Calculation of the QED contributions to the electron anomalous magnetic moment on graphics processors

Sergey Volkov

SINP MSU, Dubna branch
DLNP JINR, Dubna

CALC2018, Dubna

AMM of the electron (theory and experiment)

The measured value [2011]:

a_e=0.00115965218073(28)

The most accurate prediction (T. Kinoshita et al. [2018]):

$$a_e = a_e(QED) + a_e(hadronic) + a_e(electroweak),$$

$$a_e(QED) = \sum_{n\geq 1}\left(\frac{\alpha}{\pi}\right)^n a_e^{2n},$$

$$a_e^{2n} = A_1^{(2n)} + A_2^{(2n)}(m_e/m_\mu) + A_2^{(2n)}(m_e/m_\tau) + A_3^{(2n)}(m_e/m_\mu, m_e/m_\tau)$$

a_e=0.00115965218203 2(13)(12)(720)

(α^{-1}=137.035998995(85) – independent from a_e)

Uncertainties come from:

$$A_1^{(10)},\ \ a_e(hadronic) + a_e(electroweak),\ \ \alpha$$

T. Aoyama, T. Kinoshita, M.Nio, Revised and improved value of the QED tenth-order electron anomalous magnetic moment, Physical Review D, 2018, V. 97, 036001.

My method was developed for computing $A_1^{(2n)}$

Motivation

- Independent calculation of $A_1^{(2n)}, n = 5,\ldots$
- Check the validity of some hypotheses and our belief in Quantum Field Theory:
  - The contributions of gauge invariant classes are relatively small, but the contributions of individual Feynman diagrams are relatively large (in absolute value)?
  - finiteness of $A_1^{(2n)}$, behavior of the whole series etc…
  - …
- Methods of high-order calculations

Universal QED contributions

$$a_e = a_e(QED) + a_e(hadronic) + a_e(electroweak),$$

$$a_e(QED) = \sum_{n\geq 1}\left(\frac{\alpha}{\pi}\right)^n a_e^{2n},$$

$$a_e^{2n} = A_1^{(2n)} + A_2^{(2n)}(m_e/m_\mu) + A_2^{(2n)}(m_e/m_\tau) + A_3^{(2n)}(m_e/m_\mu, m_e/m_\tau)$$

- J. Schwinger [1948], analytically: $A_1^{(2)} = 0.5$
- R. Karplus, N. Kroll [1949] – with a mistake

  A. Petermann [1957], C. Sommerfield [1958], analytically:

$$A_1^{(4)} = -0.328478966\ldots$$

- ~1970…~1975, 3 loops, numerically:
  1. M. Levine, J. Wright.
  2. R. Carroll, Y. Yao.
  3. T. Kinoshita, P. Cvitanović.

  T. Kinoshita, P. Cvitanović [1974]: $A_1^{(6)} = 1.195 \pm 0.026$
- E. Remiddi, S. Laporta et al., ~1965…1996, analytically: $A_1^{(6)} = 1.181241456\ldots$
- T. Kinoshita et al., numerically, 2015: $A_1^{(8)} = -1.91298(84)$
- S. Laporta, semi-analytically, 2017: $A_1^{(8)} = -1.9122457649\ldots$
- T. Kinoshita et al., numerically, 2015 (with a mistake): $A_1^{(10)} = 7.795(336)$
- T. Kinoshita et al., numerically, 2018: $A_1^{(10)} = 6.675(192)$

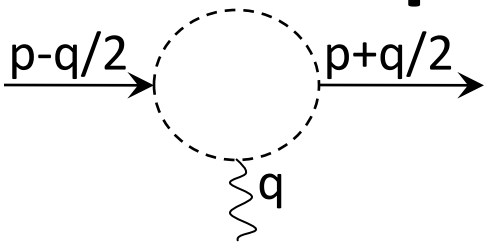
The method

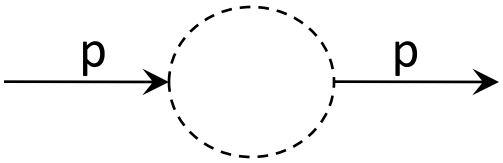
- Subtraction procedure for removing both IR and UV divergences in Feynman-parametric space for each individual Feynman diagram
- Diagram-specific importance sampling Monte Carlo integration algorithm for diagrams without lepton loops

