

MASTER'S THESIS IN THEORETICAL PHYSICS

# CONSTRAINING SUPERSYMMETRY USING SIMPLIFIED MODELS

Sebastian Liem

June, 2013

*Supervisors*

Dr. Oscar Stål

Dr. Abram Krislock



Stockholm  
University



## Abstract

Simplified models is a new approach to characterizing LHC supersymmetry results. In a simplified model only a few new particles and a single decay topology are introduced, which means that these models are comparably easy to constrain. The idea is that a set of such simplified models should encode as much as possible of supersymmetry phenomenology. In this Master's thesis we present SASS, which is short for Supersymmetry Analysis using Simplified modelS. SASS is an automated analysis framework we developed for reinterpreting simplified model limits to constrain more general supersymmetric models such as the constrained Minimal Supersymmetric Standard Model (cMSSM). As a validation of this framework we used 7 TeV limits on four hadronic simplified models published by the CMS to recreate the 7 TeV CMS  $\alpha_T$  exclusion contour in the cMSSM  $(m_0, m_{1/2})$  plane. We were able to match the exclusion from this dedicated cMSSM analysis without doing any Monte Carlo simulations, i.e. only using the masses, branching ratios, and cross sections. In addition this validation we also used 8 TeV simplified model limits to update the cMSSM constraints.

## Sammanfattning

Förenklade modeller är ett nytt sätt att beskriva supersymmetriresultat från LHC. I en förenklad modell introduceras endast ett fåtal nya partiklar och en enstaka sönderfallstopologi, vilket gör sådana modeller jämförelsevis enkel att sätta gränser på. Tanken är att en uppsättning av sådana förenklade modeller borde täcka in mycket av all möjlig supersymmetrifenomenologi. I detta examensarbete presenterar vi SASS vilket står för 'Supersymmetry Analysis using Simplified modelS'. SASS är ett automatiserat analysverktyg vi utvecklat för att omtolka gränser satta på förenklade modeller till gränser på mer generella supersymmetrimodeller som 'constrained Minimal Supersymmetric Standard Model' (cMSSM). För att validera detta verktyg har vi använt, utav CMS publicerade, 7 TeV gränser på fyra hadroniska förenklade modeller för att återskapa 7 TeV CMS  $\alpha_T$  uteslutningskonturen i cMSSM  $(m_0, m_{1/2})$ -planet. Vi lyckades matcha denna uteslutningskonturen från denna dedikerade cMSSM analys utan att göra några Monte Carlo-simuleringar, dvs endast genom att använda massorna, sönderfallskanaler, och tvärsnitt. Utöver denna validering så använde vi 8 TeV gränser för förenklade modeller för att uppdatera cMSSM gränserna.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Supersymmetry</b>	<b>5</b>
2.1	Motivations for Supersymmetry . . . . .	6
2.1.1	The Hierarchy Problem . . . . .	6
2.1.2	Dark Matter . . . . .	7
2.2	Breaking Supersymmetry . . . . .	8
2.3	Supersymmetric particles . . . . .	9
2.4	The pMSSM . . . . .	11
2.5	The cMSSM . . . . .	12
<b>3</b>	<b>Simplified Models</b>	<b>13</b>
3.1	Some simplified models: T1, T2, T1bbbb, and T2bb . . . . .	14
3.2	Simplified models and the LHC . . . . .	16
<b>4</b>	<b>Our analysis toolchain SASS</b>	<b>19</b>
4.1	Overview . . . . .	19
4.1.1	Programming language . . . . .	21
4.2	An example program . . . . .	21
4.3	The Model class . . . . .	24
4.4	The <code>decaytree</code> module . . . . .	25
4.5	The <code>sms</code> module . . . . .	26
4.5.1	The T1 model . . . . .	26
4.5.2	The T2 model . . . . .	27
4.5.3	b-jets: The T1bbbb and T2bb models . . . . .	28
4.5.4	lhcdatal . . . . .	29
4.6	The <code>run</code> module . . . . .	29
4.6.1	SUSY-HIT . . . . .	29
4.6.2	Prospino . . . . .	30

<b>5</b>	<b>Using Simplified Models to constrain the cMSSM</b>	<b>32</b>
5.1	The T1 model results . . . . .	36
5.2	The T2 results . . . . .	39
5.3	The T1bbbb results . . . . .	41
5.4	The T2bb results . . . . .	43
5.5	Constraining the cMSSM using 8 TeV results . . . . .	45
<b>6</b>	<b>Summary and outlook</b>	<b>48</b>
<b>A</b>	<b>Example SASS program</b>	<b>54</b>

# Chapter 1

## Introduction

The Standard Model of particle physics is a theory of all known fundamental particles and three of the four fundamental forces: electromagnetism, the weak force, and the strong force. It is a highly successful theory, especially with the apparent discovery of the Higgs boson. Yet the Standard Model is known to not be the final truth. It does not incorporate gravity, nor does it explain dark matter, dark energy, the matter-antimatter asymmetry of the universe, or neutrino oscillations. It also has a hierarchy problem, the measured low mass of the Higgs bosons is only possible in the Standard Model with excessive fine tuning.

Considerable effort is therefore spent developing what is known as ‘Beyond the Standard Model’ theories, one of which is supersymmetry. Supersymmetry as an extension to the Standard Model solves the hierarchy problem and provides a suitable dark matter candidate. Discovering supersymmetry are one of the main goals of the Large Hadron Collider (LHC).

This report documents our efforts in using LHC results on simplified models to constrain supersymmetric theories. Simplified models is a new approach to phenomenology in which one tests for specific new particles and decays rather than full particle physics theories.

To understand the reasons for using simplified models we need to begin by describing how particle physics phenomenology is usually done. Particle physics phenomenology is the field of testing theoretical particle physics theories against the results from high-energy experiments such as the LHC. To the uninitiated reader it might seem strange that there are physicists dedicated to phenomenology. Isn’t comparing theories with experiments fundamental to all physics, or indeed science? Yes, of course it is, but not every physicist has to do everything themselves. Specialisation is, after all, key in modern science. In particle physics the task of comparing theory and experiment is so onerous that phenomenology has grown to become its own field.

Supersymmetry phenomenology, or indeed any beyond the Standard Model phenomenology, involves investigating the experimental consequences of the new theory: what, if any, new particles are predicted and what are their properties? How can we constrain the parameters of a new theory? How can we compare one potential theory with others?

Traditionally to test a particle physics theory against experimental results we need to calculate what the theory predicts in an experimental situation, such as in a particle accelerator detector. In practice this means that we have to assemble, and/or develop, a sophisticated and advanced set of software tools; we call this set an analysis toolchain. The exact makeup of this toolchain of course depends on the theory we are testing, as well as the experimental situation.

We will now describe the canonical approach for such a toolchain. The first task such a canonical toolchain does is to compute all interesting theoretical predictions: the masses of new particles, their decay widths, branching ratios and production cross sections. This information is then used to do Monte Carlo simulations of the physical situation in the experiment. In the case of the LHC, we would simulate what particles are produced if we collide two proton beams with a certain centre-of-mass energy, and how these produced particles would decay. All this is done under the assumption that the theory we are testing is true.

The next step is crucial: using the toolchain, we calculate how the detector responds to this simulated physics, i.e. what data we would get from the actual instruments if our simulated physics would occur. This simulated data is then analysed in the same way as the actual experimental data. The theoretical results and the experimental results can then finally be compared in an appropriate fashion.

This canonical approach has the advantage of being conceptually straightforward, but it suffers from three major disadvantages: it is model dependent, it requires access to the experimental data, and the step of simulating the detector response requires intimate knowledge of the detector.

Model dependence means that the resulting analysis only applies to the particular theory we tested, that the whole analysis has to be redone for each theory of interest. This is particularly problematic for the situation supersymmetry phenomenology is in today, the cMSSM,<sup>1</sup> the simplest supersymmetric theory, is becoming more and more disfavoured by experiments [1], while there is no clear singular replacement for it. Instead there are many different competing alternatives.

This is not a problem in itself; one can just let every theorist test their own favourite theory. But it becomes a problem when combined with the other two major disadvantages: that this approach require intimate knowledge of the detec-

---

<sup>1</sup>cMSSM is short for the constrained Minimal Supersymmetric Standard Model

tor and access to the actual experimental data. In practical terms this means that to do the analysis as described, we need the detector simulators as developed by the detector collaborations. In the case of the LHC, the ATLAS and CMS collaborations have chosen not to make these simulators publicly available. Neither do they publish the raw data, only final results. This means that theorists not affiliated with these collaborations cannot test their theories themselves, at least not with any accuracy.

Simplified models are alternative analysis approach that has become interesting, as it tries to avoid these problems. The idea of simplified models is that of presenting experimental results for a specific model, such as the cMSSM, one instead presents results on very simple phenomenological models instead. Each of these simplified models encapsulate a specific phenomenological feature common to many different theories. This means that using simplified models would make the analysis, if not fully model independent, at least less model dependent.

We use the T1 model as defined by the CMS collaboration [2] as an example of such an simplified model. In the T1 model the existence of a new particle, the gluino, is postulated. It is the supersymmetry partner of the gluon and as such is neutral, coloured, and fermionic. The model also postulate the existence of the neutralino, another supersymmetric and neutral, but colourless, fermion. Furthermore the gluino is heavier than the neutralino, and decays into the neutralino and a  $q\bar{q}$  pair. The neutralino is a stable particle. The detector collaborations can constrain this simplified model by putting limits on the gluino pair production cross section. This limit is, of course, a function of the masses of the gluino and the neutralino.

This solves the problem of needing to know the gritty details of the detector, as theories would not directly be compared to the experimental data, but instead be compared against the various simplified model limits published by the collaborations. We have developed an analysis toolchain to automate the comparison of general supersymmetry model predictions to results for simplified models. We have named the toolchain SASS, which is short for Supersymmetric Analysis using Simplified models, and we will describe its workings and put it to use in this report.

As for the disposition of this report; we have divided it into five chapters. In chapter 2 we begin with an introduction to supersymmetry. We discuss what supersymmetry is, motivations, and the resulting phenomenology. In chapter 3 we introduce the simplified models; we define the different simplified models we use in this report and discuss the results available from the LHC collaborations. Chapter 4 documents SASS, the analysis toolchain we have developed. We detail the workings of its various components and how SASS calculates the theoretical predictions to be compared with the experimental results.



In chapter 5 we put SASS to use by constraining the cMSSM using the simplified models results from the CMS collaboration. We compare these constraints with the official cMSSM exclusion plot from the CMS collaboration as a validation test for SASS. In the sixth and final chapter we summarise our results and discuss possible future research endeavours.

# Chapter 2

## Supersymmetry

In this chapter we introduce supersymmetry as an extension of the Standard Model of particle physics. We have chosen a phenomenological discussion rather than a theoretical one, for a more complete introduction we can recommend [3], a resource which has informed much of this chapter.

Supersymmetry is a proposed extension to the Standard Model of particle physics, and it is a symmetry between fermions and bosons. More technically a supersymmetric transformation turns a bosonic state into a fermionic state and vice versa. So if we let  $Q$  be a generator of supersymmetric transformations we can schematically write

$$Q |\text{fermion}\rangle = |\text{boson}\rangle \quad Q |\text{boson}\rangle = |\text{fermion}\rangle. \quad (2.0.1)$$

To conserve spin the operator  $Q$  must be a fermionic operator with spin  $1/2$ , which means that supersymmetry is a spacetime symmetry. This is an important point as the algebra of such symmetries are limited by the Haag-Lopuszański-Sohnius relaxation [4] of Coleman-Mandula's no-go theorem [5]. The Coleman-Mandula theorem states that the only possible symmetries of the  $S$ -matrix<sup>1</sup> are the Poincaré symmetry and the internal, i.e. gauge, symmetries, this seems to forbid any extra spacetime symmetry such as supersymmetry. Haag, Lopuszański and Sohnius however relaxed this by allowing anticommuting symmetry generators in addition to commuting, which allows for supersymmetry to also be a symmetry of the  $S$ -matrix.

The algebra of supersymmetry is highly constrained by Haag-Lopuszanski-Sohnius, but we will not discuss the algebra in this report. Instead we will settle for stating some of the results.

---

<sup>1</sup>The  $S$ -matrix is the evolution operator from  $t = -\infty$  to  $t = +\infty$ . That is to say it encodes all experimentally visible particle physics. Thus a symmetry of the  $S$ -matrix is a symmetry of nature.

The single particle state of the irreducible representations of the supersymmetry algebra is called a supermultiplet. Each supermultiplet encapsulates both a fermion and a boson state, which are each others superpartners. This mean that the generator  $Q$  does not transform a fermion into any boson;  $Q$  transforms the fermion to its bosonic superpartner. And, of course, it transforms a boson to its fermionic superpartner.

$Q$  commutes with all the generators of gauge transformations, which means that the superpartners should carry the same charges, colours, *et cetera*. This implies that the particle content of the Standard Model is not enough; there simply is no way to organise the Standard Model particles into supermultiplets.

$Q$  also commutes with the mass operator so these new superpartners should have the same masses as their associate Standard Model particle. This cannot be the case, as the new particles would then have already been discovered. This implies that if supersymmetry exists it must be a broken symmetry. How it is broken still an open research question.

For the remainder of this report we will only discuss what is called the Minimal Supersymmetric Standard Model (MSSM). We can have several supersymmetries in a theory, each with its own set of generators. The MSSM realises the  $N = 1$  supersymmetry that is consistent with the Standard Model and has minimal field content and minimal interactions. At the moment there is not much point in studying anything more complicated, at least not from a phenomenological perspective.

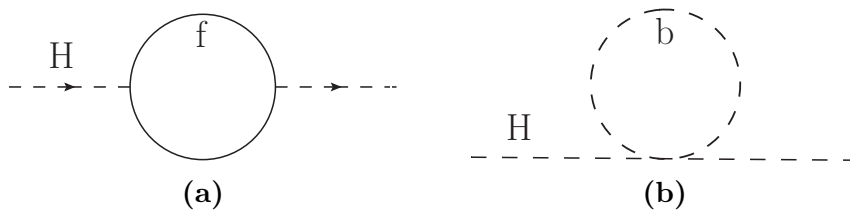
## 2.1 Motivations for Supersymmetry

### 2.1.1 The Hierarchy Problem

The hierarchy problem is a problem in theoretical particle physics which can be formulated in different ways. One way is in the form of the question, why is gravity so much weaker than the weak force? This formulation makes it easy to understand the question, but difficult to find and understand the answer. A more technical formulation tends to be easier to answer. Our technical formulation of the hierarchy problem involves the Higgs boson and its sensitivity to new physics. In the Standard Model the neutral part of the Higgs field is a complex scalar  $H$  with the potential

$$V = m_H^2 |H|^2 + \lambda |H|^4 \quad (2.1.1)$$

For the Higgs field to do its job, breaking the electroweak symmetry in the Standard Model,  $H$  needs a non-vanishing vacuum expectation value (vev) at the potential's minimum. This happens if  $\lambda > 0$  and  $m_H^2 < 0$ . The trouble is that



**Figure 2.1:** Feynman diagrams of one-loop corrections to  $m_H^2$  from (a) fermions and (b) bosons.

each particle that is coupled, in any way, to the Higgs field adds large corrections to  $m_H^2$ . Let say we have a fermion and a boson coupled to the Higgs field via the Lagrangian terms  $-\lambda_f H \bar{f} f$  and  $\lambda_b |H|^2 |b|^2$  respectively.

Figure 2.1 shows two diagrams of quantum corrections to the Higgs mass parameter  $m_H^2$  arising from this f- and b-field. These add the following corrections to  $m_H^2$

$$\Delta m_H^2 = -\frac{|\lambda_f|^2}{8\pi^2} \Lambda_{UV}^2 + \dots \quad \Delta m_H^2 = \frac{\lambda_b}{16\pi^2} \Lambda_{UV}^2 + \dots \quad (2.1.2)$$

where  $\Lambda_{UV}$  is the ultraviolet momentum cutoff which regulates the loop integral. The size of  $\Lambda_{UV}$  should be of the order of the energy scale where new physics becomes relevant. In absence of new physics this scale is so large that  $m_H^2$  becomes  $\gg 0$  which means that the Higgs field is unable to do its job of breaking the electroweak symmetry without unnatural amount of fine tuning.

