Towards three-loop QCD corrections to the time-like splitting functions

Oleksandr Gituliar

Institute of Nuclear Physics (Cracow)

Sven-Olaf Moch

University of Hamburg

Radcor-Loopfest Symposium
Los Angeles (USA)
15 Jun 2015

Three-loop time-like $g \to q$ splitting function

- **three-loop** aka **NNLO** aka $\mathcal{O}(\alpha_s^3)$
- **time-like** hard scale $q^2 > 0$
- off-diagonal $P^T_{qg}(x, \alpha_s(q^2))$ term

Splitting functions in perturbation theory

Splitting Functions

- universal quantities in QCD (process independent)
- govern collinear evolution in hard scattering processes with hadrons
  - in initial state (a space-like hard-scale $Q^2 = -q^2 > 0$)

$$\frac{d}{d\ln q^2}\, f_a^h(x, q^2) \;=\; \int_x^1 \frac{dz}{z}\, P_{ab}^S\left(z, \alpha_s(q^2)\right) f_b^h\left(\frac{x}{z},\, q^2\right)$$

  - in final state (a time-like hard-scale $Q^2 = q^2 > 0$)

$$\frac{d}{d\ln q^2}\, D_a^h(x, q^2) \;=\; \int_x^1 \frac{dz}{z}\, P_{ba}^T\left(z, \alpha_s(q^2)\right) D_b^h\left(\frac{x}{z},\, q^2\right)$$

- can be computed in perturbation theory

$$P_{ba}^T\left(x, \alpha_s(q^2)\right) = \frac{\alpha_s}{4\pi}\, P_{ba}^{(0)\,T}(x) + \left(\frac{\alpha_s}{4\pi}\right)^2 P_{ba}^{(1)\,T}(x) + \left(\frac{\alpha_s}{4\pi}\right)^3 P_{ba}^{(2)\,T}(x) + \ldots$$

Splitting Functions at NLO: available results

Space-like

- axial gauge
  - Principal Value
    Curci, Furmanski, Petronzio '80;
    Ellis, Vogelsang '98
  - Mandelstam-Leibbrandt
    Bassetto, Heinrich, Kunszt, Vogelsang '98
  - New Principal Value
    OG, Jadach, Skrzypek, Kusina '14
- Feynman gauge Floratos, Kounnas, Lacaze '81
- Mellin space Moch, Vermaseren '99

Time-like

- axial gauge
  - Principal Value
    Furmanski, Petronzio '80
- Feynman gauge Floratos, Kounnas, Lacaze '81
- Mellin space Mitov, Moch '06
- analytic continuation ($P^S_{ab} \to P^T_{ba}$)
  Stratmann, Vogelsang '96;
  Blumlein, Ravindran, van Neerven '00;
  Moch, Vogt '07

Analytic Continuation Technique

- Analytic continuation in energy $-q^2 \to +q^2$
  - Drell-Yan-Levy relation Drell, Levy, Yan '70
  - Gribov-Lipatov relation Gribov, Lipatov '72

- Relation between space- and time-like fragmentation functions

$$F^T(x) = -x\, F^S\left(\frac{1}{x}\right)$$

- Examples

$$p_{qq}^{T(0)}(x) = p_{qq}^{S(0)}(x) = \frac{1+x^2}{1-x} \qquad p_{gq}^{T(0)}(x) = p_{qg}^{S(0)}(x) = x + (1-x)^2$$

$$p_{qg}^{T(0)}(x) = p_{gq}^{S(0)}(x) = \frac{1+(1-x)^2}{x} \qquad p_{gg}^{T(0)}(x) = p_{gg}^{S(0)}(x) = \frac{(1-x+x^2)^2}{x(1-x)}$$

- Beware: naive version of these relations is not valid beyond LO

Splitting Functions at NNLO: available results

Space-like

- Mellin space Moch, Vermaseren, Vogt '04

Time-like

- analytic continuation ($P^S_{ab} \to P^T_{ba}$)
  - NNLO non-singlet
    Mitov, Moch, Vogt '06
  - NNLO singlet (P^T_{qq}, P^T_{gg})
    Moch, Vogt '07
  - NNLO singlet (P^T_{gq}, P^T_{qg})
    Almasy, Moch, Vogt '11

"In summary, these considerations are still not sufficient to definitely fix the right-hand-side of $P^{T(2)}_{qg}$.
As an estimate of the remaining uncertainty we suggest to use the offset: ..." Almasy, Moch, Vogt '11

Introduction: Summary

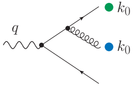
Three-loop time-like $P_{qg}^{T(2)}$ splitting function

is still missing and should be calculated directly.

