From Numerical To Analytical Amplitudes

Giuseppe De Laurentis

with Daniel Maitre

IPPP - Durham University

0. Table of Content

1.1 Motivation (1)

Cross sections at hadron colliders:

$$\sigma_{2\to n-2} = \sum_{a,b} \int dx_a dx_b \; f_{a/h_1}(x_a, \mu_F) f_{b/h_2}(x_b, \mu_F) \; \hat{\sigma}_{ab\to n-2}(\mu_F, \mu_R)$$

$$d\hat{\sigma}_n = \frac{1}{2\hat{s}} d\Pi_{n-2} \; (2\pi)^4 \delta^4 \left(\sum_{i=1}^{n} p_i \right) \; \overline{|\mathcal{A}(p_i, \mu_F, \mu_R)|^2}$$

Improving the hard scattering prediction (powers of coupling):

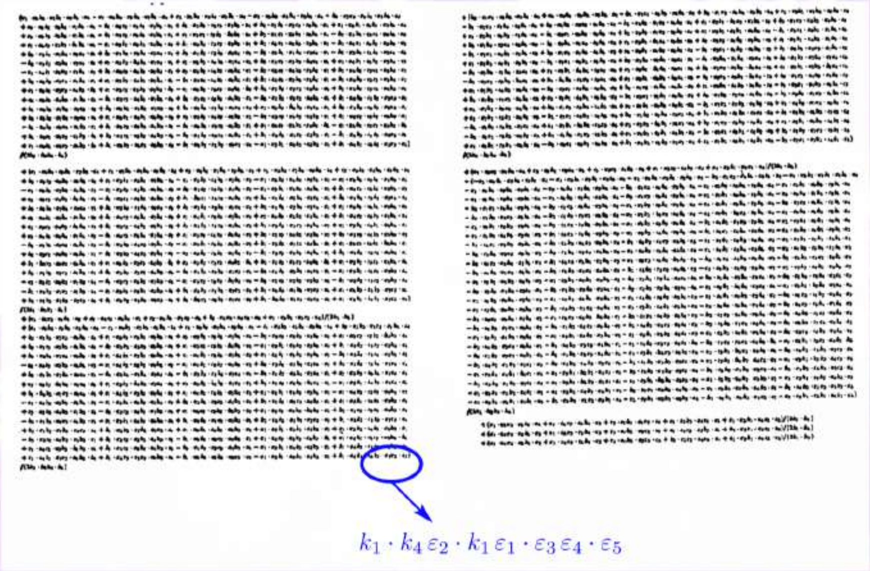
loop\mult	**4**	**5**	**6**	**7**
0	2	3	4	5
1	4	5	6	7
2	6	7	8	9

LO / NLO / NNLO scale variations ~50% / ~10% / ~1%

Infrared divergences cancel only between virtual corrections and real emissions

1.1 Motivation (2)

Brute force calculations are a mess:



Often results are much easier:

$$A^{tree}(1_g^+ 2_g^+ 3_g^+ 4_g^- 5_g^-) = \frac{i \langle 45 \rangle^4}{\langle 12 \rangle \langle 23 \rangle \langle 34 \rangle \langle 45 \rangle \langle 51 \rangle}$$

1.2 Color Ordered Amplitudes

Relation to the full amplitude @ tree level:

$$\mathcal{A}_n^{tree}(p_i,\lambda_i,a_i) = \; g^{n-2} \sum_{\sigma\in S_n/Z_n} \mathrm{Tr}(T^{a_{\sigma(1)}}\dots T^{a_{\sigma(n)}}) A_n^{tree}(\sigma(1^{\lambda_1}),\dots,\sigma(n^{\lambda_n})).$$

Color decomposition at one loop:

$$\mathcal{A}_n^{1-loop}(p_i,\lambda_i,a_i) = \; g^{n} \sum_{\sigma\in S_n/Z_n} N_c \mathrm{Tr}(T^{a_{\sigma(1)}}\dots T^{a_{\sigma(n)}}) A_{n;1}(\sigma(1^{\lambda_1}),\dots,\sigma(n^{\lambda_n}))$$
$$+ \sum_{c=2}^{\lfloor n/2\rfloor+1} \sum_{\sigma\in S_n/Z_{n;c}} \mathrm{Tr}(T^{a_{\sigma(1)}}\dots T^{a_{\sigma(c-1)}})\mathrm{Tr}(T^{a_{\sigma(c)}}\dots T^{a_{\sigma(n)}}) A_{n;c}(\sigma(1^{\lambda_1}),\dots,\sigma(n^{\lambda_n}))$$

