Top quark pairs at two loops, multiple polylogarithms and Reduze2

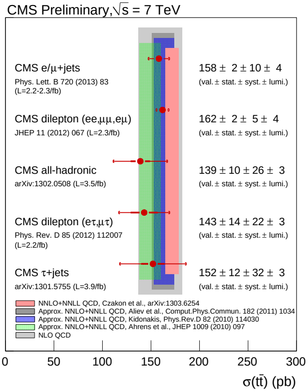
Andreas v. Manteuffel

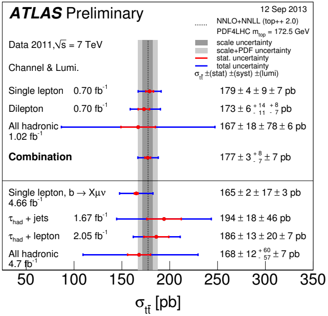
JOHANNES GUTENBERG UNIVERSITÄT MAINZ

PRISMA

RADCOR 2013
Lumley Castle
22-27 September

Top pair production at the LHC





- **LHC precision** below NLO accuracy already
- **full** ***WWbb*** at NLO: see talks by S. Kallweit, J. Schlenk
- **mass schemes**: see talk by Dowling
- recent **threshold/resummations**, **approx. NNLO**:
  Beneke, Falgari, Klein, Piclum, Schwinn, Ubiali, Yan; Aliev, Lacker, Langenfeld, Moch, Uwer, Vogt, Wiedermann; Cacciari, Czakon, Mangano, Mitov, Nason; Ahrens, Ferroglia, Neubert, Pecjak, Yang; Kidonakis, see talk by A. Penin
- **full NNLO** total σ (numerical methods):
  Bärnreuther, Czakon, Fiedler, Mitov '08-'13, see talk by M. Czakon

Analytic NNLO calculation

motivation for analytic approach:

- understand structure
- robust, fast, precise numerical evaluations
- cross-check

ingredients:

- $\boxed{\textbf{VV}}$: two-loop ME for $q\bar{q} \to t\bar{t}$ and $gg \to t\bar{t}$
  leading N_c, (light) fermionic: Bonciani, Ferroglia, Gehrmann, Maitre, AvM, Studerus '08-'13
  poles: Ferroglia, Neubert, Pecjak, Yang '09
  small mass: Czakon, Mitov, Moch '06
  one-loop^2: Kniehl, Körner, Merebashvili, Rogal '05-'08, Anastasiou, Aybat '08
- $\boxed{\textbf{RV}}$: one-loop ME for $t\bar{t}$ + 1 parton

  Dittmaier, Uwer, Weinzierl '07, '09; Bevilacqua, Czakon, Papadopoulos, Worek '10, '11; Melnikov, Schulze '10
- $\boxed{\textbf{RR}}$: tree level ME for $t\bar{t}$ + 2 partons
- $\boxed{\textbf{subtraction terms}}$: up to 2 unresolved partons needed

  Kosower '97; Weinzierl '03; Frixione, Grazzini '04; Gehrmann-De Ridder, Gehrmann, Glover '05; Somogyi, Trocsanyi, Del Duca '06-'08; Abelof, Boughezal, Daleo, Dekkers, Gehrmann, Gehrmann-De Ridder, Maitre, Luisoni, Ritzmann '06-'12; Gehrmann-De Ridder, Glover, Pires '10-'11; Bernreuther, Bogner, Dekkers '11; Bierenbaum, Czakon, Mitov '11

Method of calculation

complexity:

- 2 independent ratios of scales, tensor rank 4
- 256 master integrals (w/products, wo/crossings)
- advanced mathematical functions

this talk: light-fermionic two-loop corrections to $gg \to t\bar{t}$

Recipe

1. generate **Feynman diagrams** with `QGRAF` by Nogueira
2. build **interference** terms with `Reduze 2`
3. **reduce** scalar integrals to masters via IBPs with `Reduze 2`
4. **renormalize**: $\overline{MS}$, pole mass
5. **solve masters** with differential equations
6. **optimisation of functional basis** for multiple polylogs

$\Rightarrow$ **analytic result** in terms of multiple polylogarithms,
allows fast **numerical evaluation**, **expansions**, ...

```
    ____           __              
   / __ \___  ____/ /_  ______  ___ 
  / /_/ / _ \/ __  / / / /_  / / _ \
 / _, _/  __/ /_/ / /_/ / / /_/  __/
/_/ |_|\___/\__,_/\__,_/ /___/\___/ 
```

