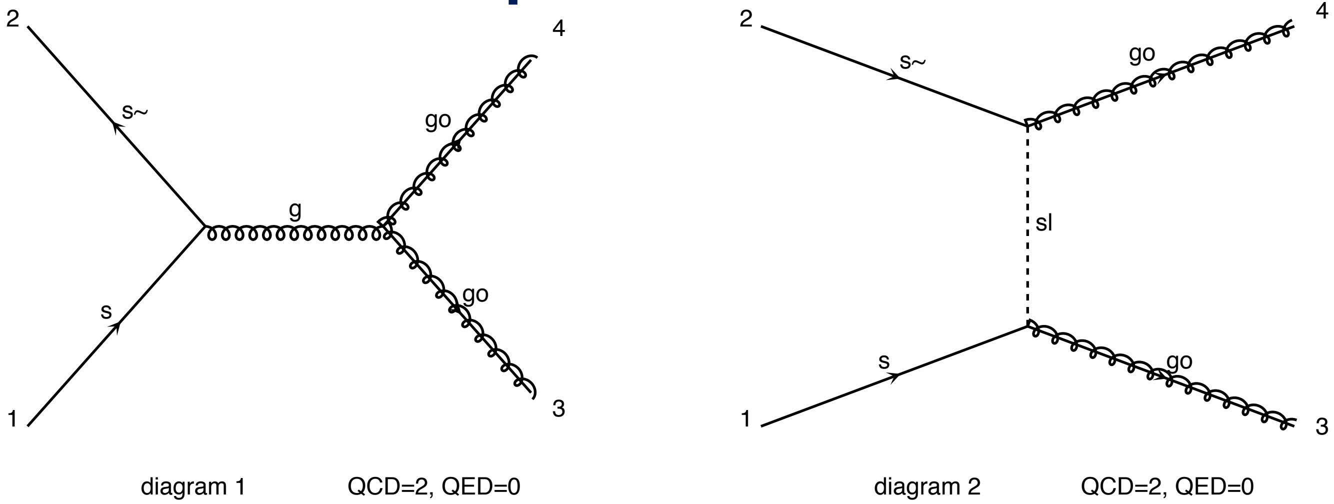
MadGraph5

Olivier Mattelaer
IPPP/Durham

Matrix-Element

Calculate a given process (e.g. gluino pair)

- Determine the production mechanism



- Evaluate the matrix-element

$|\mathcal{M}|^2$ ➡Need Feynman Rules!

- Phase-Space Integration

$$\sigma = \frac{1}{2s}\int |\mathcal{M}|^2 d\Phi(n)$$

Easy enough

Hard

Now

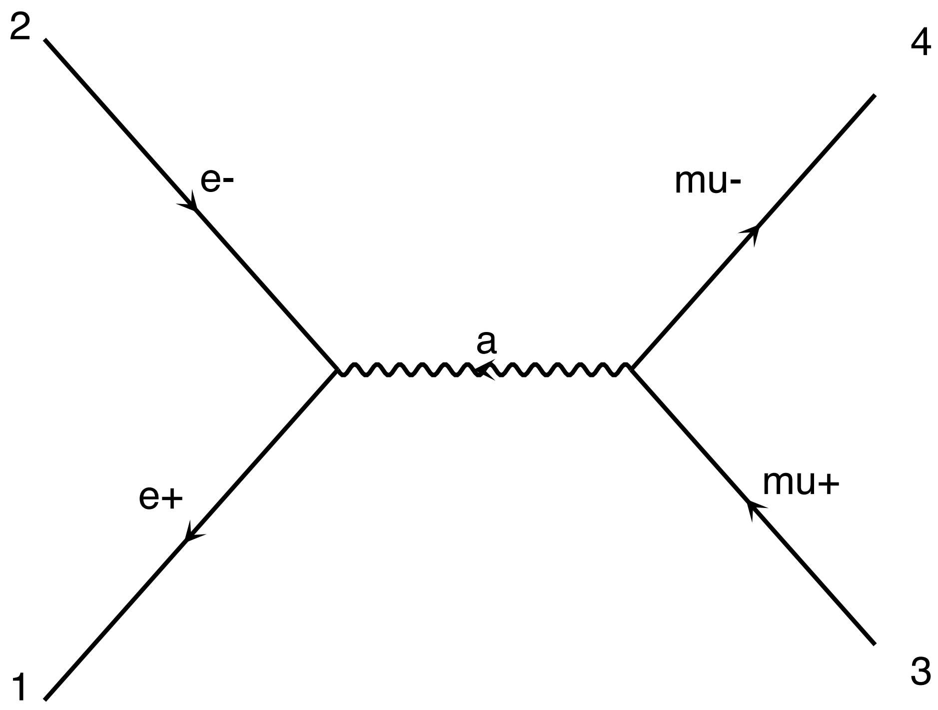
Very Hard (in general)

monday

Plan For Today

- Computation of the matrix-element
  - Tree Level
  - Loop
- Tools/functionality of MG5_aMC

Matrix Element



$$\mathcal{M} = e^2 (\bar{u}\gamma^\mu v)\frac{g_{\mu\nu}}{q^2}(\bar{v}\gamma^\nu u)$$

$$\frac{1}{4}\sum_{pol}|\mathcal{M}|^2 = \frac{1}{4}\sum_{pol}\mathcal{M}^*\mathcal{M}$$

$$\sum_{pol}\bar{u}u = \not{p} + m$$

$$\rightarrow \frac{e^4}{4q^4}Tr[\not{p}_1\gamma^\mu\not{p}_2\gamma^\nu]Tr[\not{p}_3\gamma_\mu\not{p}_4\gamma_\nu]$$

$$\rightarrow \frac{8e^4}{q^4}\left[(p_1.p_3)(p_2.p_4) + (p_1.p_4)(p_2.p_3)\right]$$

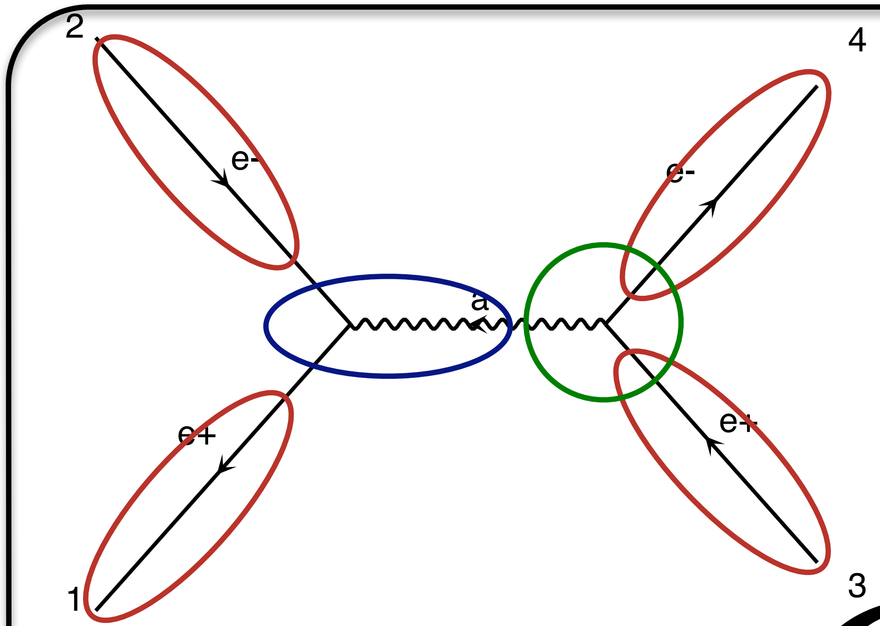
Very Efficient !!!

Only for 2→2 and 2→3

Because the number of terms rises as N^2

Idea

- Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> |$\mathcal{M}$|^2
  - ➡ Loop on Helicity and average the results



$$\mathcal{M} = (\bar{u}\, e\, \gamma^{\mu} v)\, \frac{g_{\mu\nu}}{q^2}\, (\bar{v}\, e\, \gamma^{\nu} u)$$

Numbers for given helicity and momenta

Calculate propagator wavefunctions

Finally evaluate amplitude (c-number)

Lines present in the code.

$$\begin{cases} \bar{v}_1 = fct(\vec{p}_1, m_1) \\ u_2 = fct(\vec{p}_2, m_2) \\ v_3 = fct(\vec{p}_3, m_3) \\ \bar{u}_4 = fct(\vec{p}_4, m_4) \\ W_a = fct(\bar{v}_1, u_2, m \\ \mathcal{M} = fct(v_3, \bar{u}_4, W \end{cases}$$

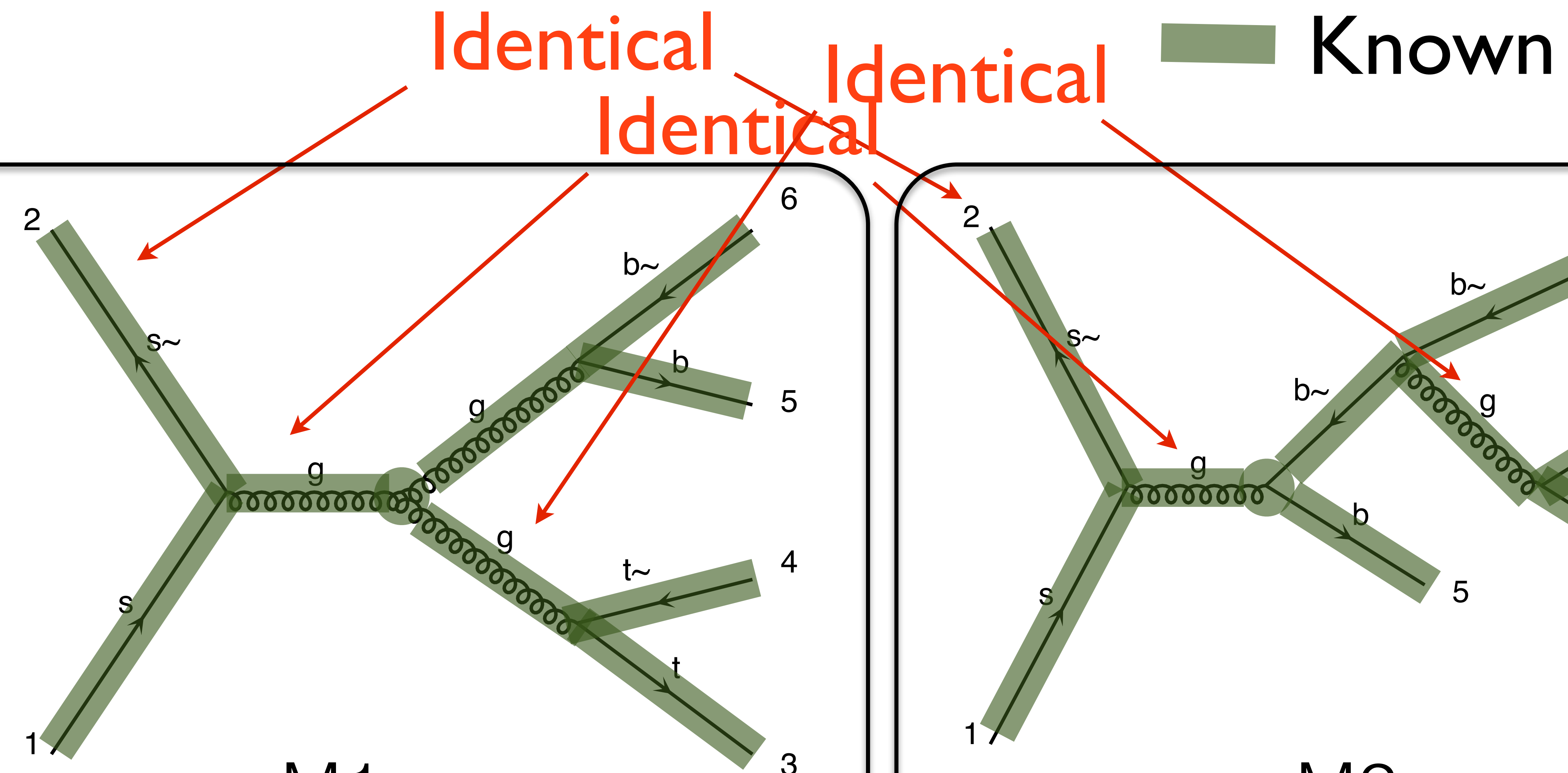
$$u(p) = \begin{pmatrix} \omega_{-\lambda}(p)\chi_\lambda(\vec{p}) \\ \omega_\lambda(p)\chi_\lambda(\vec{p}) \end{pmatrix}$$

