Computational tools for scattering amplitudes at NLO QCD

Andreas van Hameren

The Henryk Niewodniczański Institute of Nuclear Physics

Polish Academy of Sciences

Loops and Legs in Quantum Field Theory

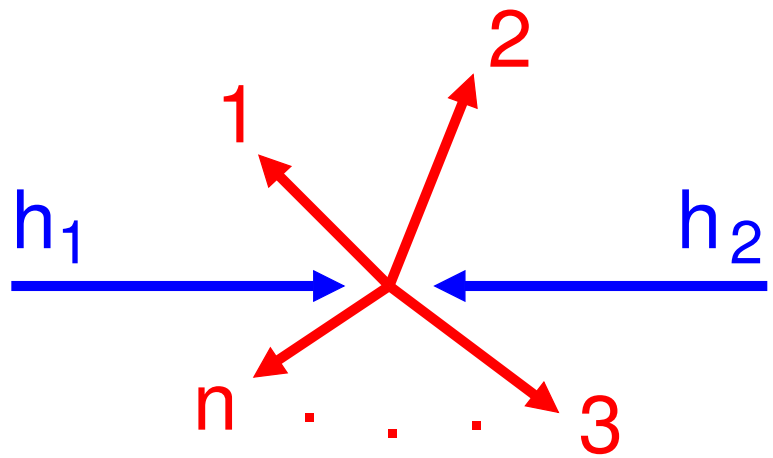
17–04–2012 Wernigerode

Partly supported by FNP grant Homing plus/2010–2/6

Outline

- Introduction
- HELAC-NLO
  - $pp(p\bar{p}) \to W^+W^- b\bar{b} \to 4\ell\, b\bar{b}$
  - $pp(p\bar{p}) \to t\bar{t}\, jj$
- Multi-precision in one-loop calculations
- ONELOOP
- Summary

Hard scattering cross sections within collinear factorization



PDFs are related to the structure of the hadrons, universal to the scattering process

$$\sigma_{h_1,h_2\to n}(p_1,p_2) = \sum_{a,b}\int dx_1\,dx_2\; f_a(x_1,\mu)\, f_b(x_2,\mu)\; \hat{\sigma}_{a,b\to n}(x_1p_1, x_2p_2;\mu)$$

$$\hat{\sigma}_{a,b\to n}(p_a,p_b;\mu) = \int d\Phi(p_a,p_b\to\{p\}_n)\; |\mathcal{M}_{a,b\to n}(p_a,p_b\to\{p\}_n;\mu)|^2\; \mathcal{O}(p_a,p_b,\{p\}_n)$$

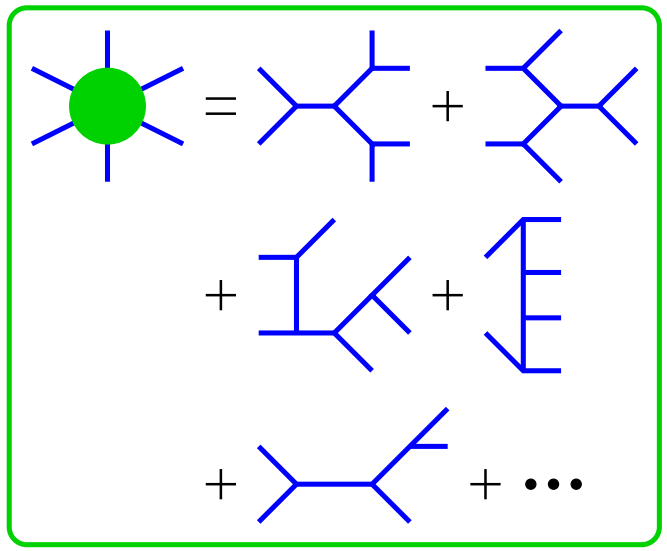
Phase space (includes spin/color summation) governs the kinematics

Matrix element (squared) contains model parameters, governs the dynamics

Observable, imposes phase space cuts

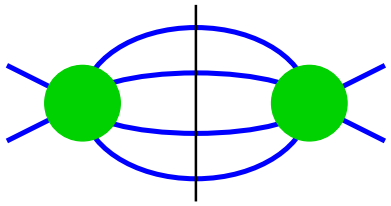
Perturbative cross section

Hard cross sections can be calculated within perturbative QCD, which allows to pay for precision with complexity. Higher orders imply more "unobserved" degrees of freedom that have to be integrated.



