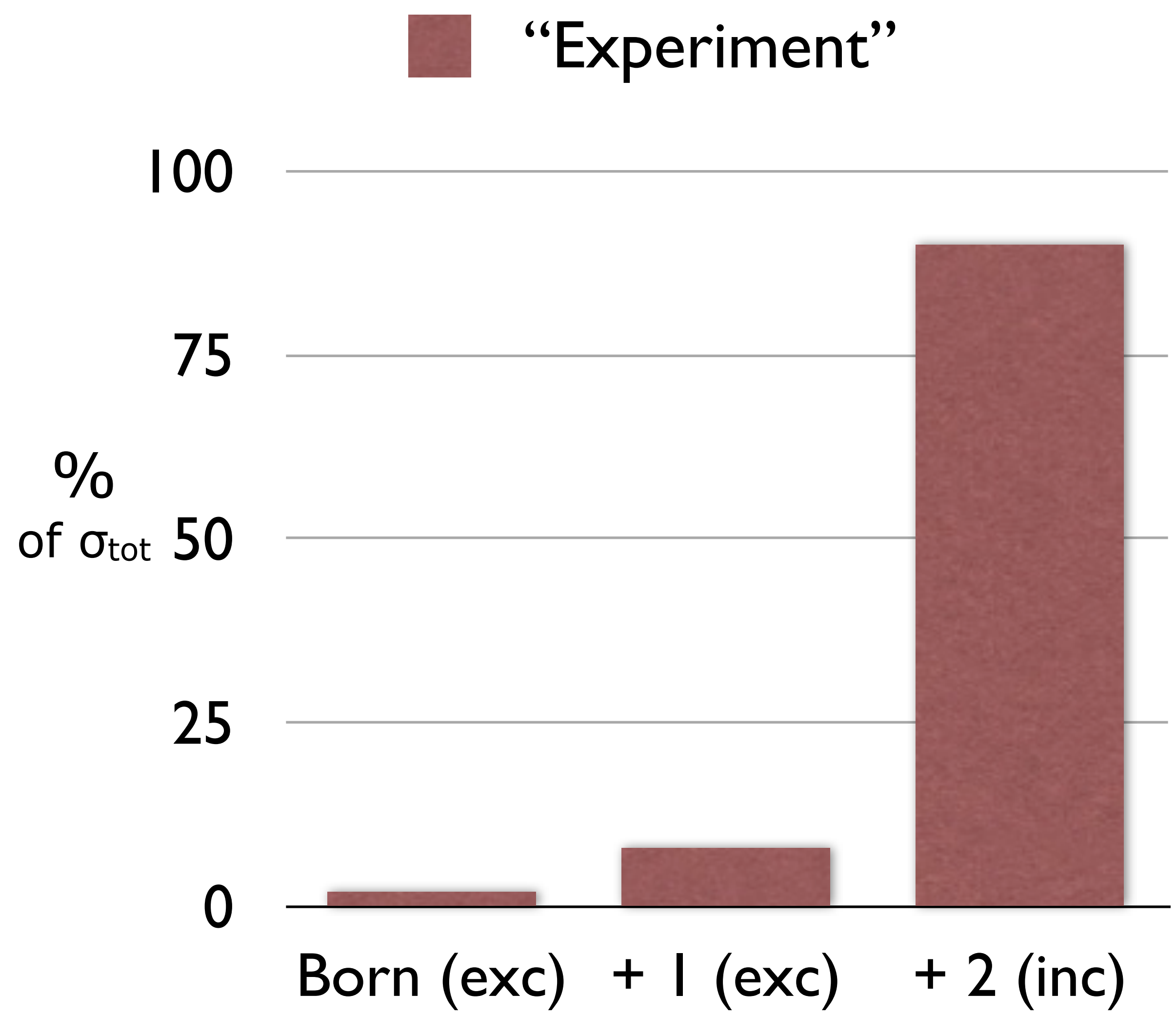
Inclusive = n or more jets

Evolution

$$Q \ll Q_X$$



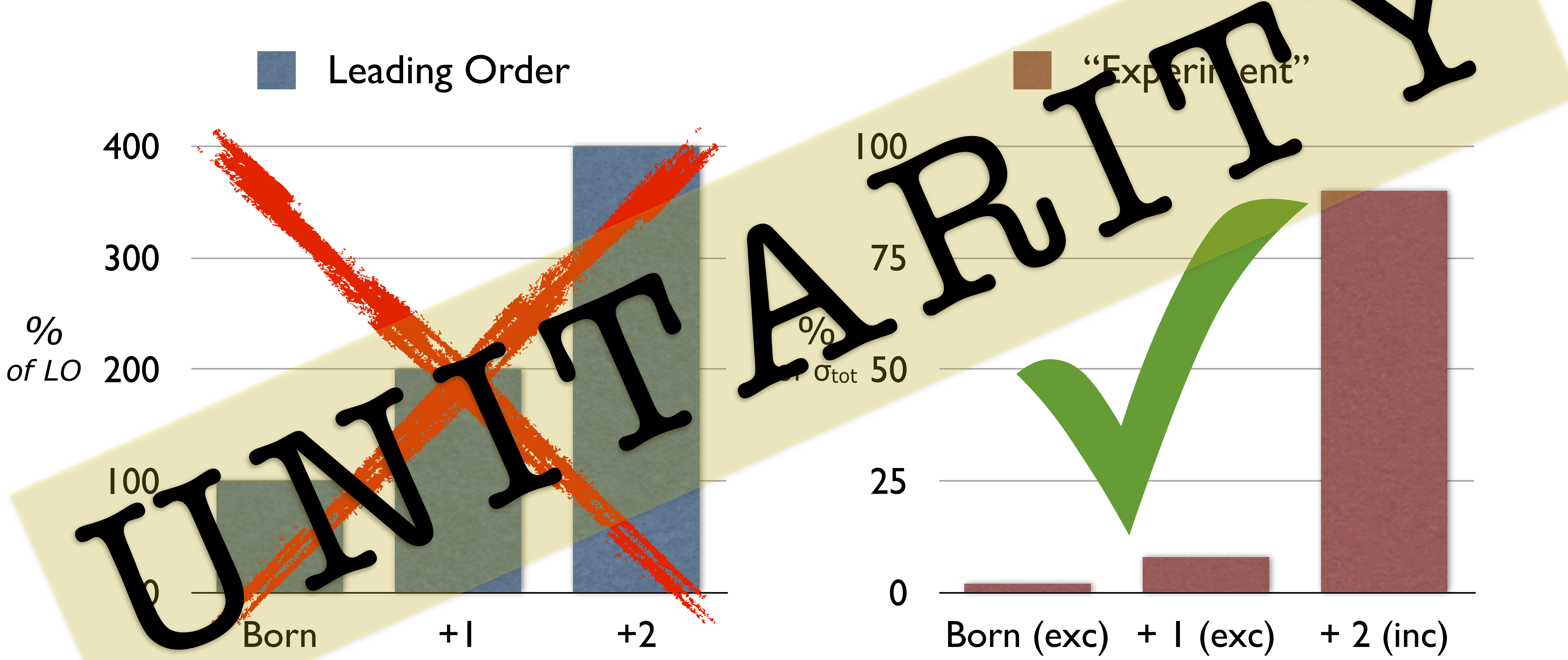
Cross Section Diverges



Cross Section Remains = Total (IR safe)
Number of Partons Diverges (IR unsafe)

Evolution

$$Q \ll Q_X$$



Cross Section Diverges

Cross Section Remains = Total (IR safe)
Number of Partons Diverges (IR unsafe)

Unitarity → Evolution

Unitarity

Kinoshita-Lee-Nauenberg:
(sum over degenerate quantum states = finite)

$$\text{Loop} = -\ \text{Int}(\text{Tree}) + \text{F}$$

Parton Showers neglect F

→ *Leading-Logarithmic (LL) Approximation*

Imposed by Event *evolution*:

When (X) branches to (X+1):
Gain one (X+1). Loose one (X).

→ *evolution equation with kernel* $\frac{d\sigma_{X+1}}{d\sigma_X}$

Evolve in some measure of *resolution*
~ hardness, 1/time … ~ fractal scale

→ includes both real (tree) and virtual (loop) corrections

- Interpretation: the structure evolves! (example: X = 2-jets)
  - Take a jet algorithm, with resolution measure “Q”, apply it to your events
  - At a very crude resolution, you find that everything is 2-jets

Unitarity → Evolution

Unitarity

Kinoshita-Lee-Nauenberg:
(sum over degenerate quantum states = finite)

$$\text{Loop} = -\,\text{Int}(\text{Tree}) + \text{F}$$

Parton Showers neglect F

→ *Leading-Logarithmic (LL) Approximation*

Imposed by Event *evolution*:

When (X) branches to (X+1):
Gain one (X+1). Loose one (X).

→ *evolution equation with kernel* $\frac{d\sigma_{X+1}}{d\sigma_X}$

Evolve in some measure of *resolution*
~ hardness, 1/time … ~ fractal scale

→ includes both real (tree) and virtual (loop) corrections

- Interpretation: the structure evolves! (example: X = 2-jets)
  - Take a jet algorithm, with resolution measure “Q”, apply it to your events
  - At a very crude resolution, you find that everything is 2-jets
  - At finer resolutions → some 2-jets migrate → 3-jets = $\sigma_{X+1}(Q) = \sigma_{X;\text{incl}} - \sigma_{X;\text{excl}}(Q)$

Unitarity → Evolution

Unitarity

Kinoshita-Lee-Nauenberg:
(sum over degenerate quantum states = finite)

$$\text{Loop} = -\,\text{Int}(\text{Tree}) + \text{F}$$

Parton Showers neglect F

→ *Leading-Logarithmic (LL) Approximation*

Imposed by Event *evolution*:

When (X) branches to (X+1):
Gain one (X+1). Loose one (X).

→ *evolution equation with kernel* $\frac{d\sigma_{X+1}}{d\sigma_X}$

Evolve in some measure of *resolution*
~ hardness, 1/time … ~ fractal scale

→ includes both real (tree) and virtual (loop) corrections

- Interpretation: the structure evolves! (example: X = 2-jets)
  - Take a jet algorithm, with resolution measure "Q", apply it to your events
  - At a very crude resolution, you find that everything is 2-jets
  - At finer resolutions → some 2-jets migrate → 3-jets = $\sigma_{X+1}(Q) = \sigma_{X;incl} - \sigma_{X;excl}(Q)$
  - Later, some 3-jets migrate further, etc → $\sigma_{X+n}(Q) = \sigma_{X;incl} - \sum\sigma_{X+m<n;excl}(Q)$

Unitarity → Evolution

Unitarity

Kinoshita-Lee-Nauenberg:
(sum over degenerate quantum states = finite)

$$\text{Loop} = -\,\text{Int(Tree)} + \text{F}$$

Parton Showers neglect F

→ *Leading-Logarithmic (LL) Approximation*

Imposed by Event *evolution*:

When (X) branches to (X+1):
Gain one (X+1). Loose one (X).