$$\omega_\pm(p) \equiv \sqrt{E \pm |\vec{p}|}.$$

$$\chi_+(\vec{p}) = \frac{1}{\sqrt{2|\vec{p}|(|\vec{p}|+p_z)}} \begin{pmatrix} |\vec{p}| + p_z \\ p_x + ip_y \end{pmatrix},$$

$$\chi_-(\vec{p}) = \frac{1}{\sqrt{2|\vec{p}|(|\vec{p}|+p_z)}} \begin{pmatrix} -p_x + ip_y \\ |\vec{p}| + p_z \end{pmatrix}.$$

Real case



M1

Number of routines: 10

2(N+1)

M2

Number of routines: 10

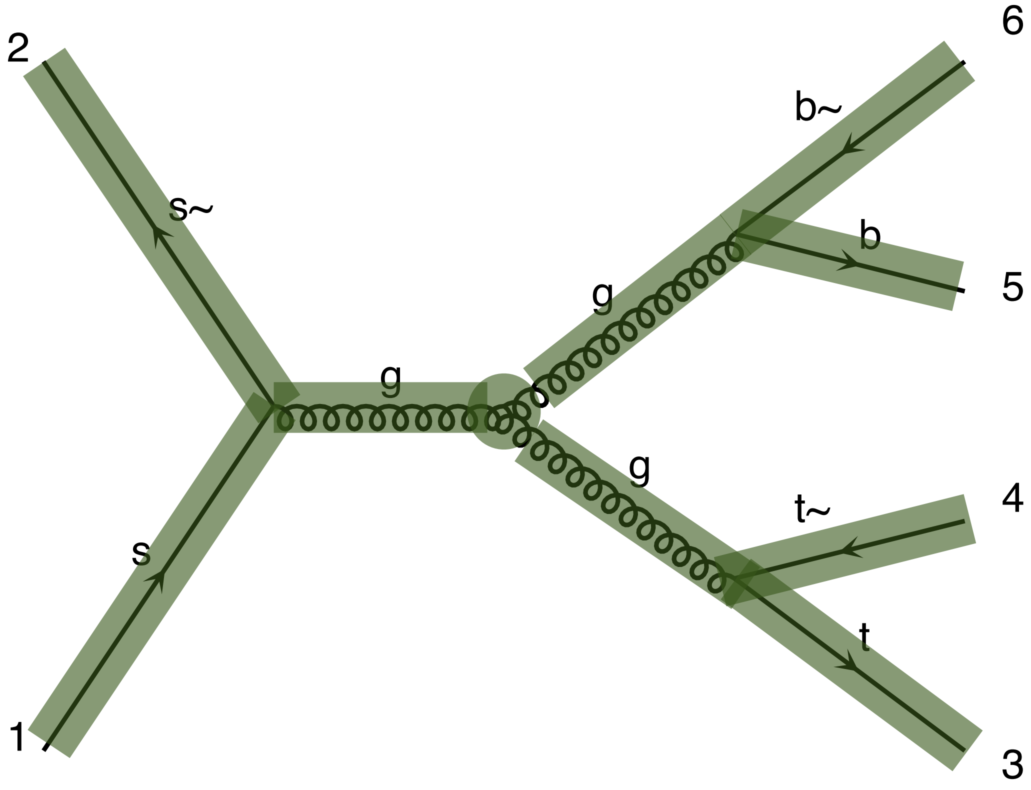
2(N+1)

Number of routines for both: 12

$$|M|^2 = |M_1 + M_2|^2$$

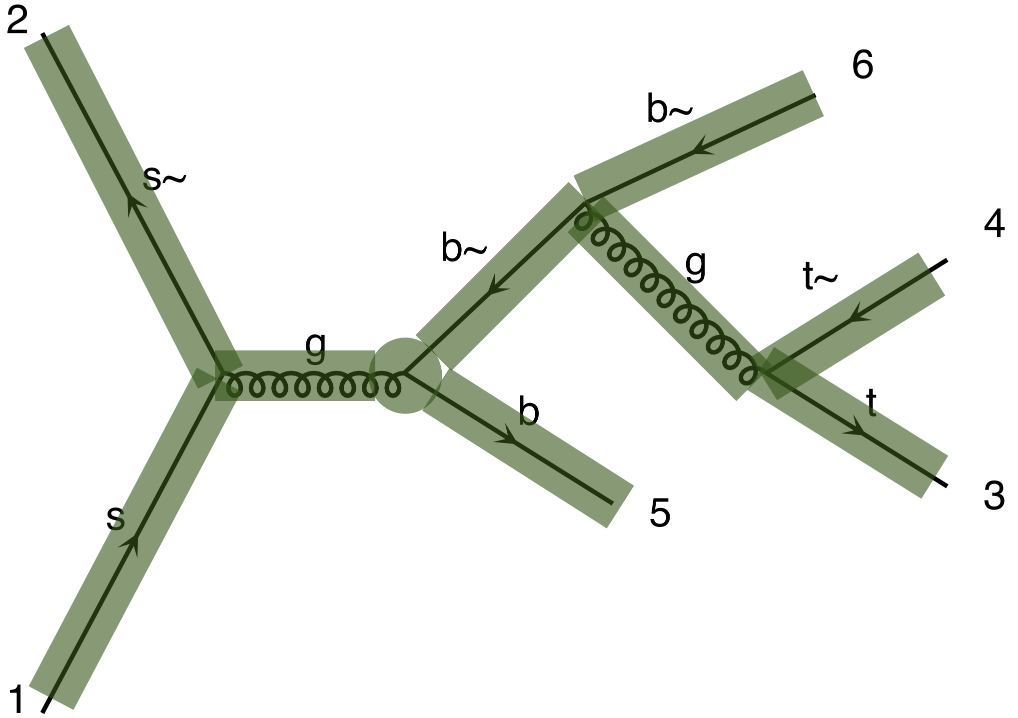
Real case

🟩 Known



Number of routines: 10

2(N+1)



Number of routines: 10

2(N+1)

Number of routines for both: 12

$$N!*2(N+1) \longrightarrow N! \xrightarrow{\text{recursion}} 2^N$$

HELAS

- **Original HELicity Amplitude Subroutine library**
  [Murayama, Watanabe, Hagiwara]
- **One routine by Lorentz structure**
  - ➡**MSSM** [cho, al] hep-ph/0601063 (2006)
  - ➡**HEFT** [Frederix] (2007)
  - ➡**Spin 2** [Hagiwara, al] 0805.2554 (2008)
  - ➡**Spin 3/2** [Mawatari, al] 1101.1289 (2011)

