APROPOS: A Program to Obtain Properties Of SUSY

Matthias Schlaffer, Ludwig-Maximilians Universität München, Germany

Supervisor: Prof. Dr. Georg Weiglein

September 11, 2011

Abstract

APROPOS is a program which combines several programs in order to provide a tool to obtain various observables of SUSY models from the high energy input and to scan the parameter space. In this version the implemented SUSY models are mGMSB, mAMSB and CMSSM. APROPOS is designed such that it is easily extendable to retrieve more observables and to include more models than the above mentioned.

This report explains the structure of the program and serves as a manual. In addition, some results are presented.

Contents

1. Introduction

Even though the standard model (SM) explains many experimental results up to a high precision, it is clear that it cannot be complete. One promising theory for physics beyond the SM is supersymmetry (SUSY). It solves several problems of the SM automatically:

- When considering the SM as effective field theory and calculating the Higgs self-energy, it diverges quadratically and must be extremely fine-tuned to get small Higgs masses. When assuming SUSY, the contribution of the fermion loops, which cause this divergence, is cancelled by the contribution of the corresponding sfermions. So SUSY makes the fine-tuning needless.

- Under the assumption of R-parity, SUSY predicts a stable, weakly interacting particle, i.e. a perfect candidate for dark matter.

- The gauge couplings unify at high energies if SUSY is broken below or approximately at 1 TeV. In the SM this is not the case.

- The SUSY algebra includes the Poincaré algebra as well as the algebras of the gauge couplings, i.e. combines gravity with the other forces.

However the minimal supersymmetric extension of the SM (MSSM) introduces more than 100 new parameters. Since most of them are used to describe the way SUSY is broken, they can be reduced to very few parameters when assuming a specific breaking mechanism and/or making some assumptions about general properties. This is usually done in order to get a model within which SUSY properties can be studied.

APROPOS combines SOFTSUSY (version 3.1.7) [1], FeynHiggs (version 2.8.5) [2–5], HiggsBounds (version 3.4.0 beta) [6,7], MW_Calc [8], which makes use of LoopTools [9], and SDECAY (version 1.3) [10] to retrieve relevant observables from the high energy input of SUSY models. By comparing these theoretical predictions with measured values, it is possible to determine if a model makes good predictions for some choice of input parameters.

2. SUSY models

It is very difficult if not impossible to break SUSY directly. Therefore it is usually assumed that SUSY is broken in a hidden sector which we cannot see and is then mediated by some mechanism to the visible sector which we can access and observe. The breaking mechanisms, i.e. models, implemented in APROPOS so far are:

- Constrained MSSM (CMSSM), where gaugino masses unify as well as the sfermion and Higgs masses [11–26]. It is described by [27]:
  1. The common scalar mass m_0
  2. The common fermion mass $m_{1/2}$

3. The ratio of the Higgs vacuum expectation values $\tan(\beta)$
4. The sign of the bilinear Higgs term in the superpotential $\text{sign}(mu)$
5. The common trilinear coupling A_0

- Minimal Gauge Mediated Symmetry Breaking (mGMSB), where SUSY breaking is mediated by gauge couplings [28, 29]. Described by [27]:
  1. The scale of soft SUSY breaking felt by the low-energy sector Λ
  2. The overall messenger scale M_{mess}
  3. The ratio of the Higgs vacuum expectation values $\tan(\beta)$
  4. The sign of the bilinear Higgs term in the superpotential $\text{sign}(mu)$
  5. The messenger index N_5
  6. The Gravitino mass factor c_{grav}
- Minimal Anomaly Mediated Symmetry Breaking (mAMSB), where SUSY breaking is mediated by the super-Weyl anomaly [30, 31], described by [27]:
  1. The common scalar mass m_0
  2. The gravitiono mass $m_{3/2}$
  3. The ratio of the Higgs vacuum expectation values $\tan(\beta)$
  4. The sign of the bilinear Higgs term in the superpotential $\text{sign}(mu)$

3. The code structure

The structure of the program and the header files will be shown using the example of mGMSB. The only structural difference between mAMSB, CMSSM and mGMSB is that in mGMSB.cc there is an additional query to ensure $\Lambda \leq M_{mess}$, which would lead to unphysical results otherwise.

All produced files are named after the same scheme *basename + number + appendix* and saved in the output directory. The output directory as well as the variable basename is specified in the parameter file "parameter.in", by default as "run" and "SLHA_", respectively. By *number* the different points in the parameter grid are distinguished and *appendix* specifies the origin of the file. In the following *basename + number* is omitted.

3.1. Header files

3.1.1. Reader.cc

The functions in Reader.cc are used to retrieve the parameters from the parameter input file. It contains the functions

- `double ReadDouble(std::string &file, std::string key)`,

- `int ReadInt(std::string &file, std::string key)` and
- `std::string ReadStr(std::string &file, std::string key)`.

They search in the file specified by `file` for a line which starts with "`key =`" and return the part behind the equal sign as double, integer and string, respectively. Everything behind the hash sign (#) is ignored, it can therefore be used to make comments.

