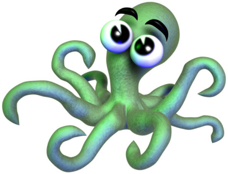
MadGolem - automating NLO calculations for New Physics

David López-Val
ITP, Ruprecht-Karls Universität Heidelberg

in collaboration with D. Gonçalves-Netto, T. Plehn (Heidelberg U.), I. Wigmore (Edinburgh U.), K. Mawatari (Vrije U.)

arXiv:1108.1250 arXiv:1203.6358

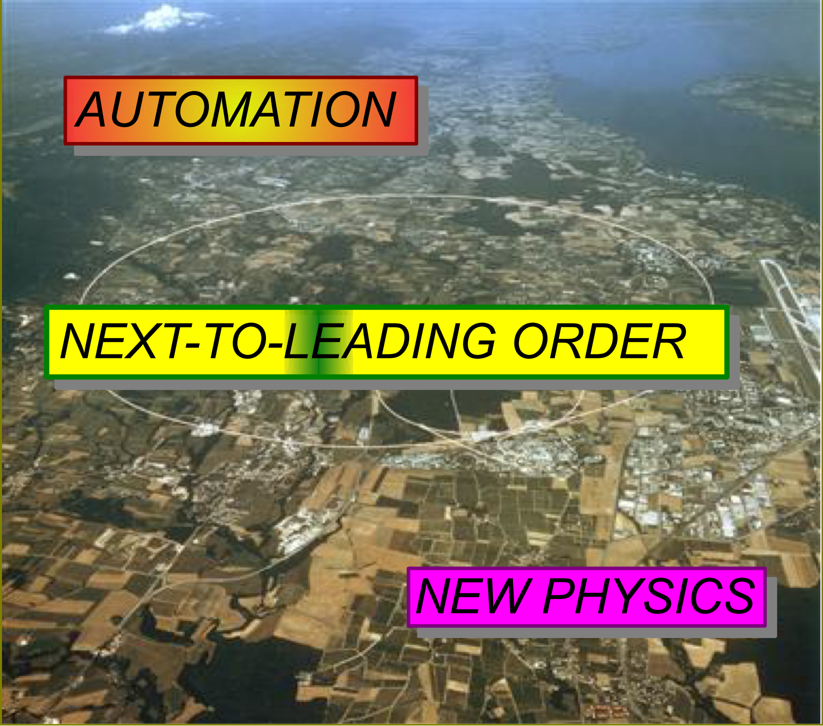


Loops & Legs

April 19th, Wernigerode, Germany

Outline

Outline


AUTOMATION
NEXT-TO-LEADING ORDER
NEW PHYSICS

Tools for New Physics Searches at Hadron Colliders

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

MadGolem

[Binoth, Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1108.1250 [hep-ph]]

- Fully automated calculation of NLO QCD corrections for arbitrary $2 \to 2$ processes in a generic BSM framework soon to be public !

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

MadGolem

[Binoth, Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1108.1250 [hep-ph]]

- Fully automated calculation of NLO QCD corrections for arbitrary $2 \to 2$ processes in a generic BSM framework soon to be public !

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

MadGolem: *raison-d'etre*

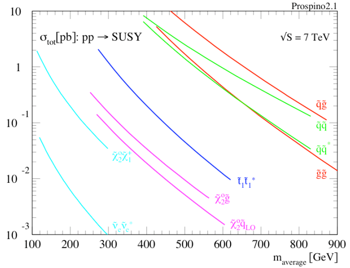
Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses ...

Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

PROSPINO



Benakker, Höper, Krämer, Plehn, Spira, Zerwas

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

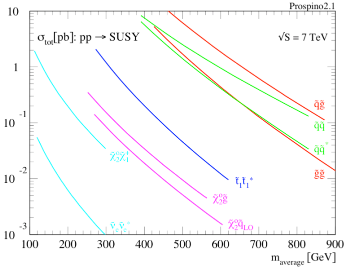
Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

$\Uparrow$

PROSPINO

$\Downarrow$



Benakker, Höper, Krämer, Plehn, Spira, Zerwas

MadGolem: *raison-d'etre*

Why NLO ?

- QCD corrections quantitatively relevant : $K \sim 1.5$
- QCD corrections qualitatively relevant: scale dependence, normalization & shape of distributions, gluon radiation, new partonic subprocesses . . .

Why Automation ?

- Many models & processes $\leftrightarrow$ analogue technical challenges
- Cost & time saving, robustness, accessibility
- Eases validation, engages Theory/Experiment interchange

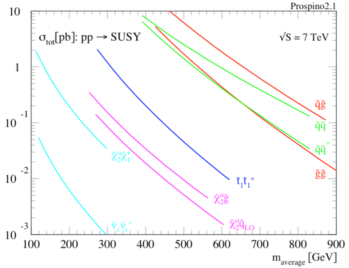
Automating

$\Uparrow$

PROSPINO

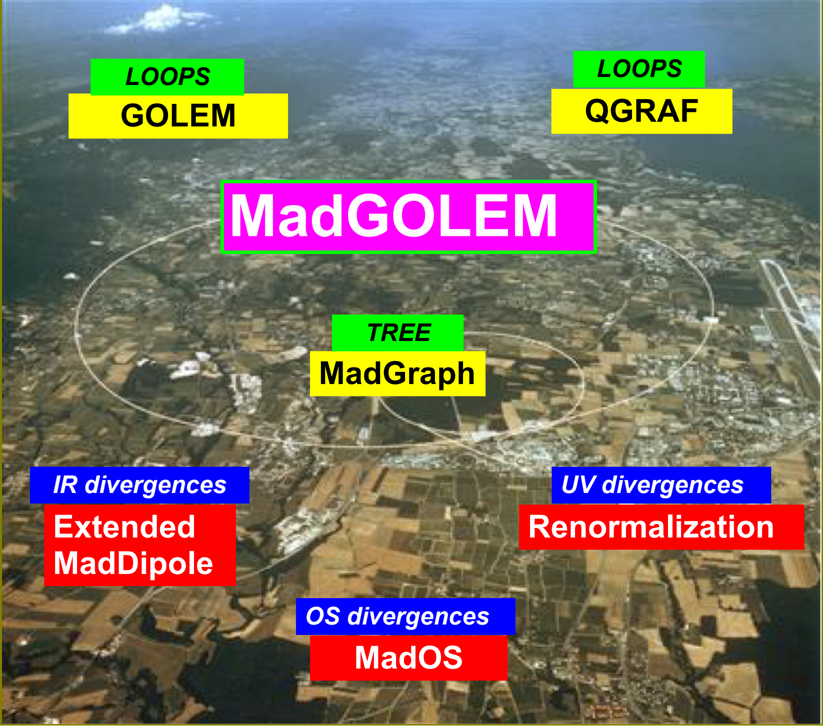
$\Downarrow$

Generalizing

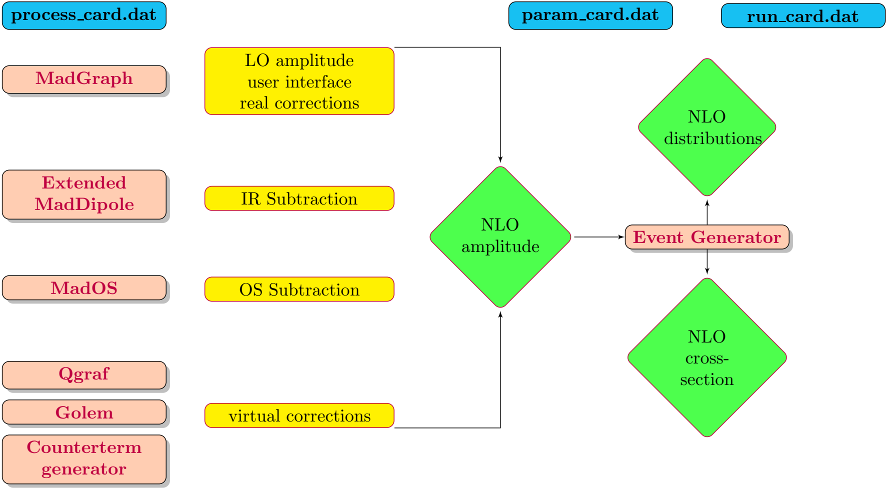


Benakker, Höper, Krämer, Plehn, Spira, Zerwas

Outline


LOOPS
GOLEM
LOOPS
QGRAF
MadGOLEM
TREE
MadGraph
IR divergences
Extended MadDipole
UV divergences
Renormalization
OS divergences
MadOS