Chiral Perturbation

BNV Model

SLIH

Effective Field Theory

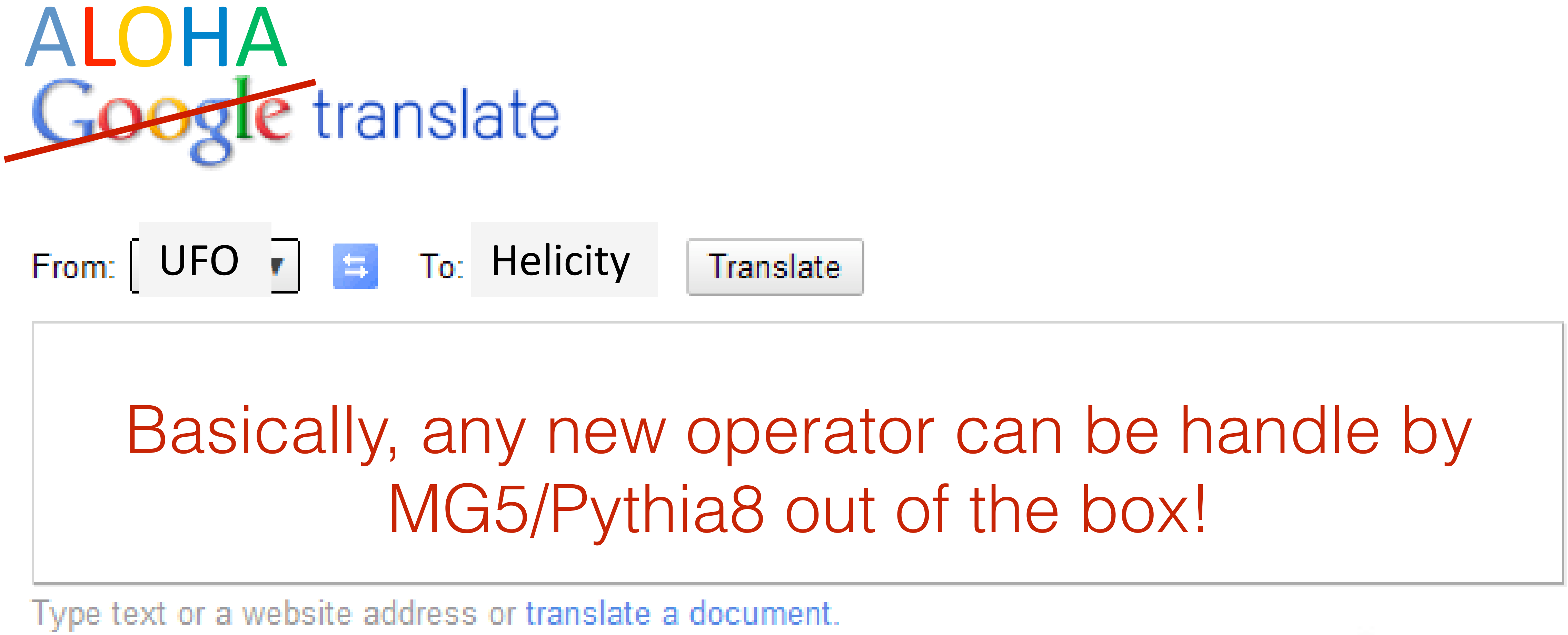
NMSSM

Chromo-magnetic operator

Full HEFT

Black Holes

ALOHA



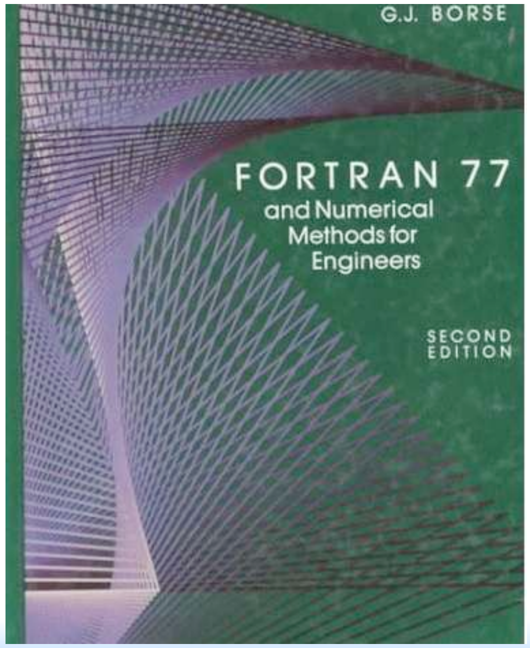
ALOHA

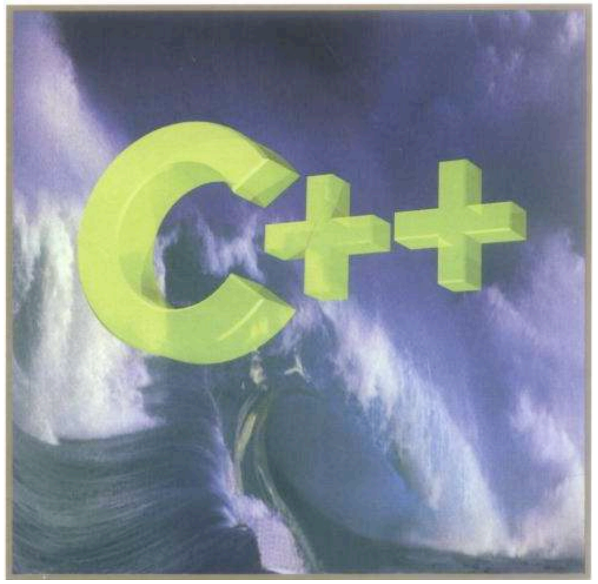
~~Google~~ translate

From: UFO ▾ ⇆ To: Helicity Translate

Basically, any new operator can be handle by MG5/Pythia8 out of the box!

Type text or a website address or translate a document.





To Remember

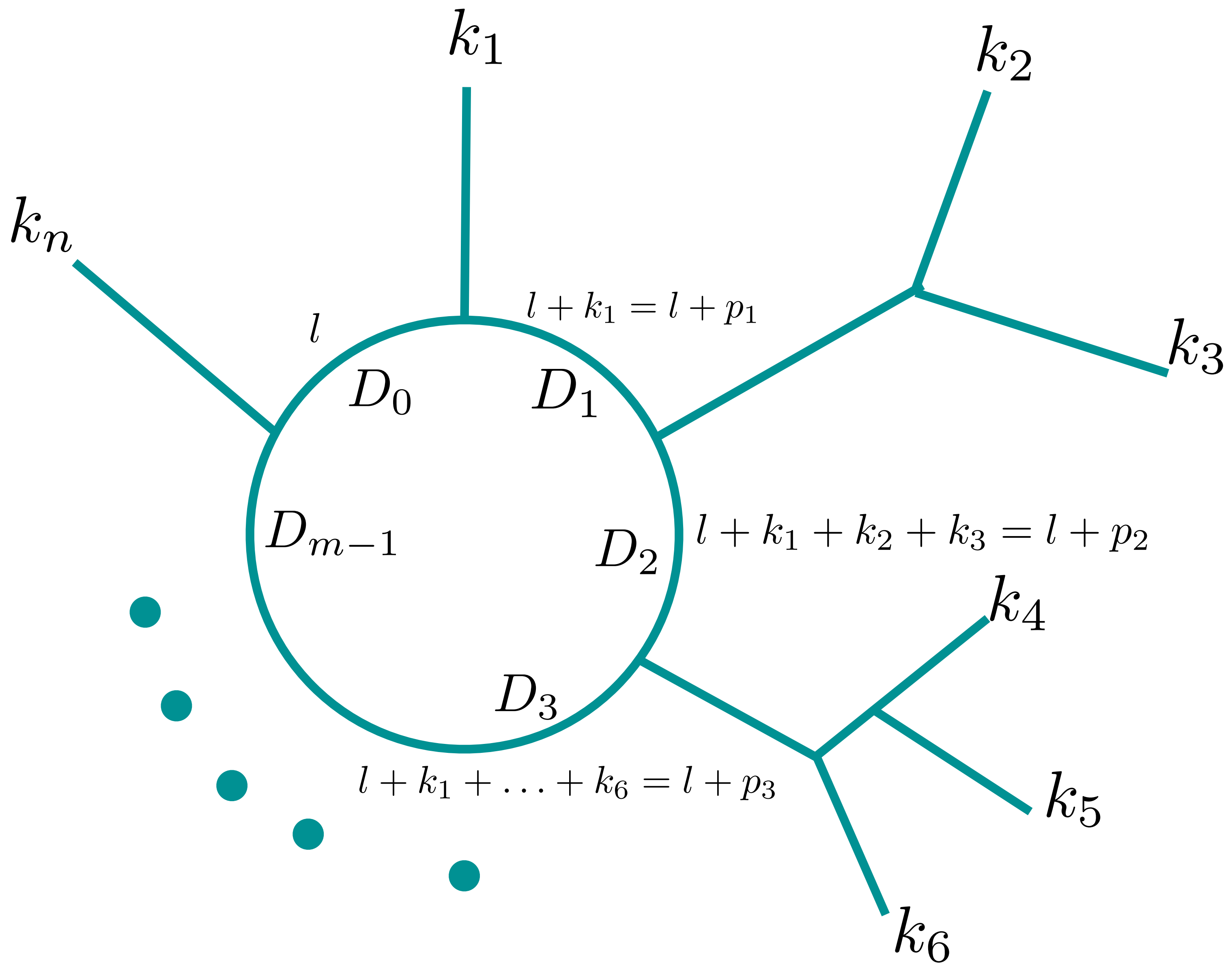
- Numerical computation faster than analytical computation
- We are able to compute matrix-element
    - ➡ for large number of final state
    - ➡ for any BSM theory

Loop Computation

Loop Computation

One Loop

- Consider this *m*-point loop diagram with *n* external momenta



- The integral to compute is

$$\int d^dl\,\frac{N(l)}{D_0D_1D_2\cdots D_{m-1}}$$

$$D_i=(l+p_i)^2-m_i^2$$

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Basis of scalar integrals

- The a, b, c, d and R coefficients depend only on external parameters and momenta

$$\mathcal{M}^{\text{1-loop}} = \sum_{i_0<i_1<i_2<i_3} d_{i_0 i_1 i_2 i_3} \text{Box}_{i_0 i_1 i_2 i_3}$$

$$+ \sum_{i_0<i_1<i_2} c_{i_0 i_1 i_2} \text{Triangle}_{i_0 i_1 i_2}$$

$$+ \sum_{i_0<i_1} b_{i_0 i_1} \text{Bubble}_{i_0 i_1}$$

$$+ \sum_{i_0} a_{i_0} \text{Tadpole}_{i_0}$$

$$+ R + \mathcal{O}(\epsilon)$$

$$D_i = (l+p_i)^2 - m_i^2$$

$$\text{Tadpole}_{i_0} = \int d^d l \frac{1}{D_{i_0}}$$

$$\text{Bubble}_{i_0 i_1} = \int d^d l \frac{1}{D_{i_0} D_{i_1}}$$

$$\text{Triangle}_{i_0 i_1 i_2} = \int d^d l \frac{1}{D_{i_0} D_{i_1} D_{i_2}}$$

$$\text{Box}_{i_0 i_1 i_2 i_3} = \int d^d l \frac{1}{D_{i_0} D_{i_1} D_{i_2} D_{i_3}}$$

- All these scalar integrals are known and available in computer libraries (FF [v. Oldenborgh], QCDLoop [Ellis, Zanderighi], OneLOop [v. Hameren])

Divergences

- The a, b, c, d and R coefficients depend only on external parameters and momenta

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = &\sum_{i_0<i_1<i_2<i_3} d_{i_0i_1i_2i_3}\,\text{Box}_{i_0i_1i_2i_3}\\ &+\sum_{i_0<i_1<i_2} c_{i_0i_1i_2}\,\text{Triangle}_{i_0i_1i_2}\\ &+\sum_{i_0<i_1} b_{i_0i_1}\,\text{Bubble}_{i_0i_1}\\ &+\sum_{i_0} a_{i_0}\,\text{Tadpole}_{i_0}\\ &+R+\mathcal{O}(\epsilon)\end{aligned}$$

$$D_i = (l+p_i)^2 - m_i^2$$

$$\begin{aligned}\text{Tadpole}_{i_0} &= \int d^dl \frac{1}{D_{i_0}}\\ \text{Bubble}_{i_0i_1} &= \int d^dl \frac{1}{D_{i_0}D_{i_1}}\\ \text{Triangle}_{i_0i_1i_2} &= \int d^dl \frac{1}{D_{i_0}D_{i_1}D_{i_2}}\\ \text{Box}_{i_0i_1i_2i_3} &= \int d^dl \frac{1}{D_{i_0}D_{i_1}D_{i_2}D_{i_3}}\end{aligned}$$

➡The coefficients d, c, b and a are finite and do not contain poles in 1/ϵ

➡The 1/ϵ dependence is in the scalar integrals (and the UV renormalization)