Reduze 2 - Distributed Feynman Integral Reduction

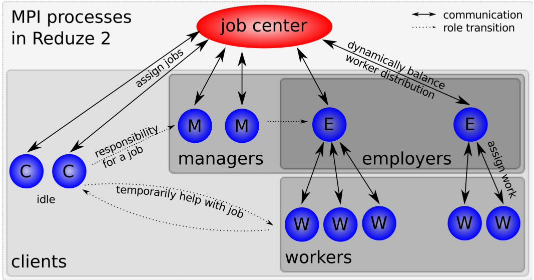
A.v.M., Studerus

arXiv:1201.4330

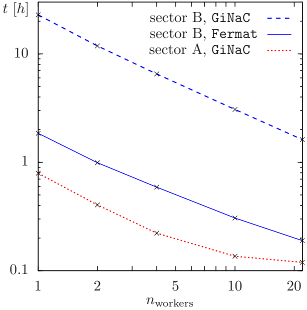
http://projects.hepforge.org/reduze

uses GiNaC *[Bauer, Frink, Kreckel] and* Fermat *[Lewis]*

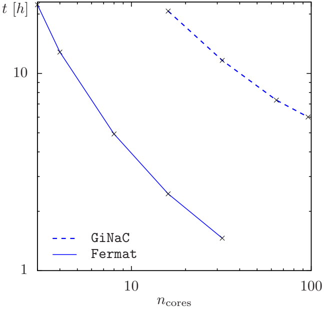
Reduze 2: distributed Laporta algorithm



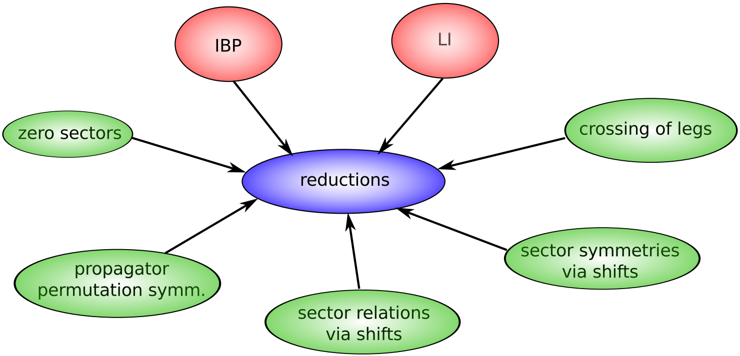
reduction of single sector:



reduction of sector selection:



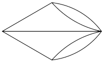
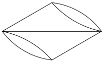
Removing ambiguities for integrals



Reduze 2: sectors, graphs and matroids

$$\begin{array}{ccc} \boxed{\textbf{integral A}\text{ (propagators)}} & \longleftarrow\quad \exists\text{shift ?}\quad\longrightarrow & \boxed{\textbf{integral B}\text{ (propagators)}} \\ \big\downarrow \textit{construct} & & \big\downarrow \textit{construct} \\ \boxed{\textbf{graph A}} & \longleftarrow\quad \text{isomorphic ?}\quad\longrightarrow & \boxed{\textbf{graph B}} \end{array}$$

- **problem**: graphs not unique !

e.g.:  ⟵ same propagators but not isomorphic ⟶ 

- **solution**: select unique representative by allowing for **twists**

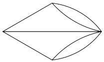
based on: $\boxed{\textbf{first Symanzik polynomials } \mathcal{U} \text{ isomorphic} \Leftrightarrow \text{graphs isomorphic } \textbf{up to twists}}$

theorem by Bogner, Weinzierl (2010), proof based on Whitney's theorem for isomorphisms of graph matroids

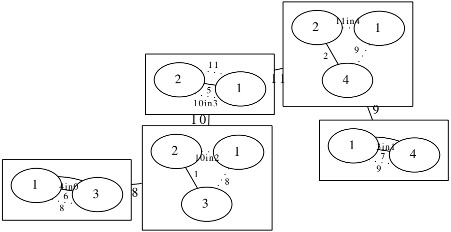
Algorithm: shift finder

1. generate graph for sector
2. colour edges according to masses
3. connect external legs with a new vertex
4. decompose into triconnected components (Hopcroft, Tarjan '73; Gutwenger, Mutzel '01)
5. minimize graph by twists
6. check for graph isomorphism (McKay '81)

example:



$\longrightarrow$



tree of triconnected components
(dashed "virtual edges" mark positions for Tutte twists)

Application example

auto-generated shifts for non-planar double box family:

figures : less

Datei Bearbeiten Ansicht Verlauf Lesezeichen Einstellungen Hilfe


```
---
sectormappings:
  name: box2n
  zero_sectors:
    t=0: [0]
    t=1: [1, 4, 16, 64, 256]
    t=2: [3, 5, 6, 12, 17, 18, 20, 24, 48, 65, 66, 68, 72, 80, 96, 192, 257, 260, 272, 320]
    [ ... truncated ]
  sectors_without_graph:
    t=3: [22, 28, 52, 82, 84, 88, 104, 112, 276]
    t=4: [23, 29, 30, 53, 54, 60, 83, 85, 86, 89, 90, 92, 105, 106, 108, 113, 114, 116, 120, 21
    t=5: [31, 55, 61, 87, 91, 93, 94, 107, 109, 110, 115, 117, 118, 121, 122, 124, 211, 213, 21
    [ ... truncated ]
  sector_relations:
    3: [[box2p, 3], [[k1, k1], [k2, k2]]]
    6: [[box2p, 3], [[k1, -p1+k2], [k2, k1]]]
    7: [[box2p, 11], [[k1, k1-p1], [k2, k2]]]
    12: [[box2p, 3], [[k1, k1-p1], [k2, k2-p2]]]
    13: [[box2p, 11], [[k1, k1-p1], [k2, k2-p2]]]
    15: [[box2p, 58], [[k1, -p1+k2], [k2, k1-p1-p2]]]
    18: [[box2p, 3], [[k1, k2+p3], [k2, k1]]]
    19: [[box2px13x24, 11], [[k1, k1+p3], [k2, k2]]]
    24: [[box2p, 3], [[k1, k2+p3], [k2, k1-p2]]]
    25: [[box2px13x24, 11], [[k1, k1+p3], [k2, k2-p2]]]
    26: [[box2px12x34, 11], [[k1, k2+p3], [k2, k1-p2]]]
    27: [[box2px13, 58], [[k1, k2+p3], [k2, k1-p2+p3]]]
    [ ... truncated ]
  crossed_sector_relations:
    x12:
      207: [[box2n, 207], [[k1, -k2-p2], [k2, -k1-p1]]]
      335: [[box2n, 335], [[k1, -k2-p2], [k2, -k1-p1]]]
      463: [[box2n, 463], [[k1, -k2-p2], [k2, -k1-p1]]]
    x123:
      335: [[box2n, 335], [[k1, -k2-p2], [k2, k1+p1-k2]]]
    x1234:
      335: [[box2nx14x23, 335], [[k1, -k1+p1+k2-p3]]]
    x124:
      335: [[box2nx14x23, 335], [[k1, -k1+p1+k2-p3], [k2, -k1+p1+p2-p3]]]
    x1243:
      335: [[box2nx12x34, 335], [[k2, k1-k2+p2]]]
  sector_symmetries:
    207:
      - [[k1, -k1-p1], [k2, -k2-p2]]
      - [[k1, -k2-p2], [p1, p2], [k2, -k1-p1], [p2, p1], [p3, p1+p2-p3]]
      - [[k1, k2], [p1, p2], [k2, k1], [p2, p1], [p3, p1+p2-p3]]
    463:
      - [[k1, k2], [p1, p2], [k2, k1], [p2, p1], [p3, p1+p2-p3]]
(END)
```


figures : less

New features in Reduze 2

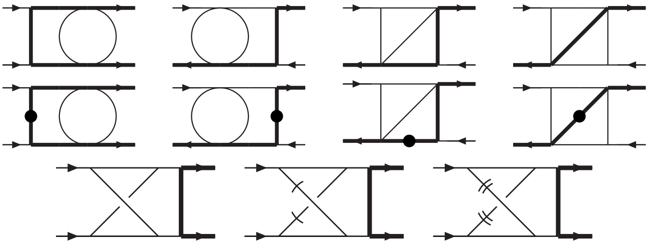
new features in upcoming release:

- **bilinear propagators** (shift finder etc.)

$$\frac{1}{q_1 q_2 - m^2}$$

  used extensively for 3-loop heavy flavour Wilson coefficients in DIS, see talk by A. de Freitas
- **cut propagators**
- **Symanzik polynomial** based integral matcher (see also Pak (2011))
- **family finder**

New master integrals



(thick line: massive, thin line: massless, dot: squared propagator, ticks: numerator)

AvM, Studerus [arXiv:1306.3504]

- solved with method of **differential equations**
  see talks by L. Tancredi, J. Henn
- constants: regularity, symmetry, Mellin-Barnes
- used `MB.m` by Czakon '05, planar: `AMBRE` by Gluza, Kajda, Riemann '07 + Yundin '10
- checks: `SecDec 2.1` by Borowka, Heinrich '13
- crossed+uncrossed subtopos: **non-trivial functional identities**, explicite imaginary parts

Multiple polylogarithms

Remiddi, Gehrmann; Goncharov:

Definition of Goncharov's multiple polylogarithms (GPLs)

$$G(\vec{0}_n; x) = \frac{1}{n!} \log^n(x),$$

$$G(a; x) = \int_0^x dt\ f_a(t), \qquad \text{for } a \neq 0$$

$$G(a, \vec{b}; x) = \int_0^x dt\ f_a(t) G(\vec{b}; t), \qquad \text{for } a \neq 0$$

with weight functions for (complex) weight a:

$$f_a(x) = \frac{1}{x - a}$$

- $G(a \neq 0; x) = \log\left(\frac{a-x}{a}\right)$, $G(\vec{0}_{n-1}, 1; x) = Li_n(x)$, $G(\vec{0}_n, \vec{1}_p; x) = S_{n,p}(x)$
- for weights $0, 1, -1$, GPLs specialize to harmonic polylogarithms (HPLs), Remiddi, Vermaseren (1999)
- shuffle algebra

Generalised weights

generalised integration measure

$$\frac{\mathrm{d}t}{t-w} \to \frac{f'(t)}{f(t)}\mathrm{d}t = \mathrm{d}\ln(f(t))$$

defines **generalised weight** $[f(o)]$:

$$G([f(o)], w_2, \ldots, w_n; x) = \int_0^x \mathrm{d}t \frac{f'(t)}{f(t)} G(w_2, \ldots, w_n; t)$$