MadGolem from inside: modules and flowchart



Handling the loops

♠ GENERATION $\longleftrightarrow$ Qgraf [Nogueira]

Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

Handling the loops

- ♠ GENERATION $\longleftrightarrow$ Qgraf [Nogueira]

  Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

- ♠ TRANSLATION $\longleftrightarrow$ Qgraf-Golem

  Feynman diagrams $\xrightarrow{\text{BASH,PERL,FORM}}$ Amplitude

Handling the loops

- ♠ GENERATION $\longleftrightarrow$ Qgraf [Nogueira]

  Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

- ♠ TRANSLATION $\longleftrightarrow$ Qgraf-Golem

  Feynman diagrams $\xrightarrow{\text{BASH,PERL,FORM}}$ Amplitude

vertices propagators color Lorentz structures relative signs

Handling the loops

- GENERATION $\longleftrightarrow$ Qgraf [Nogueira]

$$\text{Model files} \xrightarrow{\text{FORTRAN}} \text{Feynman diagrams}$$

- TRANSLATION $\longleftrightarrow$ Qgraf-Golem

$$\text{Feynman diagrams} \xrightarrow{\text{BASH,PERL,FORM}} \text{Amplitude}$$

vertices propagators color Lorentz structures relative signs

Handling the loops

- ♠ GENERATION ⟷ Qgraf [Nogueira]

  Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

- ♠ TRANSLATION ⟷ Qgraf-Golem

  Feynman diagrams $\xrightarrow{\text{BASH,PERL,FORM}}$ Amplitude

vertices propagators color Lorentz structures relative signs

Handling the loops

- GENERATION ⟷ Qgraf [Nogueira]

$$\text{Model files} \xrightarrow{\text{FORTRAN}} \text{Feynman diagrams}$$

- TRANSLATION ⟷ Qgraf-Golem

$$\text{Feynman diagrams} \xrightarrow{\text{BASH,PERL,FORM}} \text{Amplitude}$$

vertices propagators color Lorentz structures relative signs

Handling the loops

- ♠ GENERATION ⟷ Qgraf [Nogueira]

  Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

- ♠ TRANSLATION ⟷ Qgraf-Golem

  Feynman diagrams $\xrightarrow{\text{BASH,PERL,FORM}}$ Amplitude

vertices propagators color Lorentz structures relative signs

Handling the loops

- **GENERATION** $\longleftrightarrow$ **Qgraf** [Nogueira]

  Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

- **TRANSLATION** $\longleftrightarrow$ **Qgraf-Golem**

  Feynman diagrams $\xrightarrow{\text{BASH,PERL,FORM}}$ Amplitude

  vertices propagators color Lorentz structures relative signs

- **CALCULATION** $\longleftrightarrow$ **Golem** [Binoth et al.]

  Amplitude $\xrightarrow{\text{BASH,PERL,FORM,MAPLE}}$ Reduced amplitude

Handling the loops

- GENERATION $\longleftrightarrow$ Qgraf [Nogueira]

Model files $\xrightarrow{\text{FORTRAN}}$ Feynman diagrams

- TRANSLATION $\longleftrightarrow$ Qgraf-Golem

Feynman diagrams $\xrightarrow{\text{BASH,PERL,FORM}}$ Amplitude

vertices propagators color Lorentz structures relative signs

- CALCULATION $\longleftrightarrow$ Golem [Binoth et al.]

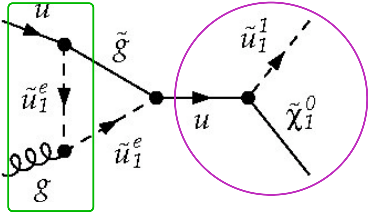
Amplitude $\xrightarrow{\text{BASH,PERL,FORM,MAPLE}}$ Reduced amplitude

$$\mathcal{M}^{\text{NLO}} = \underbrace{\mathcal{M}^{\text{NLO}}_{[\text{color}/\text{helicity}/\text{1l}-\text{function}]}}_{\text{partial amplitudes}} \times \underbrace{\mathcal{B}_{\text{color}} \otimes \mathcal{B}_{\text{hel}} \otimes \mathcal{B}_{\text{1Lfunction}}}_{\text{basis}}$$

Step 1: Generation of the one-loop amplitude

One-loop amplitude from QGRAF: qgraf_nlo.dat

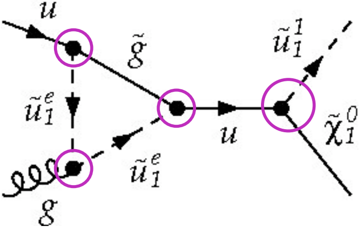
```
+ 1 *
inp([field.u], idx2r2, p1) *
inplorentz(+1, iv2r2L1, p1, ZERO  ) *
inpcolor(1, iv2r2C3) *
inp([field.g], idx3r1, p2) *
inplorentz(+2, iv3r1L2, p2, ZERO  ) *
inpcolor(2, iv3r1C8) *
out([field.ul], idx1r3, p3) *
outlorentz(+0, iv1r3L0, p3, MUL   ) *
outcolor(1, iv1r3C3) *
out([field.n1], idx1r1, p4) *
outlorentz(+1, iv1r1L1, p4, MN1   ) *
outcolor(2, iv1r1C1) *
vertex(iv1,GULN1P  ,ONE,
[field.n1], idx1r1, +1, -p4, iv1r1L1, +1, iv1r1C1,
[field.u], idx1r2, +1, p3+p4, iv1r2L1, +3, iv1r2C3,
[field.ulx], idx1r3, -0, -p3, iv1r3L0, -3, iv1r3C3) *
vertex(iv2,GQLGOP  ,ONE,
[field.go], idx2r1, +1, k1-p1, iv2r1L1, +8, iv2r1C8,
[field.u], idx2r2, +1, p1, iv2r2L1, +3, iv2r2C3,
[field.ulx], idx2r3, -0, -k1, iv2r3L0, -3, iv2r3C3) *
vertex(iv3,GC      ,ONE,
[field.g], idx3r1, +2, p2, iv3r1L2, +8, iv3r1C8,
[field.ul], idx3r2, +0, k1, iv3r2L0, +3, iv3r2C3,
[field.ulx], idx3r3, -0, -k1-p2, iv3r3L0, -3, iv3r3C3) *
```



Step 1: Generation of the one-loop amplitude

One-loop amplitude from QGRAF: qgraf_nlo.dat

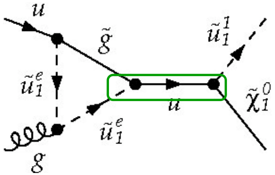
```
+ 1 *
inp([field.u], idx2r2, p1) *
inplorentz(+1, iv2r2L1, p1, ZERO  ) *
inpcolor(1, iv2r2C3) *
inp([field.g], idx3r1, p2) *
inplorentz(+2, iv3r1L2, p2, ZERO  ) *
inpcolor(2, iv3r1C8) *
out([field.ul], idx1r3, p3) *
outlorentz(+0, iv1r3L0, p3, MUL   ) *
outcolor(1, iv1r3C3) *
out([field.n1], idx1r1, p4) *
outlorentz(+1, iv1r1L1, p4, MN1   ) *
outcolor(2, iv1r1C1) *
vertex(iv1,GULN1P   ,ONE,
[field.n1], idx1r1, +1, -p4, iv1r1L1, +1, iv1r1C1,
[field.u], idx1r2, +1, p3+p4, iv1r2L1, +3, iv1r2C3,
[field.ulx], idx1r3, -0, -p3, iv1r3L0, -3, iv1r3C3) *
vertex(iv2,GQLGOP   ,ONE,
[field.go], idx2r1, +1, k1-p1, iv2r1L1, +8, iv2r1C8,
[field.u], idx2r2, +1, p1, iv2r2L1, +3, iv2r2C3,
[field.ulx], idx2r3, -0, -k1, iv2r3L0, -3, iv2r3C3) *
vertex(iv3,GC       ,ONE,
[field.g], idx3r1, +2, p2, iv3r1L2, +8, iv3r1C8,
[field.ul], idx3r2, +0, k1, iv3r2L0, +3, iv3r2C3,
[field.ulx], idx3r3, -0, -k1-p2, iv3r3L0, -3, iv3r3C3) *
```