The subtraction procedure

- FULLY AUTOMATED AT ANY ORDER OF THE PERTURBATION SERIES.
- UV and IR divergences are eliminated point-by-point in Feynman-parametric space for each individual Feynman diagram. No regularization is required.
- Subtraction by a forest formula with linear operators. Each operator transforms Feynman amplitude of some UV-divergent subdiagram G’ (in momentum space) to the polynom with the degree that is less or equal to $\omega(G')$.
- The subtraction is equivalent to the on-shell renormalization => no residual renormalizations, no calculations of renormalization constants, no other manipulations.

Operators

$\Gamma_\mu(p,q) =$ 

$\Sigma(p) =$ 

- A – projector of AMM

$$\bar{u}_2\Gamma_\mu(p,q)u_1 = \bar{u}_2(f(q^2)\gamma_\mu - g(q^2)\sigma_{\mu\nu}q^\nu/(2m) + h(q^2)q_\mu)u_1$$

$$\sigma_{\mu\nu} = (\gamma_\mu\gamma_\nu - \gamma_\nu\gamma_\mu)/2, \qquad (p-q/2)^2 = (p+q/2)^2 = m^2$$

$$(\hat{p} - \hat{q}/2 - m)u_1 = (\hat{p} + \hat{q}/2 - m)u_2 = 0$$

$$A\Gamma_\mu = \gamma_\mu \lim_{q^2\to 0} g(q^2)$$

- U – intermediate operator

$$\Gamma_\mu(p,0) = a(p^2)\gamma_\mu + b(p^2)p_\mu + c(p^2)\hat{p}p_\mu + d(p^2)(\hat{p}\gamma_\mu - \gamma_\mu\hat{p}) \qquad \Sigma(p) = r(p^2) + s(p^2)\hat{p}$$

$$U\Gamma_\mu = \gamma_\mu a(m^2) \qquad U\Sigma = r(m^2) + s(m^2)\hat{p}$$

IR-safe!

For the other types of divergent subgraphs, U=Taylor expansion at 0 up to ω order.

- L – on-shell renormalization for vertex-like subdiagrams

$$L\Gamma_\mu = \gamma_\mu(a(m^2) + b(m^2)m + c(m^2)m^2)$$

can produce additional IR divergences

Forest formula for AMM

A set of subgraphs of a diagram is called a **forest** if any two elements of this set don't overlap.

$\mathcal{F}$[G] – the set of all forests of UV-divergent subgraphs in G that contain G.

$\mathbb{I}$[G] – the set of all vertex-like UV-divergent subgraphs in G that contains the vertex that is incident to the external photon line of G.

$$\tilde{f}_G = \sum_{\substack{F=\{G_1,\ldots,G_n\}\in\mathcal{F}[G] \\ G'\in\mathbb{I}[G]\cap F}} (-1)^{n-1} K_{G_1}^{G'} \ldots K_{G_n}^{G'} f_G$$