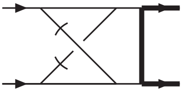
example:

$$G([o^2+1]; x) = \int_0^x \mathrm{d}t \frac{2t}{t^2+1} = \int_0^x \mathrm{d}t \frac{1}{t-i} + \int_0^x \mathrm{d}t \frac{1}{t+i} = G(i; x) + G(-i; x)$$

root-free symbol:

$$\mathcal{S}\left(G([f(o)]; x)\right) = \mathcal{S}\left(\ln\left(f(x)\right)\right) = f(x)\,.$$

- see: integrated jet mass distribution in SCET, AvM, Schabinger, Zhu [arXiv:1309.3560]
- related: Ablinger, Blümlein, Schneider '11 (cyclotomic polylogs)



$$= \frac{x^2}{m^4(1-x)^2(1-x+x^2+xy)} \sum_{i=-4}^{0} I_i \epsilon^i + \mathcal{O}(\epsilon)\,,$$

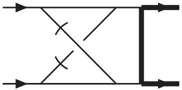
$$I_{-4} = \frac{7}{384}$$

$$\begin{aligned}
I_{-3} = &-\frac{5}{192} G(-(1-x+x^2)/x;y) + \frac{1}{64} G(-1;y) - \frac{5}{192} G([1-o+o^2];x) - \frac{1}{16} G(1;x) \\
&+ \frac{11}{192} G(0;x) - \frac{5}{192} i\pi
\end{aligned}$$

$$\begin{aligned}
I_{-2} = &+\frac{1}{192} G(-(1-x+x^2)/x;y)^2 - \frac{1}{32} G(-(1-x+x^2)/x;y) G(-1;y) \\
&+ \frac{1}{96} G(-(1-x+x^2)/x;y) G([1-o+o^2];x) + \frac{1}{8} G(-(1-x+x^2)/x;y) G(1;x) \\
&- \frac{7}{96} G(-(1-x+x^2)/x;y) G(0;x) - \frac{1}{64} G(-1;y)^2 - \frac{1}{32} G(-1;y) G([1-o+o^2];x) \\
&+ \frac{1}{32} G(-1;y) G(0;x) + \frac{1}{192} G([1-o+o^2];x)^2 + \frac{1}{8} G([1-o+o^2];x) G(1;x) \\
&- \frac{7}{96} G([1-o+o^2];x) G(0;x) + \frac{1}{16} G(1;x)^2 - \frac{3}{16} G(0;x) G(1;x) + \frac{1}{12} G(0;x)^2 \\
&+ \frac{1}{96} i\pi G(-(1-x+x^2)/x;y) - \frac{1}{32} i\pi G(-1;y) + \frac{1}{96} i\pi G([1-o+o^2];x) \\
&+ \frac{1}{8} i\pi G(1;x) - \frac{7}{96} i\pi G(0;x) - \frac{35}{1152} \pi^2
\end{aligned}$$

$I_{-1} = \ldots$ (very lengthy)

where $x \equiv \frac{\sqrt{1-4m^2/s}-1}{\sqrt{1-4m^2/s}+1}$, $y = -t/m^2$



$$= \frac{x^2}{m^4(1-x)^2(1-x+x^2+xy)} \sum_{i=-4}^{0} I_i \epsilon^i + \mathcal{O}(\epsilon)\,,$$

simplified version (via symbols):

$$I_{-4} = \frac{7}{384}$$

$$I_{-3} = -\frac{1}{32} \ln(y_1+z_1) + \frac{1}{64} \ln y_1 - \frac{5}{192} \ln z_1 + \frac{1}{32} i\pi$$

$$I_{-2} = \frac{1}{64} \ln^2(y_1+z_1) - \frac{1}{64} \ln^2 y_1 + \frac{1}{192} \ln^2 z_1 - \frac{1}{32} \ln y_1 \ln z_1 + \frac{1}{16} \ln z_1 \ln(y_1+z_1)$$
$$- \frac{1}{32} i\pi \ln(y_1+z_1) - \frac{1}{16} i\pi \ln z_1 - \frac{47}{1152}\pi^2$$

$$I_{-1} = -\frac{1}{32} G\left(1,0,0;\frac{y_1 z_1}{y_1+z_1}\right) - \frac{1}{32} i\pi G\left(1,0;\frac{y_1 z_1}{y_1+z_1}\right) + \frac{1}{16}\ln^2 y_1 \ln z_1 - \frac{1}{16} \ln z_1 \ln^2(y_1+z_1)$$
$$+ \frac{1}{8} i\pi \ln z_1 \ln(y_1+z_1) + \frac{5}{96}\pi^2 \ln(y_1+z_1) - \frac{1}{24}\pi^2 \ln y_1 - \frac{1}{144}\ln^3 z_1 + \frac{29}{288}\pi^2 \ln z_1$$
$$- \frac{55\zeta(3)}{192} - \frac{5}{96} i\pi^3$$

new functional basis: $\mathrm{Li}_3\left(\frac{y_1 z_1}{y_1+z_1}\right)$, $\mathrm{Li}_2\left(\frac{y_1 z_1}{y_1+z_1}\right)$, $\ln(y_1+z_1)$, $\log y_1$, $\ln z_1$
where $y_1 \equiv -t/m^2+1$, $z_1 \equiv -u/m^2+1$

similarly for poles of corner integral, instead of 28 (65 expanded) GPLs

Algorithms for Multiple Polylogarithms

main algorithms:

1. **normal form for specific arguments**
   - independent of symbol calculus
   - uses Vollinga, Weinzierl '04 for numerical evaluation, fits constants
2. **coproduct based normal form** for general choice of basis
   - based on Goncharov '02, Brown '11, Duhr '12, Duhr, Gangl, Rhodes '11
   - handles generalised weights
   - identifies products (e.g. $G(0,1;x) + G(1,0;x) \to G(0;x)G(1;x)$)
   - matches irreducible factors *at symbol level*
   - uses Vollinga, Weinzierl '04 for numerical evaluation, fits constants
3. **construct new basis** with desired properties
   - based on Duhr, Gangl, Rhodes '11, apply to generalised weights

Definition of symbol map

Let G be a multiple polylogarithm with

$$\mathrm{d}G = \sum\nolimits_i \hat{G}_i \,\mathrm{d}\ln(R_i)$$

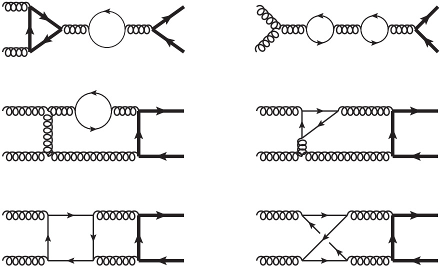
where R_i is a rational function of the polylog arguments. The **symbol map** $\mathcal{S}$

$$\mathcal{S}(G) = \sum\nolimits_i \mathcal{S}(\hat{G}_i) \otimes R_i \,,$$

associates a tensor with the polylogarithm.

- $N = 4$ remainder function simplification: Goncharov, Spradlin, Vergu, Volovich ('10)
- extension to **coproduct** formalism: *const* $\times$ polylog
- simple example: **supplementary slides**

Light fermionic two-loop corrections to $gg \to t\bar{t}$



+ more

Bonciani, Ferroglia, Gehrmann, AvM, Studerus [`arXiv:1309.4450`]

gg **channel**: 789 two-loop diagrams (+ ghost init.)

$$
\begin{aligned}
\mathbf{2\,Re}\left\langle \mathcal{M}^{(0)} | \mathcal{M}^{(2)} \right\rangle = 2C_F N_c \Big(& N_c^3 \mathbf{A} + N_c \mathbf{B} + \frac{1}{N_c}\mathbf{C} + \frac{1}{N_c^3}\mathbf{D} \\
& + N_c^2 n_l \mathbf{E_l} + n_l \mathbf{F_l} + \frac{n_l}{N_c^2}\mathbf{G_l} + N_c n_l^2 \mathbf{H_l} + \frac{n_l^2}{N_c}\mathbf{I_l} \\
& + N_c^2 n_h \mathbf{E_h} + n_h \mathbf{F_h} + \frac{n_h}{N_c^2}\mathbf{G_h} + N_c n_h^2 \mathbf{H_h} + \frac{n_h^2}{N_c}\mathbf{I_h} \\
& + N_c n_l n_h \mathbf{H_{lh}} + \frac{n_l n_h}{N_c}\mathbf{I_{lh}} \Big)
\end{aligned}
$$

Structure of result and optimisation of basis

(A) initial form given by masters:
evaluation time $\mathcal{O}(\text{min})$, not very stable

(B) primary normal form:

$$G(\ldots;y), \quad \text{weights} \in \left\{-1, 0, -\frac{1}{x}, -x, -\frac{(1+x^2)}{x}, -\frac{(1-x+x^2)}{x}\right\}$$

$$G(\ldots;x), \quad \text{weights} \in \left\{-1, 0, 1, [1+o^2], [1-o+o^2]\right\}$$

note: jump in transcendentality weight from 2 for ϵ^{-1} to 4 for ϵ^0 for E_l, F_l, G_l

(C) optimised functional basis:

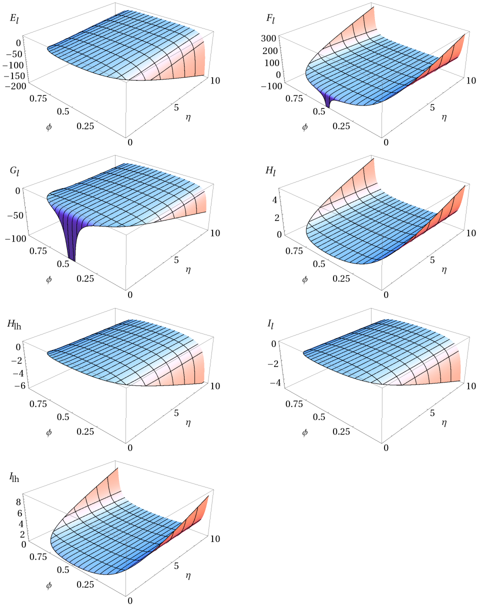
choose real valued $\ln R_1$, $\mathrm{Li}_n(R_1)$, $\mathrm{Li}_{2,2}(R_1, R_2)$ with
$R_i \in \left\{\pm x,\ \pm x^2,\ -\frac{1}{y},\ -y,\ -\frac{y}{x},\ -x(x+y),\ \frac{x+y}{y},\ -\frac{x+z(x,y)}{x+y}, \ldots\right\}$

structure:

- no spurious letters in symbol
- all generalised weights eliminated (multivariate cancelations)

numerical evaluation:

- **faster by factor 15**
- **more stable**
- eval time $\mathcal{O}(1s)$



$\eta = s/(4m^2) - 1$, $\phi = -(t - m^2)/s$

Conclusions

- $t\bar{t}$: analytic two-loop corrections
  - leading N_c + (light) fermionic
  - massive non-planar double box (update to appear, Henn, AvM, Smirnov)
  - expansions in kinematical limits
- Reduze 2: open source tool
  - parallelized reductions of Feynman integrals
  - graph matroid algorithm: determine shifts between sectors or diagrams
  - new features: bilinear propagators, ...
  - determination of integral basis: talk by R. Schabinger
- multiple polylogarithms:
  - highly automated treatment of functional identities (coproduct etc.)
  - generalised weights eliminate weights with roots
  - optimized basis for fast and stable numerical evaluation
  - next big step: iterated integrals beyond multiple polylogs

Supplementary Slides

6 Available jobs in Reduze 2

7 Definition of matroid

8 Rules for symbol

9 Example for symbol calculus

10 Coproduct: systematic approach

Available jobs in Reduze 2

andreas : bash

Datei Bearbeiten Ansicht Verlauf Lesezeichen Einstellungen Hilfe

```
andreas@chili:~$ reduze -h jobs

List of available job types:

apply_crossings:                  Generates reduction results for crossed sectors.
cat_files:                        Concatenates files.
collect_integrals:                Collects all integrals appearing in the input file.
compute_diagram_interferences:    Computes interferences of diagrams.
compute_differential_equations:   Computes derivatives of integrals wrt invariants.
export:                           Exports to FORM, Mathematica or Maple format.
find_diagram_shifts:              Matches diagrams to sectors via graphs.
find_diagram_shifts_alt:          Matches diagrams to sectors via combinatorics.
generate_identities:              Generates identities like IBPs for given seeds.
generate_seeds:                   Generates integrals from a sector.
insert_reductions:                Inserts reductions in expressions.
normalize:                        Simplifies linear combinations and equations.
print_reduction_info_file:        Analyzes reductions in a file.
print_reduction_info_sectors:     Analyzes reductions available for sectors.
print_sector_info:                Prints diagrams and other information for sectors.
reduce_files:                     Reduces identities in given files.
reduce_sectors:                   Reduces integrals from a selection of sectors.
run_reduction:                    Low-level job to run a reduction.
select_reductions:                Selects reductions for integrals.
setup_sector_mappings:            Finds shifts between sectors via graphs.
setup_sector_mappings_alt:        Finds shifts between sectors via combinatorics.
sum_terms:                        Sums terms.
test:                             Empty job template.
verify_same_terms:                Verifies two files contain the same terms.

andreas@chili:~$
```

andreas : bash

detailed description: see builtin on-line help

Definition of matroid

A matroid is a pair $(E, \mathcal{I})$ where E finite ground set, $\mathcal{I}$ collection of subsets of E, the **"independent sets"**, and

- $\emptyset \in \mathcal{I}$
- if $I \in \mathcal{I}$ and $I' \subset I$ then also $I' \in \mathcal{I}$
- if $I_1, I_2 \in \mathcal{I}$, $|I_1| < |I_2|$ then $\exists e \in I_2 - I_1$ with $I_1 \cup \{e\} \in \mathcal{I}$

- generalises notion of **linear dependency**
- graph matroid: dependencies of **edges**
- application to Feynman graph: propagators relevant, no reference to vertices

propagators of two vacuum diagrams related by shift $\Leftrightarrow$ graph matroids isomorphic

Definition of symbol map

Let G be a multiple polylogarithm with

$$\mathrm{d}G = \sum_i \hat{G}_i \, \mathrm{d}\ln(R_i)$$

where R_i is a rational function of the polylog arguments. The **symbol map** $\mathcal{S}$

$$\mathcal{S}(G) = \sum_i \mathcal{S}(\hat{G}_i) \otimes R_i \, ,$$

associates a tensor with the polylogarithm.

examples:

- $\mathcal{S}(\ln x) = x$
- $\mathcal{S}(\mathrm{Li}_3\, x) = -((1-x) \otimes x \otimes x)$
- $\mathcal{S}(G(1,0,-1,-1,x)) = (1+x) \otimes (1+x) \otimes x \otimes (1-x)$