But notice the difference in sign for the fermion and boson corrections to  $m_H^2$ . It would be very nice if we could systematically introduce suitable bosons for each fermion, and vice versa, to cancel these quadratic corrections. This cancellation should, of course, also extend into the higher order corrections, not just the leading order correction shown here. Supersymmetry does just this, it cancels all quadratic corrections to  $m_H^2$  in all orders of the perturbation theory. This is, of course, a strong theoretical motivation for supersymmetry.

### 2.1.2 Dark Matter

Current cosmological understanding, i.e. the  $\Lambda$ -CDM model, is that the universe consists of  $\sim 25\%$  dark matter [6], what it actually consists of is unknown. As matter in dark matter implies it is massive, i.e. the hypothetical dark matter particle is not massless. The dark in dark matter, in turn, alludes to the fact that it does not interact with photons. The dark matter particle does not carry electric charge.

The only particles in the Standard Model that have these characteristics are neutrinos. Neutrinos are however so-called *hot* dark matter, they have so small

mass that they travel at near speed of light. Its speed means that hot dark matter cannot explain the amount of structure formation observed in the universe; it is simply too fast to be gravitationally bound. What is needed is slow, or cold, dark matter, which is why CDM, being short for cold dark matter, is part of the name of the  $\Lambda$ -CDM model. But there is no cold dark matter candidate in the Standard Model. Supersymmetry, however, does provide such a candidate: the lightest neutralino [7]. It is a type of WIMP, a weakly interacting massive particle, with a mass typically in the hundreds of GeV range.

For the neutralino to be a dark matter candidate it has to be stable, otherwise it would have dissipated over the history of the universe. For it to be stable we need to impose  $R$ -parity. All Standard Model particles are assigned  $R$ -parity of 1 while all new supersymmetric particles have  $R$ -parity of  $-1$ . If  $R$ -parity is conserved that would mean that the lightest supersymmetric particle (LSP), often the lightest neutralino, is stable.

The fact that supersymmetry, originally designed to solve problems in theoretical particle physics, also could solve the dark matter problem in cosmology is a good argument for the existence of supersymmetry.

## 2.2 Breaking Supersymmetry

As we have stated earlier, if supersymmetry exists it must be a broken symmetry. If it was not broken the new particles introduced would have the same masses as their Standard Model partners. This would mean that they should have already been discovered, which, of course, they have not been.

There are however restriction on supersymmetry breaking. The couplings connecting the new particles to the Higgs field must still match up with the corresponding couplings of the Standard Model particles. Otherwise, supersymmetry would not solve the hierarchy problem, as the quadratic corrections to the Higgs mass would not be cancelled. This requires that supersymmetry must then be broken using what is known as soft supersymmetric breaking. What is meant with that is that the Lagrangian can be split into two parts

$$\mathcal{L} = \mathcal{L}_{\text{SUSY}} + \mathcal{L}_{\text{soft}} \quad (2.2.1)$$

where  $\mathcal{L}_{\text{SUSY}}$  contains all couplings and interactions while  $\mathcal{L}_{\text{soft}}$  contains the mass terms. If  $\mathcal{L}_{\text{SUSY}}$  preserves supersymmetry but  $\mathcal{L}_{\text{soft}}$  breaks it we get different masses for the new particles and still solve the hierarchy problem.

From a theoretical perspective the symmetry breaking should be spontaneous, i.e. the Lagrangian is invariant but the vacuum state is not, much like the electroweak symmetry breaking in the Standard Model. However, no one quite knows the right mechanism for supersymmetry breaking. So instead we introduce the

$\mathcal{L}_{\text{soft}}$  part explicitly. In a way this encodes our ignorance, because these explicit terms also come with extra parameters. A lot of extra parameters in fact; approximately a hundred.

This is troublesome since it makes the most general MSSM impossible to study systematically. In order to do phenomenological studies we have to somehow reduce the number of parameters. We will discuss this further later in this report

## 2.3 Supersymmetric particles

As stated earlier, supersymmetry orders the Standard Model particles and new supersymmetric particles into supermultiplets. The usual statement is that each Standard Model particle is superpartnered with a new supersymmetric particle. It is slightly more complicated than this due to the preservation of spin and degrees of freedom. A more correct statement would be that each Standard Model degree of freedom is associated with an supersymmetric degree of freedom.

To exemplify: The Standard Model fermion is a spin- $1/2$  particle which means it has two chiralities, or two degrees of freedom. Its superpartner, however, has spin-0, i.e. one degree of freedom. This simply does not add up, and to preserve spin each fermion in the Standard Model must be partnered with two superpartners, one for each chirality. Neutrinos still only have one superpartner, as there are no right-handed neutrinos in the Standard Model.

The supersymmetric Higgs sector is more complicated than just introducing a superpartner. The reason for this is that having one Higgs supermultiplet leads to a gauge anomaly for the electroweak gauge symmetry. A gauge anomaly, usually, is a gauge breaking Feynman diagram that is not cancelled by another diagram. This means the theory is inconsistent as the Lagrangian is still manifestly gauge invariant. This can, however, be resolved by having two Higgs supermultiplets, arranged so that they cancel each others gauge anomaly out.

In table 2.1 we list the new particles of the MSSM. The superpartners of fermions, sfermions, are named by prepending an s to the fermion's name. The superpartners of gauge bosons, gauginos, are named by appending an -ino, giving us gluino, higgsino, wino, *et cetera*. Also note that the symbol for the superpartner is the symbol for the Standard Model particle with a tilde over, e.g. q, quark, becomes  $\tilde{q}$ , squark. As for the properties of these new particles; besides spin and mass the superpartners all carry the same quantum numbers as their Standard Model counterparts.

In the table we also see that the gauge eigenstates are not always the mass eigenstate for many of these particles. For example, we have the charginos which are a linear combination of the charged wino and charged higgsinos. There are two of them labelled  $\tilde{\chi}_1^\pm$  and  $\tilde{\chi}_2^\pm$ , the latter being the heavier one. There are also

Name	Spin	Gauge Eigenstate	Mass Eigenstate
Higgs bosons	0	$H_u^0, H_d^0, H_u^+, H_d^-$	$h^0, H^0, A^0, H^\pm$
Squarks	0	$\tilde{u}_L, \tilde{u}_R, \tilde{d}_L, \tilde{d}_R$	– ” –
		$\tilde{s}_L, \tilde{s}_R, \tilde{c}_L, \tilde{c}_R$	– ” –
		$\tilde{t}_L, \tilde{t}_R, \tilde{b}_L, \tilde{b}_R$	$\tilde{t}_1, \tilde{t}_2, \tilde{b}_1, \tilde{b}_2$
Sleptons	0	$\tilde{e}_L, \tilde{e}_R, \tilde{\nu}_e$	– ” –
		$\tilde{\mu}_L, \tilde{\mu}_R, \tilde{\nu}_\mu$	– ” –
		$\tilde{\tau}_L, \tilde{\tau}_R, \tilde{\nu}_\tau$	$\tilde{\tau}_1, \tilde{\tau}_2, \tilde{\nu}_\tau$
Neutralinos	1/2	$\tilde{W}^0, \tilde{H}_u^0, \tilde{H}_d^0, \tilde{B}$	$\tilde{\chi}_1^0, \tilde{\chi}_2^0, \tilde{\chi}_3^0, \tilde{\chi}_4^0$
Charginos	1/2	$\tilde{W}^\pm, \tilde{H}_u^\pm, \tilde{H}_d^\pm$	$\tilde{\chi}_1^\pm, \tilde{\chi}_2^\pm$
Gluino	1/2	$\tilde{g}$	– ” –

**Table 2.1:** The new particle content of the MSSM.

the neutral mass eigenstates, called neutralinos, which are mixtures of the bino, zino and the neutral higgsinos. There are four of them labelled  $\tilde{\chi}_1^0, \tilde{\chi}_2^0, \tilde{\chi}_3^0$  and  $\tilde{\chi}_4^0$ , in order of increasing mass. The third generation sfermions and the Higgs bosons also have a mass eigenstate different from the gauge eigenstates due to mixing effects.

## 2.4 The pMSSM

As we wrote earlier, the unconstrained MSSM has around a hundred parameters in addition to the nineteen of the Standard Model. This is far too many for doing any kind of phenomenological study, for that we need to make additional assumptions in order to reduce the number of parameters.

The phenomenological MSSM (pMSSM) is one specific set of such assumptions. It is called phenomenological because it is solely motivated by phenomenological reasons. The pMSSM does not care why supersymmetry is broken, only that it is. To get the pMSSM we make three distinct assumptions motivated from different experimental constraints:

- i All soft supersymmetry breaking parameters are real.
- ii The matrices for sfermion masses and trilinear couplings are diagonal in flavour space.
- iii First and second sfermion generation universality at low energies.

The first assumption means that we have no additional sources of CP-violation. The second eliminates any tree-level flavour changing neutral currents. The third assumption means that the masses of the first and second generation sfermions are the same. In the end we are left with 19 parameters, which we list in table 2.2.

Parameter	Meaning
$\tan \beta$	ratio of the vevs of the two Higgs doublets.
$M_A$	CP-odd Higgs boson mass parameter
$\mu$	Higgs(ino) mass parameter
$M_1, M_2, M_3$	bino, wino and gluino mass parameters.
$A_t, A_b, A_\tau$	third generation trilinear couplings.
$m_{\tilde{q}}, m_{\tilde{u}_R}, m_{\tilde{d}_R}, m_{\tilde{l}}, m_{\tilde{e}_R}$	first/second generation sfermion masses.
$m_{\tilde{Q}}, m_{\tilde{t}_R}, m_{\tilde{b}_R}, m_{\tilde{L}}, m_{\tilde{\tau}_R}$	third generation sfermion masses.

**Table 2.2:** The parameters of the pMSSM.



## 2.5 The cMSSM

The constrained MSSM (cMSSM) differs from the pMSSM in that it has a scenario for how supersymmetry breaking is mediated. The soft supersymmetry breaking is mediated through a hidden sector, which is a set of fields which interact with out visible sector only via gravity. The cMSSM is therefore often also called mSUGRA, which stands for minimal supergravity. In effect we end up with four different unification and universality assumptions

- i Gauge coupling unification.
- ii Unification of all gaugino masses.
- iii Universal scalar, i.e. sfermion and Higgs boson, masses
- iv Universal trilinear couplings.

These assumptions are all boundary conditions set at the grand unification scale. To get the low energy physics interesting for phenomenology we need to use the appropriate renormalization group equations.

In the end, we are left with just four continuous parameters and the sign of the fifth; see table 2.3. This is obviously a great simplification over the hundred in the unconstrained MSSM or even the 19 parameters of the pMSSM. This is why the cMSSM in many ways has been the default supersymmetric theory to study.

Parameter	Meaning
$\tan \beta$	ratio of the vevs of the two Higgs doublets.
$m_{1/2}$	universal gaugino mass.
$m_0$	universal scalar mass.
$A_0$	universal trilinear couplings.
$\text{sign}(\mu)$	sign of the Higgs(ino) mass parameter.

**Table 2.3:** The parameters of cMSSM.

# Chapter 3

## Simplified Models

In this chapter we discuss simplified models. We begin with the basic idea of a simplified model, then describe the specific models we use in this report. We end by discussing simplified model results from the LHC collaborations.

The simplified model is a rather new idea. It was developed as an approach to describe, in a somewhat model independent way, the LHC new physics results [8, 9, 10].

A simplified model is a ‘model’ where the existence of only a few new particles are postulated. As an example: in the model we call T1 two particles are postulated to exist, the ‘gluino’, and the ‘neutralino’. the decay topology for the new particles is also specified in a simplified model. In our example the neutralino is stable while the gluino always decays into a  $q\bar{q}$  pair and a neutralino.

Such a simple model only has few parameters; the T1 simplified model has two: the masses of the gluino and the neutralino. Not having that many parameters makes it easy to put limits on simplified models, at least in contrast to the pMSSM, or even the cMSSM.

A single simplified model is not, however, very interesting in itself; it is most definitely not a good model of nature. An entire set of different simplified models is another matter. With each model characterizing a single phenomenological ‘feature’, a set of them could potentially cover much, if not all, of interesting supersymmetry phenomenology.

One analogue that might be illustrative is the basis of a vector space. A set of simplified models could be thought of as the basis of the signature space of supersymmetric theories. So to test a supersymmetric theory we would first project it onto this set of simplified models, i.e. figure out how much weight each phenomenological ‘feature’ has in the full theory. Then we would use that to calculate the appropriate theoretical prediction to compare with the simplified model limits.

### 3.1 Some simplified models: T1, T2, T1bbbb, and T2bb

We have implemented and used four different simplified models in this work: the T1, T2, T1bbbb and T2bb simplified models. All of these models are hadronic; i.e. they involve squarks, gluinos, and quarks, but not any leptons or sleptons.

Each simplified model is labelled according to the same scheme as defined by the CMS collaboration in [2]. The labels are in the form  $TN_x$  where the T, as in topology, is constant. The N is an integer between one and six. Odd N specify gluino pair production and even N specify squark-antisquark production. The actual value of N specify the decay topology: 1,2 for direct decay, 3,4 for one cascade decay and one direct decay and 5,6 for two cascade decays. The x is a text string which is used if further characterization of the model is necessary.

The T1 simplified model is the collision of two protons which produce two gluinos. These gluinos then each decay directly into a  $q\bar{q}$  pair and a neutralino. Illustration of the T1 simplified model is shown in figure 3.1a. The experimental signature for the T1 model is then four jets<sup>1</sup> from the quarks and some missing energy as the neutralino cannot be detected by LHC detectors.

The T2 simplified model involves squarks instead of gluinos. Two squarks are produced which then each decay into a quark and a neutralino. See figure 3.1b for the schematic Feynman diagram. The experimental signature is thus two jets and missing energy.

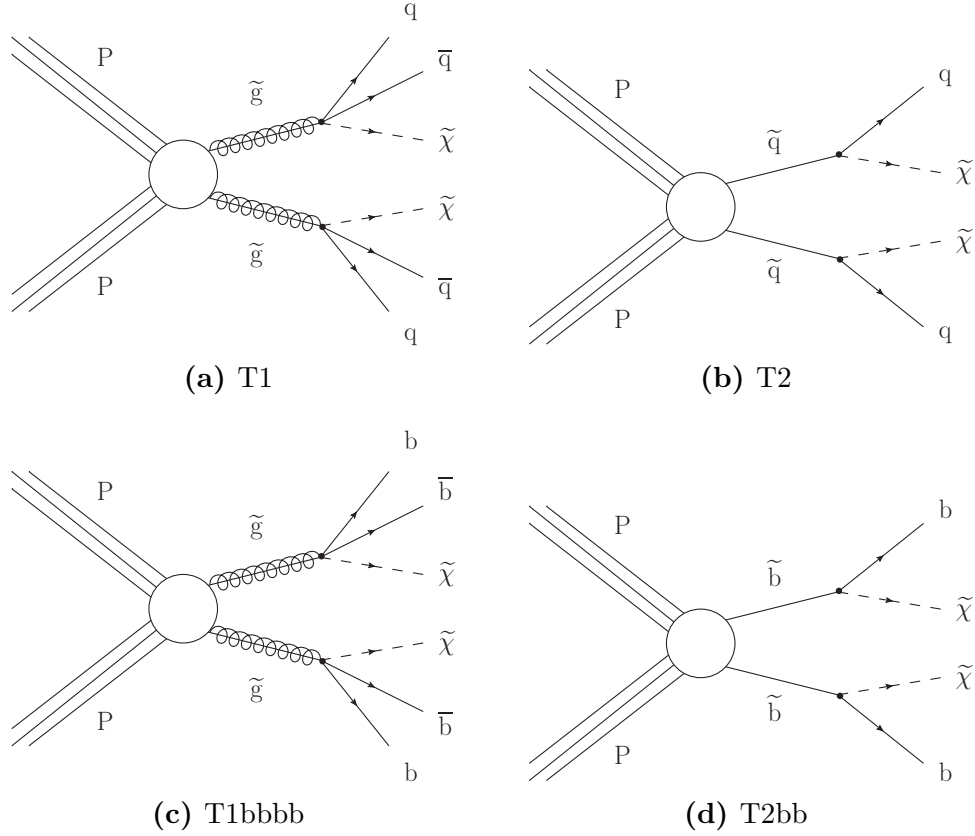
We want to emphasise the complementarity of the T1 and T2 simplified models. Both models demand that there is no intermediate particle between the initial squark/gluino and the neutralino. If the squark masses are lower than the gluino mass,  $m_{\tilde{q}} < m_{\tilde{g}}$ , gluinos can decay via squarks. Which results in a low efficiency for the T1 model. The T2 model, which has squarks as initial particles, is on the other hand still efficient. If the mass hierarchy is reversed, the reverse is true: the T1 model is useful while T2 is less so.

