Master integrals and generalized polylogarithms: focus on fast and efficient evaluation

Damiano Tommasini
(N.C.S.R. Demokritos)

In collaboration with
Costas Papadopoulos, Hjalte Frellesvig and Chris Wever

Funded by: ΑΡΙΣΤΕΙΑ-1283 HOCTools

OPERATIONAL PROGRAMME EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION & RELIGIOUS AFFAIRS, CULTURE & SPORTS
MANAGING AUTHORITY
European Union
European Social Fund
NSRF 2007-2013
programme for development
EUROPEAN SOCIAL FUND
Co-financed by Greece and the European Union

UCLARadcor-Loopfest 2015

18 June 2015

Outline

- Introduction: looking forward for 2-loop calculations

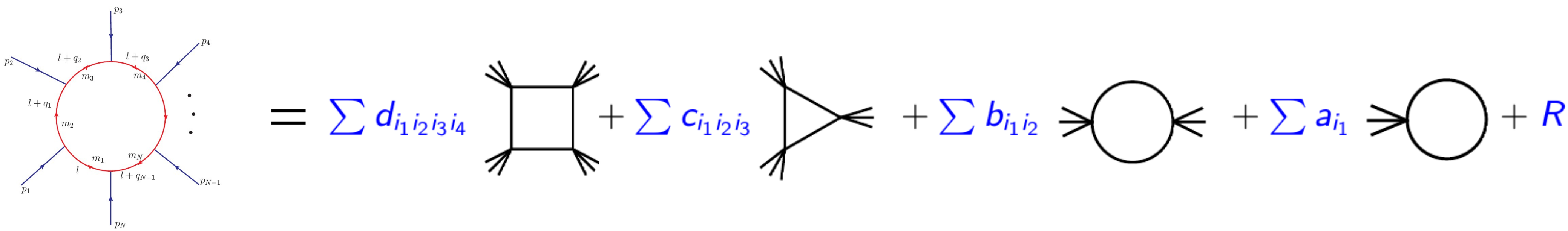
- Evaluation of Master Integrals expressed as Goncharov Multiple polylogarithms (GPs) and Li functions

See also Papadopoulos, Wever, Tancredi's... talks

- Looking forward for new algorithms from $\mathrm{Li}_n(x)$ to $\mathrm{Li}_{22}(x,y)$

Introduction

- Amplitude reduction @ 1-loop (OPP method) lead to so called NLO revolution



- Highly desirable: reduction @ 2-loops

  - Need to compute Master Integrals

  - Understand how to do amplitude reduction

Both steps are done in several cases, but still a lot has to be done.