Step 1: Generation of the one-loop amplitude

One-loop amplitude from QGRAF: qgraf_nlo.dat

```
vertex(iv4,GQLGOM  ,ONE,
[field.ux], idx4r1, -1, -p3-p4, iv4r1L1, -3, iv4r1C3,
[field.go], idx4r2, +1, -k1+p1, iv4r2L1, +8, iv4r2C8,
[field.ul], idx4r3, +0, k1+p2, iv4r3L0, +3, iv4r3C3) *
prop([field.u], idx4r1, idx1r2) *
propcolor(+3, iv4r1C3, iv1r2C3) *
proplorentz(+1, p3+p4, ZERO  , iv4r1L1, iv1r2L1) *
prop([field.ul], idx2r3, idx3r2) *
propcolor(+3, iv2r3C3, iv3r2C3) *
proplorentz(+0, k1, MUL   , iv2r3L0, iv3r2L0) *
prop([field.go], idx4r2, idx2r1) *
propcolor(+8, iv4r2C8, iv2r1C8) *
proplorentz(+1, k1-p1, MGO   , iv4r2L1, iv2r1L1) *
prop([field.ul], idx3r3, idx4r3) *
propcolor(+3, iv3r3C3, iv4r3C3) *
proplorentz(+0, k1+p2, MUL   , iv3r3L0, iv4r3L0)
```



Step 2: Translation of the one-loop amplitude

One-loop amplitude upon translation: GRAPH_MGOLEM_LOOP.h

```
diagram1 =  + Den(k1 + k2,0)*Den( - k3 - k4,0)*intM(Den(q1,0),Den(k1
     + k2 + q1,0))*SUNSum(Col14,3)*SUNSum(Glu15,8)*SUNSum(Col18,3)*
    SUNT(Glu2,Col14,Col1)*SUNT(Glu15,Col3,Col18)*SUNT(Glu15,Col18,
    Col14)*GG1^3*scalar3*Pi^(-2) * ( 1/32*Spinor(k4,MN1,-1)*g_(2,7_,
       k3,Lor5,k1,Lor5,k1,Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2
        + 1/32*Spinor(k4,MN1,-1)*g_(2,7_,k3,Lor5,k1,Lor5,k2,Lorhx2)*
       Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2 + 1/32*Spinor(k4,MN1,-1)*g_(
       2,7_,k3,Lor5,k2,Lor5,k1,Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*
       GULN1P2 + 1/32*Spinor(k4,MN1,-1)*g_(2,7_,k3,Lor5,k2,Lor5,k2,
       Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2 + 1/32*Spinor(k4,MN1
       ,-1)*g_(2,7_,k3,Lor5,q1,Lor5,k1,Lorhx2)*Spinor(k1,0,1)*
       e2(Lorhx2)*GULN1P2 + 1/32*Spinor(k4,MN1,-1)*g_(2,7_,k3,Lor5,q1
       ,Lor5,k2,Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2 + 1/32*
       Spinor(k4,MN1,-1)*g_(2,7_,k4,Lor5,k1,Lor5,k1,Lorhx2)*Spinor(k1
       ,0,1)*e2(Lorhx2)*GULN1P2 + 1/32*Spinor(k4,MN1,-1)*g_(2,7_,k4,
       Lor5,k1,Lor5,k2,Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2 + 1/
       32*Spinor(k4,MN1,-1)*g_(2,7_,k4,Lor5,k2,Lor5,k1,Lorhx2)*
       Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2 + 1/32*Spinor(k4,MN1,-1)*g_(
       2,7_,k4,Lor5,k2,Lor5,k2,Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*
       GULN1P2 + 1/32*Spinor(k4,MN1,-1)*g_(2,7_,k4,Lor5,q1,Lor5,k1,
       Lorhx2)*Spinor(k1,0,1)*e2(Lorhx2)*GULN1P2 + 1/32*Spinor(k4,MN1
       ,-1)*g_(2,7_,k4,Lor5,q1,Lor5,k2,Lorhx2)*Spinor(k1,0,1)*
```

Step 3: Calculation of the one-loop amplitude

One-loop amplitude upon translation: GRAPH_MGOLEM_LOOP.h

```
# Colour basis has   2 elements.
#
NCOLS :=   2:
COL[  1] :=  dd(Col022,Col021)*dd(Col1,Col3):
COL[  2] :=  dd(Col022,Col3)*dd(Col1,Col021):
NC := 3:
TR := 1/2:
NF := 5:
#
# epsilon_tensor basis has   1 elements.
#
NEPS :=   1:
EPSTEN[  1] :=  1:
#
# Function basis has  4 elements.
#
NUM_LOC_FUNS :=  4:
FUN[  1] :=  BUBd4(S12,0,0):
FUN[  2] :=  BUBd4(S12,MUL2,MG02):
FUN[  3] :=  BUBd4EPS(S12,0,0):
FUN[  4] :=  BUBd4EPS(S12,MUL2,MG02):
#
#  12 helicity amplitudes found
#
NUM_HELIS :=  12:
HELI[  1]:=[1, 1, 5, 1]:
HELI[  2]:=[1, 1, 5, -1]:
```

Step 3: Calculation of the one-loop amplitude

One-loop amplitude upon translation: GRAPH_MGOLEM_LOOP.h

```
GRAPH_COEFF has indices: NGRAPH,NHELI,NCOL,NEPSTEN,NFUN

GRAPH_COEFF[  1,  8,  1,  1,  1]:=-1/36*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  2,  1,  1]:=1/12*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  1,  1,  2]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  2]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  3]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  3]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  4]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  4]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  5]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  5]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  6]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  6]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  7]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  7]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  8]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  8]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  9]:=1/36*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  2,  1,  9]:=-1/12*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  1,  1, 10]:=0:
GRAPH_COEFF[  1,  8,  2,  1, 10]:=0:
GRAPH_COEFF[  1,  8,  1,  1, 11]:=0:
GRAPH_COEFF[  1,  8,  2,  1, 11]:=0:
SPINOR_FAC[  1,  8    ] :=  InvSpaa(k1,k2)*InvSpaa(k1,k4)*InvSpaa(k2,k3)*InvSpbb(k1,k3):
```

Step 3: Calculation of the one-loop amplitude

One-loop amplitude upon translation: GRAPH_MGOLEM_LOOP.h

```
GRAPH_COEFF has indices: NGRAPH,NHELI,NCOL,NEPSTEN,NFUN

GRAPH_COEFF[  1,  8,  1,  1,  1]:=-1/36*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  2,  1,  1]:=1/12*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  1,  1,  2]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  2]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  3]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  3]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  4]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  4]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  5]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  5]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  6]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  6]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  7]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  7]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  8]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  8]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  9]:=1/36*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  2,  1,  9]:=-1/12*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  1,  1, 10]:=0:
GRAPH_COEFF[  1,  8,  2,  1, 10]:=0:
GRAPH_COEFF[  1,  8,  1,  1, 11]:=0:
GRAPH_COEFF[  1,  8,  2,  1, 11]:=0:
SPINOR_FAC[  1,  8     ] :=  InvSpaa(k1,k2)*InvSpaa(k1,k4)*InvSpaa(k2,k3)*InvSpbb(k1,k3):
```

