Integrand-based methods at one loop

Andreas van Hameren

Institute of Nuclear Physics
Polish Academy of Sciences
Kraków

presented at

Quarkonia as tools 2023

10-01-2023

This research was supported by grant No. 2019/35/B/ST2/03531 of the Polish National Science Centre.

Tree-level matrix elements without expressions

- At the end of the 90s we realized we need to be able to calculate cross sections for the multi-jet events that would be abundant at LHC.

- The textbook approach to get an *expression* for $|\mathcal{M}|^2$ as function of just external momenta $p_2, p_2, \dots, p_n$ seems hopeless.

N	ϕ^3	$\phi^3 + \phi^4$
4	3	4
5	15	25
6	105	220
7	945	2,485
8	10,395	34,300
12	654,729,075	5,348,843,500
13	13,749,310,575	140,880,765,025
14	316,234,143,225	4,063,875,715,900

number of graphs

Exploding QCD, Draggiotis 2002

Do we actually need complete expressions for matrix elements?

- There is no other option than the Monte Carlo method to calculate cross sections:
  - we are facing arbitrary phase space cuts
  - we want to produce (plots of) differential cross sections of arbitrary observables
- we only need the numerical value (single real number) of $|\mathcal{M}|^2$ provided numerical values of $p_1, p_2, \dots, p_n$ for each event.

Tree-level matrix elements without expressions

The fact that we only need numerical values of $|\mathcal{M}|^2$ creates some opportunities

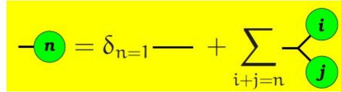
- use spin/helicity amplitudes

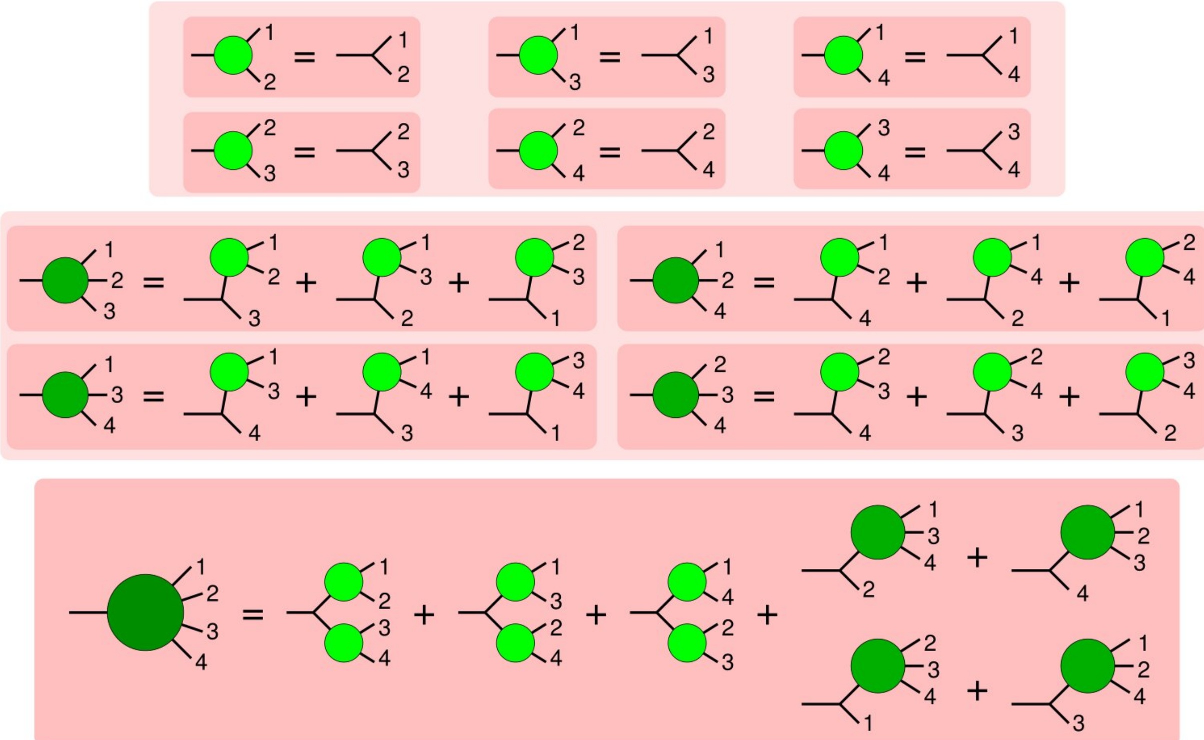
$$|\mathcal{M}|^2\left(\{p_i\}_{i=1}^n\right) = \sum_{\{\lambda\}} \left|\mathcal{M}\left(\{p_i,\lambda_i\}_{i=1}^n\right)\right|^2$$

