

Direct numerical computation and its application to the higher-order radiative corrections



K Kato, E de Doncker, T Ishikawa and F Yuasa

ACAT2017, 21-25 August 2017

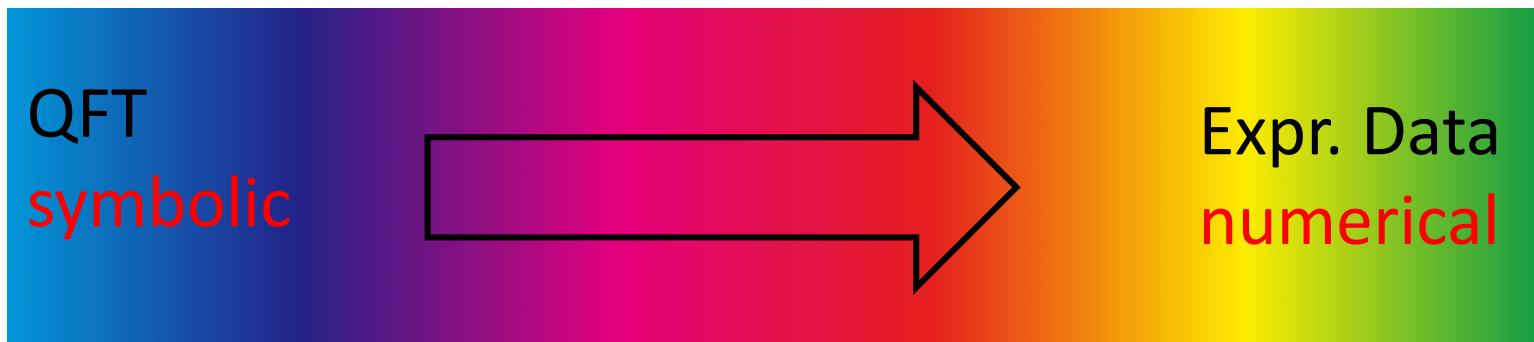
University of Washington, Seattle

Introduction

- Precise theoretical prediction in HEP
 - need of the large scale computation
 - (many particle final states, higher order radiative correction,...)
 - sometimes beyond man-power
 - solution: automated system for QFT
- Library for multi-loop integrals is required for the general external momenta, various masses in EW/SUSY,
 - to be used as a vital unit in an automated system for the perturbative computation

Library for multi-loop integrals

Analytical methods and Numerical methods



DCM(Direct Computation Method),
A ‘maximally’ numerical method
Numerical Integral + Series extrapolation

multi-loop integrals and singularity

singularity → regularization → numerical integration

Target

$$I = (-)^N \frac{\Gamma(N - nL/2)}{(4\pi)^{nL/2}} \int_0^1 \prod_{r=1}^N dx_r \delta(1 - \sum x_r) \frac{Num}{U^{n/2} (V - i\varepsilon)^{N-nL/2}}$$

Case-2 : UV divergence

Take n off 4, ε finite [M^0]

$$n = 4 - 2\varepsilon$$

Case-1 : zero denominator
keep ε finite [M^2]

All cases are already
handled by DCM

Case-3 : IR divergence
Take n off 4 or finite λ
 $n = 4 + 2\varepsilon$

DCM: Direct computation method

DCM= regularized integration
 + series extrapolation

⊗ Calculate the integral with finite ε 's $I(\varepsilon_j)$

For finite values, the integral is convergent numerically.

