Axiloop: a tool for the symbolic calculation of splitting kernels at higher orders

Oleksandr Gituliar

oleksandr@gituliar.org

Institute of Nuclear Physics
Polish Academy of Sciences,
Cracow, Poland
http://www.ifj.edu.pl

5 September 2012

Splitting kernels and their application

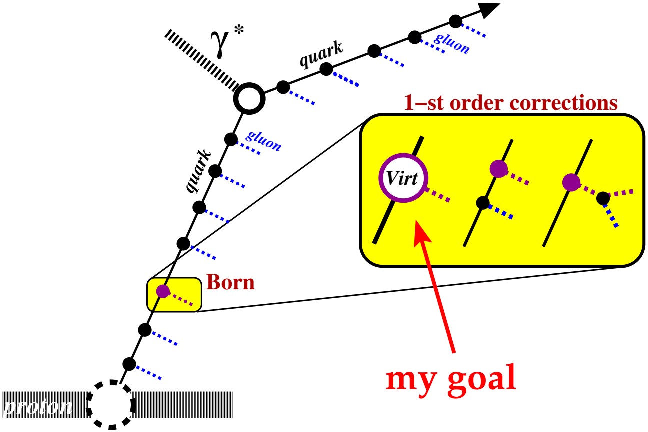
- **Inclusive splitting kernels** ($P_{\rm LO}(\alpha_s, x)$, $P_{\rm NLO}(\alpha_s, x)$, etc.) are formally defined in context of the *collinear factorization theorem* [EGMPR79] and calculated in [CFP80]. They are mainly used to solve DGLAP evolution equation and to build Monte-Carlo (ladder) in a one dimension.
- **Exclusive splitting kernels** ($P_{\rm LO}(\alpha_s, \vec{q})$, $P_{\rm NLO}(\alpha_s, \vec{q}_1, \vec{q}_2)$, etc.) are not formally defined yet. They are required to build complete Monte-Carlo event generators.

Brief history of Monte-Carlo QCD

- LO Hard Process + LO Ladder – Pythia, Herwig (1980s)
- NLO Hard Process + LO Ladder – MC@NLO, PowHEG (2000s)
- NLO Hard Process + NLO Ladder – KrKMC (ongoing)

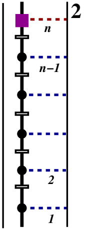
NLO corrections to the ladder in parton shower MC

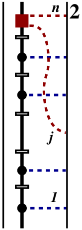
from S.Jadach 2011



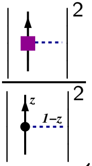
LO with NLO-corrected end-ladder splitting kernel

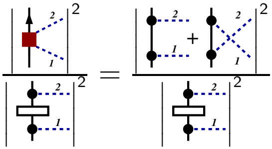
from S.Jadach 2011

$$PDF(x,Q) = e^{-S_{ISR}}$$ 

$$+\, e^{-S_{ISR}} \sum_{j=1}^{n-1}$$ 

,

where $\beta_0^{(1)}(\vec{k}_1) =$ 

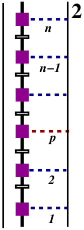
, and $W(\vec{k}_1, \vec{k}_2) =$ 

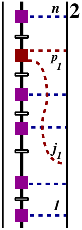
$- 1.$

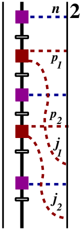
my goal

NLO-corrected splitting kernels all over the ladder

from S.Jadach 2011

$PDF(x,Q) = e^{-S_{ISR}} \sum_{n=0}^{\infty} \Bigg\{$


$+ \sum_{p_1=1}^{n} \sum_{j_1=1}^{p_1-1}$


$+ \sum_{p_1=1}^{n} \sum_{p_2=1}^{p_1-1} \sum_{\substack{j_1=1 \\ j_1 \neq p_2}}^{p_1-1} \sum_{\substack{j_2=1 \\ j_2 \neq p_1, j_2}}^{p_2-1}$


$+ \dots \Bigg\}$

The idea is to apply all possible combinations of NLO corrections on top of LO ladder of emissions.