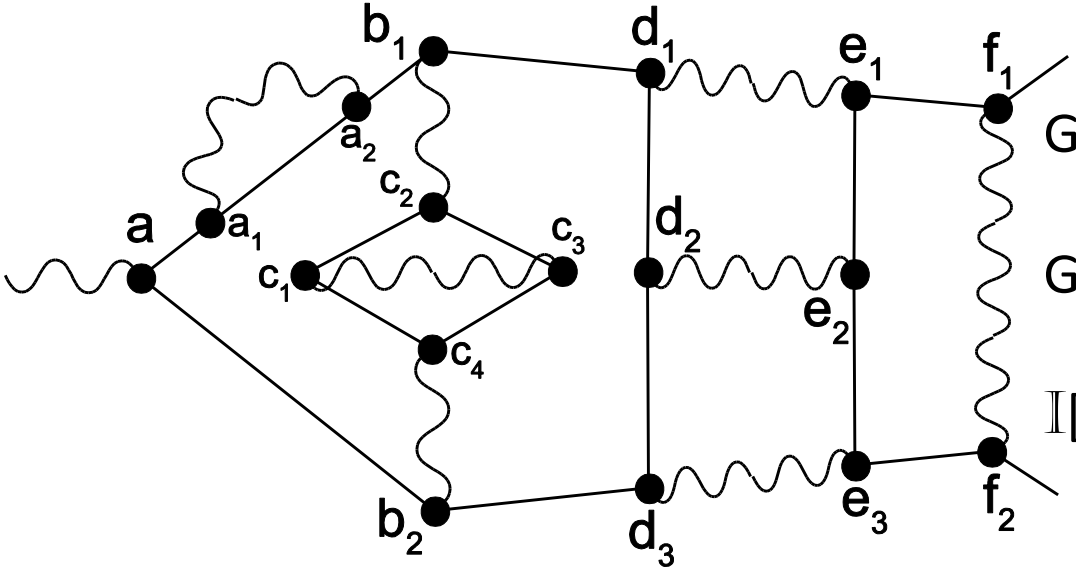
$$K_{G''}^{G'} = \begin{cases} A_{G'} \text{ for } G' = G'' \\ U_{G''} \text{ for } G'' \notin \mathbb{I}[G], \text{ or } G'' \subseteq G' \text{ and } G'' \neq G' \\ L_{G''} \text{ for } G'' \in \mathbb{I}[G],\ G' \subseteq G'',\ G'' \neq G,\ G'' \neq G' \\ (L_{G''} - U_{G''}) \text{ for } G'' = G,\ G' \neq G \end{cases}$$

$$\bar{f}_G = \text{coefficien t before } \gamma_\mu \text{ in } \tilde{f}_G$$

$$a_e = \sum_G \bar{f}_G$$

Details: S. Volkov, J. Exp. Theor. Phys. (2016), V. 122, N. 6, pp. 1008-1031

Example



$G_c=aa_1a_2b_1b_2c_1c_2c_3c_4$

$G_e=aa_1a_2b_1b_2c_1c_2c_3c_4d_1d_2d_3e_1e_2e_3$

$\mathbb{I}[G]=\{G_c,G_e,G\}$

Other UV-divergent subgraphs:

electron self-energy – a_1a_2, vertex-like – $c_1c_2c_3$, $c_1c_3c_4$,

photon self-energy – $c_1c_2c_3c_4$,

photon-photon scattering – $G_d=aa_1a_2b_1b_2c_1c_2c_3c_4d_1d_2d_3$

$$\tilde{f}_G=\left[A_G(1-U_{G_e})(1-U_{G_c})-(L_G-U_G)A_{G_e}(1-U_{G_c})-(L_G-U_G)(1-L_{G_e})A_{G_c}\right]\cdot$$
$$\cdot(1-U_{G_d})(1-U_{c_1c_2c_3c_4})(1-U_{c_1c_2c_3}-U_{c_1c_3c_4})(1-U_{a_1a_2})f_G$$

Importance sampling Monte Carlo

- Integral: $\int_\Omega f(x)dx$
- Probability density function: $g(x)$
- Approximation: $(1/N)\Sigma_{1\le j\le N}(f(x_j)/g(x_j))$
- Variance: $V(f,g)= \int_\Omega (f(x)^2/g(x))dx-(\int_\Omega f(x)dx)^2$
- Error estimation: $\sigma^2 \approx V(f,g)/N$
- The goal is to minimize $V(f,g)$ by choosing $g(x)$.

NON-ADAPTIVE MONTE CARLO WORKS FINE FOR HIGH-ORDER CALCULATIONS IN QFT!!!