→ *evolution equation with kernel* $\frac{d\sigma_{X+1}}{d\sigma_X}$

Evolve in some measure of *resolution*
~ hardness, 1/time … ~ fractal scale

→ includes both real (tree) and virtual (loop) corrections

- Interpretation: the structure evolves! (example: X = 2-jets)
  - Take a jet algorithm, with resolution measure “Q”, apply it to your events
  - At a very crude resolution, you find that everything is 2-jets
  - At finer resolutions → some 2-jets migrate → 3-jets = $\sigma_{X+1}(Q) = \sigma_{X;incl} - \sigma_{X;excl}(Q)$
  - Later, some 3-jets migrate further, etc → $\sigma_{X+n}(Q) = \sigma_{X;incl} - \sum\sigma_{X+m<n;excl}(Q)$
  - This evolution takes place between two scales, $Q_{in} \sim \mathbf{s}$ and $Q_{end} \sim \mathbf{Q_{had}}$

Unitarity → Evolution

Unitarity

Kinoshita-Lee-Nauenberg:
(sum over degenerate quantum states = finite)

$$\text{Loop} = -\,\text{Int(Tree)} + \text{F}$$

Parton Showers neglect F

→ *Leading-Logarithmic (LL) Approximation*

Imposed by Event *evolution*:

When (X) branches to (X+1):
Gain one (X+1). Loose one (X).

→ *evolution equation with kernel* $\frac{d\sigma_{X+1}}{d\sigma_X}$

Evolve in some measure of *resolution*
~ hardness, 1/time … ~ fractal scale

→ includes both real (tree) and virtual (loop) corrections

- Interpretation: the structure evolves! (example: X = 2-jets)
  - Take a jet algorithm, with resolution measure "Q", apply it to your events
  - At a very crude resolution, you find that everything is 2-jets
  - At finer resolutions → some 2-jets migrate → 3-jets = $\sigma_{X+1}(Q) = \sigma_{X;incl} - \sigma_{X;excl}(Q)$
  - Later, some 3-jets migrate further, etc → $\sigma_{X+n}(Q) = \sigma_{X;incl} - \sum\sigma_{X+m<n;excl}(Q)$
  - This evolution takes place between two scales, Q_{in} ~ **s** and Q_{end} ~ $\mathbf{Q_{had}}$
- $\sigma_{X;tot} = \text{Sum}\,(\sigma_{X+0,1,2,3,\ldots;excl}) = \text{int}(d\sigma_X)$

Evolution Equations

Evolution Equations

What we need is a differential equation

- Boundary condition: a few partons defined at a high scale (Q_F)
- Then evolves (or “runs”) that parton system down to a low scale (the hadronization cutoff ~ 1 GeV) → It’s an evolution equation in Q_F

Evolution Equations

What we need is a differential equation

Boundary condition: a few partons defined at a high scale (Q_F)

Then evolves (or "runs") that parton system down to a low scale (the hadronization cutoff ~ 1 GeV) → It's an evolution equation in Q_F

Close analogue: nuclear decay

Evolve an unstable nucleus. Check if it decays + follow chains of decays.

Decay constant

$$\frac{\mathrm{d}P(t)}{\mathrm{d}t} = c_N$$

Probability to remain undecayed in the time interval $[t_1,t_2]$

$$\Delta(t_1,t_2) = \exp\left(-\int_{t_1}^{t_2} c_N \, \mathrm{d}t\right) = \exp\left(-c_N \, \Delta t\right)$$

$$= 1 - c_N \Delta t + \mathcal{O}(c_N^2)$$

Decay probability per unit time

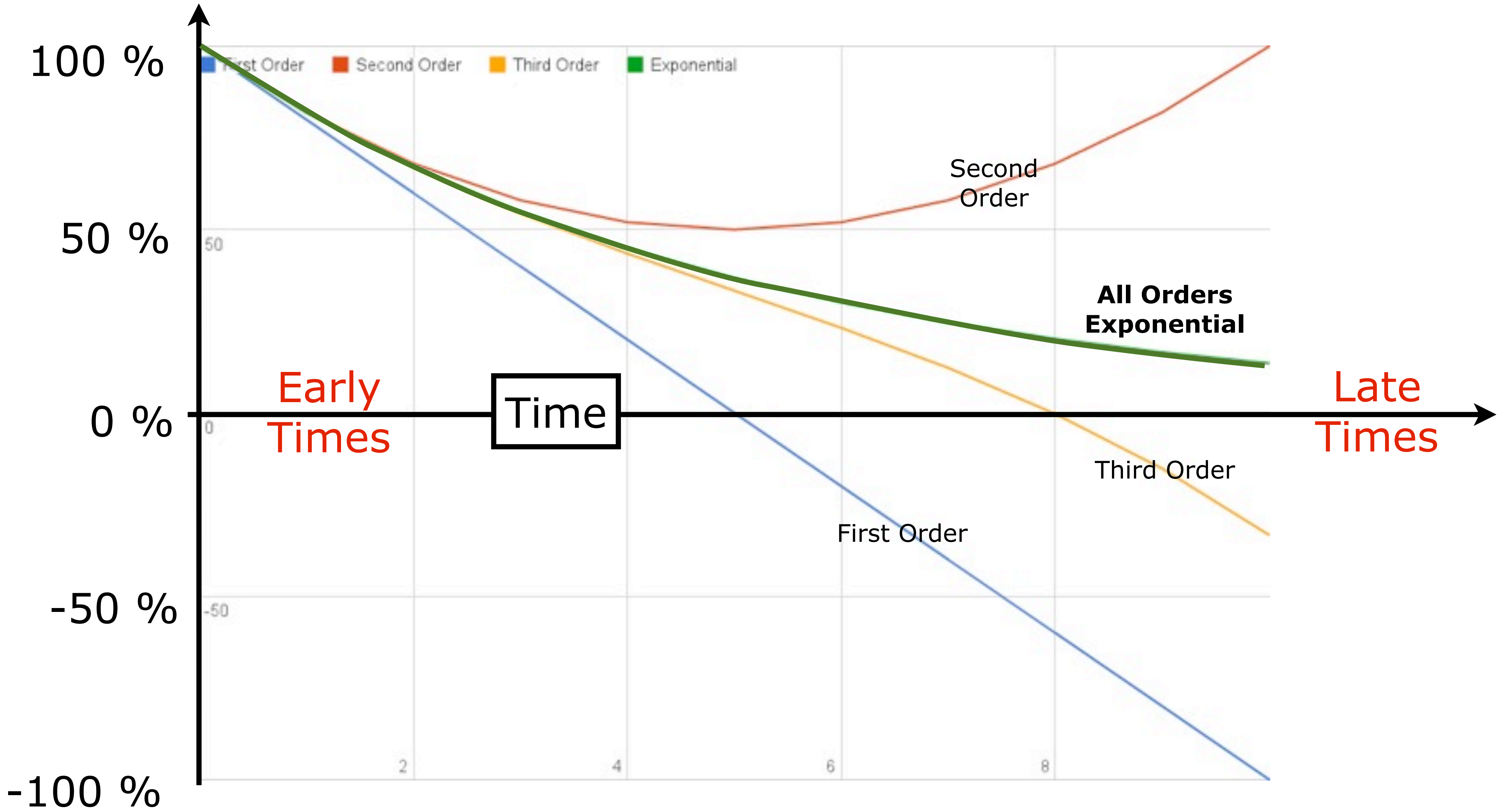
$$\frac{\mathrm{d}P_{\mathrm{res}}(t)}{\mathrm{d}t} = \frac{-\mathrm{d}\Delta}{\mathrm{d}t} = c_N \, \Delta(t_1,t)$$

(requires that the nucleus did not already decay)

$\Delta(t_1,t_2)$: "Sudakov Factor"

Nuclear Decay

Nuclei remaining undecayed after time t $= \Delta(t_1, t_2) = \exp\left(-\int_{t_1}^{t_2} \mathrm{d}t\, \frac{\mathrm{d}\mathcal{P}}{\mathrm{d}t}\right)$



The Sudakov Factor

In nuclear decay, the Sudakov factor counts:

How many nuclei remain undecayed after a time t

Probability to remain undecayed in the time interval $[t_1,t_2]$

$$\Delta(t_1,t_2) = \exp\left(-\int_{t_1}^{t_2} c_N \, \mathrm{d}t\right) = \exp\left(-c_N \, \Delta t\right)$$

The Sudakov Factor

In nuclear decay, the Sudakov factor counts:

How many nuclei remain undecayed after a time t

Probability to remain undecayed in the time interval $[t_1, t_2]$

$$\Delta(t_1, t_2) = \exp\left(-\int_{t_1}^{t_2} c_N \, \mathrm{d}t\right) = \exp\left(-c_N \, \Delta t\right)$$

The Sudakov factor for a parton system counts:

The probability that the parton system doesn't evolve (branch) when we run the factorization scale (~1/time) from a high to a low scale

Evolution probability per unit "time"

$$\frac{\mathrm{d}P_{\mathrm{res}}(t)}{\mathrm{d}t} = \frac{-\mathrm{d}\Delta}{\mathrm{d}t} = c_N \, \Delta(t_1, t)$$

(replace t by shower evolution scale)

(replace c_N by proper shower evolution kernels)

What’s the evolution kernel?

