# **SPheno** 4.0.0: extensions beyond SPheno 3.3.0

W. Porod[1]

Institut für Theoretische Physik und Astrophysik, Universität Würzburg
D-97074 Würzburg, Germany

**Abstract**

We give a brief summary describing additional features and models implemented in `SPheno` 4.0. beyond those of `SPheno` 3.3.0. In addition a complete list of SLHA extensions is given.

## 1. Introduction

The main features of `SPheno` are described in [1, 2]. Starting with version `SPheno` 4.0.0 a two-scale matching has been implemented, where the Standard Model (SM) gauge couplings, Yukawa couplings and the Higgs-self coupling are calculated from experimental input at the scales $m_Z$ and $m_t$. The SM is then matched to a supersymmetric model at the scale $M_{SUSY}$. The details of this procedure can be found in ref. [3] as in this addendum we merely collect the corresponding flags and error messages.

A further model has been implemented in addition: general mirage mediation as given in ref. [4]. This model is specifed at the scale of grand unification $M_{GUT}$ by the seven parameters $m_{3/2}$, $\alpha$, $\tan\beta$, $a_3$, $c_m$, $c_{H_u}$ and $c_{H_d}$ at $M_{GUT}$. Alternatively one can replace $c_{H_u}$ and $c_{H_d}$ by the values of the superpotential parameter $\mu$ and the mass $m_A$ of the pseudoscalar Higgs boson at the electroweak scale.

The SLHA2 conventions [5] have extended accordingly. For the convenience of the reader we give in the next sections the complete set of extensions of the SLHA2 conventions made for both the input and the output, respectively. This includes also models which are specifically listed in [2].

## 2. Extensions to SLHA

In this section we describe the `SPheno` specific extensions to the SUSY Les Houches Accord (SLHA) [5, 6]. We start first with extensions to existing blocks and then discuss new blocks which either control the behaviour of `SPheno` or contain additional model parameters for MSSM extensions. Note, that all additional Yukawa couplings have been implemented in complex forms and the corresponding information can be passed by using the corresponding blocks starting with `IM` [5].

### *2.1. Extensions of existing blocks*

#### *2.1.1. Block* `EXTPAR`

This has been extend to include the possibility to include the input for general mirage mediation as given in [4] by the following entries

210: gravitino mass $m_{3/2}$

211: parameter $\alpha$ characterizing the scale of gaugino unification $\mu_{mir} = M_{GUT} e^{-8\pi^2/alpha}$

212: parameter $a_3$ which is a measure how much the trilinear couplings deviate from the minimal mirage medation

213: $c_m$ measure of flux contribution to soft-masses squared of sfermions

[1] porod@physik.uni-wuerzburg.de

*Preprint submitted to Elsevier* *May 22, 2017*

214: $c_{H_u}$ measure of flux contribution to soft-masses squared of $H_u$

215: $c_{H_d}$ measure of flux contribution to soft-masses squared of $H_d$

Alternative to $c_{H_u}$ and $c_{H_d}$ one can also give at the electroweak scale the values of $\mu$ (entry 23) and the mass of the pseudoscalar Higgs bosons $m_A$ (entry 26).

*2.1.2. Block* `MINPAR`

In case of extending the model by a minimal $SU(5)$ as used in [7] this block gets extended by the following entries

7: $SO(10)$ scale where the universal soft SUSY breaking parameters are defined.

8: extra $D$-terms due to the breaking of $SO(10)$ to $SU(5)$

9: $\lambda$-coupling of the Higgs 24-plet to the $\bar{5}_H$

10: $\lambda'$-coupling of the Higgs 24-plet to the $5_H$

*2.1.3. Block* `MODSEL`

In the case that generation mixing is switched on, i.e. the entry 6 contains a non-zero value, then independent of this value flavour violation is switched on in the (s)lepton as well as in the (s)quark sector.

An additional switch has been added to flag 1 (choice of SUSY breaking model)

4: general mirage mediation according to ref. [4]

Seven switches have been added to flag 3 (particle content), of which 111, 112, 113 and 114 correspond to the extensions proposed in [8]:

2: includes the particle content of a minimal $SU(5)$ model between $M_{\mathrm{GUT}}$ and a user chosen $SO(10)$ scale, where the SUSY boundary conditions are set. The details of this model are described in [7]. In this case the mass parameters of the right handed neutrinos are stored in the block `MNURNURIN` (section 2.2.4) and the corresponding neutrino Yukawa couplings can be stored in the block `YNURLHUIN` (section 2.2.12). The data is understood to be defined at the GUT-scale. The additional $SU(5)$ parameters as well as the $SO(10)$ scale are specified as extensions of the block `MINPAR`, see section 2.1.2

3: includes three right-handed (s)neutrinos with a common mass for all three neutrinos. The neutrino Yukawa couplings $Y_\nu$ can be specified at the GUT-scale, see section 2.2.12, and the mass of the right-neutrinos at their proper scale, see section 2.2.4.