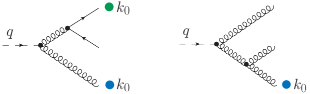
Outline:

1. The Method
    - mass factorization
    - fragmentation functions for $e^+e^- \to \gamma^*, \phi^* \to$ partons
2. Final-State Integration (analytic)
    - IBP reduction
    - differential equations for masters
    - choice of the appropriate basis for masters
    - boundary conditions

Cross-section at LO



P^T_{qq}, P^T_{gq}



P^T_{qg}, P^T_{gg}

- Unpolarized differential cross-section

$$\frac{1}{\sigma_{\rm tot}} \frac{d^2\sigma^H}{dx\, d\cos\theta} = \frac{3}{8}(1+\cos^2\theta)\, \mathcal{F}_T(x) + \frac{3}{4}\sin^2\theta\, \mathcal{F}_L(x) + \frac{3}{4}\cos\theta\, \mathcal{F}_A(x)$$

- Mass factorization relations Vermaseren, Vogt, Moch '05

$$\mathcal{F}^{(1)}_{T,g}(x) = -\frac{1}{\epsilon} P^{(0)}_{gq}(x) + c^{(1)}_{T,g}(x) + \epsilon\, a^{(1)}_{T,g}(x) + \epsilon^2 b^{(1)}_{T,g}(x)$$

$$\mathcal{F}^{(1)}_{L,g}(x) = c^{(1)}_{L,g}(x) + \epsilon\, a^{(1)}_{L,g}(x) + \epsilon^2 b^{(1)}_{L,g}(x)$$

- Kinematic variables

$$x = \frac{2q\cdot k_0}{q^2} \qquad q^2 = s > 0 \qquad 0 < x \le 1$$

Fragmentation Functions

- Fragmentation functions

$$\mathcal{F}_T(x,\epsilon) = \frac{2}{2-m}\left(\frac{q\cdot k_0}{q^2}W_\mu{}^\mu + \frac{k_0^\mu k_0^\nu}{q\cdot k_0}W_{\mu\nu}\right)$$

$$\mathcal{F}_L(x,\epsilon) = \frac{k_0^\mu k_0^\nu}{q\cdot k_0}W_{\mu\nu}$$

$$\mathcal{F}_A(x,\epsilon) = i\,\frac{3}{(m-2)(m-3)}\frac{q^\alpha k_0^\beta}{q^2}\,\epsilon_{\mu\nu\alpha\beta}W_{\mu\nu}$$

- Hadronic tensor

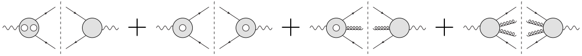
$$W_{\mu\nu}(x,\epsilon) = \frac{x^{m-3}}{4\pi}\int \mathrm{dPS(n)}\; M_\mu(n)\; M_\nu(n)$$

  - $m = 4 - 2\epsilon$
  - $\mathrm{dPS(n)}$ — n-particle final-state phase-space
  - $M^\mu(n)$ — amplitudes for the process $\gamma^*, \phi^*(q^\mu) \to \langle n \text{ partons}\rangle$

NLO contributions to P^T_{gq}

- Transverse Fragmentation Function

$\mathcal{F}^{(2)}_T =$



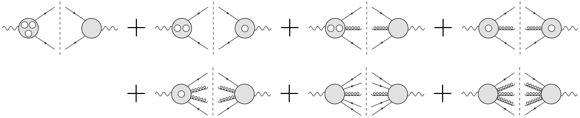
- Mass Factorization Vermaseren, Vogt, Moch '05

$$\mathcal{F}^{(2)}_T = \frac{1}{\epsilon^2}\left\{\frac{1}{2}P^{(0)}_{gi}P^{(0)}_{iq} + \frac{1}{2}\beta_0 P^{(0)}_{gq}\right\} - \frac{1}{\epsilon}\left\{\frac{1}{2}P^{(1)}_{gq} + P^{(0)}_{gi}c^{(1)}_i\right\}$$
$$+ \left\{c^{(2)}_g - P^{(0)}_{gi}a^{(1)}_i\right\} + \epsilon\left\{a^{(2)}_g - P^{(0)}_{gi}b^{(1)}_i\right\}$$