DGLAP splitting functions

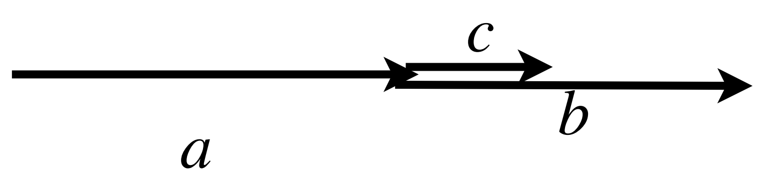
Can be derived from *collinear limit* of MEs $(p_b+p_c)^2 \to 0$

+ evolution equation from invariance with respect to $Q_F \to$ RGE

DGLAP

(E.g., PYTHIA)

$$\mathrm{d}\mathcal{P}_a = \sum_{b,c} \frac{\alpha_{abc}}{2\pi} P_{a\to bc}(z)\,\mathrm{d}t\,\mathrm{d}z\ .$$



$p_b = z\, p_a$

$p_c = (1\text{-}z)\, p_a$

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

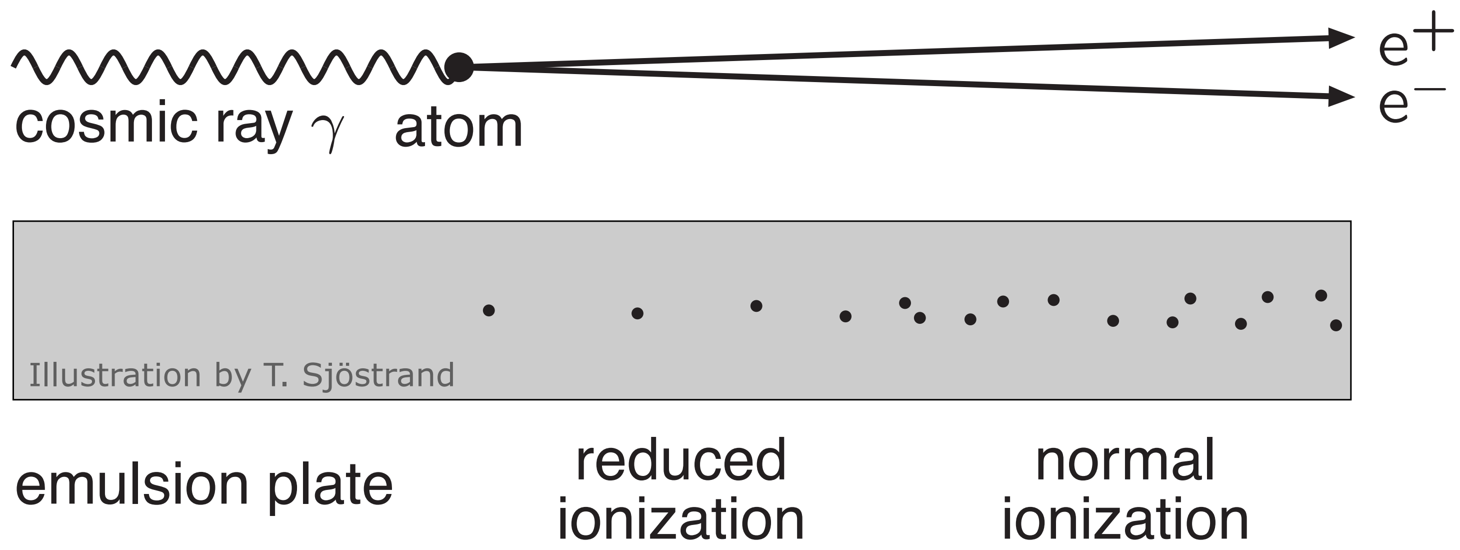
$$\mathrm{d}t = \frac{\mathrm{d}Q^2}{Q^2} = \mathrm{d}\ln Q^2$$

… with Q^2 some measure of “hardness”
= event/jet resolution
measuring parton virtualities / formation time / …

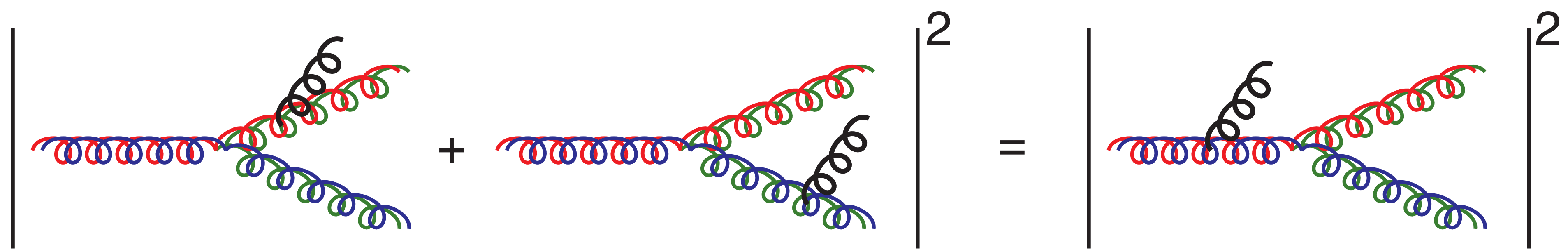
Note: there exist now also alternatives to AP kernels (with same collinear limits!): dipoles, antennae, …

Coherence

QED: Chudakov effect (mid-fifties)



QCD: colour coherence for **soft** gluon emission

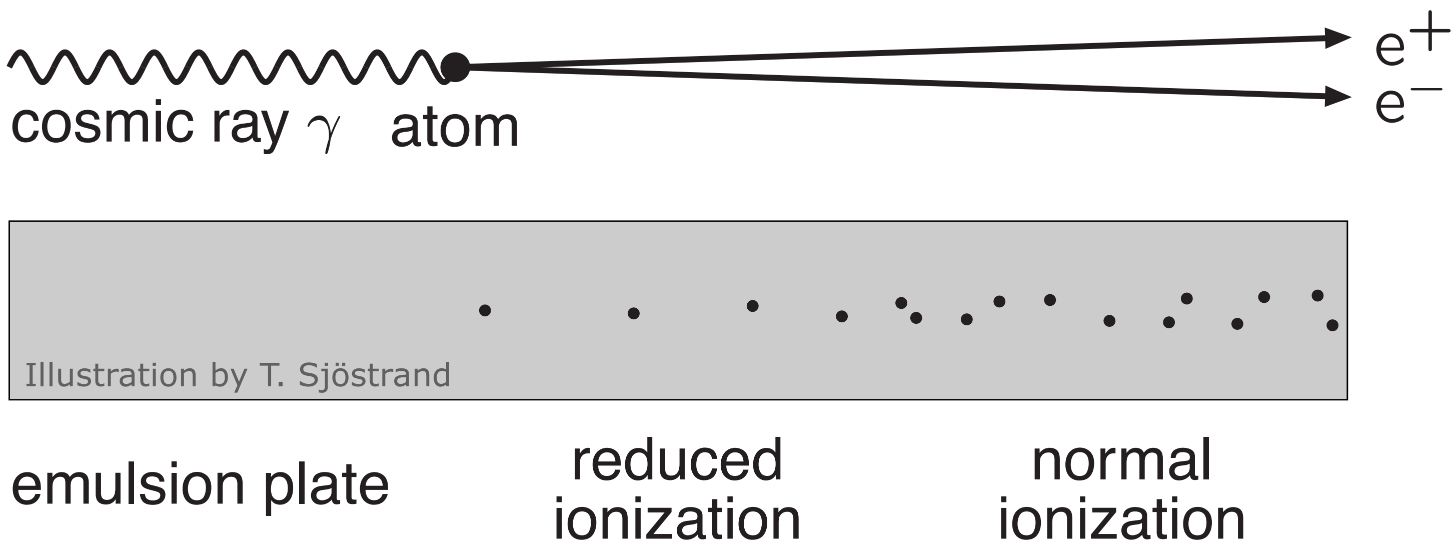


→ an example of an interference effect that can be treated probabilistically

More interference effects can be included by matching to full matrix elements

Coherence

QED: Chudakov effect (mid-fifties)



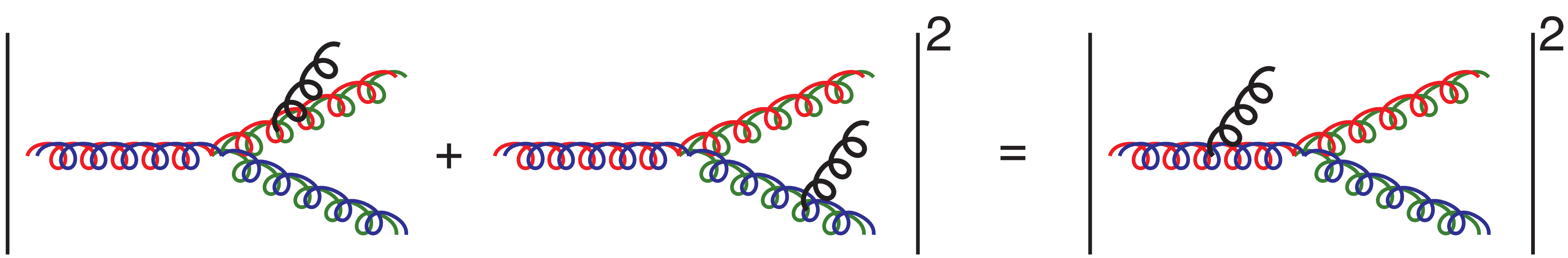
Approximations to Coherence:

Angular Ordering (HERWIG)

Angular Vetos (PYTHIA)

Coherent Dipoles/Antennae (ARIADNE, Catani-Seymour, VINCIA)

QCD: colour coherence for **soft** gluon emission



→ an example of an interference effect that can be treated probabilistically

More interference effects can be included by matching to full matrix elements

Coherence at Work

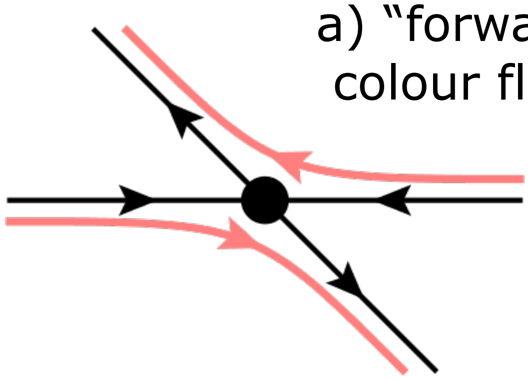
Example taken from: Ritzmann, Kosower, PS, PLB718 (2013) 1345

Example: quark-quark scattering in hadron collisions

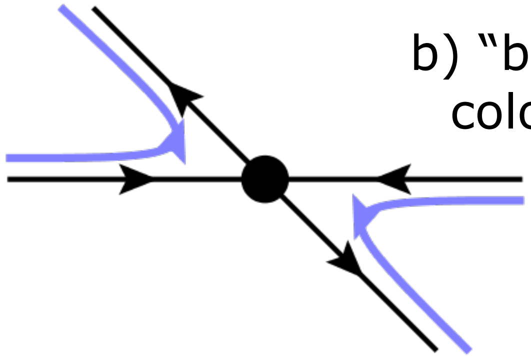
Consider one specific phase-space point (eg scattering at 45°)

2 possible colour flows: a and b

a) "forward" colour flow



b) "backward" colour flow



Another good recent example is the SM contribution to the Tevatron top-quark forward-backward asymmetry from coherent showers, see: PS, Webber, Winter, JHEP 1207 (2012) 151