5: includes one pair of 15-plet to realize the seesaw II where the formulas of [9] including the corrections presented in [10] and the 2-loop contributions to the RGEs of the gauge couplings and gaugino mass parameters have been implemented. This is an alternative to flag 112 neglecting the 2-loop running of the seesaw paramters between the triplet scale and the GUT-scale. This implies somewhat less accuracy compared to the complete case but is a good approximation, with relative differences below one per-cent, if the triplet-scale is above $5 \cdot 10^{13}$ GeV. The additional model data are specified in the blocks `M15IN`, `YHD15THDIN`, `YHU15TBHUIN` and `Y15IN`, see sections 2.2.1, 2.2.9, 2.2.10 and 2.2.7, respectively.

111: includes three right-handed (s)neutrinos which are included at their proper mass scale. The neutrino Yukawa couplings $Y_\nu$ can be specified at the GUT-scale, see section 2.2.12, and the masses of the right-neutrinos at their proper scale, see section 2.2.4.

112: includes one pair of Higgs 15-plet to realize the seesaw II where the complete 2-loop RGEs as given in [11] are used. The additional model data are specified in the blocks `M15IN`, `YHD15THDIN`, `YHU15TBHUIN` and `Y15IN`, see sections 2.2.1, 2.2.9, 2.2.10 and 2.2.7, respectively.

113: includes three Higgs 24-plets to realize the seesaw type III where the complete 2-loop RGEs as given in [11] are used. The additional model data are specified in the blocks `M24IN` and `Y24IN`, see sections 2.2.3 and 2.2.8, respectively.

114: includes one Higgs triplet to realize the seesaw II where the formulas of [9] including the corrections presented in [10] and the 2-loop contributions to the RGEs of the gauge couplings and gaugino mass parameters have been implemented. The additional model data are specified in `M15T15TBIN`, `YHD15THDIN`, `YHU15TBHUIN` and `YL15TLIN`, see sections 2.2.2, 2.2.9, 2.2.10 and 2.2.7, respectively.

### *2.2. New input blocks*

Some of these blocks have become part of the proposal given in ref. [8]: `SEESAWGENERATIONS`. In the output the blocks will be given without the ending IN. It is understood that the input values are given at the GUT scale as a default.

#### *2.2.1. Block* `M15IN`

This gives the mass $M_T$ of the 15-plet at the GUT scale. In addition the indices (1,1) have to be given to make the 1-generation case compatible with the case of several generations of 15-plets. The data are given in the format

```
(2x,2i3,2x,1p,e16.8,0p,2x,'# ',a)
```

At the scale $M_T$ the 15-plet is split into three different represenations denoted by $S$, $T$, and $Z$ [9] which have different masses due to RGE effects. The corresponding output blocks at this scale are `M15S15SB`, `M15T15TB` and `M15Z15ZB` and the same data format as for `M15IN` is used.

#### *2.2.2. Block* `M15T15TBIN`

This gives the mass $M_T$ of the SU(2) triplet at the GUT scale. In addition the indices (1,1) have to be given to make the 1-generation case compatible with the case of several generations of triplets. The data are given in the format

```
(2x,2i3,2x,1p,e16.8,0p,2x,'# ',a)
```

#### *2.2.3. Block* `M24IN`

Here one can specify the mass matrix of the 24-plets $M_{Wij}$ at $M_{\mathrm{GUT}}$ for the seesaw type III model using the formulas of [11], where the data are given in the `FORTRAN` format

```
(1x,2i3,3x,1p,e16.8,3x,'#',a)
```

where the first two integers in the format correspond to $i$ and $j$ and the double precision number to the mass parameter.

At the different scales corresponding to the mass parameters of the $SU(2)$ triplets the various mass matrices for the masses of the singlet, $SU(2)$-tripelt, the $SU(3)$-octet and the $X$-particles are given in the blocks `M24B24B`, `M24W24W`, `M24G24G` and `M24X24X`, respectively

#### *2.2.4. Block* `MNURNURIN`

In this block one can specify the masses of the right-handed neutrinos within the seesaw I model. The masses $m_{Ri}$ are specified in the `FORTRAN` format

```
(1x,2i3,3x,1p,e16.8,3x,'#',a).
```

Note, that the program assumes that the mass parameters are given in the basis where the mass matrix of the right handes neutrinos is diagonal.

Table 1: Default values for fitting R-parity violating parameters if the entries in block `NeutrinoBoundsIn` are not specified. The values are taken from [12] and correspond to the 1 $\sigma$ range but for $|U_{e3,max}|^2$ which is 90% CL.

| | | | | | |
|---|---|---|---|---|---|
| $\tan^2\theta_{atm,min}$ | 0.8182 | $\tan^2\theta_{sol,min}$ | 0.4286 | $\|U^2_{e3,min}\|^2$ | 0 |
| $\tan^2\theta_{atm,max}$ | 1.3256 | $\tan^2\theta_{sol,max}$ | 0.4970 | $\|U_{e3,max}\|^2$ | 0.035 |
| $\Delta m^2_{atm,min}$ | $2.36 \cdot 10^{-21}$ GeV$^2$ | $\Delta m^2_{sol,min}$ | $7.46 \cdot 10^{-23}$ GeV$^2$ | | |
| $\Delta m^2_{atm,max}$ | $2.54 \cdot 10^{-21}$ GeV$^2$ | $\Delta m^2_{sol,max}$ | $7.83 \cdot 10^{-23}$ GeV$^2$ | | |

