A Flavor Kit for BSM models

Florian Staub

BCTP Bonn

in collaboration with

W. Porod, A. Vicente

Helmholtz-Alliance Linear Collider Forum

Bonn, 30. April 2014

Introduction

- New physics will not only show up in cross sections
  $\rightarrow$ interplay between different searches for new physics
- BSM gives often new contributions to flavor observables

universitätbonn bctp

Introduction

- New physics will not only show up in cross sections
  → interplay between different searches for new physics
- BSM gives often new contributions to flavor observables
- Several public codes available to calculate flavor observables
  - `MicrOmegas` [Belanger, Boudjema, Pukhov, Semenov]
  - `NMSSM-Tools` [Ellwanger, Hugonie]
  - `SPheno` [Porod, FS]
  - `superiso` [Mahmoudi]
  - `SuseFlav` [Chowdhury, Garani, Vempati]
  - `SUSY_Flavor` [Rosiek, Chankowski, Dedes, Jager, Tanedo]
  - ...

Introduction

- New physics will not only show up in cross sections
  $\rightarrow$ interplay between different searches for new physics
- BSM gives often new contributions to flavor observables
- Several public codes available to calculate flavor observables
  - MicrOmegas [Belanger, Boudjema, Pukhov, Semenov]
  - NMSSM-Tools [Ellwanger, Hugonie]
  - SPheno [Porod, FS]
  - superiso [Mahmoudi]
  - SuseFlav [Chowdhury, Garani, Vempati]
  - SUSY_Flavor [Rosiek, Chankowski, Dedes, Jager, Tanedo]
  - ...

Restrictions

- Work only for specific models
- Hardly possible for user to extent list of calculated observables

Calculation of Flavor observables in a nutshell

To calculate flavor observables in a given model one needs

1. Expressions for vertices and masses
2. Expressions for Wilson coefficients
3. Expressions for observables
4. Numerical values for everything

Calculation of Flavor observables in a nutshell

To calculate flavor observables in a given model one needs

1. Expressions for vertices and masses → SARAH
2. Expressions for Wilson coefficients → FeynArts/FormCalc
3. Expressions for observables → literature
4. Numerical values for everything → SPheno

Calculation of Flavor observables in a nutshell

To calculate flavor observables in a given model one needs

1. Expressions for vertices and masses $\rightarrow$ SARAH
2. Expressions for Wilson coefficients $\rightarrow$ FeynArts/FormCalc
3. Expressions for observables $\rightarrow$ literature
4. Numerical values for everything $\rightarrow$ SPheno

Let's combine the different tools!

SARAH and SPheno

'Spectrum Generator Generator'

SARAH writes Fortran source-code using the obtained information about vertices, masses, RGEs in a given model. This provides new SPheno modules which calculate

- Two-loop RGEs
- One-loop corrected mass spectrum
- Decay widths and branching ratios
- Flavor observables (more in a second)

in a wide range of BSM models.

→ Implementation of new models in SPheno in a modular way without the need to write any line of source code by hand.

FlavorKit

- So far SARAH had already included some flavor observables in the SPheno output
- The necessary expressions for the amplitudes have been hard-coded in SARAH

$\rightarrow$ It has been difficult and time consuming to implement new observables

FlavorKit

- So far SARAH had already included some flavor observables in the SPheno output
- The necessary expressions for the amplitudes have been hard-coded in SARAH

$\rightarrow$ It has been difficult and time consuming to implement new observables

FlavorKit

The calculation of flavor observables is now based on external files which ...

- ... provide the generic expressions of the Wilson coefficients
- ... the formulae to combine coefficients to observables

Both can be extended by the user.

New observables

To calculate new observables the user has to provide two files

- A steering file:
  defines the necessary operators and the position in the SPheno spectrum file

New observables

To calculate new observables the user has to provide two files

- A steering file:
  defines the necessary operators and the position in the `SPheno` spectrum file
- A Fortran file:
  gives Fortran code to combine operators to observables

New observables

To calculate new observables the user has to provide two files

- A steering file:
  defines the necessary operators and the position in the SPheno spectrum file
- A Fortran file:
  gives Fortran code to combine operators to observables

Both files have to be put into the FlavorKit subdirectory of SARAH

$\rightarrow$ The observables are included automatically in the SPheno output.

Example $l \to l_j\gamma$: Steering file

The Steering file reads

```
NameProcess = "LLpGamma";

NameObservables = {{muEgamma, 701, "BR(mu->e gamma)"},
                   {tauEgamma, 702, "BR(tau->e gamma)"},
                   {tauMuGamma, 703, "BR(tau->mu gamma)"}};

NeededOperators = {K2L, K2R};

Body = "LLpGamma.f90";
```

K2L, K2R are the coefficients of the dipole operator

$$\mathcal{L}_{\ell\ell\gamma} = e\,\bar{\ell}_\beta \left[i m_{\ell_\alpha} \sigma^{\mu\nu} q_\nu \left(K_2^L P_L + K_2^R P_R \right) \right] \ell_\alpha A_\mu + h.c.$$

Example $l \to l_j\gamma$: Fortran file

```
Real(dp) :: width
Integer :: i1, gt1, gt2

Do i1=1,3
If (i1.eq.1) Then           ! mu -> e gamma
 gt1 = 2
 gt2 = 1
Elseif (i1.eq.2) Then       !tau -> e gamma
 ...
End if

width=0.25_dp*mf_l(gt1)**5*(Abs(K2L(gt1,gt2))**2 &
          & +Abs(K2R(gt1,gt2))**2)*Alpha

If (i1.eq.1) Then
  muEgamma = width/(width+GammaMu)
Elseif (i1.eq.2) Then
 ...
End if
End do
```

