MARTY

UNIVERSITÉ DE GENÈVE

MARTY Status

An independent C++ program for loop-level Feynman diagram calculations in BSM scenarios

Website: https://marty.in2p3.fr/
Main publication: Comput. Phys. Commun. 264 (2021) 107928, arXiv: 2011.02478 [hep-ph]
Latest release: `MARTY-1.5` (30th May 2022)
On github:

- *Latest stable version:* https://github.com/docbrown1955/marty-public/
- *Test suite:* https://github.com/docbrown1955/test-suite/

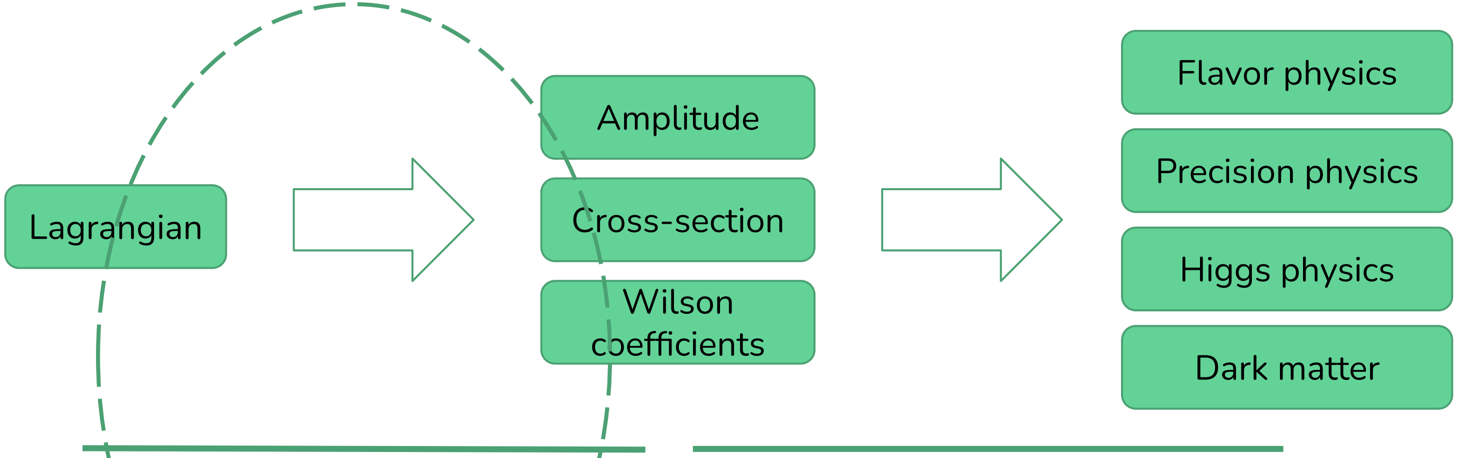
Workshop on Automatic Phenomenology
Institut Henri Poincaré
8 June 2022

Grégoire Uhlrich

Plan

Part I. Context and principles

The challenge of BSM phenomenology



- Model dependent
- Analytical calculation
- Long and error prone

- Mostly model independent
- Numerical calculation
- Well-known codes

Guiding principles

1. **Generality** (as much as possible) i.e. not specialized for any of the following:
- BSM scenarios
- Particle types (e.g. spin)
- Process types (decay, 2 ➔ 2, tree-level, one-loop)

2. **Independence**
- Fully free and open-source code
- Get rid of Mathematica
- Implement a built-in symbolic computation library

3. **Software development standards**

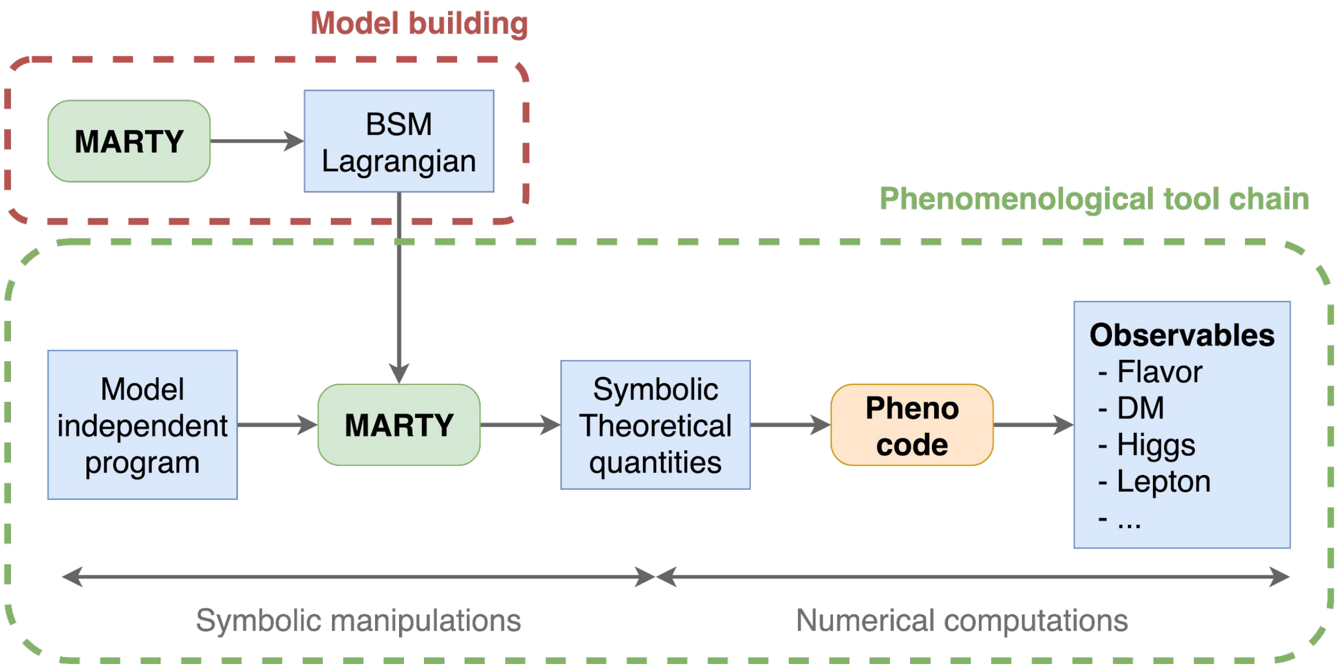
Software ecosystem around MARTY

MARTY Feature	Similar code
Symbolic computations	Mathematica
Representation theory (groups, algebras)	LieART
Feynman rules calculations	FeynRules, LanHEP
Diagram finding, diagram rendering	FeynArts, CalcHEP/CompHEP, MadGraph5_aMC@NLO
(Squared) amplitude calculation	FORM + FormCalc, CalcHEP/CompHEP, MadGraph5_aMC@NLO
Wilson coefficient calculation	FormFlavor
Code generation	FormCalc
Spectrum generator generator	SARAH/FlexibleSUSY