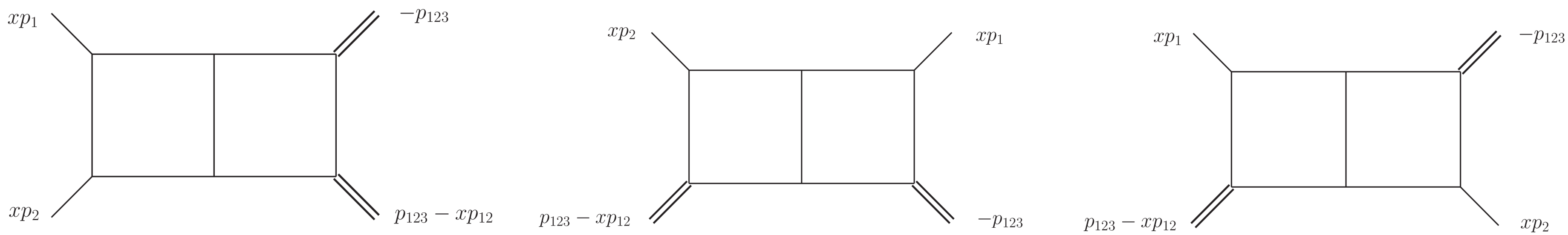
Planar double boxes

$$S = (q_1+q_2)^2 = (q_3+q_4)^2, \quad T = (q_1-q_3)^2 = (q_2-q_4)^2$$

$$U = (q_1-q_4)^2 = (q_2-q_3)^2, \qquad q_3^2 = M_3^2,\, q_4^2 = M_4^2$$

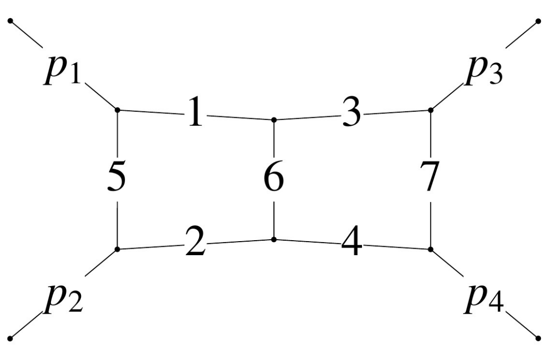
- x-parametrization

[**PTW**: C. G. **P**apadopoulos, D**T**, C. **W**ever '14]



- xyz-parameterization

[**HMS**: J. M. **H**enn, K. **M**elnikov, V. A. **S**mirnov '14;
GMT: T. **G**ehrmann, A. von **M**anteuffel, L. **T**ancredi '15]

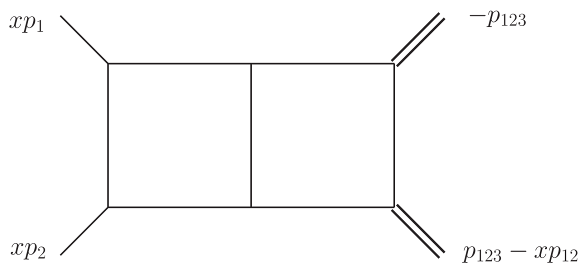


Planar double boxes

$$S=(q_1+q_2)^2=(q_3+q_4)^2, \quad T=(q_1-q_3)^2=(q_2-q_4)^2$$
$$U=(q_1-q_4)^2=(q_2-q_3)^2, \qquad q_3^2=M_3^2, q_4^2=M_4^2$$

- x-parametrization

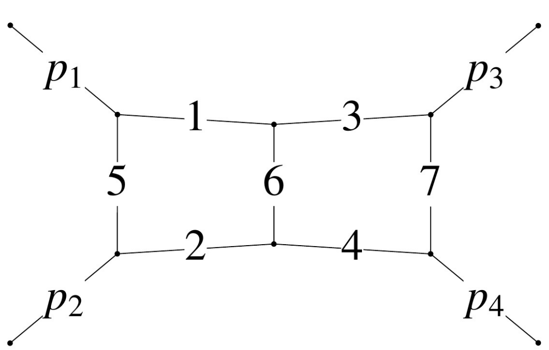
[**PTW**: C. G. **P**apadopoulos, D**T**, C. **W**ever '14]



$$q_1=xp_1, \quad q_2=xp_2, \quad q_3=p_{123}-xp_{12}, \quad q_4=p_{123},$$
$$p_i^2=0, \qquad s_{12}:=p_{12}^2, \qquad s_{23}:=p_{23}^2, \qquad q:=p_{123}^2,$$
$$S=s_{12}x^2, \qquad T=q-(s_{12}+s_{23})x,$$
$$M_3^2=(1-x)(q-s_{12}x), \qquad M_4^2=q$$

- xyz-parameterization

[**HMS**: J. M. **H**enn, K. **M**elnikov, V. A. **S**mirnov '14;
GMT: T. **G**ehrmann, A. von **M**anteuffel, L. **T**ancredi '15]



$$\mathrm{P12}: \ p_1=-q_3, \ p_2=-q_4, \ p_3=q_1, \ p_4=q_2$$
$$\mathrm{P13}: \ p_1=-q_3, \ p_2=q_1, \ p_3=-q_4, \ p_4=q_2$$
$$\mathrm{P23}: \ p_1=q_2, \ p_2=-q_4, \ p_3=-q_3, \ p_4=q_1$$

$$s=\bar{m}^2(1+\bar{x})^2, \qquad p_3^2=\bar{m}^2\bar{x}^2(1-\bar{y}^2)$$
$$t=-\bar{m}^2\bar{x}((1+\bar{y})(1+\bar{x}\bar{y})-2\bar{z}\bar{y}(1+\bar{x})) \quad p_4^2=\bar{m}^2(1-\bar{x}^2\bar{y}^2)$$

$$s=m^2(1+x)(1+xy), \ t=-m^2xz, \ p_3^2=m^2, \ p_4^2=m^2x^2y$$

Planar double boxes

- The parameterizations are different, but they describe the same physics
- Example: mapping valid for PTW to HMS $q<s12$