*2.2.5. Block* `NeutrinoBoundsIn`

One can use `SPheno` to obtain R-parity violating parameters consistent with neutrino data. The corresponding default values are given in table 2.2.5. This block can be used to modify them. The `FORTRAN` format is

```
(1x,i2,3x,1p,e16.8,0p,3x,#,1x,a)}
```

and the entries correspond to

1: $\Delta m^2_{atm,min}$ ... lower bound on the athmospheric mass difference in $GeV^2$

2: $\Delta m^2_{atm,max}$ ... upper bound on the athmospheric mass difference in $GeV^2$

3: $\tan^2\theta_{atm,min}$ ... lower bound on the tan squared of the athmospheric mixing angle

4: $\tan^2\theta_{atm,max}$ ... upper bound on the tan squared of the athmospheric mixing angle

5: $\Delta m^2_{sol,min}$ ... lower bound on the solar mass difference in $GeV^2$

6: $\Delta m^2_{sol,max}$ ... upper bound on the solar mass difference in $GeV^2$

7: $\tan^2\theta_{sol,min}$ ... lower bound on the tan squared of the solar mixing angle

8: $\tan^2\theta_{sol,max}$ ... upper bound on the tan squared of the solar mixing angle

9: $|U^2_{e3,min}|^2$ ... lower bound on the mixing element $U_{e3}$ squared (reactor angle)

10: $|U^2_{e3,max}|^2$ ... upper bound on the mixing element $U_{e3}$ squared

*2.2.6. Block* `SPhenoInput`

This block sets the `SPheno` specific flags. The FORTRAN format is

```
(1x,i2,3x,1p,e16.8,0p,3x,#,1x,a)}
```

and the entries correspond to

1: sets the error level

2: if 1 the the SPA conventions [13] are used

3: takes a spectrum which is given by an external program

4: introduces an extension of the SLHA output: in the case of flavour violation, flavour ordered states are used instead of mass ordered states.

6: if 1 then the neutrino Yukawa couplings will be set at the largest of the corresponding seesaw particle instead of at $m_{GUT}$. This applies for all three seesaw types.

9: Starting with version 3.3.0 the formulas of [14] are used to resum the chirally enhanced terms in the calculation of the Yukawa couplings of $b$-quark and $\tau$ lepton as this improves the numerical stability for large trilinear couplings. In case one wants to use the previous implemention for this resummation, one has to set this entry to 1.

10: Starting with version 3.3.3 the renormalistion scale $M_{EWSB}$ is calculated using the tree-level values of the stop masses in contrast to previous versions where the loop-corrected masses had been used. In case one wants to use loop-corrected masses , one has to set this entry to 1.

11: if 1 then the branching ratios of the SUSY and Higgs particles are calculated, if 0 then this calculation is omitted.

12: sets minimum value for a branching ratios, so that it appears in the output

13: if 0 then the branching ratios of the decays $h \to VV^*$ are folded with the branching ratios of the off-shell vector boson, otherwise these branching ratios are written as 2-body decays. 0 is the default.

21: if 1 then the cross sections of SUSY and Higgs particles in $e^+e^-$ annihilation are calculated, if 0 then this calculation is omitted.

22: sets the center of mass energy $E_{cms}$

23: sets the electron polarisation $P_m$

24: sets the positron polarisation $P_p$

25: whether to use initial state radation in the calculation of the cross sections

26: sets minimum value for a cross section, so that it appears in the output

31: sets the value of $M_{\text{GUT}}$, otherwise $M_{\text{GUT}}$ is determined by the condition $g_1 = g_2$

32: sets strict unification, i.e. $g_1 = g_2 = g_3$

34: sets the relative precision with which the masses are calculated, default is $10^{-6}$

35: sets the maximal number of iterations in the calculation of the masses, default is 40

36: whether to write out debug information for the loop calculations

38: this entry sets the loop order of the RGEs:

   1 use one-loop SM and SUSY-RGEs

   2 use two-loop SM and SUSY-RGEs (default)

   3 use three-loop SM-RGEs and two-loop SUSY-RGEs

41: sets the width of the Z-boson $\Gamma_Z$, default is 2.49 GeV

42: sets the width of the W-boson $\Gamma_W$, default is 2.06 GeV

45: if 1 then Higgs masses calculation will be performed at 1-loop level if the 2-loop corrections are equal or larger then the 1-loop parts.

48: if 0 then use 2-loop QCD corrections to $Y_t$ and $\alpha_s$ at $m_Z$ (default); if 1 use 3-loop fit formula as given in [15].

49: if 0 use two-loop matching as described in [3] (default); if 1 use the previous matching as done in `SPheno` 3.3 and before.

50: the new default is to run the SM to $M_{SUSY}$ and then include the SUSY spectrum. In the older versions, the complete SUSY spectrum was included already at the scale $m_Z$. Setting this flag to 1 switches from the new approach to the old one.

80: if not set 0 the program exists with a non-zero value if a problem has occured

90: if 1 add R-parity to a high scale spectrum calculated either from mSUGRA, GMSB or AMSB boundary conditions

91: if 1 than bilinear parameters are calculated such that neutrino data are fitted in the experimenatal allowed range (the range can be changed using the Block `NeutrinoBoundsIn`, see section 2.2.5)

92: if 1 gives in case of R-parity violation only the $4 \times 4$ MSSM part of the neutrino/neutralino mixing matrix $N$ and the correspondingly the $2 \times 2$ parts of the charged lepton/chargino mixing matrices $U$ and $V$ as well as the block for the stau mixing. This is in particular useful in case one uses the program `Prospino` [16] or older versions of the program `Phythia` [17].