Diagram-specific probability density functions

- Integral: $\int_{z_1,\ldots,z_M>0} f(z_1,\ldots,z_M)\delta(z_1+\ldots+z_M-1)dz$
- Hepp sectors: $z_{j_1} \geq z_{j_2} \geq \ldots \geq z_{j_M}$
- Density: $C \cdot \frac{\prod_{l=2}^{M} (z_{j_l} / z_{j_{l-1}})^{Deg(\{j_l, j_{l+1},\ldots,j_M\})}}{z_1 \cdot z_2 \cdot \ldots \cdot z_M}$,

Deg is defined on subsets of {1,…,M}

(the idea of E.Speer, J. Math. Phys. 9, 1404 (1968))

- My ideas are:

1) how to calculate *Deg*(s) for each set s
   (taking into account the infrared behavior etc.)
2) how to generate samples fastly

Obtaining *Deg*(s)

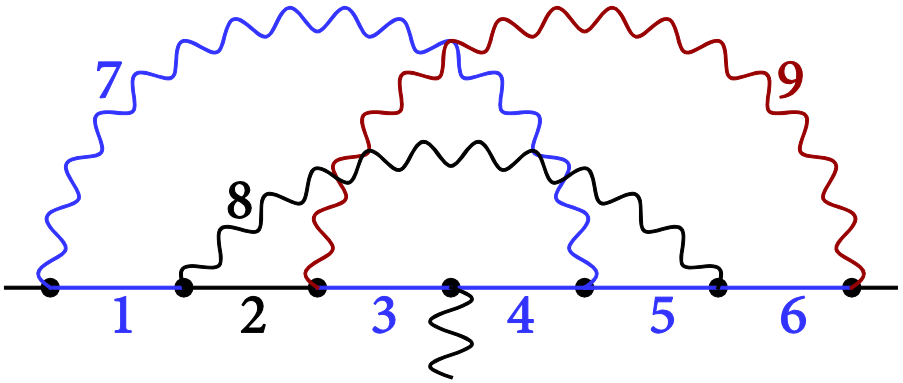
- Sector: $z_{j_1} \geq z_{j_2} \geq \ldots \geq z_{j_M}$
- Density: $C \cdot \dfrac{\prod_{l=2}^{M} (z_{j_l} / z_{j_{l-1}})^{Deg(\{j_l, j_{l+1}, \ldots, j_M\})}}{z_1 \cdot z_2 \cdot \ldots \cdot z_M},$

•The rules are constructed using ultraviolet degrees of divergence (with the sign '-') of I-closures of sets

(the full description taking into account divergent subdiagrams is in arXiv:1705.05800)

•I$\mathrm{Clos}(s)$=sUs', where *s'* is the set of all photon lines for which the electron path connecting their ends is contained in *s*