P12-P13:

$$\bar{x} = -1 + (s12x)/q$$

$$\bar{y} = (q - qx)/(q - s12x)$$

$$\bar{z} = (q - qx + s23x)/(q - s12x)$$

P23:

$$\bar{x} = -1 + (s12x)/q$$

$$\bar{y} = (q - qx)/(q - s12x)$$

$$\bar{z} = (q - (s12 + s23)x)/(q - s12x)$$

Planar double boxes

- Different parameterizations, different solution strategies ⟹ analytic results looks differently (clearly they are equivalent)
- Sample of results for P12, as function of GPs

$$G[1,x]\left(G\left[\frac{s12+s23}{s12},x\right]\left(-\frac{G\left[0,\frac{q}{s12},x\right]}{s12^2}+\frac{G\left[0,\frac{q}{q-s23},x\right]}{s12^2}\right)+G\left[0,1-\frac{s23}{q}\right]\left(-\frac{36\,G\left[0,\frac{q}{s12},x\right]}{s12^2}-\frac{36\,G\left[\frac{q}{s12},\frac{q}{q-s23},x\right]}{s12^2}+\frac{6\,G\left[\frac{q}{s12},\frac{s12+s23}{s12},x\right]}{s12^2}\right)+\ldots\right.$$

x-parametrization

$$G[-1,x]\,(i\pi-G[0,y]+2\,G[0,z])+i\pi\,G\left[-\frac{1}{y},x\right]-i\pi\,G\left[-\frac{1}{z},x\right]+G[0,z]\left(2\,G\left[-\frac{1}{y},x\right]-G\left[-\frac{1}{z},x\right]-G\left[-\frac{z}{y},x\right]\right)-\ldots$$

xyz-parametrization

Similar situation for Non-planar double boxes

Computational timing

- We can evaluate the Master Integrals by **`Mathematica`** interfaced with **`GiNac`** (just for timing comparison).
- We test some “random” phase space poins:

$$\{ S , T , p_3^2 , p_4^2 \}$$

phys1a = {5.687, -0.243333, 0.017, 5.}

phys1b = {8., -2., 3., 1.}

phys1c = {130.05, -20.85, 66.7, 5.5}

phys2a = {5.64667, -1.52222, 0.213333, 3.}

phys2b = {3., -0.666667, 0.5, 1.}

phys2c = {226.875, -25.1667, 175.125, 2.33333}

Computational timing

- Speed comparison for different solutions in different points (all times are expressed in seconds)

P12-29 G111111100 (scalar)			
point	HMS	PTW	GMT trad
phys1a	0,3	1600	18
phys1b	8,3	260	1,1
phys1c	11	200	8,4
phys2a	0,63	470	14
phys2b	3,1	630	8,7
phys2c	11	210	14

- PTW is in general much slower than other solutions
- There are large and almost random differences in computational times between different **libraries** but also in different **points**. Those difference are partially related to computing MI in edges of phase space, but also on explicit expressions as letters inside GPs

INSERT: GPs evaluation in `GiNaC`

- Definition $$G(z_1,\ldots,z_k;y) \;=\; \int\limits_0^y \frac{dt_1}{t_1-z_1}\int\limits_0^{t_1}\frac{dt_2}{t_2-z_2}\ldots\int\limits_0^{t_{k-1}}\frac{dt_k}{t_k-z_k}.$$

- Evaluation: algorithm dependent on the input values

[J. Vollinga, S. Weinzierl '04]

- $|y| \leq |z_j|$ for all j $\quad y/z_1 \neq 1$

$$G(z_1,\ldots,z_k;y) = \sum_{j_1=1}^{\infty}\cdots\sum_{j_k=1}^{\infty}\frac{1}{j_1+\ldots+j_k}\left(\frac{y}{z_1}\right)^{j_1}\frac{1}{j_2+\ldots+j_k}\left(\frac{y}{z_2}\right)^{j_2}\cdots\frac{1}{j_k}\left(\frac{y}{z_k}\right)^{j_k}$$

- if $|y| > |z|$ $\quad G(z,y) = G(y,z) - G(0,z) + I\pi G(0,y)$
  in general if $|y| > |z_i|$ $\quad G(\ldots z_i \ldots, y) = \sum(\ldots)G(\ldots)$