In case of the entries 22, 23 and 24 the program accepts up to 100 combinations of these quantities in a single run.

*2.2.7. Block* `Y15IN`

Here one can specify the neutrino Yukawa $Y^T_{ij}$ coupling at $M_{\rm GUT}$ for the seesaw type II model with a complete 15-plet at the GUT scale [9–11], where the data is given in the `FORTRAN` format

```
(1x,3i3,3x,1p,e16.8,3x,'#',a)
```

where the first integers in this format corresponds to $i$, the second is always 1 as there is only 15-plet present and third one corresponds to $j$. The double precision number gives the corresponding entry of the Yukawa coupling.

At the scale $M_T$ three different Yukawa couplings $Y_S$, $Y_T$ and $Y_Z$ are present [9] which are stored in the blocks `YD15SD`, `YL15TL` and `YD15ZL` using the format as for the input.

*2.2.8. Block* `Y24IN`

Here one can specify the neutrino Yukawa $Y^{III}_{ij}$ coupling at $M_{\rm GUT}$ for the seesaw type III model using the formulas of [11], where the data are given in the `FORTRAN` format

```
(1x,2i3,3x,1p,e16.8,3x,'#',a)
```

where the first two integers in the format correspond to $i$ and $j$ and the double precision number to Yukawa coupling.

*2.2.9. Block* `YHD15THDIN`

Here one can specify the Yukawa $\lambda_1$ coupling at $M_{\rm GUT}$ for the seesaw type II model where the data is given in the `FORTRAN` format

```
(1x,3i3,3x,1p,e16.8,3x,'#',a)
```

where the integers in this format are all 1 as in the implemented model only one $H_d$ and pair of 15-plets (triplets) are present. The double precision number gives the Yukawa coupling.

*2.2.10. Block* `YHU15TBHUIN`

Here one can specify the Yukawa $\lambda_2$ coupling at $M_{\rm GUT}$ for the seesaw type II model where the data is given in the `FORTRAN` format

```
(1x,3i3,3x,1p,e16.8,3x,'#',a)
```

where the integers in this format are all 1 as in the implemented model only one $H_u$ and pair of 15-plets (triplets) are present. The double precision number gives the Yukawa coupling.

*2.2.11. Block* `YL15TLIN`

Here one can specify the neutrino Yukawa $Y^T_{ij}$ coupling at $M_{\rm GUT}$ for the seesaw type II model using the formulas of [9], where the data is given in the `FORTRAN` format

```
(1x,3i3,3x,1p,e16.8,3x,'#',a)
```

where the first integers in this format corresponds to $i$, the second is always 1 as there is only triplet present and third one corresponds to $j$. The double precision number gives the corresponding entry of the Yukawa coupling.

*2.2.12. Block* `YNURLHUIN`

This block specifies the neutrino Yukawa couplings $Y_\nu$ at the GUT scale and the corresponding superpotential term is given by $W = Y_{\nu,ij}\hat{\nu}_i^C \hat{L}_j \hat{H}_u$. It is assumed that the right-handed neutrinos are in the mass eigenbasis. The real parts are specified in the block `YNuRLHuIN` with the FORTRAN format

```
(1x,3i3,3x,1p,e16.8,3x,'#',a)
```

and the imaginary parts in the block `IMYNuRLHuIN` with the same FORTRAN input. The thrird integer is always 1 as only $H_u$ is considered in the implemented model.

*2.3. New output blocks*

*2.3.1. Blocks to transfer data to* `HiggsBounds`

The program `HiggsBounds` [18, 19] can be used to calculate constraints from the Higgs sectors in a large class of models. For the data transfer the additional blocks `HiggsBoundsInputHiggsCouplingsBosons` and `HiggsBoundsInputHiggsCouplingsFermions` are required where varios ratios of couplings are stored. In `HiggsBoundsInputHiggsCouplingsFermions` the ratios of couplings of $h^0$, $H^0$ and $A^0$ to third generation fermions are stored, whereas `HiggsBoundsInputHiggsCouplingsBosons` contains the ratios of couplings to gauge bosons. In the latter case we give all required trilinear couplings including the loop induced coupling to gluons where we have taken the formulas of ref. [20]. The required loop-induced quartic couplings of one Higgs boson to two gluons and one Z-boson is not calculated and, thus, set to zero.