NNLO contributions to P_{gq}^T

- Transverse Fragmentation Function

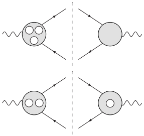
$\mathcal{F}_T^{(3)} =$



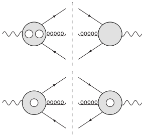
- Mass Factorization Vermaseren, Vogt, Moch '05

$$
\begin{aligned}
\mathcal{F}_T^{(3)} \quad = \quad & -\frac{1}{\epsilon^3}\left\{\frac{1}{6}P_{gi}^{(0)}P_{ij}^{(0)}P_{jq}^{(0)} + \frac{1}{2}\beta_0 P_{gi}^{(0)}P_{iq}^{(0)} + \frac{1}{3}\beta_0^2 P_{gq}^{(0)}\right\} \\
+ \quad & \frac{1}{\epsilon^2}\left\{\frac{1}{6}P_{gi}^{(0)}P_{iq}^{(1)} + \frac{1}{3}P_{gi}^{(1)}P_{iq}^{(0)} + \frac{1}{3}\beta_1 P_{gq}^{(0)} + \frac{1}{2}P_{gi}^{(0)}P_{ij}^{(0)}c_j^{(1)} + \beta_0\left(\frac{1}{3}P_{gq}^{(1)} + \frac{1}{2}P_{gi}^{(0)}c_i^{(1)}\right)\right\} \\
- \quad & \frac{1}{\epsilon}\left\{\frac{1}{3}P_{gq}^{(2)} + \frac{1}{2}P_{gi}^{(1)}c_i^{(1)} + P_{gi}^{(0)}c_i^{(2)} - \frac{1}{2}P_{gi}^{(0)}P_{ij}^{(0)}a_j^{(1)} - \frac{1}{2}\beta_0 P_{gi}^{(0)}a_i^{(1)}\right\} \\
+ \quad & \left\{c_g^{(3)} - P_{gi}^{(0)}a_i^{(2)} - \frac{1}{2}P_{gi}^{(1)}a_i^{(1)} + \frac{1}{2}P_{gi}^{(0)}P_{ij}^{(0)}b_j^{(1)} + \frac{1}{2}\beta_0 P_{gi}^{(0)}b_i^{(1)}\right\}
\end{aligned}
$$

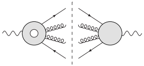
NNLO contributions to P^T_{gq}



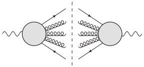
Pure-virtual contributions contain overall $\delta(1-x)$ factor.
We do not consider such contributions.



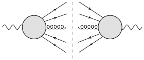
Can be extracted from Garland, Gehrmann, Glover, Koukoutsakis, Remiddi '01
Calculated by Duhr, Gehrman, Jaquier 1411.3587 [hep-ph]



One-loop helicity amplitudes by Bern, Dixon, Kosower '97
Final-state integration is of NLO complexity — simple.



Contribution is known from analytical continuation by Almasy, Moch, Vogt '11

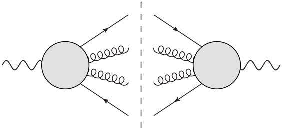


Unknown!

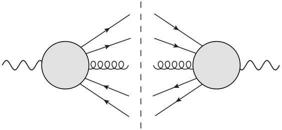
Final-state integration

The main challenge of the calculation is n-particle final-state integration:

$$\int \mathrm{dPS}(n) = \int \prod_{i=0}^{n-1} \mathrm{d}^m k_i \delta^+(k_i^2)\ \delta\left(x - \frac{2q\cdot k_0}{q^2}\right)\ \delta(q - \sum_{j=0}^{n-1} k_j)$$



