FlexibleSUSY: A spectrum generator generator for SUSY and non-SUSY models

P. Athron, M. Bach, D. Harries, T. Kwasnitza, J.-h. Park, T. Steudtner, D. Stöckinger, J. Ziebell, A. Voigt

https://flexiblesusy.hepforge.org/

Dartmouth-TRIUMF HEP Tools Bootcamp
27.10.2017

RWTH AACHEN UNIVERSITY

Contents

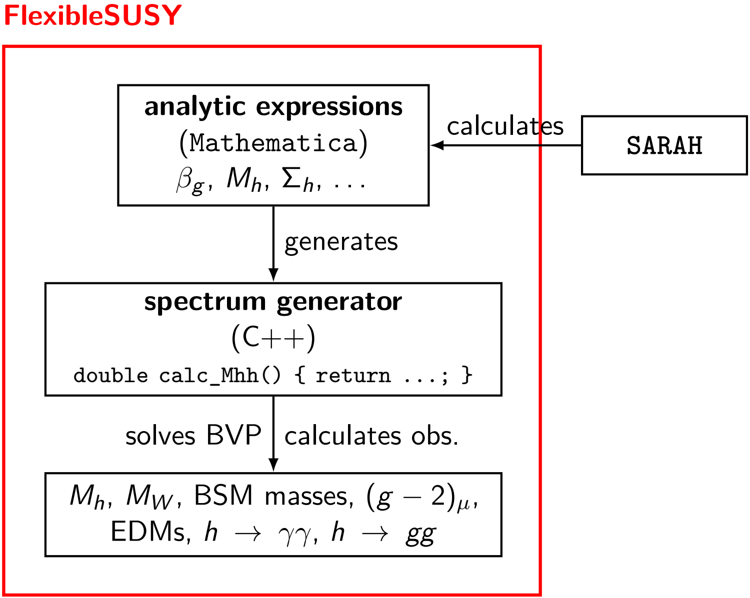
1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
   - Build the spectrum generator
   - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
   - Calculating $(g-2)_\mu$
   - NLL resummation of M_h (FlexibleEFTHiggs)

Contents

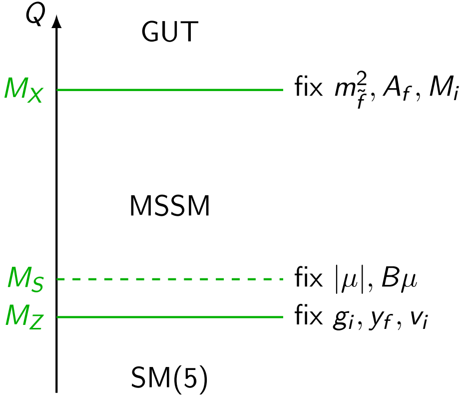
1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
   - Build the spectrum generator
   - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
   - Calculating $(g-2)_\mu$
   - NLL resummation of M_h (FlexibleEFTHiggs)

What is FlexibleSUSY?

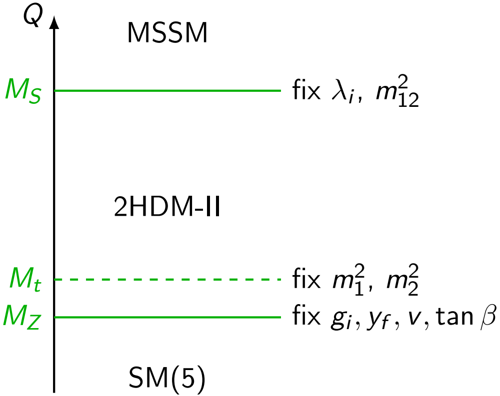
What is FlexibleSUSY?