3.1.2. convert.cc

In convert.cc several small functions are collected. They are used to execute minor, but often used, tasks:

- `double StrToDouble(std::string N)`
- `int StrToInt(std::string N)`
- `std::string IntToStr(int N)`
- `std::string FirstItem(std::string line)`
- `std::string SecondtItem(std::string line)`
- `std::string ThirdItem(std::string line)`
- `double Round(double number, int digits)`
- `std::string Length(int &N, int L)` and
- `double ABS(double N)`.

The first three convert one data type into another. When the string `N` contains a sign which is not `+-.eE012345678` or `9`, the program is aborted in the case of `StrToDouble` and `StrToInt`.

`FirstItem`, `SecondItem` and `ThirdItem` extract the first, second and third word of `line`, respectively. Words are separated by spaces and a word can be everything (letters, numbers, signs,...) that contains no spaces. These functions are used to extract values of variables out of SLHA-files.

`Length` is used to insert 0 in front of `N` until its length is `L`, its main purpose is to produce nice formatted output files. `Round` returns `number` after it has been rounded to a precision determined by `digits` and `ABS` returns the absolute value of `N`.

3.1.3. SLHAfunctions.cc

SLHAfunctions contains all functions which are used either while writing or reading from a SLHA file and which are model independent:

- `bool isgood(double l, double u, double s)`

- std::vector<std::string> allParameters(std::vector<std::string> &need, std::map<std::string, std::string> &have)
- void ShowWrongParameters(std::vector<std::string> &vect, std::string &model)
- void SMinput(std::ofstream &ofs)
- void SOFTSUSYinput(std::ofstream &ofs)
- std::string SLHAvalue(std::string &file, std::string Blockname, int N) and
- std::string SLHAvalue2(std::string &file, std::string Blockname, int N).

isgood checks if the continuous input parameters are chosen such that they do not cause infinite loops in the code, i.e. that one obtains a number exceeding the upper limit u when adding the step size s several times to the lower limit l. If everything is fine, it returns true, else false.

allParameters takes the vector need generated by mGMSBparameter 3.1.4 and modifies it such that it contains a "_min", "_max" and "_stp_size" version of each variable and an entry "sign(mu)". Then the map have, which contains all parameters of the input file, is checked against this extended vector to verify that all parameters are given and no undefined parameters were provided. Every missing or unknown parameter is written in a new vector which is returned. It is then processed by ShowWrongParameters which lists all missing and unknown parameters.

SMinput and SOFTSUSYinput are used to write the corresponding SLHA blocks in the output stream ofs. In SOFTSUSYinput some input parameters for SOFTSUSY are defined.

Finally the functions SLHAvalue and SLHAvalue2 are used to extract values out of a SLHA file specified by file. Both first search for a line beginning with "BLOCK" followed by Blockname. No distinction is made between upper and lower case to be compatible with more programs. When they have found the block, they search in this block for a line beginning with the key N and return the second or third item of this line, respectively. When they do not find the key, an error message is prompted and "0" is returned.

3.1.4. mGMSB.cc

In the model file mGMSB.cc all functions which are different for each model are collected. This allows to easily extend APROPOS to more models. For detailed instructions see section 4.1.In case of mGMSB the functions are called:

- std::vector<std::string> mGMSBparameter()

- int mGMSBwrite(std::map<std::string, std::string> &list, std::string &place, std::string &basename, int &len) and
- void evaluatemGMSB(std::string &dir).

mGMSBparameter just creates a vector containing all parameters needed for this model, except for sign(mu), the sign of the bilinear Higgs term in the superpotential. This vector is needed to check if all parameters were specified in the input file.

mGMSBwrite calls isgood 3.1.3 for each continuous parameter and in case of a positive return loops over all input parameters, writing a SLHA file for each possible combination. These files are numerated and end with "_input". They are saved in the output directory and contain the blocks "MINPAR", "SMINPUTS"and "SOFTSUSY". If isgood finds infinite loops no SLHA files are created but an error message is prompted.

The function evaluatemGMSB extracts all wanted parameters of the files produced during executing the main program and lists them in two tables, one with header to be human readable and the other one without header or delimiters to be easily readable by other programs as for example ROOT. If more variables are to be extracted one needs just to add the appropriate lines.

3.1.5. HBwithFH

The version of HBwithFH is for the most parts identical to the one included in HiggsBounds. It is extended such that it prints the result of HiggsBounds and $(g-2)_\mu$ in a SLHA-like block at the end of its output and it includes a function[1] to write "HBOutFile.dat".

3.2. Main program

First of all, the main program reads the paths to the used programs and other variables needed for running the program from the parameter file "parameter.in". Usually this file does not need to be changed after it has been adjusted once to the given system. In order to keep the results from different runs separated, a unique output directory is created. Then the function writeSLHA is called.