Rules for symbols

- $R_1 \cdots \otimes (R_a R_b) \otimes \cdots R_k = R_1 \cdots \otimes R_a \otimes \cdots R_k + R_1 \cdots \otimes R_b \otimes \cdots R_k$ (log law)
- $R_1 \cdots \otimes (cR_a) \otimes \cdots R_k = R_1 \cdots \otimes R_a \otimes \cdots R_k$ for constant c
- preserves shuffle product

- $N = 4$ remainder function simplification: Goncharov, Spradlin, Vergu, Volovich ('10)
- extension to coproduct formalism: *const* $\times$ polylog

Example for symbol calculus

goal: derive "simplification formula" for $\mathrm{Li}_2(1/x)$ with $0 < x < 1$, $\mathrm{Im}\, x = \varepsilon$

$$\begin{aligned}\mathcal{S}(\mathrm{Li}_2(1/x)) &= -(-1+1/x)\otimes(1/x)\\ &= (1-x)\otimes x - x\otimes x\\ &= \mathcal{S}(-\mathrm{Li}_2(x) - (1/2)\ln^2 x)\end{aligned}$$

reproduces the highest degree part of the full answer

$$\mathrm{Li}_2(1/x) = -\mathrm{Li}_2(x) - (1/2)\ln^2 x + i\pi\ln x - (2/3)\pi^2$$

note: works at **highest degree** only

- $\mathcal{S}(\ln(-x)) = \mathcal{S}(\ln(x))$: no info on discontinuity
- $\mathcal{S}(\pi) = \mathcal{S}(\zeta_3) = 0$: no constants

"integrating the symbol" $\Rightarrow$ **algorithmic reduction** (structured with shuffle eliminators)
Duhr, Gangl, Rhodes ('11)

What about **subleading degree** *terms (const $\times$ polylog) ?*

- accessible by **coproduct**: Goncharov ('02), Brown ('11)
- extended symbol calculus based on coproduct by Duhr ('12) with:

$$\Delta(\pi) = \pi \otimes 1$$
$$\Delta(\zeta_k) = \zeta_k \otimes 1 + 1 \otimes \zeta_k \qquad \text{for } k \text{ (odd)}$$

example:

$$\begin{aligned}\Delta_{1,1}\left(\mathrm{Li}_2(1/x)\right) &= -\ln(1-1/x) \otimes \ln(1/x) \\ &= \ln(1-x) \otimes \ln(x) - \ln(x) \otimes \ln(x) + i\pi \otimes \ln(x) \\ &= \Delta_{1,1}\left(-\mathrm{Li}_2(x) - (1/2)\ln^2(x) + i\pi\ln(x)\right)\end{aligned}$$

reproduces identity up to pure constant, fix by limits or numerical evaluation:

$$\mathrm{Li}_2(1/x) - \left(-\mathrm{Li}_2(x) - (1/2)\ln^2(x) + i\pi\ln(x)\right) = -6.5797362673929\cdots = -(2/3)\pi^2$$

Definition of the coproduct

For a multiple polylogarithm

$$I(a_0; a_1, \ldots, a_n; a_{n+1}) = \int_{a_0}^{a_{n+1}} \frac{\mathrm{d}t}{t - a_n} I(a_0; a_1, \ldots, a_{n-1}; t)$$

the coproduct Δ is defined according to Goncharov ('02):

$$\Delta\left(I(a_0; a_1, \ldots, a_n; a_{n+1})\right) = \sum_{0=i_1<\ldots<i_{k+1}=n} I(a_0; a_{i_1}, \ldots, a_{i_k}; a_{n+1}) \otimes \prod_{p=0}^{k} I(a_{i_p}; a_{i_p+1}, \ldots, a_{i_{p+1}-1}; a_{i_{p+1}}$$

examples:

- $\Delta(\ln(x)) = 1 \otimes \ln(x) + \ln(x) \otimes 1$
- $\Delta(\mathrm{Li}_2(x)) = 1 \otimes \mathrm{Li}_2(x) - \ln(1-x) \otimes \ln(x) + \mathrm{Li}_2(x) \otimes 1$
- $\Delta(\ln(x)\ln(y)) = 1 \otimes (\ln(x)\ln(y)) + \ln(x) \otimes \ln(y) + \ln(y) \otimes \ln(x) + (\ln(x)\ln(y)) \otimes 1$

Rules for the coproduct

- coassociativity $(\mathrm{id} \otimes \Delta)\,\Delta = (\Delta \otimes \mathrm{id})\,\Delta$
- compatible with product: $\Delta(a \cdot b) = \Delta(a) \cdot \Delta(b)$ where $(a_1 \otimes a_2) \cdot (b_1 \otimes b_2) \equiv (a_1 \cdot b_1) \otimes (a_2 \cdot b_2)$

note: coproduct means "decomposition"