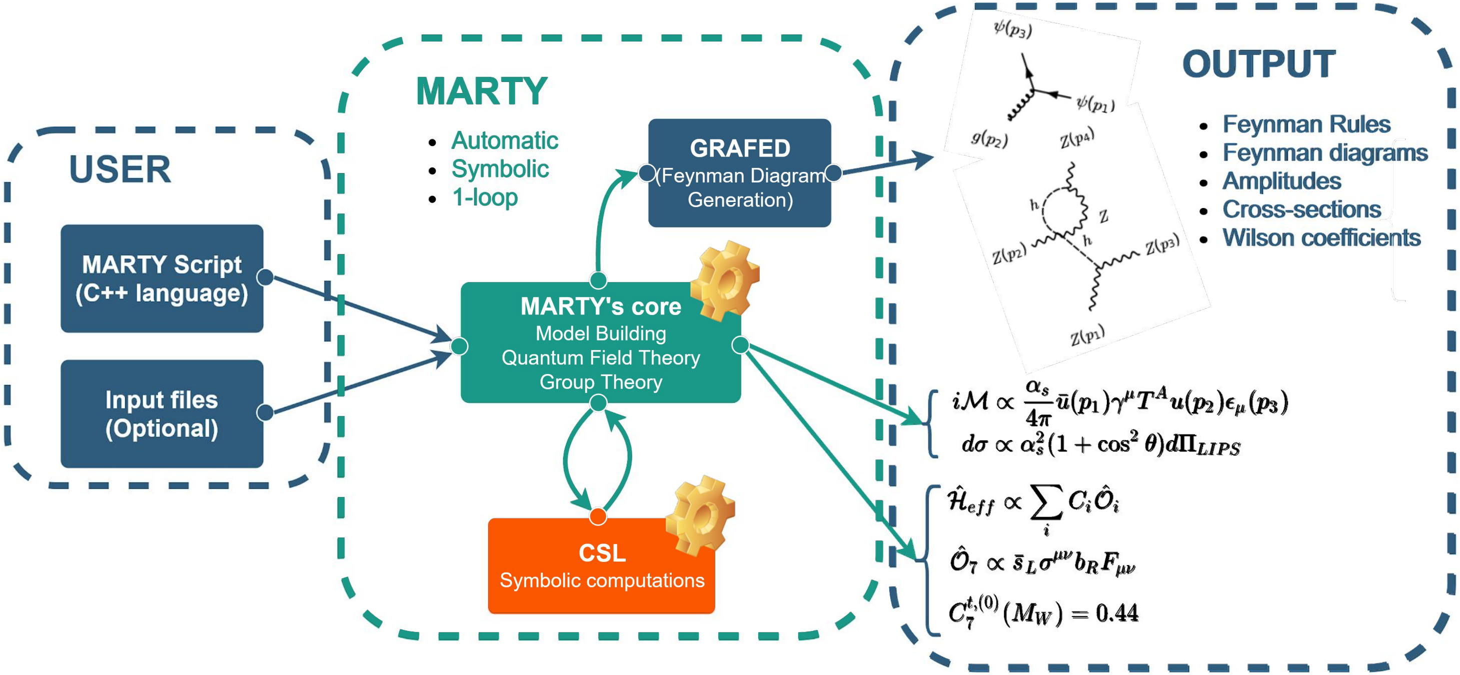
MARTY

Part II. Overview

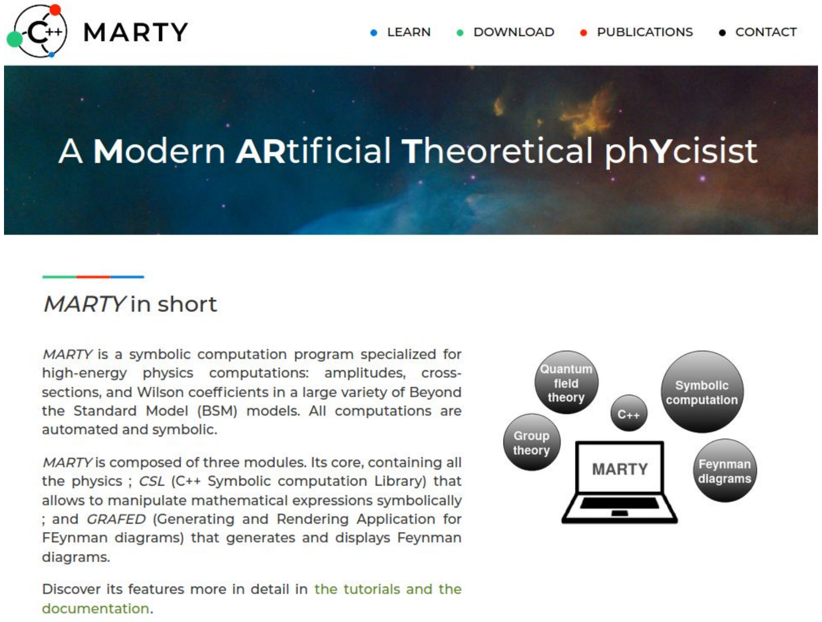
Purpose


Model building
MARTY
BSM Lagrangian
Phenomenological tool chain
Model independent program
MARTY
Symbolic Theoretical quantities
Pheno code
Observables
- Flavor
- DM
- Higgs
- Lepton
- ...
Symbolic manipulations
Numerical computations

Design


USER
MARTY Script (C++ language)
Input files (Optional)
MARTY
Automatic
Symbolic
1-loop
GRAFED (Feynman Diagram Generation)
MARTY's core
Model Building
Quantum Field Theory
Group Theory
CSL
Symbolic computations
OUTPUT
Feynman Rules
Feynman diagrams
Amplitudes
Cross-sections
Wilson coefficients
$i\mathcal{M} \propto \frac{\alpha_s}{4\pi}\bar{u}(p_1)\gamma^\mu T^A u(p_2)\epsilon_\mu(p_3)$
$d\sigma \propto \alpha_s^2(1+\cos^2\theta)d\Pi_{LIPS}$
$\hat{\mathcal{H}}_{eff} \propto \sum_i C_i\hat{O}_i$
$\hat{O}_7 \propto \bar{s}_L\sigma^{\mu\nu}b_R F_{\mu\nu}$
$C_7^{t,(0)}(M_W) = 0.44$

Documentation → https://marty.in2p3.fr



MARTY

• LEARN • DOWNLOAD • PUBLICATIONS • CONTACT

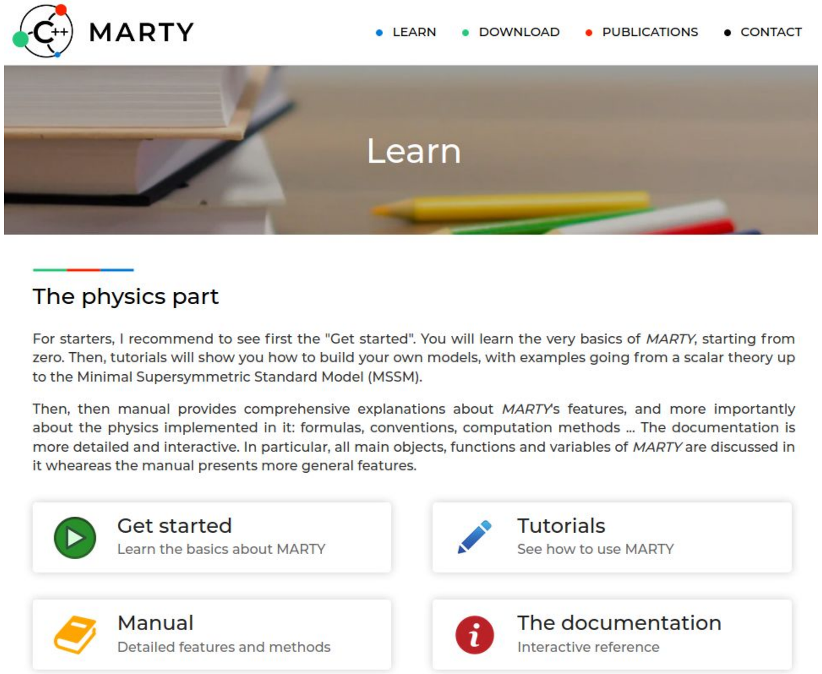
A Modern ARtificial Theoretical phYcisist

MARTY in short

MARTY is a symbolic computation program specialized for high-energy physics computations: amplitudes, cross-sections, and Wilson coefficients in a large variety of Beyond the Standard Model (BSM) models. All computations are automated and symbolic.

MARTY is composed of three modules. Its core, containing all the physics ; *CSL* (C++ Symbolic computation Library) that allows to manipulate mathematical expressions symbolically ; and *GRAFED* (Generating and Rendering Application for FEynman diagrams) that generates and displays Feynman diagrams.

Discover its features more in detail in the tutorials and the documentation.



MARTY

• LEARN • DOWNLOAD • PUBLICATIONS • CONTACT

Learn

The physics part

For starters, I recommend to see first the "Get started". You will learn the very basics of *MARTY*, starting from zero. Then, tutorials will show you how to build your own models, with examples going from a scalar theory up to the Minimal Supersymmetric Standard Model (MSSM).

Then, then manual provides comprehensive explanations about *MARTY*'s features, and more importantly about the physics implemented in it: formulas, conventions, computation methods ... The documentation is more detailed and interactive. In particular, all main objects, functions and variables of *MARTY* are discussed in it wheareas the manual presents more general features.

Get started — Learn the basics about MARTY

Tutorials — See how to use MARTY

Manual — Detailed features and methods

The documentation — Interactive reference

Part III. Main Features

UNIVERSITÉ DE GENÈVE

MARTY Models

- **Quantum Field Theory**
  - 4-dimensional Minkowski space-time
  - Spin 0, ½, 1
  - Model building utilities
- **Group theory**
  - Semi-simple Lie groups (SU(N), SO(N), Sp(N), E_6, E_7, E_8, F_4, G_2)
  - Representation theory
  - Algebra generators
  - Simplifications, traces

$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} + \left(D^\mu H\right)^\dagger D_\mu H + \mu H^\dagger H - \lambda\left(H^\dagger H\right)^2,$$

Model building in MARTY

- **High energy Lagrangian**
  - Less fields, less interactions
  - Depends less on conventions
  - **Much less error prone**
- **Automated top-down generic procedures**
  - Field redefinition, rotation
  - Expression replacement
  - Gauge and flavor symmetry breaking
  - Field rotation
  - Symbolic (bi-)diagonalization
- … Can also give explicitly the full Lagrangian

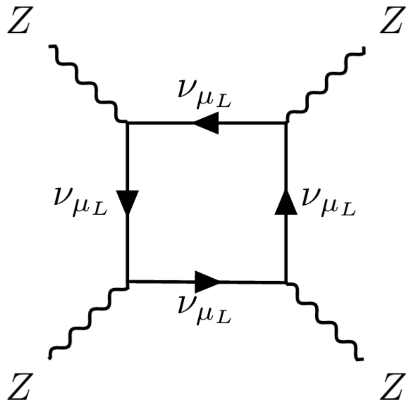
$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} + (D^{\mu}H)^{\dagger}D_{\mu}H + \mu H^{\dagger}H - \lambda\left(H^{\dagger}H\right)^{2},$$

$$H \rightarrow \begin{pmatrix} 0 \\ \frac{h+v}{\sqrt{2}} \end{pmatrix}$$

Gauge symmetry breaking +
Field redefinition

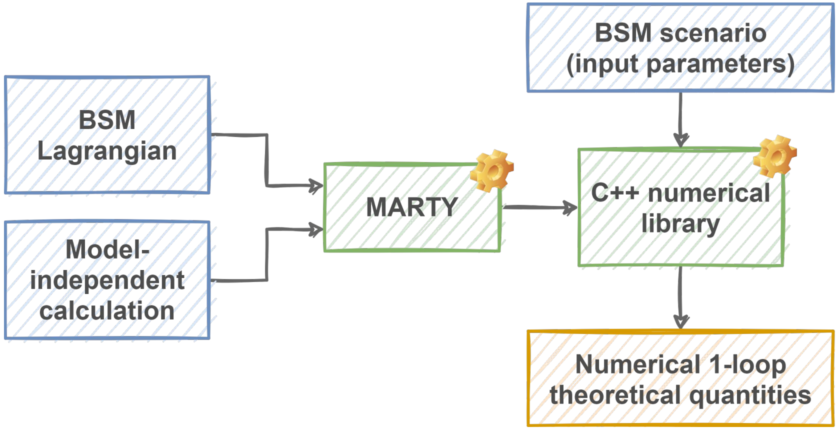
Calculations with MARTY

- Fully **symbolic and automated**
- Up to 5 external particles, at **1-loop**
- **Amplitudes, squared amplitudes, Wilson coefficients**
- **Simplifications**
  - Dirac algebra
  - Group algebra
  - Tensor reduction for momentum integrals
  - Dimensional regularization
  - Equations of motion (Dirac equation)
  - Definition of abbreviations
  - ...



