SLAC

New developments in MadGraph/MadEvent 4

Johan Alwall (SLAC)

Fabio Maltoni (UCL)
Simon de Visscher (UCL)
Michel Herquet (UCL)
Tilman Plehn (Edinburgh)
Tim Stelzer (UIUC)
Rikkert Frederix (UCL)
Claude Duhr (UCL)
David Rainwater (Rochester)

JHEP 0709:028,2007 (arXiv:0706.2334)

ALCPG 2007, Fermilab, Oct 22-26, 2007

SLAC

- Question:

  How do I easily and efficiently generate events for (almost) any process, in (almost) any model for any collider?

- Answer:

  Use MadGraph/MadEvent 4!

SLAC

MadGraph: What is it ?

- By T. Stelzer and W.F. Long [Phys. Com. 81 (1994) 357-371]
- Given a process, produces tree-level Feynman diagrams and a Fortran subroutine that computes the squared amplitudes using the HELAS helicity amplitude library
  Examples:
  - pp > W+W-jjj, QCD = 3, QED = 2 (model: sm)
  - e+e- > x1+x1- a a, QED = 4 (model: mssm)
- Reads the files particles.dat and interactions.dat for particle content and interaction vertices of the model
- Handles processes with up to 8 final states particles (and up to 10000 diagrams)

SLAC

MadEvent: What is it ?

- By F. Maltoni and T. Stelzer [JHEP 0302:027, 2003]
- Multi-purpose event generator
- Uses the process information (matrix elements and phase space mappings) produced by MadGraph
- Efficient and general phase-space integration using Single-Diagram-Enhanced multichannel integration:
  - ➔ Uses the squared diagrams as basis for multi-channel integration

$$f_i = \frac{|A_{\rm tot}|^2}{\sum_i |A_i|^2} |A_i|^2$$

  - ➔ Interference terms cannot introduce new poles
- Trivially parallelizable – makes cluster use efficient

SLAC

What's new in MG/ME 4?

- Web-oriented, modular software structure
- Three dedicated clusters (US, Belgium and Italy)
- New models
  - SUSY, 2HDM, Higgs EFT and (soon) UED
  - Framework for easy user model implementation
- Multiple/inclusive processes in single run
- Pythia (hadronization) and PGS (detector sim.) packages for complete event simulation on-line
- Matching with Pythia parton showers
- Beam polarization and (soon) beam strahlung (for ILC)

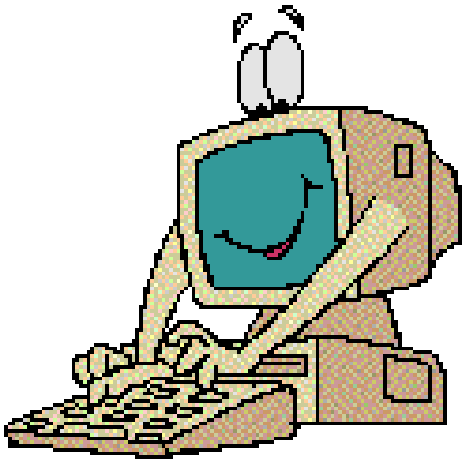
SLAC

How do I use MG/ME 4?

1. Open your browser
2. Go to one of our sites
3. Create a process
4. Generate events

Sounds easy? It is!

Let me show you!