➡When we have solved this system (and included the UV renormalization) we have the full dependence on the soft/collinear divergences in terms of coefficients in front of the poles. These divergences should cancel against divergences in the real emission corrections (according to KLN theorem)

$$\text{Virtual} \sim v_0 + \frac{v_1}{\epsilon} + \frac{v_2}{\epsilon^2}$$

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

- Passarino-Veltman
- OPP

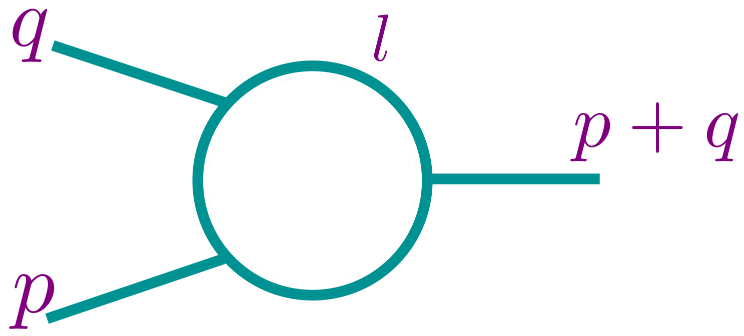
Standard Approach

- Passarino-Veltman reduction:

$$\int d^d l \, \frac{N(l)}{D_0 D_1 D_2 \cdots D_{m-1}} \to \sum_i \text{coeff}_i \int d^d l \, \frac{1}{D_0 D_1 \cdots}$$

  - Reduce a general integral to "scalar integrals" by "completing the square"

- Let's do an example:
Suppose we want to calculate this triangle integral



$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l \cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p \cdot p & 2p \cdot q \\ 2p \cdot q & 2q \cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l \cdot p] \\ [2l \cdot q] \end{pmatrix}$$

Gram Determinant: G

Passarino-Veltman

Resolution (dropping the mass)

Resolution (dropping the mass)

- express the integral as simpler integral
- contracting with 2*p and 2*q

$$\int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2} = \int \frac{d^n l}{(2\pi)^n} \frac{(l+p)^2 - l^2 - p^2}{l^2 (l+p)^2 (l+q)^2}$$

$$[2l \cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p \cdot p & 2p \cdot q \\ 2p \cdot q & 2q \cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l \cdot p] \\ [2l \cdot q] \end{pmatrix}$$

Gram Determinant: G

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

Gram Determinant: G

Resolution (dropping the mass)

- express the integral as simpler integral

$$\int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2} = \int \frac{d^n l}{(2\pi)^n}\frac{(l+p)^2-l^2-p^2}{l^2(l+p)^2(l+q)^2}$$

$$= \int \frac{d^n l}{(2\pi)^n}\frac{1}{l^2(l+q)^2} - \int \frac{d^n l}{(2\pi)^n}\frac{1}{(l+p)^2(l+q)^2} - p^2 \int \frac{d^n l}{(2\pi)^n}\frac{1}{l^2(l+p)^2(l+q)^2}$$

Scalar Integral: Know analytically

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^nl}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix}2p_\mu\\ 2q_\mu\end{pmatrix}(p^\mu\ \ q^\mu)\begin{pmatrix}C_1\\ C_2\end{pmatrix} = \boxed{\begin{pmatrix}2p\cdot p & 2p\cdot q\\ 2p\cdot q & 2q\cdot q\end{pmatrix}}\begin{pmatrix}C_1\\ C_2\end{pmatrix} = \boxed{\begin{pmatrix}[2l\cdot p]\\ [2l\cdot q]\end{pmatrix}}$$

Gram Determinant: G

Final Step

- Inverting the Gram Determinant

$$\begin{pmatrix}C_1\\ C_2\end{pmatrix} = G^{-1}\begin{pmatrix}[2l\cdot p]\\ [2l\cdot q]\end{pmatrix}$$

- We have an expression in term of scalar integral

$$\int \frac{d^nl}{(2\pi)^n}\frac{l^\mu}{(l^2-m_1^2)((l+p)^2-m_2^2)((l+q)^2-m_3^2)} = (\ p^\mu\ \ \ q^\mu\ )\begin{pmatrix}C_1\\ C_2\end{pmatrix}$$

🟪 Already computed

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

- Passarino-Veltman
- OPP

OPP Reduction

- The decomposition to scalar integrals presented before works at the level of the integrals

$$
\begin{aligned}
\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0i_1i_2i_3}\,\text{Box}_{i_0i_1i_2i_3} \\
& + \sum_{i_0<i_1<i_2} c_{i_0i_1i_2}\,\text{Triangle}_{i_0i_1i_2} \\
& + \sum_{i_0<i_1} b_{i_0i_1}\,\text{Bubble}_{i_0i_1} \\
& + \sum_{i_0} a_{i_0}\,\text{Tadpole}_{i_0} \\
& + R + \mathcal{O}(\epsilon)
\end{aligned}
$$

- If we would know a similar relation at the integrand level, we would be able to manipulate the integrands and extract the coefficients without doing the integrals

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \boxed{\tilde{d}_{i_0i_1i_2i_3}(l)} \right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \boxed{\tilde{c}_{i_0i_1i_2}(l)} \right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \boxed{\tilde{b}_{i_0i_1}(l)} \right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \boxed{\tilde{a}_{i_0}(l)} \right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

Spurious term

spurious terms

- The functional form of the spurious terms is known (it depends on the rank of the integral and the number of propagators in the loop) [del Aguila, Pittau 2004]
  - for example, a box coefficient from a rank 1 numerator is

$$\tilde{d}_{i_0 i_1 i_2 i_3}(l) = \tilde{d}_{i_0 i_1 i_2 i_3}\, \epsilon^{\mu\nu\rho\sigma}\, l^\mu p_1^\nu p_2^\rho p_3^\sigma$$

  (remember that p_i is the sum of the momentum that has entered the loop so far, so we always have $p_0 = 0$)

  - The integral is zero

$$\int d^d l \frac{\tilde{d}_{i_0 i_1 i_2 i_3}(l)}{D_0 D_1 D_2 D_3} = \tilde{d}_{i_0 i_1 i_2 i_3} \int d^d l \frac{\epsilon^{\mu\nu\rho\sigma}\, l^\mu p_1^\nu p_2^\rho p_3^\sigma}{D_0 D_1 D_2 D_3} = 0$$

How it works...

$$N(l) = \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i$$

$$+ \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i$$

$$+ \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i$$

$$+ \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i$$

$$+\tilde{P}(l) \prod_{i}^{m-1} D_i$$

$$= 0$$

To solve the OPP reduction, choosing special values for the loop momenta helps a lot

For example, choosing *l* such that

$$D_0(l^\pm) = D_1(l^\pm) =$$
$$= D_2(l^\pm) = D_3(l^\pm) = 0$$

sets all the terms in this equation to zero except the first line

There are two (complex) solutions to this equation due to the quadratic nature of the propagators

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now we choose l such that

$$D_0(l^i) = D_1(l^i) = D_2(l^i) = 0$$

sets all the terms in this equation to zero except the first and second line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now, choosing *l* such that

$$D_0(l^i) = D_1(l^i) = 0$$

sets all the terms in this equation to zero except the first, second and third line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now, choosing *l* such that

$$D_1(l^i) = 0$$

sets the last line to zero

🟪 Coefficient computed in a previous step

How it works...

$$N(l) = \boxed{\sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i}$$

$$+ \boxed{\sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i}$$

$$+ \boxed{\sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i}$$

$$+ \boxed{\sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i}$$

$$+\tilde{P}(l) \prod_{i}^{m-1} D_i$$

Now, choosing arbitrary *l*

🟪 Coefficient computed in a previous step

How it works...

$$N(l) = \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i$$

$$+ \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i$$

$$+ \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i$$

$$+ \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i$$

$$+ \tilde{P}(l) \prod_{i}^{m-1} D_i$$

We have our Numerator!

🟪 Coefficient computed in a previous step

d dimensions

- In the previous consideration I was very sloppy in considering if we are working in 4 or d dimensions
- In general, external momenta and polarization vectors are in 4 dimensions; only the loop momentum is in d dimensions

- To be more correct, we compute the integral

$$\int d^d l \, \frac{N(l, \tilde{l})}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \cdots \bar{D}_{m-1}}$$

$$\boxed{\underset{\text{d dim}}{\underset{\nearrow}{\bar{l}}} = \underset{\text{4 dim}}{\underset{\uparrow}{l}} + \underset{\text{epsilon dim}}{\underset{\nwarrow}{\tilde{l}}}}$$

$$\bar{D}_i = (\bar{l} + p_i)^2 - m_i^2 = (l + p_i)^2 - m_i^2 + \tilde{l}^2 = D_i + \tilde{l}^2$$

$$l \cdot \tilde{l} = 0 \qquad \bar{l} \cdot p_i = l \cdot p_i \qquad \bar{l} \cdot \bar{l} = l \cdot l + \tilde{l} \cdot \tilde{l}$$

Implications

- The decomposition in terms of scalar integrals has to be done in d dimensions
- This is why the rational part R is needed

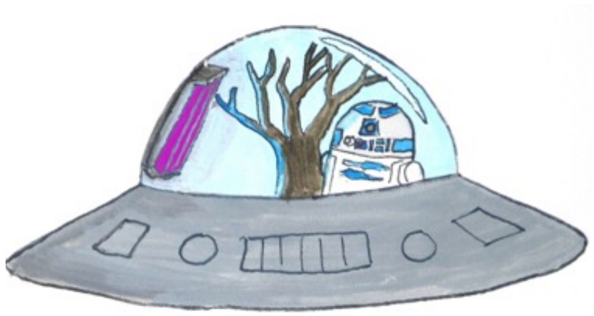
$$\begin{aligned}
&\sum_{0\leq i_0<i_1<i_2<i_3}^{m-1} d(i_0 i_1 i_2 i_3) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}\bar{D}_{i_1}\bar{D}_{i_2}\bar{D}_{i_3}} \\
&+ \sum_{0\leq i_0<i_1<i_2}^{m-1} c(i_0 i_1 i_2) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}\bar{D}_{i_1}\bar{D}_{i_2}} \\
&+ \sum_{0\leq i_0<i_1}^{m-1} b(i_0 i_1) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}\bar{D}_{i_1}} \\
&+ \sum_{i_0=0}^{m-1} a(i_0) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}} \\
&+ R\,.
\end{aligned}$$

Rational terms

- In the OPP method, they are split into two contributions, generally called

$$R = \boxed{R_1} + \boxed{R_2}$$

- Both have their origin in the UV part of the model, but only R1 can be directly computed in the OPP reduction and is given by the CutTools program
  - R1: originates from the propagator (calculate by CutTools)
  - R2: originates from the numerator (need in the model)



Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

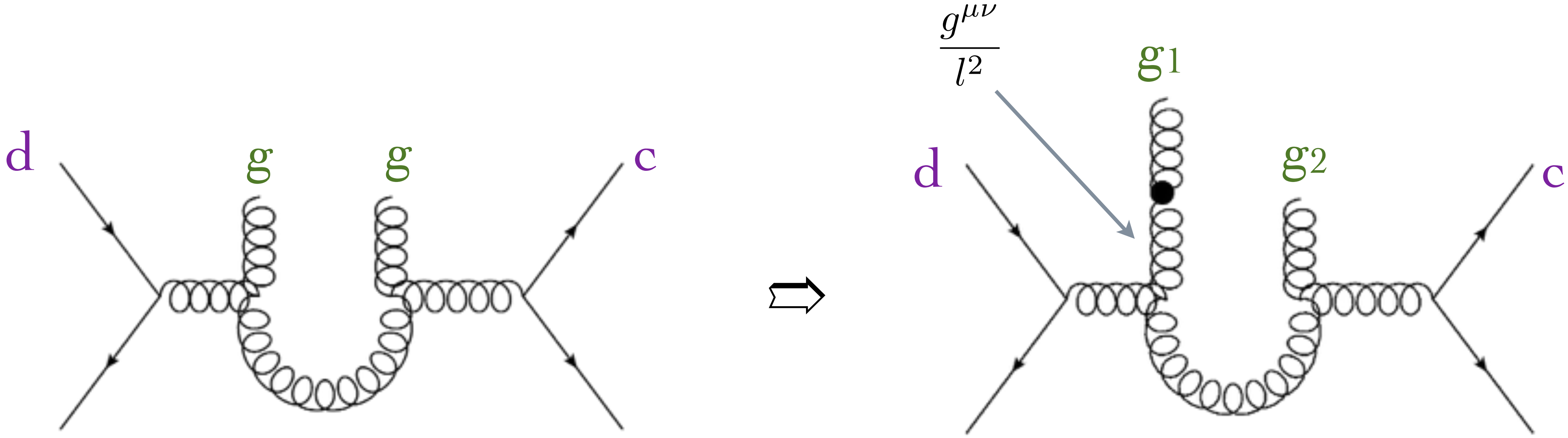
- Passarino-Veltman
- OPP

One Tool

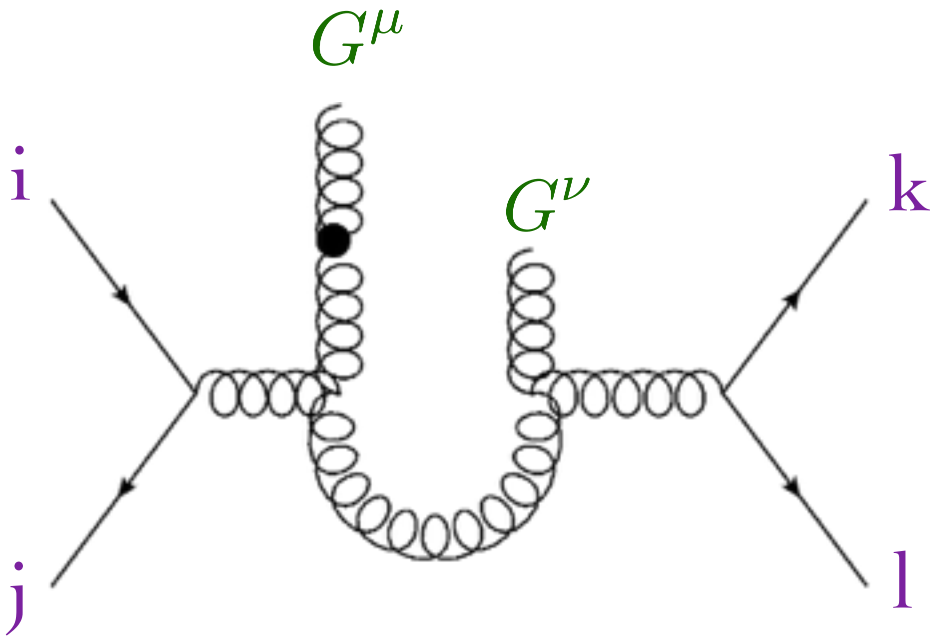
- MadLoop

Numerator

- We want to use (modified) HELAS method



- Closing the lorentz trace :



⇒

$$\delta^{\mu\nu} = \sum_{i=0}^{4} \underbrace{\delta^{\mu i}}_{G^\mu} \underbrace{\delta^{i\nu}}_{G^\nu}$$

External Wavefunction for HELAS

MadLoop

- Other modifications :
  - ➥ Allow for the loop momentum to be complex
  - ➥ Remove the denominator of the loop propagators
  - ➥ Close the color trace

A
L
O
H
A

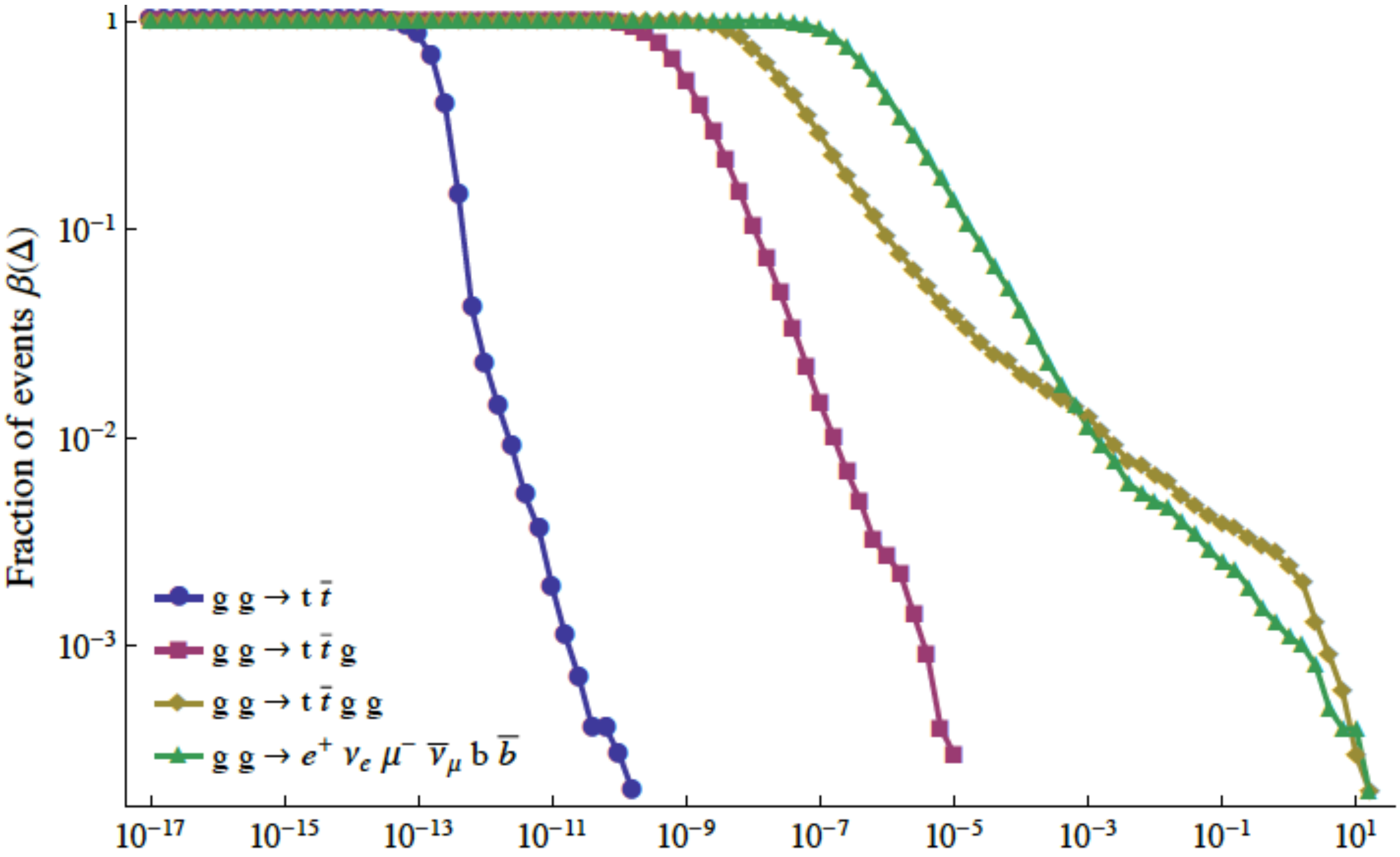
- Ok, now this gives you $\mathcal{N}(l^\mu)$, the integrand numerator to be fed to CT!

- But this is **SLOW**!!
- We have to compute this numerator ~ 50 times for each phase-space point!
- Idea instead of computing the numerator compute the polynomial form

$$\mathcal{N}(l^\mu) = \sum_{r=0}^{r_{max}} C^{(r)}_{\mu_0\mu_1\cdots\mu_r} l^{\mu_0} l^{\mu_1} \ldots l^{\mu_r}$$

[S. Pozzorini & al. hep-ph/1111.5206]

Numerical Stability

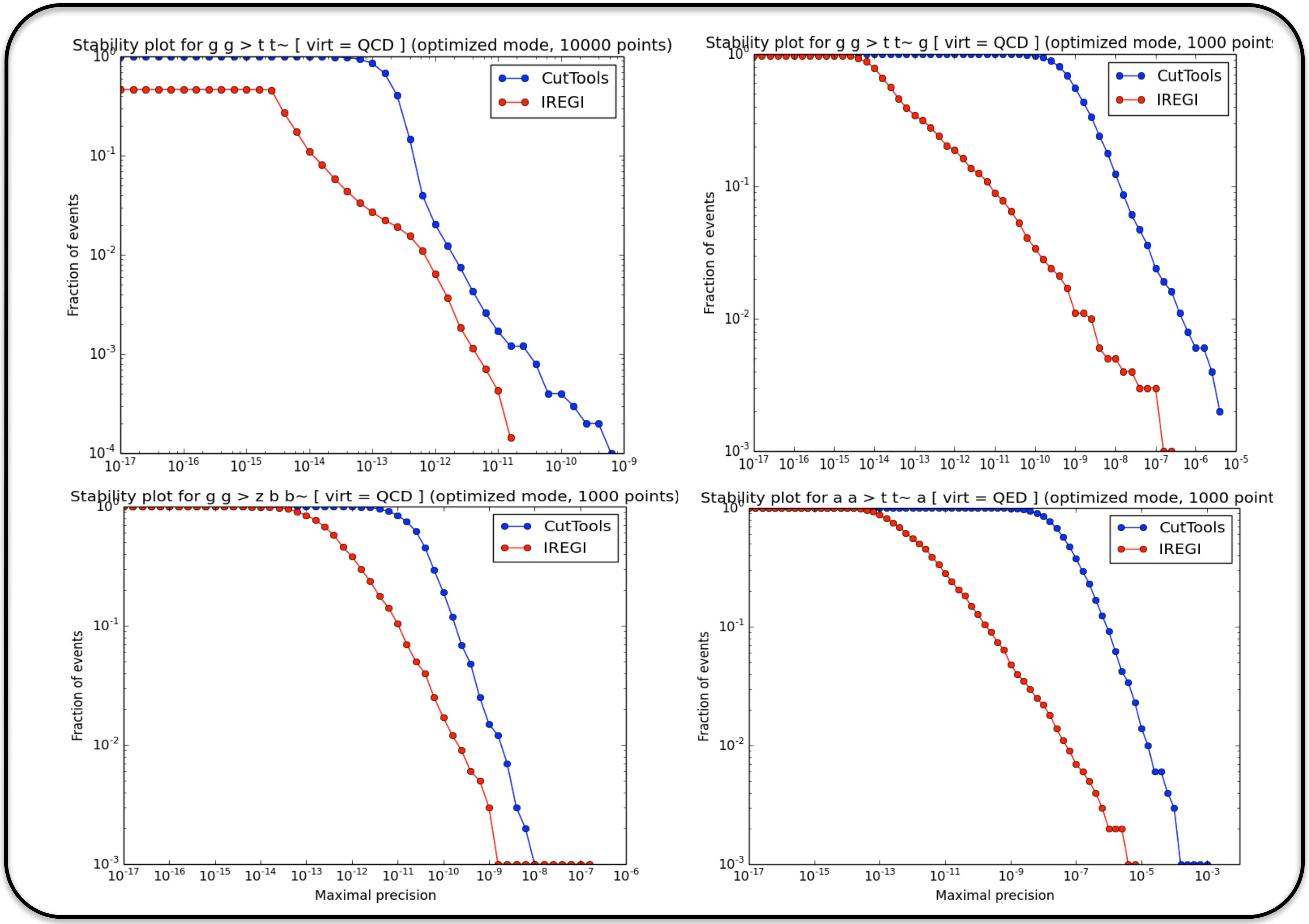


- For 2 to 4 processes, ~7% of the Phase-space point have a precision worse than 1e-3