$\sim$ NLO $(n = 4)$

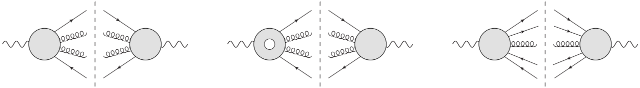


$\sim$ NNLO $(n = 5)$

We attack such integrals with IBP identities and differential equations

Do not miss Methods Sessions on Thursday and Friday afternoons

Integration-by-Parts Identities



QGRAF	• 8 x 8 diagrams	• 210 x 8 diagrams	• 48 x 48 diagrams
FORM • trace of gamma matrices • index contraction • color traces • partial fractioning			
Mathematica • analyze symmetries • split by topologies	• 495 integrals	• 38 357 integrals	• 55 614 integrals
LiteRed • find IBP reduction rules • find masters	• ∼10 h • 8 masters	• ∼100 h • 83 masters	• >1000 h • ∼80 masters

We consider to switch to Reduze2 for NNLO calculations since LiteRed can not solve some sectors.

Differential Equations for Masters

- general representation

$$\frac{\partial f_i}{\partial x} = \sum_{j=1}^{n} a_{ij}(x,\epsilon)\, f_j(x,\epsilon)$$

- can be easily solved (matrix multiplication and HPL integration)
  - as ϵ-series when $a_{ij}(x,\epsilon) = 0$ for $\epsilon \to 0$
  - in HPLs when alphabet is $\{x, 1-x, 1+x\}$
- particular example: $a_{ij}(x,\epsilon)$ at NLO for $\gamma^* \to q\bar{q}gg$

$$\begin{pmatrix}
\frac{(2\epsilon-1)(2x-1)}{(1-x)x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{3\epsilon-2}{(1-x)x} & -\frac{3\epsilon-1}{x} & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{(2\epsilon-1)}{\epsilon^2(1-x)x(x+1)} & 0 & -\frac{6\epsilon-1}{x+1} & \frac{2}{x+1} & 0 & 0 & 0 & 0 \\
\frac{(2\epsilon-1)}{\epsilon x^2(x+1)} & -\frac{(2\epsilon-1)(3\epsilon-1)}{x^2} & \frac{2(6\epsilon-1)}{x(x+1)} & \frac{2\epsilon\left(x^2+3x-2\right)}{(1-x)x(x+1)} & 0 & 0 & 0 & 0 \\
\frac{2\left(x^2+4x+1\right)}{\epsilon^2(1-x)x^3(x+1)^3} & \frac{2(2\epsilon-1)(x-1)}{\epsilon x^3(x+1)^2} & \frac{2(6\epsilon-1)(x-1)}{x^2(x+1)^3} & \frac{4\left(x^2+1\right)}{x^2(x+1)^3} & -\frac{(2\epsilon+1)(2x+1)}{x(x+1)} & 0 & 0 & 0 \\
0 & -\frac{(2\epsilon-1)}{\epsilon(1-x)x} & 0 & 0 & 0 & \frac{2\epsilon}{1-x} & 0 & 0 \\
-\frac{4}{\epsilon^2(1-x)^3x^3(x+1)} & -\frac{2(2\epsilon-1)(x-2)}{\epsilon(1-x)^2x^3} & -\frac{2(6\epsilon-1)}{(1-x)x^2(x+1)} & \frac{4\left(x^2+1\right)}{(1-x)^2x^2(x+1)} & 0 & -\frac{4\epsilon}{(1-x)^2x} & \frac{(2\epsilon+1)(2x-1)}{(1-x)x} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{(2\epsilon+1)(2x-1)}{(1-x)x}
\end{pmatrix}$$

Master Algorithm

- make system regular as $\epsilon \to 0$
  - Algorithm I (coming soon)
- make system zero-diagonal
  - Algorithm II
- make system zero-triangular
  - Algorithm III
- solve system (see Henn '13)
- find boundary conditions