Example $l \to l_j\gamma$: Result

After running SARAH and compiling the SPheno module the spectrum files produced by SPheno include the new observable:

```
# SUSY Les Houches Accord 2 - NMSSM
# SPheno module generated by SARAH
...
Block FlavorKitLFV # lepton flavor violating observables
     701     1.61451131E-14  # BR(mu->e gamma)
     702     5.67628390E-16  # BR(tau->e gamma)
     703     2.15514014E-17  # BR(tau->mu gamma)
...
```

New operators

The generic expressions for the coefficients of new operators can be calculated with an additional package (PreSARAH):

- Easy way to define operators and color flow
- Uses FeynArts/FormCalc to calculate expressions
- Writes all necessary files for SARAH

New operators

The generic expressions for the coefficients of new operators can be calculated with an additional package (PreSARAH):

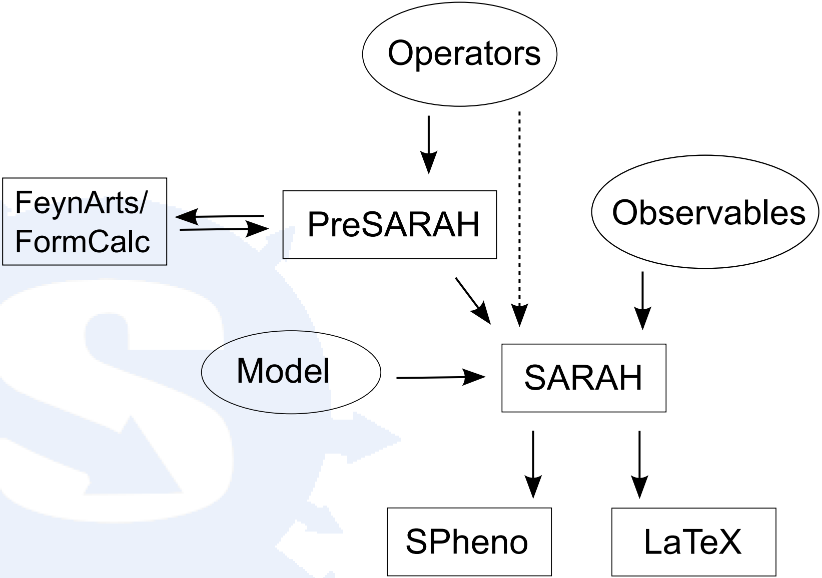
- Easy way to define operators and color flow
- Uses FeynArts/FormCalc to calculate expressions
- Writes all necessary files for SARAH

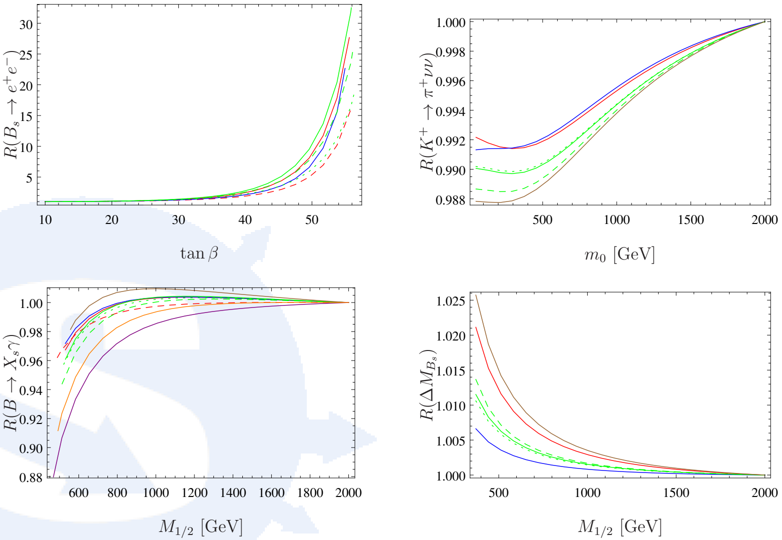
```
NameProcess="2d2L";

ConsideredProcess = "4Fermion";
FermionOrderExternal={2,1,4,3};
NeglectMasses={1,2,3,4};

ExternalFields= {DownQuark,bar[DownQuark],
                                   ChargedLepton,bar[ChargedLepton]};

AllOperators={{OddllSLL,Op[7].Op[7]}, (* [d PL d][l PL l] *)
               {OddllSRL,Op[6].Op[7]}, (* [d PR d][l PL l] *)
...
};
```


Operators
FeynArts/
FormCalc
PreSARAH
Observables
Model
SARAH
SPheno
LaTeX



FlavorKit, SPhenoMSSM (dashed), SPheno 3.3 SUSY_Flavor 1, SUSY_Flavor 2, MicrOmegas, SuperIso

Conclusion

- The `FlavorKit` allows the user to implement easily new operators/observables in `SARAH`
- `SARAH` creates source code for `SPheno` to calculate the observables for a given model
- We made use of this to (re-) implement in `SARAH`
  - Br($l_i \to l_j \gamma$), Br($l \to 3l'$), Br($Z \to ll'$)
  - CR($\mu - e, A$), Br($\tau \to l + P$)
  - Br($B \to X_s \gamma$), Br($B^0_{s,d} \to l\bar{l}$), Br($B \to sl\bar{l}$), Br($K \to \mu\nu$)
  - Br($B \to q\nu\nu$), Br($K^+ \to \pi^+ \nu\nu$), Br($K_L \to \pi^0 \nu\nu$)
  - $\Delta M_{B_s,B_d}$, ΔM_K, ϵ_K, Br($B \to K\mu\bar{\mu}$)
  - Br($B \to l\nu$), Br($D_s \to l\nu$)
- `FlavorKit` will be included in `SARAH` 4.2.0 (Beta-Version available)

(`http://sarah.hepforge.org/`)