GoSam

A program for the computation of one loop virtual contribution to scattering amplitudes

Francesco Tramontano
CERN Theory Group

on behalf of the GoSam collaboration:
G. Cullen, N. Greiner, G. Heinrich, G. Luisoni
P. Mastrolia, G. Ossola, T. Reiter and me

MadGraph workshop 2011, Academia Belgica - Roma

Automatic tools for the computation of virtual one loop correction

- FeynArts/FormCalc/LoopTools (public) Thomas Hahn et al
- Helac-NLO Bevilacqua, Czakon, van Hameren, Papadopoulos, Pittau, Worek
- MadLoop Hirschi,Frederix,Frixione,Garzelli,Maltoni,Pittau

  uses CutTools (public [Ossola, Papadopoulos, Pittau]) and MadGraph
- Golem-Samurai (Samurai public [Mastrolia, Ossola, Reiter, Tramontano])

  Cullen, Greiner, GH, Luisoni, Mastrolia, Ossola, Reiter, Tramontano
- NGluon (public) [Badger, Biedermann, Uwer]
- dedicated programs also involve high level of automation

  Denner, Dittmaier, Pozzorini et al, VBFNLO coll., MCFM, Blackhat, Rocket, ...

NEW DEVELOPMENTS IN ONE LOOP COMPUTATIONS

- Pioneering works:

  - Improvements in the computation of tensor integrals
    Binoth et al GOLEM95
    Denner & Dittmaier

  - Application of unitarity to the computation of one loop amplitudes
    Bern, Dixon, Kosower
    Britto, Cachazo, Feng

  - Reduction at the integrand level
    Ossola, Papadopoulos, Pittau
    Ellis, Giele, Kunszt, Melnikov

OPP integrand decomposition: 4-dim

- At integrand level the structure is enriched by polynomial terms that integrate to zero (I multiplied with all the propagators)

$$
\begin{aligned}
N(q) &= \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d(i_0 i_1 i_2 i_3) + \tilde{d}(q; i_0 i_1 i_2 i_3) \right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i + \sum_{i_0<i_1<i_2}^{m-1} \left[c(i_0 i_1 i_2) + \tilde{c}(q; i_0 i_1 i_2) \right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
&+ \sum_{i_0<i_1}^{m-1} \left[b(i_0 i_1) + \tilde{b}(q; i_0 i_1) \right] \prod_{i \neq i_0, i_1}^{m-1} D_i + \sum_{i_0}^{m-1} \left[a(i_0) + \tilde{a}(q; i_0) \right] \prod_{i \neq i_0}^{m-1} D_i
\end{aligned}
$$

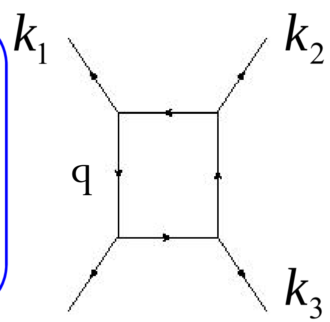
- A choice of q fulfilling 4-ple cut condition: $D_{i0} = D_{i_1} = D_{i_2} = D_{i_3} = 0$ will single out just one polynomial

$$\Delta_{i_0 i_1 i_2 i_3} = \left[d(i_0 i_1 i_2 i_3) + \tilde{d}(q; i_0 i_1 i_2 i_3) \right]$$

$\tilde{d}$ can **only** be of the type $q.p$

where $p = \varepsilon_{\alpha\beta\gamma} k_1^{\alpha} k_2^{\beta} k_3^{\gamma}$

[proof in OPP 2007]



- Once fitted such polynomial we can subtract it from both sides and repeat the game with another multiple cut condition -> recursive solution

- For each phase space point the only requirement for the reduction is the knowledge of the numerical value of the numerator function N for a small set of values of the loop momentum variable, solutions of the multiple cut conditions

Extension to D-dim

- fix a parametric form for the loop momentum in terms of a linear combination of four known 4-vectors e_i suitably chosen

$$\bar{\not{q}} = \not{q} + \not{\mu} \qquad \bar{q}^2 = q^2 - \mu^2 \qquad q = x_1 e_1 + x_2 e_2 + x_3 e_3 + x_4 e_4$$

the vanishing term (spurious term in the OPP terminology) are then polynomials of x_i and $\boldsymbol{\mu^2}$

- The problem is to fit the coefficients in the polynomials Δ

$$N(\bar{q}) = \sum_{i<<m}^{n-1} \Delta_{ijk\ell m}(\bar{q}) \prod_{h\neq i,j,k,\ell,m}^{n-1} \bar{D}_h + \sum_{i<<\ell}^{n-1} \Delta_{ijk\ell}(\bar{q}) \prod_{h\neq i,j,k,\ell}^{n-1} \bar{D}_h + \\ + \sum_{i<<k}^{n-1} \Delta_{ijk}(\bar{q}) \prod_{h\neq i,j,k}^{n-1} \bar{D}_h + \sum_{i<j}^{n-1} \Delta_{ij}(\bar{q}) \prod_{h\neq i,j}^{n-1} \bar{D}_h + \sum_{i}^{n-1} \Delta_{i}(\bar{q}) \prod_{h\neq i}^{n-1} \bar{D}_h$$

✓ Example: 3-ple cut residue

$$\Delta_{ijk}(\bar{q}) = c_{3,0}^{(ijk)} \boxed{+ c_{3,7}^{(ijk)} \mu^2} - \left((c_{3,1}^{(ijk)} \boxed{+ c_{3,8}^{(ijk)} \mu^2}) x_4 + (c_{3,4}^{(ijk)} \boxed{+ c_{3,9}^{(ijk)} \mu^2}) x_3 \right) (e_1 \cdot e_2) + \\ + \left(c_{3,2}^{(ijk)} x_4^2 + c_{3,5}^{(ijk)} x_3^2 \right) (e_1 \cdot e_2)^2 - \left(c_{3,3}^{(ijk)} x_4^3 + c_{3,6}^{(ijk)} x_3^3 \right) (e_1 \cdot e_2)^3 .$$

✓ with the 3 cut conditions: $D_i = D_j = D_k = 0$ one fixes x_1, x_2 and the product $x_3 x_4$

Amplitudes & Master Integrals

$$\mathcal{A}_n = \sum_{i<j<k<\ell}^{n-1} \left\{ c_{4,0}^{(ijk\ell)} I_{ijk\ell}^{(d)} + \frac{(d-2)(d-4)}{4} c_{4,4}^{(ijk\ell)} I_{ijk\ell}^{(d+4)} \right\}$$
$$+ \sum_{i<j<k}^{n-1} \left\{ c_{3,0}^{(ijk)} I_{ijk}^{(d)} - \frac{(d-4)}{2} c_{3,7}^{(ijk)} I_{ijk}^{(d+2)} \right\}$$
$$+ \sum_{i<j}^{n-1} \left\{ c_{2,0}^{(ij)} I_{ij}^{(d)} + c_{2,1}^{(ij)} J_{ij}^{(d)} + c_{2,2}^{(ij)} K_{ij}^{(d)} - \frac{(d-4)}{2} c_{2,9}^{(ij)} I_{ij}^{(d+2)} \right\}$$
$$+ \sum_{i}^{n-1} c_{1,0}^{(i)} I_i^{(d)}$$

$$\int d^d\bar{q} \frac{\bar{q}\cdot e_2}{\bar{D}_i \bar{D}_j} = J_{ij}^{(d)}$$
$$\int d^d\bar{q} \frac{(\bar{q}\cdot e_2)^2}{\bar{D}_i \bar{D}_j} = K_{ij}^{(d)}$$

The sources of rational terms are the integrals with μ^2 powers in the numerator

They are generated by the reduction algorithm(R1), but could also be present ab initio in the numerator function as a consequence of the d-dimensional algebraic manipulations (R2)

$$\int d^d\bar{q} \frac{\mu^2}{\bar{D}_i \bar{D}_j} = -\frac{(d-4)}{2} I_{ij}^{(d+2)}$$
$$\int d^d\bar{q} \frac{\mu^4}{\bar{D}_i \bar{D}_j \bar{D}_k \bar{D}_\ell} = \frac{(d-2)(d-4)}{4} I_{ijk\ell}^{(d+4)}$$
$$\int d^d\bar{q} \frac{\mu^2}{\bar{D}_i \bar{D}_j \bar{D}_k} = -\frac{(d-4)}{2} I_{ijk}^{(d+2)}$$