Example: BVP in the CMSSM



Example: BVP in the MSSM with light Higgs sector



Contents

1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
    - Build the spectrum generator
    - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
    - Calculating $(g-2)_\mu$
    - NLL resummation of M_h (FlexibleEFTHiggs)

Features for all models (SUSY and non-SUSY)

Observables

M_h, M_W, BSM masses,
$(g-2)_\mu$, EDMs,
$h \to \gamma\gamma$, $h \to gg$

Flexibility

multipe BVP solvers,
user-defined BCs,
modular C++ code,
SLHA input/output,
`Mathematica` input/output,
SQLite output

High precision

2L RGEs + 1L self energies (via `SARAH`) + 1L thresholds,
NLL resummation for M_h (`FlexibleEFTHiggs`)

High speed

multi-threading,
smart linear algebra,
lazy evaluation

Additional model-specific precision corrections

MSSM

3L RGEs, 3L M_h, 2L $M_{H,A,H^\pm}$, 2L $(g-2)_\mu$, 2L thresholds

NMSSM

2L M_h, 2L $M_{H,A,H^\pm}$, 2L thresholds

Split-MSSM

3L M_h, 2L thresholds for y_t, 2L thresholds for λ, $\tilde{g}_{ip}$ the MSSM

SM

3L RGEs, 3L M_h, 3L thresholds for α_s, $y_{t,b}$, 2L thresholds for λ to the MSSM (`HSSUSY`)

THDM-II

2L thresholds for λ_i to the MSSM

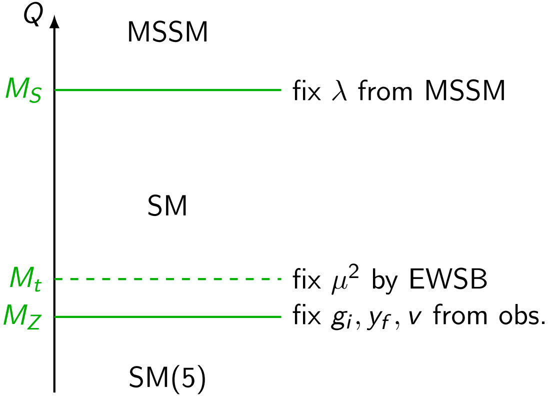
THDM-II $+ \tilde{h} + \tilde{g}$

1L thresholds for λ_i to the MSSM

Contents

1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
   - Build the spectrum generator
   - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
   - Calculating $(g-2)_\mu$
   - NLL resummation of M_h (FlexibleEFTHiggs)

Hands-on example: High-scale MSSM (HSSUSY)



FlexibleSUSY's HSSUSY model provides the most precise Higgs mass calculation available in the MSSM for $M_S \gg M_t$.

Step 0: Chose a SARAH model

We chose SARAH's SM model with the parameters:

```
g1, g2, g3, Yu, Yd, Ye, \[Lambda], mu2, v
```

and the non-interaction Lagrangian

```
LagNoHC = mu2 conj[H].H -
          1/2 \[Lambda] conj[H].H.conj[H].H;
LagHC   = -(Yd conj[H].d.q + Ye conj[H].e.l
            + Yu u.q.H);
```

See Florian's talk from Monday, 23th October 2017 9am

Step 1: Create FlexibleSUSY model file

```
$ cd FlexibleSUSY/
$ mkdir model_files/HSSUSY/
$ touch model_files/HSSUSY/FlexibleSUSY.m.in
```

Step 2: Define model information

Put into the model file model_files/HSSUSY/FlexibleSUSY.m.in:

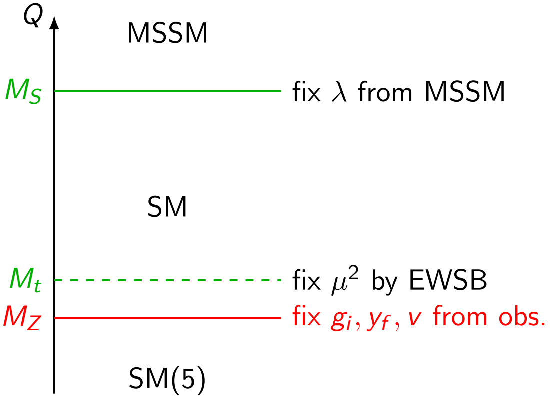
```
FSModelName = "HSSUSY";
FSEigenstates = SARAH`EWSB;
FSDefaultSARAHModel = SM;
```

Step 3: Define MSSM input parameters

```
EXTPAR = {
    {0, MS},
    {1, M1Input},
    {2, M2Input},
    {3, M3Input},
    {4, MuInput},
    {5, mAInput},
    {6, MEWSB},
    {7, AtInput},
    {25, TanBeta}
};

FSAuxiliaryParameterInfo = {
    {msq2, { LesHouches -> MSQ2IN,
             ParameterDimensions -> {3,3},
             InputParameter -> True } },
    ...
};
```