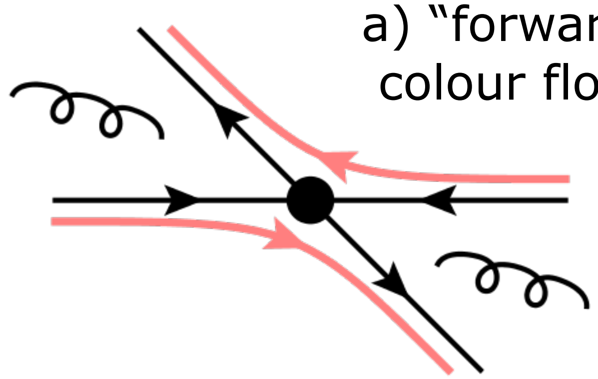
Coherence at Work

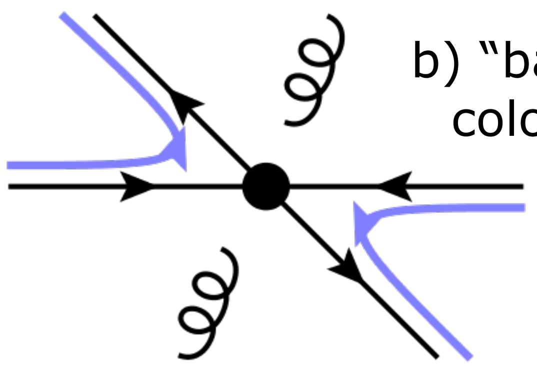
Example taken from: Ritzmann, Kosower, PS, PLB718 (2013) 1345

Example: quark-quark scattering in hadron collisions

Consider one specific phase-space point (eg scattering at 45°)

2 possible colour flows: a and b





Another good recent example is the SM contribution to the Tevatron top-quark forward-backward asymmetry from coherent showers, see: PS, Webber, Winter, JHEP 1207 (2012) 151

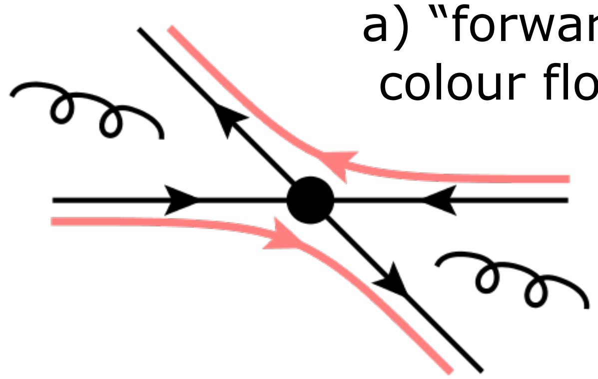
Coherence at Work

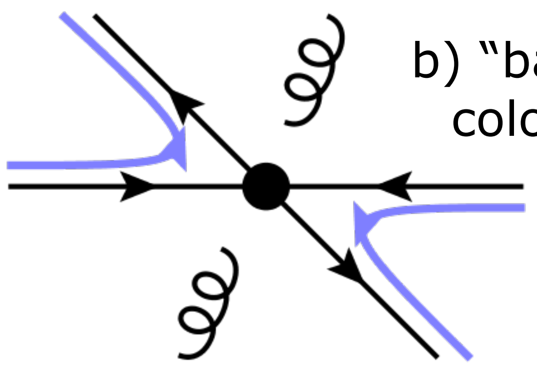
Example taken from: Ritzmann, Kosower, PS, PLB718 (2013) 1345

Example: quark-quark scattering in hadron collisions

Consider one specific phase-space point (eg scattering at 45°)

2 possible colour flows: a and b





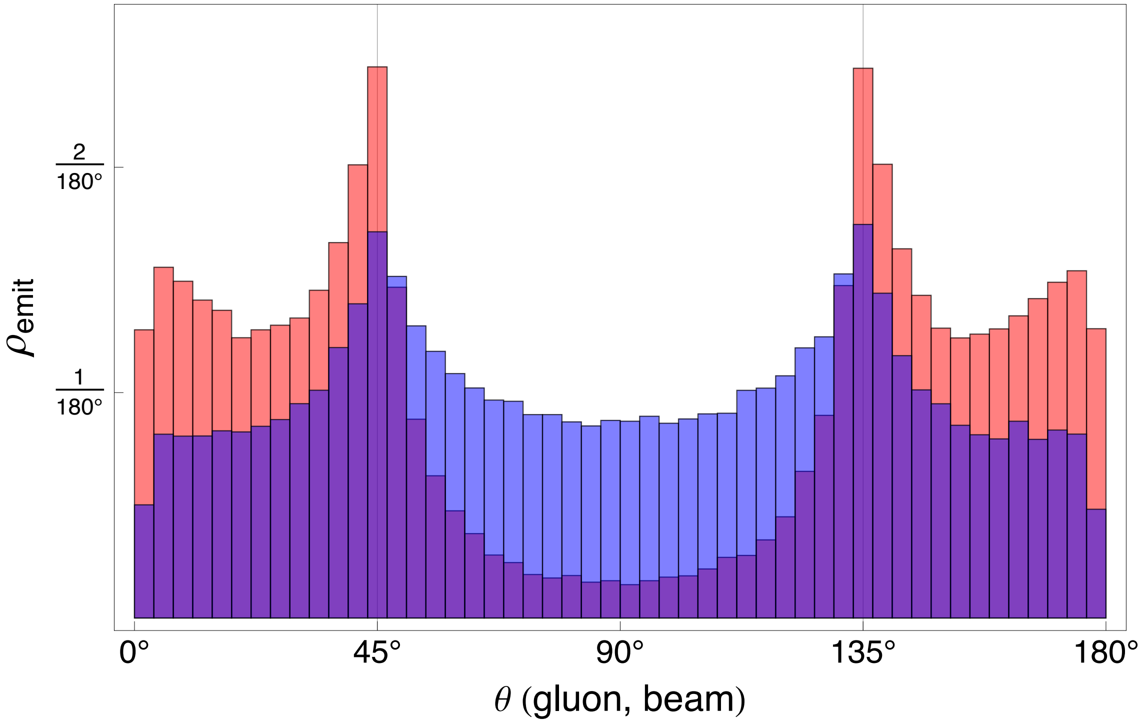
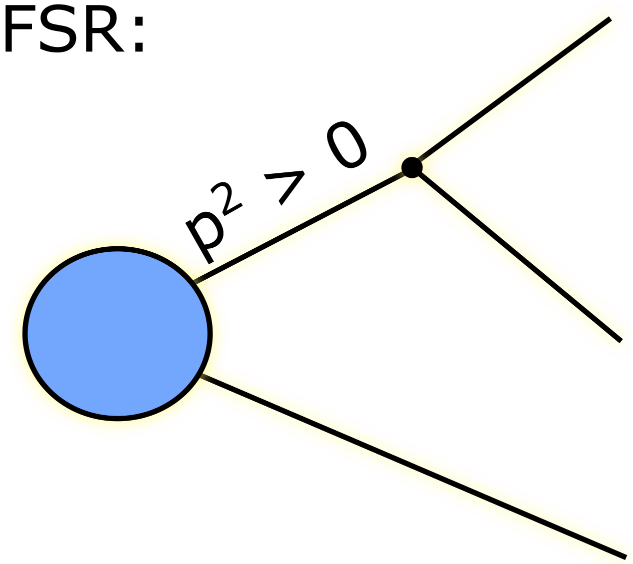


Figure 4: Angular distribution of the first gluon emission in $qq \to qq$ scattering at 45°, for the two different color flows. The light (red) histogram shows the emission density for the forward flow, and the dark (blue) histogram shows the emission density for the backward flow.

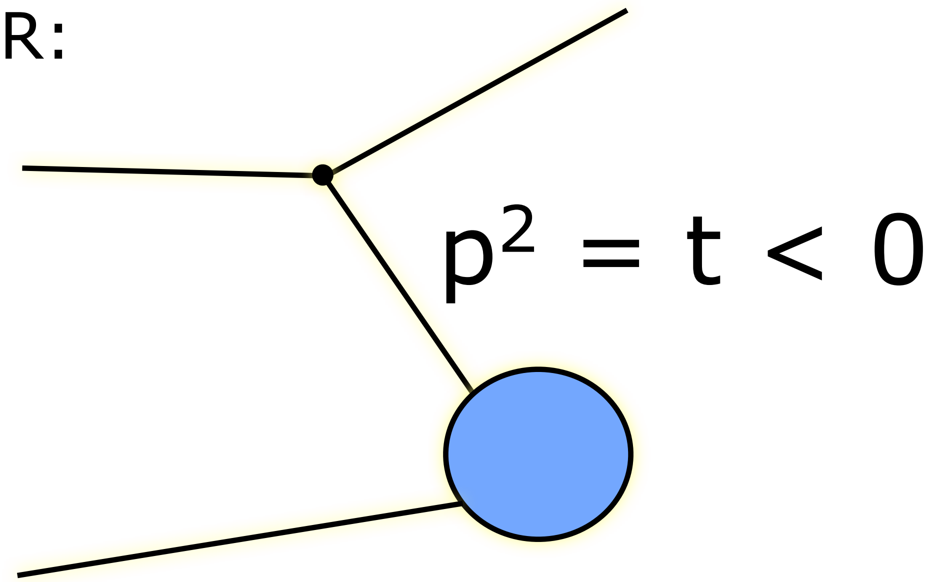
Another good recent example is the SM contribution to the Tevatron top-quark forward-backward asymmetry from coherent showers, see: PS, Webber, Winter, JHEP 1207 (2012) 151

Initial-State vs Final-State Evolution

FSR:



ISR:



Virtualities are Timelike: $p^2>0$

Start at $Q^2 = Q_F^2$
"Forwards evolution"

Virtualities are Spacelike: $p^2<0$

Start at $Q^2 = Q_F^2$
Constrained backwards evolution towards boundary condition = proton

Separation meaningful for collinear radiation, but not for soft …

(Initial-State Evolution)

DGLAP for Parton Density

$$\frac{\mathrm{d}f_b(x,t)}{\mathrm{d}t} = \sum_{a,c} \int \frac{\mathrm{d}x'}{x'} f_a(x',t) \frac{\alpha_{abc}}{2\pi} P_{a\to bc}\left(\frac{x}{x'}\right)$$