On-shell methods are quite flexible:

- Different implementations

  - Sewing tree level amplitudes

+ works with gauge invariant objects
- still not easy to automate the rational terms for general one loop amplitudes

Blackhat: Recursive bootstrap approach [Ita, Bern, Dixon, Febres Cordero, Kosower, Maitre]
Rocket: Tree level amplitudes in different dimensions [Ellis, Giele, Kunszt, Melnikov, Zanderighi]
Samurai: works with D-dimensional tree level amplitudes [Mastrolia, Ossola, Reiter, FT]

  - Diagrammatic approach:

+ contain all the information on the rational terms
- single diagrams are not gauge invariant objects: big cancellations

  - Fully numerical construction through the single cut:
    Helac-NLO [Bevilacqua, Czakon, Papadopoulos, Worek]
    MadLoop [Hirschi, Frederix, Frixione, Garzelli, Maltoni Pittau]

  - Fully algebraic method: **GoSam**
    [Cullen, Greiner, Heinrich, Luisoni, Mastrolia, Ossola, Reiter, FT]

Main features of the "Algebraic Way":

- Amplitudes generated with Feynman diagrams
- Algebraic manipulations are allowed before starting the numerical integration
- The generation of numerators is executed separately from the numerical reduction
- Optimization: grouping of diagrams, smart caching
- Control over sub-parts of the computation (move in/out subsets of diagrams)
- Algebra in dimension d, different schemes

Great flexibility in the reduction
Choice between different algorithms at runtime

Golem/Samurai

Algebraic generation of **d-dimensional integrands** via **Feynman diagrams**

Reduction at the Integrand Level: **d-dimensional extension of OPP reduction**

Target: provide an **automated tool** for stable evaluation of one-loop matrix elements

- **be general/model independent** (QCD, EW, BSM)
- **interface with existing tools** like MadEvent, Sherpa, PowHEG, ...
- build upon **open source tools only** (i.e. Samurai, Golem95, QGraf, Form, Spinney, Haggies, QCDLoop, OneLOop)
- **support open standards**

GoSam

An automated amplitude generation based on Feynman diagrams
(distributed as a python package)

- FORM
  J.A.M. Vermaseren, (1991)
- QGRAF [Fortran – open source]
  P. Nogueira, (1993)
- Haggies [Java – open source]
  T. Reiter, (2009)
- Spinney [Form – open source]
  Cullen, Koch-Janusz, Reiter, (2010)

Default Option: **Samurai**

P.Mastrolia, G.Ossola, T.Reiter, FT (2010)

- OPP Reduction Algorithm G.Ossola, C. Papadopoulos, R, Pittau (2007)
- d-dimensional extension Ellis, Giele, Kunszt, Melnikov (2008)
- Coefficients of Polynomials via DFT Mastrolia et al. (2008)
- Model-independent Computation of the full Rational Term

Samurai-2.1 will be public soon:

✓ complex masses
✓ scalar integral cashing
✓ code optimization (w Haggies)
✓ tensorial reconstruction
✓ reduced samplings
✓ passing of the invariants
✓ updeted scalar intergral library
✓ bug fixes

Other options available (at runtime):

Golem95
Binoth, Cullen, Guillet, Heinrich, Kleinschmidt, Pilon, Reiter, Rodgers (2008)

Tensorial Integrand-level Reduction
G, Heinrich, G.Ossola, T.Reiter, FT (2010)

Last Step: multiply all coefficients with the corresponding Master Integral

QCDloop
(Ellis, Zanderighi)

OneLOop
(A. van Hameren)

Golem95C
(Cullen, Guillet, Heinrich, Kleinschmidt, Pilon, Reiter, Rodgers)

Upgrade of *Golem95* library, real and complex masses supported

`http://projects.hepforge.org/~golem/95/`

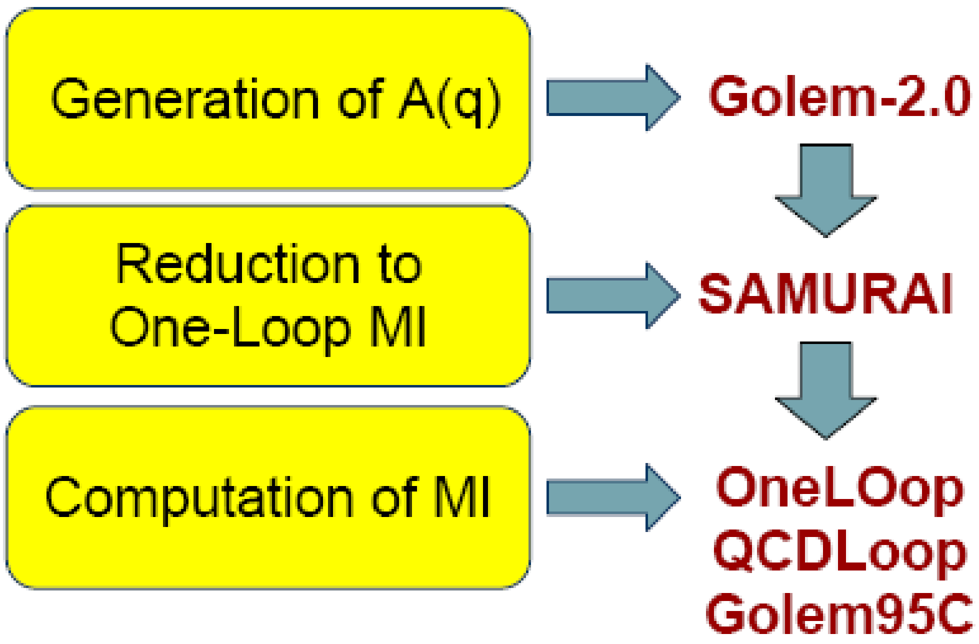
NEW!

LoopTools
(T. Hahn)

NEW!

PJFry
(V. Yundin)

Standard GoSam reduction:



This process is **fully automated**

Walking through GoSam…

Preparation of the "card": we use as example $u\,\bar{d} \to \bar{s}\,c\,e^-\,\bar{\nu}_e\,\mu^+\,\nu_\mu$

```
#!/bin/env /usr/local/bin/golem-main.py
process_name=nloop4
process_path=/home/tramonta/codici/processes/nloop4
                    [directory previously created]
in=u,d~
out=s~,c,e-,nebar,mu+,nmu
model=smdiag
                    [model can be generated through FeynRules and LanHEP]

model.options=mW=80.376,mZ=91.1876,GF=0.0000116639
                    [for sm and smdiag the other couplings will be generated]

order=gw,4,4
order=gs,2,4

zero=mC,mS,mU,mD,mmu,me
one=gs,e

helicities=-+-+-++-

qgraf.bin=/home/tramonta/codici/QGRAF/qgraf
qgraf.options=onshell,nosnail,notadpole
qgraf.verbatim=\
 true = iprop[phim,phip,chi,H,B,T,0,0];\n\
 true = iprop[Wp,2,2];

extensions=samurai,dred       [golem95, pjfry]
r2=explicit         [implicit, only, none]
abbrev.limit=500
abbrev.level=diagram          [group, helicity]

form.bin=form
fc.bin=gfortran -O2
```

… and also

- Ready to use UFO model files: there will be examples in the release
  - Support up to spin 2 but this part is still untested

- Python framework for deep level diagram selection
  - useful to split the computation conveniently

- Very basic existing interface with POWHEG
  - spin correlated and color correlated born matrix elements available

Where the card is:

$ golem-main nloop4.rc

In the process directory

$ make source

$ make compile

$ cd doc

$ make

Documentation:

GoSam: $u\bar{d} \to \nu_\mu \mu^+ e^- \bar{\nu}_e \bar{s} c$

Diagrams

tramonta

2011-09-20 (19:55:21)