Step 4: Define low-scale boundary condition



Step 4: Define low-scale boundary condition

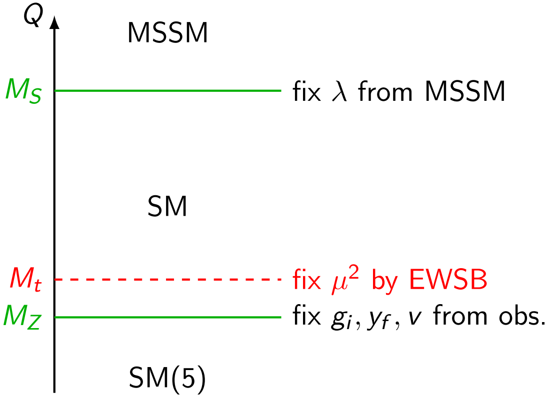
Fix SM parameters $v^{\overline{\text{MS}}}$, $y_f^{\overline{\text{MS}}}$ and $g_i^{\overline{\text{MS}}}$ at the scale $Q = M_Z$:

```
LowScale = LowEnergyConstant[MZ];

LowScaleInput = {
   {v, 2 MZMSbar / Sqrt[3/5 g1^2 + g2^2]},
   {Yu, Automatic},
   {Yd, Automatic},
   {Ye, Automatic}
   (* gauge couplings are fixed automatically *)
};
```

`MZMSbar` $= m_Z^{\overline{\text{MS}}}(Q)$

Step 5: Define EWSB-scale boundary condition



Step 5: Define EWSB-scale boundary condition

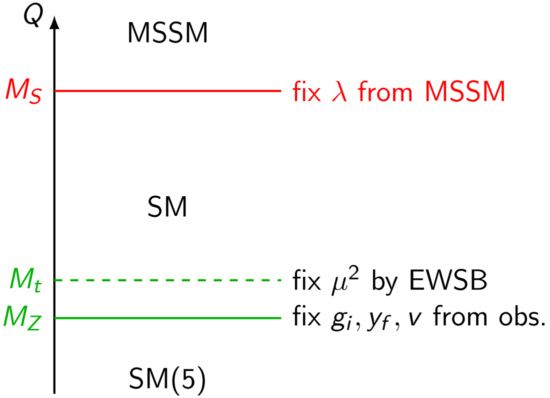
Fix μ^2 by EWSB condition at the scale $Q = \texttt{MEWSB}$:

```
EWSBOutputParameters = { mu2 };

SUSYScale = MEWSB;

SUSYScaleInput = {
   FSSolveEWSBFor[EWSBOutputParameters]
};
```

Step 6: Define high-scale boundary condition



Step 6: Define high-scale boundary condition

Fix λ by matching to the MSSM at scale $Q = \texttt{MS}$ at tree-level + leading 1-loop [0705.1496]:

$$\lambda(Q) = \frac{1}{4}\left[\frac{3}{5}g_1^2 + g_2^2\right]\cos^2 2\beta + \frac{3}{8}\frac{y_t^4}{\pi^2}\left[\frac{X_t^2}{M_S^2} - \frac{X_t^4}{12M_S^4}\right]$$

with $X_t = A_t - \mu/\tan\beta$.

```
HighScale = MS;

HighScaleInput = {
   {\[Lambda], lambdaTree + lambda1L}
};

lambdaTree = (3/5 g1^2 + g2^2)/4 Cos[2 ArcTan[TanBeta]]^2;

lambda1L = With[{yt = Yu[3,3],
                 Xt = AtInput - MuInput/TanBeta},
   3/8 yt^4/Pi^2 ( Xt^2/MS^2 - Xt^4/(12 MS^4) )
];
```

Step 7: Enable SM-specific higher order corrections

Higher-order corrections from the literature

[hep-ph/9911434, hep-ph/9912391, hep-ph/0004189, 1407.4336]:

$$m_t = M_t + \Delta^{(1)} m_t + \Delta^{(2)} m_t + \Delta^{(3)} m_t$$

$$\alpha_s = \frac{\alpha_s^{\mathrm{SM}(5)}}{1 + \Delta^{(1)}\alpha_s + \Delta^{(2)}\alpha_s + \Delta^{(3)}\alpha_s}$$

$$M_h^2 = m_h^2 + \Delta^{(1)} m_h^2 + \Delta^{(2)} m_h^2 + \Delta^{(3)} m_h^2$$

```
UseSM3LoopRGEs     = True; (* 3L RGEs *)
UseYukawa3LoopQCD = True; (* 3L corr. to yt *)
UseSMAlphaS3Loop   = True; (* 3L corr. to alpha_s *)
UseHiggs2LoopSM    = True; (* 2L contributions to Mh *)
UseHiggs3LoopSM    = True; (* 3L contributions to Mh *)
```