Decomposition in terms of basis integrals:

$$A_{n;1}^{1-loop} = \sum_i d_i I^i_{Box} + \sum_i c_i I^i_{Triangle} + \sum_i b_i I^i_{Bubble} + R$$

1.3 Spinor Helicity (1)

The lowest-laying representations of the Lorentz group are:

(j_-, j_+)	dimension	name	quantum field	kinematic variable
(0, 0)	1	scalar	h	m
(0, 1/2)	2	right-handed Weyl spinor	$\chi_{R\,\alpha}$	λ_α
(1/2, 0)	2	left-handed Weyl spinor	$\chi_L^{\dot\alpha}$	$\bar\lambda^{\dot\alpha}$
(1/2, 1/2)	4	rank-two spinor/four vector	$A^\mu / A^{\dot\alpha\alpha}$	$P^\mu / P^{\dot\alpha\alpha}$
$(1/2, 0) \oplus (0, 1/2)$	4	bispinor (Dirac spinor)	Ψ	u, v

Weyl spinors are sufficient to represent the kinematics of massless particles, recall:

$$\det(P^{\dot\alpha\alpha}) = m^2 \to 0 \;\Rightarrow\; P^{\dot\alpha\alpha} = \bar\lambda^{\dot\alpha}\lambda^\alpha, \quad \lambda_\alpha = \begin{pmatrix} \sqrt{p^0+p^3} \\ \frac{p^1+ip^2}{\sqrt{p^0+p^3}} \end{pmatrix} \;\&\; \lambda^\alpha = \epsilon^{\alpha\beta}\lambda_\beta = \begin{pmatrix} \frac{p^1+ip^2}{\sqrt{p^0+p^3}} \\ -\sqrt{p^0+p^3} \end{pmatrix}$$

1.3 Spinor Helicity(2)

Angle and square brackets:

$$\langle ij\rangle = \lambda_i\lambda_j = (\lambda_i)^{\alpha}(\lambda_j)_{\alpha} \qquad [ij] = \bar{\lambda}_i\bar{\lambda}_j = (\bar{\lambda}_i)_{\dot{\alpha}}(\bar{\lambda}_j)^{\dot{\alpha}}$$

Some relations:

$$s_{ij} = \langle ij\rangle[ji]$$

$$\langle i \mid (j+k) \mid l] = (\lambda_i)^{\alpha}(\not{P}_j + \not{P}_k)_{\alpha\dot{\alpha}}\bar{\lambda}_l^{\dot{\alpha}}$$

$$\langle i \mid (j+k) \mid (l+m) \mid n\rangle = (\lambda_i)^{\alpha}(\not{P}_j + \not{P}_k)_{\alpha\dot{\alpha}}(\bar{\not{P}}_l + \bar{\not{P}}_m)^{\dot{\alpha}\alpha}(\lambda_n)_{\alpha}$$

$$tr_5(ijkl) = tr(\gamma^5 \not{P}_i\not{P}_j\not{P}_k\not{P}_l) = [i|j|k|l|i\rangle - \langle i|j|k|l|i]$$

1.3 Spinor Helicity(3)

Examples in python:

```
oInvariants = Invariants(6)
pprint(oInvariants.invs_3[:4])
pprint(oInvariants.invs_s[:8])
```

```
[⟨1|(2+3)|1], ⟨1|(2+6)|1], ⟨1|(3+4)|1], ⟨1|(4+5)|1]]
[s_123, s_124, s_125, s_134, s_135, s_145, s_234, s_235]
```

```
oParticles = Particles(6); oParticles.fix_mom_cons(real_momenta=False)
pprint(gmpTools.to_complex(oParticles.compute("⟨1|2⟩") *
                           oParticles.compute("[2|1]")))
pprint(gmpTools.to_complex(oParticles.compute("s_12")))
```

```
(12.2300736146-8.90029543403e-308j)
(12.2300736146-4.45014771701e-308j)
```

2.1 Singular limits (1)

Singular limits give us information about the poles of the amplitude:

$$\langle ij \rangle \to \varepsilon, \quad f \to \varepsilon^{\alpha} \;\Rightarrow\; log(f) \to \alpha \cdot log(\varepsilon)$$

$\Rightarrow$ The slope of $log(f)(\varepsilon)$ gives us the type of singularity, if any exists.