  and evaluate $\mathcal{M}\left(\{p_i,\lambda_i\}_{i=1}^n\right)$ numerically (roughly 2^n complex numbers). So only need to be able to calculate (spin/helicity) *amplitudes* instead of matrix elements. The sum can be made part of the Monte Carlo procedure, have polarized events.

- given external momenta p_i (real numbers), calculate
  - external spinors/polarization vectors u_i or ϵ_i (complex numbers)
  - currents $\bar{u}_i\gamma^\mu u_j$ (complex numbers)
  - propagators *eg.* $\gamma_\mu(p_i^\mu + p_j^\mu)/(p_i+p_j)^2$ (complex numbers)
  - whole graph (complex number)
- Of course, this need to be automated into a computer program.
- Avoid organization in terms of graphs.

Recursive computation





Caravaglios, Moretti 1995

How to proceed to one loop?

We know a one-loop amplitude involving a set of ℓoop-momentum-dependent denominators $\mathcal{D}_i(\ell) = (\ell + K_i)^2 - m_i^2 + i\eta$, with some enumeration $i = 0, 1, \ldots$, can be decomposed as

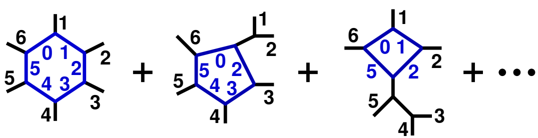
$$\mathcal{A}^{\text{loop}} = \sum_{i,j,k,l} C_{ijkl}\,\text{Box}_{ijkl} + \sum_{i,j,k} C_{ijk}\,\text{Tri}_{ijk} + \sum_{i,j} C_{ij}\,\text{Bub}_{ij} + \sum_{i} C_i\,\text{Tad}_i + \mathcal{R} + \mathcal{O}(\varepsilon)\ ,$$

where the *master integrals* are defined as

$$\text{Box}_{ijkl} = \int \frac{d^{4-2\varepsilon}\ell}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)\mathcal{D}_l(\ell)}\ , \qquad \text{Tri}_{ijk} = \int \frac{d^{4-2\varepsilon}\ell}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)}\ ,$$

$$\text{Bub}_{ij} = \int \frac{d^{4-2\varepsilon}\ell}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)}\ , \qquad \text{Tad}_i = \int \frac{d^{4-2\varepsilon}\ell}{\mathcal{D}_i(\ell)}\ .$$

In practice, we should imagine that $\mathcal{A}^{\text{loop}}$ is an *ordered* subamplitude, and we can write it with a common denominator as



$$= \quad \mathcal{A}^{\text{loop}} = \int d^{4-2\varepsilon}\ell\, \frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)}$$

How to find the coefficients $C_{ijkl}, C_{ijk}, C_{ij}, C_i$ and $\mathcal{R}$ as complex numbers in a strategy compatible with the strategy at tree-level?