Step 8: Build the spectrum generator

```
1 $ ./createmodel --name=HSSUSY
2 $ ./configure --with-models=HSSUSY
3 $ make -j4
```

1: creates the directory `models/HSSUSY/` with makefile module

2: searches for required libraries and creates the `Makefile`

3: calls `SARAH`, generates the C++ code and compiles it

Finally: Run HSSUSY

Run HSSUSY using the provided SLHA input file `models/HSSUSY/LesHouches.in.HSSUSY`:

```
$ cd models/HSSUSY/
$ ./run_HSSUSY.x --slha-input-file=LesHouches.in.HSSUSY
Block SPINFO
     1   FlexibleSUSY
     2   2.0.1
     5   HSSUSY
     9   4.12.0
Block EXTPAR
    0   2000                      # MS
    4   2000                      # Mu(MS)
    6   173.34                    # MEWSB
    7   2000                      # At(MS)
   25   10                        # TanBeta(MS)
Block MASS
        24     8.03785042E+01   # VWp
        25     1.18510822E+02   # hh
...
```

Contents

1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
   - Build the spectrum generator
   - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
   - Calculating $(g-2)_\mu$
   - NLL resummation of M_h (FlexibleEFTHiggs)

Parameter scan

We'll use the Mathematica interface of FlexibleSUSY to perform a scan over M_S. (Scanning at the C++ level or via GAMBIT is also possible.)

Open Mathematica and type:

```
Get["models/HSSUSY/HSSUSY_librarylink.m"];
```

Now we can use these functions (and more):

FSHSSUSYOpenHandle[<parameters>]

FSHSSUSYCalculateSpectrum[<handle>]

FSHSSUSYCalculateObservables[<handle>]

FSHSSUSYCloseHandle[<handle>]

...

Parameter scan

```
handle = FSHSSUSYOpenHandle[
    fsSettings -> {
        calculateStandardModelMasses -> 1
    },
    fsModelParameters -> {
        MS      -> 5000,
        MEWSB   -> 173.34,
        TanBeta -> 5,
        MuInput -> 5000,
        AtInput -> 1000
    }
];

spec = FSHSSUSYCalculateSpectrum[handle];

FSHSSUSYCloseHandle[handle];
```

See [1710.03760] for detailed documentation.

Parameter scan

```
In[]:= Print[spec];

Out[]=
{HSSUSY -> {
   M[hh] -> 116.74076603621857,
   M[VWp] -> 80.16049791392693,
   Pole[M[hh]] -> 117.38153372733366,
   Pole[M[VWp]] -> 80.37359140369604,
   ...
   \[Lambda] -> 0.22248507579380888,
   Yu -> {{7.493388340878909*^-6, 0., 0.},
          {0., 0.0034112235116303 61, 0.},
          {0., 0., 0.9384438041061343}},
   ...
   SCALE -> 173.34
 }
}
```

Parameter scan

Wrap everything into a function (TB $= \tan\beta$, Xtt $= X_t/M_S$):

```
CalcMh[MSUSY_, TB_, Xtt_] :=
    Module[{handle, spec},
        handle = FSHSSUSYOpenHandle[
            fsSettings -> { ... },
            fsSMParameters -> { ... },
            fsModelParameters -> {
                MS -> MSUSY, MEWSB -> 173.34,
                TanBeta -> TB, MuInput -> MSUSY,
                AtInput -> (Xtt + 1/TB) MSUSY,
                msq2 -> MSUSY^2 IdentityMatrix[3],
                msu2 -> MSUSY^2 IdentityMatrix[3],
                msd2 -> MSUSY^2 IdentityMatrix[3],
                msl2 -> MSUSY^2 IdentityMatrix[3],
                mse2 -> MSUSY^2 IdentityMatrix[3]
            }
        ];
        spec = FSHSSUSYCalculateSpectrum[handle];
        FSHSSUSYCloseHandle[handle];
        Pole[M[hh]] /. (HSSUSY /. spec)
       ];
```