The quarks which the gluino and squarks decay into in the T1 and T2 model are first or second generation quarks. For bottom quarks we have two variant models: the T1bbbb and the T2bb simplified models. The T1bbbb model is very similar to the T1 model; the difference is that the quarks produced are all bottom quarks instead of the first and second generation squarks of the T1 model. See figure 3.1c for the Feynman diagram. The experimental signature is therefore also slightly different; four b-jets, jets arising from bottom quarks, and missing energy. In the

---

<sup>1</sup>Jets are narrow cones of particles produced when quarks or gluons are hadronised. Coloured particles, such as quarks and gluons, cannot exist individually. So when a quark is produced other quarks are created from the vacuum to combine into uncoloured hadrons. A shower of particles, or a jet, is thus produced.

T2bb simplified model sbottom squarks are produced in the proton-proton collision instead of the first and second generation squarks as in the T2 model. These sbottoms then each decay into a bottom quark and a neutralino, see figure 3.1d for the Feynman diagram. Again the experimental signature is slightly different from the T2 model: two b-jets, instead of two regular jets, and missing energy.

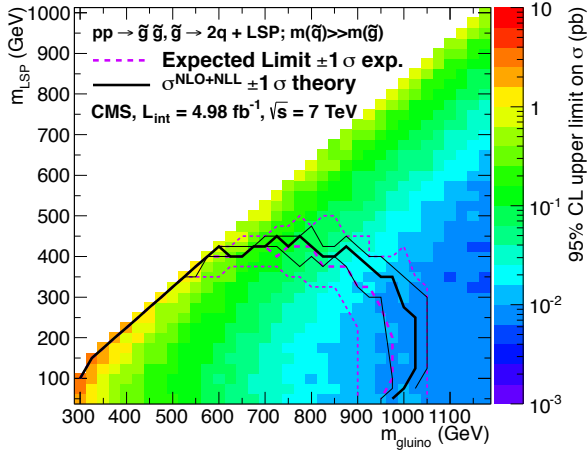


**Figure 3.1:** Schematic Feynman diagrams for the four simplified models discussed in this report. Diagrams (a) and (c) both involve gluino pair production followed by the gluinos decaying into a  $q\bar{q}$  pair and a neutralino. In (a) the produced quark jets are all first and second generation, while in (c) they are bottoms. Diagrams (b) and (d) shows squark pair production, where each squark decays into a quark and a neutralino. In (b) it is all first and second generation squarks and quarks, while in (d) sbottoms are produced which decay exclusively into bottoms.

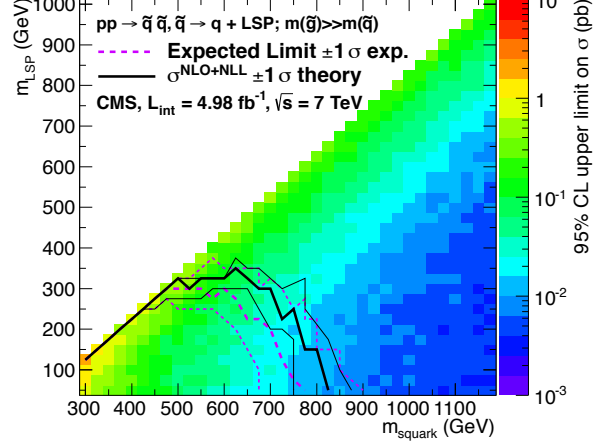
## 3.2 Simplified models and the LHC

Both the ATLAS and the CMS collaborations have started publishing limits on various simplified models [2, 11, 12, 13, 14]. To clarify how these results take form we have included the CMS results on the T1, T2, T1bbbb, and T2bb models from [15], see figure 3.2. These results are upper limits at 95% confidence level on the production cross section parametrised by either the squark or gluino mass and the neutralino, also known as the LSP, mass. This is from a data sample corresponding to a integrated luminosity of  $L_{\text{int}} = 4.98 \text{ fb}^{-1}$  from collisions with a centre-of-mass energy of  $\sqrt{s} = 7 \text{ TeV}$ . Also indicated are the observed (solid line) and expected (dashed line) excluded region assuming NLO+NLL cross sections for the initial particle pair production.

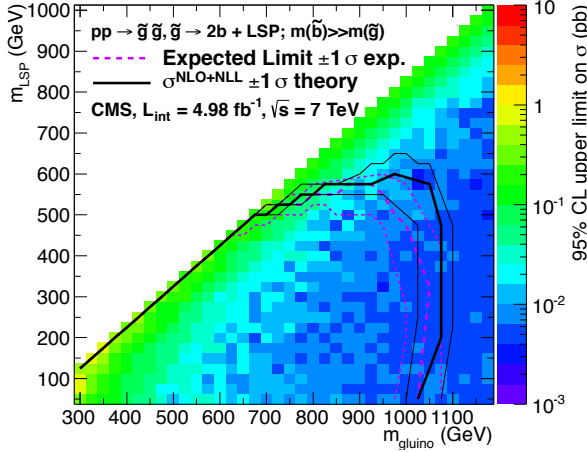
We also have included the corresponding 8 TeV simplified models results [16], see figure 3.3. The data sample used in that analysis corresponds to a integrated luminosity of  $L_{\text{int}} = 11.7 \text{ fb}^{-1}$  at a centre-of-mass energy  $\sqrt{s} = 8 \text{ TeV}$ . An interesting thing to note in the 8 TeV results is that the observed excluded regions are, for all models, lower than the expected. In fact we see that only in the T1bbbb and T2bb models the observed exclusion contours have increased in any significant way compared to the 7 TeV results.



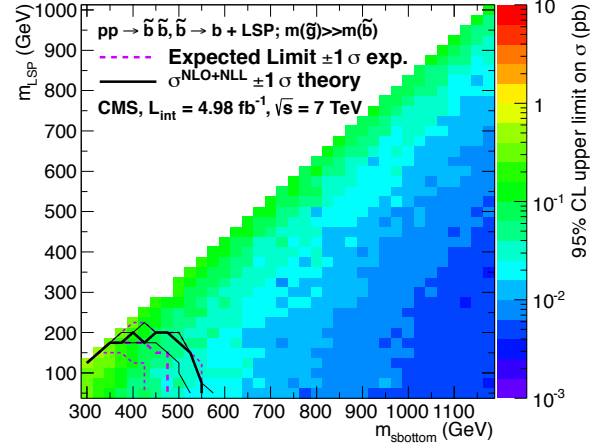
(a) T1



(b) T2

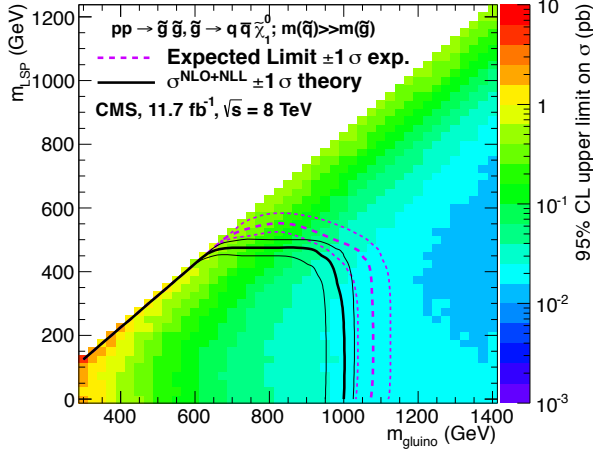


(c) T1bbbb

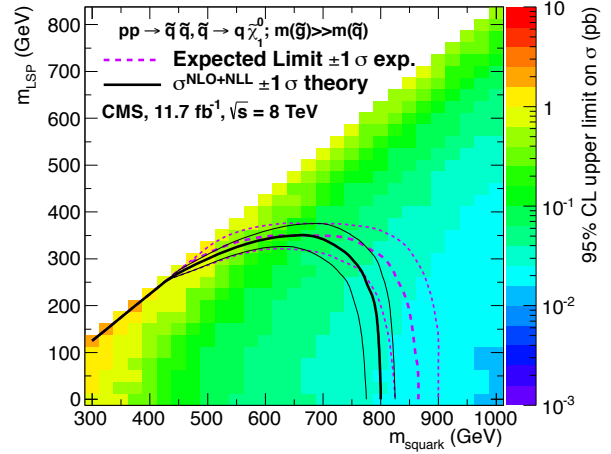


(d) T2bb

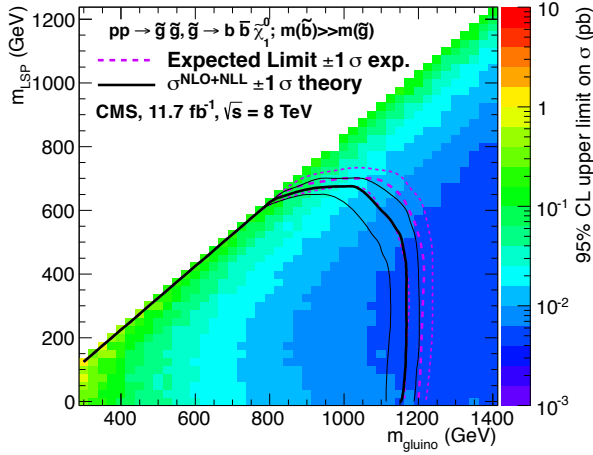
**Figure 3.2:** Upper limits at 95% confidence level from CMS [15] on the production cross section for the four simplified models we have used in this report. The data sample corresponds to an integrated luminosity of  $L_{\text{int}} = 4.98 \text{ fb}^{-1}$  from the CMS detector with  $pp$  collisions at a centre-of-mass energy of  $\sqrt{s} = 7 \text{ TeV}$ . The limits are given as a function of the lightest neutralino (LSP) mass,  $m_{\text{LSP}}$ , and either the squark mass,  $m_{\text{squark}}$ , or gluino mass,  $m_{\text{gluino}}$ .



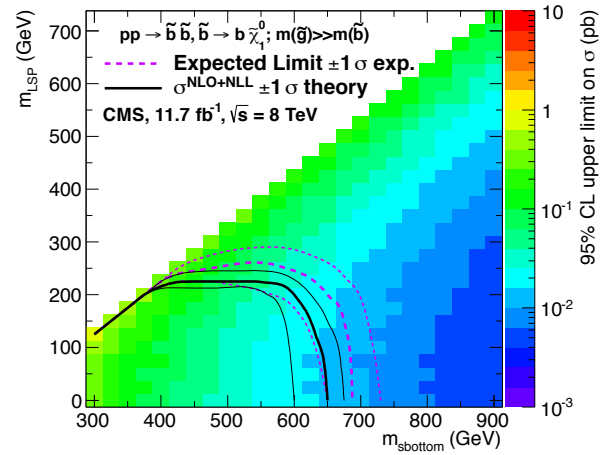
(a) T1



(b) T2



(c) T1bbbb



(d) T2bb

**Figure 3.3:** Upper limits at 95% confidence level from CMS [16] on the production cross section for the four simplified models we have used in this report. The data sample corresponds to an integrated luminosity of  $L_{\text{int}} = 11.7 \text{ fb}^{-1}$  from the CMS detector with  $pp$  collisions at a centre-of-mass energy of  $\sqrt{s} = 8 \text{ TeV}$ . The limits are given as a function of the lightest neutralino (LSP) mass,  $m_{\text{LSP}}$ , and either the squark mass,  $m_{\text{squark}}$ , or gluino mass,  $m_{\text{gluino}}$ .

# Chapter 4

## Our analysis toolchain SASS

In this chapter we describe, in detail, the analysis toolchain we have developed and used in this report. Its name is SASS which stands for Supersymmetry Analysis using Simplified modelS. The purpose of this framework is to automatically apply simplified model results from LHC to a specific parameter point in either the cMSSM or the pMSSM. To fulfil this purpose the toolchain has to be able, for a specified parameter point, to compute theoretical predictions for the simplified models and then access the LHC results for comparison.

Our description will begin with an overview of the different parts of SASS and then continue by going through an example program, which illustrates how SASS can be used. Then we will describe each component of SASS in more detail.

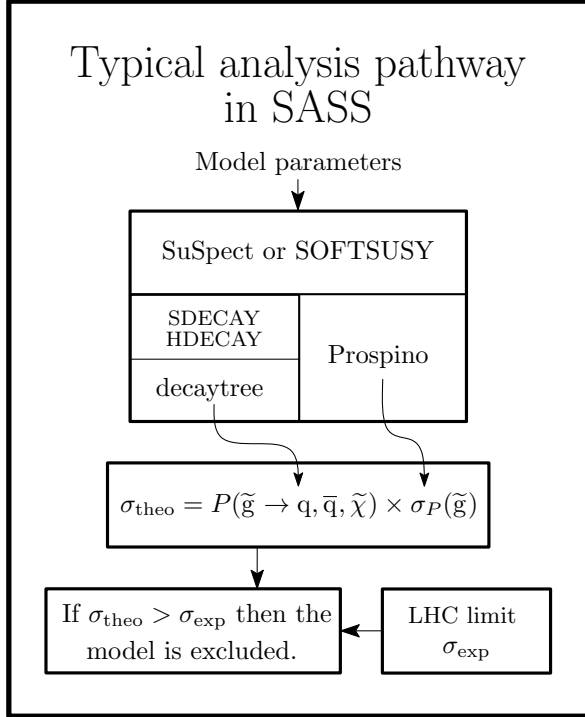
For the reader more interested in the physics rather than technical documentation we recommend skipping most of this chapter; the interesting parts are the overview in 4.1 and section 4.5. In which describes how we calculated the theoretical predictions.

### 4.1 Overview

The experimental results from the LHC [15, 16] comes as upper limits on production cross sections; see section 3.2 for more details. To use these limits we need to calculate an appropriate theoretical value to compare with the limits. In SASS this theoretical values is the pair production cross section multiplied by the proportion of decays that match the experimental signature of the particular simplified model. For SASS to be able to compute this it also has to calculate the mass spectrum, decays, decay final states, and pair production cross sections.

SASS is divided into several modules, each responsible for one of these computational tasks, bonded together by an overall framework. Figure 4.1 is a schematic overview of a typical use of the framework using its most important modules.





**Figure 4.1:** Flowchart of a typical analysis process in SASS. First the mass spectrum is calculated (SuSpect) then the decays (SDECAY & HDECAY) and the production cross section,  $\sigma_P(\tilde{g})$  (Prospino). decaytree calculates  $P(\tilde{g} \rightarrow q, \bar{q}, \tilde{\chi})$ , the proportion of sparticles that will decay so it matches the simplified model of interest. Here that model is exemplified by the T1 model. These then combined into a theoretical prediction  $\sigma_{\text{theo}}$  which is compared with the experimental result  $\sigma_{\text{exp}}$ .