If no file name has been given over to APROPOS when calling it, writeSLHA reads the SUSY parameters which specify the model and the corresponding high energy input from the file called "input", else from the indicated file. While the file is read, it is copied to the output directory and all parameters are written into a map. Everything behind a hash sign (#) is ignored such that comments can be included. If sign(μ) was not given in the input file, its value is set by default to +1, all other parameters need to be specified. To check for wrong parameters the functions mGMSBparameter 3.1.4, allParameters 3.1.3 and ShowWrongParameters 3.1.3 are called successively. If no wrong parameters were found and the model is known, the function to actually write the SLHA files is called.

[1] implemented by Lisa Zeune

In case of mGMSB it is called `mGMSBwrite` 3.1.4. Finally `writeSLHA` returns the number of written SLHA files, which all have the ending "_input".

The produced high energy input files are processed one after the other. First the spectrum is calculated by softsusy, generating a "_spectrum" file. This file is then used by HBwithFH 3.1.5 to determine the anomalous magnetic moment of the muon $(g-2)_\mu$ and to check if this choice of parameters has already been excluded by Higgs searches. The output of HBwithFH is saved in a file "_HBwithFH", the automatically produced file "Key.dat" is saved as "_HBwithFH.Key". HBwithFH also writes a file called "HBOutFile.dat" which is renamed to "MathInputFile.dat" and completed by data from the "_spectrum" file to serve as input file for MW_Calc. MW_Calc then calculates the mass of the W boson and writes the results in "_MathOutputFile". Finally, SDECAY uses the "_spectrum" files to calculate branching ratios.

If no error has occurred so far, the model dependent evaluation function 3.1.4 is called, otherwise the point number will be kept to inform the user in the end.

4. Setting up and running APROPOS

In the "parameter.in" file the paths to the executables of the used programs have to be set properly. An example file is provided in appendix A. In addition the path to LoopTools has to be adjusted twice (!) in MW_Calc_Matthias[2]. Also a file called "sdecay.in" has to be present, defining settings of SDECAY.

Together with the APROPOS source code three bash scripts `compileAPROPOS.sh`, `compileHBwithFH.sh` and `compileall.sh` are delivered. They include all commands to compile the source code and link the headers correctly. In `compileHBwithFH.sh` the correct paths to the HiggsBounds and FeynHiggs libraries must be set. Also, if new header files are created, `compileAPROPOS.sh` and `compileall` must be changed accordingly.

When everything has been compiled and the "parameter.in" has been set, APROPOS can be executed by `APROPOS [file]`, where file is an optional parameter.

The input file has to contain the model, an lower and upper limit and a step size for each continuous parameter. The structure is key = value, where (#) can be used to make comments. For example input files see appendix B.

4.1. Adding new models

When a new model is to be added, a header file analogous to mGMSB.cc has to be written. Also the bash scripts `compileAPROPOS.sh` and `compileall.sh` have to be changed accordingly. Finally the model has to be included in the main program by adding the appropriate lines at the 4 places indicated by:

[2]This is a slightly different version than the one given to me by L. Zeune. The order of the input parameters is changed (tanbeta is at a different place) and the absolute value of the calculated mass is returned.

```
//******************************//
// modify to include more models //
//******************************//
```

5. Results

In the following plots the blue shaded part shows where the anomalous magnetic moment of the muon $(g-2)_\mu$ is in the 2σ interval $((30 \pm 2 \cdot 9) \cdot 10^{-10})$ and the red part shows where the W boson mass M_W is within the 1σ interval $((80.399 \pm 0.023)\text{GeV})$. The area in the parameter space where the lightest Higgs is heavier than 113 GeV is highlighted in yellow. This is the experimental limit on the Higgs mass so far, including the theoretical uncertainty of about 1.5 GeV. The lines of constant squark and gluino mass are drawn in green and cyan, respectively. The black lines show pre-defined benchmark lines.

The Higgs bound sets almost no constraint on the parameter space for CMSSM with $\tan(\beta) = 10$ as it can be seen in figure 1. However the constraints from $(g-2)_\mu$ and M_W allow only small $m_{1/2}$ below approximately 400 GeV. On the other hand it is known from direct SUSY searches at LHC that the squark and gluon masses cannot be much below 1 000 GeV. This excludes the area allowed by $(g-2)_\mu$ and M_W. Furthermore the latest exclusion graphs from LHC show that in this plot everything with $m_{1/2} \lesssim 540$ GeV is excluded.