Parameter scan over X_t/M_S

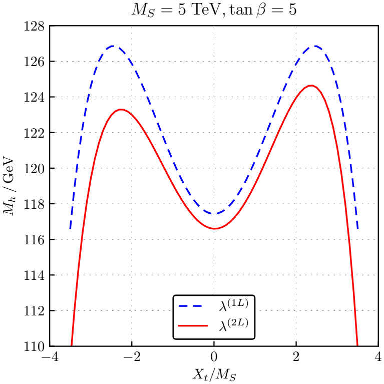
Scan over $X_t/M_S \in [-3.5, 3.5]$ with $M_S = 5\,\text{TeV}$, $\tan\beta = 5$:

```
MS = 5000;
TB = 5;

LaunchKernels[];

data = ParallelMap[
          { N[#], CalcMh[MS, TB, #] }&,
          Range[-3.5, 3.5, 0.1]
       ];
```

Parameter scan over X_t/M_S



FlexibleSUSY's HSSUSY model implements all known 1- and 2-loop corrections from [1407.4081, 1703.08166]. This is the most precise Higgs mass calculation in the MSSM for $M_S \gg M_t$.

Estimate the EFT uncertainty

The pure EFT calculation neglects terms of $O(v^2/M_S^2)$.
Estimate these terms by adding

$$\lambda(Q) = \lambda^{(0L)} + \Delta\lambda^{(1L)} + \delta_{\text{EFT}} \frac{y_t^4}{(4\pi)^2} \frac{v^2}{M_S^2}$$

with $\delta_{\text{EFT}} = 0$ or 1.

```
EXTPAR = {
   ...
   {100, DeltaEFT} (* <-- new input parameter *)
};

HighScaleInput = {
   {\[Lambda], lambdaTree + lambda1L + lambda1LEFT}
};

lambdaTree  = ...;
lambda1L    = ...;
lambda1LEFT = DeltaEFT Yu[3,3]^4/(4 Pi)^2 v^2/MS^2;
```

Estimate the EFT uncertainty

Extend our wrapper function:

```
CalcMh[MSUSY_, TB_, Xtt_, deltaEFT_] :=
    Module[{handle, spec},
        handle = FSHSSUSYOpenHandle[
            fsSettings -> { ... },
            fsSMParameters -> { ... },
            fsModelParameters -> {
                ...
                DeltaEFT -> deltaEFT (* <-- set DeltaEFT *)
            }
        ];
        spec = FSHSSUSYCalculateSpectrum[handle];
        FSHSSUSYCloseHandle[handle];
        Pole[M[hh]] /. (HSSUSY /. spec)
       ];
```

Estimate the EFT uncertainty

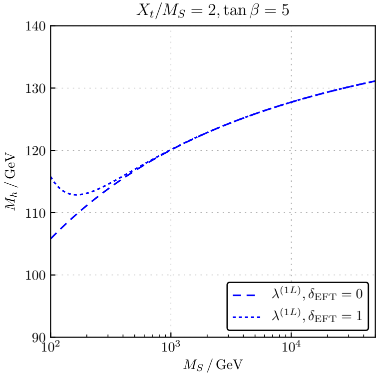
Scan over $M_S \in [100, 5000]$ GeV with $X_t/M_S = 2$, $\tan\beta = 5$:

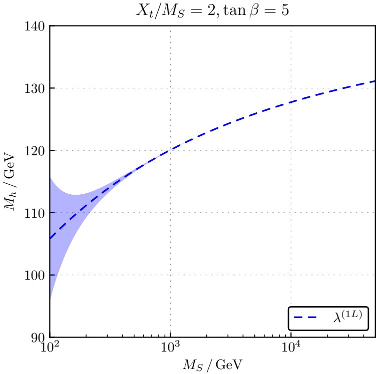
```
Xtt = 2;
TB  = 5;

LaunchKernels[];

data = ParallelMap[
          {
            N[#],
            CalcMh[#, TB, Xtt, 0],
            CalcMh[#, TB, Xtt, 1]
          }&,
          LogRange[100, 5 10^4, 100]
       ];
```

Estimate the EFT uncertainty





Contents

1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
   - Build the spectrum generator
   - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
   - Calculating $(g-2)_\mu$
   - NLL resummation of M_h (FlexibleEFTHiggs)

MRSSM

MRSSM = a minimal supersymmetric model with unbroken continuous R-symmetry.