In the schematic we see that most of these modules are adaptations of publicly available codes. We use SUSY-HIT 1.3 [17] to calculate particles masses and their decays. SUSY-HIT is itself a software bundle consisting of SuSpect 2.41 [18] for calculating sparticle masses, SDECAY 1.3b [19] for sparticle decays and HDECAY 3.4 [20] for Higgs boson decays. We also use Prospino 2.1 [21, 22] to calculate the various pair production cross sections we need. To interoperate with, and between, these different codes we make use of the Supersymmetry Les Houches Accord [23] file format, which is a standardised file format for storing information on particle masses, decays, mixing, model parameters and so on. The experimental limits on simplified models from the CMS collaboration are stored in root files. Therefore we had to use ROOT 5.34/03 [24] and PyROOT to access these experimental results. Section 4.5.4 holds more details.

`decaytree` is the one major component developed entirely by us. The purpose of it is to calculate the probability for a particle to follow a certain decay chain, or a set of decay chains. With it we can calculate the probability for an initial particle, such as a gluino, to decay in such a way that it produces a particular final state, such as two b-quarks and a neutralino.

`decaytree` does this by first constructing a tree of all possible decay paths for an initial particle. Then it calculates all combination of paths through the tree. This gives us a list of possible final states, each final state being the result of one

particular combination of decay paths. More importantly, each final state in this list is accompanied by the probability of each final state.

`decaytree` then also provides ways to filter the final states, e.g. to only look at final states matching the signature of two jets and missing energy. This makes it easy to compute the probability of a particle decaying into a specified set of final states. `decaytree` is described more in detail in section 4.4.

### 4.1.1 Programming language

Our programming language of choice for SASS is Python (version 2.7.3). Python’s strong abstractions, dynamical typing, and freely available code libraries lends Python to rapid prototyping and makes it easy to do structural changes in the code base. This is important in a framework, as it means it can be adapted to the, often changing, needs of the user.

The main downside of Python, compared to C or Fortran, is its speed. We have alleviated this by using Fortran for the more CPU intensive components. Or rather, the publicly available codes we have used for these intensive tasks all use Fortran.

## 4.2 An example program

Before describing the different parts of SASS we begin with an example of those parts in use. We will go through, step-by-step, a short program that tests a cMSSM parameter point against the T1 simplified model. The full program can be found in appendix A. This program illustrates the use of the toolchain while introducing some of the major components of SASS.

The program is all in one short file and it begins with the Python’s `import` statement.

```
import Model, sms, run
```

This loads three core modules of SASS. `Model` defines the `Model` object which is a central part of SASS; it is how a model, i.e. a theory with specific parameter values, is represented. The `sms` module collects all code related to simplified models and `run` defines a number of functions for common tasks performed when testing models.

The first interesting statement in the program creates a `Model` object:

```
# Create a model
m = Model.Model(name='example', directory='./')
```

This object stores all information we will compute about the model we are testing. We named the object ‘example’ in want of a better name, and specify where the

object is to create its directory. It will create, if it does not exist, the directory `./example/` in which it will store all model related files.

One of these model related files stored in the model directory is `Model.log` to which we write log messages. Either explicitly as here

```
# There is builtin logging to a model specific file.
m.log.write('This is a example')
```

or implicit by evoking other SASS functions; most of them log their actions to this file.

One example of the common tasks codified by the `run` module is the `setCMSSMPoint()`:

```
# Set our desired cMSSM point
run.setCMSSMPoint(model=m,
                  m0=200,
                  m12=300,
                  A0=0,
                  tanb=10,
                  musign=+1)
```

`setCMSSMPoint()` specifies that our `Model` object is a `cMSSM` model and sets its five model parameters to numerical values. Another function, `runSUSYHIT()`, can then run SUSY-HIT to calculate particle masses and decays.

```
# calculate particle masses and their decays
run.runSUSYHIT(model=m,
               workdir=<path to SUSY-HIT installation>)
```

The function sets up the input SLHA file SUSY-HIT needs, runs the SUSY-HIT binary, and then reads in and stores the results into the `Model` object. The argument `workdir` is used to specify the path to the SUSY-HIT installation. `runProspino()` is very similar in function as `runSUSYHIT()`.

```
# Calculate the cross section.
run.runProspino(model=m, binary='T1.run',
               workdir=<path to Prospino installation>)
```

The function prepares input files, executes the external program, and then store the results. The binary executed is a Prospino program which calculates the production cross sections. Using the `binary` function argument, one can specify which Prospino binary to use. Here we are executing `T1.run` which calculates the gluino pair production cross section, which is what is needed for the T1 simplified model. When implementing a new simplified model in SASS a Prospino program which computes the appropriate cross sections must be implemented. See the Prospino documentation [21, 22] for how to do that. The argument `workdir` specifies the path to the Prospino installation.

For each simplified model we have a different Python module, here `sms.T1`, which defines the function `theoSigma()`.

```
# Calculate the theory predicted cross section
theo = sms.T1.theoSigma(model=m)
```

The function calculates the simplified model cross section predicted by the model. Another function defined by each simplified model module is `expSigma()`

```
# Retrieve the experimental value.
exp = sms.T1.expSigma(model=m,
                      datafile='sample.root')
```

which returns the experimental result relevant for the model. The argument `datafile` specifies the location of the root file containing the experimental result. With both the experimental result and the theoretical prediction computed we can now compare them.

```
# Compare theory with experiment.
if theo > exp:
    m.log.write('Excluded by T1.')
else:
    m.log.write('Not excluded by T1.')
```

Here we did nothing fancy with the result; we just log the observation that one number is bigger than the other.

The last thing our program does is store the results on disk so we can use them at a later date without having to redoing all the calculations.

```
# Store the results
m.sigmas['T1'] = (theo, exp)
m.save()
m.writeSLHAFile()
```

The first statement just stores the theoretical and experimental values in our `Model` object. The statement `m.save()` serialises, or pickles in Python parlance, and saves the `Model` object to a file 'Model.pickle' in the model directory. The object can then be loaded, or unpickled, from that file when needed. The last statement `m.writeSLHAFile()` writes an SLHA file to the model directory with all the information that fits into the SLHA format. That is to say, a large subset, but still a subset, of the information contained in the `Model` object. The major advantage of the SLHA file is that it is human-readable, which the pickled object is decidedly not.

With this our example program is finished. Running it takes 56 seconds on a high-end desktop computer,<sup>1</sup> with 50 of these seconds being spent by Prospino

---

<sup>1</sup>The high-end desktop here being an Mac Pro from 2009 with a 2.26 GHz Intel Xeon CPU.

to calculate the production cross section. A more realistic program would test hundreds, if not thousands, of parameter points, all of them in parallel using Python's `multiprocessing` module. However the code would still not be much longer. The final program we used to produce the T1 simplified model results in figure 5.4. is 60 lines, just twice that of our example.

### 4.3 The Model class

In the heart of SASS lies the `Model` class. An object of this class manages all state related to a model<sup>2</sup>. Here, state means SLHA blocks (masses, decays etc.), production cross sections, `decaytree`'s final states, intermediate results from expensive computations, and the final results for each of the simplified models.

When an `Model` object is initialised the user has to supply a name for the model and a file directory. The object then creates a model directory, named after the model name and placed under the specified file directory. In this model directory all files related to the model are stored.

One of these files stored in the model directory is called `Model.log`. This file is used by various SASS components to log messages related to the model. The user can also explicitly write messages using the method `log.write()`.

The `Model` class defines several methods for accessing commonly used data: `mass()` returns the mass of specified particle, `crossec()` the pair production cross section and `lsp()` tells you which supersymmetric particle is the lightest one.

The `Model` class handles its own persistency, by which we mean that it defines methods that allow the object, and the data store within, to persist after the program that created it has terminated. The `Model` class defines two different sets of methods for this, two different approaches. The first one uses Python's `Pickle` module to save the object as it is, albeit in serialised form, to a file. The method is `save()` and, if not specified otherwise, it saves the object to the file `Model.pickle` in the model directory.

The inverse operation `load()` is also defined; by default it loads the object in `Model.pickle` and writes over all attributes in the calling object. The `Model` module also defines a function `loadModel()` that encapsulates the common operation of creating a empty `Model` object and then loading a serialised object from a `Model.pickle`.

The other approach to information persistence is SLHA files. We have used the PySLHA [25] library to handle SLHA files. The method `writeSLHAFile()` writes the information in the `Model` object to, by default, `Model.slha` in the

---

<sup>2</sup>Here, model means a theory with specific parameter values, e.g. one model is cMSSM with  $m_0 = 200$  GeV,  $m_{1/2} = 300$  GeV,  $A_0 = 0$ ,  $\tan\beta = 10$  and  $\mu > 0$ .

model directory. We also have the inverse `readSLHAFile()` which, as the name suggests, reads a SLHA file and stores the blocks and decays in the `Model` object. It uses two methods to do this: `readBlock()` and `readDecay()` each responsible for blocks and decays. These two methods can also be used to load one specific block or decay.

## 4.4 The decaytree module

`decaytree` is a Python module we have written to calculate all possible final states, and their probabilities, given an initial particle. The module lets the initial particle decay, and then lets all of its daughter particles decay further. This is then repeated until there are only stable particles left, i.e. what we call a final state.

Of course, in general, each unstable particle can decay in several different ways, so at each step the list of all possible final states grows. Let us exemplify; we have a particle A that decays into B and C or D and E. So far we then have two final states if B, C, D and E is all stable. But say that B has four possible decays, C has three, and D two, while E is still stable. Then we have  $4 \times 3 + 2 \times 1 = 14$  different final states.

The tables of all possible decays for each particle comes with the `Model` object. These tables also define if a particle is stable or not; if a particle has an entry in the decay table it is evidently unstable. If it has not, then the particle is treated as stable.

`finalStatesList()` is the main function of `decaytree` and it is the function which, as the name states, computes the list of all possible final states. However, this list does not only contain all possible final states but also has the probability of each final state.

`decaytree` also defines `selectFinalStates()` which allows us to filter the final states list according to a set of conditions. With it we can, for example, select all final states with a certain number of jets and at least one neutralino.

Another function, `signatureProbability()`, builds on that capability. It too takes a set of conditions and filters the final states list, but then it sums together all the probabilities of matching final states. In other words it calculates the probability of the initial particle to decay into a particular signature. So, for instance, we can calculate the probability for two gluinos to decay into four b-jets and two neutralinos as needed by for the T1bbb simplified model.

## 4.5 The sms module

The `sms` module collects all modules related to simplified models; `lhcddata` is one of them and it provides access to the root files with the LHC results. Also we have organised the code so that the implementation of each simplified model is in its own module. Each module implementing a simplified model must define two functions: `expSigma()`, which returns the experimental cross section limit for that specific simplified model, and `theoSigma()`, which calculates the theoretical prediction with which to compare the experimental result with.

### 4.5.1 The T1 model

The T1 simplified model is straightforward: two gluinos are produced which then each decay into two quarks and a neutralino. The two functions to calculate the experimental and theoretical values are also similarly straightforward. `expSigma()` simply uses the `lhcddata` module to extract the experimental value given the gluino and neutralino masses.

As we have discussed in the overview, to calculate the theoretical prediction, the responsibility of the `theoSigma()` function, we need two numbers. One of these is the predicted pair production cross section; in this case we need the  $\tilde{g}\tilde{g}$  one. `theoSigma()` assumes that this cross section is already calculated using Prospino and appropriately stored in the `Model` object. See section 4.6.2 for the details on that.

The second number `theoSigma()` calculates itself; the signature probability  $P_{\text{sig}}$  which is the probability for a gluino to decay into a final state compatible with the experimental signature of the T1 simplified model. To do this we need all possible final states for a decaying gluino, for this we use our `decaytree` module.

When calculating this final states list we treat the squarks as stable, which means we can later filter those out any final states that include squarks. We want to do this because one of the assumptions of the T1 model is that the squarks are heavier than the gluino, i.e. no cascade decays. This also avoids overlap with the T2 model.

After we have this final states list we compute the signature probability  $P_{\text{sig}}$  by applying the T1 conditions

$$n_{\text{jet}} = 4 \quad n_{\tilde{\chi}} \geq 2 \quad n_{\tilde{q}} = 0 \quad (4.5.1)$$

The greater than or equal sign deserves an explanation. The experimental signature of the  $\tilde{\chi}$  is missing energy which makes it impossible to determine the exact number of neutralinos.

To sum it all up, our theoretical predicted cross section, as calculated by `theoSigma()`, can be expressed as

$$\sigma_{\text{theo}} = \sigma_{\tilde{g}\tilde{g}} \times P(n_{\text{jet}} = 4, n_{\tilde{\chi}} \geq 2, n_{\tilde{q}} = 0 | 2 \tilde{g} \text{ decaying}) \quad (4.5.2)$$

The astute reader may notice that this calculation does not account for the kinematical factor. To factor in kinematics we would have to do a full Monte Carlo simulation, something we have avoided on purpose as Monte Carlo simulations are computationally expensive.

All is not lost however, it is important to note that the kinematical factor can only lower the predicted cross section. This means that the  $\sigma_{\text{theo}} > \sigma_{\text{exp}}$  condition does not strictly imply that the model is excluded. We can however note that a model not excluded by the limits, i.e.  $\sigma_{\text{theo}} < \sigma_{\text{exp}}$ , is, in fact, even less excluded if we factor in the kinematics.

### 4.5.2 The T2 model

The T2 simplified model is on the surface as simple as the T1 model; two squarks are produced which then each decay into a quark and a neutralino. The complexity enter the picture when we consider what is meant with a squark. The experimental limits on T2 [15, 16] are parametrised by squark mass and neutralino mass. The squark can be a first or second generation squark, which means we have eight different particles all with different masses and production cross sections. In the T2 model's `expSigma()`, the function responsible for extracting the experimental limit, we have used the mean mass of the squarks as the ‘squark’ mass, as this is the only reasonable approach.

In the calculation of the theoretical prediction, i.e. `theoSigma()`, the approach is similar to the T1 model calculation. The signature, or conditions, on for the T2 model are

$$n_{\text{jet}} = 2 \quad n_{\tilde{\chi}} \geq 2 \quad n_{\tilde{g}} = 0 \quad (4.5.3)$$

The question becomes on which list of final states to apply these conditions to. There is no decay table for the ‘squark’ as the different squarks may decay differently. The solution is to apply the condition on each combination of two squarks in turn.

These probabilities combined with the pair production cross section for each combination of squarks can then be combined to calculate our theoretical prediction. It can be succinctly stated as



$$\sigma_{\text{theo}} = \sum_i \sum_{j \geq i} \sigma_{\tilde{q}_i, \tilde{q}_j} \times P(n_{\text{jet}} = 2, n_{\tilde{\chi}} \geq 2, n_{\tilde{g}} = 0 \mid \tilde{q}_i, \tilde{q}_j \text{ decaying}) \quad (4.5.4)$$

where the  $i$  and  $j$  sums over all first and second generation squarks and their antiparticles, and the  $j \geq i$  condition is there to avoid double counting. Please note that, as explained above, kinematical factors are not taken into account.

We also need to define the signature probability of the T2 model. We do this by weighting the signature probability for each quark pair by their production cross sections. This easiest done by dividing  $\sigma_{\text{theo}}$  with the sum of all production cross sections  $\sigma_{\text{tot}}$ . Thus, we have

$$P_{\text{sig}} = \frac{\sigma_{\text{theo}}}{\sigma_{\text{tot}}} \quad (4.5.5)$$

### 4.5.3 b-jets: The T1bbbb and T2bb models

In the T1 and T2 case above we have used the term jet, by which we mean the shower of particles produced by the hadronization of a quark (or gluon). So when we counted jets we actually counted quarks, and only first or second generation quarks at that, i.e. no b-jets.