Example: IClos({1,3,4,5,6,7})={1,3,4,5,6,7,9}



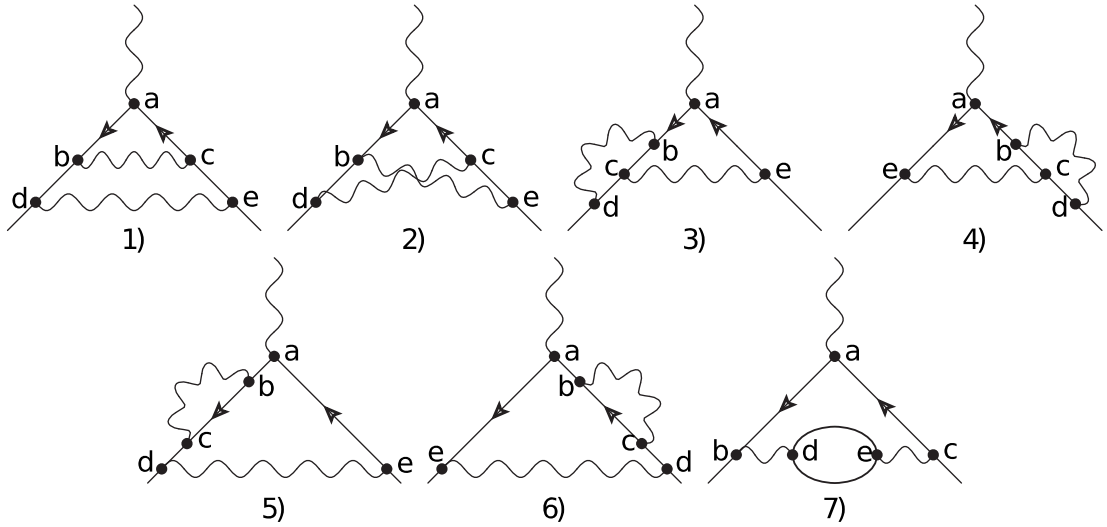
Realization and numerical results

- Monte Carlo integration on 1 GPU of NVidia Tesla K80
- 2 loops: all Feynman diagrams (with lepton loops: old, 2015)
- 3 loops: all Feynman diagrams (with lepton loops: old, 2015)
- 4 loops: diagrams without electron loops
- up to 6 loops: ladder diagrams

Realization on GPU from NVidia Tesla K80

- Computer with GPU leased from Google Cloud (free trial)
- Most part of the Monte Carlo is performed on GPU (integrand evaluation, sample generation, most of processing): 19968 GPU threads
- Problem: memory speed
  Solution: memory-optimized code for integrands
  (most of operations are performed with GPU register memory)
- Problem: big integrand code size, compilation speed
  Solution:
  shared libraries (dynamically linking),
  each CUDA kernel contains ~5000 operations,
  many shared libraries for 1 Feynman diagram
- Problem: round-off errors caused by numerical subtraction of divergences
  Solution:
  Interval Arithmetic (double-precision, GPU supports all that is needed)
  Eliminated Interval Arithmetic (fast, but less distinct intervals)
  128-bit mantissa numbers (uses the GPU register memory)
  256-bit mantissa numbers

2 loops: all Feynman diagrams



#	My value	Analytical value (Petermann, 1957)
1	0.77747774(18)	0.77747802
2	-0.4676475(17)	-0.46764544
3,4	-0.0640193(19)	-0.564021–(1/2)log(λ^2/m^2)
5,6	-0.5899758(14)	-0.089978+(1/2)log(λ^2/m^2)
7	0.0156895(25)	0.0156874

2015: $A_1^{(4)} = -0.328513(87)$

2018: $A_1^{(4)}[\text{no lepton loops}] = -0.3441651(34)$

Analytical, 1957:

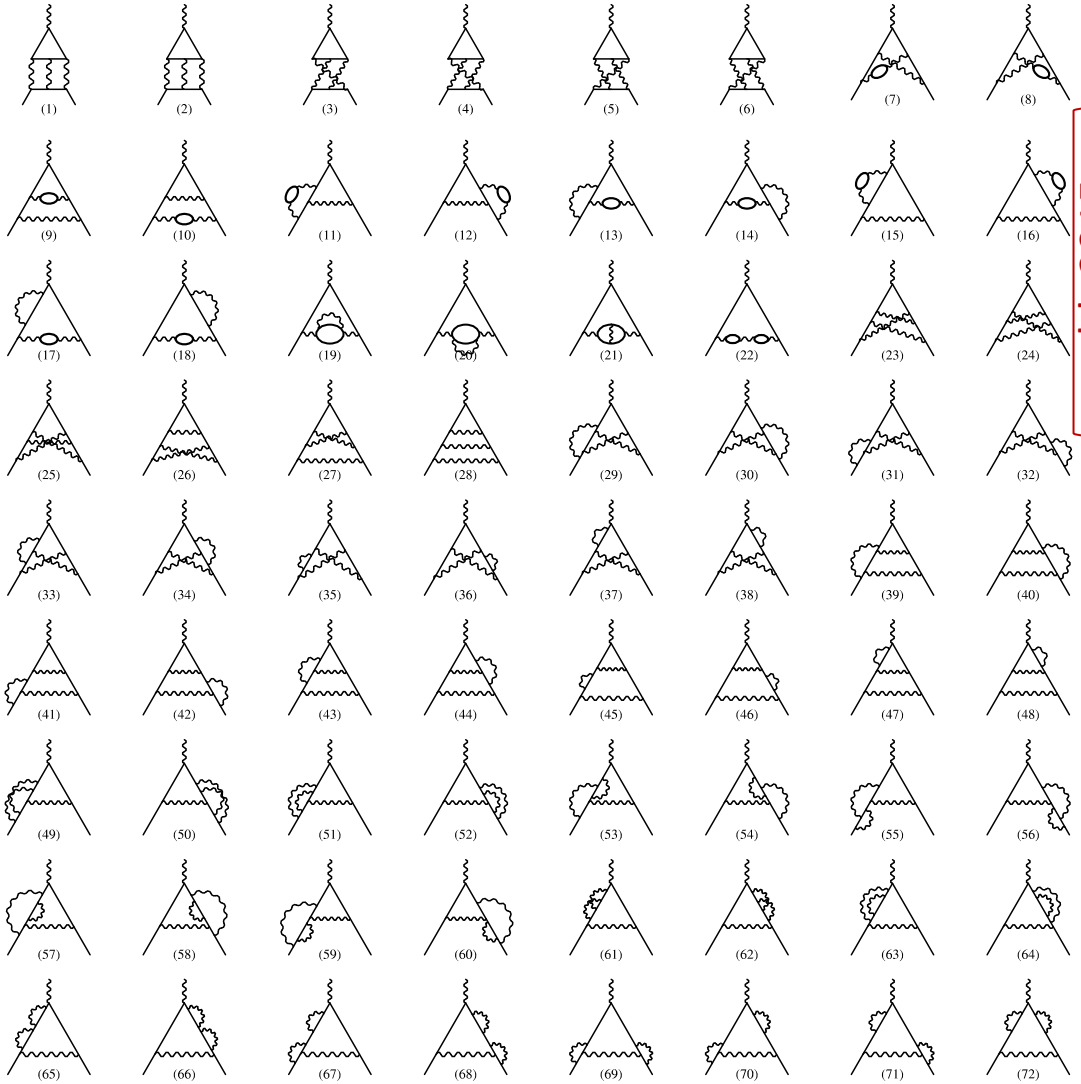
$A_1^{(4)} = -0.328478966\ldots$

$A_1^{(4)}[\text{no lepton loops}] = -0.3441663\ldots$

⇐ old: 2015

3 loops: all Feynman diagrams $A_1^{(6)}[\text{no lepton loops}] = 0.90485(10)$

Analytical (1996): 0.904979



3-loop Feynman diagrams for electron's AMM. Plot courtesy of F.Jegerlehner

Comparison with known analytical values

old: 2015

#	My value	Analyt. val.	Reference
1-6	0.3708(14)	0.3710	[10]
7-10	0.04989(20)	0.05015	[4,5]
11-12,15-16	-0.08782(15)	-0.08798	[2,4]
13-14,17-18	-0.11230(17)	-0.11234	[3,4]
19-21	0.05288(13)	0.05287	[1]
22	0.002548(20)	0.002559	[1]
23-24	1.861914(17)	1.861908	[11]
25	-0.0267956(78)	-0.026799	[12]
26-27	-3.176700(22)	-3.176685	[8]
28	1.790285(19)	1.790278	[8]
29-30	-1.757945(15)	-1.757936	[12]
33-34,37-38	0.455517(26)	0.455452	[8,11]
31-32,35-36	1.541644(37)	1.541649	[7,9]
39-40	-0.334691(14)	-0.334695	[11]
41-48	-0.402749(46)	-0.402717	[6,7]
49-68	0.533289(54)	0.533355	[6-9,11,12]
69-72	0.421080(43)	0.421171	[6,7,9]