Code generation

- Main output of MARTY
- Translate maths into C++ functions
- Scalar expressions (no free index)
  - Squared amplitudes
  - Wilson coefficients
- Self-contained library
  - Spectrum generator
  - Analytical results in numerical functions



Generation of spectrum generators

- Available for **all MARTY models**
- **Tree-level** spectrum generator
- Mass expressions e.g. $M_Z = \frac{1}{2\cos\theta_W} gv$
- Decay widths *(new in MARTY-1.5)*
  - All particles, on demand
  - Tree-level calculation at least
- Mass diagonalization
  - Masses
  - Mixings
  - Bi-diagonalization (two mixing matrices U_L, U_R)

General principle of MARTY's libraries

```
// Define input parameters
param_t params;
params.M_W = 80.379;
params.theta_W = 0.49;
// ...

// Calculate the spectrum (optional)
updateSpectrum(params);

// Evaluate numerically the theoretical
// quantities generated by MARTY
cout << f(params) << endl;
cout << g(params) << endl;
```

Part IV. Some physics implementations

Wick theorem - Finding Feynman diagrams

- **LSZ formula** for amplitude calculation
- Time-ordered product of quantum fields
- **Wick theorem:** contract fields two-by-two
- Example: A simple scalar theory

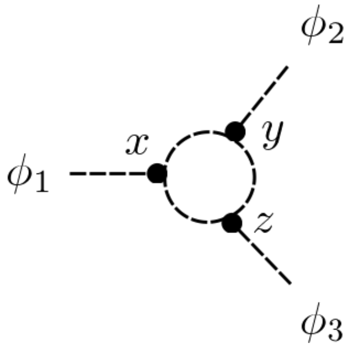
$$\mathcal{L}_{\text{int}} = \frac{\lambda}{3!}\phi^3$$

$$\Rightarrow i\mathcal{M}(\phi \to \phi\phi)_{1-\text{loop}}$$

$$\frac{1}{3!}\left(\frac{\lambda}{3!}\right)^3 \langle \phi_1\phi_2\phi_3 \cdot \phi_x\phi_x\phi_x \cdot \phi_y\phi_y\phi_y \cdot \phi_z\phi_z\phi_z \rangle$$

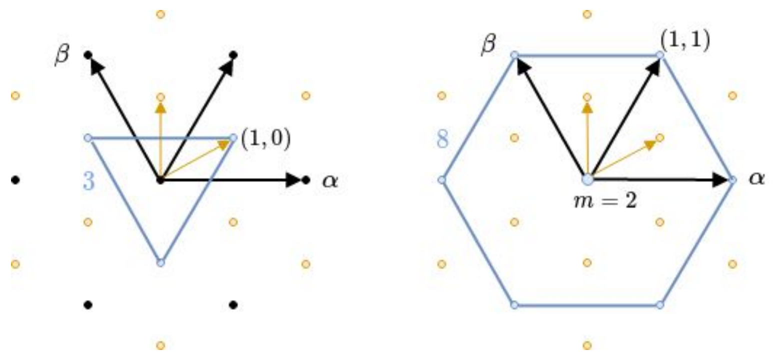
$$\ni \frac{\lambda^3}{1296} \langle \phi_1\phi_2\phi_3 \cdot \phi_x\phi_x\phi_x \cdot \phi_y\phi_y\phi_y \cdot \phi_z\phi_z\phi_z \rangle = 1\times$$

2×3, 2×2, 1×1, 3×3, 1×3, 1×2



Particles as irreducible representations

- **All semi-simple Lie algebras**
- SU(3) is $\mathbf{A_2}$**: 2 dinkin labels**
- Tensor product decomposition

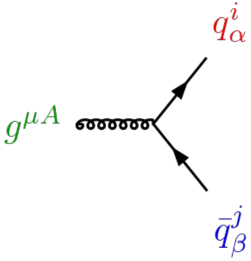



```
auto SU3 = CreateAlgebra(algebra::Type::A, 2);

auto q     = GetIrrep(SU3, {1, 0});
auto qBar  = GetIrrep(SU3, {0, 1});
auto gluon = GetIrrep(SU3, {1, 1});

cout << "3 x 3 x 3 = " << q * q * q << endl;
// >> 3 x 3 x 3 = 1 + 8 + 8 + 10 (Total dim = 27)

cout << "8 x 8     = " << gluon * gluon << endl;
// >> 8 x 8     = 1 + 8 + 8 + 10 + 10 + 27 (Total dim = 64)
```

Algebra generators

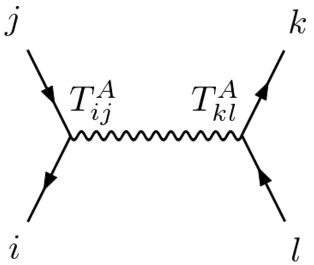
Simplify group generators

- Different for each group and representation
- Few open line simplifications
- Trace calculation in all representations



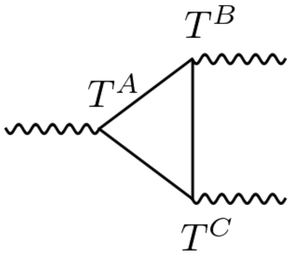
$$= i g_s \gamma^\mu_{\alpha\beta} T^A_{ij}$$

Open line simplifications



$$T^A_{ij} T^A_{kl} = \frac{1}{2}\left(\delta_{il}\delta_{kj} - \frac{1}{N}\delta_{ij}\delta_{kl}\right)$$

Traces in loops



$$\mathrm{Tr}(T^A T^B T^C) = I_1 d^{ABC} - i I_2 f^{ABC}$$

Diracology

Spin ½ particles

- Gamma matrices
- Simplify open fermion currents
- Evaluate traces of fermion loops

Useful definitions

$$\gamma^5 \equiv i\gamma^0\gamma^1\gamma^2\gamma^3$$

$$P_{L(R)} \equiv \frac{1 \mp \gamma^5}{2}$$

$$\sigma^{\mu\nu} \equiv \frac{i}{2}[\gamma^\mu, \gamma^\nu]$$

$$C \equiv -i\gamma^0\gamma^2$$

Multiple properties

$$\gamma^\mu\gamma_\mu = D$$

$$\{\gamma^\mu, \gamma^5\} = 0$$

$$\gamma^\mu P_L = P_R\gamma^\mu$$

$$C^2 = -1$$

$$C\gamma^\mu = -(\gamma^\mu)^T C$$

$$\dots$$

The Clifford algebra

$$\{\gamma^\mu, \gamma^\nu\} = 2g^{\mu\nu}$$

Trace identities

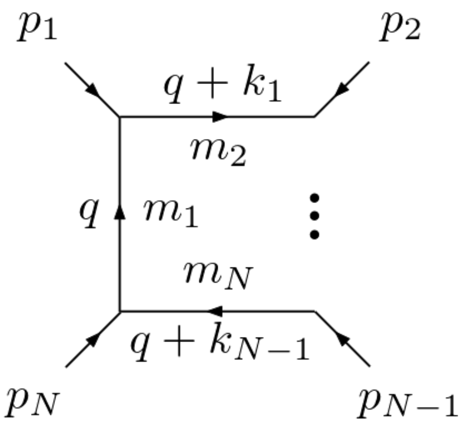
$$\mathrm{Tr}\left(\gamma^{\mu_1}\gamma^{\mu_2}\dots\gamma^{\mu_{2n}}\right) = \sum_{i=2}^{2n}(-1)^i g^{\mu_1\mu_i}\,\mathrm{Tr}\left(\gamma^{\mu_2}\dots\hat{\gamma}^{\mu_i}\dots\gamma^{\mu_{2n}}\right)$$

$$\mathrm{Tr}\left(\gamma^{\mu_1}\gamma^{\mu_2}\dots\gamma^{\mu_{2n}}\gamma^5\right) = \sum_{i=1}^{2n-1}(-1)^{\lfloor\frac{i-1}{2}\rfloor}\sum_{j=i+1}^{2n}(-1)^{i+j+1}g^{\mu_i\mu_j} \cdot \mathrm{Tr}\left(\gamma^{\mu_1}\dots\hat{\gamma}^{\mu_i}\dots\hat{\gamma}^{\mu_j}\dots\gamma^{\mu_{2n}}\gamma^5\right)$$

Quantum loop corrections

One-loop momentum integral:

$$I = \int \frac{d^4q}{i\pi^2} \frac{\prod_{i=1}^{n} q^{\mu_i}}{\prod_{j=0}^{m-1}((q-k_j)^2 - m_j^2)}$$



3-point function of rank 2:

$$C^{\mu\nu} \equiv \int \frac{d^4q}{i\pi^2} \frac{q^\mu q^\nu}{(q^2 - m_1^2)((q-k_1)^2 - m_2^2)((q-k_2)^2 - m_3^2)}$$
$$= C_{00} g^{\mu\nu} + C_{11} k_1^\mu k_1^\nu + C_{22} k_2^\mu k_2^\nu + C_{12}(k_1^\mu k_2^\nu + k_1^\nu k_2^\mu)$$