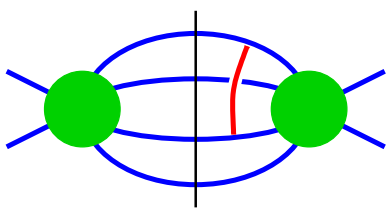
Leading Order:

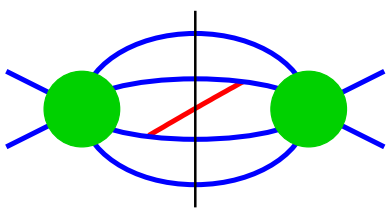
$$\hat{\sigma}^{\mathrm{LO}}_{a,b\to n} = \int d\Phi_n \left|\mathcal{M}^{(0)}_{a,b\to n}\right|^2 \mathcal{O}^{\mathrm{LO}}_n$$



Next-to-Leading Order:

$$\hat{\sigma}^{\mathrm{NLO}}_{a,b\to n} = \int d\Phi_n \, 2\Re\left(\mathcal{M}^{(0)}_{a,b\to n}\,\mathcal{M}^{(1)}_{a,b\to n}\right)\mathcal{O}^{\mathrm{LO}}_n + \int d\Phi_{n+1}\left|\mathcal{M}^{(0)}_{a,b\to n+1}\right|^2 \mathcal{O}^{\mathrm{NLO}}_{n+1}$$





The one-loop amplitude $\mathcal{M}^{(1)}_{a,b\to n}$ and the soft/collinear divergencies in the integration over Φ_{n+1} complicate the calculation considerably.

Helac–NLO

http://helac–phegas.web.cern.ch/helac–phegas/

- based on automatic LO platform HELAC/PHEGAS Cafarella,Kanaki,Papadopoulos,Worek
- real-radiation with dipole-subtraction HELAC-DIPOLES Czakon,Papadopoulos,Worek
- one-loop amplitudes with HELAC-1LOOP AvH,Papadopoulos,Pittau
  - OPP reduction with CUTTOOLS Ossola,Papadopoulos,Pittau
  - scalar integrals with ONELOOP AvH
  - rational contribution Draggiotis,Garzelli,Malamos,Papadopoulos,Pittau
- phase space integration of real-subtracted with KALEU AvH
- virtual contribution for "any" process (presented up to hexagons so far).
- so far, produced full differential distributions for
  - $pp \to t\bar{t}\,b\bar{b}$ Bevilacqua,Czakon,Papadopoulos,Pittau,Worek
  - $pp \to t\bar{t} + 2j$ Bevilacqua,Czakon,Papadopoulos,Worek
  - $pp \to W^+W^-b\bar{b} \to 4\ell\, b\bar{b}$ Bevilacqua,Czakon,AvH,Papadopoulos,Worek
- used in combination with POWHEG BOX (Alioli,Nason,Oleari,Re) for $2 \to 3$ processes at NLO combined with parton shower
  - $pp \to t\bar{t} + j$ Kardos,Papadopoulos,Trocsanyi
  - $pp \to t\bar{t} + H$, $pp \to t\bar{t} + Z$ Kardos,Garzelli,Papadopoulos,Trocsanyi