→ Sudakov for ISR

$$\begin{aligned}
\Delta(x, t_{\max}, t) &= \exp\left\{ - \int_t^{t_{\max}} \mathrm{d}t' \sum_{a,c} \int \frac{\mathrm{d}x'}{x'} \frac{f_a(x',t')}{f_b(x,t')} \frac{\alpha_{abc}(t')}{2\pi} P_{a\to bc}\left(\frac{x}{x'}\right) \right\} \\
&= \exp\left\{ - \int_t^{t_{\max}} \mathrm{d}t' \sum_{a,c} \int \mathrm{d}z \, \frac{\alpha_{abc}(t')}{2\pi} P_{a\to bc}(z) \, \frac{x' f_a(x',t')}{x f_b(x,t')} \right\} ,
\end{aligned}$$

(Initial-State Evolution)

DGLAP for Parton Density

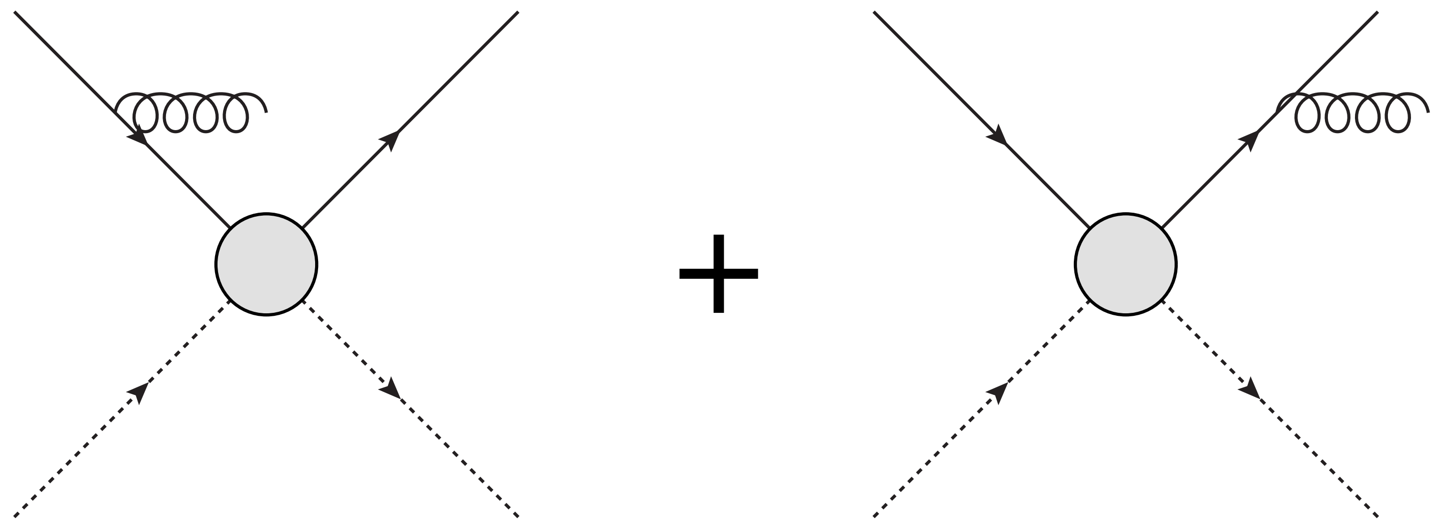
$$\frac{\mathrm{d}f_b(x,t)}{\mathrm{d}t} = \sum_{a,c} \int \frac{\mathrm{d}x'}{x'} f_a(x',t) \frac{\alpha_{abc}}{2\pi} P_{a\to bc}\left(\frac{x}{x'}\right)$$

→ Sudakov for ISR

$$\begin{aligned} \Delta(x, t_{\max}, t) &= \exp\left\{ -\int_t^{t_{\max}} \mathrm{d}t' \sum_{a,c} \int \frac{\mathrm{d}x'}{x'} \frac{f_a(x',t')}{f_b(x,t')} \frac{\alpha_{abc}(t')}{2\pi} P_{a\to bc}\left(\frac{x}{x'}\right) \right\} \\ &= \exp\left\{ -\int_t^{t_{\max}} \mathrm{d}t' \sum_{a,c} \int \mathrm{d}z \frac{\alpha_{abc}(t')}{2\pi} P_{a\to bc}(z) \frac{x' f_a(x',t')}{x f_b(x,t')} \right\} , \end{aligned}$$

Initial-Final Interference

Who emitted that gluon?



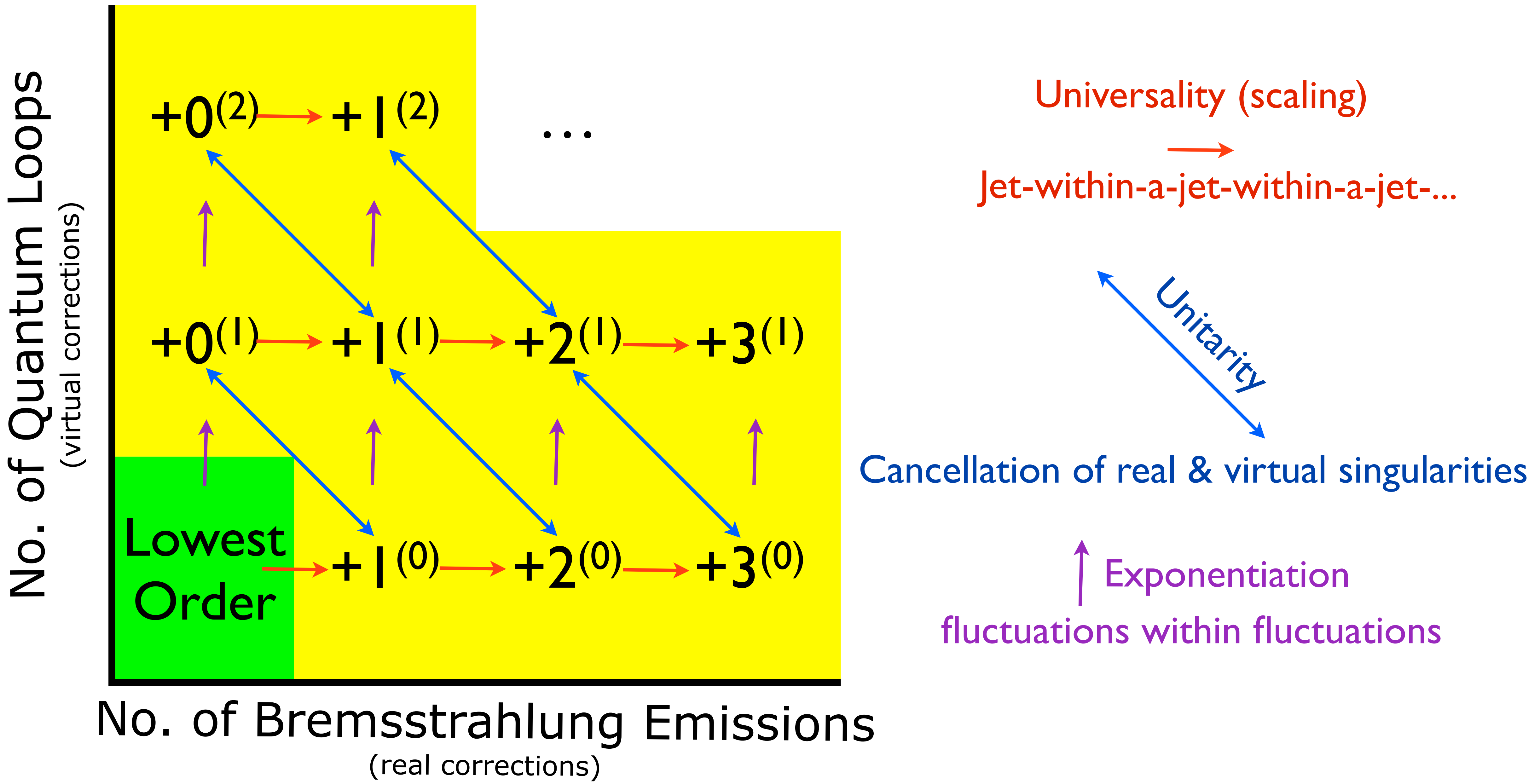
Real QFT = sum over amplitudes, then square → interference (IF coherence)
Respected by dipole/antenna languages (and by angular ordering), but not by conventional DGLAP (→ all PDFs are "wrong")

Separation meaningful for collinear radiation, but not for soft …

Bootstrapped Perturbation Theory

Start from an **arbitrary lowest-order** process (green = QFT amplitude squared)

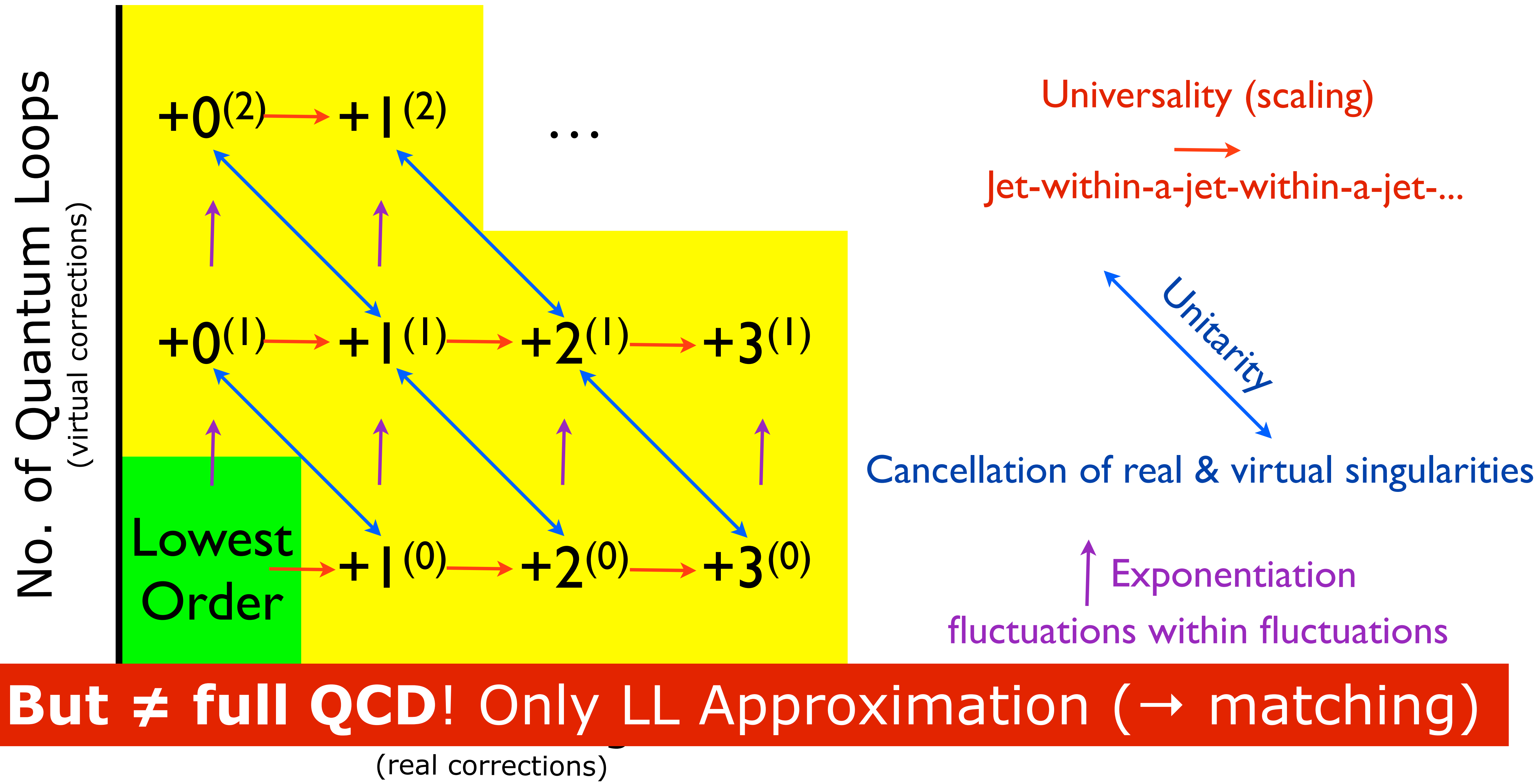
Parton showers generate the bremsstrahlung terms of the rest of the perturbative series (approximate infinite-order resummation)