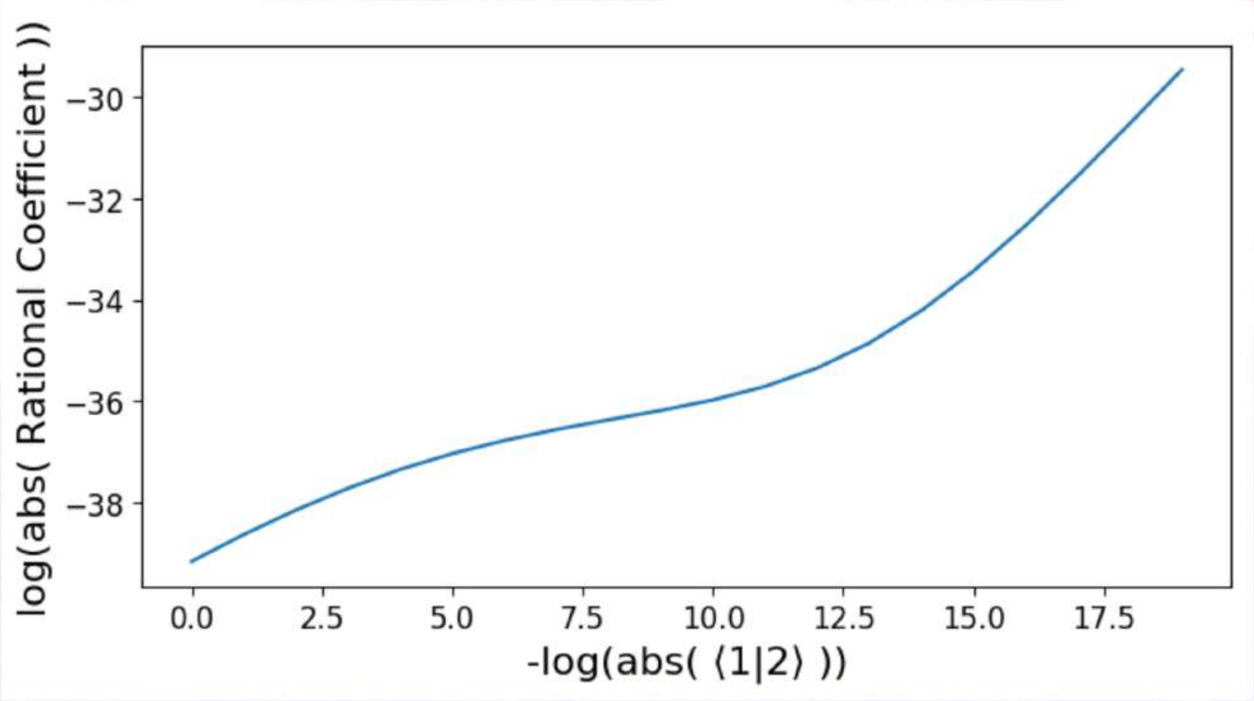
Constructing the phase space ("..." in the output below hide all O(~1) spinor variables):

```
Particles.randomise_all(); oParticles.set("⟨1|2⟩", 10 ** -30)
Particles.phasespace_consistency_check(oInvariants.full, silent=False);
```

```
Consistency check:
The largest momentum violation is 4.91479794421e-308
The largest on shell violation is 7.33937163474e-307
⟨1|2⟩ = 1e-30
...
```