⊗ Estimate the integral by extrapolation $I = \lim_{\varepsilon \rightarrow 0} I(\varepsilon_j)$

- Extrapolation by Wynn's algorithm
- Linear solver (LU decomposition)

Numerical integration

Numerical integration packages

DQ ... DQAGE/DQAGS routine in Quadpack package
(<http://www.netlib.org/quadpack/>)

ParInt package ... Adaptive method
(<https://cs.wmich.edu/parint/>)

DE ... Double exponential formula
(<http://www.sciencedirect.com/science/article/pii/S037704270000501X>)

Parallel computing in multi-core environment

MPI(Message Passing Interface) ... distributed memory

OpenMP(Multi-Processing) ... shared memory

extrapolation

$$\{\varepsilon_j\} \quad (j=1..n) \rightarrow \text{Integration} \rightarrow \{I(\varepsilon_j)\}$$

Wynn's algorithm

$$a(j, k+1) = a(j+1, k-1) + \frac{1}{a(j+1, k) - a(j, k)}$$

Input

$$a(j, 0) = I(\varepsilon_j), \quad a(j, -1) = 0$$

Linear solver

$$\varepsilon_j^K I(\varepsilon_j) = C_0 + C_1 \varepsilon_j + C_2 \varepsilon_j^2 + \cdots + C_n \varepsilon_j^n$$

Larger n is NOT always good, but an appropriate n exists.

ACAT2016 DCM

4	 8 		
3	 7 		
2	4 	5 	6
Loops	Self energy	Vertex	Box



massless

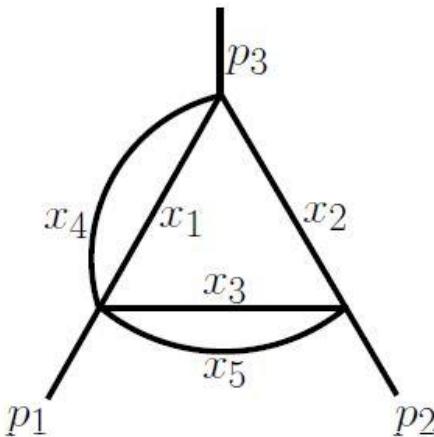


UV divergence
in integral part

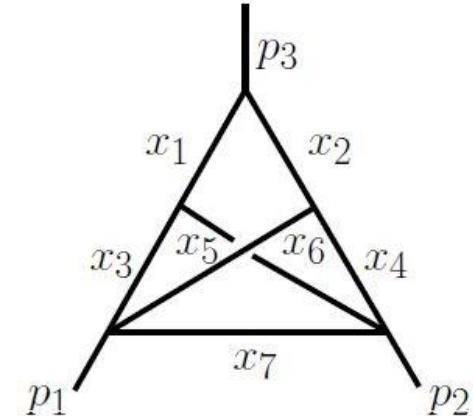
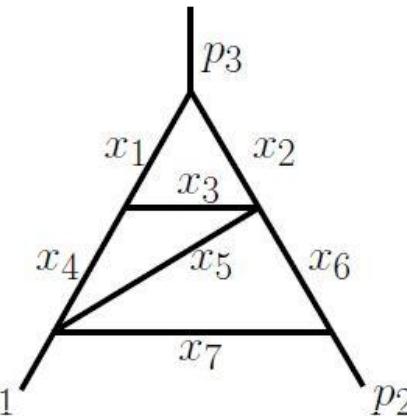


(computed)
dimension of integral

3-loop vertex(scalar)



(b)



(c)

massless

$$p_1^2 = p_2^2 = 0, p_3^2 = s$$

E de Doncker and F Yuasa
Procedia Computer Science 108C (2017) 1773–1782

Analytic results

- (a) T. Gehrmann, G. Heinrich, T. Huber, and C. Studerus (2006)
- (b,c) G. Heinrich, T. Huber, and D. Maître. (2008)

Result (a) 4-dim.