Abstract

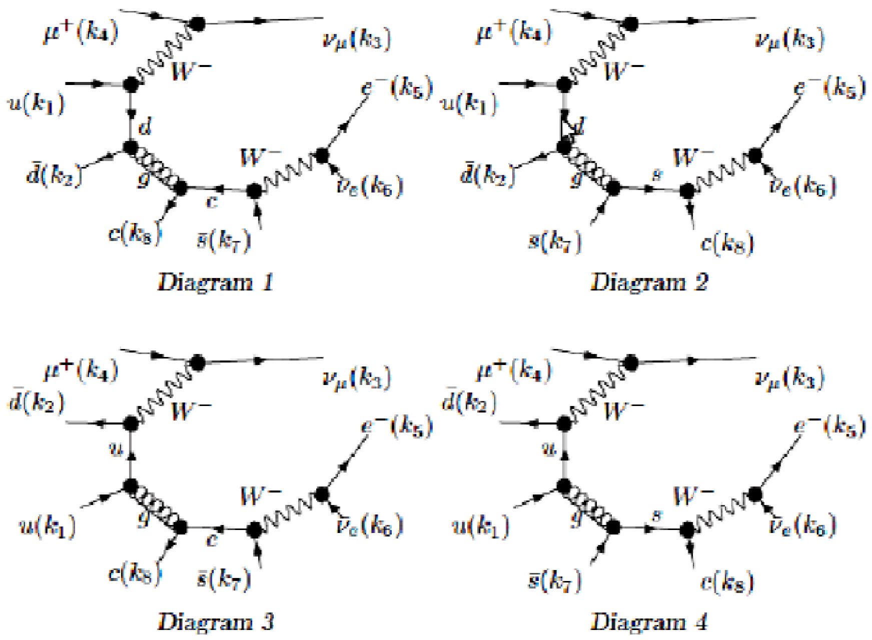
This process consists of 4 tree-level diagrams and 86 NLO diagrams. GoSam has identified 6 groups of NLO diagrams by analyzing their one-loop integrals.

Documentation:

Contents

Documentation:

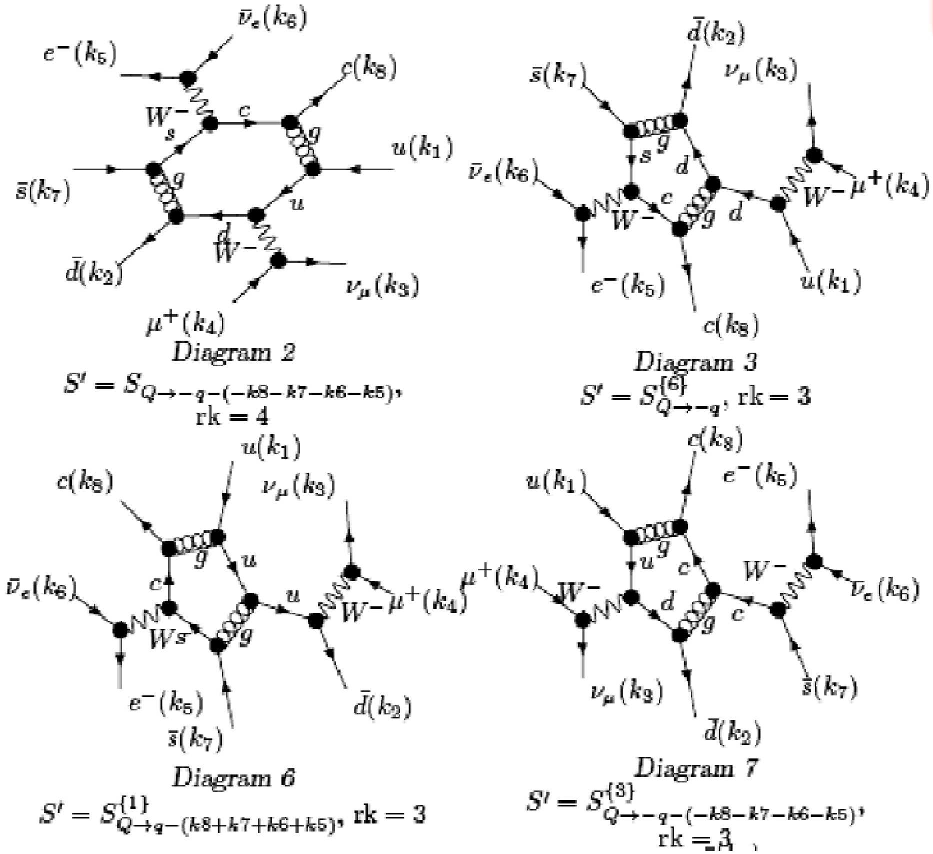
Building the code : check the details before the run



Building the code : check the details before the run

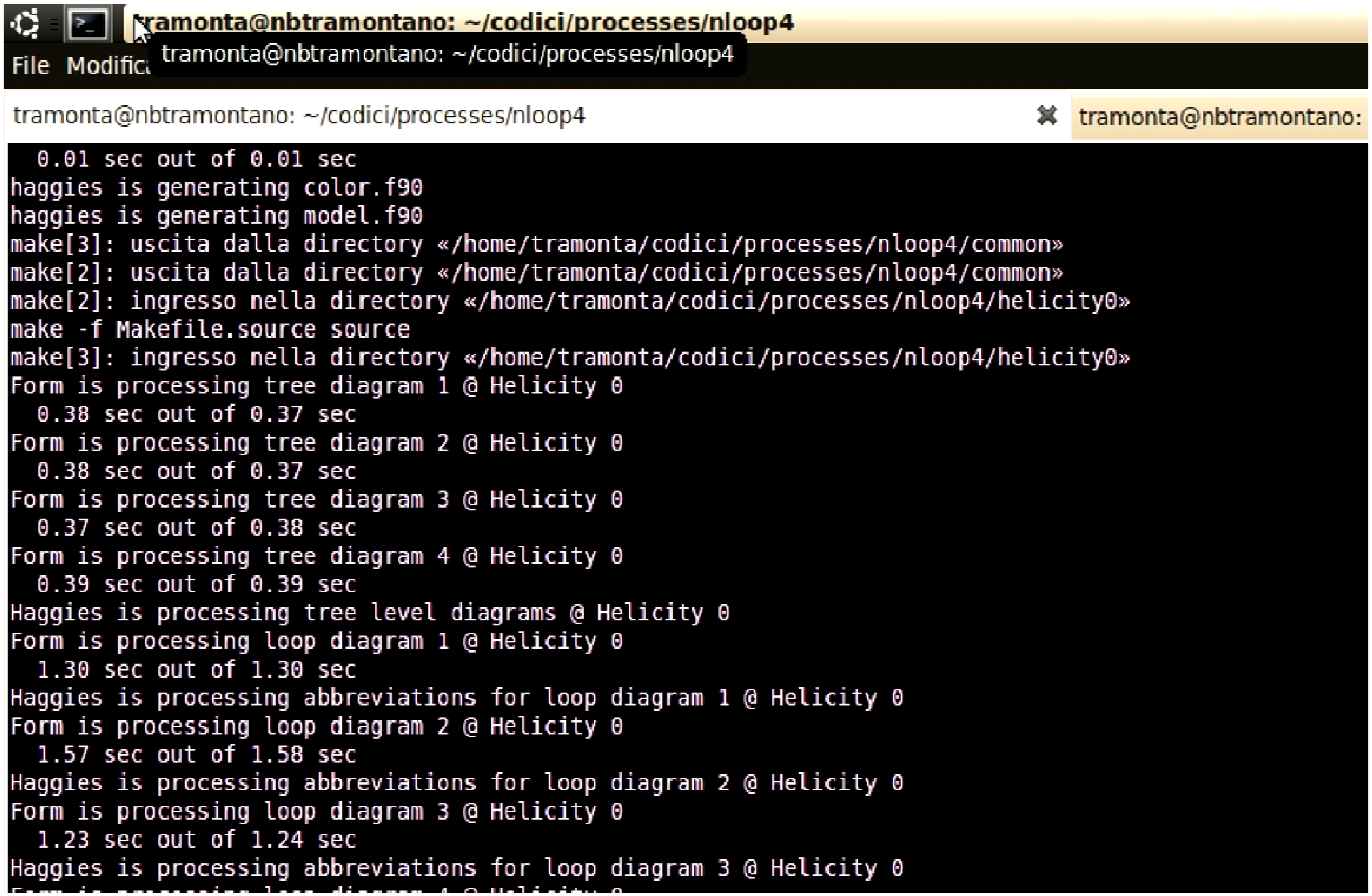
Documentation:

5.5.1 Diagrams (69)

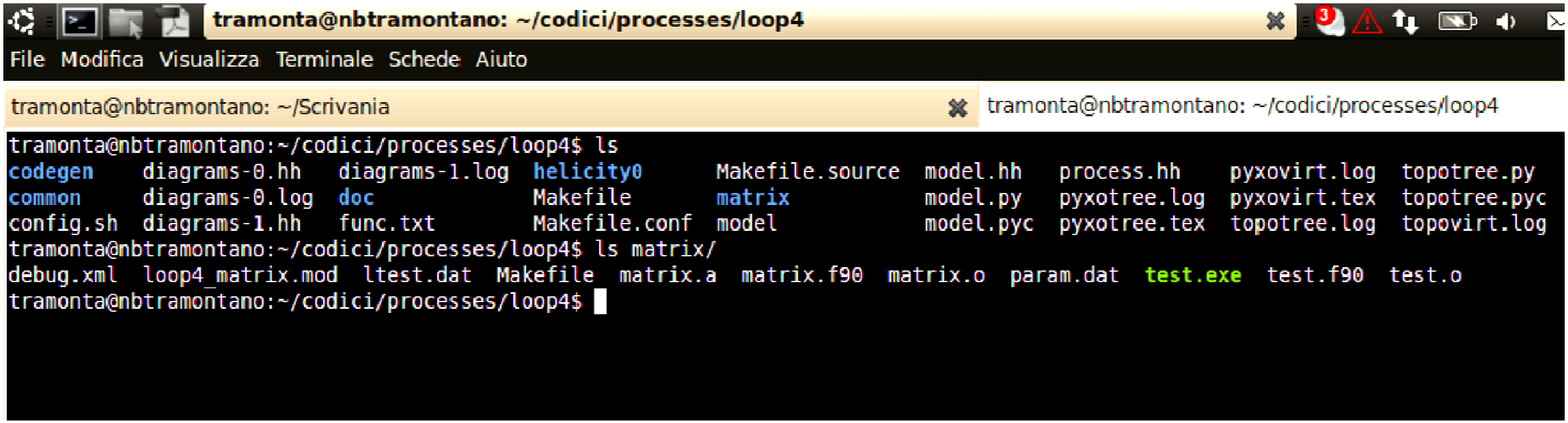


- Loop diagrams are grouped into sets of diagrams which share loop propagators
  - The numerator is constructed dinamically cut by cut
  - Strongly reduced calls to samurai
  - Cashing of the scalar integral and of a number of things calculated by samurai

Building the code : Spinney+ Haggies



Execution: all the code is ready, available as a library - option: Autotool




```
#              LO:  0.467952292378097E-19
# NLO, finite par  -15.91575130714159
# NLO, single pol    7.587050524202378
# NLO, double pol   -5.333333333333385
# IR,  single pol   -0.746282775357444
# IR,  double pol   -5.333333333333333
```

Melia, Melnikov, Rontsch, Zanderighi

```
# NLO, finite par  -15.91575
# NLO, single pol    7.587051
# NLO, double pol   -5.333333
```

After renormalization:
check of the poles with the one
form integrated dipoles built-in

```
#              LO:  0.467952292378097E-19
# NLO, finite par  -14.91575130714159
# NLO, single pol   -0.746282809130955
# NLO, double pol   -5.333333333333385
# IR,  single pol   -0.746282775357444
# IR,  double pol   -5.333333333333333
```

Timing (2.27 GHz, Gfortran -O2): 5.4ms

Alternative Path: the "Tensorial Way"

Tensorial Reconstruction at the Integrand Level

[with G.Heinrich, G.Ossola and T.Reiter (2010)]

In this work:

- We **tested** the methods for the **detection of instabilities**
- We proposed a **"rescue-system"** alternative to higher precision routines
- We proposed an **optimized reconstruction method**

Idea: tensorial reconstruction performed at the integrand level by means of a sampling in the integration momentum.

$$\mathcal{N}(q) = \sum_{r=0}^{R} C_{\mu_1 \dots \mu_r} q_{\mu_1} \dots q_{\mu_r} \implies \hat{N}(q)$$

$\hat{N}(q)$ is the "reconstructed numerator" written as a tensor
– numerically identical to the initial $\mathcal{N}(q)$ –

Tensorial reconstruction

rewrite numerator function as a linear combination of tensors

$$\mathcal{N}(q) \;=\; \sum_{r=0}^{R} C_{\mu_1\ldots\mu_r} q_{\mu_1} \ldots q_{\mu_r}$$

$$C_{\mu_1\ldots\mu_r} q_{\mu_1} \cdots q_{\mu_r} \;=\; \sum_{(i_1,i_2,i_3,i_4)\vdash r} \hat{C}^{(r)}_{i_1\, i_2\, i_3\, i_4} \cdot (q_1)^{i_1} (q_2)^{i_2} (q_3)^{i_3} (q_4)^{i_4}$$

determine the coefficients by sampling q in a bottom-up approach

Level 0:

$$q = (0,0,0,0)\ ,\ \mathcal{N}(0,0,0,0) \equiv \mathcal{N}^{(0)} = C_0$$

Level 1: 4 systems, each sampling a monomial depending on one component of q only

$$\mathcal{N}^{(1)}(q) \equiv \mathcal{N}(q) - \mathcal{N}^{(0)}$$

$$q = (x,0,0,0) \quad \Rightarrow \quad \mathcal{N}^{(1)}(q) \equiv x\, C_1 + x^2\, C_{11} + \ldots + x^R\, C_{\underbrace{11\ldots1}_{R \text{ times}}}$$

$$q = (0,y,0,0) \quad \Rightarrow \quad \mathcal{N}^{(1)}(q) \equiv y\, C_2 + y^2\, C_{22} + \ldots + y^R\, C_{\underbrace{22\ldots2}_{R \text{ times}}}$$

x, y… can be chosen to be Real numbers !

No paradigm that the amplitudes need to be sampled with complex q

Precision Tests

Use the decomposition of the numerator function $N(\bar{q})$ after determining all coefficients

$$
\begin{aligned}
N(\bar{q}) &= \sum_{i<<m}^{n-1} \Delta_{ijk\ell m}(\bar{q}) \prod_{h\neq i,j,k,\ell,m}^{n-1} \bar{D}_h + \sum_{i<<\ell}^{n-1} \Delta_{ijk\ell}(\bar{q}) \prod_{h\neq i,j,k,\ell}^{n-1} \bar{D}_h + \\
&+ \sum_{i<<k}^{n-1} \Delta_{ijk}(\bar{q}) \prod_{h\neq i,j,k}^{n-1} \bar{D}_h + \sum_{i<j}^{n-1} \Delta_{ij}(\bar{q}) \prod_{h\neq i,j}^{n-1} \bar{D}_h + \sum_{i}^{n-1} \Delta_{i}(\bar{q}) \prod_{h\neq i}^{n-1} \bar{D}_h
\end{aligned}
$$

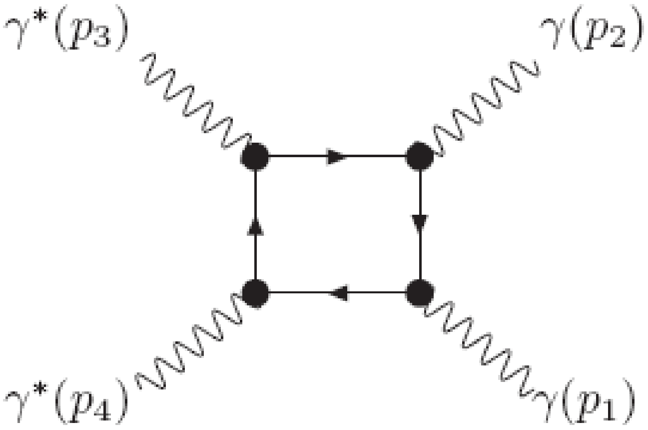
1. **Global** $(N = N)$**-test**
2. **Local** $(N = N)$**-test**
3. **Power-test**

Are those methods **reliable** in detecting **unstable phase space points**?