This is because the b-jets have their own simplified models, the T1bbbb and T2bb models, which we also have implemented in SASS. The T1bbbb simplified model is very similar to the T1 model with the key difference that all produced quarks have to be bottoms. Another difference is that the Higgs bosons are also treated as stable as they otherwise almost always decay to bottoms. We do this because we are not interested in those b-jets produced in that fashion. We also treat top quarks as stable, which otherwise would decay into bottom quarks and contribute. It is discarded in the T1bbbb model. This avoids double counting as there is dedicated simplified models for top quarks that account for this top squark contribution. The signature for the T1bbbb model is

$$n_{\text{bjet}} = 4 \quad n_{\tilde{\chi}} \geq 2 \quad n_{\tilde{g}} = 0 \quad (4.5.6)$$

The T2bb simplified model is the bottom quark version of the T2 model. Instead of, as in T2, producing first or second generation squark in the T2bb model sbottoms are produced. These then decay into bottoms and neutralinos. So, as in T1bbbb, we count b-jets rather than jets for the T2bb model. And exactly as in the T1bbbb model we treat the Higgs bosons and the top quark as stable particle. We have the signature

$$n_{\text{bjet}} = 2 \quad n_{\tilde{\chi}} \geq 2 \quad n_{\tilde{g}} = 0 \quad (4.5.7)$$

The signature probabilities  $P_{\text{sig}}$  and the theoretical predictions  $\sigma_{\text{theo}}$  for the T1bbbb and T2bb models are otherwise calculated as for the T1 respectively T2 models.

#### 4.5.4 lhcddata

The `lhcddata` module uses ROOT's Python bindings, PyROOT, to access root files containing the simplified model limits from LHC. We encapsulate the root file with the class `Limit`. When a `Limit` object is created it's given the path to the root file and the name of the particular limit to access. So

```
lim = lhcddata.Limit('example.root', 'limit_T1')
```

opens the file *example.root* and access the T1 limit. After the object creation we can easily retrieve limits by the method `GetLimit()` given the gluino mass, for T1, or the squark mass, for T2, and the LSP mass, all in GeV.

### 4.6 The run module

The `run` module collects various functions useful for user-written programs. As an example of such a function we have `setCMSSMPoint()`. What `setCMSSMPoint()` does is to configure a `Model` object as a cMSSM model and sets the five parameters of the cMSSM to numerical values. The function does this by creating, and setting the appropriate values within the SLHA blocks `MODESEL`, `SMINPUTS`, and `MINPAR` in the `Model` object.

But the major functions of `run` are the ones that run external programs such as Prospino and SUSY-HIT. These functions act as interfaces to external codes, allowing an user to run them on a model without leaving SASS. In this section we will spend some time describing these interfaces.

#### 4.6.1 SUSY-HIT

SUSY-HIT [17], short for SUpect-SdecaY-Hdecay-InTerface, is a code package for calculating the mass spectrum and decays for Higgs bosons and supersymmetric particles in the context of MSSM. It consists of three integrated programs: SuSpect [18] for the masses, SDECAY [19] for the decays of supersymmetric particles and HDECAY [20] for the Higgs bosons decays. The version of SUSY-HIT we have used is 1.3 which includes: SuSpect 2.41, SDECAY 1.3b and HDECAY 3.4.

The `run` module encapsulates SUSY-HIT with the function `runSUSYHIT()`. As arguments the function takes a `Model` object with appropriate blocks defined and populated, e.g. configured as an cMSSM point by `setCMSSMPoint()`.

The mechanics of the function are straightforward. We begin by creating a SLHA file to act as input for SUSY-HIT, then we run the SUSY-HIT binary. Then we parse the resulting SLHA file and store the results into the `Model` object. We then finally store the object to disk so the results are saved for the future. We log all these actions using the log file associated with the specific `Model` object.

`runSUSYHIT()` takes an additional, optional, argument named `workdir`. If specified `runSUSYHIT()` uses that as the SUSY-HIT directory, so `workdir` must be a path to functioning SUSY-HIT installation or `runSUSYHIT()` will fail. We implemented this functionality as we needed to run multiple instances of SUSY-HIT in parallel. SUSY-HIT makes use of intermediate files which makes it impossible to run parallel instances in the same directory. Using multiple directories are acceptable, if inelegant, workaround.

## 4.6.2 Prospino

Prospino [21, 22] computes the next-to-leading order production cross sections for supersymmetric particles at hadron colliders such as the LHC. We have used Prospino 2.1. The colliding protons are composite objects which means that, for example, the gluino pair production cross section is calculated using

$$\sigma(pp \rightarrow \tilde{g}\tilde{g}) = \sum_{i=q,g} \sum_{j=q,g} \int dx_1 dx_2 f_i(x_1) f_j(x_2) \sigma_P(ij \rightarrow \tilde{g}\tilde{g}) \quad (4.6.1)$$

i.e. sum over the different parton (quarks and gluons) cross sections,  $\sigma_P$ , convoluted with the appropriate parton distribution functions<sup>3</sup>,  $f_i$  and  $f_j$ . In figure 4.2 we show the production cross section for various particle pairs as a function of mass.

It is very important that these cross sections are calculated to at least the next-to-leading order as the processes we are interested in are all QCD processes. As the gauge coupling for QCD is large the size of the next-to-leading corrections can be sizeable.

Prospino is encapsulated by `runProspino()` which needs a `Model` object with particle masses already calculated by `runSUSYHIT()`. The function begins by creating SLHA file for input to Prospino, and then runs the binary much like all external program interface functions.

There is a slight variation in the treatment of Prospino's output. Prospino's output is non-standard, as there are no standards that apply. The output is a text file containing the leading order cross section, the next-to-leading order cross section, errors and such.

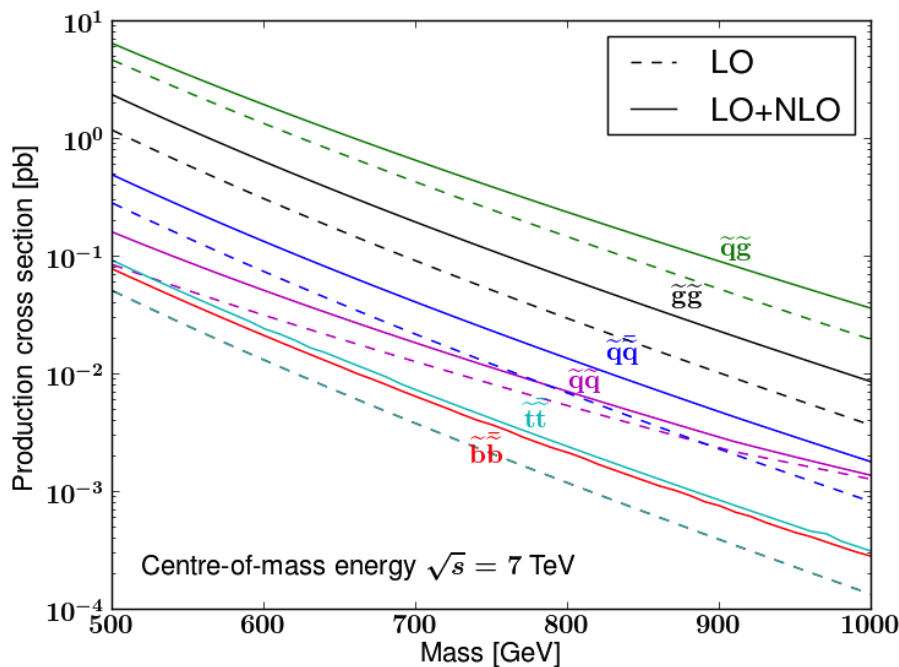
`runProspino()` copies this file to the model directory, saves it, and then it parses all Prospino results files in the model directory and creates a custom SLHA

---

<sup>3</sup>Prospino uses the CTEQ 6.6 [26] parton distribution functions.

block called `CROSSSEC` containing the leading order and next-to-leading order cross sections. The block is stored in the `Model` object which is then saved to disk.

`runProspino()` has an optional argument called `binary`. It specifies which Prospino binary to use. We have written different binaries for each simplified models. The simplest is `T1.run` for the T1 model which calculates the  $\tilde{g}\tilde{g}$  production cross section. The `T2.run`, while still not complex, does more. It calculates the  $\tilde{q}\tilde{q}$  and  $\tilde{q}\tilde{\bar{q}}$  production cross sections in all quark flavour combinations.



**Figure 4.2:** Cross sections as function of mass for different particle pair production from  $pp$  collisions calculated using Prospino [21, 22]. When calculating the gluino cross section the squark masses are set to 10 TeV, and vice versa. In the  $\tilde{q}\tilde{g}$  case both the squark and gluino masses are set to the value of the x-axis. Note that the LO  $\tilde{b}\tilde{b}$  line is hidden behind the LO  $\tilde{t}\tilde{t}$  line.

# Chapter 5

## Using Simplified Models to constrain the cMSSM

In this chapter we use SASS to reinterpret the limits on simplified models as limits on the cMSSM. cMSSM is very well-studied and it is therefore an interesting benchmark model to test SASS and the simplified model method against.

The experimental results we will use for our analysis are from the CMS collaboration [15]. They analysed a data sample from the CMS detector with a integrated luminosity of  $4.98 \text{ fb}^{-1}$  and a centre-of-mass energy of  $\sqrt{s} = 7 \text{ TeV}$ . For the analysis they used the kinematic variable  $\alpha_T$ , which for dijet events is defined as

$$\alpha_T = \frac{E_T^{j_2}}{M_T} \quad (5.0.1)$$

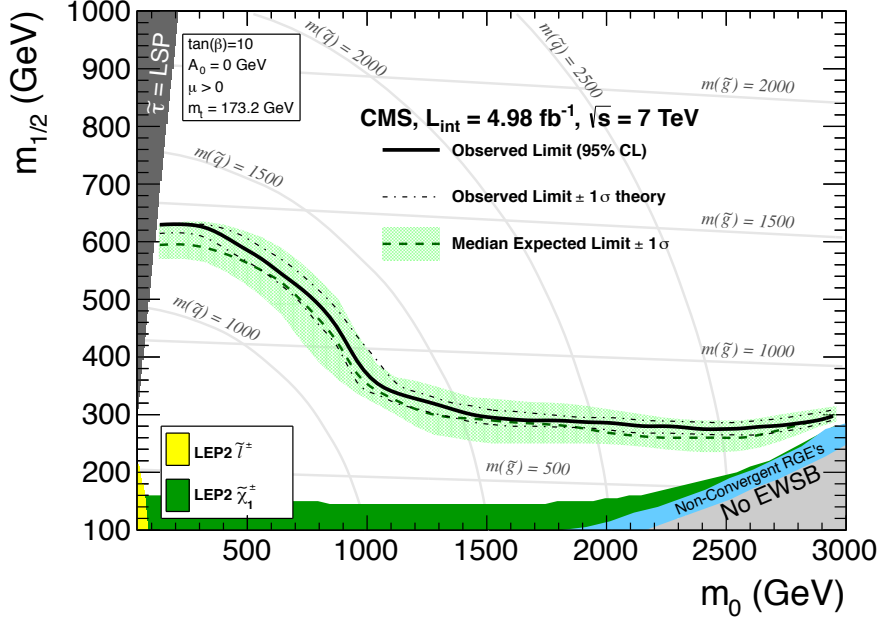
where  $E_T^{j_2}$  is the energy of the least energetic jet that is deposited transversely to the proton beams.  $M_T$  is the so-called transverse mass of the system defined as

$$M_T = \sqrt{\left(\sum E_T^{j_i}\right)^2 - \left(\sum p_x^{j_i}\right)^2 - \left(\sum p_y^{j_i}\right)^2}. \quad (5.0.2)$$

For a dijet event where the two jets are back-to-back,  $E_T^{j_1} = E_T^{j_2}$ , the value of  $\alpha_T$  is 0.5. A value of  $\alpha_T$  greater than 0.5 indicates that the two jets are not back-to-back and must therefore be recoiling against some missing energy, potentially a neutralino.

CMS interprets the data in the context of the T1, T2, T1bbbb, T2bb, T1tttt, and T2tt simplified models and put limits on the corresponding production cross sections. We will make use of those four results which do not involve the top quark: T1, T2, T1bbbb, and T2bb.

The corresponding 8 TeV results for these simplified models is also available [16]. However, in the 7 TeV paper the authors also use the data to constrain the

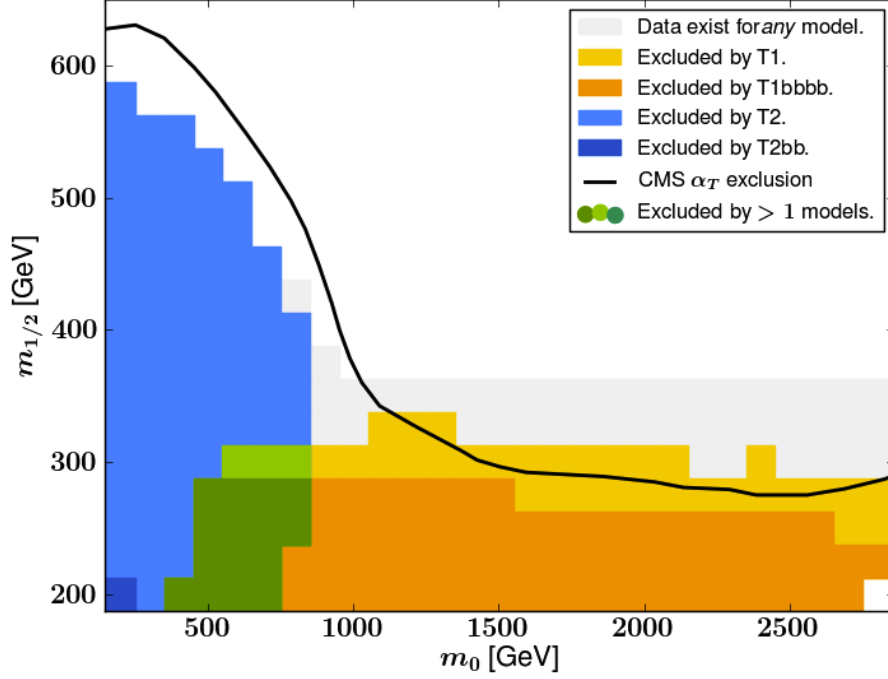


**Figure 5.1:** CMS exclusion contours at 95% confidence level in the cMSSM  $(m_0, m_{1/2})$  plane using the kinematical variable  $\alpha_T$  [15]. The other cMSSM parameters are  $\tan\beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ .

cMSSM  $(m_0, m_{1/2})$  parameter plane, see figure 5.1. This they do not do in the 8 TeV paper. We therefore chose to use the 7 TeV results as we are interested in recreating that cMSSM limit using their simplified model results. As both limits is derived from the same data sample they should in principle agree. We use this fact to check the validity of SASS and our analysis method.

The plane in the cMSSM parameter space the CMS paper put their limit on is the  $(m_0, m_{1/2})$  plane with  $A_0 = 0$  GeV,  $\tan\beta = 10$  and  $\mu > 0$ . The top mass they used is  $m_t = 173.2$  GeV. We do a grid scan through this plane; we let  $m_0$  vary from 200 GeV to 2800 GeV in steps of 100 GeV and  $m_{1/2}$  from 200 GeV to 600 GeV in steps of 25 GeV. Each of these points we analyse using SASS and test against the T1, T2, T1bbbb and T2bb simplified models.

The results are shown in figure 5.2 which depicts the excluded parameter points for each simplified model. A point is marked as excluded at 95% confidence level if the theoretical production cross section is larger than the experimental limit. The dark yellow region is excluded by the constraints on both the T1 and T1bbbb models, while the light yellow is excluded by only the T1 model limits. Similarly the dark blue region is excluded by the constraints on both the T2 and T2bb models, while the light blue is excluded by only the T2 model limit. The green regions are the intersection of the blue and yellow regions; the dark green by T1,



**Figure 5.2:** Exclusion in the cMSSM ( $m_0, m_{1/2}$ ) plane using the 7 TeV CMS limits on the T1, T1bbbb, T2 and T2bb simplified models [15]. The other cMSSM parameters are  $\tan\beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ . The light green area is excluded by both the T2 and T1bbbb models. The dark green area is excluded by the T1, T1bbbb, and T2 models. Note that the region excluded by T2bb is also excluded by T2, and similarly the T1bbbb region by the T1. Also depicted is the CMS  $\alpha_T$  exclusion contour from figure 5.1.