$$I_a = C_{-2} \frac{1}{\varepsilon^2} + C_{-1} \frac{1}{\varepsilon} + C_0$$

INTEGRAL Fig 1(a)						
ℓ	E_r	$T[s]$	RES. C_{-2}	RES. C_{-1}	RES. C_0	
38	6.2e-11	324.7				
39	1.1e-10	326.2	-0.159493974	-2.2785290		
40	1.6e-10	327.7	-0.167414029	-1.5629825	-16.11779	
41	2.2e-10	329.2	-0.166606927	-1.6784806	-10.62839	
42	3.3e-10	330.1	-0.166670468	-1.6656668	-11.59303	
43	6.1e-10	330.4	-0.166666467	-1.6667337	-11.47985	
44	8.3e-10	331.4	-0.166666675	-1.6666632	-11.48974	
analytic		(6):	-0.166666667	-1.6666667	-11.48913	

$\varepsilon_\ell = 1.11^{-\ell}$ Integral by ParInt on *thor* cluster (4 x 16 procs., MPI) in long double precision. Max 50B evaluation. 332s per iteration. Extrapolation by linear solver.

Result (b,c) 6-dim.

$$I_b = C_{-1} \frac{1}{\varepsilon} + C_0$$

INTEGRAL Fig 1(b)					
ℓ	E_r	$T[s]$	RES. C_{-1}	RES. C_0	
25	7.4e-06	972			
26	1.2e-05	972	-10.98454	-31.1676	
27	1.7e-05	972	-12.111262	-63.5043	
28	2.4e-05	972	-12.22893	-71.5243	
29	3.4e-05	972	-12.34703	-75.2208	
analytic		(7):	-12.34658	-75.3620	

$$I_c = C_0 + C_1 \varepsilon$$

INTEGRAL Fig 1(c)					
ℓ	E_r	$T[s]$	RES. C_0	RES. C_1	
25	1.3e-07	631			
26	1.1e-07	631	33.8889518	338.4550	
27	9.7e-08	631	34.1049441	293.1244	
28	8.4e-08	631	34.0967036	295.9858	
29	8.2e-08	631	34.0969222	295.8739	
analytic		(8):	34.0969298	295.8700	

$$\varepsilon_\ell = 1.11^{-\ell}$$

$$\varepsilon_\ell = 1.2^{-\ell}$$

Integral by ParInt on *thor* cluster (4 x 16 procs., MPI) in long double precision. Evaluation (b) max 125B (c) max 80B

