Phase-space integration at NLO

1. $\hat{\sigma}^{NLO} = \int d^4\Phi_n \, B(\Phi_n)$

$+ \int d^4\Phi_n \int_{loop} V(\Phi_n) \, d^4\ell$ ← separately divergent

$+ \int d^4\Phi_{n+1} \, R(\Phi_{n+1})$ ↙ but sum is finite (after P.S. integration) [for infra-red safe observables].

~~Subtraction method: Add and subtract a term that renders~~

Use dimensional regularization to render the integrals finite.

⇒ cannot be used when using M.C. integration

↳ only integer number of dimensions allowed when doing numerical integration

TWO EXTRA PAGES HERE!

<u>TRICK</u> Subtraction method: Add and subtract a term that renders ~~both~~ real and virtual contributions separately finite:

$$\hat{\sigma}^{NLO} = \int d^4\Phi_n \, B(\Phi_n) + \int d^4\Phi_n \Big[\overbrace{\int_{loop} d^d\ell \, V(\Phi_n) + \int A(\tilde{\Phi}_{n+1})}^{\text{finite}} \Big]_{\epsilon=0}$$

poles in ϵ

$$+ \int d^4\Phi_{n+1} \Big[R(\Phi_{n+1}) - A(\tilde{\Phi}_{n+1}) \Big]$$

divergent

Subtraction vs. Phase-space Slicing. 1A.1

Suppose we want to compute the integral

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-2\epsilon} f(x) \quad , \text{ where } f(x) = \frac{g(x)}{x} \text{ and}$$

$g(x)$ is <u>not</u> divergent

* Subtraction:

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-2\epsilon} f(x) = \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-2\epsilon}\left[f(x) - \frac{g(0)}{x} + \frac{g(0)}{x}\right]$$

$$= \lim_{\epsilon\to 0}\int_0^1 dx \left[\frac{g(x)-g(0)}{x^{1+2\epsilon}} + g(0)\,\frac{x^{-2\epsilon}}{x}\right]$$

$$= \lim_{\epsilon\to 0}\left[g(0)\int_0^1 dx\,\frac{x^{-2\epsilon}}{x} + \int_0^1 dx\,\frac{g(x)-g(0)}{x^{1+2\epsilon}}\right]$$

$$= \lim_{\epsilon\to 0} g(0)\,\frac{-1}{2\epsilon} + \int_0^1 dx\,\frac{g(x)-g(0)}{x} .$$

* P.S. slicing: introduce a small parameter δ

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-2\epsilon} f(x) = \lim_{\epsilon\to 0}\left[\int_0^\delta dx\, x^{-2\epsilon} f(x) + \int_\delta^1 dx\, x^{-2\epsilon} f(x)\right]$$

$$= \lim_{\epsilon\to 0}\left[\int_0^\delta dx\,\frac{g(0)}{x}\,x^{-2\epsilon} + \int_\delta^1 dx\,\frac{g(x)}{x^{1+2\epsilon}}\right]$$

$$= \lim_{\epsilon\to 0} -\frac{\delta^{-2\epsilon}}{2\epsilon}\, g(0) + \int_\delta^1 dx\,\frac{g(x)}{x}$$

$$= \lim_{\epsilon\to 0}\left[\frac{-1}{2\epsilon} + \log\delta\right] g(0) + \int_\delta^1 dx\,\frac{g(x)}{x}$$

Subtraction vs PS slicing II

1A-2

The poles in $\frac{1}{\epsilon}$ are the same in both cases. The virtual corrections have a similar pole → cancellation according to KLN.

~~Including virtual~~ So, we have

Slicing: $$\int_\delta^1 dx \; \frac{g(x)}{x} + \log \delta \; g(0)$$

Subtraction: $$\int_0^1 dx \; \frac{g(x) - g(0)}{x}$$

NOTE: different kinematics for counter-event!

* Terms $O(\delta)$ are neglected in slicing (Some can be included, but almost never done in practice)

* In slicing, one ~~but~~ has to prove that the results are independent of δ when $\delta \to 0$. This is typically only done for σ_{tot}, but should in principle be done for every observable.

* Subtraction results do not involve approximations.

* Both methods feature ~~large~~ cancellations between large numbers. If $\lim_{x\to 0} O(x) \neq O(0)$ or the bin-size is too small, numerical instabilities render the computation useless

  → We already know that! KLN is not sufficient. One must also have infra-red safe observables and one cannot ask for infinite resolution.

* Subtraction more flexible than slicing → hard to think about slicing for NNLO computations.

(1B)

$A(\tilde{\Phi}_{n+1})$ should be defined such that

1. it exactly matches the singular behavior of $R(\tilde{\Phi}_{n+1})$

2. its form is convenient for M.C. integration techniques

3. it is exactly integrable in d dimensions over the one-parton subspace (leading the soft and/or collinear divergences)

4. it's universal, i.e. "process independent"

   → "overall factor" times Born process

Subtraction methods

1. Catani-Seymour subtraction

- most used
- clearly written paper on how to implement
- clear picture starting from soft divergence
- proven to ~~...~~ work in simple as well as (very) complicated processes
- automation ~~available~~ in publicly available packages:
  MadDipole, AutoDipole, Helac-dipoles, Sherpa

2. FKS subtraction (Frixione, Kunszt, Signer)

- not so well ~~spread~~ known
- (probably) more efficient because less subtraction needed
- collinear divergences are the starting point
- proven to work for simple as well as (very) complicated processes.
- preferred method when interfacing NLO to parton shower for an exclusive prescription of final state.
- Implemented in the MadFKS code, which will become publicly available in the ~~next~~ near ~~future~~ future.