Integrand-based approach

Does the one-loop decomposition into scalar integrals hold at the integrand level in 4-dim?

$$\frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)} \stackrel{?}{=} \sum_{i,j,k,l} \frac{C_{ijkl}}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)\mathcal{D}_l(\ell)} + \sum_{i,j,k} \frac{C_{ijk}}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)} + \sum_{i,j} \frac{C_{ij}}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)} + \sum_i \frac{C_i}{\mathcal{D}_i(\ell)}$$

Integrand-based approach

Ossola, Papadopoulos, Pittau 2007
Ellis, Giele, Kunszt 2008

At the integrand-level in 4-dim, terms appear that integrate to zero

$$\frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)} = \sum_{i,j,k,l} \frac{C_{ijkl} + \tilde{\mathcal{C}}_{ijkl}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)\mathcal{D}_l(\ell)} + \sum_{i,j,k} \frac{C_{ijk} + \tilde{\mathcal{C}}_{ijk}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)} + \sum_{i,j} \frac{C_{ij} + \tilde{\mathcal{C}}_{ij}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)} + \sum_{i} \frac{C_{i} + \tilde{\mathcal{C}}_{i}(\ell)}{\mathcal{D}_i(\ell)}$$

The coefficients C are the same as at integral-level, and the polynomials $\tilde{\mathcal{C}}$ have finite rank

Integrand-based approach

Ossola, Papadopoulos, Pittau 2007
Ellis, Giele, Kunszt 2008

At the integrand-level in 4-dim, terms appear that integrate to zero

$$\frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)} = \sum_{i,j,k,l} \frac{C_{ijkl} + \tilde{\mathcal{C}}_{ijkl}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)\mathcal{D}_l(\ell)} + \sum_{i,j,k} \frac{C_{ijk} + \tilde{\mathcal{C}}_{ijk}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)} + \sum_{i,j} \frac{C_{ij} + \tilde{\mathcal{C}}_{ij}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)} + \sum_{i} \frac{C_{i} + \tilde{\mathcal{C}}_{i}(\ell)}{\mathcal{D}_i(\ell)}$$

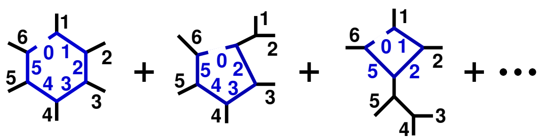
The coefficients C are the same as at integral-level, and the polynomials $\tilde{\mathcal{C}}$ have finite rank

$\Longrightarrow$ problem solved! (in principle)

- choose as many values of ℓ as there are coefficients (including the ones inside the polynomials $\tilde{\mathcal{C}}$)

- left-hand side can be evaluated for each value of ℓ using tree-level techniques

$$\frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)} =$$



- invert linear system numerically

Integrand-based approach

Ossola, Papadopoulos, Pittau 2007
Ellis, Giele, Kunszt 2008

At the integrand-level in 4-dim, terms appear that integrate to zero

$$\frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)} = \sum_{i,j,k,l} \frac{C_{ijkl} + \tilde{\mathcal{C}}_{ijkl}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)\mathcal{D}_l(\ell)} + \sum_{i,j,k} \frac{C_{ijk} + \tilde{\mathcal{C}}_{ijk}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)} + \sum_{i,j} \frac{C_{ij} + \tilde{\mathcal{C}}_{ij}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)} + \sum_{i} \frac{C_{i} + \tilde{\mathcal{C}}_{i}(\ell)}{\mathcal{D}_i(\ell)}$$

The coefficients C are the same as at integral-level, and the polynomials $\tilde{\mathcal{C}}$ have finite rank

Choose values of ℓ such that denominators $\mathcal{D}_i(\ell)$ vanish:

- in 4-dim, at most 4 can vanish simultaneously, with 2 solutions $\ell_{1,2}$. So the right-hand side only involves the term with those 4 denominators (say i, j, k, l) (multiply left and right with those) $\Longrightarrow$ equation with only 2 unknown coefficents $C_{ijkl}, \tilde{C}_{ijkl}$

- find all box-coefficients for all combinations of i, j, k, l, and proceed with triangle coefficients $C_{ijk}, \tilde{C}^{(\rho)}_{ijk}$, $\rho = 1, \ldots, 6$ using