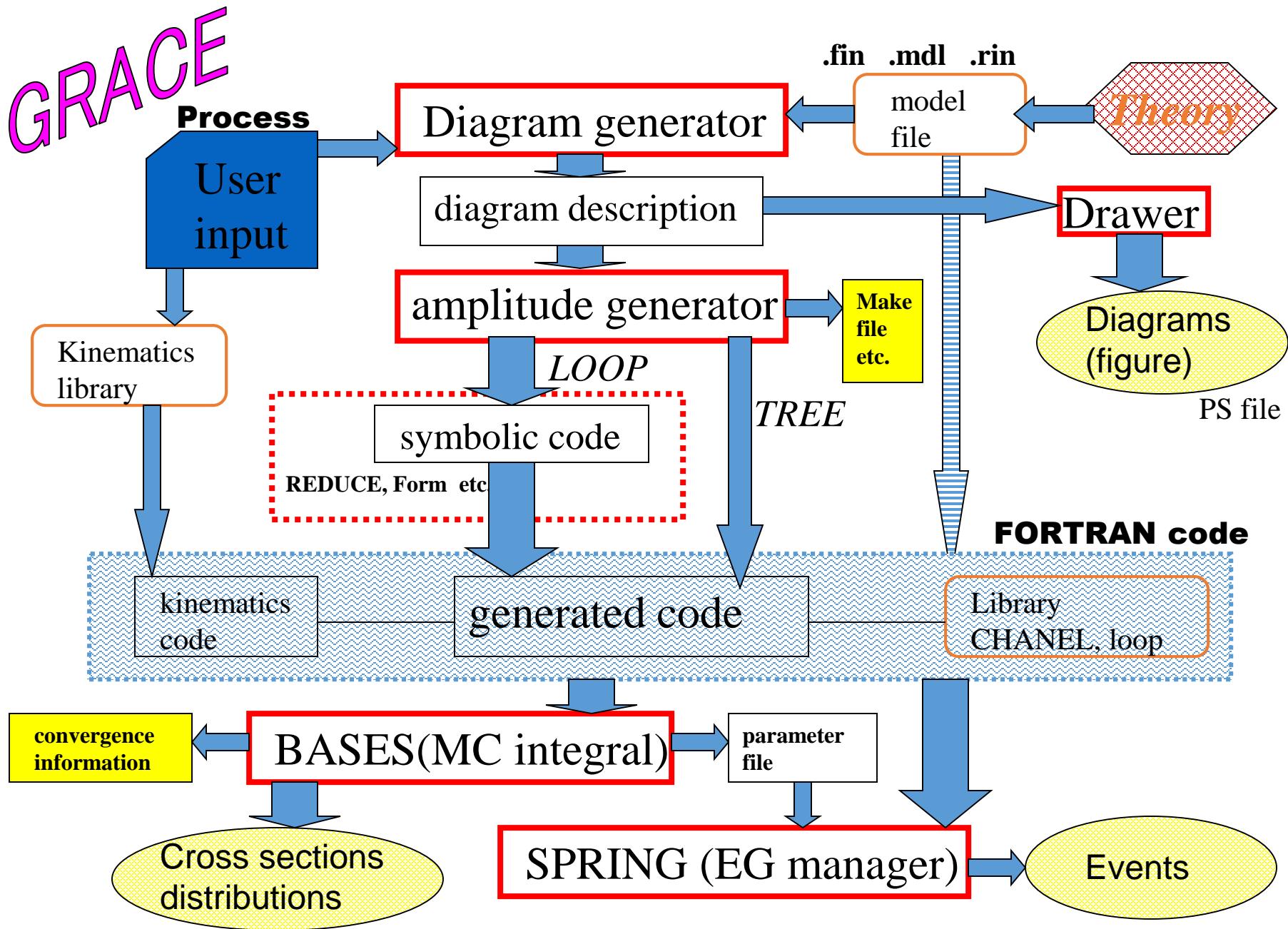
ACAT2017 DCM

4	 8		
3	 7	 6	
2	 4	 5	 6
Loops	Self energy	Vertex	Box

★ massless
★ massive

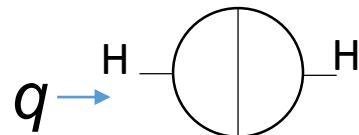
UV divergence
in integral part

n (computed)
dimension of integral



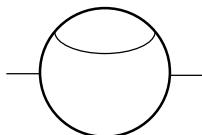
GRACE H-H 2-loop 2point function in EW(w NLG, no tadpole) ... 3082 diagrams(inc. C.T.)

2-loop diagrams



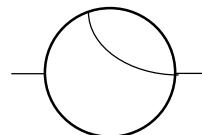
416

4-dim. L.F.



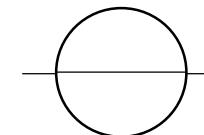
544

3-dim. L.F.
(1 dim. Trivial)



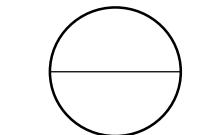
72 (& reversed)

3-dim.



18

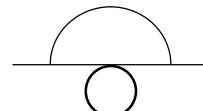
2-dim.