Approaching the Gram - I

- We approach a kinematic configuration which can lead to large cancellations

- Fermion loop with two massless and two massive vector particles

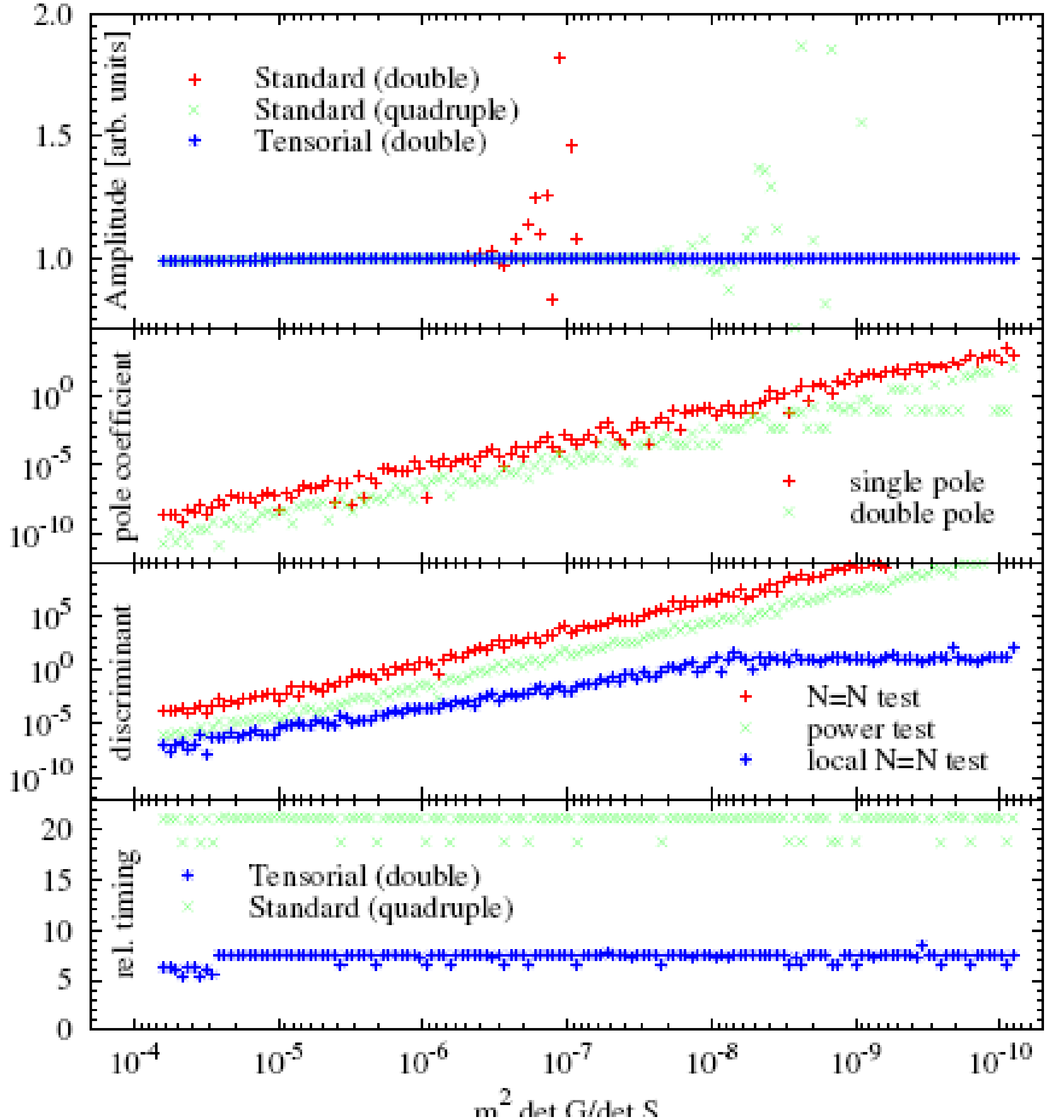


$$p_{1,2} = (E, 0, 0, \pm E) \quad p_{1,2}^2 = 0$$
$$p_{3,4} = (E, 0, \pm Q \sin\theta, \pm Q \cos\theta)$$
$$p_{3,4}^2 = m^2$$
$$E = \sqrt{m^2 + Q^2}$$

- The Gram-det vanishes when $Q \to 0$ (m and θ are fixed)

$$\det G = 32\, E^4 Q^2 \sin^2\theta$$

Approaching the Gram - II



Standard(double):
2x SAMURAI

Standard(quadruple):
2x Kin + integrals
4x Algorithm

Tensorial(double):
Reconstruction paired with numerical evaluation of tensor integras with GOLEM95

Ways to use the tensorial reconstruction

- **"Rescue-system"**
  - Unstable points will be automatically reprocessed using the tensorial decomposition + tensor integrals with `Golem 95`
  - Tensorial "master" integrals appears to be less costly than multi-precision routines
- **"Hybrid method"** for improved timing
  - The reduction of $\hat{N}(q)$ can be faster than that of $\mathcal{N}(q)$

# Lines	Time ratio "hybrid"/standard	
N	Rank = 4	Rank = 6
1	1.3	1.6
10	1.1	1.4
100	0.51	0.85
1000	0.30	0.59
10000	0.27	0.55

Example: Alternative reduction paths

Samurai/Tensorial Reduction/Golem95

$u\bar{u} \rightarrow d\bar{d}$

1. Evaluation with Samurai, sampling of diagram groups
2. Evaluation with Samurai, sampling of individual diagrams
3. Tensorial Reconstruction + Reduction of numetens with Samurai
4. Evaluation with Golem95

Method	finite part	single pole	double pole
1	-3.433053565229151	-14.62937842683104	-5.333333333333338
2	-3.433053565229129	-14.62937842683102	-5.333333333333342
3	-3.433053565229163	-14.62937842683104	-5.333333333333342
4	-3.433053565229146	-14.62937842683102	-5.333333333333332

More on the rational terms:

- Treatment strictly related the way the numerator function is furnished
  - Classified in two categories: R = R1 + R2
- R1 develops automatically performing the D-dimensional reduction of the tensors spanning the 4-dimensional part of the loop momentum
- R2 are present in the UV diagrams: bubbles, rank 2 and 3 triangles and rank4 boxes.
- At least two possibilities for R2 automatic computation:
  - for any fixed gauge theory calculate once and for all the contribution from all the diagrams that can generate R2 terms and define a set of tree level Feynman rules that give the R2 contribution for any process: **MadLoop approach**
  - Alternatively: construct the numerator function by implementing (few and universal) algebraic rules to get the R2 term on a diagram by diagram basis: **GoSam approach**

In GoSam we implemented two completely independent approaches for the R2 computation

- A full implementation of the **'t Hooft-Veltman** computation scheme including axial coupling renormalization constant (epsilon and mu2 terms are generated)

- An algebraic implementation producing **DRED** results (mu2 terms are generated):
  the first algebraic operations (with form) consist of combine all the powers of the loop momentum (using 4 dimensional algebra) and then expand them using:

$$\bar{\not{q}} = \not{q} + \not{\mu} \qquad \bar{q}^2 = q^2 - \mu^2$$

  for the fermion traces there is nothing special to do because the diagram generation (field-dressing of the topologies) already works along a reading point prescription

We also added transformation formulas to **CDR** so we provide output in **DRED** or **HV** or **CDR**