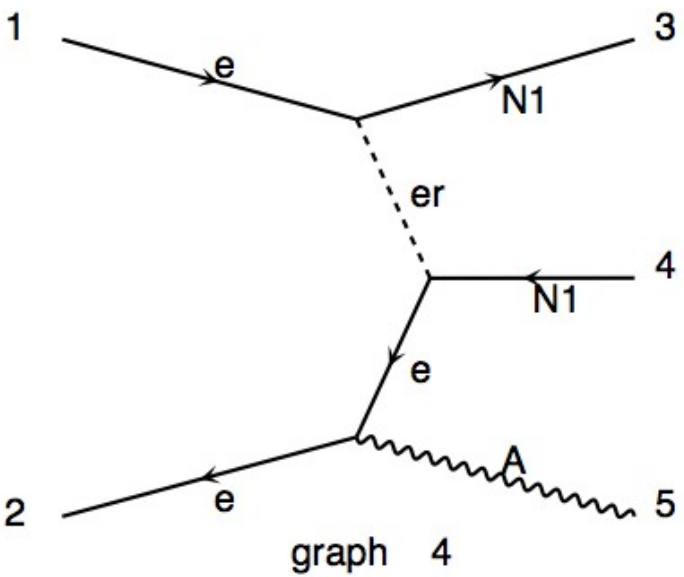


SLAC

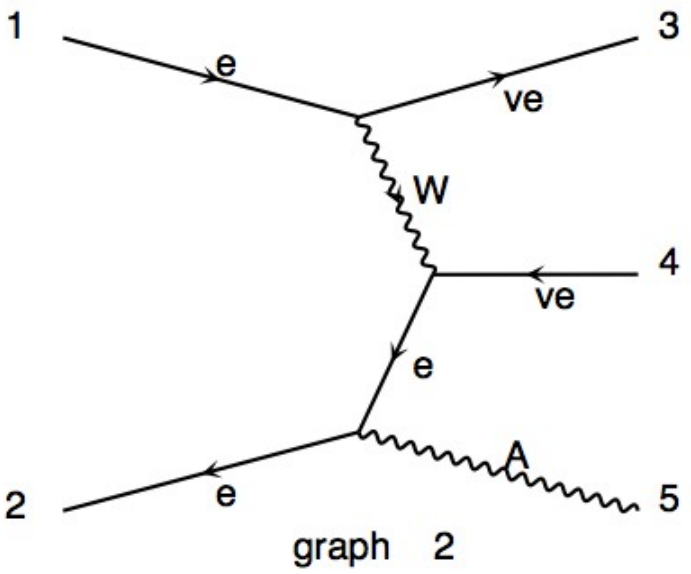
Example: 20-minute MG project

Effects of polarization on detection of dark matter production at the ILC through radiative photons (see Dreiner, Kittel, Langenfeld arXiv:0707.1642)

- Signal process: e+e- > $\chi_1^0\ \chi_1^0\ \gamma$
- Main background: e+e- > $\nu_l\ \nu_l\ \gamma$