- `MARTY` applies the decomposition
- Adds also local terms from dim. reg.
- Numerical values of form factors come from `LoopTools`

General Fierz identities

UNIVERSITÉ DE GENÈVE

One-loop Wilson coefficients of 4-fermion processes

- Fermion ordering (simple Fierz)
- Simplification of fermion current products (double Fierz)

The fermion current

$$(\bar{\psi}_i \Gamma^A \psi_j) \equiv (\Gamma^A)_{ij}$$

The simple and double Fierz identities

$$(\Gamma^A)_{14} (\Gamma^B)_{32} = \frac{1}{\lambda^2} \sum_{C,D} \mathrm{Tr}\left(\Gamma^A \Gamma_C \Gamma^B \Gamma_D\right) (\Gamma^D)_{12} (\Gamma^C)_{34}$$

$$\left(\hat{\Gamma}^A\right)_{12} \left(\hat{\Gamma}^B\right)_{34} = \frac{1}{\lambda^4} \sum_{C,D,E,F} \mathrm{Tr}\left(\hat{\Gamma}^A \Gamma_C \hat{\Gamma}^B \Gamma_D\right) \mathrm{Tr}\left(\Gamma^D \Gamma_E \Gamma^C \Gamma_F\right) \cdot (\Gamma^F)_{12} (\Gamma^E)_{34}$$

The canonical basis

$$\Gamma^A \in \{P_L, P_R, \gamma^\mu P_L, \gamma^\mu P_R, \sigma^{\mu\nu}\}$$

$$\Gamma_A \in \left\{P_L, P_R, \gamma_\mu P_R, \gamma_\mu P_L, \frac{1}{2}\sigma_{\mu\nu}\right\}$$

$$\to \mathrm{Tr}\left(\Gamma^A \Gamma_B\right) = \lambda \delta^A_B$$

An example of double Fierz

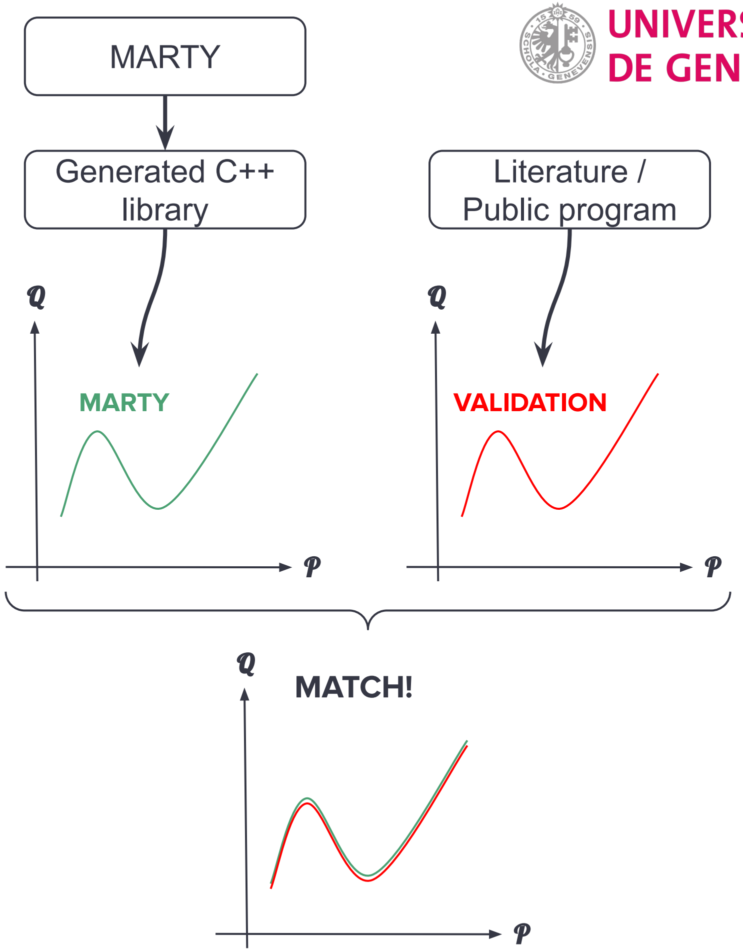
$$(\gamma^\mu \gamma^\nu \gamma^\rho P_L)_{12} (\gamma_\rho \gamma_\nu \gamma_\mu P_L)_{34} = 4 (\gamma^\mu P_L)_{12} (\gamma_\mu P_L)_{34}$$

Part V. Validations

Validation principle

Test a quantity $\mathcal{Q}$ w.r.t. to parameters $\mathcal{P}$

- **Known examples** (SM, pMSSM or THDM)
- **Different types of calculations:**
  - Squared amplitudes
  - Wilson coefficients
  - Some amplitude properties
- **Test behavior** w.r.t. one or two main parameters
- **Automated test suite:**
  - Integrate each validation example in the tests
  - Run the tests before each release



Squared amplitude validations

Process	Model	Order	Parameters	Specificities
h → XX // W → l nu // Z → ss	SM	Tree		Spin sums for spins ½, 1
ee → (Z,ϒ) → μμ	SM	Tree	√s, cos(θ)	Interference, A_{FB}
gg → tt	SM	Tree	√s	SU(3) weights, f_{ABC} coupling, ghosts
h → ϒϒ / h → gg	SM	1-loop		Integral, local terms, SU(3) traces
WW → WW	SM	Tree	√s	Large calc., interference + cancellation
SS → SS // F_1F_1 → F_1F_2V		1-loop		Simplification fermion-number violation
$SUSY^2$ → SM^2 (~3000 p.)	MSSM	Tree	√s, cos(θ)	MSSM model, large variety of diagrams

→ Currently integrating it in the automated test suite (WIP)

Wilson coefficient validations (1-loop)

Process	Model	Contribution	Parameters	Specificities
C_7, C_7' ($b \to s\Upsilon$)	SM, MSSM	top (+ stops)	m_t / (μ, M_2)	Magnetic operator, Goldstones, WWA vertex
C_8, C_8' ($b \to sg$)	SM, MSSM	top (+ stops)	m_t / (μ, M_2)	Chromo-magnetic operator, unitary gauge
$(g-2)_\mu$ ($\mu \to \mu\Upsilon$)	SM, MSSM	All	m_μ / (μ, M_2)	Same as C_7, identical fermions
C_9, C_{10} ($b \to s\mu\mu$)	SM	All	m_t	4-fermion operator, Fierz identities
C_9, C_{10} ($b \to s\mu\mu$)	MSSM	All	Random	Chargino interactions
C_7, C_8, C_9, C_{10}	GTHDM	top	m_{H+}	Reproduce 2111.10464
B_s mixing	Multiple VLQ models	Decoupling limit		Operators with identical external particles

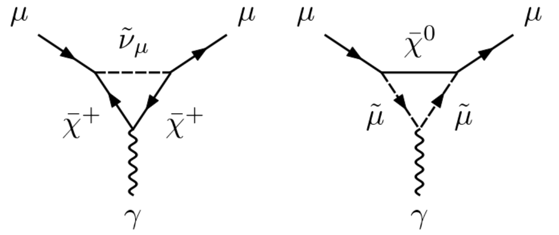
Example: $(g\text{-}2)_\mu$ (pMSSM)

Magnetic dipole moment

$$i\mathcal{M} \ni a_\mu \frac{ie}{4m_\mu} \left(\bar{\mu}\sigma^{\mu\nu}\mu\right) F_{\mu\nu}$$

Tension with experiments

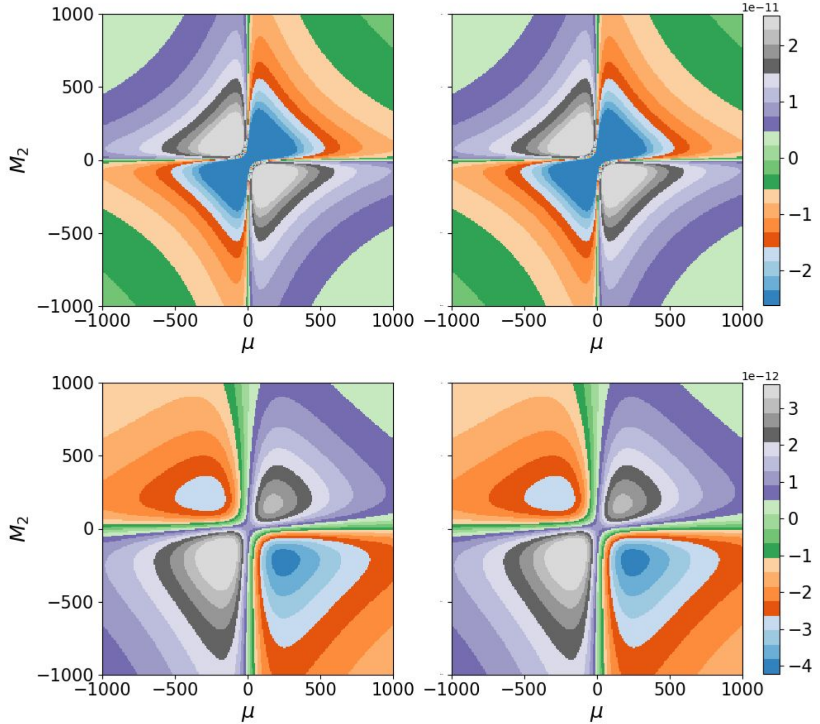
$$a_\mu^{\text{EXP}} - a_\mu^{\text{SM}} = (251 \pm 59) \times 10^{-11}$$