➡ Previous solution pass to quadruple precision (extremelly slow)

IREGI

- New Solution use IREGI: a TIR program
  - ➡ Slower than previous method but faster than quadruple precision
  - ➡Usually less uncertainty (and not for the same PS point)



[H.-shao]

To Remember

- The main trick is to decompose in scalar integral
- OPP: works at the integrand level
- TIR: works at the integral level
- Loop evaluation is very slow
- Loop evaluation can be “unstable”

MadGraph5 : The platform

Various package in MG5_aMC@NLO

Plan

exemple: HEFT

- Model Description
- Width Computation
- Decay Chain
- Interference

Will not be cover

- Re-Weighting method
- Scale Variation
- TauDecay
- MadDM
- MadWeight
- Standalone
- external matrix element provider (Pythia8 and Matchbox)

MadGraph

1991 HELAS

199

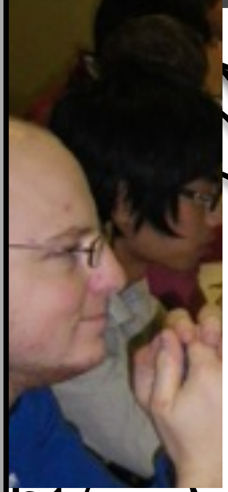
200

200

2011 MadGraph5

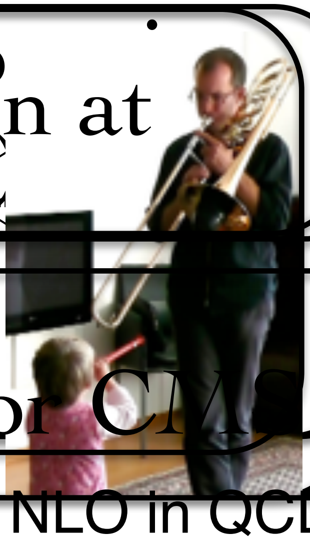
2014 MadGraph5_aMC@NLO

MAD stands for Madison



ht/...)

- Link to Pythia/PGS
- Suite of Routine which allow to
- Huge improvement
- Fully Automatic computation at
- Matching/Merging
- the matrix element for any (SM) process (SMADGRAPH)
- Support for the MSSM.
- NLO* (cross-section)
- NLO* matched to PS
- Official/Main SM generator for CMS



*NLO= NLO in QCD

Effective Field Theory

$$\mathcal{L} = \mathcal{L}_{SM} + \sum \frac{c_i}{\Lambda^2} \mathcal{O}_i$$

- Only few Operators for one process and different effects

Weak Boson production

Conserving CP

$$\mathcal{O}_{WWW} = \mathrm{Tr}[W_{\mu\nu} W^{\nu\rho} W_{\rho}^{\mu}]$$
$$\mathcal{O}_W = (D_\mu \Phi)^\dagger W^{\mu\nu} (D_\nu \Phi)$$
$$\mathcal{O}_B = (D_\mu \Phi)^\dagger B^{\mu\nu} (D_\nu \Phi)$$

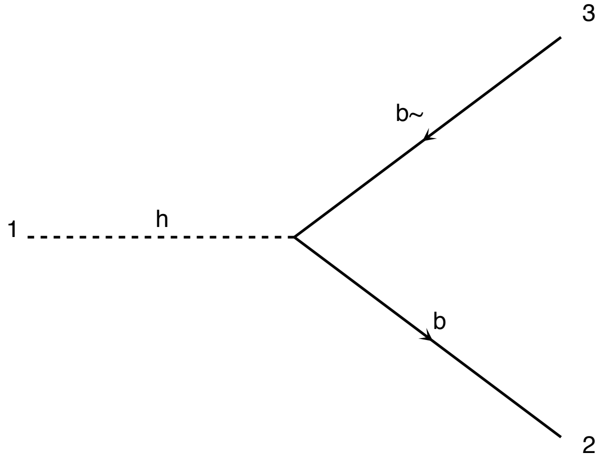
Not Conserving CP

$$\mathcal{O}_{\tilde{W}WW} = \mathrm{Tr}[\tilde{W}_{\mu\nu} W^{\nu\rho} W_{\rho}^{\mu}]$$
$$\mathcal{O}_{\tilde{W}} = (D_\mu \Phi)^\dagger \tilde{W}^{\mu\nu} (D_\nu \Phi)$$

2-body decay

2 body decay

$$\Gamma = \frac{1}{2MS}\int d\Phi_2 |\mathcal{M}|^2$$



- By Lorentz Invariance the matrix element is constant over the phase-space.

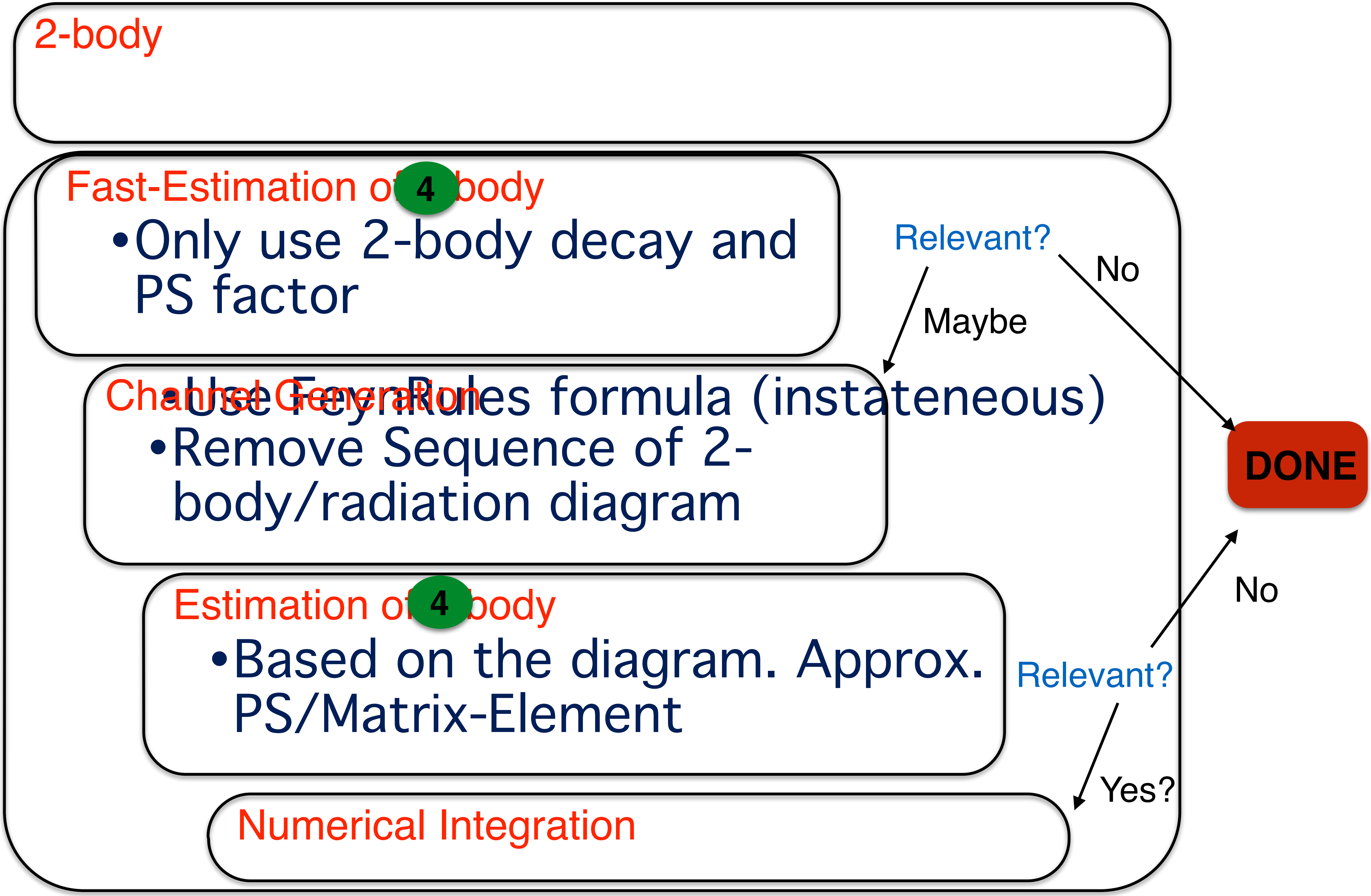
$$\Gamma = \frac{\sqrt{\lambda(M^2, m_1^2, m_2^2)}|\mathcal{M}|^2}{16\pi S M^3}$$

$$\lambda(M^2, m_1^2, m_2^2) = \left(M^2 - m_1^2 - m_2^2\right)^2 - 4m_1^2 m_2^2$$

- Calculable analytically by FeynRules

MadWidth

hep-ph/1402.1178



2-body

Fast-Estimation of 4 body

- Only use 2-body decay and PS factor

Channel Generation

- Use FeynRules formula (instateneous)