*2.3.2. Block* `SEESAWGENERATIONS`

This gives the number of generations of heavy particles involved in the corresponding seesaw mechanism [8]. Here the first entry gives the field and the second the numbe of generations. For the first entry the following numbers are used:

1: right-handed neutrinos

15: 15-plets

24: 24-plets

The data is given in the `FORTRAN` format

```
(1x,i2,3x,i3,"# ",a)
```

*2.3.3. Block* `SPhenoLowEnergy`

In this block the calculated values of the low energy observables are given:

1 $BR(b \to s\gamma)$

2 $BR(b \to s\mu^+\mu^-)$

3 $BR(b \to s\sum_i \nu_i\nu_i)$

4 $BR(B_d^0 \to e^+e^-)$

5 $BR(B_d^0 \to \mu^+\mu^-)$

6 $BR(B_d^0 \to \tau^+\tau^-)$

7 $BR(B_s^0 \to e^+e^-)$

8 $BR(B_s^0 \to \mu^+\mu^-)$

9 $BR(B_s^0 \to \tau^+\tau^-)$

10 $BR(B_u \to \tau^+\nu)$

11 $BR(B_u \to \tau^+\nu)/BR(B_u \to \tau^+\nu)_{SM}$

12 $\Delta(M_{B_s^0})$ [in ps$^{-1}$]

13 $\Delta(M_{B_d^0})$ [in ps$^{-1}$]

16 $\epsilon_K$

17 $\Delta(M_K)$

18 $BR(K_L \to \pi^0\nu\nu)$

19 $BR(K^+ \to \pi^+\nu\nu)$

20 SUSY contribution to the anomalous magnetic moment of the electron $\Delta(\frac{g-2}{2})_e$

21 SUSY contribution to the anomalous magnetic moment of the muon $\Delta(\frac{g-2}{2})_\mu$

22 SUSY contribution to the anomalous magnetic moment of the tau $\Delta(\frac{g-2}{2})_\tau$

23 electric dipole moment of the electron $d_e$

24 electric dipole moment of the muon $d_\mu$

25 electric dipole moment of the tau $d_\tau$

26 $BR(\mu \to e\gamma)$

27 $BR(\tau \to e\gamma)$

28 $BR(\tau \to \mu\gamma)$

29 $BR(\mu^+ \to e^+e^+e^-)$

30 $BR(\tau^+ \to e^+e^+e^-)$

31 $BR(\tau^+ \to \mu^+\mu^+\mu^-)$

39 SUSY contribution to the $\rho$-parameter

40 $BR(Z^0 \to e^\pm\mu^\mp)$

41 $BR(Z^0 \to e^\pm\tau^\mp)$

42 $BR(Z^0 \to \mu^\pm\tau^\mp)$

Note, that for the calculation of all observables we include all phases and flavour mixing.

## 3. Installation and implementing new models

### *3.1. Installation*

`SPheno` can be downloaded from

`http://projects.hepforge.org/spheno/`

where the latest tar-ball `SPheno4.x.y.tar.gz` can found as well as older versions. Unpacking will create the directory `SPheno4.x.y` where `x` and `y` are integers corresponding to the sub-version. This directory will contain the following subdirectories:

- `bin`: here the executable `SPheno` will be stored
- `doc`: contains the `SPheno` documentations
- `include`: here all the mod-files are stored

- `input`: contains input example files
- `lib`: here the library `libSPheno.a` will be stored
- `output`: contains the output files corresponding to the examples stored in `input`
- `src`: contains the source code

The directory `SPheno4.x.y` contains a Makefile which can be used to compile `SPheno`. The default compiler is Intels ifort, but by typing `make F90=compiler` on the console one can use a different compiler where `compiler` has to replaced by the compiler's name. The following compilers have been added NAG nagfor, Lahey lf95 and g95.

It is well known that compilation of the module `RGEs.F90` can be time consuming due to the length of the 2-loop RGEs for the seesaw models of type II and type III. For this reason they are not compiled by default. If the corresponding RGEs should be included then the line

```
PreDef = -DGENERATIONMIXING -DONLYDOUBLE
```

should be replaced by

```
PreDef = -DGENERATIONMIXING -DONLYDOUBLE -DSEESAWIII
```

i.e. add `-DSEESAWIII`.

In the case that one want to have quadruple precision in various parts of the code instead of double precision, one has to take out the `-DONLYDOUBLE` in the line mentioned above. Note that this can substantially slow down `SPheno`. Moreover, not all parts are yet implemented with quadruple precision. The main focus has been on the loop functions as well as on mixing between neutralinos and neutrinos in case of R-parity violation.

### *3.2. Implementing new models*

New models can easily implemented using the `SARAH` package [21, 22]. For this purpose one has to put the code generated by `SARAH` in a new directory within the directory `SPheno4.x.y` and run the corresponding Makefile. An additional executable will be stored in the directory `bin`.

## 4. Input and output

Starting with version `SPheno` 3.1 there are two main differences with respect to the input and output

1. `SPheno` accepts only the SLHA input format as specified and all the output is given in this format. In section 2 we have described the extensions to control program specific features as well as model extensions. The orginal `SPheno` input using the files `HighScale.in`, `StandardModel.in` and `Control.in` as well as the output in the file `SPheno.out` have been disabled. Detailed error messages and warnings will also be written to the file `Messages.out`.
2. One can provide input name and output name as command line options where the first (second) name, if present, is interpreted as input (output) filename, e.g.

   ```
   SPheno InName OutName
   ```

   takes `InName` for the file containing the input and will write the output to the file `OutName`. In case that the file `InName` is not found `SPheno` will look for a file called `LesHouches.in` as default. The default name for the output is `SPheno.spc`. The length of the names `InName` and `OutName` must not exceed 60 characters.


## Acknowledgements

It is a pleasure to thank F. Staub for many interesting and useful discussions in the course of the the development of this program.