Fields in the MRSSM on top of the MSSM:

$\hat{R}_d : (\mathbf{1}, \mathbf{2}, -\frac{1}{2}), \hat{R}_u : (\mathbf{1}, \mathbf{2}, \frac{1}{2}), \hat{S} : (\mathbf{1}, \mathbf{1}, 0), \hat{T} : (\mathbf{1}, \mathbf{3}, 0), \hat{O} : (\mathbf{8}, \mathbf{1}, 0)$

Superpotential:

$$\begin{aligned}\mathcal{W}_{\text{MRSSM}} = {} & \mathcal{W}_{\text{MSSM}}(\mu = 0) + \mu_d \hat{R}_d \cdot \hat{H}_d + \mu_u \hat{R}_u \cdot \hat{H}_u \\ & + \lambda_d \hat{S} \hat{R}_d \cdot \hat{H}_d + \lambda_u \hat{S} \hat{R}_u \cdot \hat{H}_u + \Lambda_d \hat{R}_d \cdot \hat{T} \hat{H}_d + \Lambda_u \hat{R}_u \cdot \hat{T} \hat{H}_u .\end{aligned}$$

Use `SARAH`'s `MRSSM` model.

Enable $(g-2)_\mu$ in the MRSSM model file

Add a new output block "`FlexibleSUSYLowEnergy`" and write $(g-2)_\mu$ to entry 21:

```
ExtraSLHAOutputBlocks = {
  {
     (* block name *)
     FlexibleSUSYLowEnergy, NoScale,
     {
        (* entry, FlexibleSUSY's symbol for (g-2) *)
        {21, FlexibleSUSYObservable`aMuon}
     }
  }
};
```

Available observables in FlexibleSUSY:

`aMuonGM2Calc`, `aMuonGM2CalcUncertainty`, `EDM[<particle>]`, `CpHiggsPhotonPhoton`, `CpHiggsGluonGluon`, `CpPseudoScalarPhotonPhoton`, `CpPseudoScalarGluonGluon`

Enable $(g-2)_\mu$ in the MRSSM model file

Build the MRSSM:

```
$ ./createmodel --name=MRSSM
$ ./configure --with-models=MRSSM
$ make -j4
```

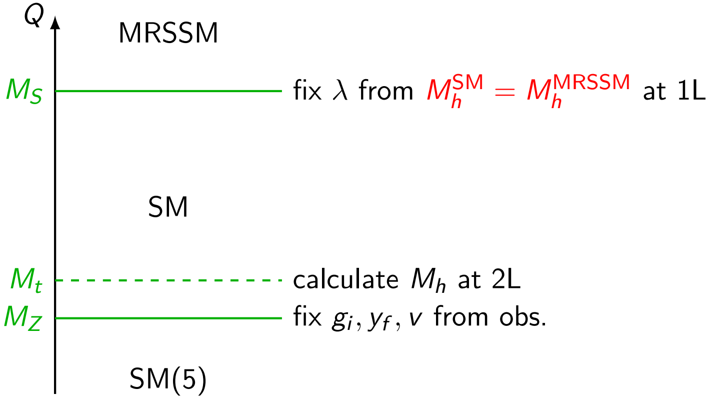
Set `FlexibleSUSY[15] = 1` in the SLHA input file and run on the command line:

```
$ cd models/MRSSM/
$ ./run_MRSSM.x --slha-input-file=LesHouches.in.MRSSM
...
Block FlexibleSUSYLowEnergy
    21    -3.86010766E-12   # Delta(g-2)_muon/2 FlexibleSUSY
```

Contents

1. What is FlexibleSUSY?
2. Features
3. Hands-on example: High-scale MSSM (HSSUSY)
   - Build the spectrum generator
   - Parameter scan
4. Hands-on example: $(g-2)_\mu$ and M_h in the MRSSM
   - Calculating $(g-2)_\mu$
   - NLL resummation of M_h (FlexibleEFTHiggs)

NLL resummation of M_h (FlexibleEFTHiggs)



See [1609.00371] for more details.