- Remove Sequence of 2-body/radiation diagram

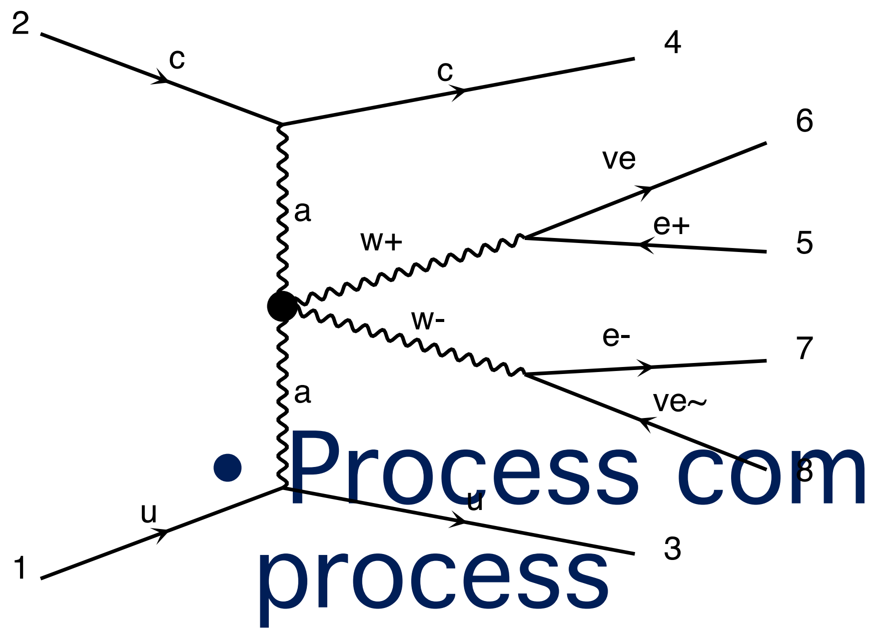
Estimation of 4 body

- Based on the diagram. Approx. PS/Matrix-Element

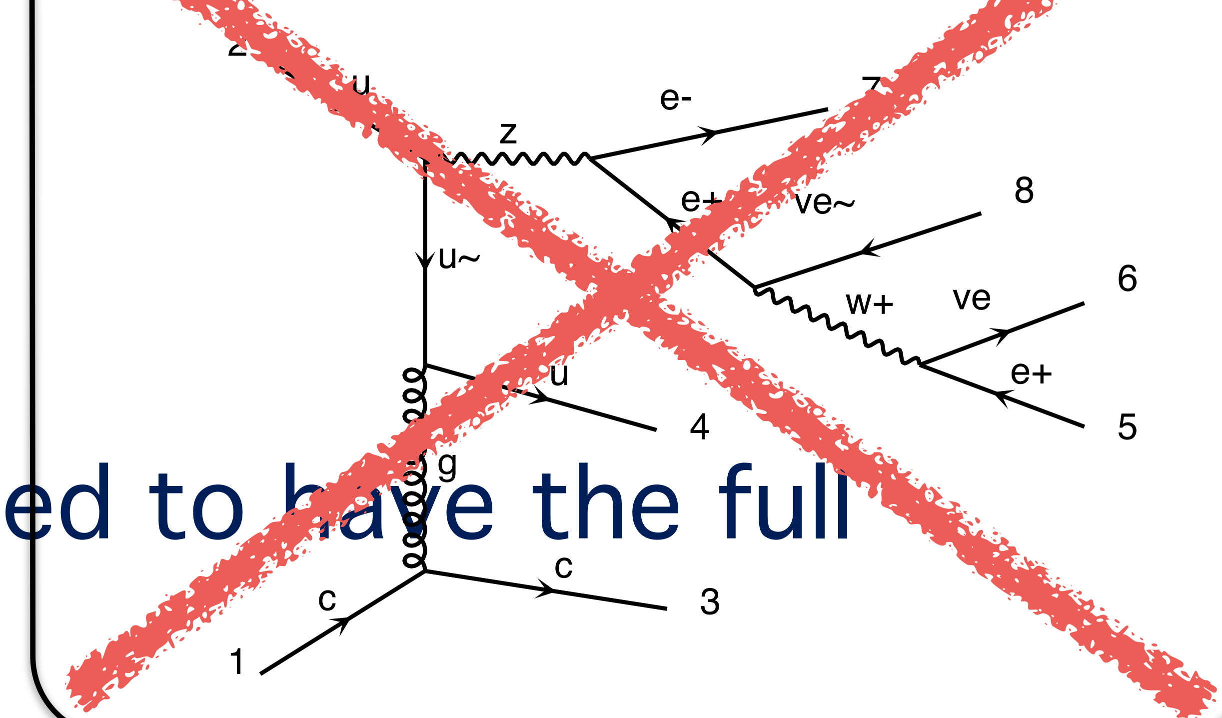
Numerical Integration

Decay

Resonant Diagram



Non Resonant Diagram



Problem

- Process complicated to have the full process

➡Including off-shell contribution

Solution

- Only keep on-shell contribution

Narrow-Width Approx.

Theory

$$\int dq^2 \left| \frac{1}{q^2 - M^2 - iM\Gamma} \right|^2 \approx \frac{\pi}{M\Gamma}\ \delta(q^2 - M^2)$$

- This is an Approximation!

$$\sigma_{full} = \sigma_{prod} * (BR + \mathcal{O}(\frac{\Gamma}{M}))$$

- This force the particle to be on-shell!

Comment

- Recover by re-introducing the Breit-wigner up-to a cut-off

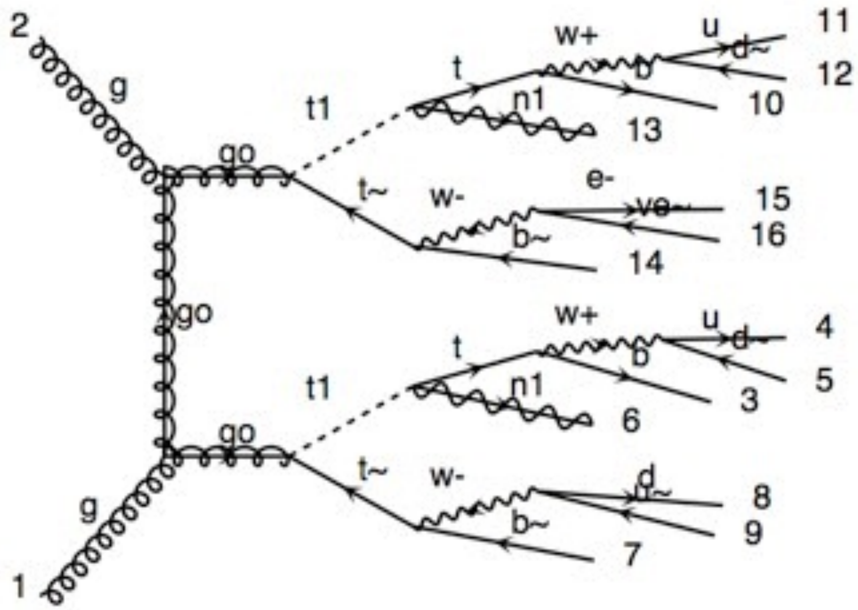
Decay chains

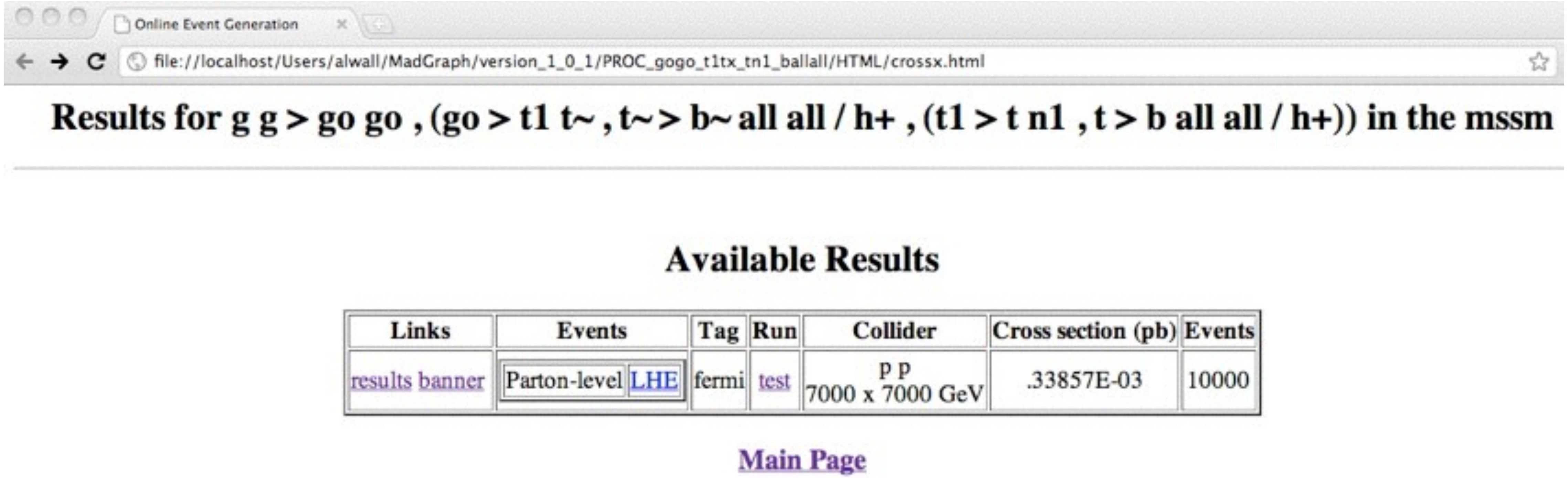
- p p > t t~ w+, (t > w+ b, w+ > l+ vl), \
  (t~ > w- b~, w- > j j), \
  w+ > l+ vl
- Separately generate core process and each decay
  - Decays generated with the decaying particle as resulting wavefunction
- Iteratively combine decays and core processes
- Difficulty: Multiple diagrams in decays

Decay chains

- Decay chains retain full matrix element for the diagrams compatible with the decay
- Full spin correlations (within and between decays)
- Full width effects
- However, no interference with non-resonant diagrams
  - ➡ Description only valid close to pole mass
  - ➡ Cutoff at |m ± nΓ| where n is set in run_card.

Decay chains





file://localhost/Users/alwall/MadGraph/version_1_0_1/PROC_gogo_t1tx_tn1_ballall/HTML/crossx.html

Results for g g > go go , (go > t1 t~ , t~ > b~ all all / h+ , (t1 > t n1 , t > b all all / h+)) in the mssm

Available Results

Links	Events	Tag	Run	Collider	Cross section (pb)	Events
results banner	Parton-level LHE	fermi	test	p p 7000 x 7000 GeV	.33857E-03	10000

Main Page

Thanks to developments in MadEvent, also (very) long decay chains possible to simulate directly in MadGraph!