T2 and T1bbbb models and the light green by T1 and T2 models. The grey regions indicate points near the T1 and T2 models that are not excluded by either of these two simplified models but for which experimental data exists. Also shown is the CMS  $\alpha_T$  exclusion contour (black line) from figure 5.1.

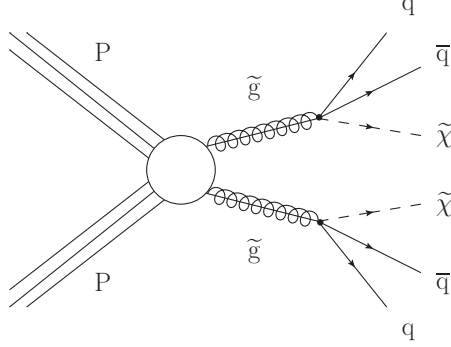
The first thing we note is the complementarity of the different simplified models; this is a reflection of two regions with different mass hierarchies. When  $m_{\tilde{g}} > m_{\tilde{q}}$ , i.e. low  $m_0$ , the T2 and T2bb models come into effect, and when  $m_{\tilde{q}} > m_{\tilde{g}}$ , high  $m_0$ , the T1 and T1bbbb applies. This complementarity also illustrates the idea behind simplified models. That for a given point in the parameter space of a theory different simplified models apply with different efficiencies. A set of simplified models acts, in a sense, as a basis for the full theories.

If we compare our combined simplified model exclusion with the CMS  $\alpha_T$  exclusion contour we see that they mostly are in agreement. We will discuss the discrepancies later as we discuss the results for each simplified model separately. For each simplified model we present two figures: one over the signature probability  $P_{\text{sig}}$ , and one exclusion plot. The signature probability  $P_{\text{sig}}$  is the fraction of initial particles that decays such that it matches the signature of the simplified model in question. Its definition, which is different for each simplified model, is found in section 4.5.

The exclusion plot shows the ratio of the theoretical prediction,  $\sigma_{\text{theo}}$ , to the experimental limit,  $\sigma_{\text{exp}}$ . If the ratio at a point is larger than one we mark it as excluded. This is of course just a crude categorisation, the ratio itself is more nuanced and informative. In the exclusion plots we also plot the  $\alpha_T$  exclusion contour from CMS [15] for ease of comparison.



## 5.1 The T1 model results



**Figure 5.3:** Schematic Feynman diagram of the T1 model.

As a reminder: the T1 simplified model is the production of one pair of gluinos which then each decay into one  $q\bar{q}$  pair and one neutralino. This is depicted in figure 5.3, for more information see section 3.1.

The results from applying the T1 simplified model on our parameter plane is shown in figure 5.4. The top figure is a map of the signature probability  $P_{\text{sig}}$  over the scanned cMSSM parameter plane. The signature probability  $P_{\text{sig}}$  is defined in section 4.5.1 for the T1 model as

$$P_{\text{sig}} = P(n_{\text{jet}} = 4, n_{\tilde{\chi}} \geq 2, n_{\tilde{q}} = 0 \mid 2 \tilde{g} \text{ decaying}). \quad (5.1.1)$$

From the map of the signature probability, we can discern some features worth noting. The most striking is the distinct low  $m_0$  region where  $P_{\text{sig}}$  is zero. In that region the squark mass is lower than the gluino mass which results in the gluinos decaying into squarks, which does not match the signature, and are therefore filtered out in the T1 model. The T1 model is simply not applicable in that region.

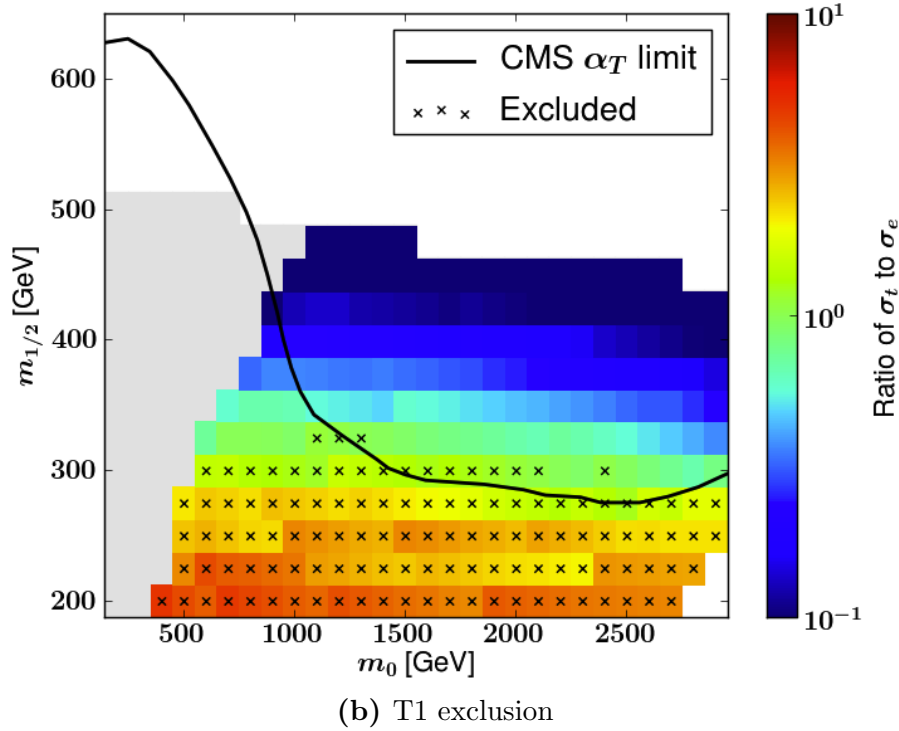
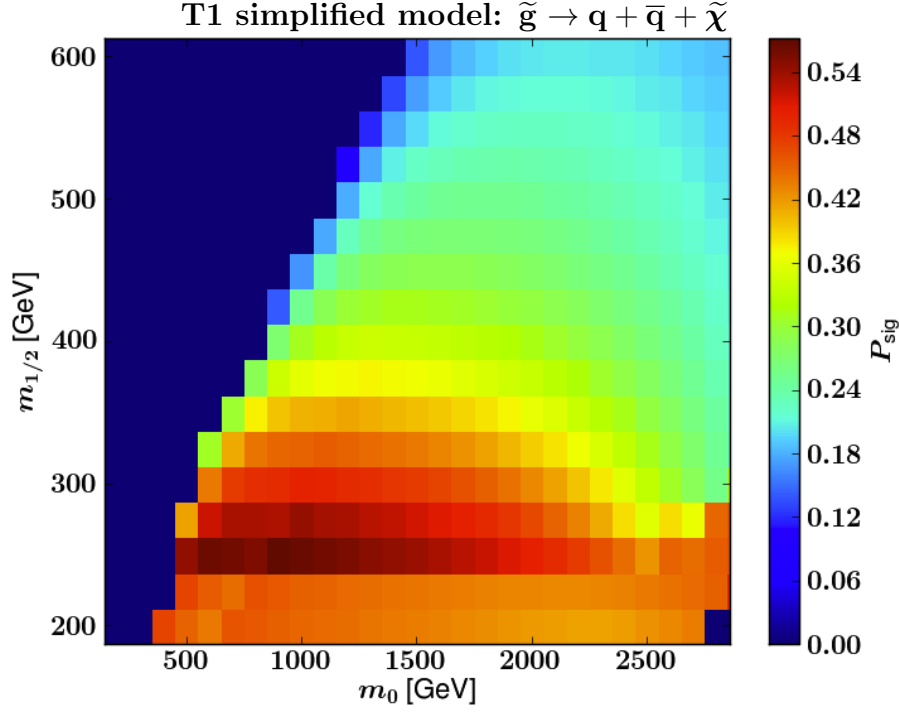
The second feature is the noticeably lower signature probability in the lines of  $m_{1/2} = 200$  and  $225$  GeV when compared to the  $m_{1/2} = 250$  GeV line. The reason behind this drop is that the gluino often does not directly decay into the lightest neutralino. It tends to decay into one of the heavier neutralinos or a chargino first. These then in turn decay into the lightest neutralino. This produces some additional particles, and when  $m_{1/2} > 250$  GeV these tends to be leptons,  $W^\pm$ , and Z bosons. These particles are ignored completely by this analysis. However, when  $m_{1/2}$  is 200 or 225 GeV these intermediate particles tend to decay into an additional  $q\bar{q}$  pair. The model efficiency is thus lowered as the T1 model demands exactly four jets. Not, as in this particular region, more than four.

This might be an unnecessarily strict requirement. In the experimental data analysis by the CMS collaboration [15] a jet has to have  $E_T > 50$  GeV to be

considered when calculating the number of jets. The additional jets produced from these intermediate particles quite possibly would fail this condition and therefore those decay topologies should not be eliminated in this analysis. SASS does not however have any facilities for estimating jet energies so we do not take this effect into account.

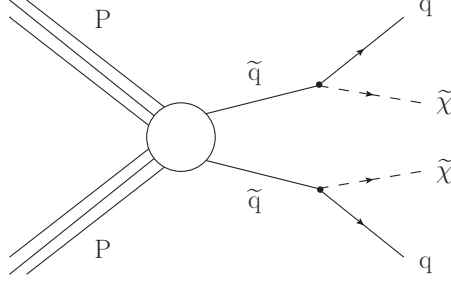
The bottom part of figure 5.4 is the exclusion plot showing the ratio of the theoretical prediction to the experimental limit,  $\sigma_t/\sigma_e$ . Points where this ratio is over one are marked excluded using a cross. Also shown is the CMS  $\alpha_T$  cMSSM exclusion contour (black line) taken from figure 5.1.

In this exclusion plot we see that we have good agreement with the  $\alpha_T$  exclusion contour. Our simplified model analysis does exclude some points above the CMS limit, however the ratio  $\sigma_t/\sigma_e$  in that whole region is very close to one, ca. 0.9–1.3, which is the value used to determine exclusion. This over-exclusion is therefore quite sensitive to the details of the computer tools used to calculate the cMSSM mass spectrum. CMS used SOFTSUSY [27] in [15] while we used SuSpect [18] for this calculation.



**Figure 5.4:** The signature probability map (a) and exclusion plot (b) for the T1 simplified model. The scanned cMSSM  $(m_0, m_{1/2})$  plane has  $\tan\beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ .  $P_{\text{sig}}$  ranges from zero to one. The non-white regions in (b) are where T1 experimental limits from the 7 TeV  $\alpha_T$  analysis [15] exist. The grey region in (b) is where  $\sigma_{\text{theo}} = 0$  pb and thus the ratio is zero. The crossed points in (b) indicate exclusion at 95% CL. Also shown in (b) is the CMS  $\alpha_T$  exclusion contour (black line) from figure 5.1.

## 5.2 The T2 results



**Figure 5.5:** Schematic Feynman diagram of the T2 model.

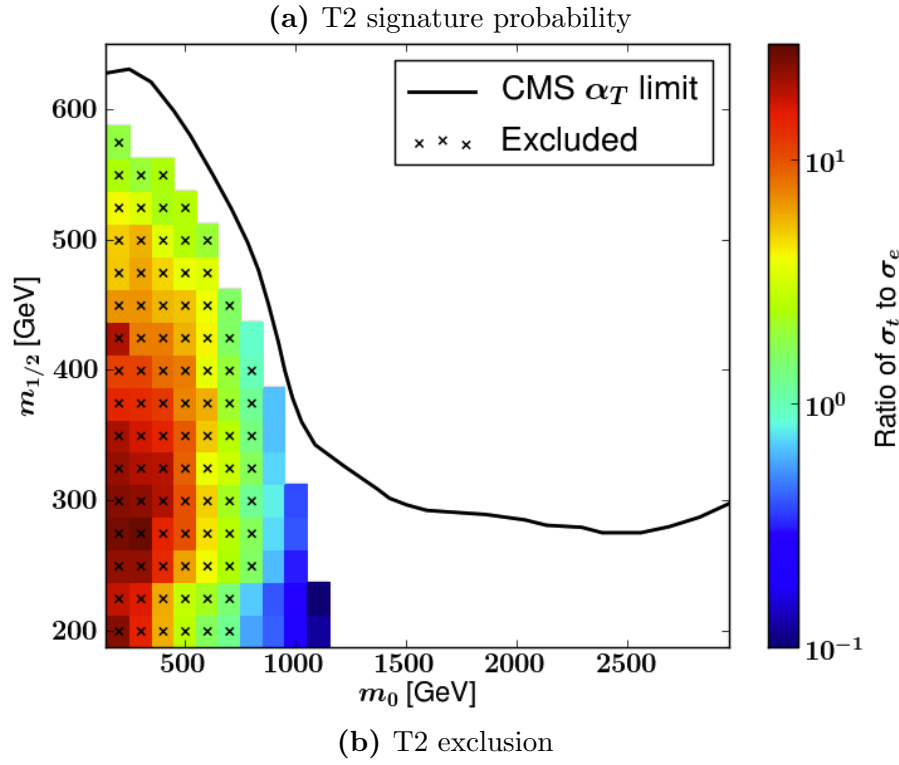
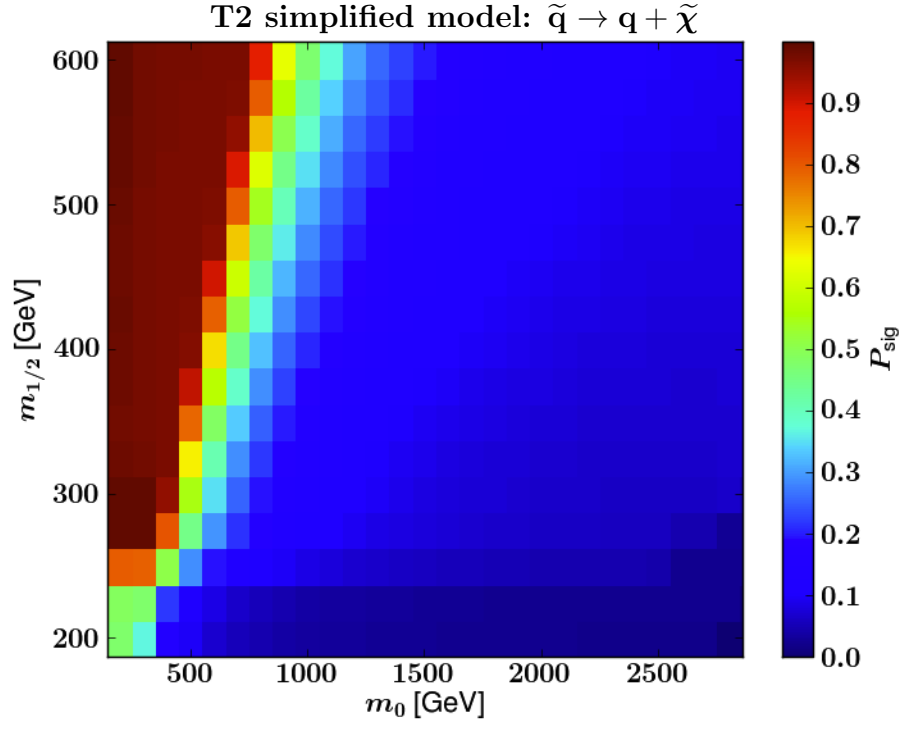
The T2 simplified model involves the production of a pair of squarks which then decay into one quark and one neutralino each. The model is depicted in figure 5.5 and see section 3.1 for more details.