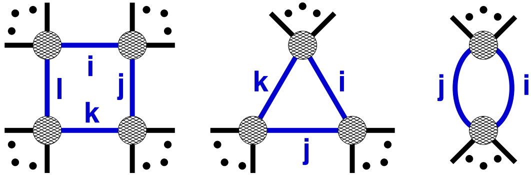
$$\frac{\mathcal{N}(\ell)}{\prod_i \mathcal{D}_i(\ell)} - \sum_{i,j,k,l} \frac{C_{ijkl} + \tilde{\mathcal{C}}_{ijkl}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)\mathcal{D}_l(\ell)} = \sum_{i,j,k} \frac{C_{ijk} + \tilde{\mathcal{C}}_{ijk}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\mathcal{D}_k(\ell)} + \sum_{i,j} \frac{C_{ij} + \tilde{\mathcal{C}}_{ij}(\ell)}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)} + \sum_{i} \frac{C_{i} + \tilde{\mathcal{C}}_{i}(\ell)}{\mathcal{D}_i(\ell)}$$

- etc.

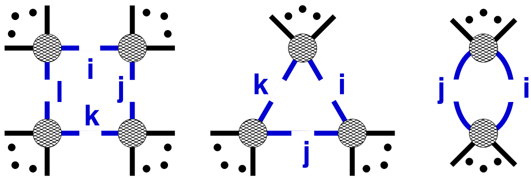
Integrand cuts

What happens to the left-hand side, the one-loop integrand, when denominators vanish?

The set of all one-loop graphs that must involve denominators i, j, k, l, (or denominators i, j, k, or denominators i, j), look like



Choosing integration momenta ℓ at the amplitude level such that those denominators vanish puts the lines on-shell. The blobs become tree-level amplitudes with 2 extra on-shell lines.



The left-hand side of the equation can be constructed in each step by sewing together *on-shell* tree-level amplitudes.

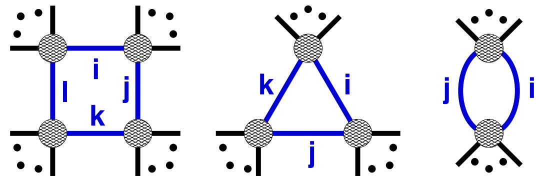
A loop momentum choice ℓ that puts a number of denominators to zero is called a *cut*, and is a solution to the *cut equations*.

The terms of the one-loop amplitude proportional to the coefficients C is called the *cut-constructable* part.

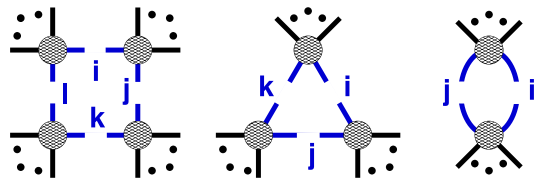
Integrand cuts

What happens to the left-hand side, the one-loop integrand, when denominators vanish?

The set of all one-loop graphs that must involve denominators i, j, k, l, (or denominators i, j, k, or denominators i, j), look like



Choosing integration momenta ℓ at the amplitude level such that those denominators vanish puts the lines on-shell. The blobs become tree-level amplitudes with 2 extra on-shell lines.



The left-hand side of the equation can be constructed in each step by sewing together *on-shell* tree-level amplitudes.

A loop momentum choice ℓ that puts a number of denominators to zero is called a *cut*, and is a solution to the *cut equations*.

$$\mathcal{A}^{\text{loop}} = \boxed{\sum_{i,j,k,l} C_{ijkl}\,\text{Box}_{ijkl} + \sum_{i,j,k} C_{ijk}\,\text{Tri}_{ijk} + \sum_{i,j} C_{ij}\,\text{Bub}_{ij} + \sum_{i} C_{i}\,\text{Tad}_{i}} + \mathcal{R} + \mathcal{O}(\varepsilon)\ ,$$

Rational terms

$$\mathcal{A}^{\rm loop} = \sum_{i,j,k,l} C_{ijkl}\,{\rm Box}_{ijkl} + \sum_{i,j,k} C_{ijk}\,{\rm Tri}_{ijk} + \sum_{i,j} C_{ij}\,{\rm Bub}_{ij} + \sum_{i} C_i\,{\rm Tad}_i + \boxed{\mathcal{R}} + \mathcal{O}(\varepsilon)\ ,$$