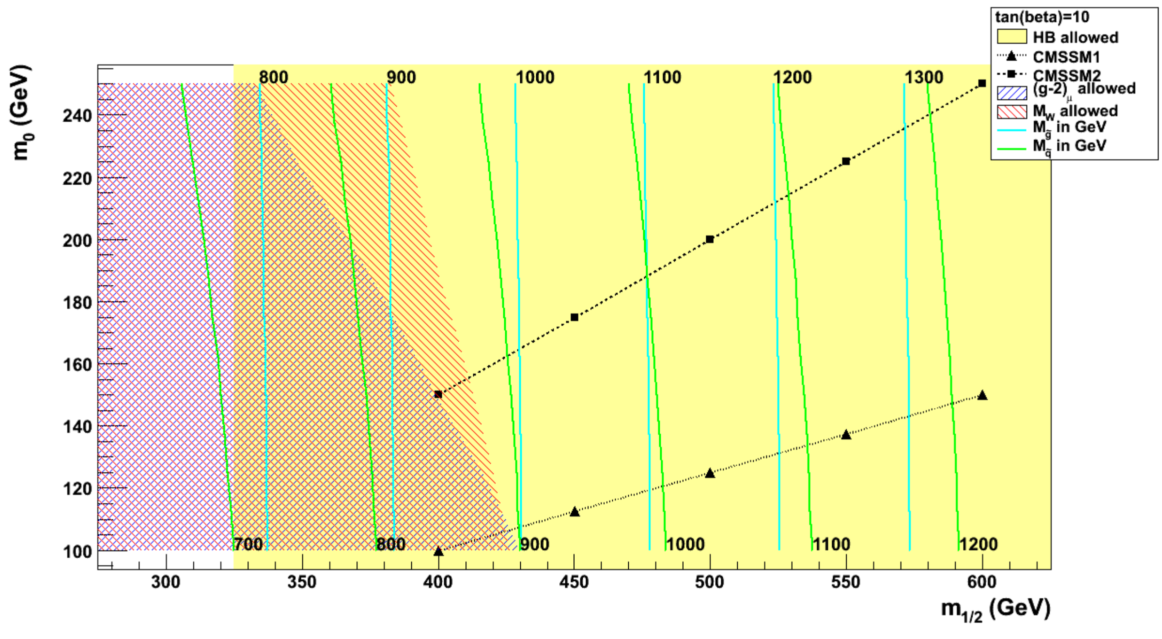

Figure 1: CMSSM with $\tan(\beta) = 10$.

When choosing a larger $\tan(\beta) = 15$ (figure 2) the area allowed by $(g-2)_\mu$ increases and shifts towards larger $m_{1/2}$, but still remains within the area excluded by the squark and gluino masses. The shifting of $(g-2)_\mu$ is easy to understand: Since

$$(g-2)_\mu \propto \frac{\tan(\beta)}{M}, \tag{1}$$

where M is a mass parameter, an increase in $\tan(\beta)$ needs also an increase in the mass parameter to keep $(g-2)_\mu$ constant. The other observables do not change with $\tan(\beta)$, leaving the M_W allowed part separated from the squark and gluino allowed part. So the CMSSM remains under some distress.