graph 4



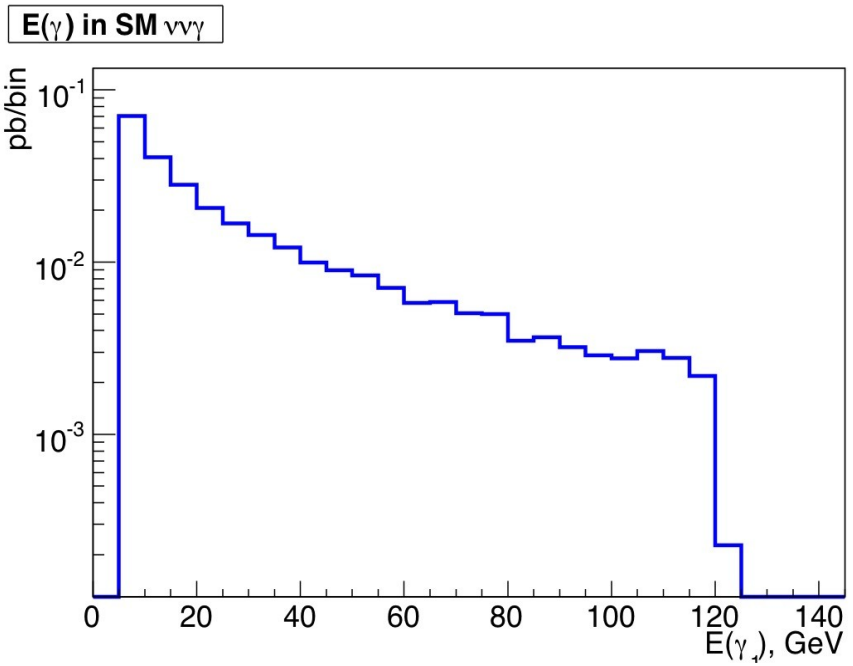
graph 2

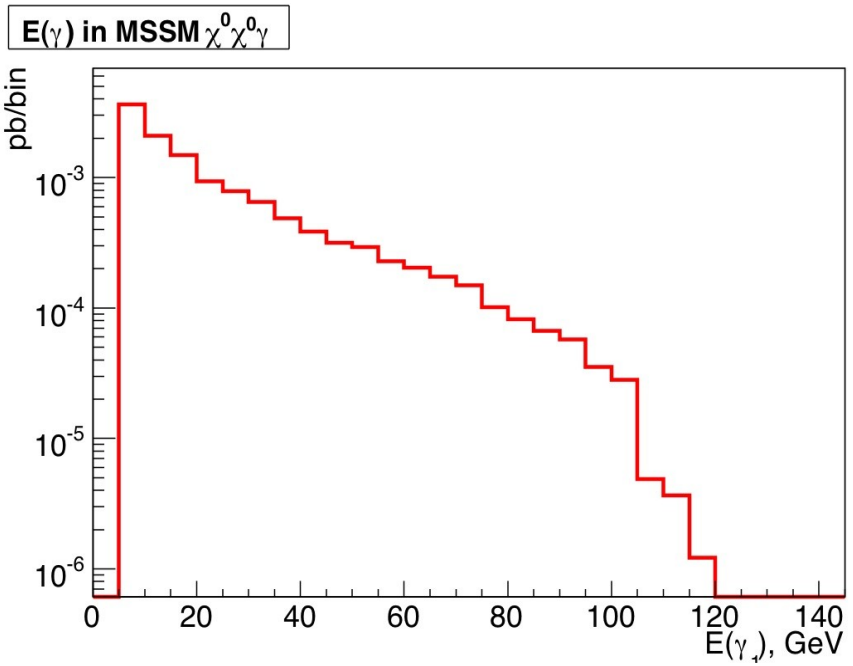
SLAC

Cross sections + plots

- MSSM point: M(χ_1^0) = 180 GeV, M(others) > 250 GeV
- Only lightest neutralino attainable at 500 GeV LC

Polarization (e- / e+)	0/0	80/0	80/-30	80/-60
$\sigma(\chi_1^0\chi_1^0\gamma)$ (fb)	4.4	7.7	9.9	12
$\sigma(\nu_l\,\nu_l\,\gamma)$ (fb)	3300	670	480	280