GoSam and Rational Terms $\mathbf{R}_2$

GoSam offers different options for calculation of $\mathbf{R}_2$

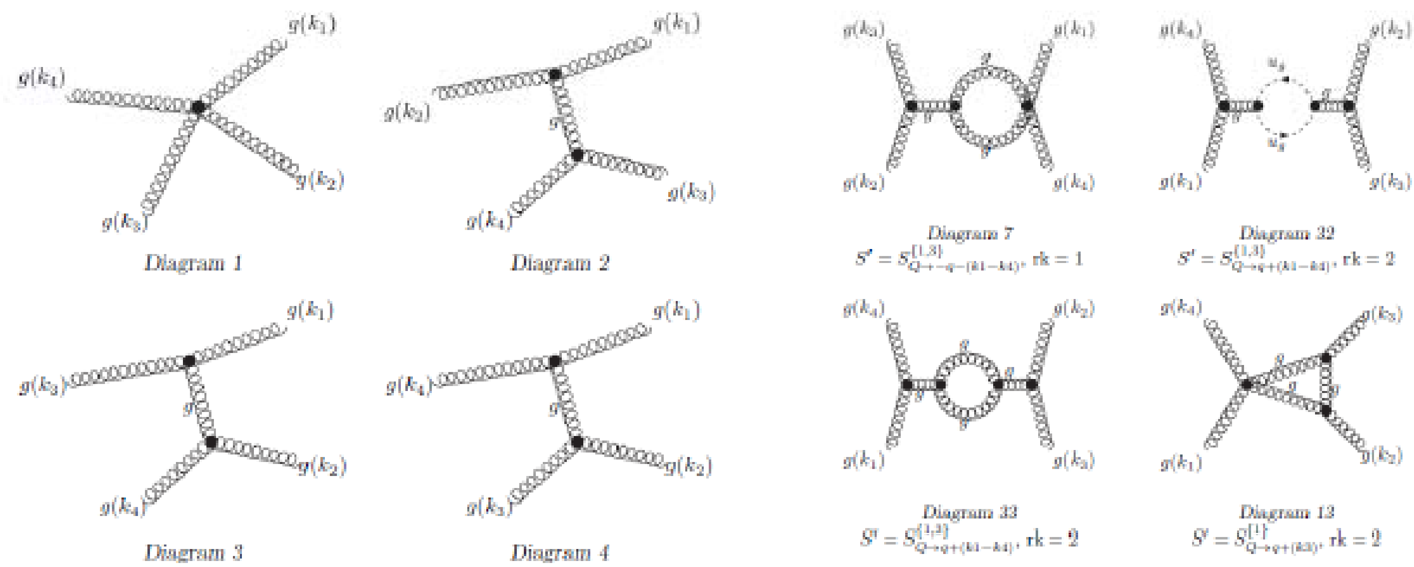
- **implicit**: R_2 terms are kept in the numerator and reduced at runtime using the d-dimensional decomposition of the numerator
- **explicit**: R_2 terms are calculated analytically (without entering in the numerical decomposition)
- **only**: only the R_2 term is kept in the final result (this option does not require any additional libraries)
- **off**: all R_2 terms are set to zero

Calculations tested with Golem/Samurai

- $u\,\bar{d} \to W^+\, s\bar{s} \to e^+\nu_e\, s\bar{s}$
- $u\,\bar{d} \to W^+\, gg \to e^+\nu_e gg$
- $d\,\bar{d} \to Z\, gg \to e^+e^-\, gg$
- $u\,\bar{d} \to W^+\, b\bar{b} \to e^+\nu_e\, b\bar{b}$ also with massive b's
- $u\,\bar{d} \to W^+\, g \to e^+\nu_e\, g$ EW corrections
- $e^+e^- \to Z \to d\,\bar{d}\,g$
- $\gamma\gamma \to \gamma\gamma\gamma\gamma$
- $q\,\bar{q} \to b\bar{b}b\bar{b}$
- $gg \to b\bar{b}b\bar{b}$
- $u\,\bar{d} \to W^+W^+\, s\bar{c} \to e^+\nu_e\mu^+\nu_\mu\, s\bar{c}$
- $u\,\bar{u} \to W^+W^-\, \bar{c}c \to e^-\bar{\nu}_e\mu^+\nu_\mu\, \bar{c}c$
- $u\,\bar{d} \to W^+W^-\, s\bar{c} \to e^-\bar{\nu}_e\mu^+\nu_\mu\, \bar{s}c$

- $d + g \to d + g$
- $d + \bar{d} \to t + \bar{t}$
- $b + g \to H + b$
- $u + \bar{u} \to g + \gamma$
- $u + g \to u + \gamma$
- $g + g \to g + \gamma$
- $g + g \to g + g$
- $g + g \to Z + g$
- $g + g \to Z + Z$
- $g + g \to W^+ + W^-$
- $b + g \to W^- + t$
- $g + g \to t + \bar{t}$

EXAMPLE: $gg \to gg$



	Golem/Samurai	hep-ph/0609054
LO	14.120983050796795	14.120983050796804
NLO/LO finite	-124.02475579423496	-124.02475579423495
NLO/LO $1/\epsilon$	44.003597347101028	44.003597347101035
NLO/LO $1/\epsilon^2$	-12.000000000000002	-12.000000000000000

Comparison with: hep-ph/0609054 Binoth, Guillet, Heinrich

EXAMPLE: $pp \to W^+W^+jj$

$$u\bar{d} \to \bar{c}se^+\nu_e\mu^+\nu_\mu$$

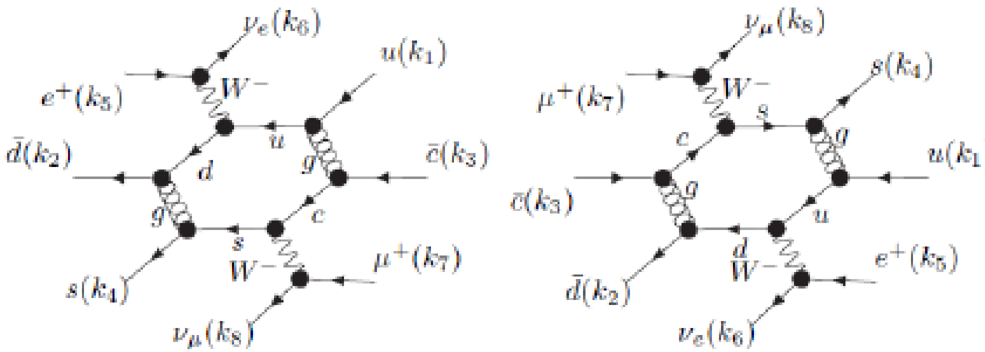


Diagram 1 *Diagram 2*

Helicities

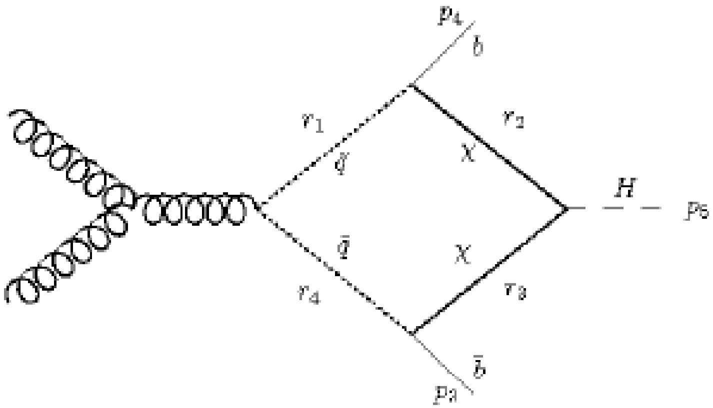
Index	1	2	3	4	5	6	7	8
0	−	+	+	−	+	−	+	−

Golem/Samurai (NLO/LO):

```
finite part 23.3596455167118
single pole 13.6255429251954
double pole -5.33333333333343
```