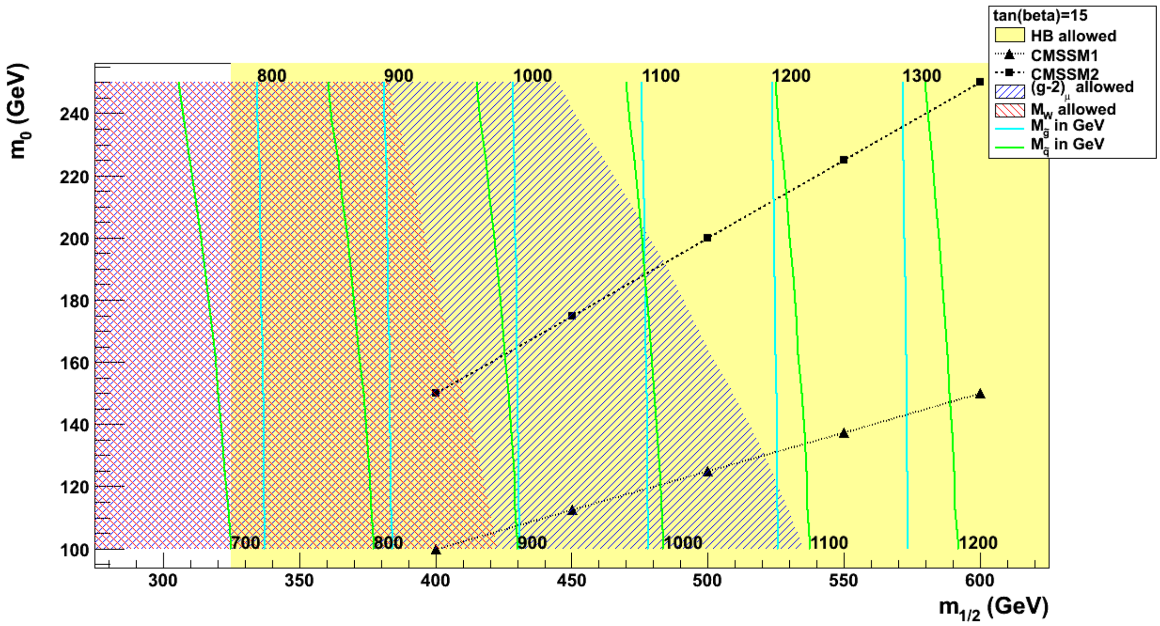

Figure 2: CMSSM with $\tan(\beta) = 15$

When considering mGMSB with $\tan(\beta) = 10$ and $N = 1$ (figure 3), it can be seen that there is some distress between the area allowed by $(g-2)_\mu$ and by the Higgs bound, too. Also the demand that the gluino and squark mass must not be much below 1 000 GeV is not in good agreement with $(g-2)_\mu$. As with CMSSM $(g-2)_\mu$ changes according to equation (1). So by choosing $\tan(\beta) = 15$ the 2σ band of $(g-2)_\mu$ is shifted towards higher Λ, while the area allowed by the Higgs bound is shifted a little downwards (figure 4). This leads to an intersection of all allowed areas near the 1 000 GeV line of the squark and gluino masses. Nevertheless it is still not much better than the CMSSM.

When increasing the messenger index N (figure 5), the Higgs bound allows almost all the parameter space, while the intervals allowed by $(g-2)_\mu$ and M_W cover both the same area. Since the squarks and gluinos are heavy in this scenario, there remains a broad strip between 46 TeV $\lesssim \Lambda \lesssim$ 56 TeV, which is in good agreement with all measurements.

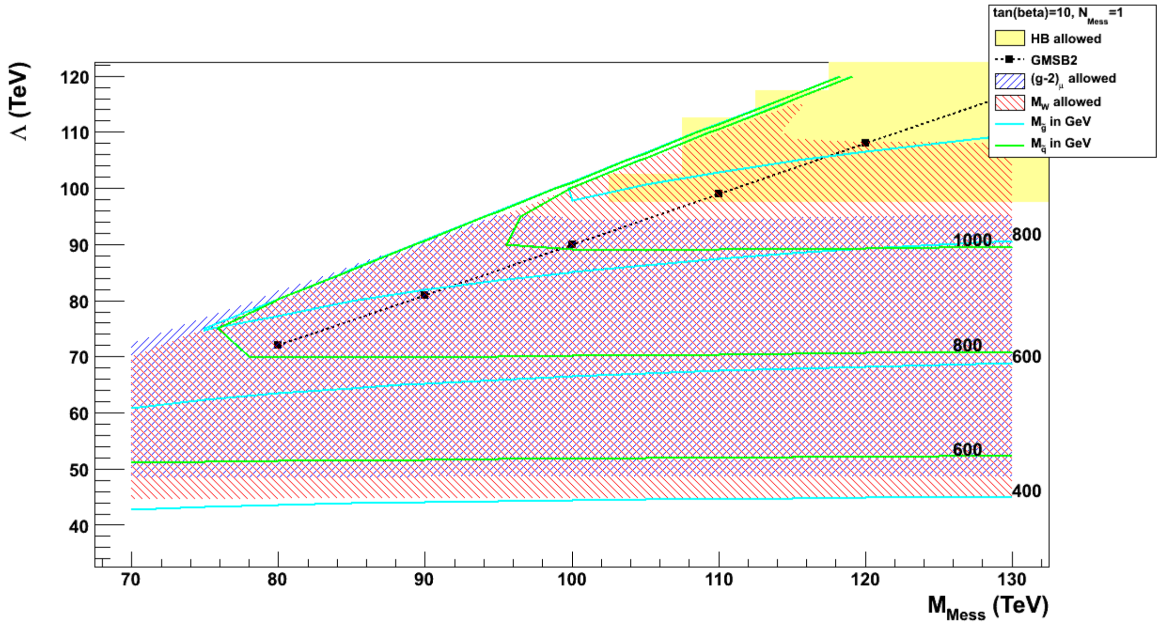

Figure 3: mGMSB with $\tan(\beta) = 10$ and $N = 1$

6. Conclusion

The previously favoured and intensively studied model CMSSM, is under some pressure due to the latest SUSY searches at LHC. Already now the best fitting point is excluded by far and a larger area of the parameter space might be excluded when the collision energy will be increased.

The mGMSB is until now in much better agreement with the measured values. Since the color splitting in this model is large, it is possible that the collision energy at LHC is not enough to produce color charged particles but is sufficient to produce color neutral particles. Due to the small cross section for the latter, more data could reveal hints for SUSY even with the present collision energy.

7. Acknowledgements

I want to thank my supervisor Georg Weiglein for spending so much time answering all of my questions so patiently. Furthermore I am grateful to Lisa Zeune for helping me to get MW_Calc running and Oscar Stål for helping with computer issues.

Last but not least, I would like to thank the Summer Student organization Team to make this interesting internship possible.