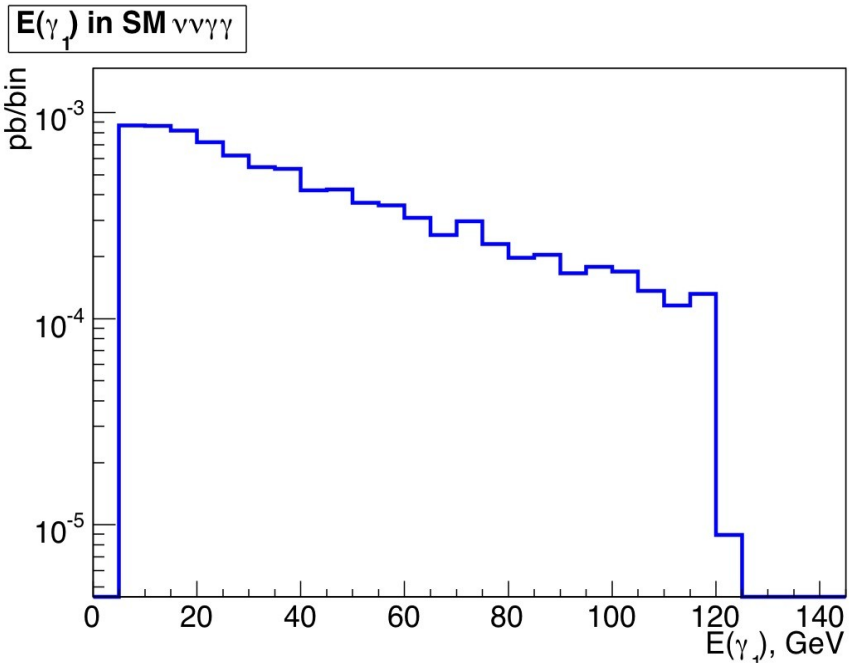
SLAC

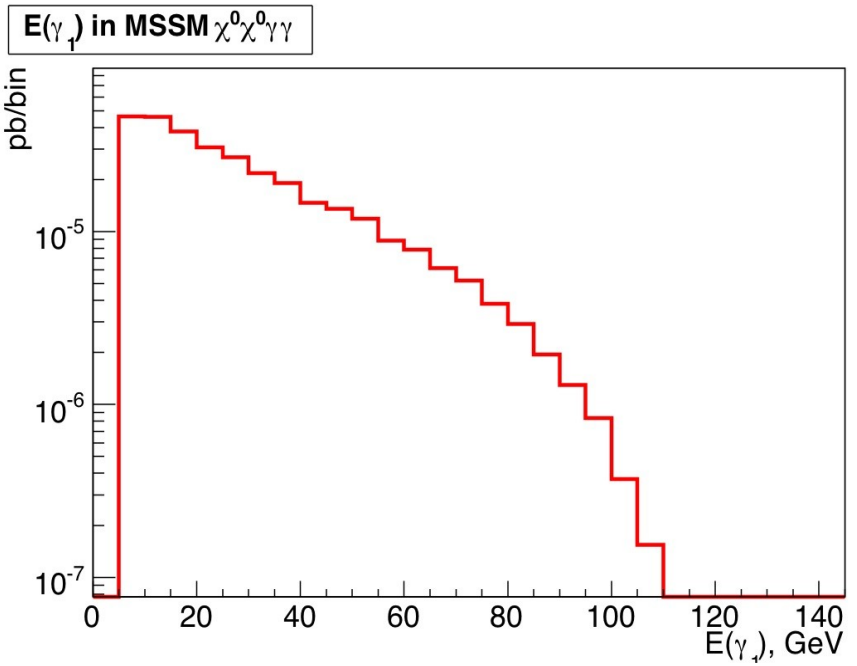
Easier to distinguish with 2γ's?

Try with e+e- > $\chi_1^0 \chi_1^0 \gamma\gamma$ to see if larger difference

Cuts: $\theta(\gamma_1,\gamma_2) > 1°$, 5 GeV < $E(\gamma_{1,2}) < E_{beam}$-$M(\chi_1^0)$

Polarization (e- / e+)	0/0	80/0	80/-30	80/-60
$\sigma(\chi_1^0\chi_1^0\gamma\gamma)$ (fb)	0.11	0.19	0.25	0.31
$\sigma(\nu_l\,\nu_l\,\gamma\gamma)$ (fb)	110	23	17	10.2





SLAC

Ongoing developments

- FeynRules – Mathematica based program to extract Feynman rules + MG files from Lagrangean (Duhr)
- Specification of complete decay chains – allows for high-multiplicity final states with full spin correlations (JA, Stelzer)

Under testing

- Matrix element analysis techniques for arbitrary processes (Artoisenet, Mattelaer)
- Grid-executable MadEvent (Stelzer)
- More tools for e+e-: Energy scans, beam strahlung, jet matching with Pythia parton showers (JA)

SLAC

Summary

- MadGraph/MadEvent 4 – an integrated tool to generate any process (signal or background)
- User friendly: Reduces overhead and errors, allows you to focus on physics!
- Run on the Web or download (parts or whole code)
- Models: SM, MSSM, 2HDM, HEFT, (soon) UED
- Easy to implement new models
- Fast – thanks to efficient and cluster-oriented generation
- Clusters found at:
  - UIUC: http://madgraph.hep.uiuc.edu/
  - UCL: http://madgraph.phys.ucl.ac.be/
  - Rome: http://madgraph.roma2.infn.it/