MARTY　Validation

χ^+

χ^0



Part VI. Conclusion

Ongoing projects

Leptogenesis / Dark Matter — Advanced

→ Tree-level squared amplitude
→ Boltzmann equation solver

Collaborator: A. Dasgupta

Dark Matter — Paper in preparation (MSSM)

→ Tree-level squared amplitude (a lot)
→ DM relic density, indirect detection

Collaborators: A. Arbey, F. Mahmoudi, M. Palmiotto

Flavor physics — Submitted [2201.04659]

→ One-loop Wilson coefficients
→ Rare B decays ($b \to s$)

Collaborators: M.A. Boussejra, F. Mahmoudi

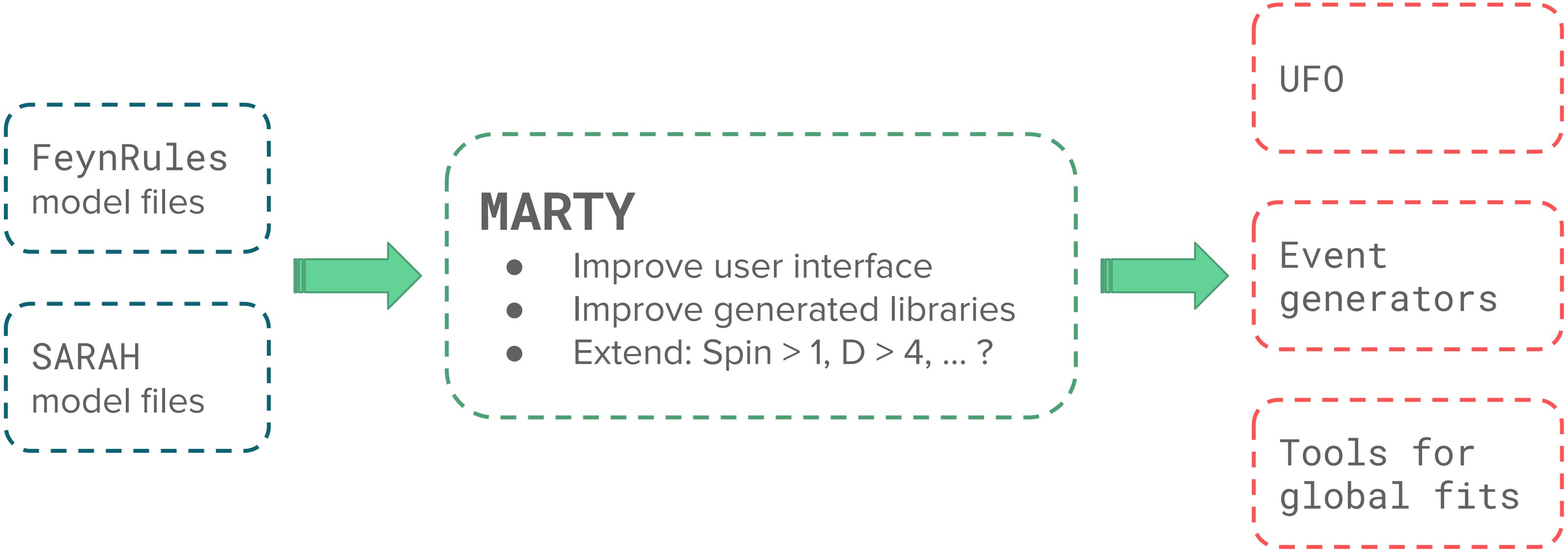
Flavor physics — Work in progress

→ One-loop Wilson coefficients
→ Constraints on Vector-Like Quarks

Collaborators: A. Deandrea, T. Hurth, F. Mahmoudi, S. Neshatpour

The interest in MARTY is growing, other projects are ramping up…

Future developments



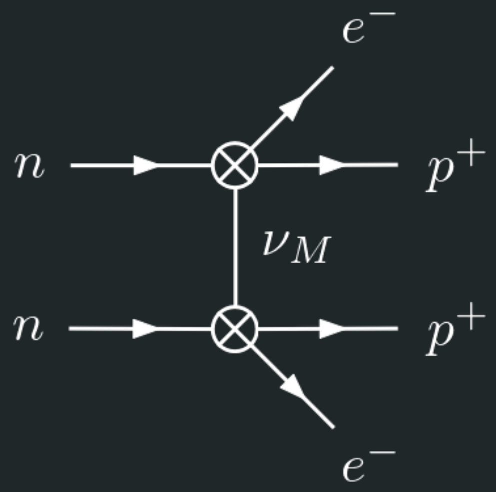
Conclusion

- Need for automation in **general Beyond the Standard Model** phenomenology
- This need will grow fast
  - Experimental deviations from the SM will pile up
  - New, larger / more complicated models
- What makes MARTY different
  - Unique C++ program, Mathematica-independent
  - Provides high-level Model Building utilities
  - Still calculates at one-loop for general BSM scenarios
  - One-loop Wilson coefficients (flavor anomalies)
- Already validated on very diverse examples
- Contribute on the public github repo !

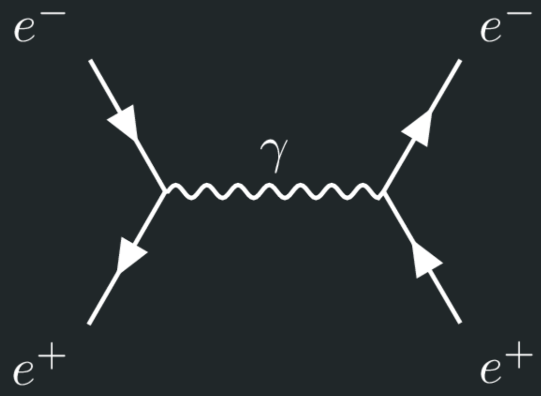
  https://github.com/docbrown1955/marty-public/

Thank you!

Backup

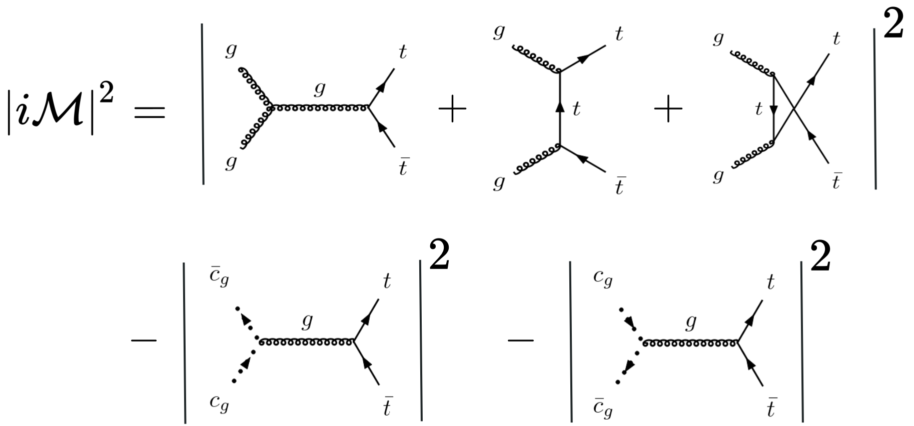


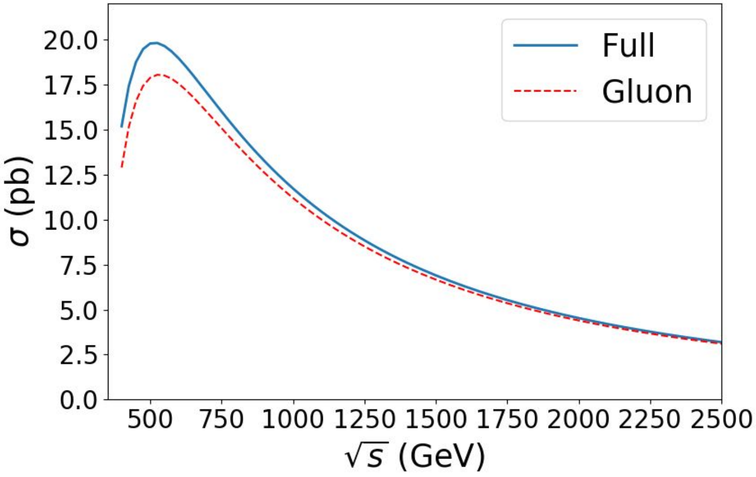
Squared amplitude validations



gg ➔ tt (SM)

Typical LHC process:

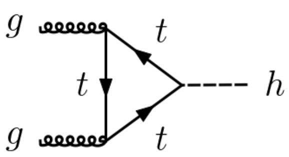




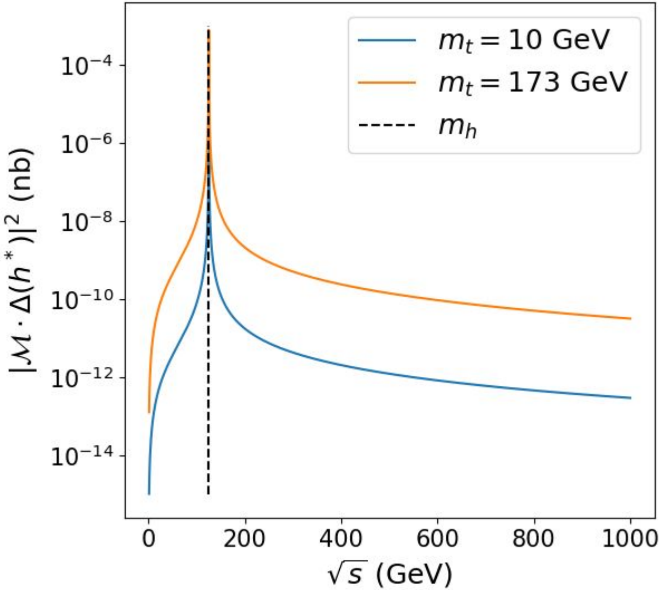
gg → h (SM)