Technical issues

For inclusive calculations it doesn't matter which gauge you use since a final result is gauge-independent, e.g. parton distribution functions.

- Axial gauge has nice factorization properties, i.e. two-particle-irreducible kernels and propagation of only physical polarizations, suitable for exclusive Monte-Carlo.

That choice requires a prescription for dealing with singularities specific only to the axial-gauge.

- Principal Value prescription (PV) does not introduce unphysical ghosts (in contrast to Mandelstam-Leibbrant (ML) prescription), which one would better to omit in exclusive Monte-Carlo simulations.

More details in [EGMPR79], [CFP80], [Hei98].

Axiloop package and its purpose

With Axiloop we aim to:

1. Automate splitting functions calculation.
2. Calculate inclusive splitting kernels at least to NLO order.
3. Provide exclusive splitting kernels at NLO order.

Tools we use:

- Wolfram Mathematica 8 and Workbench 2;
- Git and GitHub (`https://github.com/gituliar/axiloop.git`).

Our guide principles:

- readability matters;
- if it's not documented it's useless;
- if it's not tested it doesn't work.

Architectural design of Axiloop (outline)

We employ techniques developed in [CFP80], [Hei98] and perform the following calculation steps, which naturally suggest architectural design of the package:

1. **Calculate trace**
2. **Regularize** infra-red singularities
3. **Integrate over loop momenta**
4. **Renormalize** ultra-violet singularities
5. **Integrate over final-state momenta**

The remaining slides explain each of the steps above.

Axiloop: trace calculation

It is the most easy step (but not trivial!), since we employ Tracer to get the job done. Tracer is a Mathematica package for gamma algebra in arbitrary dimensions written by M.Jamin and M.Lautenbacher back in 1991, but still serves its purpose nowadays.

Syntax notation inherited from Tracer:

```
In[1]:= p.q;            /* a scalar product of two vectors      */
In[2]:= p.{mu};         /* vector's components, p_mu            */
In[3]:= {mu}.{nu};      /* a metric tensor, g_{mu nu}           */
```

This step is done in $n = 4 + 2\epsilon$ dimensions, as [CFP80] suggests, e.g.

```
In[1]:= {mu}.{mu}
Out[1]= 4 + 2 eps
```

Axiloop: regularization of IR singularities

Since we deal with massless QCD and on-shell momenta (namely, incoming and ones that cross a cut line) we should be careful about IR singularities.

E.g., the following loop-momenta integral contains both UV and IR poles, where p is on-shell massless momentum and $m = 4 - 2\eta$:

$$\int \frac{d^m l}{(2\pi)^m} \frac{1}{l^2 (l-p)^2} = i(4\pi)^{-2+\eta} \frac{\Gamma(1+\eta)}{\eta} \frac{\beta(1-\eta, 1-\eta)}{(p^2)^\eta}.$$

The solution is to keep all momenta off-shell, i.e. $p^2 \neq 0$, until we integrate over loop momenta and renormalize UV singularities.

Axiloop: integration over loop momenta

- The presence of an axial-gauge-specific denominator in the gluon propagator, i.e. $1/l\cdot n$ (where $n = (1, 0, 0, -1)$ is an axial gauge vector), complicates loop integration and introduces spurious poles. They are regularized with Principal Value prescription [CFP80]:

$$\frac{1}{l\cdot n} \to \frac{l\cdot n}{(l\cdot n)^2 + \delta^2 (P\cdot n)^2},\ \delta \to 0.$$

- The number of time-space dimensions is $4 - 2\eta$.