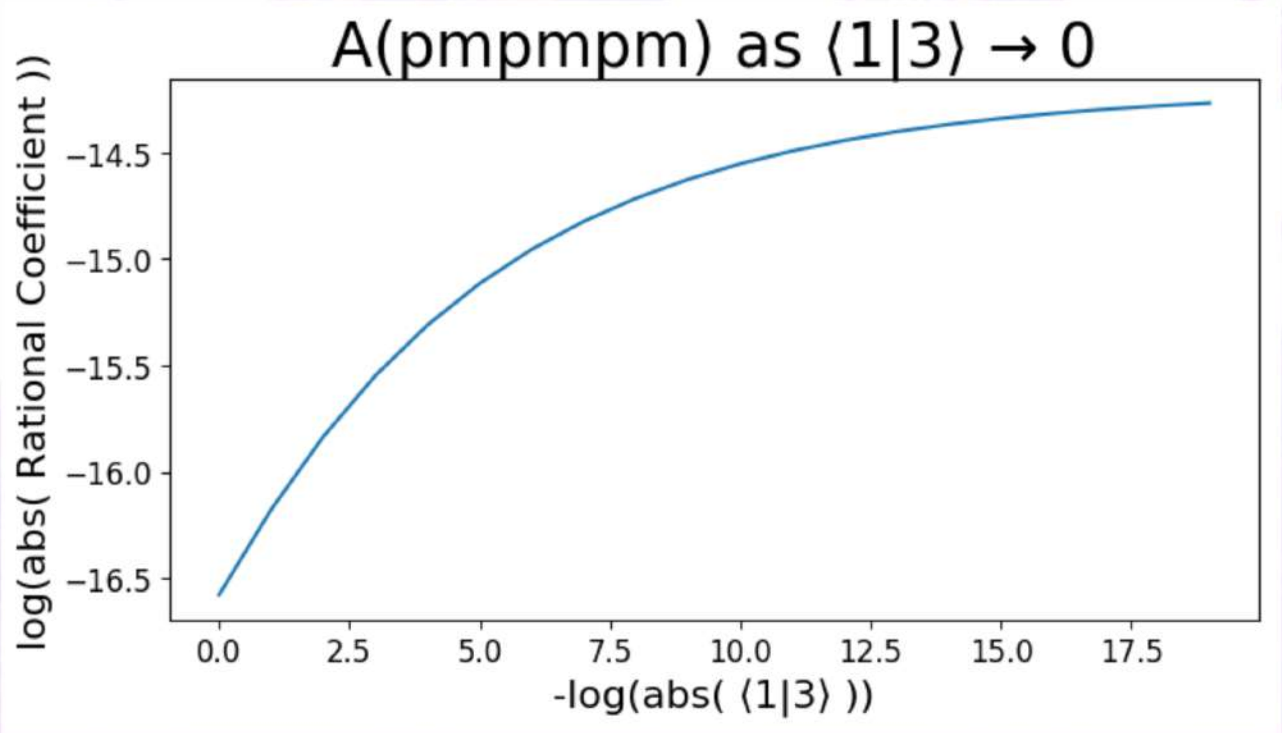
2.1 Singular limits (2)

```
# Simple pole: plot A(pmpmpm) in lim ⟨1|2⟩ → 0 ↔
```



2.1 Singular limits (3)

```
# Not a pole: plot A(pmpmpm) in lim ⟨1|3⟩ → 0↔
```



2.1 Singular limits (4)

Computing this slope for all invariants gives us the full list of poles and their order:

```
oUnknown.do_single_collinear_limits();
```

```
The least common denominator is
/⟨1|2⟩[1|2]⟨1|6⟩[1|6]⟨2|3⟩[2|3]⟨3|4⟩[3|4]⟨4|5⟩[4|5]⟨5|6⟩[5|6]s_
123s_234s_345

Mass dimension & phase weights: -2, [-2, 2, -2, 2, -2, 2]  →  1
6, [-2, 2, -2, 2, -2, 2]
```

The complexity of the numerator ansatz depends on the mass dimension.
A mass dimension of ~16 implies an ansatz of $\mathcal{O}(10^4)$ terms, which is not ideal.
Smaller denominators (i.e. a clearer pole structure) would imply easier numerators.
This can be achieved by studying double collinear limits.