Analytic continuation

- slow convergence if |y/z|~1 (Holder convolution or other strategies)

$$G(z_1,\ldots,z_w;1) \;=\; \sum_{j=0}^{w}(-1)^j\, G\left(1-z_j,1-z_{j-1},\ldots,1-z_1;1-\frac{1}{p}\right) G\left(z_{j+1},\ldots,z_w;\frac{1}{p}\right)$$

"Fast" $G(1,2,3;0.001) \rightarrow 5\cdot 10^{-4}\text{sec.}$ "Slow" $G(1,2,3;1.999) \rightarrow 4\cdot 10^{-2}\text{sec.}$

Computational timing

- Spee ... differe ... points (al ... es are expressed in ... onds)

An. cont. done by GiNaC

An. cont. done by `fibrationBasis()` in `Hyperint` code [E. Panzer '15]

P12-29 G111111100 (scalar)				
point	**HMS**	**PTW**	**PWT + fibBas**	**GMT trad**
phys1a	0,3	1600	24	18
phys1b	8,3	260	1,4	1,1
phys1c	11	200	0,84	8,4
phys2a	0,63	470	3,0	14
phys2b	3,1	630	5,8	8,7
phys2c	11	210	0,86	14

- PTW solution is native in Euclidean region of phase space. We are testing the library in physical space analytic continuation is needed

Computational timing

GPs up to weigth 4

- Speed comparison for different solutions i… points (all times are expressed in seconds)

Log, Li_n and Li_{22}

P12-29 G111111100 (scalar)					
point	**HMS**	**PTW**	**PWT + fibBas**	**GMT trad**	**GMT opt**
phys1a	0,3	1600	24	18	0,27
phys1b	8,3	260	1,4	1,1	0,16
phys1c	11	200	0,84	8,4	0,14
phys2a	0,63	470	3,0	14	0,17
phys2b	3,1	630	5,8	8,7	0,17
phys2c	11	210	0,86	14	0,19

- In GMT optimized solution is typically faster than other solutions

Computational timing

GPs up to weigth 4

Log, Li_n and Li_{22}

- Speed comparison for different solutions in points (all times are expressed in seconds)

P12-29 G111111100 (scalar)

P12-30 G1111111m0					
point	**HMS**	**PTW**	**PWT + fibBas**	**GMT trad**	**GMT opt**
phys1a	0,44	1600	25	63	1,1
phys1b	12	300	1,5	4,2	0,41
phys1c	17	230	0,86	31	0,58
phys2a	0,72	520	3,2	47	0,55
phys2b	4,8	670	6,1	35	0,66
phys2c	17	240	0,88	52	0,69

- In GMT optimized solution is typically faster than other solutions, but not always: Holder convolution is mapping back Li_n and Li_{22} to GP up to weight 4!!!

$$\mathrm{Li}_{22}(x,y) = \mathrm{G}\left(0, 1/x, 0, 1/(xy); 1\right) = G(1-1/(xy), 1, 1-1/x, 1; 1/p) + \ldots$$

Main message

- Once we have analytic solution for some MI we prefer to simplify it in order to achieve a fast evaluation
- It is NOT obvious what is the best expression, not always expression made of few GPs is the fastest expression to be evaluated.
- MAYBE: according to different regions and limits in the phase space, there could be differently optimized expressions
- Another possibility: Log, $\mathrm{Li_n}$ and Li_{22} basis of functions seems promising, but computational algorithm needs improvements in order to avoid mapping back to GPs

Fast $\mathrm{Li}_n(z)$ evaluation: Crandal algorithm

Given any complex number z, we can evaluate $\mathrm{Li}_n(z)$ by four cases:

- If $z=1$, we just plug special value $\mathrm{Li}_n(1) = \zeta(n)$
- If $|z| < 1/2$ evaluate by definition $\mathrm{Li}_n(z) = \sum_{k=1}^{\infty} z^k/k^n$
- If $|z| > 2$ use inversion relation

$$\mathrm{Li}_n(1/z) = (-1)^n\left(-\mathrm{Li}_n(z) + \frac{(2\pi i)^n}{n!} B_n\left(\frac{\log z}{2\pi i}\right) + 2\pi i \Theta(z) \frac{\log^{n-1} z}{(n-1)!}\right)$$

where B_n is the standard Bernoulli polynomial
$\Theta(z) := 1$, if $\Im(z) < 0$ or $z \in [1, \infty]$, else $\Theta = 0$