Graded decomposition with the coproduct

- Hopf algebra of multiple polylogs **graded by weight**:

$$\mathcal{H} = \bigoplus_{n=0}^{\infty} \mathcal{H}_n$$

since coproduct preserves weight we may decompose

$$\mathcal{H}_n \xrightarrow{\Delta} \bigoplus_{p+q=n} \mathcal{H}_p \otimes \mathcal{H}_q$$

and define $\Delta_{p,q}$ to be the part with values in $\mathcal{H}_p \otimes \mathcal{H}_q$

- iterated coproduct:

$$\mathcal{H} \xrightarrow{\Delta} \mathcal{H} \otimes \mathcal{H} \xrightarrow{\Delta \otimes \mathrm{id}} \mathcal{H} \otimes \mathcal{H} \otimes \mathcal{H} \xrightarrow{\Delta \otimes \mathrm{id} \otimes \mathrm{id}} \mathcal{H} \otimes \mathcal{H} \otimes \mathcal{H} \otimes \mathcal{H}$$

and corresponding parts $\Delta_{p,q,\ldots,r}$

- $\boxed{\text{symbol } \ \mathcal{S} = \text{maximally iterated coproduct } \ \Delta_{1,\ldots,1} \mod \pi}$

systematic procedure: $gg \to t\bar{t}$ **at two-loops** application:

1. fix maximum degree 4 part with symbol, substract from original expression
2. take $\Delta_{1,1,1,1}$, use unique set of real ln, gives
$$i\pi \otimes \Delta_{1,1,1}(\text{something})$$
match "something" at degree 3 to basis functions using symbols
3. take $\Delta_{2,1,1}$, use unique set of real ln, Li_2, gives
$$\pi^2 \otimes \Delta_{1,1}(\text{something})$$
match "something" at degree 2 to basis functions using symbols
4. take $\Delta_{3,1}$, use unique set of real ln, Li_2, Li_3, gives
$$\zeta(3) \otimes \text{something1} + i\pi^3 \otimes \text{something2}$$
read off "somethingi" (just logs) at degree 1
5. numerical evaluation, match against
$$i\pi\zeta_3, \quad \pi^4$$

(requires matching of degree 3 and lower, obtained by recursion)

expansions at threshold: $\beta \equiv \sqrt{1-4m^2/s} \to 0$ (fixed $\xi = (1-\cos(\theta)/2)$)
change GPLs to

$$G(\ldots;\beta), \quad \text{weights} \in \left\{-1, 0, 1, \frac{1}{1-2\xi}, -\frac{1}{1-2\xi}, [1+o^2], [3+o^2], [1+2o(1-2\xi)+o^2], \ldots\right\}$$
$$G(\ldots;\xi), \quad \text{weights} \in \{0, 1\}$$

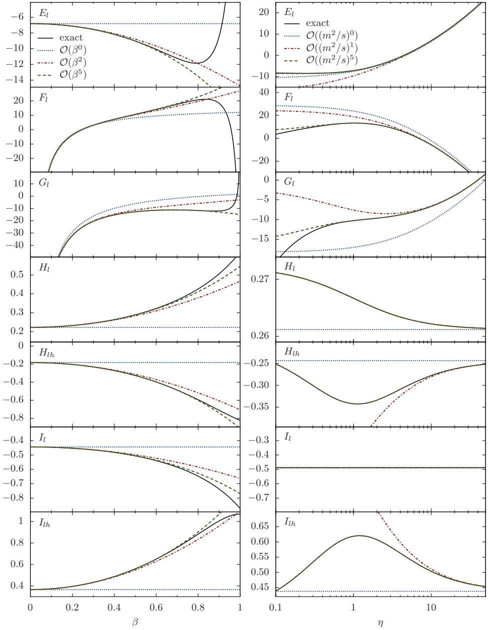
then expand in β, gives e.g.:

$$G_I = \frac{1}{\beta}\left[\frac{\zeta_2}{\varepsilon} - \frac{14\zeta_2}{3}\right] + \frac{1}{\varepsilon}\left(\frac{\zeta_2}{2} - \frac{17}{12}\right) + \frac{1}{36}\left(216\zeta_2 \ln 2 + 480 \ln 2 - 273\zeta_2 - 6\zeta_3 + 211\right)$$
$$+ \beta\left[\frac{3\zeta_2}{\varepsilon} - 4c_\theta^2\zeta_2 - 8\zeta_2\right] + \mathcal{O}(\beta^2)$$

expansion for small mass ("high energy"): $m^2/s \to 0$ (fixed $\phi = -(t-m^2)/s$)
change GPLs to:

$$G(\ldots;x), \quad \text{weights} \in \left\{-1, 0, 1, -\frac{1-\phi}{\phi}, -\frac{\phi}{1-\phi}, [1+o^2], [1-o+o^2], \left[1+o\frac{1-2\phi}{\phi}+o^2\right], \ldots\right\}$$
$$G(\ldots;\phi), \quad \text{weights} \in \{0, 1\}$$

then expand in m^2/s



left: threshold expansion, $\beta = \sqrt{1 - 4m^2/s}$, $\cos\theta = 0.7$

right: "high energy" expansion, $\eta = s/(4m^2) - 1$, $\phi = -(t - m^2)/s = 0.35$