[1] J. Mignaco, E. Remiddi, IL Nuovo Cimento, V. LX A, N. 4, 519 (1969).
[2] R. Barbieri, M. Caffo, E. Remiddi, Lettere al Nuovo Cimento, V. 5, N. 11, 769 (1972).
[3] D. Billi, M. Caffo, E. Remiddi, Lettere al Nuovo Cimento, V. 4, N. 14, 657 (1972).
[4] R. Barbieri, E. Remiddi, Physics Letters, V. 49B, N. 5, 468 (1974).
[5] R. Barbieri, M. Caffo, E. Remiddi, Ref.TH.1802-CERN (1974).
[6] M. Levine, R. Roskies, Phys. Rev. D, V. 9, N. 2, 421 (1974).
[7] M. Levine, R. Perisho, R. Roskies, Phys. Rev. D, V. 13, N. 4, 997 (1976).
[8] R. Barbieri, M. Caffo, E. Remiddi et al., Nuclear Physics B 144, 329 (1978).
[9] M. Levine, E. Remiddi, R. Roskies, Phys. Rev. D, V. 20, N. 8, 2068 (1979).
[10] S. Laporta, E. Remiddi, Physics Letters B 265, 182 (1991).
[11] S. Laporta, Physics Letters B 343, 421 (1995).
[12] S. Laporta, E. Remiddi, Physics Letters B 379, 283 (1996).

4 loops: diagrams without electron loops

My result: -2.181(10) 1 week on GPU

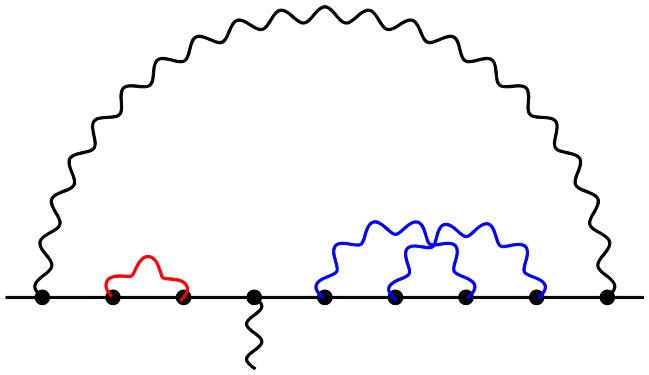
Laporta, 2017: -2.1768660277…

- 269 Feynman diagrams
- 78 classes of diagrams for comparison with the direct subtraction on the mass shell
- 6 gauge-invariant classes (k,m,m')

(k,m,n):

m and n photon lines to the right and to the left from the external photon (or vice versa), k photon lines with ends on different sides

Example of a diagram from (1,2,1):

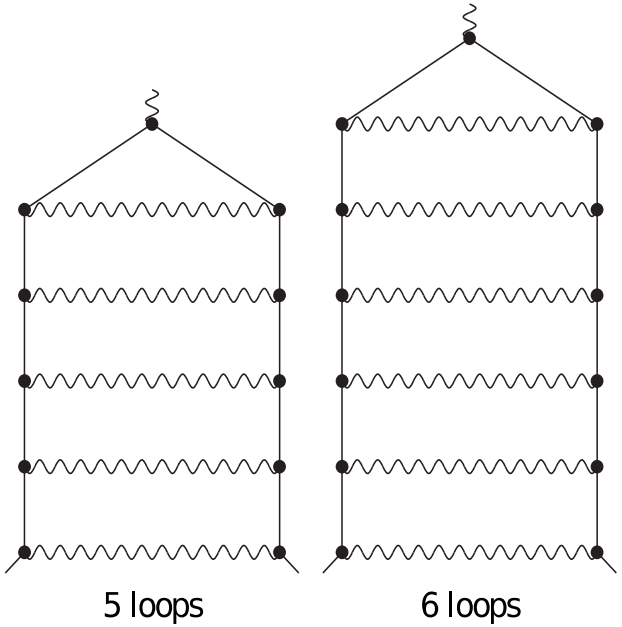


Class	Value	Laporta, 2017
(1,3,0)	-1.9710(44)	-1.97107…
(2,2,0)	-0.1415(56)	-0.14248…
(1,2,1)	-0.6220(46)	-0.62192…
(3,1,0)	-1.0424(44)	-1.04054…
(2,1,1)	1.0842(37)	1.08669…
(4,0,0)	0.5120(17)	0.51246…

Ladder diagrams: 5 and 6 loops

loops	My value	Analytical value	$N_{samples}$	time
5	11.6530(58)	11.6592…	$29 \cdot 10^9$	5 hours
6	34.31(20)	34.367	10^{10}	8 hours

All analytical values are from M. Caffo, S. Turrini, E.Remiddi, Nuclear Physics B141 (1978) 302-310.