## Appendix A. Default SM values

The following default values will be used if not given in the file `LesHouches.in`.

- CKM-matrix, Wolfenstein parameters: $\lambda = 0.2265$, $A = 0.807$, $\rho = 0.141$, $\eta = 0.343$
- gauge sector: $1/\alpha_{em}(0) = 137.0359895$, $m_Z = 91.187$ GeV, $G_F = 1.16637 \cdot 10^{-5}\text{GeV}^{-2}$, $\alpha_s^{\overline{MS}}(m_Z) = 0.1184$
- lepton masses: $m_e = 510.99891$ keV, $m_\mu = 105.658$ MeV, $m_\tau = 1.7768$ GeV
- quark masses: $m_u(2\text{ GeV}) = 3$ MeV, $m_d(2\text{ GeV}) = 5$ MeV, $m_s(2\text{ GeV}) = 105$ MeV, $m_c(m_c) = 1.27$ GeV, $m_b(m_b) = 4.2$ GeV, $m_t = 171.3$ GeV; the top mass is interpreted as on-shell mass

## Appendix B. Unsupported SLHA features

Here we list the features of the SLHA conventions [5, 6] which are not yet supported:

- In `Block EXTPAR` the following entries are currently ignored:

  27: pole mass of the charged Higgs boson
  51: (GMSB only) $U(1)_Y$ messenger index
  52: (GMSB only) $SU(2)_L$ messenger index
  53: (GMSB only) $SU(3)_C$ messenger index

- the `Block QEXTPAR`
- the `Block RVLAMLLEIN`
- the `Block RVLAMLQDIN`
- the `Block RVLAMUDDIN`
- the `Block RVTLLEIN`
- the `Block RVTLQDIN`
- the `Block RVTUDDIN`
- the `Block RVDIN`
- the `Block RVM2LH1IN`

These features will be implemented within the next updates.

## Appendix C. Error messages and warnings, interpretation of the variable kont

Here we describe how to interpret the values of the variable `kont` which is used in the error system of `SPheno`. The corresponding warnings and error messages are also given in the file 'Messages.out' if the error level is set to the appropriate value.

*Appendix C.1. Module Mathematics*

-1: step size gets to small in routine `ODEint`

-2: maximal value $> 10^{36}$ `ODEint`

-3: too many steps are required in routine `ODEint`

-4: boundary conditions cannot fullfilled in routine `ODEintB`

-5: maximal value $> 10^{36}$ `ODEintB`

-6: step size gets too small in routine `ODEintB`

-7: too many steps are required in routine `ODEintB`

-8: boundary conditions cannot fullfilled in routine `ODEintC`

-9: maximal value $> 10^{36}$ `ODEintC`

-10: step size gets too small in routine `ODEintC`

-11: too many steps are required in routine `ODEintC`

-12: step size gets too small in routine `rkqs`

-13: the size of the arrays do not match in routine `ComplexEigenSystems`

-14: potential numerical problems in routine `ComplexEigenSystems`

-15: the size of the arrays do not match in routine `RealEigenSystems`

-16: potential numerical problems in routine `RealEigenSystems`

-17: the size of the arrays do not match in routine `tqli`

-18: too many iterations in routine `tqli`

-19: too high accuracy required in routine `Dgauss`

-20: too high accuracy required in routine `DgaussInt`

-21: precision problem in routine `Kappa`

-22: step size gets too small in routine `IntRomb`

-23: too many steps are required in routine `IntRomb`

-24: singular matrix in routine `GaussJ`

-25: singular matrix in routine `InverseMatrix`

-26: inversion failed in routine `InvMat3`

-27: stepsize underflow in routine `bsstep`

-28: too much extrapolation in routine `pzextr`

-29: too much extrapolation in routine `rzextr`

-30: matrix contains `NaN` in routine `RealEigenSystems`

-31: matrix contains `NaN` in routine `ComplexEigenSystems`

*Appendix C.2. Module StandardModel*

-101: routine `CalculateRunningMasses`: $Q_{low} > m_b(m_b)$

-102: routine `CalculateRunningMasses`: $\mathrm{Max}(Q_{low}, m_b(m_b) > Q_{max})$