> 1500 registered users

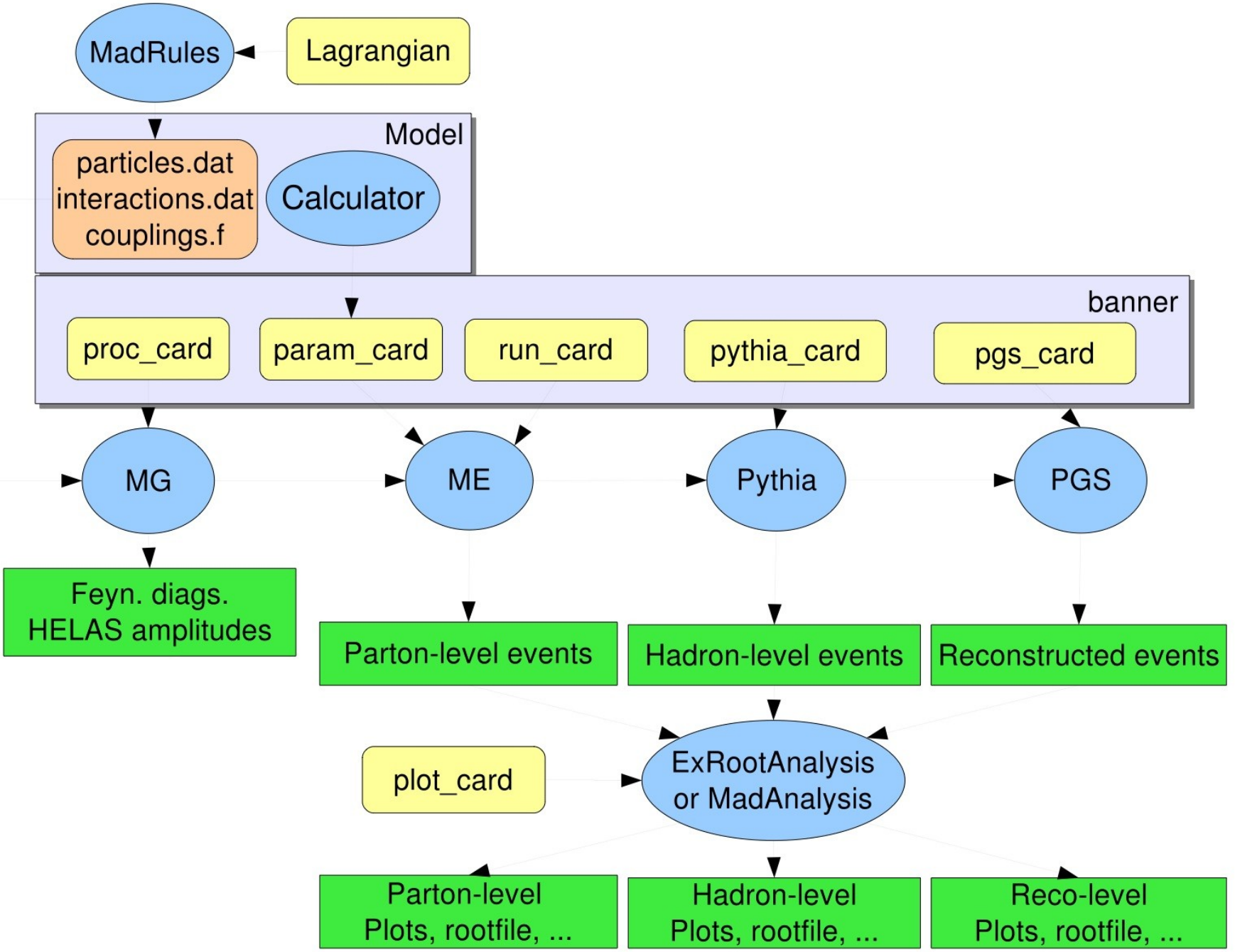
Try it out – we are grateful for your feedback!

SLAC

Backup slides

SLAC

MG/ME structure

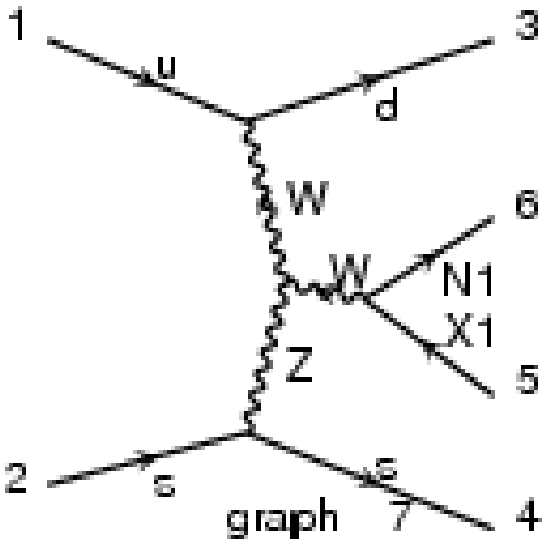


SLAC

MSSM

Hagiwara, Plehn, Rainwater, Stelzer + Alwall

- CP and R-parity conserving MSSM
- Sfermion mixing and Yukawa couplings for 3rd gen.
- Uses SUSY Les Houches input files – independent of SUSY breaking scheme
- Detailed comparison of cross sections between SMadGraph, Omega and Amegic++ (hep-ph/0512260)
- Input files for the 10 SPS points available