MadSpin

[Artoisenet, OM et al. 1212.3460]

[Frixione, Leanen, Motylinski, Webber (2007)]

offshell	spin	unweighted
No	No	YES

One Event

- smear the mass
- flat decay

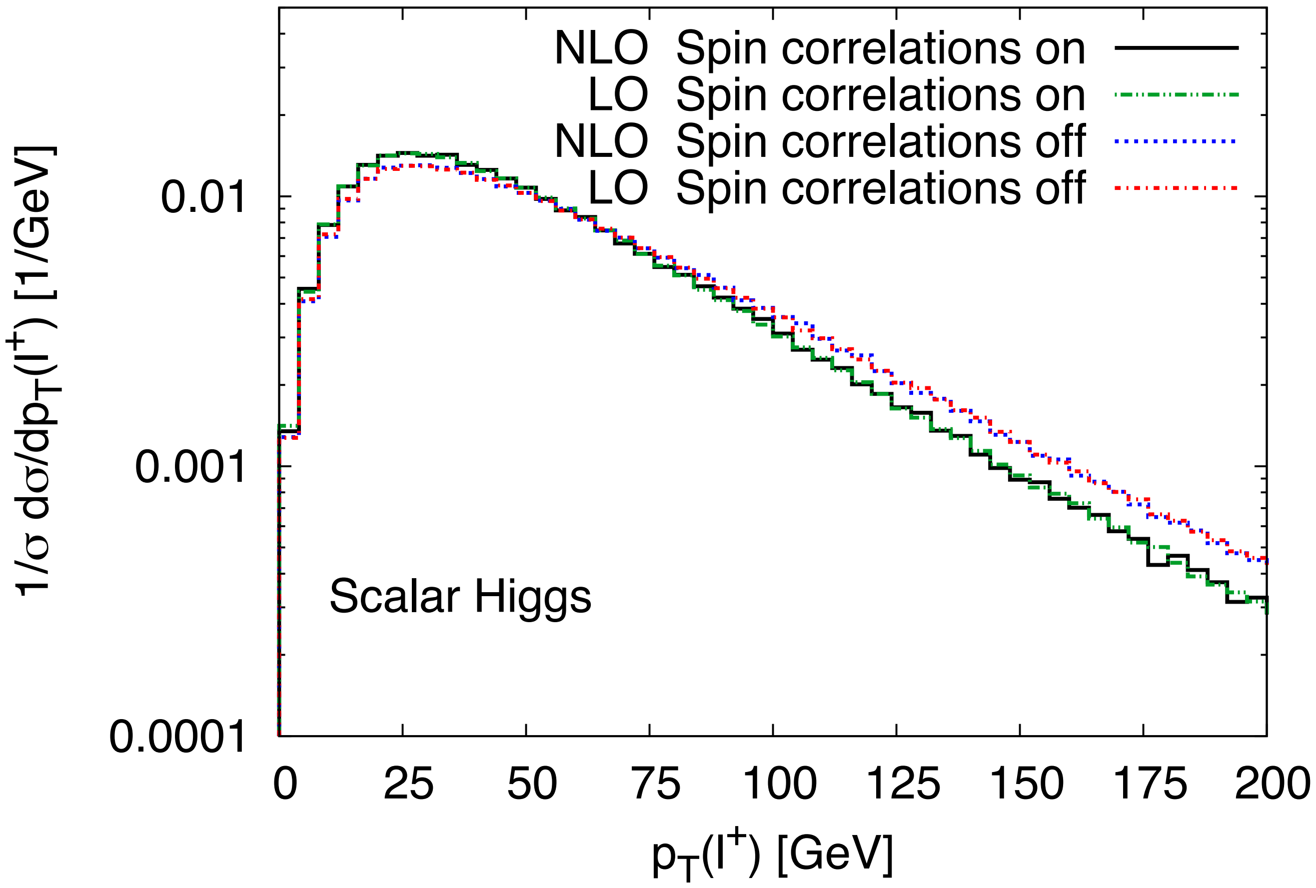
Decay Events

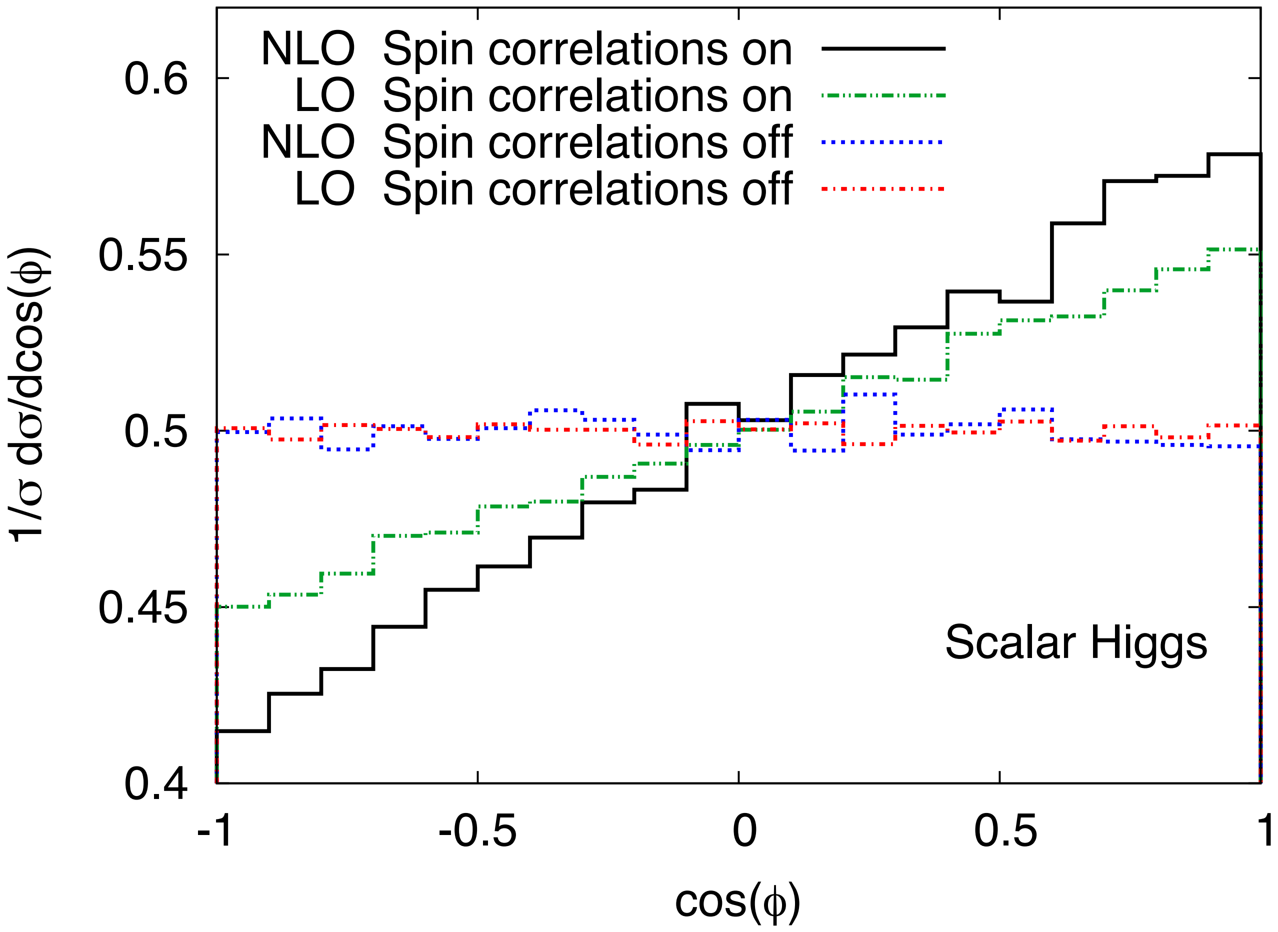
- re-weight by $|M_{LO}^{P+D}|^2/|M^{P_{LO}}|^2$

Decay Events II

- accept/reject method
- reject the decay not the event

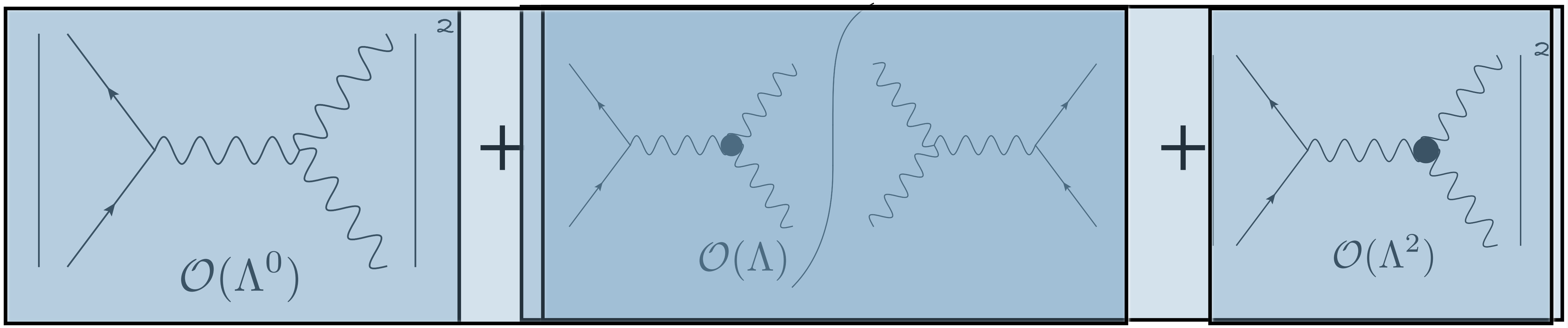
Final Sample


NLO Spin correlations on
LO Spin correlations on
NLO Spin correlations off
LO Spin correlations off
Scalar Higgs
$1/\sigma\ d\sigma/dp_T(l^+)$ [1/GeV]
0.01
0.001
0.0001
0
25
50
75
100
125
150
175
200
$p_T(l^+)$ [GeV]


NLO Spin correlations on
LO Spin correlations on
NLO Spin correlations off
LO Spin correlations off
Scalar Higgs
$1/\sigma\ d\sigma/d\cos(\phi)$
0.6
0.55
0.5
0.45
0.4
-1
-0.5
0
0.5
1
$\cos(\phi)$

Non Definite positive

Motivation:



SM
Model independent
Dominant

BSM
Model dependent
Sub-Dominant

Idea:

- Compute them separately
- Have a new syntax for such selection (NP^2=)

Status:

- Not compatible with decay chains

Type of generation

	Tree (SM)	Tree (BSM)	NLO (QCD) (SM)	NLO (QCD) (BSM)	NLO (EW) (SM)	NLO (EW) (BSM)	Loop Induced (SM)	Loop Induced (BSM)
Fix Order	✓	✓	✓	✓	✓	✗	✓	✓
+Parton Shower	✓	✓	✓	✓	✗	✗	✓	✓
Merged Sample	✓	✓	✓	?	✗	✗	✓	✓

Loop Induced:

- 2 to 2 processes: OK on a laptop
- 2 to 3 processes: OK on a small size cluster
- 2 to 4 processes: Specific case

MadGraph Functionality

- Leading Order Option
  - Support of BSM
  - Computation of the Width
  - Narrow width Approximation
    - Decay Chain
    - MadSpin
  - Systematics
- NLO
  - SM with merging