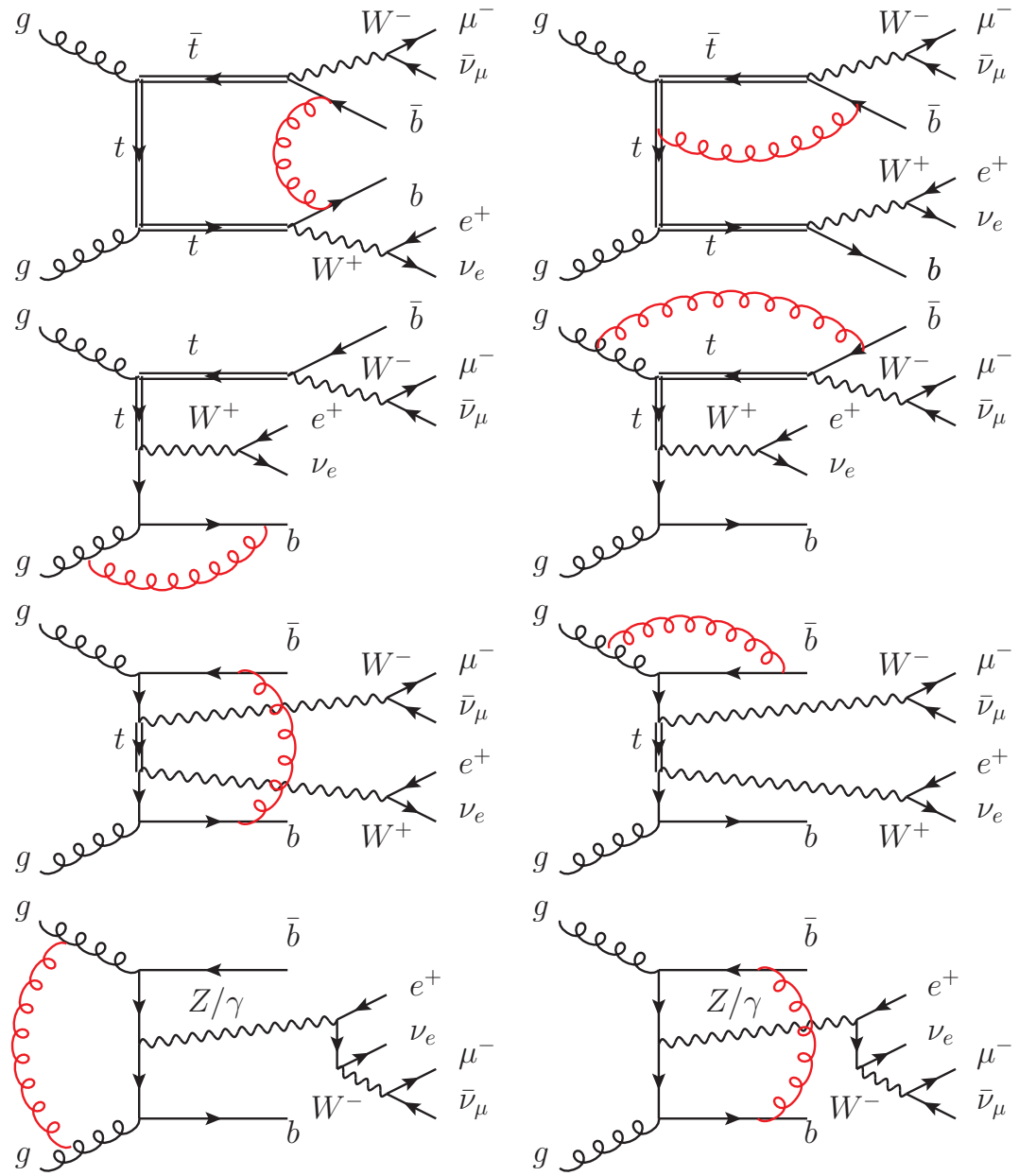
$pp(p\bar{p}) \to W^+W^-b\bar{b} \to 4l\, b\bar{b}$ at NLO QCD

computational challenges:

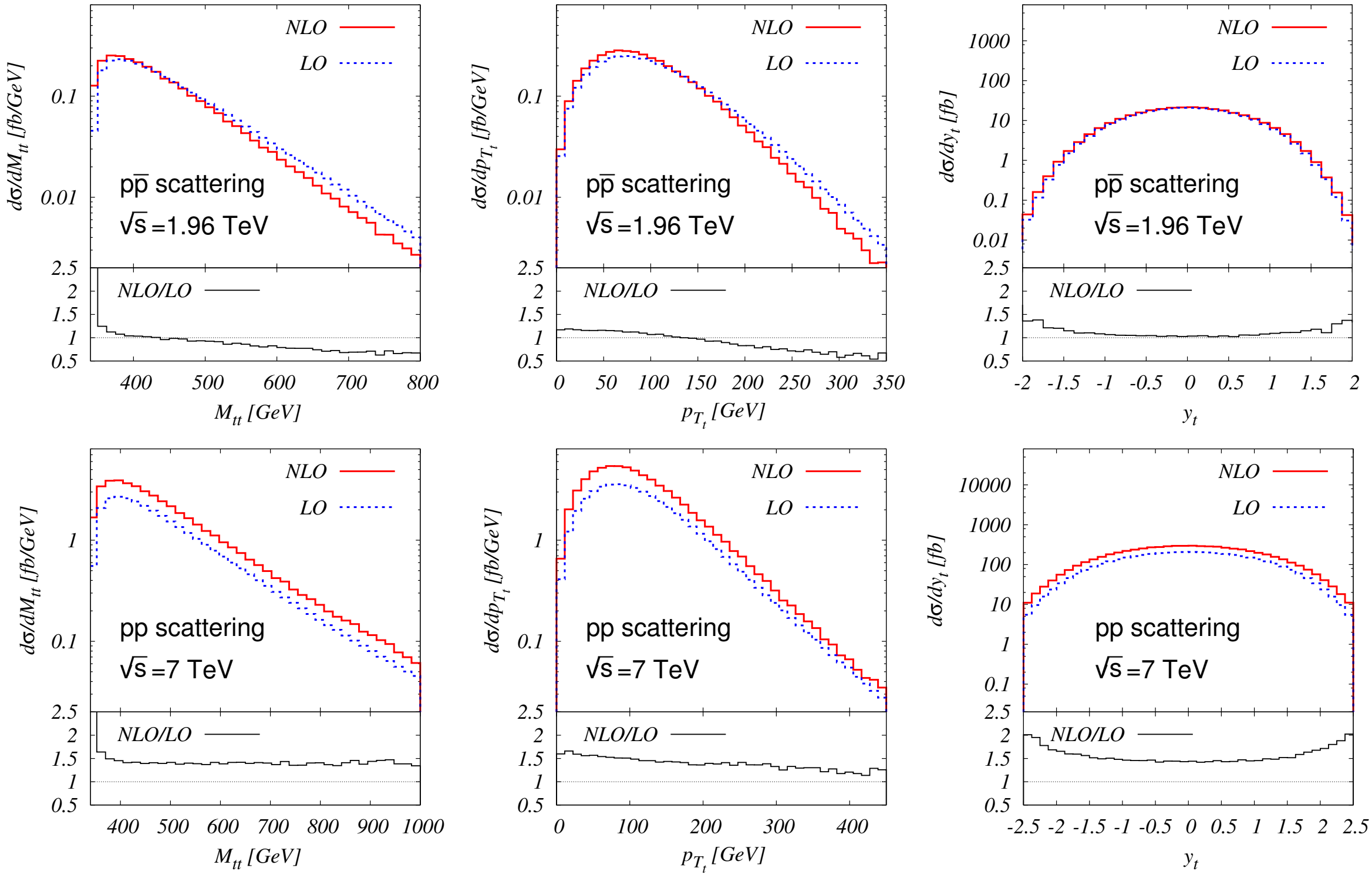
- consistent treatment of top-quarks with non-zero width requires the application of the complex-mass scheme Denner,Dittmaier,Roth,Wackeroth '99, Denner,Dittmaier,Roth,Wieders '05.

  Demands one-loop scalar functions with complex internal masses 't Hooft,Veltman '79, Dao Thi Nhung,Le Duc Ninh '09, Denner,Dittmaier '11

- real-subtracted phase space integral concerns 7-particle final state with non-trivial peak-structure.



$pp(p\bar{p}) \to W^+W^-b\bar{b} \to 4l\, b\bar{b}$ at NLO QCD



$pp(p\bar{p}) \to t\bar{t}\,jj$ at NLO QCD

Bevilacqua,Czakon, Papadopoulos,Worek

Important background for Higgs searches at Tevatron and LHC

- $H \to WW^*$ via weak boson fusion.
- $H \to b\bar{b}$ associated with a $t\bar{t}$ pair.