Leading contribution at the LHC:

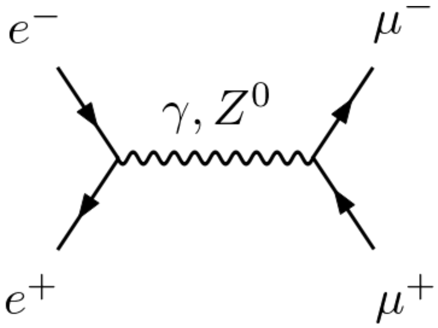
$$|i\mathcal{M}\cdot\Delta(h^*)|^2 \approx \left| \quad \right|^2$$



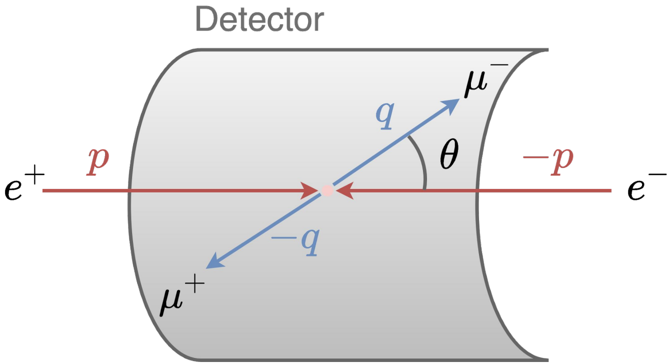
$$\times \left| \frac{1}{s-m_h^2+i\Gamma_h m_h} \right|^2$$

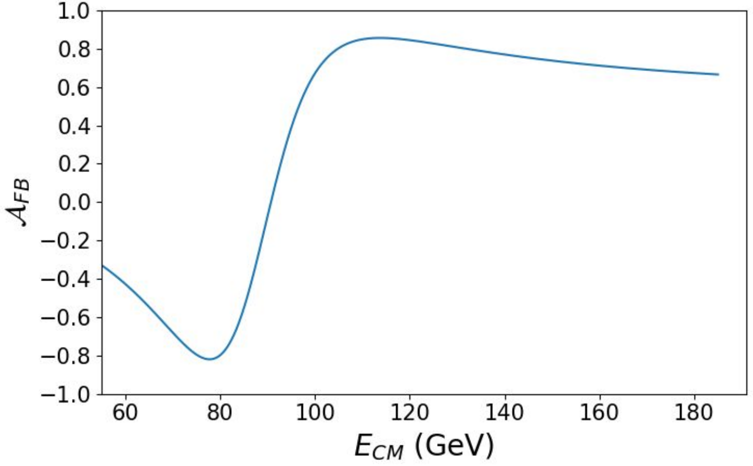


Forward-backward asymmetry (SM)



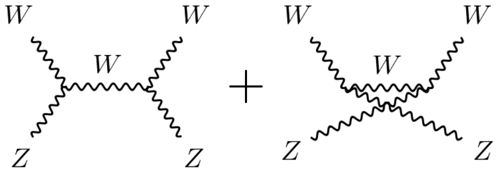
$$\mathcal{A}_{FB} \equiv 2\pi \frac{\int_0^{\pi/2} \frac{d\sigma}{d\theta} d\theta - \int_{\pi/2}^{\pi} \frac{d\sigma}{d\theta} d\theta}{\sigma}$$





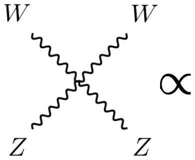
Vector boson scattering and Unitarity

$i\mathcal{M}_{3V} =$



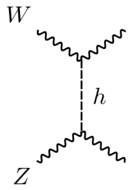
$\propto E^4$

$i\mathcal{M}_{4V} =$



$\propto E^4$

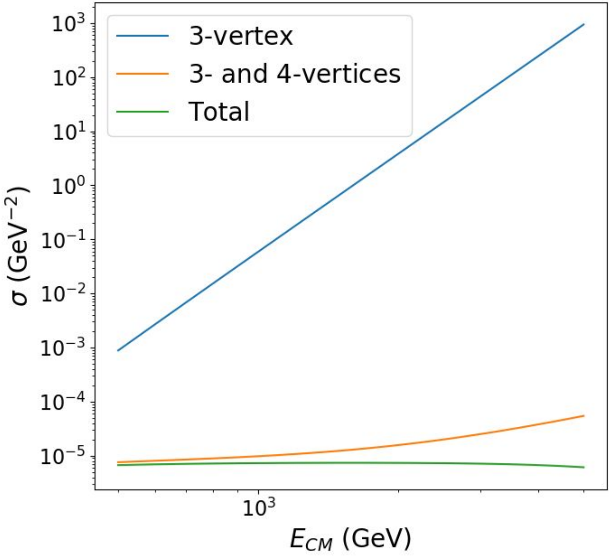
$i\mathcal{M}_h =$



$\propto E^2$

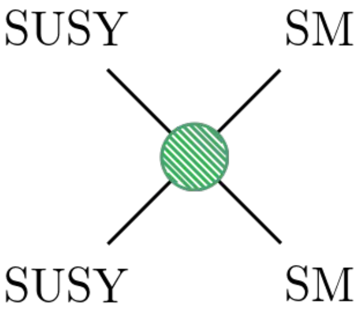
Cancellations

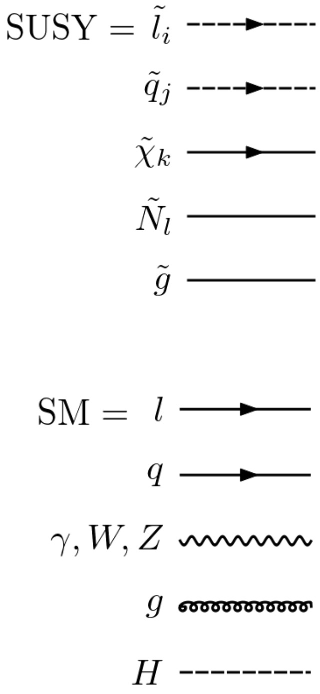
$$i\mathcal{M}_{3V+4V} \propto E^2$$
$$i\mathcal{M}_{3V+4V+h} \propto E^0$$



2 → 2 tree-level squared amplitudes (pMSSM)

With M. Palmiotto



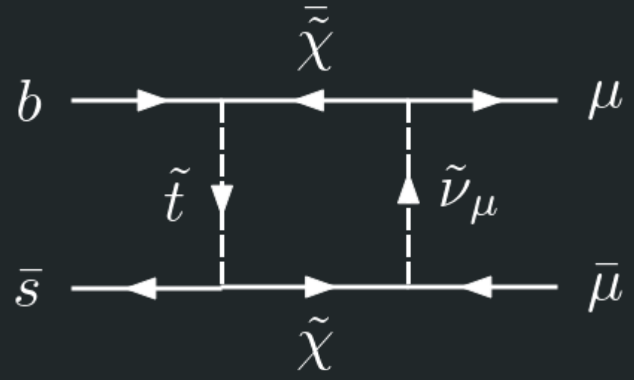


~3000 non trivial squared amplitudes

- High-energy MSSM
- Top-down procedures
- Squared amplitude
- Performance
- Parameter definitions

→ **Very good validation for MARTY**

Wilson coefficient validations

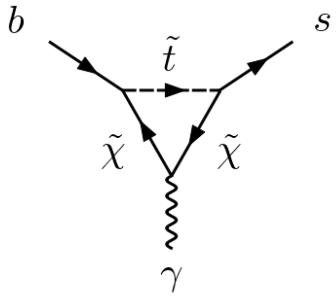


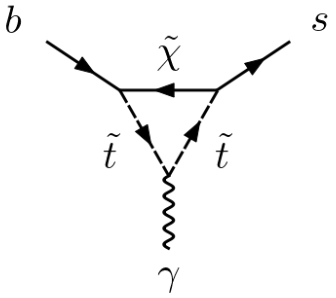
b ➜ sγ (pMSSM)

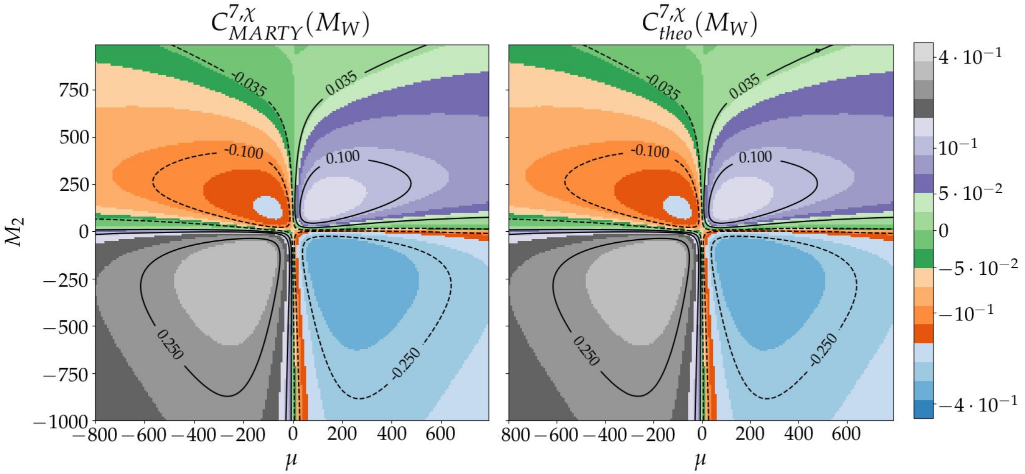
Well constrained experimentally

➜ Important to evaluate

$$i\mathcal{M} \propto C_7 \left(\bar{s}\,\sigma^{\mu\nu} P_R b\right) F_{\mu\nu}$$