*Appendix C.3. Module SusyMasses*

-201: negative mass squared in routine `ChargedScalarMassEps1nt`

-202: negative mass squared in routine `ChargedScalarMassEps3nt`

-204: $|Y_\tau|^2 < 0$ in routine `CharginoMass3`

-205: $|Y_\tau|^2 < 0$ in routine `CharginoMass5`

-206: negative mass squared in routine `PseudoScalarMassEps1nt`

-207: negative mass squared in routine `PseudoScalarMassEps3nt`

-208: negative mass squared in routine `PseudoScalarMassMSSMnt`

-210: negative mass squared in routine `ScalarMassEps1nt`

-211: negative mass squared in routine `ScalarMassEps3nt`

-212: negative mass squared in routine `ScalarMassMSSMeff`

-213: negative mass squared in routine `ScalarMassMSSMnt`

-215: $m^2_{S^0_1} < 0$ in routine `ScalarMassMSSMeff`

-216: $m^2_{P^0_1} < 0$ in routine `ScalarMassMSSMeff`

-217: $m^2_{S^+} < 0$ in routine `ScalarMassMSSMeff`

-220: negative mass squared in routine `SfermionMass1Eps1`

-221: negative mass squared in routine `SfermionMass1Eps3`

-222: negative mass squared in routine `SfermionMass1MSSM`

-223: negative mass squared in routine `SfermionMass3MSSM`

-224: negative mass squared in routine `SquarkMass3Eps`

-225: $m^2_{\tilde\nu} < 0$ in routine `TreeMassesEps1`

-226: $m^2_{\tilde\nu} < 0$ in routine `TreeMassesMSSM`

-227: $m^2_{A^0} < 0$ in routine `TreeMassesMSSM`

-228: $m^2_{H^+} < 0$ in routine `TreeMassesMSSM`

-229: $m^2_{\tilde\nu} < 0$ in routine `TreeMassesMSSM2`

-230: $m^2_{A^0} < 0$ in routine `TreeMassesMSSM2`

-231: $m^2_{H^+} < 0$ in routine `TreeMassesMSSM2`

-232: $m^2_{\tilde\nu} < 0$ in routine `TreeMassesMSSM3`

*Appendix C.4. Module InputOutput*

-302: routine `LesHouches_Input`: unknown entry for Block MODSEL

-303: routine `LesHouches_Input`: model must be specified before parameters

-304: routine `LesHouches_Input`: unknown entry for Block MINPAR

-305: routine `LesHouches_Input`: model has not been specified completly

-306: routine `LesHouches_Input`: a serious error has been part of the input

-307: routine `LesHouches_Input`: Higgs sector has not been fully specified

-308: routine `ReadMatrixC`: indices exceed the given boundaries

-309: routine `ReadMatrixR`: indices exceed the given boundaries

-310: routine `ReadVectorC`: index exceeds the given boundaries

-311: routine `ReadVectorR`: index exceeds the given boundaries

-312: routine `ReadMatrixC`: indices exceed the given boundaries

*Appendix C.5. Module SugraRuns*

-401: routine `BoundaryEW`: negative scalar mass squared as input

-402: routine `BoundaryEW`: $m_Z^2(m_Z) < 0$

-403: routine `BoundaryEW`: $\sin^2\theta_{\overline{DR}} < 0$

-404: routine `BoundaryEW`: $m_W^2 < 0$

-405: routine `BoundaryEW`: either $m_{l_D R}/m_l < 0.1$ or $m_{l_D R}/m_l > 10$

-406: routine `BoundaryEW`: either $m_{d_D R}/m_u < 0.1$ or $m_{d_D R}/m_d > 10$

-407: routine `BoundaryEW`: either $m_{u_D R}/m_d < 0.1$ or $m_{u_D R}/m_u > 10$

-408: routine `RunRGE`: entering non-perturbative regime

-409: routine `RunRGE`: nor $g_1 \neq g_2$ at $M_{\mathrm{GUT}}$ neither any other unification

-410: routine `RunRGE`: entering non-perturbative regime at $M_{\mathrm{GUT}}$

-411: routine `RunRGE`: entering non-perturbative regime at $M_{H_3}$

-412: routine `Sugra`: run did not converge

-413: routine `Calculate_Gi_Yi`: $m_Z^2(m_Z) < 0$

-414: routine `Calculate_Gi_Yi`: too many iterations to calculate $m_b(m_b)$ in the $\overline{MS}$ scheme