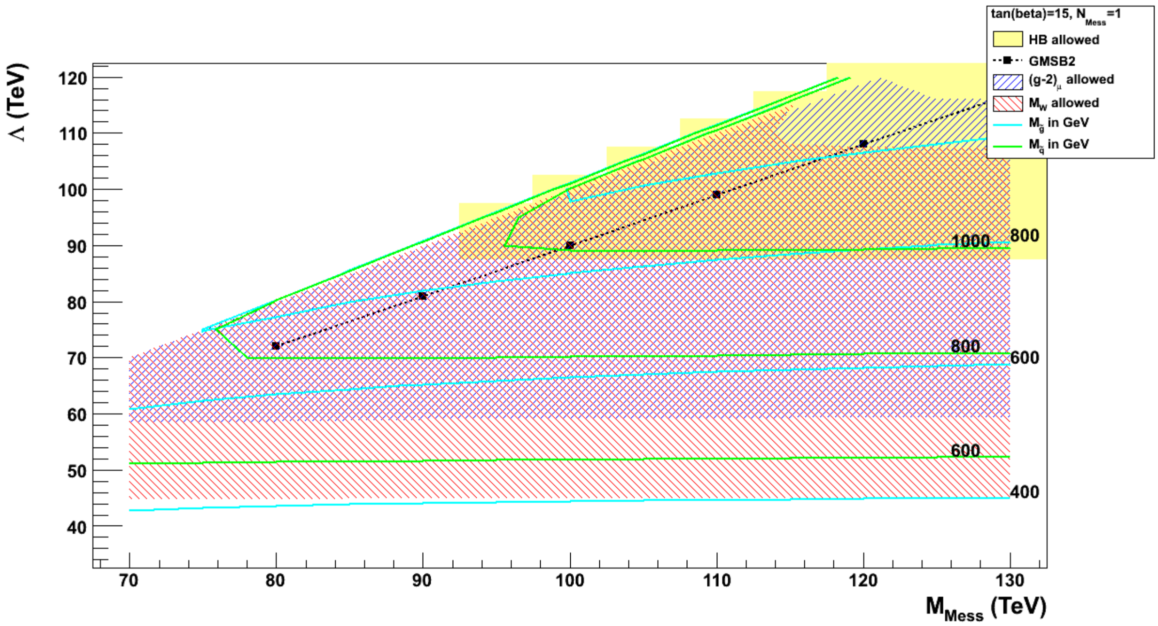

Figure 4: mGMSB with $\tan(\beta) = 15$ and $N = 1$

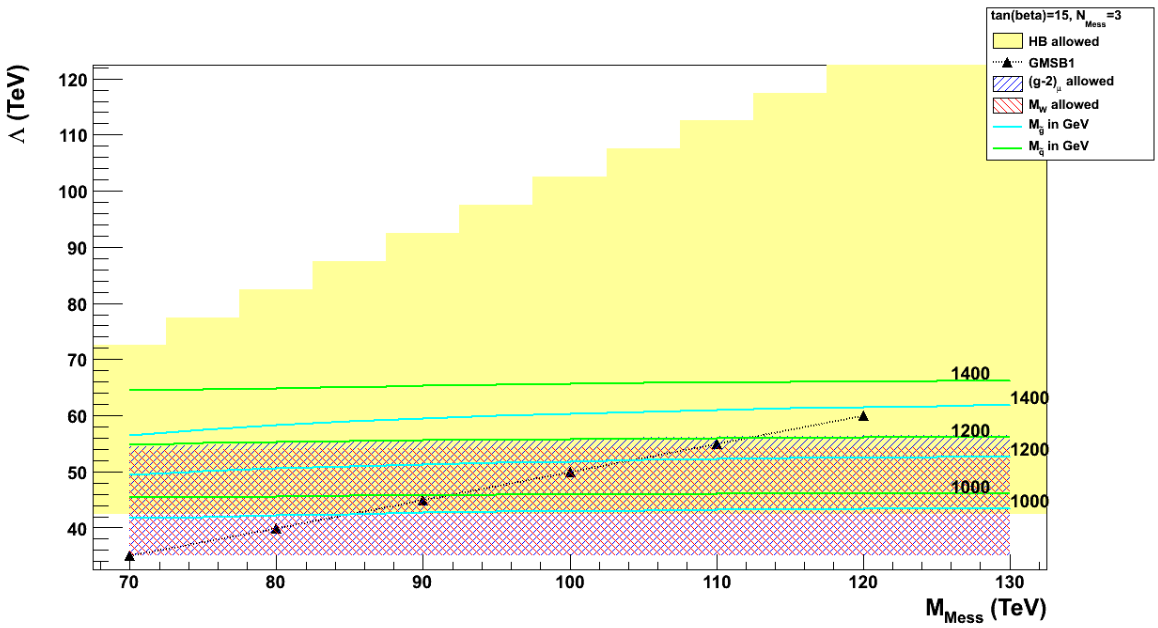

Figure 5: mGMSB with $\tan(\beta) = 15$ and $N = 3$

References

[1] B. C. Allanach, "SOFTSUSY: A C++ program for calculating supersymmetric spectra," *Comput. Phys. Commun.* **143** (2002) 305–331, `arXiv:hep-ph/0104145`.

[2] M. Frank *et al.*, "The Higgs boson masses and mixings of the complex MSSM in the Feynman-diagrammatic approach," *JHEP* **02** (2007) 047, `arXiv:hep-ph/0611326`.