(Spectrum generated by MARTY)

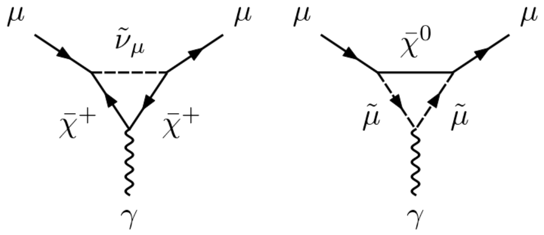
$(g\text{-}2)_\mu$ (pMSSM)

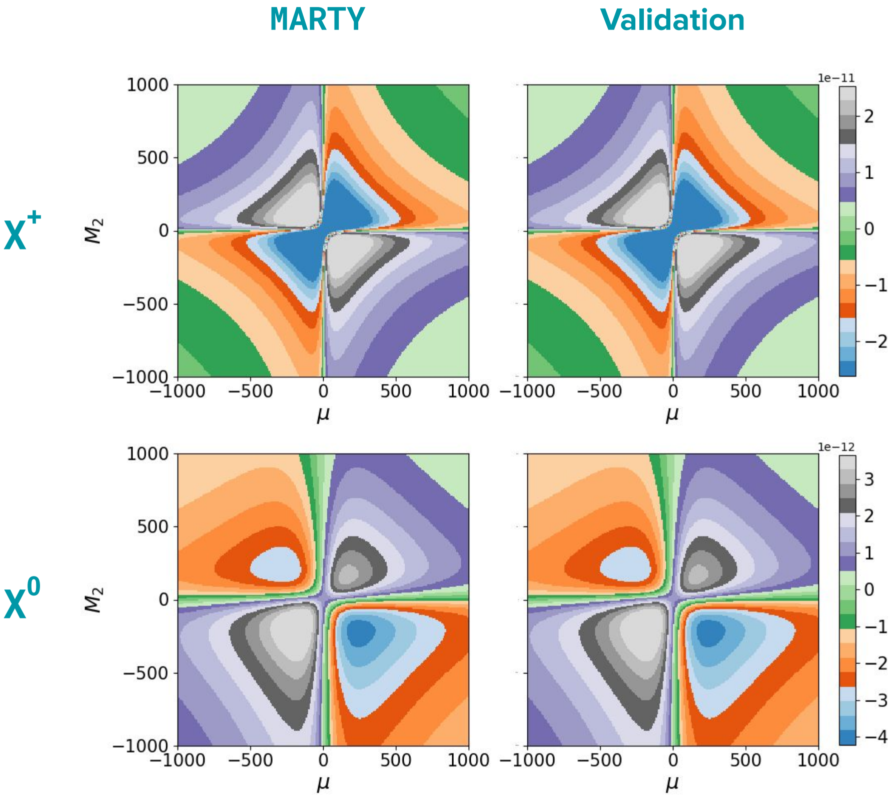
Magnetic dipole moment

$$i\mathcal{M} \ni a_\mu \frac{ie}{4m_\mu}\left(\bar{\mu}\sigma^{\mu\nu}\mu\right)F_{\mu\nu}$$

Tension with experiments

$$a_\mu^{\text{EXP}} - a_\mu^{\text{SM}} = (251 \pm 59) \times 10^{-11}$$



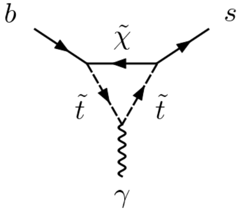


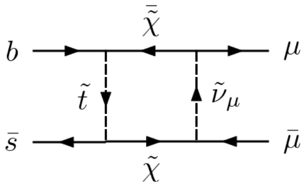
b ➔ sμμ (pMSSM)

- Main interest for flavor anomalies
- 4-fermion operators

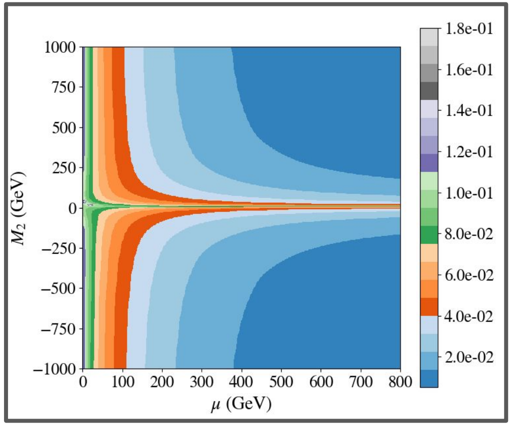
$$i\mathcal{M} \propto C_9\,(\bar{s}\gamma^{\mu}P_L b)\,(\bar{\mu}\gamma_{\mu}\mu)$$

Best fit: $\dfrac{\delta C_9^{\mu}}{C_9^{\mu,\mathrm{SM}}} \approx -0.25$

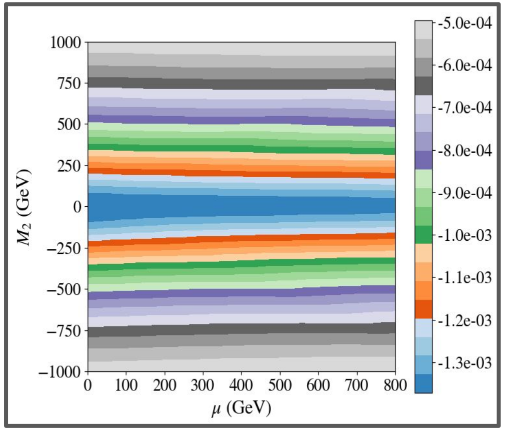




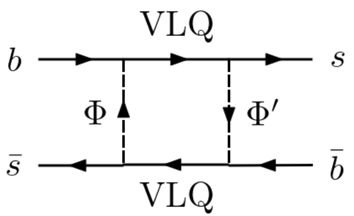
Photon penguins



Boxes



B_s- mixing with Vector-Like Quarks (VLQ)



$$\begin{aligned} D &= (3,1,-1/3) \\ Q_d &= (3,2,-5/6) \\ Q_v &= (3,2,+1/6) \\ T_d &= (3,3,-1/3) \\ T_u &= (3,3,+2/3) \end{aligned}$$

$$\mathcal{O}_{\text{LR1}} = (\bar{s}\gamma^\mu P_L b)\cdot(\bar{s}\gamma_\mu P_R b)$$

$$\mathcal{O}_{\text{VLL(VRR)}} = (\bar{s}\gamma^\mu P_{L(R)} b)\cdot(\bar{s}\gamma_\mu P_{L(R)} b)$$

$$i\mathcal{M} \propto C_{\text{LR1}}\mathcal{O}_{\text{LR1}} + C_{\text{VLL(VRR)}}\mathcal{O}_{\text{VLL(VRR)}}$$

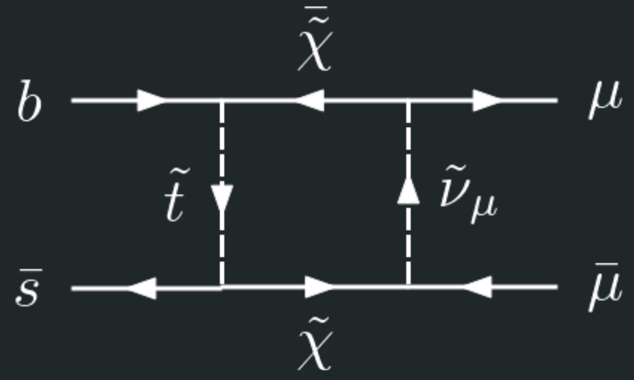
$$M_{\text{VLQ}} \gg m_h, M_Z$$

(F_m, F_n)	D	Q_d	Q_v	T_d	T_u
D	VLL, $+1/8$	LR1, $+1/4$	LR1, $-1/4$	VLL, $+1/8$	VLL, $-1/4$
Q_d	×	VRR, $+1/4$	VRR, $-1/2$	LR1, $+3/8$	LR1, $-3/8$
Q_v	×	×	VRR, $+1/4$	LR1, $-3/8$	LR1, $+3/8$
T_d	×	×	×	VLL, $+5/32$	VLL, $-1/4$
T_u	×	×	×	×	VLL, $+5/32$

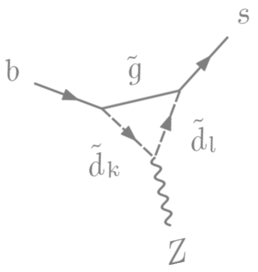
Results in NMFV-MSSM

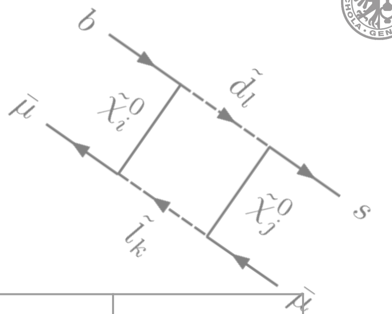
C_7, C_9 and $(g-2)_\mu$ at 1-loop

M.A. Boussejra, GU, F. Mahmoudi, to appear

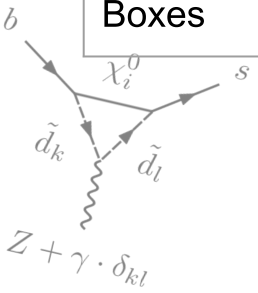


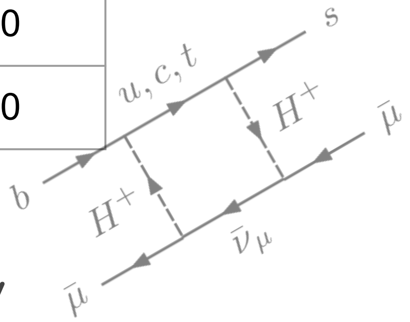
Diagrams for C_9





	Chargino	Neutralino	Gluino	Charged Higgs	Neutral Higgs
γ-penguins	240	96	24	24	0
Z-penguins	624	1344	240	78	0
Boxes	864	13824	0	12	0