-415: routine `Sugra`: $|\mu|^2 < 0$ at $m_Z$

*Appendix C.6. Module LoopMasses*

-501 negative mass squared in routine `SleptonMass_1L`

-502 $p^2$ iteration did not converge in routine `SleptonMass_1L`

-503 negative mass squared in routine `SneutrinoMass_1L`

-504 $p^2$ iteration did not converge in routine `SneutrinoMass_1L`

-505 negative mass squared in routine `SquarkMass_1L`

-506 $p^2$ iteration did not converge in routine `SquarkMass_1L`

-507 $m^2_{h^0} < 0$ in routine `LoopMassesMSSM`

-508 $m^2_{A^0} < 0$ in routine `LoopMassesMSSM`

-509 $m^2_{H^+} < 0$ in routine `LoopMassesMSSM`

-510 $|\mu|^2 > 10^{20}$ in routine `LoopMassesMSSM`

-511 $|\mu|^2 < 0$ in routine `LoopMassesMSSM`

-512 $m^2_Z(m_Z)^2 < 0$ in routine `LoopMassesMSSM`

-513 $m^2_{h^0} < 0$ in routine `LoopMassesMSSM_2`

-514 $m^2_{A^0} < 0$ in routine `LoopMassesMSSM_2`

-515 $m^2_{H^+} < 0$ in routine `LoopMassesMSSM_2`

-516 $|\mu|^2 > 10^{20}$ in routine `LoopMassesMSSM_2`

-517 $|\mu|^2 < 0$ in routine `LoopMassesMSSM_2`

-518 $m^2_Z(m_Z)^2 < 0$ in routine `LoopMassesMSSM_2`

-519 $m^2_{h^0} < 0$ in routine `LoopMassesMSSM_3`

-520 $m^2_{A^0} < 0$ in routine `LoopMassesMSSM_3`

-521 $m^2_{H^+} < 0$ in routine `LoopMassesMSSM_3`

-522 $|\mu|^2 > 10^{20}$ in routine `LoopMassesMSSM_3`

-523 $|\mu|^2 < 0$ in routine `LoopMassesMSSM_3`

-524 $m^2_Z(m_Z)^2 < 0$ in routine `LoopMassesMSSM_3`

-525 negative mass squared in routine `Sigma_SM_chirally_enhanced`

*Appendix C.7. Module TwoLoopHiggsMass*

-601: routine `PiPseudoScalar2`: $m^2_{\tilde{t}} < 0$

-602: routine `PiPseudoScalar2`: $m^2_{\tilde{b}} < 0$

-603: routine `PiPseudoScalar2`: $m^2_{\tilde{\tau}} < 0$

-604: routine `PiScalar2`: $m^2_{\tilde{t}} < 0$

-605: routine `PiScalar2`: $m^2_{\tilde{b}} < 0$

-606: routine `PiScalar2`: $m^2_{\tilde{\tau}} < 0$

-607: routine `Two_Loop_Tadpoles`: $m^2_{\tilde{t}} < 0$

-608: routine `Two_Loop_Tadpoles`: $m^2_{\tilde{b}} < 0$

-609: routine `Two_Loop_Tadpoles`: $m^2_{\tilde{\tau}} < 0$

*Appendix C.8. Module MathematicsQP*

-1001: the size of the arrays do not match in routine `ComplexEigenSystems_DP`

-1002: potential numerical problems in routine `ComplexEigenSystems_DP`

-1003: the size of the arrays do not match in routine `ComplexEigenSystems_QP`

-1004: potential numerical problems in routine `ComplexEigenSystems_QP`

-1005: the size of the arrays do not match in routine `RealEigenSystems_DP`

-1006: potential numerical problems in routine `RealEigenSystems_DP`

-1007: the size of the arrays do not match in routine `RealEigenSystems_QP`

-1008: the size of the arrays do not match in routine `Tqli_QP`

-1009: too many iterations in routine `Tqli_QP`

-1010: too many iterations in routine `Tql2_QP`

## References


[1] W. Porod, Comput. Phys. Commun. **153** (2003) 275 [arXiv:hep-ph/0301101].
[2] W. Porod and F. Staub, Comput. Phys. Commun. **183** (2012) 2458 doi:10.1016/j.cpc.2012.05.021 [arXiv:1104.1573 [hep-ph]].
[3] F. Staub and W. Porod, arXiv:1703.03267 [hep-ph].
[4] H. Baer, V. Barger, H. Serce and X. Tata, Phys. Rev. D **94** (2016) no.11, 115017 doi:10.1103/PhysRevD.94.115017 [arXiv:1610.06205 [hep-ph]].
[5] B. C. Allanach *et al.*, Comput. Phys. Commun. **180** (2009) 8 [arXiv:0801.0045 [hep-ph]].
[6] P. Z. Skands *et al.*, JHEP **0407** (2004) 036 [arXiv:hep-ph/0311123].
[7] F. Deppisch, A. Freitas, W. Porod and P. M. Zerwas, Phys. Rev. D **77** (2008) 075009 [arXiv:0712.0361 [hep-ph]].
[8] G. Brooijmans *et al.*, arXiv:1203.1488 [hep-ph].
[9] A. Rossi, Phys. Rev. D **66** (2002) 075003 [arXiv:hep-ph/0207006].
[10] F. Borzumati and T. Yamashita, Prog. Theor. Phys. **124** (2010) 761 [arXiv:0903.2793 [hep-ph]].
[11] J. N. Esteves *et al.*, Phys. Rev. D **83** (2011) 013003 [arXiv:1010.6000 [hep-ph]].
[12] T. Schwetz, M. Tortola and J. W. F. Valle, arXiv:1103.0734 [hep-ph].
[13] J. A. Aguilar-Saavedra *et al.*, Eur. Phys. J. C **46** (2006) 43 [arXiv:hep-ph/0511344].
[14] A. Crivellin, L. Hofer and J. Rosiek, JHEP **1107** (2011) 017 [arXiv:1103.4272 [hep-ph]].
[15] D. Buttazzo, G. Degrassi, P. P. Giardino, G. F. Giudice, F. Sala, A. Salvio and A. Strumia, JHEP **1312** (2013) 089 doi:10.1007/JHEP12(2013)089 [arXiv:1307.3536 [hep-ph]].
[16] W. Beenakker, R. Hopker and M. Spira, arXiv:hep-ph/9611232.
[17] T. Sjostrand, S. Mrenna and P. Z. Skands, JHEP **0605** (2006) 026 [arXiv:hep-ph/0603175].
[18] P. Bechtle *et al.*, Comput. Phys. Commun. **181** (2010) 138 [arXiv:0811.4169 [hep-ph]].
[19] P. Bechtle *et al.*, Comput. Phys. Commun. **182** (2011) 2605 [arXiv:1102.1898 [hep-ph]].
[20] M. Spira, A. Djouadi, D. Graudenz and P. M. Zerwas, Nucl. Phys. B **453** (1995) 17 [hep-ph/9504378].
[21] F. Staub, arXiv:0806.0538 [hep-ph].
[22] F. Staub, Comput. Phys. Commun. **182** (2011) 808 [arXiv:1002.0840 [hep-ph]].