$$\begin{pmatrix}
\frac{(2\epsilon-1)(2x-1)}{(1-x)x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{3\epsilon-2}{(1-x)x} & -\frac{3\epsilon-1}{x} & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{(2\epsilon-1)}{\epsilon^2(1-x)x(x+1)} & 0 & -\frac{6\epsilon-1}{x+1} & \frac{2}{x+1} & 0 & 0 & 0 & 0 \\
\frac{(2\epsilon-1)}{\epsilon x^2(x+1)} & -\frac{(2\epsilon-1)(3\epsilon-1)}{x^2} & \frac{2(6\epsilon-1)}{x(x+1)} & \frac{2\epsilon\left(x^2+3x-2\right)}{(1-x)x(x+1)} & 0 & 0 & 0 & 0 \\
\frac{2\left(x^2+4x+1\right)}{\epsilon^2(1-x)x^3(x+1)^3} & \frac{2(2\epsilon-1)(x-1)}{\epsilon x^3(x+1)^2} & \frac{2(6\epsilon-1)(x-1)}{x^2(x+1)^3} & \frac{4\left(x^2+1\right)}{x^2(x+1)^3} & -\frac{(2\epsilon+1)(2x+1)}{x(x+1)} & 0 & 0 & 0 \\
0 & -\frac{(2\epsilon-1)}{\epsilon(1-x)x} & 0 & 0 & 0 & \frac{2\epsilon}{1-x} & 0 & 0 \\
-\frac{4}{\epsilon^2(1-x)^3x^3(x+1)} & -\frac{2(2\epsilon-1)(x-2)}{\epsilon(1-x)^2x^3} & -\frac{2(6\epsilon-1)}{(1-x)x^2(x+1)} & \frac{4\left(x^2+1\right)}{(1-x)^2x^2(x+1)} & 0 & -\frac{4\epsilon}{(1-x)^2x} & \frac{(2\epsilon+1)(2x-1)}{(1-x)x} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{(2\epsilon+1)(2x-1)}{(1-x)x}
\end{pmatrix}$$

Algorithm I: finite basis

- algorithm's output

$$\begin{pmatrix}
\frac{(2\epsilon-1)(2x-1)}{(1-x)x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{3\epsilon-2}{(1-x)x} & \frac{1-3\epsilon}{x} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{(2\epsilon-1)}{(x-1)x(x+1)} & 0 & \frac{1-6\epsilon}{1+x} & \frac{2\epsilon}{x+1} & 0 & 0 & 0 & 0 \\
\frac{(2\epsilon-1)}{x^2(x+1)} & -\frac{\epsilon}{x^2} & \frac{2(6\epsilon-1)}{x(x+1)} & -\frac{2\epsilon\left(x^2+3x-2\right)}{(x-1)x(x+1)} & 0 & 0 & 0 & 0 \\
\frac{2\epsilon\left(x^2+4x+1\right)}{(1-x)x^3(x+1)^3} & \frac{2\epsilon^2(x-1)}{x^3(x+1)^2} & \frac{2\epsilon(x-1)}{x^2(x+1)^3} & \frac{4\epsilon^2\left(x^2+1\right)}{x^2(x+1)^3} & -\frac{(2\epsilon+1)(2x+1)}{x(x+1)} & 0 & 0 & 0 \\
0 & \frac{\epsilon^2}{(x-1)x} & 0 & 0 & 0 & -\frac{2\epsilon}{x-1} & 0 & 0 \\
\frac{4\epsilon}{(x-1)^3x^3(x+1)} & -\frac{2\epsilon^2(x-2)}{(x-1)^2x^3} & \frac{2\epsilon}{(x-1)x^2(x+1)} & \frac{4\epsilon^2\left(x^2+1\right)}{(x-1)^2x^2(x+1)} & 0 & -\frac{4\epsilon}{x(1-x)^2} & \frac{(2\epsilon+1)(2x-1)}{(1-x)x} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{(2\epsilon+1)(2x-1)}{(1-x)x}
\end{pmatrix}$$

- algorithm's output as $\epsilon \to 0$

$$\begin{pmatrix}
\frac{2x-1}{(x-1)x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{2}{(x-1)x} & \frac{1}{x} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{2}{(1-x)x(x+1)} & 0 & \frac{1}{x+1} & 0 & 0 & 0 & 0 & 0 \\
-\frac{2}{x^2(x+1)} & 0 & -\frac{2}{x(x+1)} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{2x+1}{x(x+1)} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{2x-1}{(1-x)x} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2x-1}{(1-x)x}
\end{pmatrix}$$