103

2-dim. (L.F.)
(1 dim. Trivial)

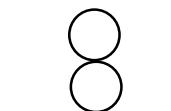
Product of 2 1-loop



128



55



52

others

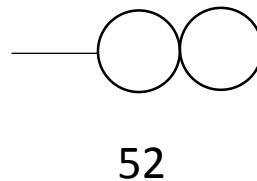
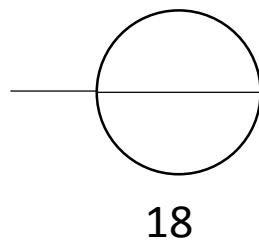
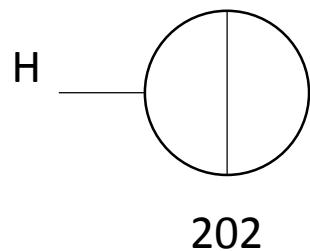


Counter terms
163

L.F. = some diagrams includes light fermions
to make zero denominator to be processed
by double regularization

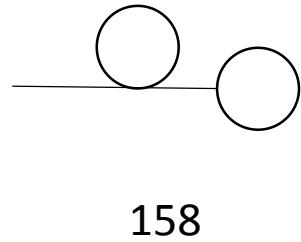
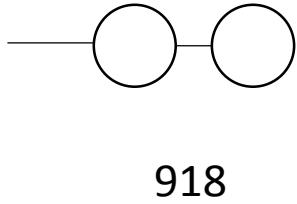
GRACE 2-loop H tadpole in EW(w NLG)

... 1934 diagrams(inc. C.T.)

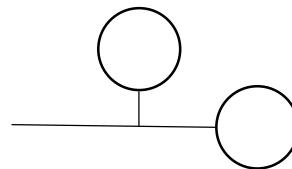


Common to
H-H 2-loop

(HH 1-loop) x (1-loop tadpole)



Product of 2 1-loop



Counter terms
55

2-loop amplitude

N is generated by GRACE
as REDUCE code

$$T = \int [d\ell_1][d\ell_2] \frac{N(\ell_1, \ell_2)}{(k_1^2 - m_1^2)(k_2^2 - m_2^2) \cdots (k_N^2 - m_N^2)} \quad n = 4 - 2\varepsilon$$

$$T = \int dx_1 \cdots dx_N \delta(1 - \sum x_k) G \quad G = \Gamma(N) \int [d\ell_1][d\ell_2] \frac{N(\ell_1, \ell_2)}{\Delta^N} \\ \Delta = \vec{\ell} \overset{t \rightarrow}{\overrightarrow{A}} \vec{\ell} + 2 \overset{t \rightarrow \rightarrow}{\vec{\ell}} \vec{b} + C$$

Transformation

$$\vec{\ell} \rightarrow \vec{\ell} - A^{-1} \vec{b}$$

$$\Rightarrow \Delta = \ell_1^2 + \ell_2^2 - V$$

$$\vec{\ell} \rightarrow O \vec{\ell}$$

$$N \rightarrow f^{00} + f^{10}\ell_1^2 + f^{01}\ell_2^2$$

$$\ell_j \rightarrow \ell_j / \sqrt{\lambda_j}$$

$$+ f^{20}(\ell_1^2)^2 + f^{11}\ell_1^2\ell_2^2 + f^{02}(\ell_2^2)^2$$

After the loop integral

$$U = \det A = \prod \lambda_j$$

$$G = \sum \frac{(\Gamma's)}{(4\pi)^n} \frac{f^{km}}{U^{n/2} V^{N-n-k-m}}$$

FORTRAN code for G is made by
REDUCE filter for each topology

Variable transformation for each topology (J=Jacobian)

$$T = \int dz_1 \cdots dz_{N-1} J G$$

$$\{z\} \in [0,1]^{N-1}$$

Feynman parameter integral is
calculated by DCM

$$G = \sum \frac{(\Gamma's)}{(4\pi)^n} \frac{f^{km}}{U^{n/2} V^{N-n-k-m}}$$

$$\longrightarrow T = \frac{1}{\varepsilon^2} C_{-2} + \frac{1}{\varepsilon} C_{-1} + C_0 + \dots$$