[3] G. Degrassi, S. Heinemeyer, W. Hollik, P. Slavich, and G. Weiglein, "Towards high-precision predictions for the MSSM Higgs sector," *Eur. Phys. J.* **C28** (2003) 133–143, `arXiv:hep-ph/0212020`.

[4] S. Heinemeyer, W. Hollik, and G. Weiglein, "The Masses of the neutral CP - even Higgs bosons in the MSSM: Accurate analysis at the two loop level," *Eur. Phys. J.* **C9** (1999) 343–366, `arXiv:hep-ph/9812472`.

[5] S. Heinemeyer, W. Hollik, and G. Weiglein, "FeynHiggs: a program for the calculation of the masses of the neutral CP-even Higgs bosons in the MSSM," *Comput. Phys. Commun.* **124** (2000) 76–89, `arXiv:hep-ph/9812320`.

[6] P. Bechtle, O. Brein, S. Heinemeyer, G. Weiglein, and K. E. Williams, "HiggsBounds: Confronting Arbitrary Higgs Sectors with Exclusion Bounds from LEP and the Tevatron," *Comput. Phys. Commun.* **181** (2010) 138–167, `arXiv:0811.4169 [hep-ph]`.

[7] P. Bechtle, O. Brein, S. Heinemeyer, G. Weiglein, and K. E. Williams, "HiggsBounds 2.0.0: Confronting Neutral and Charged Higgs Sector Predictions with Exclusion Bounds from LEP and the Tevatron," *Comput. Phys. Commun.* **182** (2011) 2605–2631, `arXiv:1102.1898 [hep-ph]`.

[8] L. Zeune, "MW_Calc.".

[9] T. Hahn and M. Perez-Victoria, "Automatized one-loop calculations in four and D dimensions," *Comput. Phys. Commun.* **118** (1999) 153–165, `arXiv:hep-ph/9807565`.

[10] M. Muhlleitner, A. Djouadi, and Y. Mambrini, "SDECAY: A Fortran code for the decays of the supersymmetric particles in the MSSM," *Comput. Phys. Commun.* **168** (2005) 46–70, `arXiv:hep-ph/0311167`.

[11] M. Drees and M. M. Nojiri, "The Neutralino relic density in minimal N=1 supergravity," *Phys. Rev.* **D47** (1993) 376–408, `arXiv:hep-ph/9207234`.

[12] H. Baer and M. Brhlik, "Cosmological Relic Density from Minimal Supergravity with Implications for Collider Physics," *Phys. Rev.* **D53** (1996) 597–605, `arXiv:hep-ph/9508321`.

[13] H. Baer and M. Brhlik, "Neutralino Dark Matter in Minimal Supergravity: Direct Detection vs. Collider Searches," *Phys. Rev.* **D57** (1998) 567–577, arXiv:hep-ph/9706509.

[14] J. R. Ellis, T. Falk, K. A. Olive, and M. Schmitt, "Supersymmetric dark matter in the light of LEP 1.5," *Phys. Lett.* **B388** (1996) 97–105, arXiv:hep-ph/9607292.

[15] J. R. Ellis, T. Falk, K. A. Olive, and M. Schmitt, "Constraints on neutralino dark matter from LEP-2 and cosmology," *Phys. Lett.* **B413** (1997) 355–364, arXiv:hep-ph/9705444.

[16] J. R. Ellis, T. Falk, G. Ganis, K. A. Olive, and M. Schmitt, "Charginos and Neutralinos in the Light of Radiative Corrections: Sealing the Fate of Higgsino Dark Matter," *Phys. Rev.* **D58** (1998) 095002, arXiv:hep-ph/9801445.

[17] V. D. Barger and C. Kao, "Relic density of neutralino dark matter in supergravity models," *Phys. Rev.* **D57** (1998) 3131–3139, arXiv:hep-ph/9704403.

[18] J. R. Ellis, T. Falk, G. Ganis, and K. A. Olive, "Supersymmetric Dark Matter in the Light of LEP and the Tevatron Collider," *Phys. Rev.* **D62** (2000) 075010, arXiv:hep-ph/0004169.

[19] J. R. Ellis, T. Falk, G. Ganis, K. A. Olive, and M. Srednicki, "The CMSSM Parameter Space at Large tan beta," *Phys. Lett.* **B510** (2001) 236–246, arXiv:hep-ph/0102098.

[20] V. D. Barger and C. Kao, "Implications of New CMB Data for Neutralino Dark Matter," *Phys. Lett.* **B518** (2001) 117–122, arXiv:hep-ph/0106189.

[21] L. Roszkowski, R. Ruiz de Austri, and T. Nihei, "New cosmological and experimental constraints on the CMSSM," *JHEP* **08** (2001) 024, arXiv:hep-ph/0106334.

[22] A. Djouadi, M. Drees, and J. L. Kneur, "Constraints on the minimal supergravity model and prospects for SUSY particle production at future linear e+ e- colliders," *JHEP* **08** (2001) 055, arXiv:hep-ph/0107316.