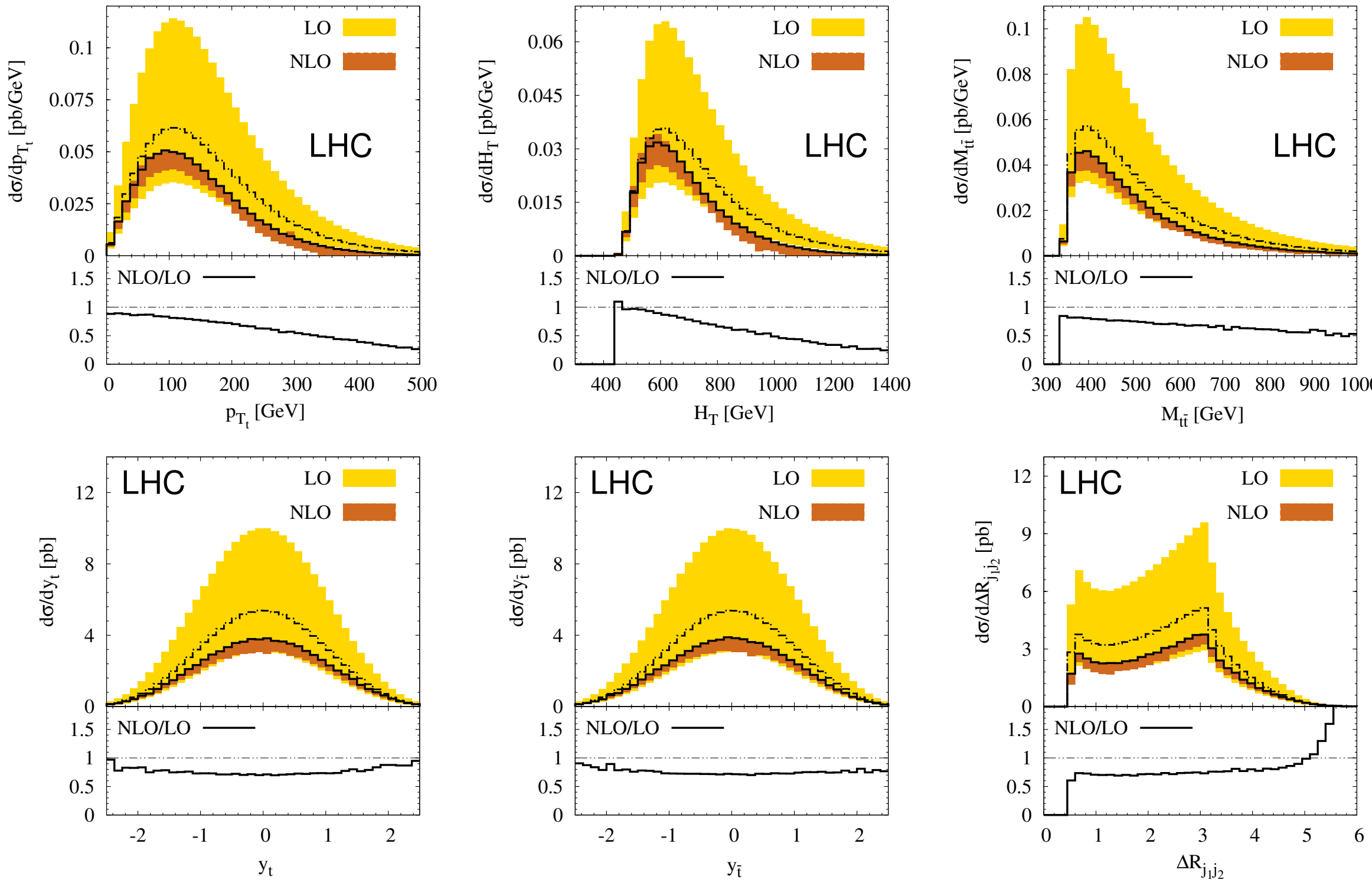
Computational challenge: it's big

subprocess	#1-loop graphs	subprocess	#graphs	#dipoles
$gg \to t\bar{t}gg$	4510	$gg \to t\bar{t}ggg$	1240	75
$gg \to t\bar{t}q\bar{q}$	1100	$gg \to t\bar{t}q\bar{q}g$	341	55
$q\bar{q} \to t\bar{t}gg$	1100	$q\bar{q} \to t\bar{t}ggg$	341	75
$gq \to t\bar{t}qg$	1100	$gq \to t\bar{t}q'\bar{q}'q$	64	25
$qg \to t\bar{t}qg$	1100	$gq \to t\bar{t}qgg$	341	65
$qq' \to t\bar{t}qq'$	205	$qg \to t\bar{t}q'\bar{q}'q$	64	25
$q\bar{q} \to t\bar{t}q'\bar{q}'$	205	$qg \to t\bar{t}qgg$	341	65
$q\bar{q} \to t\bar{t}q\bar{q}$	410	$qq' \to t\bar{t}qq'g$	64	40
		$q\bar{q} \to t\bar{t}q'\bar{q}'g$	64	35
		$q\bar{q} \to t\bar{t}q\bar{q}g$	128	45