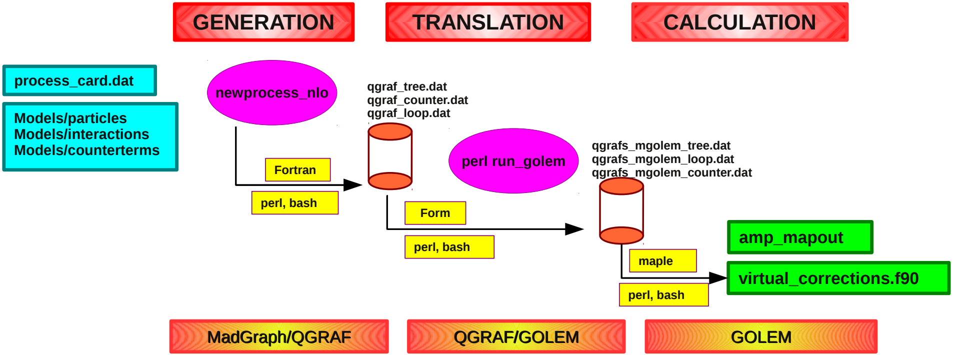
Step 3: Calculation of the one-loop amplitude

One-loop amplitude upon translation: GRAPH_MGOLEM_LOOP.h

```
GRAPH_COEFF has indices: NGRAPH,NHELI,NCOL,NEPSTEN,NFUN

GRAPH_COEFF[  1,  8,  1,  1,  1]:=-1/36*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  2,  1,  1]:=1/12*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  1,  1,  2]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  2]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  3]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  3]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  4]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  4]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  5]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  5]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  6]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  6]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  7]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  7]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  8]:=0:
GRAPH_COEFF[  1,  8,  2,  1,  8]:=0:
GRAPH_COEFF[  1,  8,  1,  1,  9]:=1/36*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  2,  1,  9]:=-1/12*(-MUL2*S23+MUL2*MN12+S23^2+S12*S23-MN12*S23)*GULN1P2*GG1^3/Pi^2:
GRAPH_COEFF[  1,  8,  1,  1, 10]:=0:
GRAPH_COEFF[  1,  8,  2,  1, 10]:=0:
GRAPH_COEFF[  1,  8,  1,  1, 11]:=0:
GRAPH_COEFF[  1,  8,  2,  1, 11]:=0:
SPINOR_FAC[  1,  8    ] :=  InvSpaa(k1,k2)*InvSpaa(k1,k4)*InvSpaa(k2,k3)*InvSpbb(k1,k3):
```

Handling the loops - summary



Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding***: *efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding**: efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding**: efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding***: *efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding****: efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding**: efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

♠ ***Genuine New Physics structures treatable***

- Majorana fermions (clashing arrows !)
- complex color & spin structures
- higher dimensional couplings

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding**: efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

♠ ***Genuine New Physics structures treatable***

- Majorana fermions (clashing arrows !)
- complex color & spin structures
- higher dimensional couplings

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding**: efficient generation & numerical evaluation*

- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

♠ ***Genuine New Physics structures treatable***

- Majorana fermions (clashing arrows !)
- complex color & spin structures
- higher dimensional couplings

Handling the loops: in best shape for BSM

♠ ***Analytical results accessible at any time***

♠ ***Dedicated coding**: efficient generation & numerical evaluation*

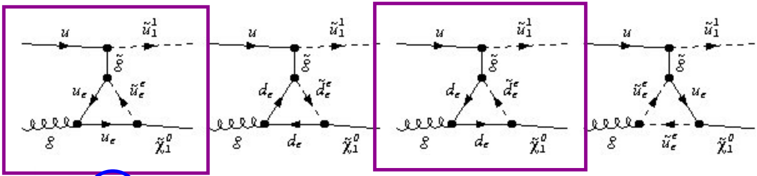
- Loop filtering – Automatic removal of vanishing partial amplitudes from further analytical processing
- Grouping of topologically equivalent one-loop diagrams
- Handling of iterative structures
- Amplitude coefficients as split shared libraries

♠ ***Genuine New Physics structures treatable***

- Majorana fermions (clashing arrows !)
- complex color & spin structures
- higher dimensional couplings

Highlighting the major improvements

♠ Grouping of topologically equivalent one-loop diagrams



```
+ 1 *2*
inp([field.u], idx1r2, p1) *
inplorentz(+1, iv1r2L1, p1, ZERO   ) *
inpcolor(1, iv1r2C3) *
inp([field.g], idx2r3, p2) *
inplorentz(+2, iv2r3L2, p2, ZERO   ) *
inpcolor(2, iv2r3C8) *
out([field.ur], idx1r3, p3) *
outlorentz(+0, iv1r3L0, p3, MUR    ) *
outcolor(1, iv1r3C3) *
out([field.n1], idx3r2, p4) *
outlorentz(+1, iv3r2L1, p4, MN1    ) *
outcolor(2, iv3r2C1) *
```

Handling the UV divergences

Including the Counterterms

$$\mathcal{L}_0 \to \mathcal{L}(Z_\phi^{1/2}\,\phi, Z_g\,g) \quad = \quad \mathcal{L}(\phi, g) + \delta\,\mathcal{L}\,(\phi, g, \delta\,Z_\phi, \delta\,g)$$

Handling the UV divergences

Including the Counterterms

$$\mathcal{L}_0 \to \mathcal{L}(Z_\phi^{1/2}\,\phi, Z_g\,g) \quad = \quad \mathcal{L}(\phi, g) + \delta\,\mathcal{L}\,(\phi, g, \delta\,Z_\phi, \delta\,g)$$

$$\underbrace{\delta\mathcal{L}(\delta\,Z_\phi, \delta\,g)}_{\text{Models/vertex\_ct.dat}} \qquad \Leftrightarrow \qquad \underbrace{\Sigma_q, \quad \Sigma_{\tilde{q}}, \quad \Sigma_g, \quad \Sigma_{\tilde{g}} \qquad @\mathcal{O}(\alpha_s)}_{\text{GOLEMproc/CT\_list.map}}$$

Handling the UV divergences

Including the Counterterms

$$\mathcal{L}_0 \to \mathcal{L}(Z_\phi^{1/2}\,\phi, Z_g\,g) \quad = \quad \mathcal{L}(\phi, g) + \delta\,\mathcal{L}\,(\phi, g, \delta\,Z_\phi, \delta\,g)$$

$$\underbrace{\delta\mathcal{L}(\delta\,Z_\phi, \delta\,g)}_{\text{Models/vertex\_ct.dat}} \qquad \Leftrightarrow \qquad \underbrace{\Sigma_q, \quad \Sigma_{\tilde{q}}, \quad \Sigma_g, \quad \Sigma_{\tilde{g}} \qquad @\mathcal{O}(\alpha_s)}_{\text{GOLEMproc/CT\_list.map}}$$

Renormalization scheme

- $\overline{\mathrm{MS}}$, for the field-strength RCs of the massless particles
- OS, for the field-strength RCs of the massive particles
- $\overline{\mathrm{MS}}$/zero-momentum, for g_s [Beenakker et al, Berge et al]
- SUSY breaking from Dimensional Regularization restored through additional finite CTs [Martin, Vaughn; Beenakker et al].

Handling the IR divergences

♠ **Dipole Subtraction:** [Catani, Seymour; Catani, Dittmaier, Seymour, Trócsányi]

$$\sigma = \int_m d\,\sigma^B + \int_{m+1} d\,\sigma^R + \int_m \left[\int_1 d\,\sigma^V \right]$$

Handling the IR divergences

♠ Dipole Subtraction: [Catani, Seymour; Catani, Dittmaier, Seymour, Trócsányi]

$$\sigma = \int_m d\sigma^B + \underbrace{\int_{m+1} d\sigma^R}_{\int \frac{dk}{k^{1+\epsilon}} \sim \frac{1}{\epsilon_{IR}}, \int \frac{d\theta}{\theta^{1+\epsilon}} \sim \frac{1}{\epsilon_{IR}}} + \int_m \left[\int_1 \underbrace{d\sigma^V}_{\int \frac{dk}{k^{1+\epsilon}} \sim \frac{1}{\epsilon_{IR}}} \right]$$

Handling the IR divergences

♠ Dipole Subtraction: [Catani, Seymour; Catani, Dittmaier, Seymour, Trócsányi]

$$\sigma = \int_m d\sigma^B + \underbrace{\int_{m+1} d\sigma^R - d\sigma^A}_{\frac{1}{\epsilon_{IR}} - \frac{1}{\epsilon_{IR}}} + \int_m \left[\int_1 \underbrace{d\sigma^V + d\sigma^A}_{\frac{1}{\epsilon_{IR}} - \frac{1}{\epsilon_{IR}}} \right]$$