Bootstrapped Perturbation Theory

Start from an **arbitrary lowest-order** process (green = QFT amplitude squared)

Parton showers generate the bremsstrahlung terms of the rest of the perturbative series (approximate infinite-order resummation)



But ≠ full QCD! Only LL Approximation (→ matching)

The Shower Operator

H = Hard process

Born $\frac{d\sigma_H}{d\mathcal{O}}\Big|_{\text{Born}} = \int d\Phi_H \; |M_H^{(0)}|^2 \; \delta(\mathcal{O} - \mathcal{O}(\{p\}_H))$

{p} : partons

But instead of evaluating O directly on the Born final state, first insert a showering operator

The Shower Operator

H = Hard process

Born $$\left.\frac{d\sigma_H}{d\mathcal{O}}\right|_{\text{Born}} = \int d\Phi_H \ |M_H^{(0)}|^2 \ \delta(\mathcal{O} - \mathcal{O}(\{p\}_H))$$

{p} : partons

But instead of evaluating O directly on the Born final state, first insert a showering operator

Born + shower $$\left.\frac{d\sigma_H}{d\mathcal{O}}\right|_{\mathcal{S}} = \int d\Phi_H \ |M_H^{(0)}|^2 \ \mathcal{S}(\{p\}_H, \mathcal{O})$$

{p} : partons

S : showering operator

The Shower Operator

Born $\left.\frac{d\sigma_H}{d\mathcal{O}}\right|_{\text{Born}} = \int d\Phi_H \; |M_H^{(0)}|^2 \; \delta(\mathcal{O} - \mathcal{O}(\{p\}_H))$

H = Hard process

{p} : partons

But instead of evaluating O directly on the Born final state, first insert a showering operator

Born + shower $\left.\frac{d\sigma_H}{d\mathcal{O}}\right|_{\mathcal{S}} = \int d\Phi_H \; |M_H^{(0)}|^2 \; \mathcal{S}(\{p\}_H, \mathcal{O})$

{p} : partons

S : showering operator

Unitarity: to first order, S does nothing

$$\mathcal{S}(\{p\}_H, \mathcal{O}) = \delta\left(\mathcal{O} - \mathcal{O}(\{p\}_H)\right) \; + \; \mathcal{O}(\alpha_s)$$

The Shower Operator

To ALL Orders

$$S(\{p\}_X, \mathcal{O}) = \Delta(t_{\mathrm{start}}, t_{\mathrm{had}})\delta(\mathcal{O} - \mathcal{O}(\{p\}_X))$$

"Nothing Happens" → "Evaluate Observable"

$$- \int_{t_{\mathrm{start}}}^{t_{\mathrm{had}}} \mathrm{d}t \frac{\mathrm{d}\Delta(t_{\mathrm{start}}, t)}{\mathrm{d}t} S(\{p\}_{X+1}, \mathcal{O})$$

"Something Happens" → "Continue Shower"

All-orders Probability that nothing happens

$$\Delta(t_1, t_2) = \exp\left(- \int_{t_1}^{t_2} \mathrm{d}t \; \frac{\mathrm{d}\mathcal{P}}{\mathrm{d}t} \right)$$

(Exponentiation)
Analogous to nuclear decay
N(t) ≈ N(0) exp(-ct)

The Shower Operator

To ALL Orders

(Markov Chain)

$$S(\{p\}_X, \mathcal{O}) = \Delta(t_{\mathrm{start}}, t_{\mathrm{had}})\delta(\mathcal{O} - \mathcal{O}(\{p\}_X))$$

"Nothing Happens" → "Evaluate Observable"

$$- \int_{t_{\mathrm{start}}}^{t_{\mathrm{had}}} \mathrm{d}t \frac{\mathrm{d}\Delta(t_{\mathrm{start}}, t)}{\mathrm{d}t} S(\{p\}_{X+1}, \mathcal{O})$$

"Something Happens" → "Continue Shower"

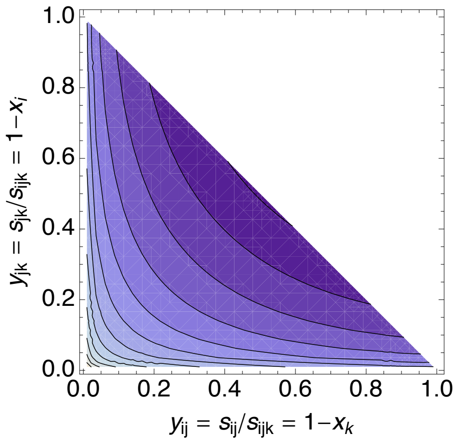
All-orders Probability that nothing happens

$$\Delta(t_1, t_2) = \exp\left(-\int_{t_1}^{t_2} \mathrm{d}t \, \frac{\mathrm{d}\mathcal{P}}{\mathrm{d}t}\right)$$

(Exponentiation)
Analogous to nuclear decay
N(t) ≈ N(0) exp(-ct)

A Shower Algorithm

Note: on this slide, I use results from the theory of Random numbers, interesting in itself but would need more time to give details



1. Generate Random Number, R ∈ [0,1]

Solve equation $R = \Delta(t_1, t)$ for t (with starting scale t_1)

Analytically for simple splitting kernels,
else numerically (or by trial+veto)
→ t scale for next branching

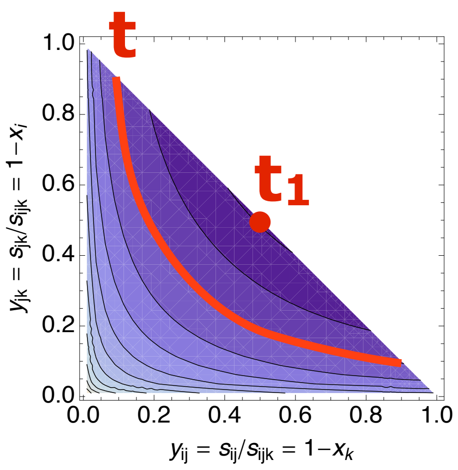
A Shower Algorithm

Note: on this slide, I use results from the theory of Random numbers, interesting in itself but would need more time to give details

1. Generate Random Number, R ∈ [0,1]

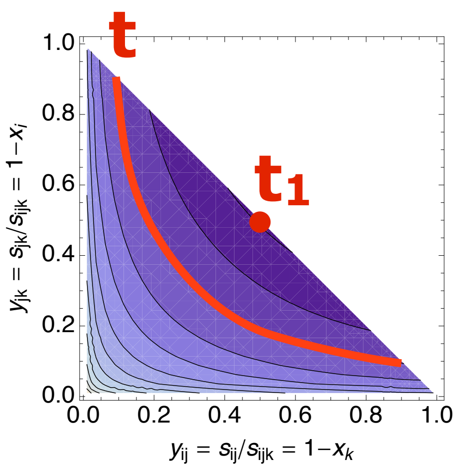
Solve equation $R = \Delta(t_1, t)$ for t (with starting scale t_1)

Analytically for simple splitting kernels,
else numerically (or by trial+veto)
→ t scale for next branching



A Shower Algorithm

Note: on this slide, I use results from the theory of Random numbers, interesting in itself but would need more time to give details



1. Generate Random Number, R ∊ [0,1]

Solve equation $R = \Delta(t_1, t)$ for t (with starting scale t_1)

Analytically for simple splitting kernels,
else numerically (or by trial+veto)
→ t scale for next branching

2. Generate another Random Number, R_z ∊ [0,1]

To find second (linearly independent) phase-space invariant

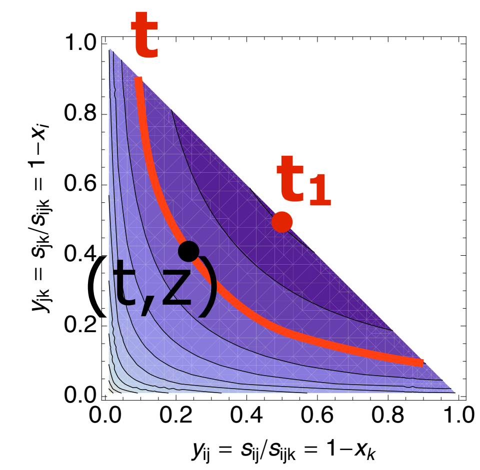
Solve equation $R_z = \frac{I_z(z,t)}{I_z(z_{\max}(t),t)}$ for z (at scale t)

With the "primitive function" $I_z(z,t) = \int_{z_{\min}(t)}^{z} \mathrm{d}z \, \left.\frac{\mathrm{d}\Delta(t')}{\mathrm{d}t'}\right|_{t'=t}$

A Shower Algorithm

Note: on this slide, I use results from the theory of Random numbers, interesting in itself but would need more time to give details

1. Generate Random Number, R ∈ [0,1]



Solve equation $R = \Delta(t_1, t)$ for t (with starting scale t_1)

Analytically for simple splitting kernels,
else numerically (or by trial+veto)
→ t scale for next branching