- 2 collider setups
- 3 jet algorithms (k_T, anti-k_T, C/A)
- 2 cone-sizes

$pp(p\bar{p}) \to t\bar{t}\, jj$ at NLO QCD

Bevilacqua,Czakon, Papadopoulos,Worek



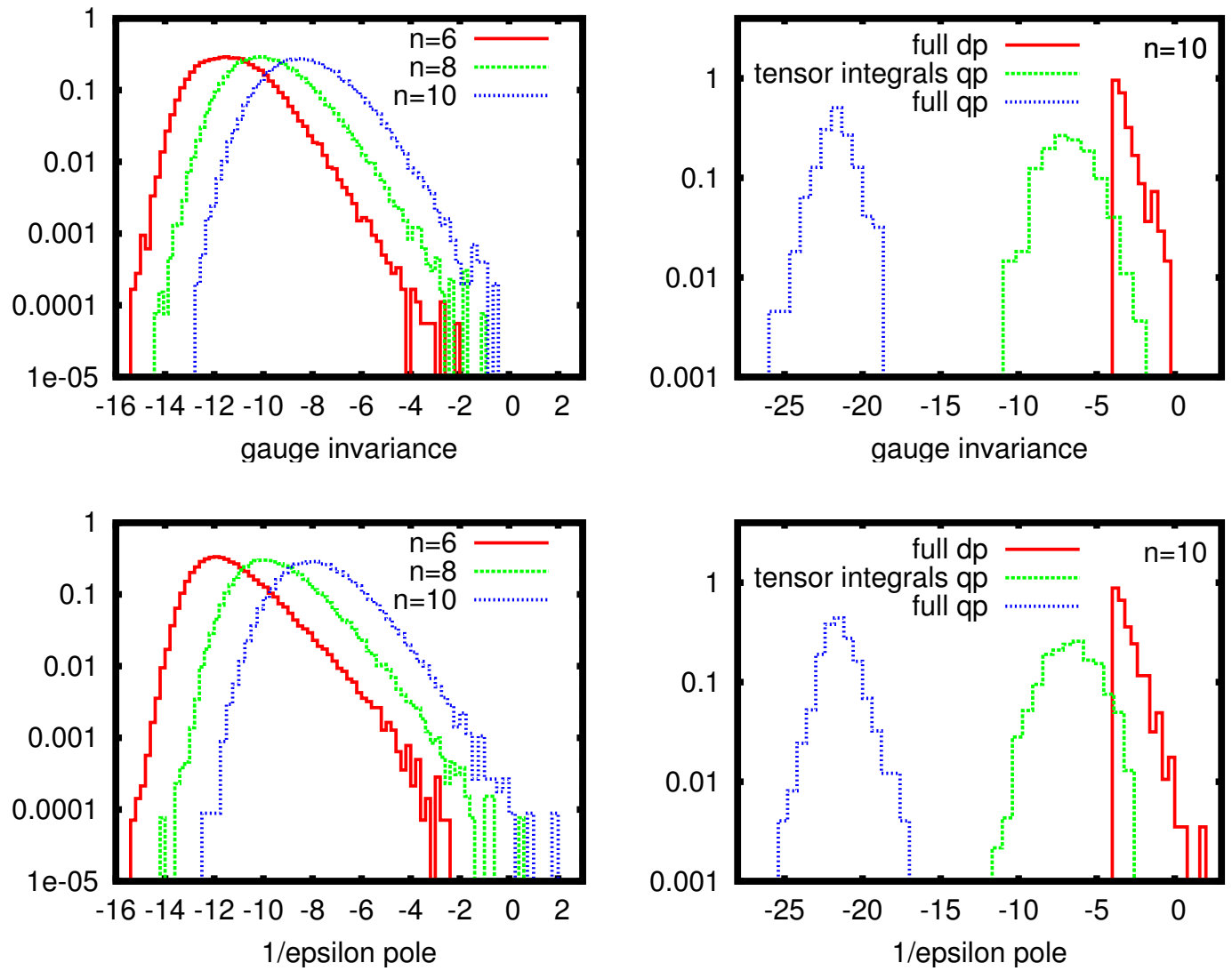
Numerical instabilities

- calculations of multi-leg one-loop amplitudes often suffer from numerical instabilities
- these are typically caused by the inversion of small near-singular linear systems
- practically, they arise via the cancellation between numerator terms, enhanced by small denominators: $((L + \varepsilon x_1) - (L + \varepsilon x_2))/\varepsilon$
- phase space points leading to numerical instabilities in general need not to be special from the physical point of view, *eg.* $p_1 = p_2$ with $p_1 \cdot p_2 \neq 0$
- within a Monte Carlo calculation, numerical instabilities may be dealt with by diagnosing phase space points as "unstable", and repeat the calculation for those at increased floating point precision
- precision is expensive: it is essential to keep the precision as low as possible for as many phase space points as possible

Example strategy

Multi-gluon one-loop amplitudes using tensor integrals, JHEP 0907 (2009) 088

- phase space points may be diagnosed by checking Ward identities (expensive)
- or by comparing the $\frac{1}{D-4}$ contribution with an analytical formula (may not be available)
- one may differentiate w.r.t. the various components in the calculation of the amplitude (tensors, coefficients)
- full higher-precision result appears to be much better than differentiated result



Left: distributions from 10^5 uniformly distributed phase space points for 6, 8 and 10 external gluons.
Right: tails of the distributions for 10 external gluons and different levels of precision.