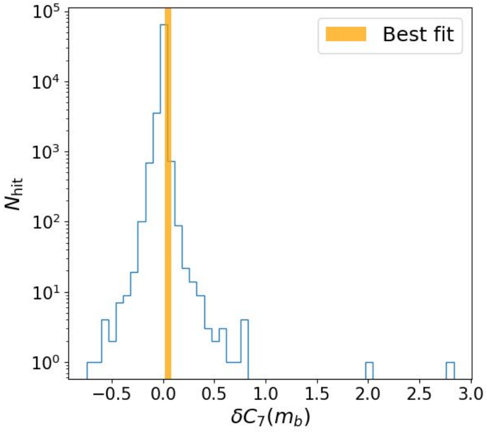
- **About 18 000 diagrams calculated by MARTY!**
- **Merges diagrams and extracts C_9 automatically**

→ **Similarly for C_7 and $(g-2)_\mu$**

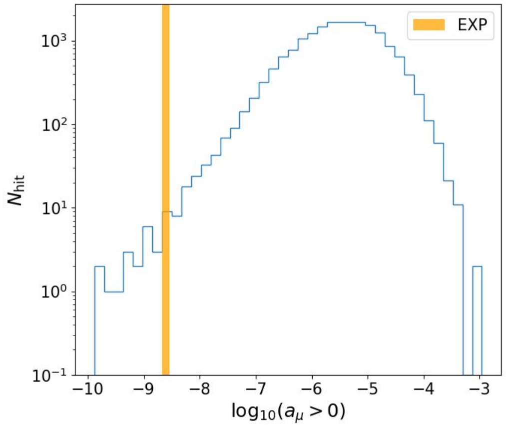
Individual distributions

- Random scan (70 000 valid scenarios)
- Check if predictions fit experimental measurements

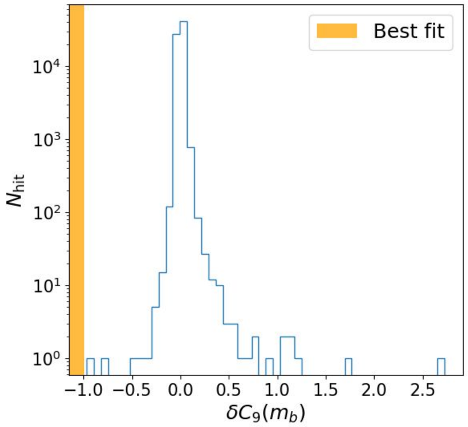
δC_7



$\delta(g-2)_\mu$



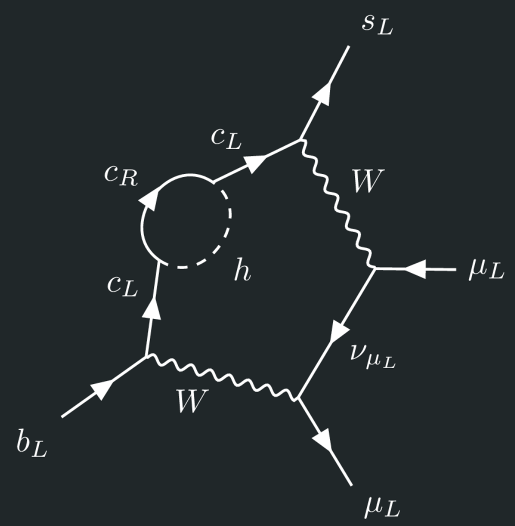
δC_9



➡ **Proof of concept, MARTY can do much more!**

Symbolic computations

What are the main challenges?



Principle

Computers are meant to manipulate numbers

- Symbols have not definite numerical values
- Requires specific implementations

Symbolic calculations

- Store symbolic expressions
- Apply simplification identities
- Manipulate large and abstract expressions

$$\Rightarrow \quad \sum_\lambda \epsilon(p)^A_{\lambda\mu}\left(\epsilon(p)^B_{\lambda\nu}\right)^\dagger = \left(-g_{\mu\nu} + \frac{p_\mu p_\nu}{M^2}\right)\cdot \delta^{AB}$$

Numerical computations

```
x = 88
y = 2.21
```

} Definite values

```
217./88*x + 800*y = 1985
```

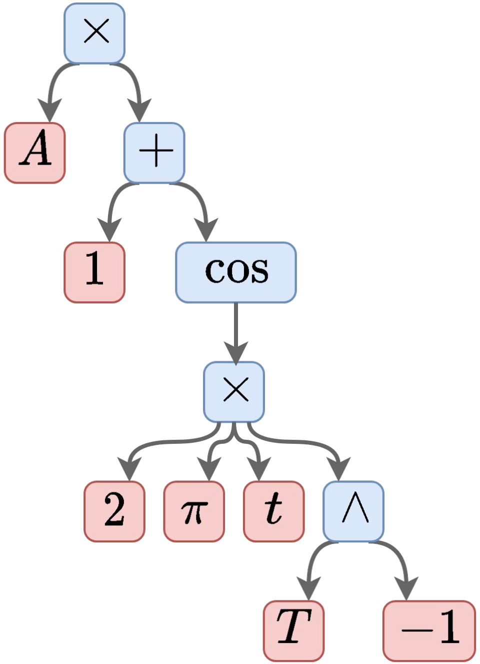
Analytical calculations

```
x undefined variable

x + x + x*2 = 4*x
```

Tree representation of expressions

- Most natural representation
- Different from a linear storage
  - Higher cost for construction / destruction
  - Lower cost for modification

- **Polymorphism**
- **Dynamic programming**



$$A\left(1+\cos\frac{2\pi t}{T}\right)$$

UNIVERSITÉ DE GENÈVE

Canonicalization

- Basic simplifications
- Keep expressions in a canonical form
- Must be quick and simple
- Absolutely necessary

$$\left[\frac{d}{dt}\cos(\pi/2+\omega t)\right]_{t=0} = \frac{d}{dt}(\pi/2+\omega t)\cdot\cos'(\pi/2+\omega t)$$
$$= \left[\left(\pi\cdot 0+\frac{0}{2}+\omega\cdot 1+0\cdot t\right)\cdot(-\sin(\pi/2+\omega t))\right]_{t=0}$$
$$= \left(\pi\cdot 0+\frac{0}{2}+\omega\cdot 1+0\cdot 0\right)\cdot(-\sin(\pi/2+\omega\cdot 0))$$

Simple derivative example

$$\left[\frac{d}{dt}\cos(\pi/2+\omega t)\right]_{t=0} = -\omega$$

Rules for canonicalization

$$\begin{aligned}
0\cdot x &\rightarrow 0\\
1\cdot x &\rightarrow x\\
(a+b)+(c+d) &\rightarrow a+b+c+d\\
(a\times b)\times(c\times d) &\rightarrow a\times b\times c\times d\\
\sin(\pi/2) &\rightarrow 1
\end{aligned}$$

Ordering

$$a + xb - 3 \stackrel{?}{=} -3 + bx + a$$

$$A_{ji} A_{jk} \stackrel{?}{=} A_{kj} A_{ij}$$

- Define a **total order** between expressions
- Key of computer algebra
- Human-readable expression
- Order makes everything simpler
  - Simpler algorithms
  - **Comparison O(N^2) → O(N)**

Apply the ordering rules

$$-3 < a < b < x$$

$$A_{ij} < A_{ji}$$

$$A_{jk} < A_{kj}$$

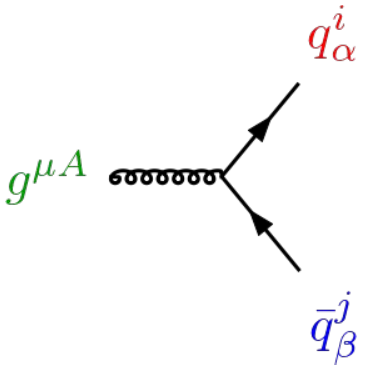
$$-3 + a + bx \stackrel{?}{=} -3 + a + bx$$

$$A_{ij} A_{jk} \stackrel{?}{=} A_{ij} A_{jk}$$

UNIVERSITÉ DE GENÈVE

Indexed tensors and theoretical physics

- Indices ubiquitous in particle physics
- High performance
- Light-weight objects
- Properties:
  - Complex conjugation
  - Transposition
  - Contractions



$$= i g_s \gamma^{\mu}_{\alpha\beta} T^A_{ij}$$

$$(\gamma^\mu)^\dagger = \gamma^0 \gamma^\mu \gamma^0$$

$$T^A_{ij} T^A_{kl} = \frac{1}{2}\left(\delta_{il}\delta_{kj} - \frac{1}{N}\delta_{ij}\delta_{kl}\right)$$

$$\gamma^\mu_{ab}\gamma_{\mu bc} = D\delta_{ac}$$