$$V = M^2 - (W/U)s$$

U, W Polynomials of x's

$$M^2 = \sum m_k^2 x_k$$

$$U = z(1-z+zt(1-t)) \quad \text{universal}$$

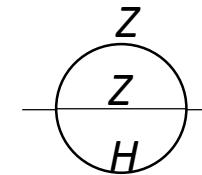
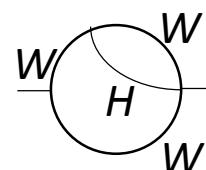
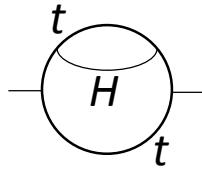
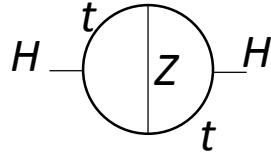
$$z = z_1, \quad t = z_2$$

$$s = q^2$$

$$\frac{d}{ds} T \rightarrow \frac{d}{ds} G = \sum \frac{(\Gamma's)}{(4\pi)^n} \left[\frac{df^{km}/ds}{U^{n/2} V^{N-n-k-m}} - C \frac{f^{km}(W/U)}{U^{n/2} V^{N-n-k-m+1}} \right]$$

performance

$$C_{-2} \frac{1}{\varepsilon^2} + C_{-1} \frac{1}{\varepsilon} + C_0$$



$$S = M_H^2$$

1 h / Coef. (II)

20s / Coef. (I)

10s / Coef. (I)

0.6 s / Coef. (I)

L.F. 246/416

L.F. 188/545

Heavy cases with light fermion
(not yet) : double extrapolation
→ Parallel computing

- (I) Intel(R) Xeon(R) CPU E5-1660 0 @ 3.30GHz
- (II) Intel(R) Xeon(R) CPU E3-1280 v5 @ 3.70GHz
- Wynn's algorithm (15 terms) $\varepsilon = 1.2^{-\ell}$
- DQ numerical integration
- Also show agreement with DE

conclusion

- DCM works well for the calculation of multi-loop integrals up to 8 dimensional parameter space.
- Application to the 2-loop radiative corrections in full electro-weak theory seems to be possible within a practical computational time and resources.

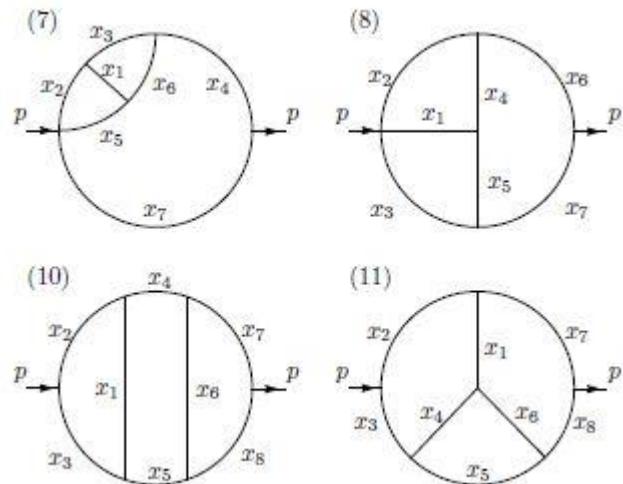
A large, multi-jet fountain is spraying water high into the air. The water droplets are visible against a clear blue sky. In the background, there are dark green evergreen trees and a range of mountains under a hazy sky.

Thank you!

3-loop self-energy

Finite integrals , no extrapolation

	dim	Result, p=1	Result, p=64	T(1) [s]	T(64) [s]	T1/T64
(7)	7	1.3264481	1.3264435	529.8	7.90	67.1
(8)	7	1.34139923	1.34139917	431.6	8.14	53.0
(10)	8	0.27960890	0.2796084	504.3	7.84	64.3
(11)	8	0.18262722	0.18262720	423.6	8.17	51.8



Comparison with Laporta(s=1, m=1)

Absolute tolerance = 5×10^{-8} ,
 Max evaluations = 5B,
 T64 on *thor* cluster with $p = 64$ processes
 (distributed over four 16-core nodes)