Handling the IR divergences

♠ Dipole Subtraction: [Catani, Seymour; Catani, Dittmaier, Seymour, Trócsányi]

$$\sigma = \int_m d\sigma^B + \underbrace{\int_{m+1} d\sigma^R - d\sigma^A}_{\frac{1}{\epsilon_{IR}} - \frac{1}{\epsilon_{IR}}} + \int_m \left[\int_1 \underbrace{d\sigma^V + d\sigma^A}_{\frac{1}{\epsilon_{IR}} - \frac{1}{\epsilon_{IR}}} \right]$$

Local (pointwise) subtraction of the IR poles

- Based on factorization of collinear&soft singularities
- Process-independent
- Analytically integrable over the single-parton phase-space containing the divergences

Handling the IR divergences

♠ **Dipole Subtraction:** [Catani, Seymour; Catani, Dittmaier, Seymour, Trócsányi]

$$\sigma = \int_m d\,\sigma^B + \underbrace{\int_{m+1} d\,\sigma^R - d\,\sigma^A}_{\frac{1}{\epsilon_{IR}} - \frac{1}{\epsilon_{IR}}} + \int_m \left[\int_1 \underbrace{d\,\sigma^V + d\,\sigma^A}_{\frac{1}{\epsilon_{IR}} - \frac{1}{\epsilon_{IR}}} \right]$$

Local (pointwise) subtraction of the IR poles

- Based on factorization of collinear&soft singularities
- Process-independent
- Analytically integrable over the single-parton phase-space containing the divergences

$$d\sigma^A = \sum_l f(\epsilon_{IR}) \times d\sigma_l^B \otimes V_l$$

$$\int_{m+1} d\,\sigma^A = \sum_l \int_m f(\epsilon_{IR}) \times d\,\sigma_l^B \otimes \int_1 V_l = f(\epsilon_{IR}) \times \int_m d\sigma_l^B \otimes I$$

I_{SUSY} (including α-dependence [Nagy, Trócsányi]) available @ MadGolem

Handling the OS Divergences

Automatized OS Subtraction available @ MadGolem [Beenakker, Höpker, Spira, Zerwas]

$$d\sigma^R \longrightarrow d\sigma^R\Big]_{\text{regular}} + d\sigma^{R*}\Big]_{\mathcal{O}(1/(p^2-m^2))}$$

$$ug \to \tilde{u}_L \tilde{\chi}_1 j \quad + \quad uu \to \tilde{u}_L \tilde{u}_L^* \to \tilde{u}_L \tilde{\chi}_1 j$$

Handling the OS Divergences

Automatized OS Subtraction available @ MadGolem [Beenakker, Höpker, Spira, Zerwas]

$$d\sigma^R \longrightarrow d\sigma^R\Big]_{\text{regular}} + d\sigma^{R*}\Big]_{\mathcal{O}(1/(p^2-m^2))}$$

$$ug \to \tilde{u}_L \tilde{\chi}_1 j \quad + \quad uu \to \tilde{u}_L \tilde{u}^*_L \to \tilde{u}_L \tilde{\chi}_1 j$$

$$\sigma = \int_{m+1} d\,\sigma^R$$

Handling the OS Divergences

Automatized OS Subtraction available @ MadGolem [Beenakker, Höpker, Spira, Zerwas]

$$d\sigma^{R} \longrightarrow d\sigma^{R}\Big]_{\text{regular}} + d\sigma^{R*}\Big]_{\mathcal{O}(1/(p^2-m^2))}$$

$$ug \to \tilde{u}_L \tilde{\chi}_1 j \quad + \quad uu \to \tilde{u}_L \tilde{u}_L^* \to \tilde{u}_L \tilde{\chi}_1 j$$

$$\sigma = \int_{m+1} d\,\sigma^{R} \longrightarrow \int_{m+1} \Big[d\,\sigma^{R} + d\,\sigma^{R*}(\Gamma_{\tilde{u}_L}) - d\sigma^{OS}(\Gamma_{\tilde{u}_L}) \Big]$$

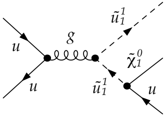
Handling the OS Divergences

Automatized OS Subtraction available @ MadGolem [Beenakker, Höpker, Spira, Zerwas]

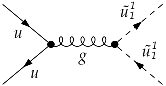
$$d\sigma^R \longrightarrow d\sigma^R\Big]_{\text{regular}} + d\sigma^{R*}\Big]_{\mathcal{O}(1/(p^2-m^2))}$$

$$ug \to \tilde{u}_L\tilde{\chi}_1 j \quad + \quad uu \to \tilde{u}_L\tilde{u}_L^* \to \tilde{u}_L\tilde{\chi}_1 j$$

$$\sigma = \int_{m+1} d\,\sigma^R \longrightarrow \int_{m+1} \left[d\,\sigma^R + d\,\sigma^{R*}(\Gamma_{\tilde{u}_L}) - d\sigma^{OS}(\Gamma_{\tilde{u}_L})\right]$$



\-



$$uu \to \tilde{u}_L\tilde{u}_L^* \to \tilde{u}_L\tilde{\chi}_1 j \quad - \quad uu \to \tilde{u}_L\tilde{u}_L \times \mathcal{B}(\tilde{u}_L \to \tilde{\chi}_1 j)$$

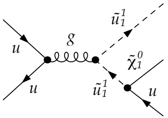
Handling the OS Divergences

Automatized OS Subtraction available @ MadGolem [Beenakker, Höpker, Spira, Zerwas]

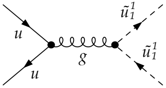
$$d\sigma^R \longrightarrow d\sigma^R\Big]_{\text{regular}} + d\sigma^{R*}\Big]_{\mathcal{O}(1/(p^2-m^2))}$$

$$ug \to \tilde{u}_L \tilde{\chi}_1 j \quad + \quad uu \to \tilde{u}_L \tilde{u}_L^* \to \tilde{u}_L \tilde{\chi}_1 j$$

$$\sigma = \int_{m+1} d\,\sigma^R \longrightarrow \int_{m+1} \left[d\,\sigma^R + d\,\sigma^{R*}(\Gamma_{\tilde{u}_L}) - d\sigma^{OS}(\Gamma_{\tilde{u}_L}) \right]$$



-



$$uu \to \tilde{u}_L \tilde{u}_L^* \to \tilde{u}_L \tilde{\chi}_1 j \quad - \quad uu \to \tilde{u}_L \tilde{u}_L \times \mathcal{B}(\tilde{u}_L \to \tilde{\chi}_1 j)$$

$$\frac{d\sigma^{OS}}{dM^2} = \sigma^{Born} \frac{m_{\tilde{u}_L} \Gamma_{\tilde{u}_L}/\pi}{(M^2 - m^2_{\tilde{u}_L}) + m^2\, \Gamma^2_{\tilde{u}_L}} + \mathcal{O}\left(\frac{1}{(M^2 - m^2_{\tilde{u}_L})} \right)$$

- Pointwise subtraction of the OS poles – analogue to CS dipoles
- Avoids double-counting & preserves gauge invariance & spin correlations
- $\Gamma_{\tilde{u}_L}$ as regulator $\Rightarrow$ dependence cancels in the final results

Outline

Testing MadGolem: virtual corrections

♠ Validation strategies

- gauge invariance
- cancellation of UV/IR divergences (analytically & numerically)
- Finite parts – numerical comparison with FeynArts/FormCalc/LoopTools [Hahn]

Testing MadGolem: virtual corrections

Validation strategies

- gauge invariance
- cancellation of UV/IR divergences (analytically & numerically)
- Finite parts – numerical comparison with FeynArts/FormCalc/LoopTools [Hahn]

An explicit example: SUSY-QCD 1-loop virtual corrections to $u\bar{u} \to \tilde{\chi}_1^0 \tilde{\chi}_1^0$

	σ^{NLO} (MadGolem)	σ^{NLO} (FormCalc)
$Z^0 - \bar{u} - u$ vertex	$-1.3064(1) \times 10^{-7}$	$-1.3064(5) \times 10^{-7}$
$\tilde{\chi}_1 - u - \tilde{u}_L$ vertex	$1.0916(3) \times 10^{-8}$	$1.091(2) \times 10^{-8}$
$\tilde{u}_L$ self-energy	$1.6375(1) \times 10^{-5}$	$1.637(1) \times 10^{-5}$
boxes	$1.858(2) \times 10^{-7}$	$1.86(4) \times 10^{-7}$