2. Generate another Random Number, R_z ∈ [0,1]

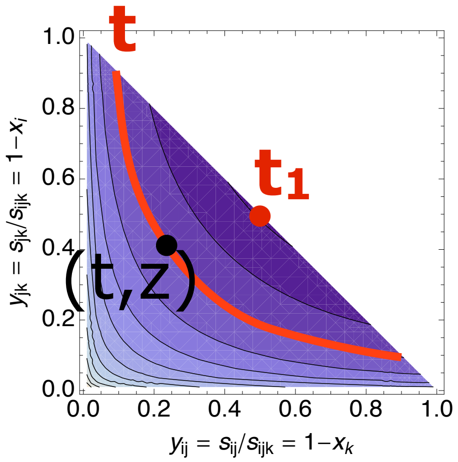
To find second (linearly independent) phase-space invariant

Solve equation $R_z = \frac{I_z(z,t)}{I_z(z_{\max}(t),t)}$ for z (at scale t)

With the "primitive function" $I_z(z,t) = \int_{z_{\min}(t)}^{z} \mathrm{d}z \, \frac{\mathrm{d}\Delta(t')}{\mathrm{d}t'}\bigg|_{t'=t}$

A Shower Algorithm

Note: on this slide, I use results from the theory of Random numbers, interesting in itself but would need more time to give details



1. Generate Random Number, R ∈ [0,1]

Solve equation $R = \Delta(t_1, t)$ for t (with starting scale t_1)

Analytically for simple splitting kernels,
else numerically (or by trial+veto)
→ t scale for next branching

2. Generate another Random Number, R_z ∈ [0,1]

To find second (linearly independent) phase-space invariant

Solve equation $R_z = \frac{I_z(z,t)}{I_z(z_{\max}(t),t)}$ for z (at scale t)

With the "primitive function" $I_z(z,t) = \int_{z_{\min}(t)}^{z} \mathrm{d}z \left.\frac{\mathrm{d}\Delta(t')}{\mathrm{d}t'}\right|_{t'=t}$

3. Generate a third Random Number, R_φ ∈ [0,1]

Solve equation $R_\varphi = \varphi/2\pi$ for φ → Can now do 3D branching

Perturbative Ambiguities

The final states generated by a shower algorithm will depend on

1. The choice of perturbative evolution variable(s) $t^{[i]}$. ← Ordering & Evolution-scale choices

2. The choice of phase-space mapping $\mathrm{d}\Phi_{n+1}^{[i]}/\mathrm{d}\Phi_n$. ← Recoils, kinematics

3. The choice of radiation functions a_i, as a function of the phase-space variables. ← Non-singular terms, Reparametrizations, Subleading Colour

4. The choice of renormalization scale function μ_R.

5. Choices of starting and ending scales. ← Phase-space limits / suppressions for hard radiation and choice of hadronization scale

Perturbative Ambiguities

The final states generated by a shower algorithm will depend on

1. The choice of perturbative evolution variable(s) $t^{[i]}$. ← Ordering & Evolution-scale choices
2. The choice of phase-space mapping $\mathrm{d}\Phi_{n+1}^{[i]}/\mathrm{d}\Phi_n$. ← Recoils, kinematics
3. The choice of radiation functions a_i, as a function of the phase-space variables. ← Non-singular terms, Reparametrizations, Subleading Colour
4. The choice of renormalization scale function μ_R.
5. Choices of starting and ending scales. ← Phase-space limits / suppressions for hard radiation and choice of hadronization scale

→ gives us additional handles for uncertainty estimates, beyond just μ_R
(+ ambiguities can be reduced by including more pQCD → matching!)

Jack of All Orders, Master of None?

Nice to have all-orders solution

But it is only exact in the singular (soft & collinear) limits

→ gets the bulk of bremsstrahlung corrections right, but fails equally spectacularly: for hard wide-angle radiation: **visible, extra jets**

… which is exactly where fixed-order calculations work!

So combine them!

See also: PS, *Introduction to QCD*, TASI 2012, arXiv:1207.2389

Jack of All Orders, Master of None?

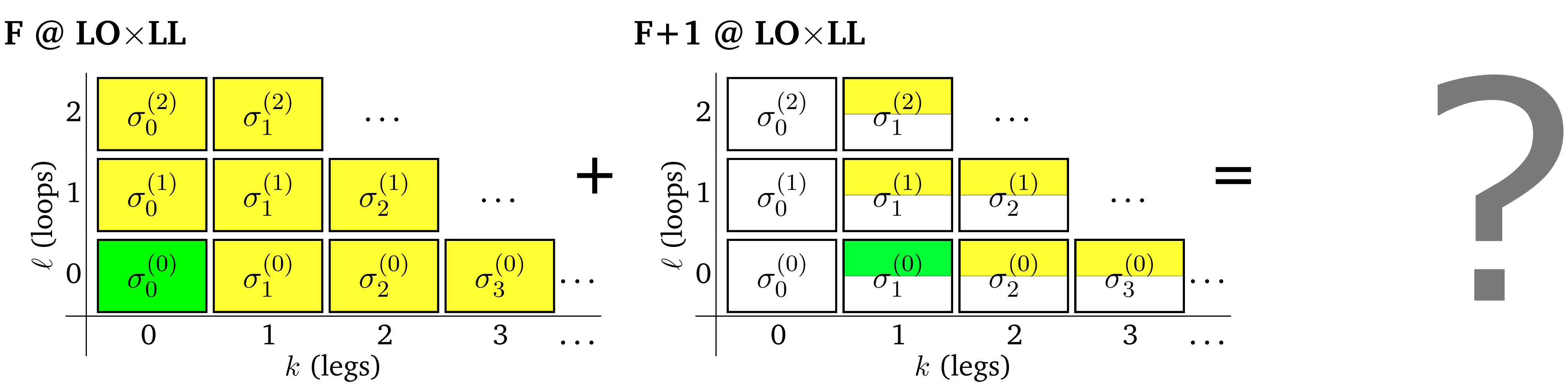
Nice to have all-orders solution

But it is only exact in the singular (soft & collinear) limits

→ gets the bulk of bremsstrahlung corrections right, but fails equally spectacularly: for hard wide-angle radiation: **visible, extra jets**

… which is exactly where fixed-order calculations work!

So combine them!



See also: PS, *Introduction to QCD*, TASI 2012, arXiv:1207.2389

Jack of All Orders, Master of None?

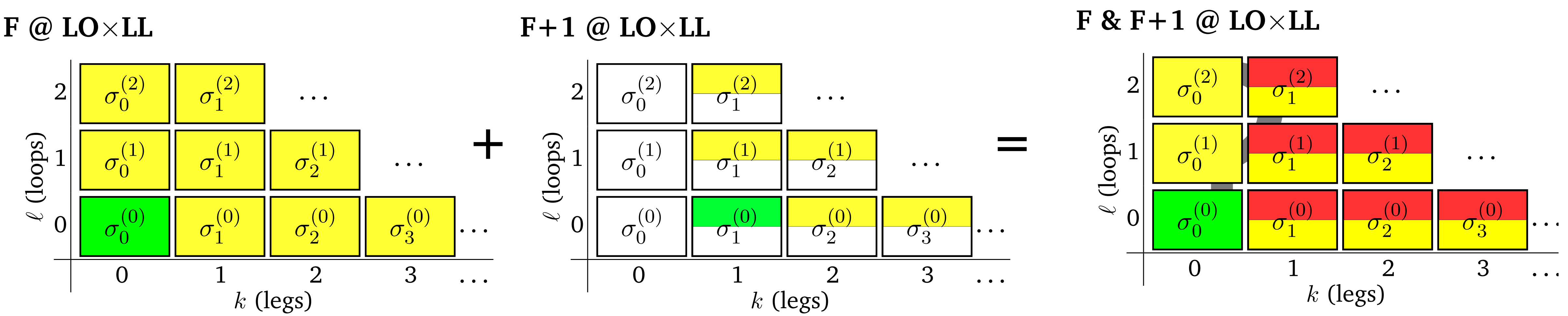
Nice to have all-orders solution

But it is only exact in the singular (soft & collinear) limits

→ gets the bulk of bremsstrahlung corrections right, but fails equally spectacularly: for hard wide-angle radiation: **visible, extra jets**

… which is exactly where fixed-order calculations work!

So combine them!



See also: PS, *Introduction to QCD*, TASI 2012, arXiv:1207.2389

Jack of All Orders, Master of None?

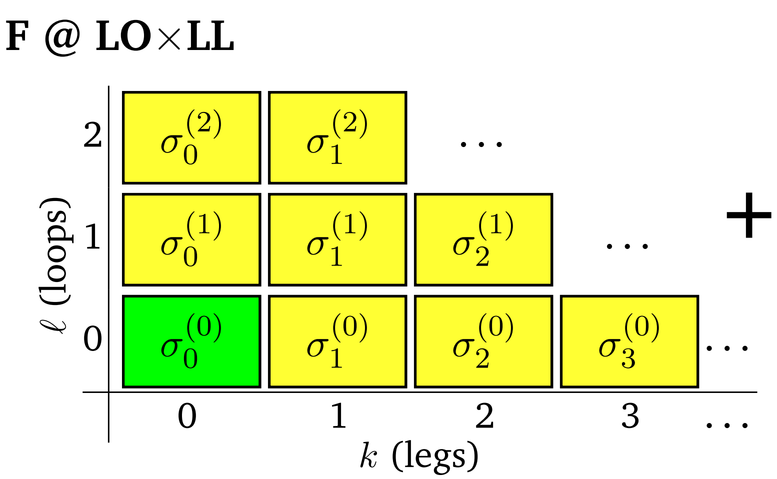
Nice to have all-orders solution

But it is only exact in the singular (soft & collinear) limits

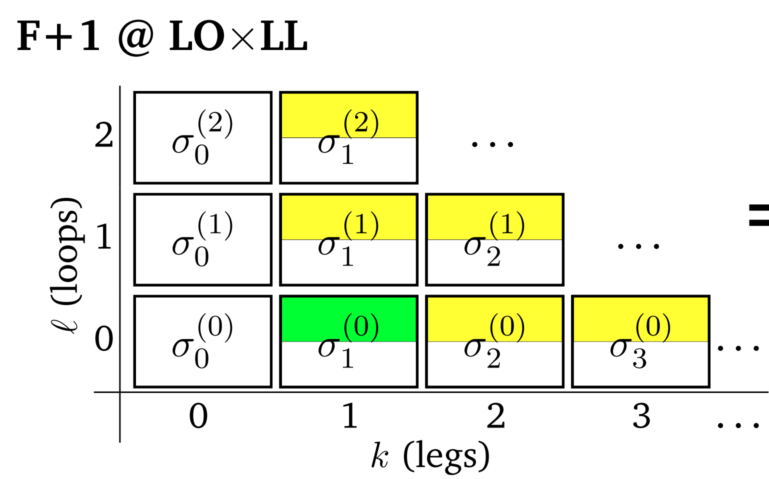
→ gets the bulk of bremsstrahlung corrections right, but fails equally spectacularly: for hard wide-angle radiation: **visible, extra jets**

… which is exactly where fixed-order calculations work!