FKS Subtraction

Easiest to explain when starting from the real emission process.

$$d\sigma^R = |M^{n+1}|^2 \, d\Phi_{n+1}$$

$\hookrightarrow$ blows up like $\frac{1}{\xi_i^2}\frac{1}{1-y_{ij}}$ with $\underbrace{\begin{matrix}\xi_i \text{ energy of parton } i \\ y_{ij} \text{ is } \cos\theta_{ij}\end{matrix}}_{\text{c.o.m. frame or lab frame.}}$

TRICK: partition the phase-space such that each partition has at most one soft and one collinear divergence

$$d\sigma^R = \sum_{ij} S_{ij} \, |M^{n+1}|^2 \, d\Phi_{n+1} \quad , \text{ with } \sum_{ij} S_{ij} = 1$$

* Precise definition of S_{ij} not important as long as it damps all singularities not related to particle i going soft or particle ij becoming collinear.

Use plus distributions to regulate the singularities

no square needed, because $d\Phi_n$ contains ξ_i as well.

$$d\tilde\sigma^R = \sum_{ij} \left(\frac{1}{\xi_i}\right)_+ \left(\frac{1}{1-y_{ij}}\right)_+ \underbrace{S_{ij} \, \xi_i \, (1-y_{ij}) \, |M^{n+1}|^2 \, d\Phi_{n+1}}_{\text{finite all over phase-space.}}$$

Remember that $\int dx \left(\frac{1}{x}\right)_+ f(x) = \int dx \, \frac{f(x) - f(0)}{x}$

$f(x)$: Event; $f(0)$: counter event

FKS subtraction continued.

- Number of partitions scales like n^2, where n is the number of external partons
- maximally <u>three</u> counter events per partition

1. soft $\xi_i \to 0$
2. collinear $y_{ij} \to 1$
3. soft-collinear $\xi_i \to 0$ and $y_{ij} \to 1$

⇒ In practice this is just one: they can have the same Born kinematics, and can therefore be summed together.

general remark

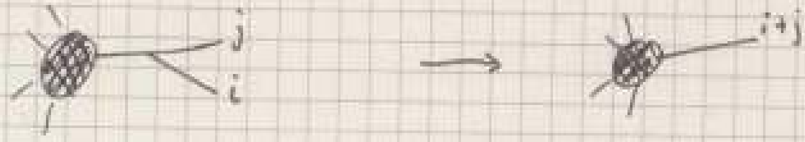
- the counter-events have different kinematics than the real-ammision: the could end-up in different bins when plotting a distribution. Also when cuts are applied, the event could pass the cuts, while the counter events don't (or vice versa).

- Even though $\xi_i^2(1-y_{ij})\,|M^{n+1}|^2$ is finite all over phase-space, to evalute numerically we need to cancel the divergences analytically. It's known how to do this:
  * for soft use Born times eikonal factors
  * for collinear use Born times Altarelli-Parisi splitting functions

From the eikonals or AP-splitting functions it's trivial to factor out the required factor ξ_i or $(1-y_{ij})$.

- The counterevents for the virtual can be obtained by integrating <u>analytically</u> over the <u>one-parton phase-space</u>, i.e. over the eikonals and AP-splitting functions, to obtain the explicit poles in

Kinematics of counterevents.



If i and j are ~~two~~ the two (on-shell) massless partons that are present in the splitting that leads to a singularity, for the counter-events we need to combine their momenta to a new <u>on-shell</u> parton that's the sum of $i+j$.

→ Problem : this is not possible without changing any of the momenta of the other partons

→ Solution FKS : shift all momenta a bit

→ Solution CS-dipoles : shift only the one parton that's color-connected with parton i in the soft limit.

→ When making cuts or plotting histograms, events and counter-events might end-up in different bins

→ Use IR-safe observables and not <u>infinite</u> resolution.

Catani - Seymour dipole subtraction

* instead of partitioning the phase-space, all singularities are subtracted in one go.

  → less integrals to do, but they are more complicated

  ~~cf.~~

* $$d\tilde{\sigma}^R = \left[\prod_{[ij]} \left(\frac{1}{\xi_i}\right)_+ \left(\frac{1}{1-y_{ij}}\right)_+ J_{ij}\,(1-y_{ij})\right] |M^{n+1}|^2 \, d\Phi_{n+1}$$

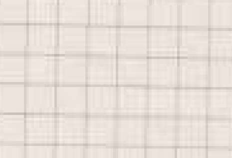
  over possible ij-pairs

* In fact, it's a bit more complicated than this;

  • Using a triple product over ijk, where parton k absorbes the recoil momentum. ~~[illegible]~~

  • Can be defined using invariants ⇒ counter-events can be computed in another frame [useful for ~~[illegible]~~ the integrated subtraction term]. A feature you ~~do~~ ~~define~~ want to have at NNLO.

  → This is also where the name comes from. The two dipoles V_{ijk} and V_{ikj} subtract the soft and collinear divergences for a parton radiated from ~~the~~ dipole formed by j and k



Summary of Integration at NLO

7

* Virtual and real-emmision corrections are separately divergent; only their sum is finite (after P.S. integration).

* Numerical P.S. integration → cannot use dimensional regularization for the real emmision corrections as is.

* two solutions:

  1. P.S. slicing

  2. subtraction

* Generally assumed that subtraction is "better": no biasses (exact) and proven to work for very complicated processes.

* two ~~widely~~ widely used methods for subtraction:

  1. FKS-subtraction

  2. CS-dipole subtraction.

* Not possible to do event unweighting:

  - even though the integrals are finite, they are not bounded. $\left(\int dx \frac{1}{\sqrt{x}}\right)$. ~~Furthermore counter~~, so no maximum to unweight against: a single event can have an "infinite" weight.

  - Furthermore event and counterevents have different kinematics → which one to use for the unweighted events?

* Not easy to match with shower → possible double counting