3-loop self-energy

UV-div. (up to 3rd order)

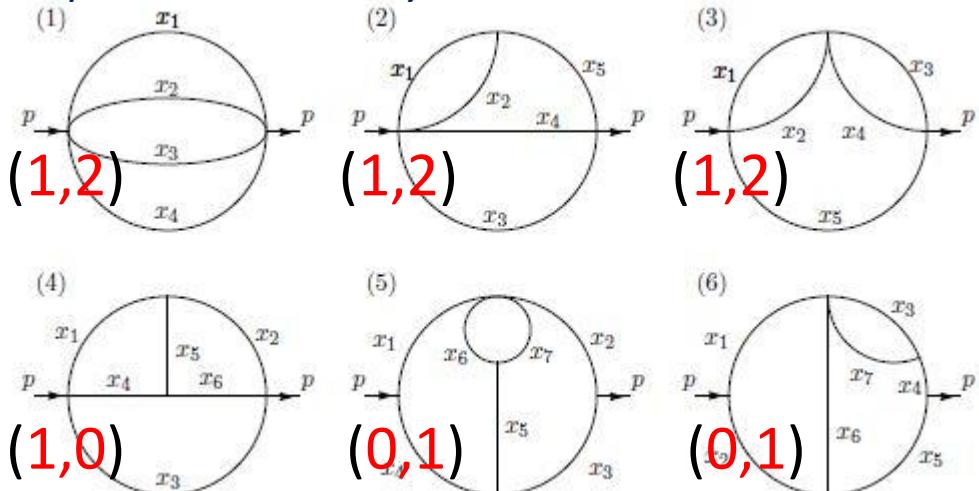
(5) Ladybug

Laporta (s=1, m=1)

$$0.923631826 \frac{1}{\epsilon} - 2.42349163 + \dots$$

Other diagrams are also computed.

Comparison with analytical results is OK.



Divergence order
(Gamma, Integral)

DCM (DE)

$$C_{-1} = 0.92370 \pm 0.434 \times 10^{-3}$$

$$C_0 = -2.4201 \pm 0.424 \times 10^{-1}$$

E5-2687W v3
@ 3.10GHz
Quadruple prec.
40 thread

Neval	mesh	Elapsed (Ks)	CPU(Ms)
105	0.1265988	65	2.1
95	0.1253191	37	1.2
83	0.1267232	18	0.57

6-dim.
Max eval
 $= (10^2)^6$
 $= 10^{12}$

extrapolation

0.49493857234
0.53603234731
0.57465585827
0.6106239846
0.6438487350
0.6743208746
0.702092730
0.7272628131
0.749962533
0.7703450682
0.7885762731
0.804827478
0.819269932
0.832070673
0.843389581
0.85337740
0.8621745
0.869910

0.92370 0.9236528

linear solver

4-loop self-energy massless, finite and UV div.

$$BC = -0.00173611111111/\varepsilon - 0.016927083333 - 0.01184293016\varepsilon + \dots$$

(1) $-0.00173611111109 \quad -0.016927083381 \quad -0.011842916$
 Elapsed : 48s : ParInt, thor cluster 64 threads

(2) $BC = 5.184638776/\varepsilon - 2.582436090 + 70.39915145\varepsilon + \dots$ BC
 5.1846392 -2.582434 70.39877
 Elapsed : 4.8h : DE, CPU KEKSC(SR-16000,64thread)

Analytic results: P.A. Baikov
 and K.G.Chetyrkin NPB 837
 (2010) 186-220

(3) $BC = 55.58525391567 + O(\varepsilon)$
 55.585150 Elapsed : 554 s
 eval. 300B, ParInt, thor cluster 64 threads

(4) $BC = 52.01786874361 + O(\varepsilon)$
 52.017714 Elapsed : 659 s
 eval. 275B, ParInt, thor cluster 64 threads