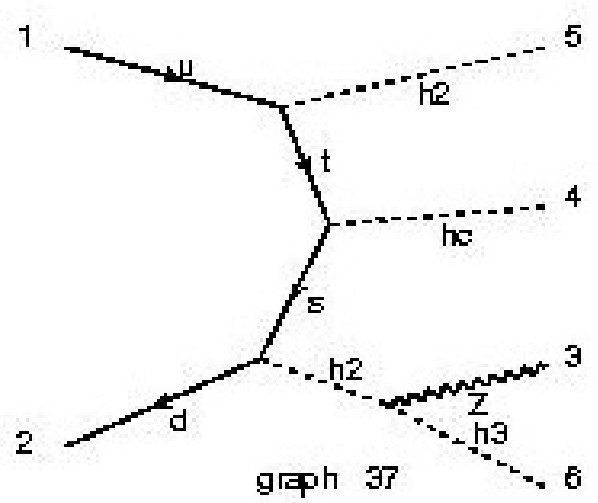


SLAC

General 2HDM

de Visscher, Herquet

- Completely general 2HDM, with FCNC and CP violation
- New tree-level calculator (Herquet) with a web interface, TwoHiggsCalc, to generate the param_card for MadEvent
- Generic basis or Higgs basis, intensive use of recent basis invariance techniques (e.g. hep-ph/0504050)
- Tested in the SM & MSSM limits
- Sample files for various cases
- Simplified version without FCNC and off-diag. CKM elements

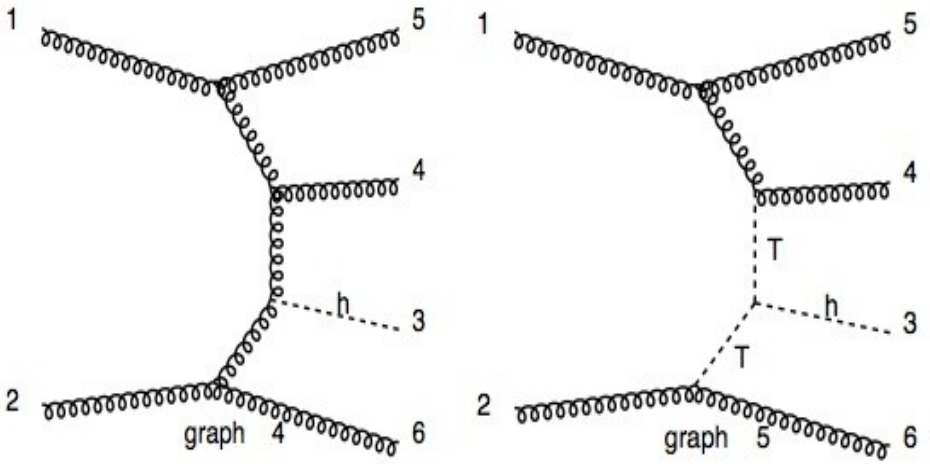


SLAC

Higgs EFT

Frederix

- Effective couplings of Higgs to gluons
  - Effective non-propagating tensor particle to allow Higgs couplings to more than 3 gluons
  - Several new HELAS subroutines
  - Works for scalar and pseudo-scalar neutral Higgs bosons



SLAC

Implementing a new model

Ways to implement your own model in MG/ME:

- Modify existing model (e.g. changing couplings)
- User model framework (e.g. subspace of larger models)
  - New particles
  - New interactions
  - Expressions for the new couplings
  - Perl script generates all files needed by MadEvent!
- FeynRules
  - Directly from Lagrangean to implementation

Work in progress

SLAC

User model generation

de Vissher

particles.dat

```
#Name anti_Name  Spin    Linetype Mass Width Color Label Model
#xxx   xxxx      SFV      WSDC     str  str  STO    str  PDG code
#MODEL EXTENSION
tp       tp~      F        S      TPMASS TPWID  T   TP    8
zp       zp       V        W      ZPMASS ZPWID  S   ZP   32
# END
```

interactions.dat

```
#   USRVertex
tp tp g  GG    QCD
tp t  zp GTPZP QED
t  tp zp GTPZP QED
```

couplings.f

```
c************************************
c UserMode couplings
c************************************

      GTPZP(1)=dcmplx(ee*param1,Zero)
      GTPZP(2)=dcmplx(ee*param1,Zero)
```