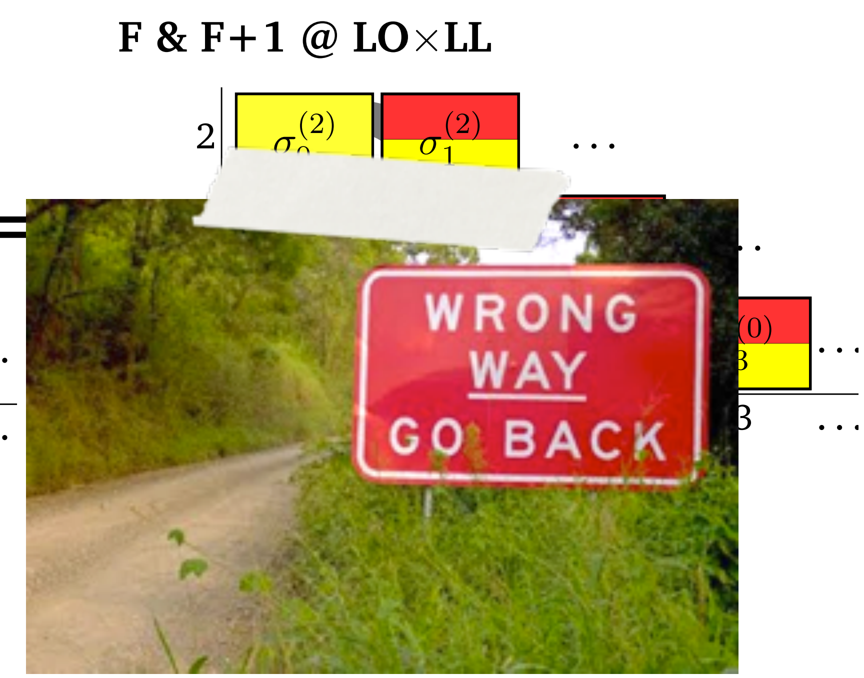
So combine them!



+



=



See also: PS, *Introduction to QCD*, TASI 2012, arXiv:1207.2389

Jack of All Orders, Master of None?

Nice to have all-orders solution

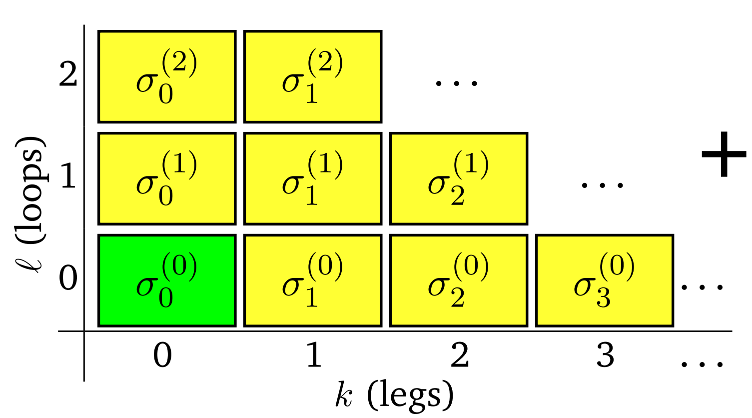
But it is only exact in the singular (soft & collinear) limits

→ gets the bulk of bremsstrahlung corrections right, but fails equally spectacularly: for hard wide-angle radiation: **visible, extra jets**

… which is exactly where fixed-order calculations work!

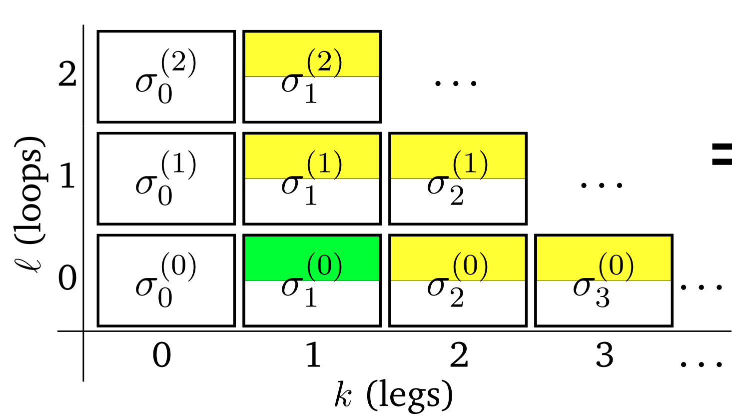
So combine them!

F @ LO×LL



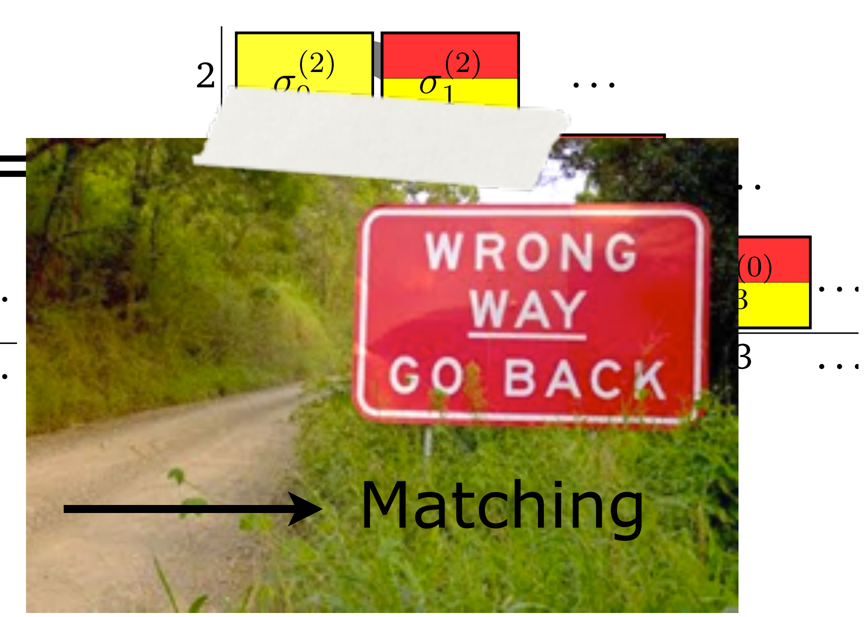
+

F+1 @ LO×LL



=

F & F+1 @ LO×LL



See also: PS, *Introduction to QCD*, TASI 2012, arXiv:1207.2389

Summary: Two ways to compute Quantum Corrections

Standard Paradigm: consider a single physical system; a single physical process

- Explicit solutions (to given perturbative order)
  - Standard-Model: typically NLO or NNLO
  - Beyond-SM: typically LO or NLO
- Limited generality

LO: Leading Order (Born)
NLO = Next-to-LO, …

Event generators: consider all possible physical processes (within perturbative QFT)

- Approximate solutions
  - Process-dependence = subleading correction (→ matching)
- Maximum generality
  - Emphasis is on universalities; physics
  - Common property of all processes is, for instance, limits in which they factorize!

Summary: Parton Showers

Aim: generate events in as much detail as mother nature

→ Make stochastic choices ~ as in Nature (Q.M.) → Random numbers

Factor complete event probability into separate universal pieces, treated independently and/or sequentially (Markov-Chain MC)

Improve lowest-order perturbation theory by including 'most significant' corrections

Resonance decays (e.g., $t \to bW^+$, $W \to qq'$, $H^0 \to \gamma^0\gamma^0$, $Z^0 \to \mu^+\mu^-$, …)

Bremsstrahlung (FSR and ISR, exact in collinear and soft* limits)

Hard radiation (matching, discussed tomorrow)

Hadronization (strings/clusters, discussed tomorrow)

Additional Soft Physics: multiple parton-parton interactions, Bose-Einstein correlations, colour reconnections, hadron decays, …

Coherence*

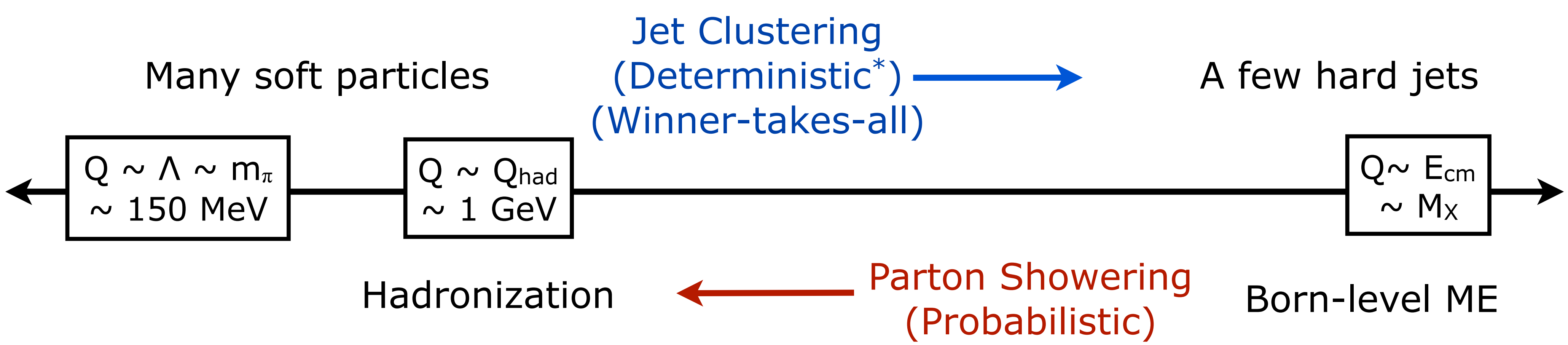
Soft radiation → Angular ordering or Coherent Dipoles/Antennae

See also: **1)** MCnet Review (long): Phys.Rept. 504 (2011) 145-233 and/or **2)** PDG Review on Monte Carlo Event Generators, and/or **3)** PS, TASI Lectures (short): arXiv:1207.2389

Jets vs **Parton Showers**

Jet clustering algorithms

Map event from low E-resolution scale (i.e., with many partons/hadrons, most of which are soft) to a higher E-resolution scale (with fewer, hard, IR-safe, jets)



Parton shower algorithms

Map a few hard partons to many softer ones

Probabilistic → closer to nature.
Not uniquely invertible by any jet algorithm*

(* See "Qjets" for a probabilistic jet algorithm, arXiv:1201.1914)
(* See "Sector Showers" for a deterministic shower, arXiv:1109.3608)