2.2 Doubly singular limits (1)

Constructing the doubly singular limit is similar, but the phase space will be less "clean".

```
Particles.randomise_all()
Particles.set_pair("⟨1|2⟩", 10 ** -30, "⟨2|3⟩", 10 ** -30)
Particles.phasespace_consistency_check(oInvariants.full, silent=False);
```

```
Consistency check:
The largest momentum violation is 2.22507385851e-308
The largest on shell violation is 1.11253692925e-307
⟨3|(1+2)|4] = 1.5432242072e-31
⟨3|(1+2)|(2+4)|3⟩ = 1.62794486639e-31
⟨1|(2+3)|(2+6)|1⟩ = 3.80546997131e-31
⟨1|(2+3)|(3+5)|4⟩ = 4.75196693403e-31
⟨1|3⟩ = 6.23276815214e-31
⟨1|(2+3)|6] = 8.06317867575e-31
⟨1|(2+3)|(3+4)|5⟩ = 8.30471602635e-31
⟨1|2⟩ = 1e-30
⟨2|3⟩ = 1e-30
⟨2|(1+3)|(1+6)|2⟩ = 1.17643508678e-30
⟨3|(1+2)|5] = 1.20019665595e-30
⟨2|(1+3)|6] = 1.58676359058e-30
⟨2|(1+3)|(3+5)|4⟩ = 1.60273825588e-30
```

2.2 Doubly singular limits (2)

Reconstructing the behaviour in the limit involves again the slope of a log plot. We obtain:

```
Showing: scaling in limit / degeneracy of ps / cleaned ps
Unknown.do_double_collinear_limits(silent=True)
Unknown.collinear_data
```

	⟨1|2⟩	[1|2]	⟨1|6⟩	[1|6]	⟨2|3⟩	[2|3]	⟨3|4⟩	[3|4]
⟨1|2⟩	1	1/2/2	1/30/5	1/3/2	1/31/5	1/3/2	1/2/2	2/12/3
[1|2]	1/2/2	1	1/3/2	1/31/5	1/3/2	1/30/5	2/12/3	1/2/2
⟨1|6⟩	1/30/5	1/3/2	1	1/2/2	1/2/2	2/12/3	1/10/2	2/4/2
[1|6]	1/3/2	1/31/5	1/2/2	1	2/12/3	1/2/2	2/4/2	1/10/2
⟨2|3⟩	1/31/5	1/3/2	1/2/2	2/12/3	1	1/2/2	1/30/6	1/3/2
[2|3]	1/3/2	1/30/5	2/12/3	1/2/2	1/2/2	1	1/3/2	1/31/5
⟨3|4⟩	1/2/2	2/12/3	1/10/2	2/4/2	1/30/6	1/3/2	1	1/2/2
[3|4]	2/12/3	1/2/2	2/4/2	1/10/2	1/3/2	1/31/5	1/2/2	1
⟨4|5⟩	1/10/2	2/3/2	1/2/2	2/12/3	1/2/2	2/12/3	1/31/5	1/3/2

2.2 Doubly singular limits (3)

Let's look at a three-mass triangle

```
oTriangle21 = LoadResults(settings.base_res_path + "6g_pmpmpm_G/triangle(21)
```

```
# Slope in lim ⟨3|(1+2)|4], Δ_135 → 0↔
```

```
Consistency check:
The largest momentum violation is 2.01390696191e-306
The largest on shell violation is 8.6948552675e-305
Δ_135 = 1e-60
⟨3|(1+2)|4] = 1e-30
Π_351 = 7.45150146018e-15
Ω_351 = 3.45578125795e-14
...
The slope in this limit is: 6.5. Need square roots?
```

2.2 Doubly singular limits (4)

All branch cuts should have been taken care of by generalised unitarity cuts.

We should be able to explain this behaviour without introducing square roots.

```
# 4 Δ_135 = Π_351 ^ 2 + 4 ⟨3|(1+2)|4] ⟨4|(1+2)|3]↔
```

```
(32.3597139589-1.52020841368e-307j)
(32.3597139589-2.22507385851e-307j)
```

```
# Π_351 = s_123 - s_124
print(gmpTools.to_complex(oParticles.compute("s_123") -
                          oParticles.compute("s_124")))
print(gmpTools.to_complex(oParticles.compute("Π_351")))
```

```
(12.9065205416-0.852511476633j)
(12.9065205416-0.852511476633j)
```