The integration is delegated to IntegrateLoop function from Axiloop:

$$\int_{\mathrm{PV}} \texttt{expr}\, \mathrm{d}^{4-2\eta}\texttt{l} \ \to\ \texttt{IntegrateLoop[expr, l]}$$

Axiloop: integration over loop momenta (example)

Inside `IntegrateLoop` we:

1. Identify `l`-dependent terms within initial expression and convert them into the internal representation.
2. Transform/simplify internal representation to a set of known integrals.
3. Convert internal representation back to the mathematical notation.

```
/* A trivial example of a loop integral ...                              */
In[1]:= IntegrateLoop[1/(l.l (l-k).(l-k) l.n), l]
Out[1]= I(4Pi)^(-2+eta) Gamma[1+eta] (k.k)^(-eta) P0/x

/* ... and a more complicated another one.                               */
In[2]:= IntegrateLoop[l.p/(l.l (l-p).(l-p) l.n (l-k).n), l]
Out[2]= -I(4Pi)^(-2+eta) Gamma[1+eta] (p.p)^(1-eta) (B1+B3)/(2x)
```

where `P0`, `B1`, and `B3` are known functions [Hei98] of x, η, and δ, e.g.

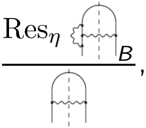
$$\mathrm{B1} = \frac{\beta(1-\eta, 1-\eta)}{\eta}, \quad \mathrm{B3} = \frac{2 - I_0(\delta)}{\eta} + I_1(\delta) - \mathrm{Li}_2(1) + 4;$$

spurious poles contain δ-dependent terms.

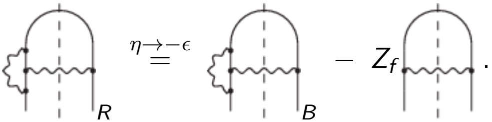
Axiloop: renormalization of UV singularities

We define renormalization constant, Z_f, as

$$Z_f(\alpha_s, x, \delta) = \lim_{\epsilon \to 0}$$



and renormalize UV-singularities as follows



Note that at this point we deal with two dimensional regularization constants, ϵ and η, which regularize IR and UV poles respectively.

Axiloop: integration over final-state momenta

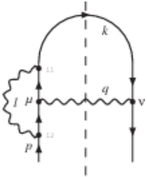
NLO splitting kernels have up to 2 particles in final state to integrate over:

- 0-particle integration is trivial
- 1-particle integration is implemented using Mathematica's `Integrate` function since a phase space is quite simple:

$$
\begin{aligned}
\mathrm{PS}_1^{\mathrm{r}} &= 2\pi z \int \frac{\mathrm{d}^m k}{(2\pi)^m} \delta(x-z)\, \delta\big((p-k)^2\big) \\
&= \frac{1}{16\pi^2} \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \int_0^{Q^2} \mathrm{d}k^2 k^{-2\epsilon}\, x(1-x)^{-\epsilon}
\end{aligned}
$$

- 2-particle integration is not implemented yet.

Example: splitting kernel of topology (c)




```
In[1]:= Kernel[
  FP[k]**FV[i1]**FP[l-k]**FV[mu]**FP[l-p]**FV[i2]**GP[i1,i2,l],
  FPx[p]**GPx[mu,nu,q],
  FV[nu]**FP[k],
  LO
]
```

This code returns:

- exclusive and inclusive expressions for splitting kernels; and
- renormalization constant, $Z(\alpha_s, x, \delta)$.

(`Axiloop.nb` file in our Github repository contains explicit results; they are quite long to be shown here.)

Summary

Already implemented:

- a complete algorithm for inclusive splitting kernels;
- integration over one loop and one final-state momenta;
- renormalization procedure;
- two topologies contributing to non-singlet (fermionic) kernels.

Work in progress:

- more topologies of splitting kernels;
- integration over two final-state momenta;
- exclusive splitting kernels in 4 dimensions suitable for Monte-Carlo.

References

G. Curci, W. Furmanski, and R. Petronzio.
Evolution of parton densities beyond leading order: the non-singlet case.
Nucl. Phys., B175:27–92, 1980.

R. Ellis, H. Georgi, M. Machacek, H. Politzer, and G. Ross.
Perturbation theory and the parton model in qcd.
Nucl. Phys., B152:285–329, 1979.

Gudrun Heinrich.
Improved techniques to calculate two-loop anomalous dimensions in QCD.
PhD thesis, Swiss Federal Institute of Technology, Zurich, 1998.