Fast $\mathrm{Li}_n(z)$ evaluation: Crandal algorithm

Given any complex number z, we can evaluate $\mathrm{Li}_n(z)$ by four cases:

- If $z=1$, we just plug special value $\mathrm{Li}_n(1) = \zeta(n)$
- If $|z| < 1/2$ evaluate by definition $\mathrm{Li}_n(z) = \sum_{k=1}^{\infty} z^k/k^n$
- If $|z| > 2$ use inversion relation

$$\mathrm{Li}_n(1/z) = (-1)^n \left(-\mathrm{Li}_n(z) + \frac{(2\pi i)^n}{n!} B_n \left(\frac{\log z}{2\pi i} \right) + 2\pi i \Theta(z) \frac{\log^{n-1} z}{(n-1)!} \right)$$

- Else ($|z| \sim 1$) use log expansion:

$$\mathrm{Li}_n(z) = \sum_{m=0}^{\infty}{}' \frac{\zeta(n-m)}{m!} \log^m z + \frac{\log^{n-1} z}{(n-1)!} \left(H_{n-1} - \log(-\log z) \right)$$

Log expansion of $\mathrm{Li}_n(z)$ around one nicely captures its singular structure. Furthermore it is efficient formula for numerical evaluation

Fast $\mathrm{Li}_{22}(x,y)$ evaluation: Crandal-style algorithm

Given any complex numbers (x,y), we can evaluate $\mathrm{Li}_{22}(x,y)$ by four cases:

- If x=y=1, we just plug special value $\mathrm{Li}_{22}(1,1) = \pi^4 / 120$
- If $|x|$ and $|xy|$ are "small" evaluate by definition

$$\mathrm{Li}_{2,2}(x,y) = \sum_{j_1=1}^{\infty}\sum_{j_2=1}^{\infty} \frac{x^{j_1}}{(j_1+j_2)^2}\frac{(xy)^{j_2}}{j_2^2}$$

- If $|x|$ and $|xy|$ are "large" use inversion relations, for example:

$$\mathrm{Li}_{22}(1/x,1/y) = \mathrm{Li}_{22}(y,x) + \frac{31\pi^4}{360} + 3\mathrm{Li}_4(y) - 3\mathrm{Li}_4(x) + \ldots$$

The actual numerical values for "small" and "large" will be defined empirically after numerical implementation in order to minimize the computational cost. Approximatively ½ and 2.

Fast $\mathrm{Li}_{22}(x,y)$ evaluation: Crandal-style algorithm

Given any complex numbers (x,y), we can evaluate $\mathrm{Li}_{22}(x,y)$ by four cases:

- If $x=y=1$, we just plug special value $\mathrm{Li}_{22}(1,1) = \pi^4/120$
- If $|x|$ and $|xy|$ are "small" evaluate by definition

$$\mathrm{Li}_{2,2}(x,y) = \sum_{j_1=1}^{\infty}\sum_{j_2=1}^{\infty} \frac{x^{j_1}}{(j_1+j_2)^2}\frac{(xy)^{j_2}}{j_2^2}$$

- If $|x|$ and $|xy|$ are "large" use inversion relations, for example:

$$\mathrm{Li}_{22}(1/x,1/y) = \mathrm{Li}_{22}(y,x) + \frac{31\pi^4}{360} + 3\mathrm{Li}_4(y) - 3\mathrm{Li}_4(x) + \ldots$$

- Else ($|x|\sim 1$) use generalization of log expansion (Li_2 function is needed):

$$\mathrm{Li}_{22}(x,y) \simeq \mathrm{Polynomia}\left(\{x,y,\log(\{x,y\}),\mathrm{Li}_2(\{x,y\})\}\right)$$

Work in progress: we have analytic expressions, but further simplifications has to be done.

Summary

- Not all the needed two-loop Master Integrals are known yet, but it is already time to start thinking how to make a library of them
- In particular it is important trying to understand how to simplify them, but first of all we must understand what does it mean: “simplified”
- Large set of MI can be expressed in Log, Li_{n} and Li_{22} basis (not all, some require Elliptic functions): faster algorithms are welcome!
- There is room to improve the computational algorithms, soon news concerning Li_{22} will come!

Thanks!