3.1 Partial fraction decompositions (1)

Forbidden pairs	Forced pairs	Optional pairs
$\langle 12\rangle, [12]$: 1.0, 2 → 2	$\langle 12\rangle, [45]$: 2.0, 2 → 2	$\langle 12\rangle, \langle 23\rangle$: 2.0, 30 → 5
$\langle 12\rangle, \langle 34\rangle$: 1.0, 2 → 2		$\langle 12\rangle, [34]$: 2.0, 12 → 3
		$\langle 16\rangle, [45]$: 2.0, 12 → 3

```
# Degeneracy in 'cleaned' phase space
print(oUnknown.true_friends["⟨1|2⟩", "⟨2|3⟩"])
print(oUnknown.true_friends["⟨1|2⟩", "[3|4]"])
print(oUnknown.true_friends["⟨1|6⟩", "[4|5]"])
```

```
[⟨1|2⟩, ⟨2|3⟩, ⟨3|(1+2)|6], ⟨1|(2+3)|4], s_123]
[⟨1|(2+3)|4], ⟨1|2⟩, [3|4]]
[⟨1|6⟩, [4|5], ⟨1|(2+3)|4]]
```

3.1 Partial fraction decompositions (2)

We can now decompose our denominator into smaller pieces. This denominator ansatz can be generated automatically using information from collinear limits or inserted by hand.

```
▸ # Difference in complexity↔
```

```
Least common denominator, w/ ansatz of O(10 000):
/⟨1|2⟩[1|2]⟨1|6⟩[1|6]⟨2|3⟩[2|3]⟨3|4⟩[3|4]⟨4|5⟩[4|5]⟨5|6⟩[5|6]s_
123s_234s_345
[16], [[-2, 2, -2, 2, -2, 2]]

After partial fractioning, w/ ansatz of length 15:
/⟨1|2⟩⟨2|3⟩[4|5][5|6]⟨1|(2+3)|4]⟨3|(1+2)|6]s_123
[8], [[0, 4, 0, 0, -4, 0]]
```

3.2 Fitting of generic ansatze (1)

The most generic ansatz for the given mass dimension and phase weights is built:

```
for entry in Ansatz([8], [[0, 4, 0, 0, -4, 0]])[0]:
    pprint("".join(entry))
```

```
Obtained ansatz from Daniel's spinor solve with lM, lPW: [8],
[[0, 4, 0, 0, -4, 0]]. Size: 15.
⟨1|2⟩⟨1|2⟩⟨1|2⟩⟨1|2⟩[1|5][1|5][1|5][1|5]
⟨1|2⟩⟨1|2⟩⟨1|2⟩⟨2|3⟩[1|5][1|5][1|5][3|5]
⟨1|2⟩⟨1|2⟩⟨1|2⟩⟨2|4⟩[1|5][1|5][1|5][4|5]
⟨1|2⟩⟨1|2⟩⟨2|3⟩⟨2|3⟩[1|5][1|5][3|5][3|5]
⟨1|2⟩⟨1|2⟩⟨2|3⟩⟨2|4⟩[1|5][1|5][3|5][4|5]
⟨1|2⟩⟨1|2⟩⟨2|4⟩⟨2|4⟩[1|5][1|5][4|5][4|5]
⟨1|2⟩⟨2|3⟩⟨2|3⟩⟨2|3⟩[1|5][3|5][3|5][3|5]
⟨1|2⟩⟨2|3⟩⟨2|3⟩⟨2|4⟩[1|5][3|5][3|5][4|5]
⟨1|2⟩⟨2|3⟩⟨2|4⟩⟨2|4⟩[1|5][3|5][4|5][4|5]
⟨1|2⟩⟨2|4⟩⟨2|4⟩⟨2|4⟩[1|5][4|5][4|5][4|5]
⟨2|3⟩⟨2|3⟩⟨2|3⟩⟨2|3⟩[3|5][3|5][3|5][3|5]
⟨2|3⟩⟨2|3⟩⟨2|3⟩⟨2|4⟩[3|5][3|5][3|5][4|5]
⟨2|3⟩⟨2|3⟩⟨2|4⟩⟨2|4⟩[3|5][3|5][4|5][4|5]
⟨2|3⟩⟨2|4⟩⟨2|4⟩⟨2|4⟩[3|5][4|5][4|5][4|5]
⟨2|4⟩⟨2|4⟩⟨2|4⟩⟨2|4⟩[4|5][4|5][4|5][4|5]
```