Testing MadGolem: IR divergences

♠ Validation strategies

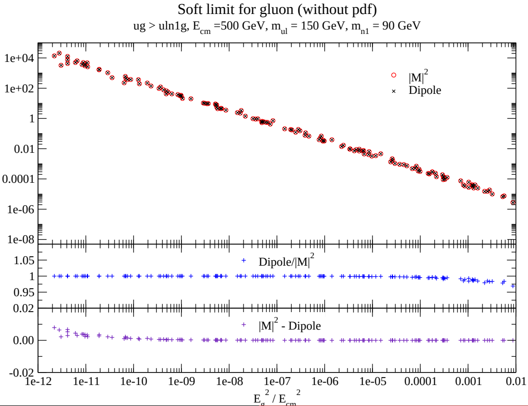
- α dependence [Nagy, Trócsányi] & behavior in the soft and collinear limits, for all dipoles
- Numerical stability and convergence

Testing MadGolem: IR divergences

Validation strategies

- α dependence [Nagy, Trócsányi] & behavior in the soft and collinear limits, for all dipoles
- Numerical stability and convergence

Diving in the soft/collinear region for $pp \to \tilde{u}_L \tilde{\chi}_1$

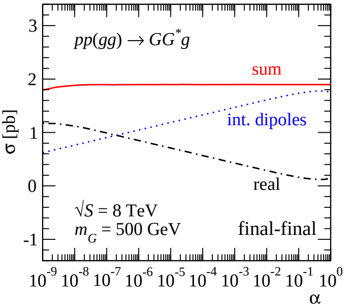


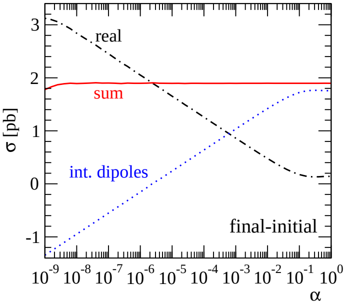
Testing MadGolem: IR divergences

Validation strategies

- α dependence [Nagy, Trócsányi] & behavior in the soft and collinear limits, for all dipoles
- Numerical stability and convergence

Checking α-dependence for $pp \to GG^*$





Testing MadGolem: On-shell subtraction

♠ Validation strategies

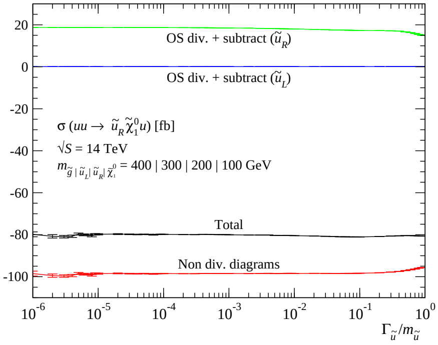
- Independence with respect to the regulator choice Γ/m
- Numerical stability and convergence

Testing MadGolem: On-shell subtraction

Validation strategies

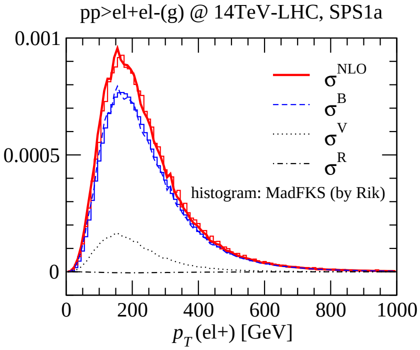
- Independence with respect to the regulator choice Γ/m
- Numerical stability and convergence

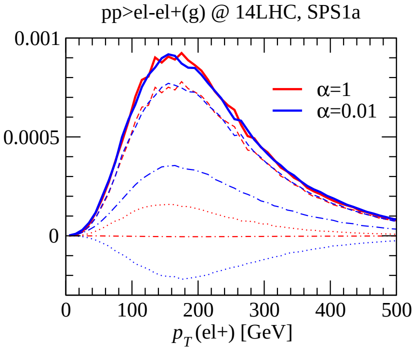
Performance of the OS Subtraction: $pp(uu \to \tilde{u}_R \tilde{\chi}_1 u)$



Testing MadGolem: distributions

♠ Cross-checking distributions: MadGolem meets MadFKS





Testing MadGolem: computing & comparing

- Systematic cross-check always involve:

Testing MadGolem: computing & comparing

- Systematic cross-check always involve:

i) UV/IR cancellation;

Testing MadGolem: computing & comparing

- Systematic cross-check always involve:

i) UV/IR cancellation;

ii) Finite parts of the one-loop amplitude - comparison to FormCalc/Looptools;

Testing MadGolem: computing & comparing

- Systematic cross-check always involve:

i) UV/IR cancellation;

ii) Finite parts of the one-loop amplitude - comparison to FormCalc/Looptools;

iii) Full NLO cross section – when available, e.g. from **GoSam**, **MadLoop** or **Prospino**

Testing MadGolem: computing & comparing

- Systematic cross-check always involve:

i) UV/IR cancellation;

ii) Finite parts of the one-loop amplitude - comparison to FormCalc/Looptools;

iii) Full NLO cross section – when available, e.g. from **GoSam**, **MadLoop** or **Prospino**

(almost) complete list of processes

SM processes	♠ $pp \to e^+e^-$ ♠ $u\bar{u} \to d\bar{d}$; $gg \to d\bar{d}$ ♠ $u\bar{u} \to t\bar{t}$, $gg \to t\bar{t}$
SUSY pair (EW)	♣ $pp \to \tilde{l}\tilde{l}^*$ ♣ $pp \to \tilde{\chi}\tilde{\chi}$; $pp \to \tilde{g}\tilde{\chi}$ ♣ $pp \to \tilde{q}\tilde{\chi}$
SUSY pair (QCD)	♦ $pp \to \tilde{q}\tilde{q}^*$; $pp \to \tilde{q}\tilde{q}$ ♦ $pp \to \tilde{q}\tilde{g}$ ♦ $pp \to \tilde{g}\tilde{g}$
Other BSM	▲ $pp \to GG^*$; ▲ $pp \to \mathrm{H}^{\pm}t$; ▲ $pp \to t + X$

Outline

Running MadGolem: the user's perspective

3-stage procedure – 3 interfaces ↔ 3 executables

Stage 1: DEFINING THE PROCESS

process_card ↔ ./newprocess_nlo

Running MadGolem: the user's perspective

3-stage procedure – 3 interfaces ↔ 3 executables

Stage 1: DEFINING THE PROCESS

process_card ↔ ./newprocess_nlo

Stage 2: COMPUTING THE AMPLITUDE

./run_golem_pl

- At this point the user is able to:
  - Select diagram topologies ⇒ detailed analysis of the virtual corrections
  - Access the **analytical output** in several stages ⇒ very useful for cross-checking (and to dig out some physics!)

Running MadGolem: the user's perspective

3-stage procedure – 3 interfaces ↔ 3 executables

Stage 1: DEFINING THE PROCESS

process_card ↔ ./newprocess_nlo

Stage 2: COMPUTING THE AMPLITUDE

./run_golem_pl

♠ At this point the user is able to:

- Select diagram topologies ⇒ detailed analysis of the virtual corrections
- Access the **analytical output** in several stages ⇒ very useful for cross-checking (and to dig out some physics!)

Stage 3: EVALUATING THE CROSS-SECTIONS

param_card.dat, run_card.dat ↔ ./generate_events_nlo 2 2 myrun

Running MadGolem: the user's perspective

♠ And the user retrieves the results !