Strategy in CutTools

R. Pittau, Comput.Phys.Commun.181(2010)1941

Numerical instabilities may be diagnosed by the application of other non-trivial formal invariance than gauge independence, in any stage of the calculation of the coefficients of the master integrals. Eg. in the OPP method

$$\mathcal{M}^{(1-\text{loop})} = \sum_i \boxed{c_i}\,\boxed{I_i}$$

process-dependent coefficients

universal master integrals

$$L(q) = \sum_j C_j(x) R_j(q, x)$$

where x is a "mass-shift". The sought coefficients are at $x = 0$. The functional dependence of $C_j(x)$ on x is known. Solving the system for different values of x, more than 1 estimate of $C_j(0)$ can be extracted. Since this is only done for diagnosis, $L(q)$ may be reconstructed with the first estimate to get the others.

OPP method: $\boxed{L(q)} = \sum_j \boxed{C_j}\,\boxed{R_j(q)}$

one-loop numerator

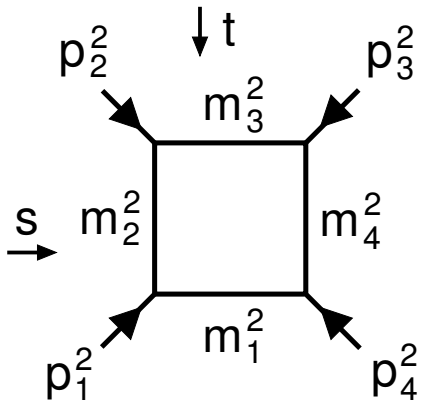
includes sought coefficients

universal polynomials

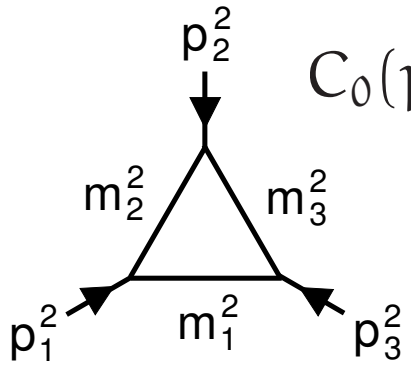
Numerical instabilities

- diagnose phase space points
- re-evaluate bad phase space points at higher floating point precision
- requires the possibility to evaluate the various components of a one-loop amplitude at higher precision, also the master integrals

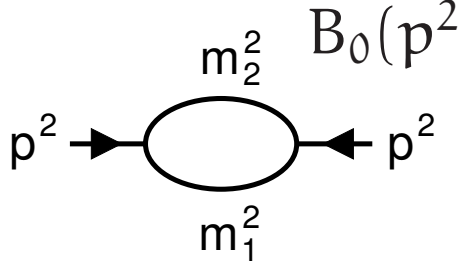
OneLOop



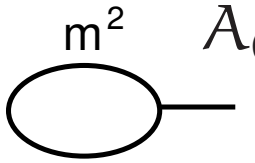
$$D_0(p_1^2, p_2^2, p_3^2, p_4^2, s, t, m_1^2, m_2^2, m_3^2, m_4^2; \mu, \varepsilon)$$



$$C_0(p_1^2, p_2^2, p_3^2, m_1^2, m_2^2, m_3^2; \mu, \varepsilon)$$



$$B_0(p^2, m_1^2, m_2^2; \mu, \varepsilon)$$



$$A_0(m^2; \mu, \varepsilon)$$

$$f(\varepsilon) = \frac{f^{(2)}}{\varepsilon^2} + \frac{f^{(1)}}{\varepsilon} + f^{(0)} + \mathcal{O}(\varepsilon)$$

$$A_0(m^2; \mu, \varepsilon) = \frac{\Gamma(1-2\varepsilon)\,\mu^{2\varepsilon}}{\Gamma(1-\varepsilon)^2\,\Gamma(1+\varepsilon)\,i\,\pi^{2-\varepsilon}} \int \frac{d^{4-2\varepsilon}q}{q^2 - m^2 + i0}$$