Technical information

	2 loops	3 loops	4 loops	5-loop ladder	6-loop ladder
Value	-0.3441651(34)	0.90485(10)	-2.181(10)	11.6530(58)	34.31(20)
Samples: total	$33\cdot10^{11}$	$81\cdot10^{11}$	$32\cdot10^{11}$	$29\cdot10^{9}$	10^{10}
Samples: fail, double-precision Eliminated IA	$71\cdot10^{8}$	$17\cdot10^{10}$	$18\cdot10^{10}$	$32\cdot10^{8}$	$12\cdot10^{8}$
Samples: fail, double-precision Interval Arithmetic (IA)	$68\cdot10^{6}$	$21\cdot10^{8}$	$13\cdot10^{8}$	$9\cdot10^{6}$	$72\cdot10^{5}$
Samples: fail, 128-bit mantissa IA	2	12590	77775	934	4504
Contribution: fail, double-precision Eliminated IA	0.002	0.4	2	5	20
Contribution: fail, double-precision IA	0.0001	0.002	0.2	0.4	3
Contribution: fail, 128-bit mantissa IA	$-2\cdot10^{-19}$	-10^{-6}	-0.0006	$4\cdot10^{-10}$	$-5\cdot10^{-5}$
Total calculation time	22 hours	5 days	7 days	5 hours	8 hours
Share in the time: double-precision Eliminated IA	19.1%	41.7%	54.5%	56.4%	42.0%
Share in the time: double-precision IA	0.1%	1.6%	9.1%	15.4%	24.4%
Share in the time: 128-bit mantissa IA	0.2%	2.7%	9.2%	6.7%	24.3%
Share in the time: 256-bit mantissa IA	0.0%	0.3%	2.1%	8.1%	5.2%
Share in the time: sample generation	63.7%	45.9%	21.7%	12.0%	3.7%
Share in the time: other operations	16.9%	7.7%	3.4%	1.3%	0.3%
GPU speed: double-precision EIA (GFlop/s)	334.24	222.72	234.26	187.93	292.67
GPU speed: double-precision EIA (GInterval/s)	53.76	63.51	142.27	103.04	240.91
GPU speed: double-precision IA (GFlop/s)	254.11	221.41	255.85	249.00	287.94
GPU speed: double-precision IA (GInterval/s)	36.23	35.80	47.22	45.60	55.81
GPU speed: 128-bit mantissa IA (GFlop/s)	0.81	1.59	1.58	1.63	1.66
GPU speed: 128-bit mantissa IA (GInterval/s)	0.11	0.23	0.26	0.30	0.32
GPU speed: 256-bit mantissa IA (MFlop/s)	0.0204	0.0881	0.3503	0.1378	4.8504
GPU speed: 256-bit mantissa IA (MInterval/s)	0.0028	0.0124	0.0537	0.0252	0.9401
Integrand code size: not compiled	887 KB	31 MB	2.5 GB	23 MB	186 MB
Integrand code size: compiled	12 MB	115 MB	4 GB	34 MB	252 MB

Thank you for your attention!

volkoff_sergey@mail.ru
sergey.volkov.1811@gmail.com

ЖЭТФ, т. 149, вып. 6, стр. 1164-1191 (2016)
J. Exp. Theor. Phys. 122, 1008 (2016)
arXiv:1507.06435 (short version)

Phys. Rev. D 96, 096018 (2017)
arXiv:1705.05800

arXiv:1807:05281 (2018)