MadGolem results
s= 8533.480± 0.793(ab)

K-factor=(P1+P2+P3)/P1= 1.119

Graph	Cross Sect(ab)	Error(ab)	Events (K)	Eff	Unwgt	Luminosity
Sum	**8533.480**	**0.793**	**0**	**0.0**		
LEADING ORDER						
P1_e-e+_ululx	7625.900	0.506	0	0.0		0.00
total LO = 7625.8999999999996						
NLO CONTRIBUTION: Virtual part						
P2_e-e+_ululxg	578.990	0.068	0	0.0		0.00
total NLO (virtual part)= 578.99000000000001						
NLO CONTRIBUTION: Real part						
P3_e-e+_ululxg	328.590	0.219	0	0.0		0.00
total NLO (real part) = 328.59000000000003						

Exploring New Physics with MadGolem

First complete fully automated NLO calculations of BSM $2 \to 2$

Exploring New Physics with MadGolem

First complete fully automated NLO calculations of BSM $2 \to 2$

$pp \to \tilde{q}\tilde{\chi}_0$

arXiv:1108.1250

$pp \to GG^*$

arXiv:1203.6358

Exploring New Physics with MadGolem

First complete fully automated NLO calculations of BSM $2 \to 2$

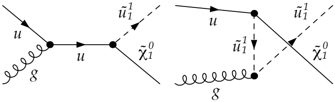
$pp \to \tilde{q}\tilde{\chi}_0$ $pp \to GG^*$

arXiv:1108.1250 arXiv:1203.6358

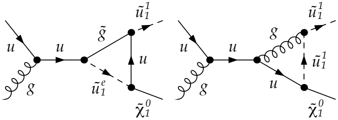
Phenomenological analysis includes

- Total rates & K factors to NLO
- Structure of NLO corrections
- scale dependence
- parameter space dependence
- Distributions
- Fixed-order NLO VS multi-jet merging comparisons

Closing in on our final endeavour



tree-level

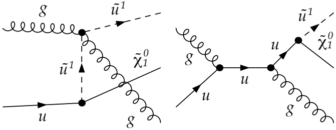


virtual corrections $\mathcal{O}(\alpha_{ew}\,\alpha_s^2)$

$\Downarrow$

MSSM renormalization

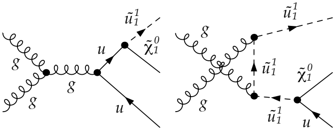
& SUSY-restoration



real corrections $\mathcal{O}(\alpha_{ew}\,\alpha_s^2)$

$\Downarrow$

ii , fi & if dipoles



OS divergent real corrections $\mathcal{O}(\alpha_{ew}\,\alpha_s^2)$

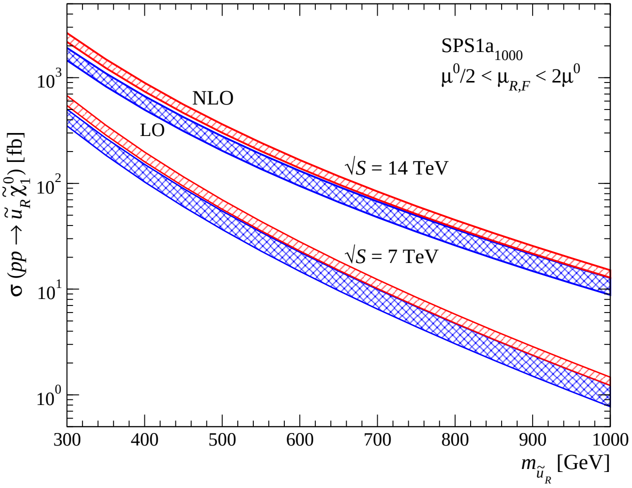
$\Downarrow$

OS subtraction

Scale dependence

Scale dependence for $pp \to \tilde{q}\tilde{\chi}^0$

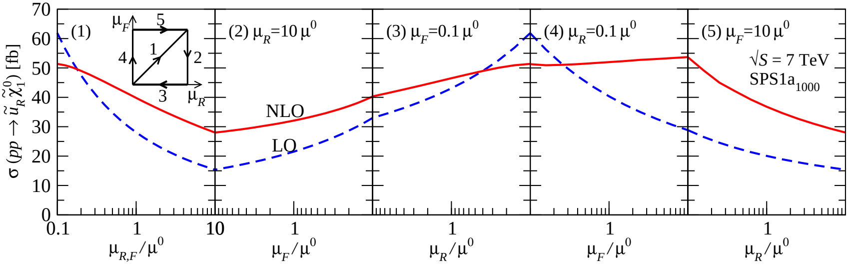
- Strongly reduced theory uncertainty: $\Delta\sigma/\sigma \sim 20\%$ @ NLO VS $\sim 70\%$ @LO



Scale dependence

Scale dependence for $pp \to \tilde{q}\tilde{\chi}^0$

- Similarly sensitive to variations of μ_F/μ_R alike



Binoth, Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1108.1250 [hep-ph]

Parameter space dependence

♠ parameter space dependence for $pp \to \tilde{q}\tilde{\chi}^0$

Total rates & K-factors for different SPS points

	$\sqrt{S}$	σ^{LO}	σ^{NLO}	K	$m_{\tilde{u}}$	$m_{\tilde{d}}$	$m_{\tilde{g}}$	$m_{\tilde{\chi}1}$
Benchmark	7	35.27	50.44	1.43	$\tilde{u}_l$: 561	$\tilde{d}_l$: 568	1000	97
	14	215.02	301.27	1.40	$\tilde{u}_r$: 549	$\tilde{d}_r$: 545		
SPS1b	7	2.77	3.99	1.45	$\tilde{u}_l$: 872	$\tilde{d}_l$: 878	938	162
	14	27.21	37.46	1.38	$\tilde{u}_r$: 850	$\tilde{d}_r$: 843		
SPS2	7	0.04	0.07	1.52	$\tilde{u}_l$: 1554	$\tilde{d}_l$: 1559	782	123
	14	1.21	1.64	1.36	$\tilde{u}_r$: 1554	$\tilde{d}_r$: 1552		
SPS3	7	3.15	4.55	1.44	$\tilde{u}_l$: 854	$\tilde{d}_l$: 860	935	161
	14	30.20	41.59	1.38	$\tilde{u}_r$: 832	$\tilde{d}_r$: 824		
SPS7	7	2.19	3.17	1.45	$\tilde{u}_l$: 896	$\tilde{d}_l$: 904	950	163
	14	22.36	30.80	1.38	$\tilde{u}_r$: 875	$\tilde{d}_r$: 870		
SPS8	7	0.65	0.95	1.45	$\tilde{u}_l$: 1113	$\tilde{d}_l$: 1122	839	139
	14	8.73	11.79	1.35	$\tilde{u}_r$: 1077	$\tilde{d}_r$: 1072		
SPS9	7	0.39	0.58	1.49	$\tilde{u}_l$: 1276	$\tilde{d}_l$: 1279	1872	187
	14	7.65	10.42	1.36	$\tilde{u}_r$: 1282	$\tilde{d}_r$: 1289		

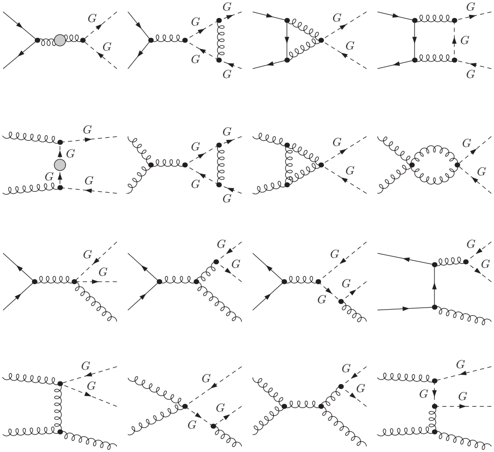
Binoth, Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1108.1250 [hep-ph]

Structure of the NLO QCD corrections

Structure of the NLO QCD corrections for $pp \to GG^*$

Virtual

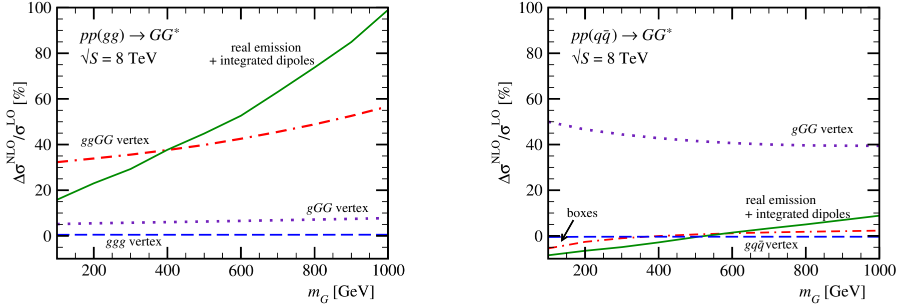
Real



Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1203.6358 [hep-ph]