[23] U. Chattopadhyay, A. Corsetti, and P. Nath, "Supersymmetric dark matter and Yukawa unification," *Phys. Rev.* **D66** (2002) 035003, arXiv:hep-ph/0201001.

[24] J. R. Ellis, K. A. Olive, and Y. Santoso, "Constraining supersymmetry," *New J. Phys.* **4** (2002) 32, arXiv:hep-ph/0202110.

[25] H. Baer *et al.*, "Updated constraints on the minimal supergravity model," *JHEP* **07** (2002) 050, arXiv:hep-ph/0205325.

[26] R. L. Arnowitt and B. Dutta, "Dark matter, muon g-2 and other accelerator constraints," arXiv:hep-ph/0211417.

[27] P. Z. Skands *et al.*, "SUSY Les Houches Accord: Interfacing SUSY Spectrum Calculators, Decay Packages, and Event Generators," *JHEP* **07** (2004) 036, `arXiv:hep-ph/0311123`.

[28] L. Ibánez and G. G. Ross, "SU(2)L x U(1) symmetry breaking as a radiative effect of supersymmetry breaking in GUTs," *Physics Letters B* **110** no. 3-4, (1982) 215 – 220.
`http://www.sciencedirect.com/science/article/pii/0370269382912394`.

[29] G. F. Giudice and R. Rattazzi, "Theories with gauge-mediated supersymmetry breaking," *Phys. Rept.* **322** (1999) 419–499, `arXiv:hep-ph/9801271`.

[30] L. Randall and R. Sundrum, "Out of this world supersymmetry breaking," *Nucl. Phys.* **B557** (1999) 79–118, `arXiv:hep-th/9810155`.

[31] G. F. Giudice, M. A. Luty, H. Murayama, and R. Rattazzi, "Gaugino Mass without Singlets," *JHEP* **12** (1998) 027, `arXiv:hep-ph/9810442`.

A. Parameter file "parameter.in"

In the "parameter.in" file the following has to be specified:

- softsusy, the path to the softsusy commandline interface softpoint.x
- HiggsBounds, the path to HBwithFH
- MWcalc, the path to MW_Calc_Matthias
- sdecay, the path to SDECAY's commandline interface run
- outputdir, the name of the output directory. This will be extended by a number.
- basename, the prefix for all output files.

The structure of the file has to be key = value. Since everything behind a hash sign (#) will be ignored one can write comments. Compare the example below.

```
# path to softsusy binary
softsusy = /afs/desy.de/user/.../softpoint.x

# path to HiggsBounds with FeynHiggs
HiggsBounds = /afs/desy.de/user/.../APROPOS/include/HBwithFH

# path to SuSpect
SuSpect = /afs/desy.de/user/.../SuSpect2/suspect2

# path to MW-Calc
MWcalc = /afs/desy.de/user/.../MW_Calc_Matthias/

# path to sdecay
sdecay = /afs/desy.de/user/.../SDECAY/run

# output directory basename
outputdir = run

# Basename for in- and output file
basename  = SLHA_
```

B. Input files

B.1. mGMSB

```
# Gauge Mediated Symmetry Breaking
```

```
model          = mGMSB

N_mess_min      = 1         # Messenger index
N_mess_stp_size = 2
N_mess_max      = 3

tanb_min        = 10        # Ratio of Higgs vacuum expectation values
tanb_stp_size   = 5
tanb_max        = 20

M_mess_min      = 70000     # Overall messenger scale
M_mess_stp_size = 5000
M_mess_max      = 130000

Lambda_min      = 35000     # Scale of soft SUSY breaking felt by the
Lambda_stp_size = 5000      # low-energy sector
Lambda_max      = 120000

c_grav_min      = 1         # Gravitino mass factor
c_grav_stp_size = 0
c_grav_max      = 1
```

B.2. mAMSB

```
# mAMSB

model         = mAMSB

m0_min        = 300   # Common scalar mass
m0_stp_size   = 75
m0_max        = 450

m32_min       = 40000 # Common gravitino mass
m32_stp_size  = 10000
m32_max       = 60000

tanb_min      = 10    # Ratio of Higgs vacuum expectation values
tanb_stp_size = 10
tanb_max      = 20

sign(mu)      = 1     # Sign of the bilinear Higgs term in the
                      # superpotential
```

B.3. CMSSM

```
# Constrained MSSM

model         = CMSSM

m0_min        = 100   # Common scalar mass
m0_stp_size   = 12.5
m0_max        = 250

m12_min       = 300   # Common gaugino mass
m12_stp_size  = 50
m12_max       = 600

tanb_min      = 10    # Ratio of Higgs vacuum expectation values
tanb_stp_size = 5
tanb_max      = 20

sign(mu)      = 1     # Sign of the bilinear Higgs term in the
                      # superpotential

A0_min        = 0     # Common trilinear coupling
A0_stp_size   = 0
A0_max        = 0
```