In figure 5.6 we show the results of applying the T2 simplified model limits to the cMSSM plane. The top figure is the signature probability map, the signature probability  $P_{\text{sig}}$  for T2 is defined in section 4.5.2. In the figure we again see that striking division we first observed in the T1 signature probability. The probability suddenly drops when the gluino mass becomes lower than the squark mass, approx.  $m_0 > \frac{1}{2}m_{1/2}$ , which means squarks then start to decay into gluinos. This is a decay channel that does not match the the T2 topology and is thus filtered out.

But it is interesting to note that gluinos tend to all decay into squarks if possible, i.e. the T1 signature probability goes to zero when  $m_{\tilde{g}} > m_{\tilde{q}}$ , as we can see that this is not the case for the squarks. So while the T2 signature probability in the  $m_0 \gtrsim 1500$  GeV region is certainly not high, it is also not negligible. A portion of the squarks still directly decay to a neutralino and a quark even if the gluino decay channel is open.

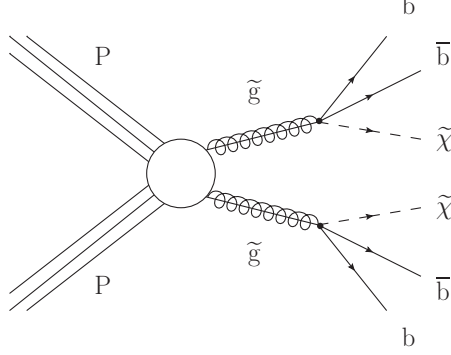
The bottom part of figure 5.6 is the exclusion plot showing the ratio of the theoretical prediction to the experimental limit,  $\sigma_t/\sigma_e$ . Points where this ratio is over one are marked excluded using a cross. Also shown is the CMS  $\alpha_T$  exclusion contour (black line) taken from figure 5.1.

We see that our exclusion is more conservative than the CMS  $\alpha_T$  limit. This is due to the unfortunate fact that the published limits on the T2 simplified model, figure 3.2, does not cover the full mass region we are interested in. The simplified model limits stop at  $m_{\tilde{q}} \approx 1180$  GeV, whereas the CMS  $\alpha_T$  exclusion contour is at  $m_{\tilde{q}} \approx 1250$  GeV, see figure 5.1. This explains why there is still a white region just under the CMS  $\alpha_T$  exclusion contour at low  $m_0$ .



**Figure 5.6:** The signature probability map (a) and the exclusion plot (b) for the T2 simplified model. The scanned cMSSM  $(m_0, m_{1/2})$  plane has  $\tan \beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ .  $P_{\text{sig}}$  ranges from zero to one. The non-white regions in (b) are where T2 experimental limits from the 7 TeV  $\alpha_T$  analysis [15] exists. The crossed points in (b) indicate exclusion at 95% CL. Also shown in (b) is the CMS  $\alpha_T$  exclusion contour (black line) from figure 5.1.

### 5.3 The T1bbbb results



**Figure 5.7:** Schematic Feynman diagram of the T1bbbb model.

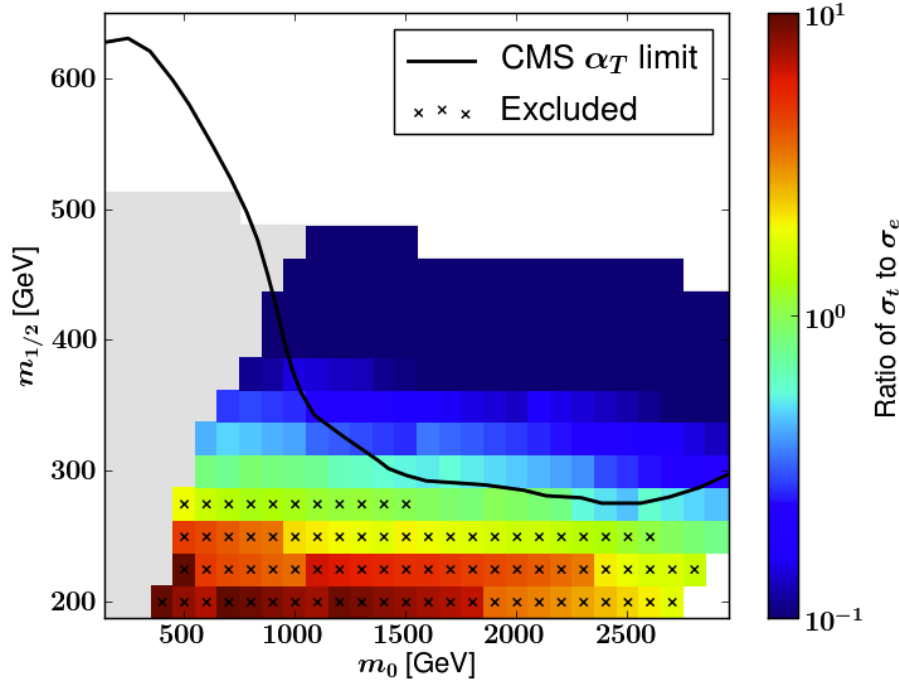
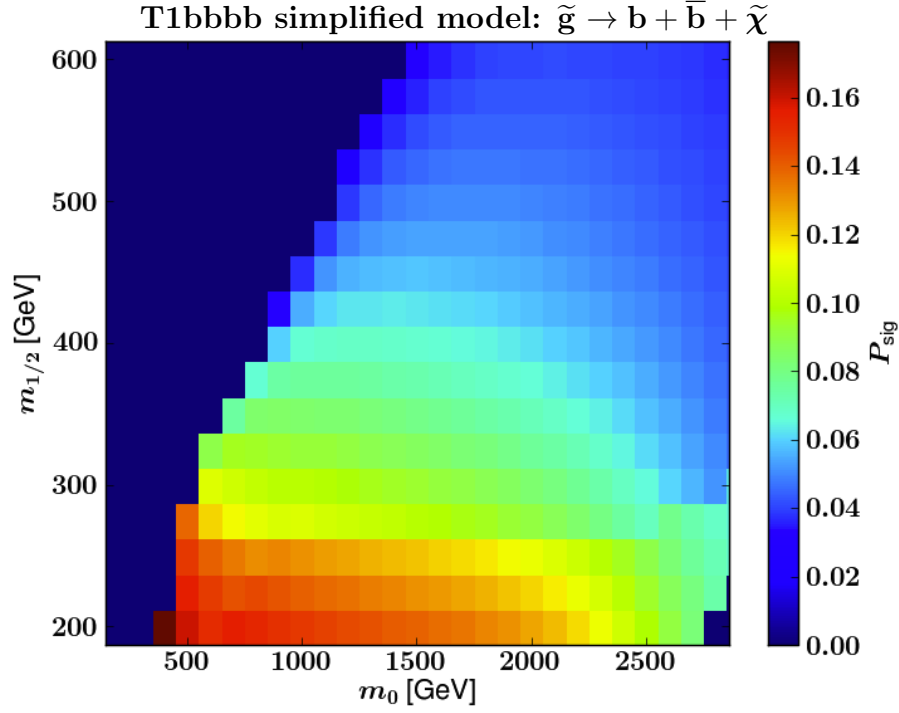
The T1bbbb simplified model is the production of two gluinos which then each decay into two bottom quarks and one neutralino. The model is depicted in figure 5.7 and see section 3.1 for more details.

In figure 5.8 we show the results of applying the T1bbbb simplified model limits to the cMSSM plane. The top figure is a map of the signature probability  $P_{\text{sig}}$  over the scanned cMSSM parameter plane. The signature probability  $P_{\text{sig}}$  is defined in section 4.5.3 for the T1bbbb model as

$$P_{\text{sig}} = P(n_{\text{b-jet}} = 4, n_{\tilde{\chi}} \geq 2, n_{\tilde{q}} = 0 \mid 2 \tilde{g} \text{ decaying}). \quad (5.3.1)$$

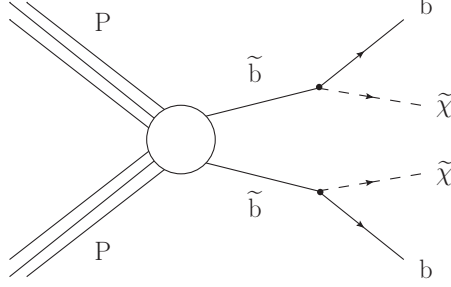
The structure of the signature probability map is very similar as the structure of the T1 model signature probability map (cf. figure 5.4). The probabilities are however lower, which is to be expected as the signature for T1bbbb is stricter than for the T1 model; only one flavour of squark, versus the four considered in the T1 case.

The bottom part of figure 5.8 is the exclusion plot showing the ratio of the theoretical prediction to the experimental limit,  $\sigma_t/\sigma_e$ . Points where this ratio is over one are marked excluded using a cross. Also shown is the CMS  $\alpha_T$  exclusion contour (black line) taken from figure 5.1. We see that that our exclusion is consistent with this exclusion contour. It is even slightly conservative, which means that for most of this region the T1 model contributes most to the combined exclusion. This is expected as in the T1 model we have contribution from four squark flavours, while in T1bbbb we only have one. The T1bbbb model becomes relevant in models where the the third generation squarks is noticeable lighter than in first and second generation squarks.



**Figure 5.8:** The signature probability map (a) and exclusion plot (b) for the T1bbbb simplified model. The scanned cMSSM  $(m_0, m_{1/2})$  plane has  $\tan \beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ .  $P_{\text{sig}}$  ranges from zero to one. The non-white regions in (b) are where T1bbbb experimental limits from the 7 TeV  $\alpha_T$  analysis [15] exists. The grey region in (b) is where  $\sigma_{\text{theo}} = 0$  pb and thus the ratio is zero. The crossed points in (b) indicate exclusion at 95% CL. Also shown in (b) is the CMS  $\alpha_t$  exclusion contour (black line) from figure 5.1.

## 5.4 The T2bb results



**Figure 5.9:** Schematic Feynman diagram of the T2bb model.

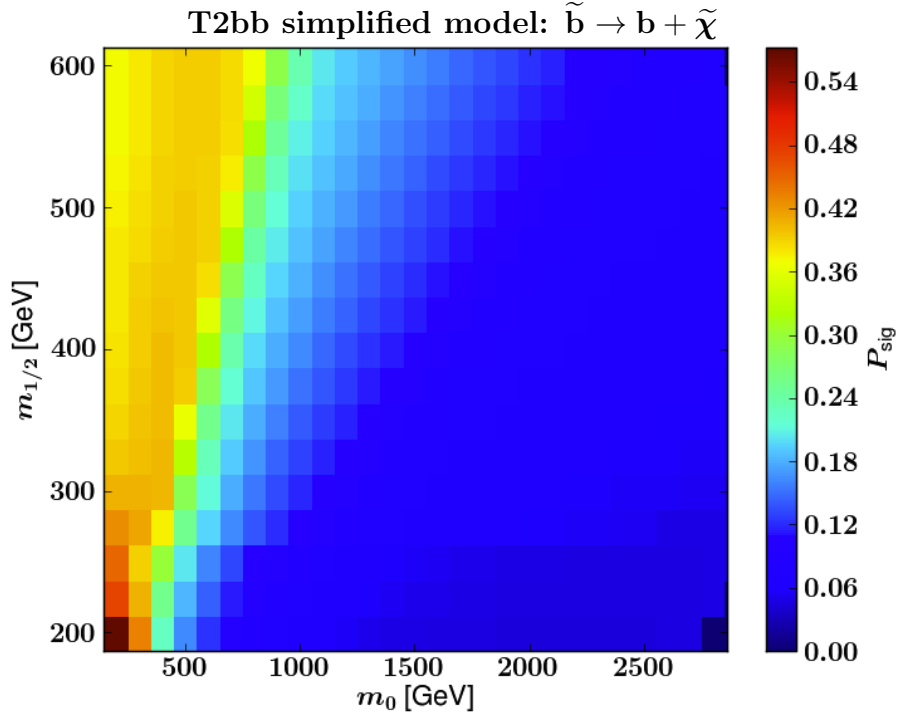
The T2bb simplified model is the production of two sbottoms which then decays into bottoms and neutralinos. The model is depicted in figure 5.7 and see section 3.1 for more details.

The top plot in figure 5.10 is the signature probability map, the signature probability  $P_{\text{sig}}$  for T2 is defined in section 4.5.3. The bottom plot is the exclusion plot showing the ratio of the theoretical prediction to the experimental limit,  $\sigma_t/\sigma_e$ . Points where this ratio is above one are marked excluded using a cross. Also shown is the CMS  $\alpha_T$  exclusion contour (black line) taken from figure 5.1.

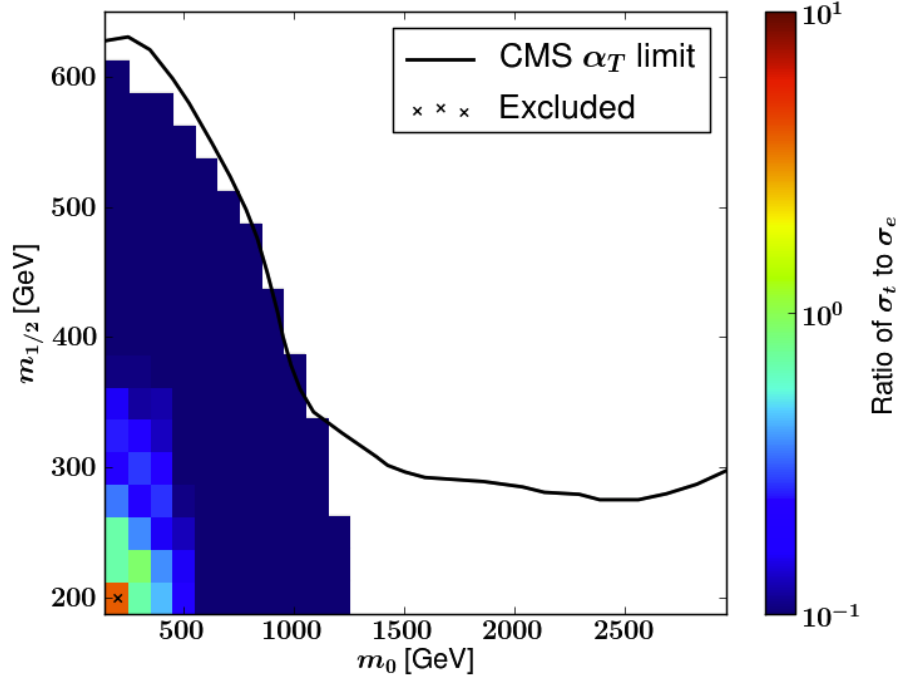
Only a single point is excluded even if the  $P_{\text{sig}}$  is only approximately half of that of the T2 model, which excludes a much larger region. This is because at 7 TeV the production cross section for sbottoms is simply too low compared to the combined production cross section of the first and second generation squarks as used in the T2 simplified model.

It is thus not a very useful model in this particular region of cMSSM parameter space. It is not very useful for cMSSM at all in fact as in the cMSSM model all the squarks have very similar mass. This means that the region excluded by the T2 model will always overshadow the region excluded by the T2bb simplified model. However, to constrain a different model where the third generation squarks are much lighter than the first and second generation squarks the T2bb simplified model would be much more useful.