Comparison with Melia, Melnikov, Rontsch, Zanderighi

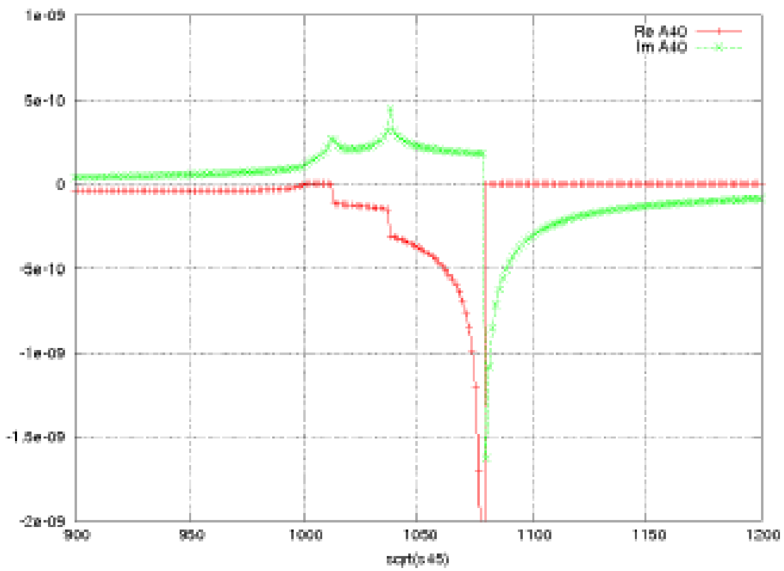
Example: GoSam, MSSM Higgs, and Complex Masses



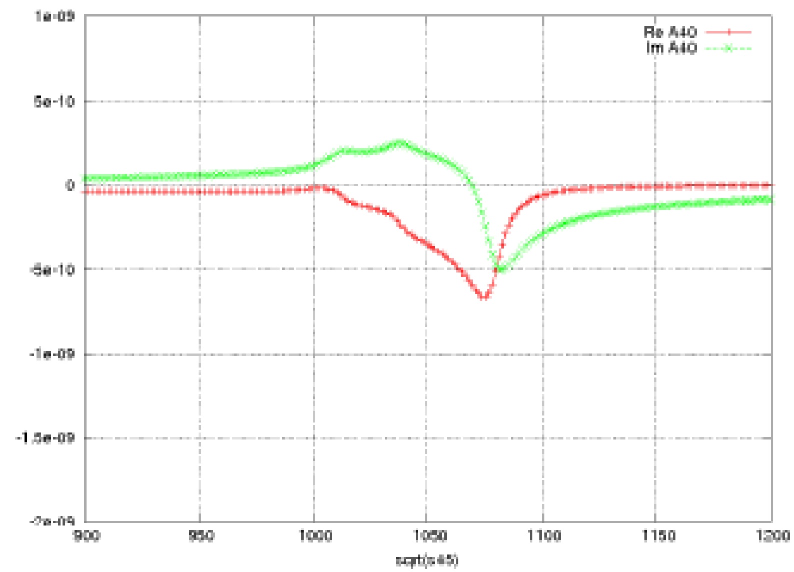
Production of a heavy neutral MSSM Higgs boson and a $\bar{b}b$ pair in gluon fusion.

The loop contains two squarks and two neutralinos

Real Masses



Complex Masses



Golem95C – arXiv:1101.5595

Example: $u\bar{d} \to W^+(e^+\nu_e)b\bar{b}$

	Timings:	30min generation+compilation
2.27GHz gfortran –O2		7msec running a single ps-point

PS-point:

```
vecs(1,:)= (/ 76.0843499,   0.00000000,  0.00000000,  76.0843499 /)
vecs(2,:)= (/ 1998.03313,   0.00000000,  0.00000000, -1998.03313 /)
vecs(3,:)= (/ 955.016763,   50.0258080,  17.0602115, -953.553032 /)
vecs(4,:)= (/ 194.222790,   4.35888776,  39.0630650, -190.204020 /)
vecs(5,:)= (/ 468.235447,   208.221739,  40.6257851, -417.390852 /)
vecs(6,:)= (/ 456.642482,  -262.606435, -96.7490617, -360.800877 /)
```

Setup:

```
ren scale: = 80.0 GeV
        mb = 4.75 GeV
        mt = 172.5 GeV
       VUD = 0.975
    widthW = 2.1054GeV
        gs = 1
        MW = 80.398 GeV
        MZ = 91.1876 GeV
        GF = 0.0000116639 GeV^-2
```

b-quark massive everywhere but in the bubble contributing to the gluon vacuum polarization

NLO/LO/ason2pi results:

MCFM:

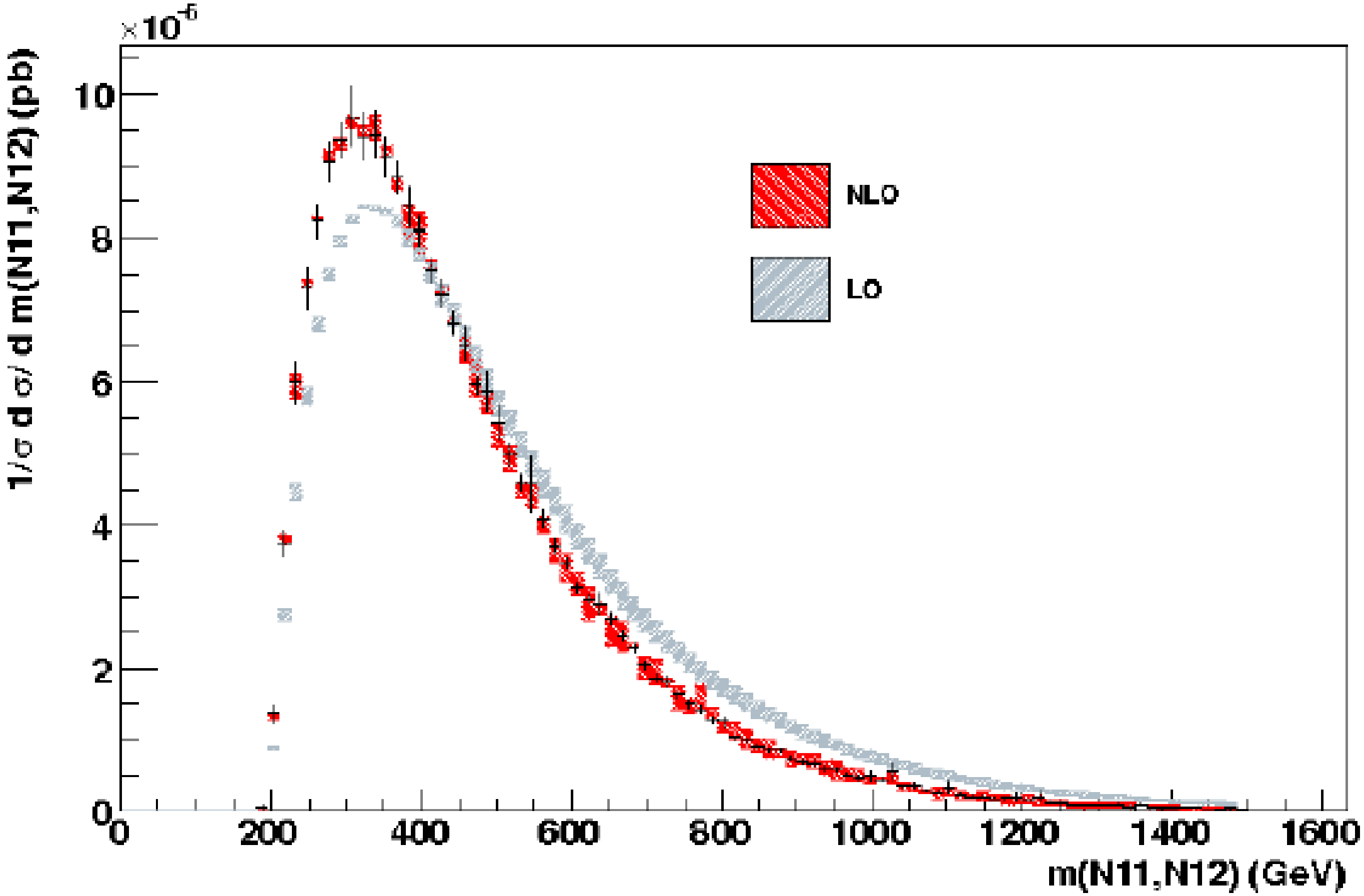
```
                LO:   1.8844347E-007
  NLO, finite par     41.217130
```

GOSAM:

```
#               LO:   1.8844347E-007
# NLO, finite par     41.217130
# NLO, single pol     26.603671
# NLO, double pol    -2.6666667
# IR,  single pol     26.603671
# IR,  double pol    -2.6666667
```