They do not contain (di)logarithms of invariants, and appear due to the regularization of divergencies, in dimensional regularization from integrals of the type

$$\int d^{4-2\varepsilon}\ell\,\frac{(\ell_{-2\varepsilon}^2)^n}{\mathcal{D}_i(\ell)\mathcal{D}_j(\ell)\cdots}$$

The terms can be separated in two groups: Ossola, Papadopoulos, Pittau 2008

$\mathcal{R}_1$ appears due to the mismatch between 4-dim and $(4-2\varepsilon)$-dim denominators. It can be calculated by interpreting the extra-dimensional components as extra mass

$$\mathcal{D}(\ell_{4-2\varepsilon}) = (\ell_4 + K_j)^2 - m_j^2 - \ell_{-2\varepsilon}^2$$

$\mathcal{R}_2$ appears due to treating numerators strictly 4-dimensional. These terms follow the structure of counter-terms in renormalization, and can be calculated as tree-level amplitudes with a single special vertex (Garzelli, Malamos, Pittau 2010,2011).

$\mathcal{R}$ can also be obtained by including 5-point integrals in the basis, and performing the cut-construction in *explicit* higher dimensions. Giele, Kunszt, Melnikov 2008

Tensor coefficient recursion

AvH 2009

OpenLoops Cascioli, Maierhofer Pozzorini, 2012

Recola Actis, Denner, Hofer Scharf, Uccirati 2013

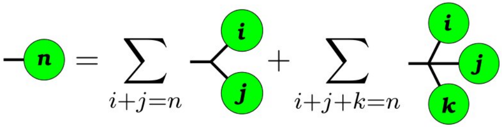
Tensor coefficient recursion

AvH 2009

OpenLoops Cascioli, Maierhofer Pozzorini, 2012

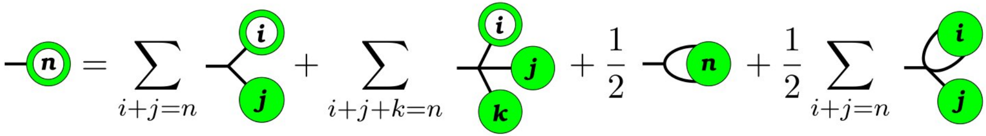
Recola Actis, Denner, Hofer Scharf, Uccirati 2013

Tree-level recursion



A blob with label j has j on-shell external lines

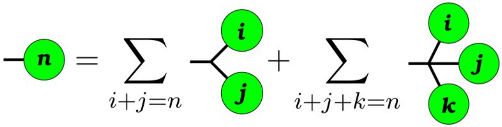
One-loop recursion



Tensor coefficient recursion

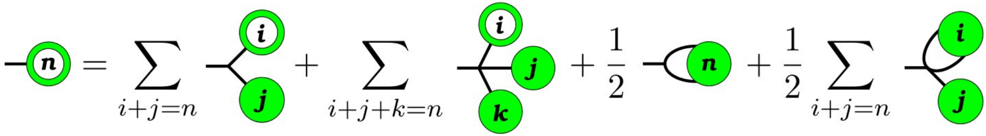
AvH 2009 | OpenLoops (Cascioli, Maierhofer, Pozzorini, 2012) | Recola (Actis, Denner, Hofer, Scharf, Uccirati 2013)

Tree-level recursion



A blob with label j has j on-shell external lines

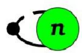
One-loop recursion



How to deal with the explicit loops in the recursion? Assume all tensor integrals with arbitrary denominators and with arbitrary rank are available (eg. via del Aguila, Pittau 2004)

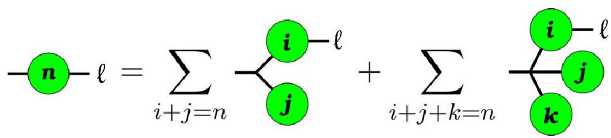
$$\mathcal{T}\big(\{i_j\},\{\nu_j\}\big) = \int d^{4-2\varepsilon}\ell\, \mathcal{J}\big(\ell;\{i_j\},\{\nu_j\}\big) \quad , \quad \mathcal{J}\big(\ell;\{i_j\}_{j=1}^{p},\{\nu_j\}_{j=1}^{r}\big) = \frac{\ell_4^{\nu_1}\ell_4^{\nu_2}\cdots\ell_4^{\nu_r}}{\mathcal{D}_{i_1}(\ell)\mathcal{D}_{i_2}(\ell)\cdots\mathcal{D}_{i_p}(\ell)}$$