- new basis g_i such that

$$g_i(x,\epsilon) = \epsilon^{n_i} f_i(x,\epsilon)$$

- where n_i are integers
- automated soon

Algorithm II: zero-diagonal basis

algorithm's input ($\epsilon \to 0$ limit)

$$\begin{pmatrix}
\frac{2x-1}{(x-1)x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{2}{(x-1)x} & \frac{1}{x} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{2}{(1-x)x(x+1)} & 0 & \frac{1}{x+1} & 0 & 0 & 0 & 0 & 0 \\
-\frac{2}{x^2(x+1)} & 0 & -\frac{2}{x(x+1)} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{2x+1}{x(x+1)} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{2x-1}{(1-x)x} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2x-1}{(1-x)x}
\end{pmatrix}$$

algorithm's output ($\epsilon \to 0$ limit)

$$\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{2}{(x+1)^2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{2(x-1)}{x(x+1)} & -\frac{1}{x} & -\frac{2}{x} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

- new basis g_i such that

$$g_i(x,\epsilon) = b_i(x,\epsilon) f_i(x,\epsilon) \quad \text{where} \quad b_i(x,\epsilon) = \exp\left(-\int \mathrm{d}x\, a_{ii}(x,\epsilon)\right)$$

  - $b_1(x,\epsilon) = \frac{1}{x(1-x)} \quad b_2(x,\epsilon) = \frac{1}{x} \quad b_3(x,\epsilon) = \frac{1}{1+x} \quad \ldots$

Algorithm III: zero-triangular basis

algorithm's input ($\epsilon \to 0$ limit)

$$\begin{pmatrix} 0 & 0 & 0 & 0\,0\,0\,0\,0 \\ 0 & 0 & 0 & 0\,0\,0\,0\,0 \\ \frac{2}{(x+1)^2} & 0 & 0 & 0\,0\,0\,0\,0 \\ \frac{2(x-1)}{x(x+1)} & -\frac{1}{x} & -\frac{2}{x} & 0\,0\,0\,0\,0 \\ 0 & 0 & 0 & 0\,0\,0\,0\,0 \\ 0 & 0 & 0 & 0\,0\,0\,0\,0 \\ 0 & 0 & 0 & 0\,0\,0\,0\,0 \\ 0 & 0 & 0 & 0\,0\,0\,0\,0 \end{pmatrix}$$

algorithm's output ($\epsilon \to 0$ limit)

$$\begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0 \end{pmatrix}$$

- new basis h_i such that

$$h_i(x,\epsilon) = \sum_{j=1}^{i-1} c_{ij}(x,\epsilon)\, g_j(x,\epsilon) + g_i(x,\epsilon)$$

where

$$c_{ij}(x,\epsilon) = -\int \mathrm{d}x \left(\sum_{j+1\leq k\leq i-1} c_{ik}(x,\epsilon)\, a_{kj}(x,\epsilon) + a_{ij}(x,\epsilon) \right)$$

- $h_3(x,\epsilon) = \frac{2}{1+x} g_1(x,\epsilon) + g_3(x,\epsilon) \quad \ldots$

Boundary conditions

- simple master integral (green functions are known)

$$f_i(x,\epsilon) = C_i^{(0)} f_i^{(0)}(x) + \epsilon\, C_i^{(1)} f_i^{(1)}(x) + \ldots$$

- corresponding inclusive master

$$F_i(\epsilon) = \int_0^1 \mathrm{d}x\; f_i(x,\epsilon) = A_i^{(0)} + \epsilon\, A_i^{(1)} + \ldots$$

- boundary conditions are found as

$$C_i^{(j)} = \frac{A_i^{(j)}}{\int_0^1 \mathrm{d}x\; f_i^{(j)}(x)}$$

- coefficients $A_i^{(j)}$ are non-trivial to calculate at NNLO

To be continued...

Summary

Done:

- **Fragmentation Functions generator**
  (`QGRAF` + `FORM` + `LiteRed` + `Mathematica`)
  → input: process up to NNLO, e.g. $\gamma^* \to q\bar{q}q\bar{q}g$
  → output: fragmentation function in terms of masters
- **Algorithm to find masters** (`Mathematica`)
  → integral basis choice
  → solutions for masters
  → boundary conditions finder (manual)

In progress:

- IBP rules with `Reduze`
- Automated boundary conditions finder
- Inclusive integrals at NNLO