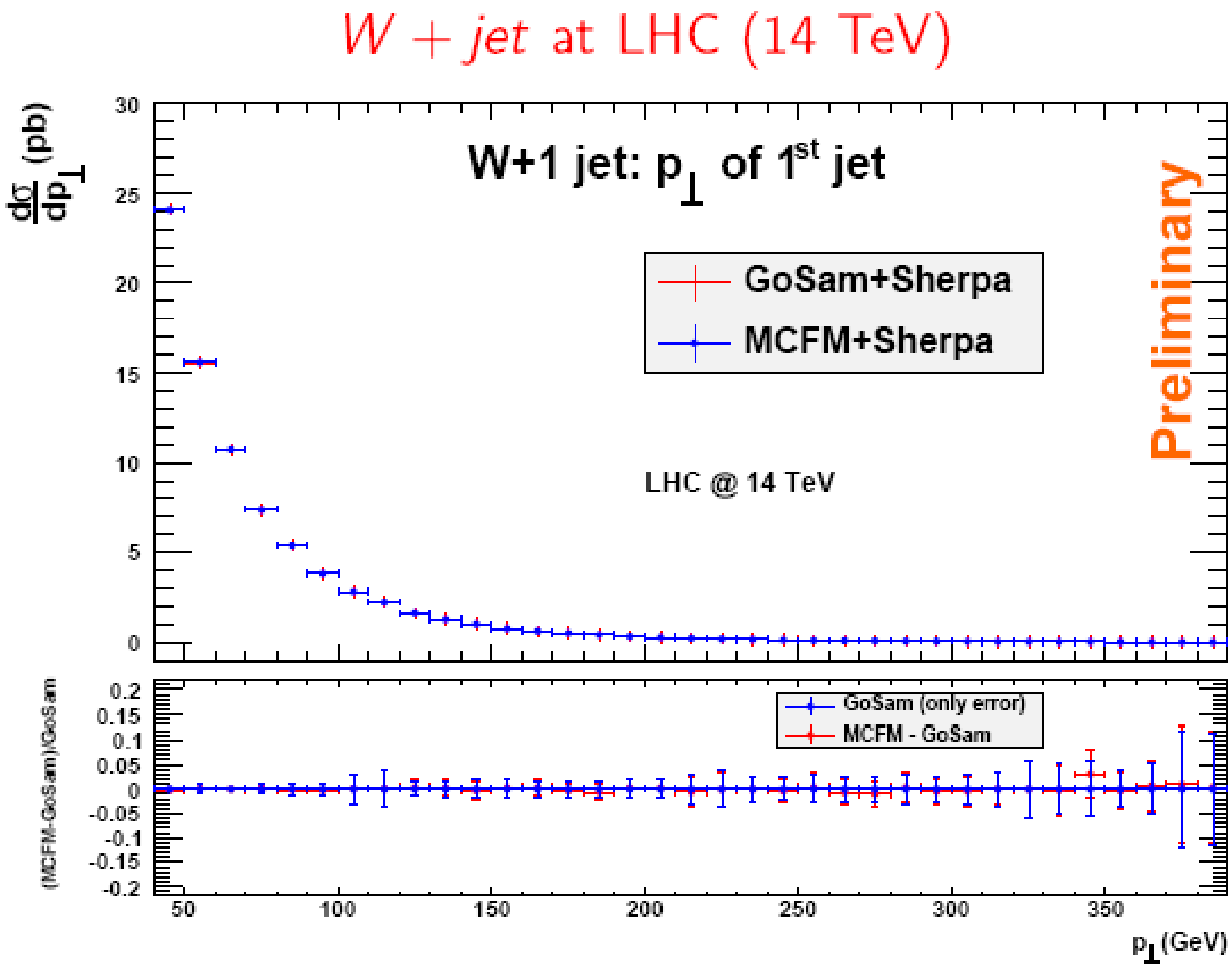
Example: GoSam and MSSM Neutralino

$pp \to \chi_1^0 \chi_1^0$



(Thanks to **Gavin Cullen** and **Nicolas Greiner**)

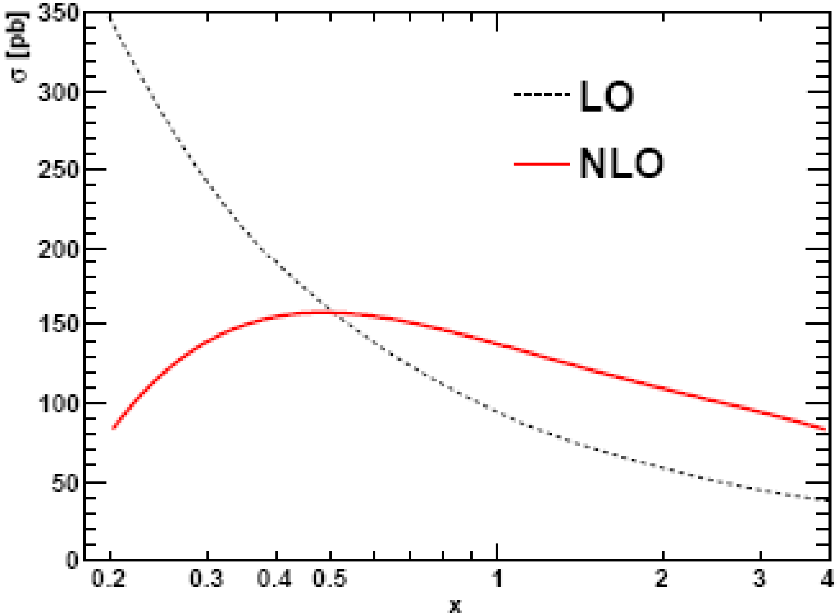
Example: GoSam + SHERPA

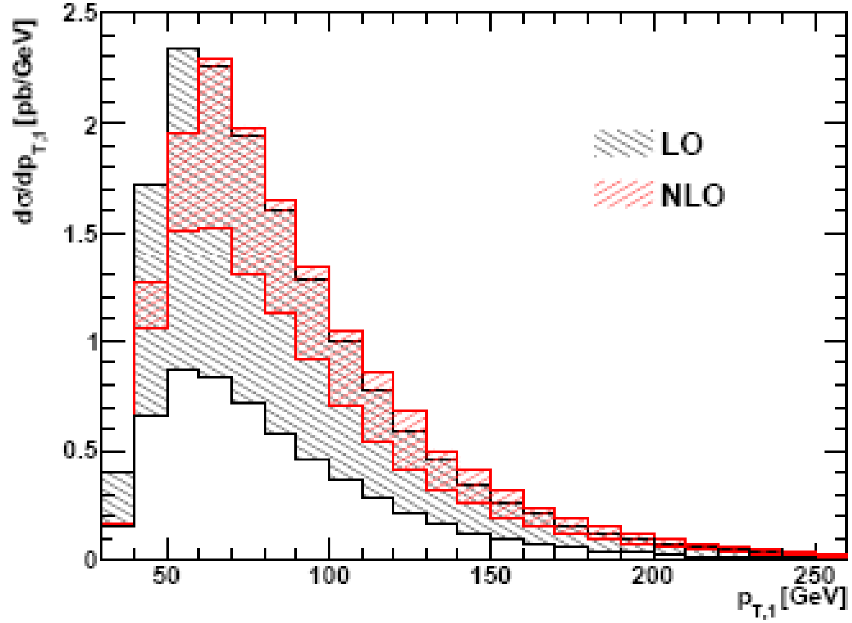


(Thanks to **Jennifer Archibald** and **Gionata Luisoni**)

Example: $pp \to b\bar{b}b\bar{b}$

by Greiner, Guffanti, Reiter, Reuter 1105.3624





- ✓ Born & Real: MadGraph
- ✓ Subtractions: MadDipole
- ✓ Virtuals: GoSam reweighted

Conclusions: Golem/Samurai

There are many valuable approaches/codes to One-Loop Calculations

GoSam is a flexible and broadly applicable tool

- it is based on Feynman diagrams
- it uses a d-dimensional reduction (no additional techniques required for rational terms)
- it will be publicly available, as soon as we complete the testing
- it uses some of the best techniques on the market

We look forward to **interacting/interfacing** with other tools

More results soon!