The cut of the explicit loop has a decomposition in terms of tensor integrands.



$$= \sum_{\text{sets}\{\mathcal{D}\},\text{sets}\{\nu\}} \mathcal{G}\big(\{\mathcal{D}\},\{\nu\}\big)\,\mathcal{J}\big(\{\mathcal{D}\},\{\nu\}\big)$$

It also follows a tree-level recursion with an extra special let with momentum ℓ.



Using the vertex relation $V^{\mu\nu\rho}(p,q+\ell) = V^{\mu\nu\rho}(p,q) + X^{\mu\nu\rho}_{\sigma}\ell^{\sigma}$ we find recursive relations for the coefficients $\mathcal{G}\big(\{\mathcal{D}\},\{\nu\}\big)$.

Public Software

Several programs apply integrand-based methods to calculate full 1-loop amplitudes or virtual contributions (user chooses process, and provides numerical values of extenal momenta):

- Helac-NLO Comput.Phys.Commun. 184 (2013) 986-997 • e-Print: 1110.1499
- MadLoop JHEP 05 (2011) 044 • e-Print: 1103.0621
- Njet Comput.Phys.Commun. 184 (2013) 1981-1998 • e-Print: 1209.0100
- GoSam Eur.Phys.J.C 74 (2014) 8, 3001 • e-Print: 1404.7096
- Recola Comput.Phys.Commun. 214 (2017) 140-173 • e-Print: 1605.01090
- OpenLoops Eur.Phys.J.C 79 (2019) 10, 866 • e-Print: 1907.13071

Programs that perform the 1-loop decomposition into scalar integrals for "arbitrary" user-provided integrands (user provides numerical values of denominator momenta, masses, and a function that evaluates the numerator as function of the loop-momentum):

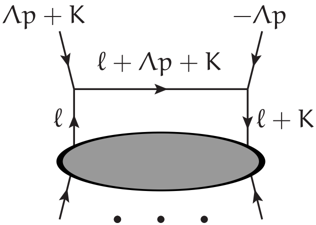
- CutTools JHEP 03 (2008) 042 • e-Print: 0711.3596 (inside Helac-NLO and MadLoop)
- Samurai JHEP 08 (2010) 080 • e-Print: 1006.0710 (the Sam in GoSam)
- Ninja JHEP 03 (2014) 115 • e-Print: 1312.6678

Linear Denominators

Integrand methods need to be constructed in conjunction with the regularization for the linear denominators.

AvH 1710.07609:

$$\frac{1}{p\cdot(\ell+K)} = \lim_{\Lambda\to\infty} \frac{2\Lambda}{(\ell+\Lambda p+K)^2}$$



Works fine (divergence of at most $\ln^2\Lambda$) for box-integrals, most triangle integrals, but

$$\frac{\mu^{2\varepsilon}}{i\pi^{2-\varepsilon} r_\Gamma}\int d^{4-2\varepsilon}\ell\, \frac{2\Lambda}{\ell^2(\ell+\Lambda p+K)^2} = 2\Lambda\left[\frac{1}{\varepsilon} + 2 - \ln\left(\frac{-2\Lambda p\cdot K}{\mu^2}\right) + \mathcal{O}(\varepsilon)\right]$$

This behavior may cancel between all graphs, but then numerators need to be expanded to sub-leading orders in Λ.

eg. quark-line : $\dfrac{\gamma_\mu(\ell^\mu + \Lambda p^\mu + K^\mu)}{(\ell+\Lambda p+K)^2} \longrightarrow \dfrac{\Lambda\gamma_\mu p^\mu}{(\ell+\Lambda p+K)^2}$ can give wrong result