Enable NLL resummation of M_h (FlexibleEFTHiggs)

Set in the model file:

```
FlexibleEFTHiggs = True;
```

Input model parameters at the SUSY scale:

```
SUSYScaleInput = {
    {mq2, LHInput[mq2]},
    {mu2, LHInput[mu2]},
    ...
    (* fix ratio of vu/vd = TanBeta *)
    {vu, Sqrt[vu^2 + vd^2] Sin[ArcTan[TanBeta]]},
    {vd, Sqrt[vu^2 + vd^2] Cos[ArcTan[TanBeta]]}
};

(* fix Sqrt[vu^2 + vd^2] in the MRSSM *)
(* VEV is the SM-like VEV in the MRSSM *)
MatchingScaleInput = {
    {vu, VEV Sin[ArcTan[vu/vd]]},
    {vd, VEV Cos[ArcTan[vu/vd]]}
};
```

Enable NLL resummation of M_h (FlexibleEFTHiggs)

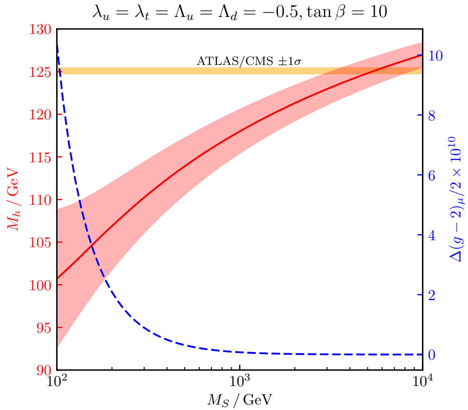
Build the MRSSMEFTHiggs:

```
$ ./createmodel --name=MRSSMEFTHiggs
$ ./configure --with-models=MRSSMEFTHiggs
$ make -j4
```

Run on the command line:

```
$ cd models/MRSSMEFTHiggs/
$ ./run_MRSSMEFTHiggs.x \
    --slha-input-file=LesHouches.in.MRSSMEFTHiggs
...
Block MASS
        25     1.25072439E+02   # hh
   1000021     5.13209677E+03   # Glu
   3000022     5.15724370E+03   # sigma0
   3000021     1.22138476E+04   # phi0
       ...
Block FlexibleSUSYLowEnergy Q= 1.00000000E+03
    21     2.61171353E-13   # Delta(g-2)_muon/2 FlexibleSUSY
```

MRSSM



$9m^2_{\tilde{q}} = 9m^2_{\tilde{u}} = m^2_{\tilde{d}} = m^2_{\tilde{l}} = m^2_{\tilde{e}} = M^2_S$, $\mu_u = \mu_d = M_S$

ΔM_h via variation of Q_{pole}, Q_{match} and $y_t^{(2L)}(M_Z)$ vs. $y_t^{(3L)}(M_Z)$

Summary

FlexibleSUSY creates precise spectrum generators for SUSY and non-SUSY models.

Observables: M_h, M_W, BSM masses, $(g-2)_\mu$, EDMs, $h \to \gamma\gamma$, $h \to gg$

High precision: 2-loop RGEs, 1-loop self energies, 1-loop thresholds, NLL resummation for M_h (`FlexibleEFTHiggs`), model-specific 2-loop and 3-loop corrections from the literature

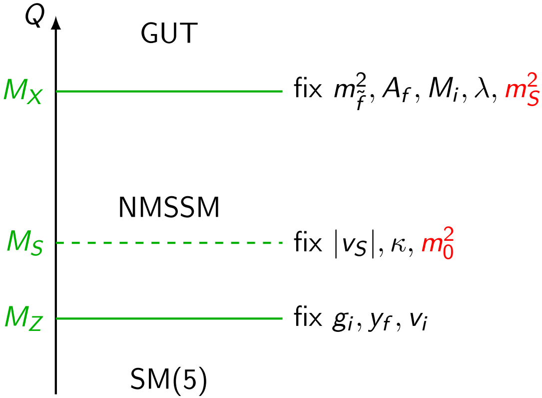
Flexibility: multiple BVP solvers, user-defined BCs, modular C++ code, SLHA input/output, `Mathematica` input/output, SQLite output

High speed: multi-threading, smart linear algebra, lazy evaluation

Planned extensions: decays, EWPOs, EFT towers

Backup

Example: CNMSSM




```
FSBVPSolvers = { SemiAnalyticSolver };
```