3.2 Fitting of generic ansatze (2)

The linear system of equations for the coefficients is then solved by numerical inversion.

```
# Chose inversion settings: cpu / gpu↔
```

```
# Fit the coefficients of the ansatz:
oTerms.fit_numerators();
# ... lot of information gets printed ...
```

```
Time elapsed in row reduction:  0.00285291671753 .
Iteration number 1: dropped_redundant: 0, dropped_zero: 10, dro
pped_total: 10.
Coeff. of ⟨1|2⟩⟨1|2⟩⟨1|2⟩⟨1|2⟩[1|5][1|5][1|5][1|5]: 1*I
Coeff. of ⟨1|2⟩⟨1|2⟩⟨1|2⟩⟨2|3⟩[1|5][1|5][1|5][3|5]: -4*I
Coeff. of ⟨1|2⟩⟨1|2⟩⟨2|3⟩⟨2|3⟩[1|5][1|5][3|5][3|5]: 6*I
Coeff. of ⟨1|2⟩⟨2|3⟩⟨2|3⟩⟨2|3⟩[1|5][3|5][3|5][3|5]: -4*I
Coeff. of ⟨2|3⟩⟨2|3⟩⟨2|3⟩⟨2|3⟩[3|5][3|5][3|5][3|5]: 1*I
This piece correctly removes the singularity ([0])

Refining the fit...
The least common denominator is
⟨2|(1+3)|5]⁴/⟨1|2⟩⟨2|3⟩[4|5][5|6]⟨1|(2+3)|4]⟨3|(1+2)|6]s_123
```

3.2 Fitting of generic ansatze (3)

Hence the result for $A(1_g^+ 2_g^- 3_g^+ 4_g^- 5_g^+ 6_g^-)$ tree amplitude:

```
print(oTerms)
```

```
+1I⟨2|(1+3)|5]⁴/⟨1|2⟩⟨2|3⟩[4|5][5|6]⟨1|(2+3)|4]⟨3|(1+2)|6]s_123
(u'165432', False)
(u'216543', True)
```

(165432, False) means: 123456 → 165432

(216543, True) means: 123456 → 216543 + swap all angle and square brackets

A less trivial result follows.

$A_R^{1-loop}(1_g^+ 2_g^- 3_g^+ 4_g^- 5_g^+ 6_g^-)$

```
# Check analytical result (displayed is difference to numerical)↔
```

```
(7.99492510354e-295+4.50739748385e-295j)
```

```
RationalPDF # Showing only first few terms
```

$$\frac{2/3i\langle 12\rangle^3[15]^3[23]s_{123}}{[45]\langle 1|2+3|1]^2\langle 1|2+3|4]\langle 1|2+3|6]\langle 3|1+2|6]}+$$

$$\frac{-2/3i\langle 12\rangle^3[15]^3[23]\langle 3|1+2|5]}{\langle 13\rangle[45][56]\langle 1|2+3|1]^2\langle 1|2+3|4]\langle 3|1+2|6]}+$$

$$\frac{1i\langle 12\rangle^3[15]^2\langle 23\rangle[23]^2[56]}{[45]\langle 1|2+3|1]\langle 1|2+3|4]\langle 1|2+3|6]^2\langle 3|1+2|6]}+$$

$$\frac{\langle 12\rangle^2[15]^2[23](-1i\langle 12\rangle[15]+2i\langle 23\rangle[35])}{[45]\langle 1|2+3|1]\langle 1|2+3|4]\langle 1|2+3|6]\langle 3|1+2|6]}+$$

$$\frac{1i\langle 12\rangle^3[15]^2[25]\langle 3|1+2|5]}{\langle 13\rangle^2[45][56]\langle 1|2+3|1]\langle 1|2+3|4]\langle 3|1+2|6]}+$$

$$\frac{2i\langle 12\rangle^2[15]^2[35]\langle 3|1+2|5]}{\langle 13\rangle[45][56]\langle 1|2+3|1]\langle 1|2+3|4]\langle 3|1+2|6]}+$$

$$\frac{-1i\langle 12\rangle^2[15]\langle 23\rangle[25]^2\langle 2|1+3|5]}{\langle 13\rangle^2[45][56]\langle 1|2+3|4]\langle 3|1+2|6]s_{123}}+$$

Thank you!

Questions?