Structure of the NLO QCD corrections

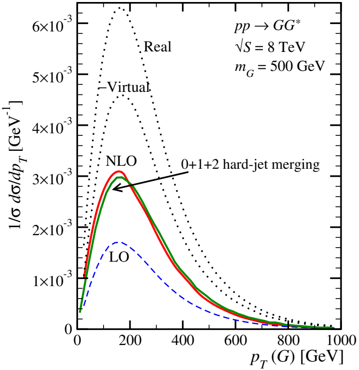
♠ Structure of the NLO QCD corrections for $pp \to GG^*$

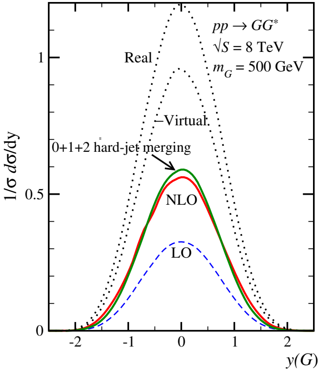


Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1203.6358 [hep-ph]

Distributions: Fixed-order versus multi-jet merging

♠ (Parton-level) distributions at NLO for $pp \to GG^*$
– compared to multi-jet merging via MLM





Gonçalves-Netto, DLV, Mawatari, Plehn, Wigmore, arXiv:1203.6358 [hep-ph]

Outline

Take-home ideas

MadGolem

Take-home ideas

MadGolem

AUTOMATION **NLO** **NEW PHYSICS**

♠ completely **AUTOMATES** the calculation of NLO QCD corrections for generic BSM $2 \to 2$ processes and their interface to Monte Carlo event generators

Take-home ideas

MadGolem

AUTOMATION **NLO** **NEW PHYSICS**

♠ completely **AUTOMATES** the calculation of NLO QCD corrections for generic BSM $2 \to 2$ processes and their interface to Monte Carlo event generators

Feynman diagrammatic / fully analytical approach

Take-home ideas

MadGolem

AUTOMATION **NLO** **NEW PHYSICS**

♠ completely **AUTOMATES** the calculation of NLO QCD corrections for generic BSM $2 \to 2$ processes and their interface to Monte Carlo event generators

Feynman diagrammatic / fully analytical approach

♠ Provides a **3-COMMAND PATHWAY** from BSM to (parton-level) event rates and distributions at NLO.

Take-home ideas

MadGolem

AUTOMATION **NLO** **NEW PHYSICS**

♠ completely **AUTOMATES** the calculation of NLO QCD corrections for generic BSM $2 \to 2$ processes and their interface to Monte Carlo event generators

Feynman diagrammatic / fully analytical approach

♠ Provides a **3-COMMAND PATHWAY** from BSM to (parton-level) event rates and distributions at NLO.

- **Generating** the amplitude: `./newprocess_nlo`
- **Processing** the amplitude `perl run_golem.pl`
- **Evaluating** the cross-section: `./bin/generate_events_nlo 2 2 myprocess`

Closing remarks

MadGolem highlights

- Fully analytical procedure
- BSM-suitable loop calculator
- Broad coverage of spin & color structures
- Automated OS Subtraction
- Complete support of NLO QCD calculations for the SM, the MSSM and several other extensions.
  - **SUSY dipoles** (with α dependence)
  - **UV counterterms** & **SUSY restoration**
- Automatic generation of (parton-level) distributions

Closing remarks

MadGolem highlights

- Fully analytical procedure
- BSM-suitable loop calculator
- Broad coverage of spin & color structures
- Automated OS Subtraction
- Complete support of NLO QCD calculations for the SM, the MSSM and several other extensions.
  - **SUSY dipoles** (with α dependence)
  - **UV counterterms** & **SUSY restoration**
- Automatic generation of (parton-level) distributions

Closing remarks

MadGolem highlights

- Fully analytical procedure
- BSM-suitable loop calculator
- Broad coverage of spin & color structures
- Automated OS Subtraction
- Complete support of NLO QCD calculations for the SM, the MSSM and several other extensions.
  - **SUSY dipoles** (with α dependence)
  - **UV counterterms** & **SUSY restoration**
- Automatic generation of (parton-level) distributions

Closing remarks

MadGolem highlights

- Fully analytical procedure
- BSM-suitable loop calculator
- Broad coverage of spin & color structures
- Automated OS Subtraction
- Complete support of NLO QCD calculations for the SM, the MSSM and several other extensions.
  - **SUSY dipoles** (with α dependence)
  - **UV counterterms** & **SUSY restoration**
- Automatic generation of (parton-level) distributions

Closing remarks

MadGolem highlights

- Fully analytical procedure
- BSM-suitable loop calculator
- Broad coverage of spin & color structures
- Automated OS Subtraction
- Complete support of NLO QCD calculations for the SM, the MSSM and several other extensions.
  - **SUSY dipoles** (with α dependence)
  - **UV counterterms** & **SUSY restoration**
- Automatic generation of (parton-level) distributions

Closing remarks

MadGolem highlights

- Fully analytical procedure
- BSM-suitable loop calculator
- Broad coverage of spin & color structures
- Automated OS Subtraction
- Complete support of NLO QCD calculations for the SM, the MSSM and several other extensions.
  - **SUSY dipoles** (with α dependence)
  - **UV counterterms** & **SUSY restoration**
- Automatic generation of (parton-level) distributions

Closing remarks

Future directions

- Complete the full automation of $2 \to 2$ @ MSSM
- Complete the cross-checking program
- Extend the supported BSM models
- Prepare the code for its public release (structure, interfaces, robustness) – $\mathcal{O}(\text{few months})$

Closing remarks

Future directions

- Complete the full automation of $2 \to 2$ @ MSSM
- Complete the cross-checking program
- Extend the supported BSM models
- Prepare the code for its public release (structure, interfaces, robustness) – $\mathcal{O}(\text{few months})$

Closing remarks

Future directions

- Complete the full automation of $2 \to 2$ @ MSSM
- Complete the cross-checking program
- Extend the supported BSM models
- Prepare the code for its public release (structure, interfaces, robustness) – $\mathcal{O}(\text{few months})$

Closing remarks

Future directions

- Complete the full automation of $2 \to 2$ @ MSSM
- Complete the cross-checking program
- Extend the supported BSM models
- Prepare the code for its public release (structure, interfaces, robustness) – $\mathcal{O}(\text{few months})$

Closing remarks

Future directions

- Complete the full automation of $2 \to 2$ @ MSSM
- Complete the cross-checking program
- Extend the supported BSM models
- Prepare the code for its public release (structure, interfaces, robustness) – $\mathcal{O}(\text{few months})$

Closing remarks

Future directions

- Complete the full automation of $2 \to 2$ @ MSSM
- Complete the cross-checking program
- Extend the supported BSM models
- Prepare the code for its public release (structure, interfaces, robustness) – $\mathcal{O}(\text{few months})$

Engaging collaboration: lines of common interest

♠ **MadGolem & MadGraph5**: [Alwall et al.]:
recode the extended MadDipole module of MadGolem into MadGraph5

♠ **MadGolem & MadDM**: [Alwall, Backovic, Kong, McCasey]:
use of the MadGolem loop calculator for the analysis of loop-induced DM decay processes $\to$ completely unmatched feature to e.g. MicroMEGAS.

♠ **MadGolem & FeynRules**: [Christensen, Duhr, Fuks]:
Automating the generation of UV counterterms for NLO $\mathcal{O}(\alpha_s)$ corrections to generic New Physics models.

♠ **MadGolem & GoSam**: [Cullen, Heinrich, et al.]:
Dealing with higher dimensional operators

Farewell

MadGolem contributes to extending bridges

Farewell

MadGolem contributes to extending bridges

Theory Experiment

Farewell

MadGolem contributes to extending bridges

Theory THANKS A LOT !! Experiment