- all IR and UV divergent cases within dimensional regularization
- all kinematical configurations relevant for collider physics
- any internal squared masses with positive real and negative imaginary part
- also supplies the Passarino-Veltman functions B_1, B_{00} and B_{11}

OneLOop

Formulas based on those from

G. 't Hooft and M.J.G. Veltman, Nucl.Phys.B153(1979)365-401
W. Beenakker, H. Kuijf, W.L. van Neerven, J. Smith, Phys.Rev.D40(1989)54
W. Beenakker and D. Denner, Nucl.Phys.B338(1990)349
A. Denner, U. Nierste, R. Scharf, Nucl.Phys.B367(1991)637
Z. Bern, L.J. Dixon and D.A. Kosower, Nucl.Phys.B412(1994)751
E.L. Berger, M. Klasen, T.M.P. Tait, Phys.Rev.D62(2000)095014
G. Duplancic and B. Nizic, Eur.Phys.J.C20(2001)357
W. Beenakker, S. Dittmaier, M. Kramer, B. Plumper, M. Spira, P.M. Zerwas, Nucl.Phys.B653(2003)151
A. Denner and S. Dittmaier, Nucl.Phys.B734(2006)62
R. Keith Ellis and G. Zanderighi, JHEP0802(2008)002
Dao Thi Nhung and Le Duc Ninh, Comput.Phys.Commun.180(2009)2258

Numerically stable implementations of

$$\frac{\log(x)}{x-1} \quad , \quad \frac{\mathrm{Li}_2(x) - \mathrm{Li}_2(y)}{x-y}$$

OneLOop–3.0

Fortran module supplies generic routine:

```
use avh_olo
call olo( rslt ,m1 )                                          ! A0
call olo( rslt ,p1 ,m1,m2 )                                   ! B0
call olo( rslt ,p1,p2,p3 ,m1,m2,m3 )                          ! C0
call olo( rslt ,p1,p2,p3,p4,p12,p23 ,m1,m2,m3,m4 )            ! D0
call olo( rslt11,rslt00,rslt1,rslt0 ,p1 ,m1,m2 )              ! B11, B00, B1, B0
```

- `rslt(i)` is the ε^{-i}-coefficient
- all input real,
  or momenta real and masses complex,
  or all input complex
- optional real input scale `mu`
  default scale set with
  `call olo_scale(mu)`
- threshold for onshellness set with
  `call olo_onshell(thrs)`

- any precision supplied by compiler
- arbitrary precision with `mpfun90` or `arprec` D.H. Bailey, X. Li, A. Kaiser
  `type(mp_complex) :: z,p1,m1,...`
- arbitrary precision may be changed during runtime
  `call olo_precision(ndecimals)`
  (necessary tables are stored and added to a list of encountered precisions)

OneLOop–3.0

Find: `www.google.com/#q=oneloop+hameren`
Download: `helac-phegas.web.cern.ch/helac-phegas/OneLOop.html`
Unpack: `tar -xzf OneLOop-3.0.tgz`
Enter directory: `cd OneLOop-3.0`

Edit makefile:

```
FC = gfortran                    # Fortran compiler
FFLAGS = -O                      # possible flags
DPKIND = kind(1d0)               # double precision kind
QPKIND =                         # quadruple precision kind
KINDMOD =                        # module providing kind parameters
MPTYPE = mpfun90                 # mpfun90 or arprec if available
TLEVEL = no                      # top-level avh_olo_-routines
MPFUN90MOD = $(HOME)/mpfun90/    # directory with mpfun90 modules
```

Create a library `libavh_olo.a`: `make`
or just a single source file `avh_olo.f90`: `make source`

The source file may also be created with a script:
`OneLOop-3.0/src/avh_pc_olo.sh -dpkind "kind(1d0)" -mptype "mpfun90"`

Summary

- Short presentation of HELAC-NLO
- Addressed the application to
  - $pp(p\bar{p}) \to W^+W^- b\bar{b} \to 4\ell\, b\bar{b}$
  - $pp(p\bar{p}) \to t\bar{t}\, jj$
- Presented ONELOOP-3.0 which can operate at any precision provided by the compiler, or at multi-precision in combination with `mpfun90` or `arprec`.