(a) T2bb signature probability



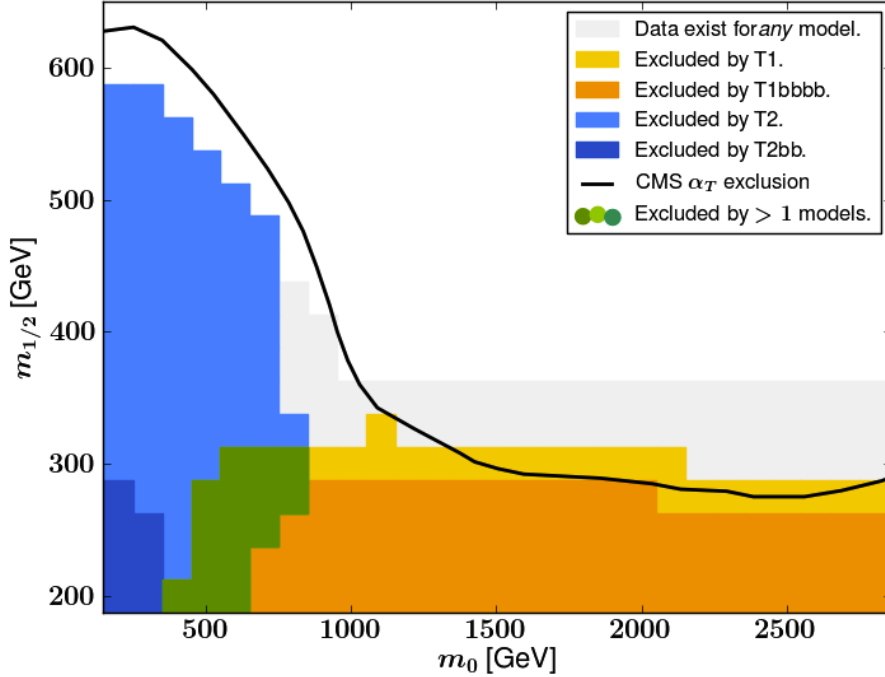
(b) T2bb exclusion

**Figure 5.10:** The signature probability map (a) and exclusion plot (b) for the T2bb simplified model. The scanned cMSSM  $(m_0, m_{1/2})$  plane has  $\tan\beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ .  $P_{\text{sig}}$  ranges from zero to one. The non-white regions in (b) are where T1bbbb experimental limits from the 7 TeV  $\alpha_T$  analysis [15] exists. The crossed points in (b) indicate exclusion at 95% CL. Also shown in (b) is the CMS  $\alpha_t$  exclusion contour (black line) from figure 5.1.

## 5.5 Constraining the cMSSM using 8 TeV results

The CMS collaboration has published updated limits on the T1, T2, T1bbbb and T2bb simplified models [16] using a data sample corresponding to integrated luminosity of  $L_{int} = 11.7 \text{ pb}^{-1}$  captured at a centre-of-mass energy of  $\sqrt{s} = 8 \text{ TeV}$ . While the 7 TeV paper [15] also used its data sample to put limits on the cMSSM plane, see figure 5.1, the 8 TeV paper [16] did not include such a limit.

We use SASS to put limits on the cMSSM model by reinterpreting the 8 TeV limits on the simplified models. We use the same cMSSM parameter plane, the  $(m_0, m_{1/2})$  plane with  $\tan\beta = 10$ ,  $A_0 = 0 \text{ GeV}$ , and  $\mu > 0$ , and we even use the same parameter points. The result of applying these 8 TeV simplified model limits on this plane is shown in figure 5.11. A point is marked as excluded at 95% confidence limit if the theoretical production cross section is larger than the



**Figure 5.11:** Exclusion in the cMSSM  $(m_0, m_{1/2})$  plane using the 8 TeV CMS limits on the T1, T1bbb, T2 and T2bb simplified models [16]. The other cMSSM parameters are  $\tan\beta = 10$ ,  $A_0 = 0 \text{ GeV}$ , and  $\mu > 0$ . The dark green area is excluded by the T1, T1bbbb, and T2 models. Note that the region excluded by T2bb is also excluded by T2, and similarly the T1bbbb region by T1. Also depicted is the 7 TeV CMS  $\alpha_T$  exclusion contour from figure 5.1.

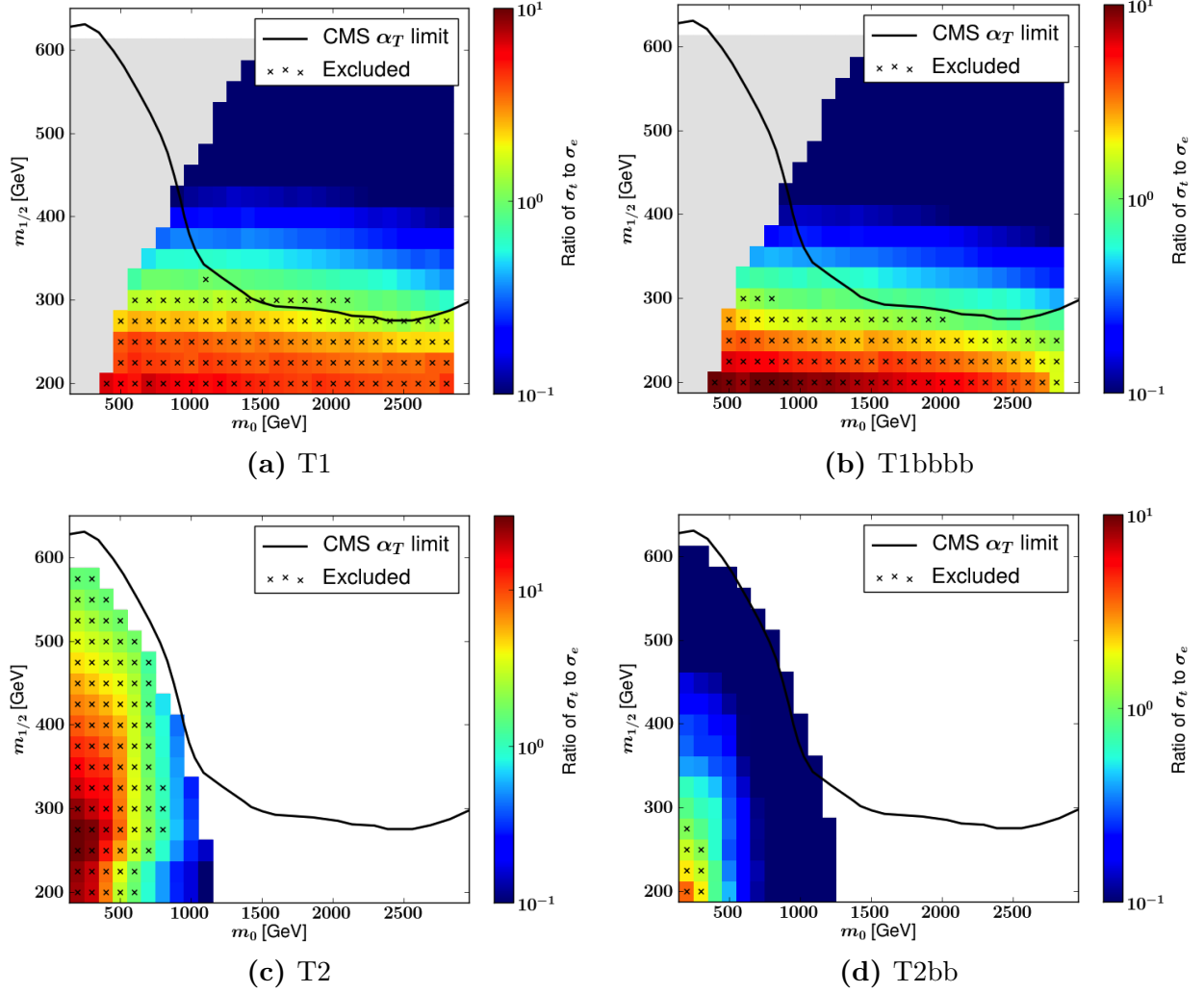
experimental limit. This is indicated using different colours. The dark yellow region is excluded by the T1 and T1bbbb models, while the light yellow is excluded by only the T1 model. Similarly the dark blue region is excluded by both the T2 and T2bb models, while the light blue is excluded by only the T2 model. The green regions are the intersection of the blue and yellow regions; the dark green by T1, T2 and T1bbbb models and the light green by T1 and T2 models. The grey regions indicate points near the T1 and T2 models and are not excluded by either of these two simplified models.

We have kept the CMS  $\alpha_T$  exclusion contour based on the 7 TeV data sample, see figure 5.1, for ease of comparing this plot with the plot based on the 7 TeV data sample, see figure 5.2. If we do this comparison we see that the excluded region has not grown from increasing the energy and using more than twice the integrated luminosity. This was unexpected as a larger data sample taken at higher energy should improve limits (in absence of a signal). To explain this we need to look at each simplified model separately, and to this end we have included the exclusion plots for each of the four simplified models at 8 TeV, see figure 5.12.

These are the same kind of plot as we showed for each model in the 7 TeV case and in these plots we plot the ratio of the theoretical prediction to the experimental limit,  $\sigma_t/\sigma_e$ . Points where this ratio is over one is marked excluded using a cross. Also shown again is the 7 TeV CMS  $\alpha_T$  exclusion contour (black line) taken from figure 5.1.

We do not include the signature probability maps we showed for each model in the 7 TeV case. This is because there is no discernible change in signature probability maps. This is expected; in the T1 and T1bbbb models the signature probability only depends on the branching ratios, which do not depend on the centre-of-mass energy. For the T2 and T2bb models the signature probability does depend on the production cross sections, which in turn depends on the centre-of-mass energy. The signature probability uses the cross sections to weight the contribution of from each squark species. In this case we find that the weight factors have not changed with the increase in energy.

If we compare the excluded regions we see that the T1bbbb and T2bb regions have increased while the T1 and T2 regions have not, the latter have even worsened in some points. This is perfectly consistent with our observation in section 3.2 where we saw that the observed exclusion contours in the simplified models limit only increased for the b-jet analyses and not for the T1 or the T2 model. This naturally translates to no better exclusion in our cMSSM plane, as both the T1 and T2 models are overshadowing the b-jet analyses and thus still dominate the combined exclusion.



**Figure 5.12:** The exclusion plots for the T1 (a), T1bbbb (b), T2 (c) and T2bb (d) simplified model. The scanned cMSSM ( $m_0, m_{1/2}$ ) plane has  $\tan \beta = 10$ ,  $A_0 = 0$  GeV, and  $\mu > 0$ .  $P_{\text{sig}}$  ranges from zero to one. The non-white regions are where experimental limits from the 8 TeV  $\alpha_T$  analysis [16] exists. The grey regions are where  $\sigma_{\text{theo}} = 0$  pb and thus the ratio is zero. The crossed points indicate that the point is excluded at 95% CL. Also shown is the 7 TeV CMS  $\alpha_T$  exclusion contour (black line) from figure 5.1.

# Chapter 6

## Summary and outlook

In this report we have, beside some introductory material on supersymmetry and simplified models, focused on documenting our newly developed analysis toolchain SASS, which is short for Supersymmetry Analysis using Simplified models. Through SASS we have constructed a way to reinterpret limits on simplified models from the CMS collaboration as constraints on more general supersymmetric theories such as the cMSSM or the pMSSM. The published results on simplified models are upper limits on the production cross section relevant for each simplified model. SASS, given a theory to test, calculates a theoretical prediction to compare these limits to. This prediction is the corresponding production cross section  $\sigma_{\text{prod}}$  for the theory multiplied by the probability that the produced particles decay so it matches the experimental signature of the simplified model in question. If we denote that signature probability as  $P_{\text{sig}}$  we can then write the theoretical prediction as  $\sigma_t = P_{\text{sig}} \times \sigma_{\text{prod}}$ . If this number is larger than the simplified model upper limit then the theory is excluded.

This approach is, of course, not without its limitations. In calculating the signature probability  $P_{\text{sig}}$ , i.e. how the initial particles decays, SASS ignores all kinematical factors. SASS only use the branching ratios<sup>1</sup> when calculating the signature probability. The benefit of ignoring the kinematics is that it allows SASS to avoid doing any costly Monte Carlo simulations. This does however mean that SASS could potentially overestimate  $\sigma_t$  in certain parts of the parameter space, since kinematic factors can only lower the estimate.

We also put SASS to use in this report, we used it to constrain the cMSSM. We began with recreating the 7 TeV CMS  $\alpha_T$  exclusion contour in the  $(m_0, m_{1/2})$  cMSSM parameter plane. We used the simplified model results based on the same data sample as the exclusion contour was derived from [15]. Using four hadronic

---

<sup>1</sup>This is not strictly true; the production cross sections is used as weights when calculating the signature probabilities involving the 'squark', i.e. combination of first and second generation squarks, as initial particle.

simplified model we were able to recreate the main features of this exclusion contour which, it is important to note, is based on a dedicated cMSSM analysis. We also used more recent simplified models results based on a 8 TeV data sample [16] to put limit on the same cMSSM parameter plane. Curiously enough the excluded region did not grow much compared to the 7 TeV results, this is due to the 8 TeV simplified models limit not improving greatly compared to their 7 TeV counterparts. These results convince us that simplified models is a very viable approach for supersymmetry phenomenology, and that SASS is useful tool embodying this approach.

As for future prospects the most obvious is to implement additional simplified models in SASS. Which simplified models to implement is of course limited by which models the LHC collaborations decide to publish results on. That being said, the first step should be implementing the T1tttt and T2tt models using the available CMS  $\alpha_T$  results [16]. The T1tttt model being the production of gluinos which then decays into four top quarks and two neutralinos. And in the T2tt model two stop squarks are produced which each decay into a top and neutralino.

Besides extending the reach of SASS by implementing more simplified models we want to use it to constrain theories of more interest than the cMSSM. We are interested in using SASS to constrain both the more general pMSSM model and more specific scenarios. One example of such scenario are the ‘well-mixed’ neutralinos scenario [28], which is based on relaxing universality assumptions in the cMSSM.

Another issue worth noting is that calculating the production cross sections using Prospino is the most time expensive part of SASS. Approximately 90% of the time is spent in Prospino, for the T2 model that is around 45 minutes. It is therefore very motivated to find a fast replacement for Prospino, one such possible replacement would be a artificial neural net trained by Prospino results.

The idea of simplified models being a basis for possible supersymmetric theories is also an idea worth investigating further. It would be interesting if we could develop some kind of simplified model algebra. This would formalise the simplified models approach, which certainly would be a boon from a statistical perspective.

## Acknowledgements

We would first like to acknowledge and thank Wolfgang Waltenberger of the CMS collaboration for giving us access to simplified models limits in a easy to use format and, very helpfully, answering our many questions. On a more personal note I would like thank my family for being so supportive and their, at least pretend, understanding. I would also like to acknowledge both my supervisors, Oscar Stål and Abram Krislock, for their invaluable guidance and help. It goes without saying that this could not be done without them. I also want to thank Helena Engström for all the shared pain and joy our physics education have entailed. Finally I would like to express my gratitude to the CoPS group for giving me a fantastic working environment.

# References

- [1] C. Streve, G. Bertone, F. Feroz, M. Fornasa, R. Ruiz de Austri, and R. Trotta, *Global fits of the cMSSM and NUHM including the LHC Higgs discovery and new XENON100 constraints*, *JCAP* **4** (2013) 13, [arXiv:1212.2636 \[hep-ph\]](#).
- [2] CMS Collaboration, *Interpretation of Searches for Supersymmetry*, Tech. Rep. CMS-PAS-SUS-11-016, CERN, Geneva, 2012.
- [3] S. P. Martin, *A Supersymmetry Primer*, in *Perspectives on Supersymmetry*. 1998. [arXiv:hep-ph/9709356](#).
- [4] R. Haag, J. T. Łopuszański, and M. Sohnius, *All Possible Generators of Supersymmetries of the S Matrix*, *Nucl.Phys.* **B88** (1975) 257.
- [5] S. Coleman and J. Mandula, *All Possible Symmetries of the S Matrix*, *Phys. Rev.* **159** (1967) 1251.
- [6] Planck Collaboration, P. A. R. Ade, N. Aghanim, C. Armitage-Caplan, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. J. Banday, and et al., *Planck 2013 results. I. Overview of products and scientific results*, [arXiv:1303.5062 \[astro-ph.CO\]](#).
- [7] J. Ellis and K. A. Olive, *Supersymmetric dark matter candidates*, p. 142. Cambridge University Press, 2010.
- [8] N. Arkani-Hamed, B. Knuteson, S. Mrenna, P. Schuster, J. Thaler, N. Toro, and L.-T. Wang, *MARMOSET: The Path from LHC Data to the New Standard Model via On-Shell Effective Theories*, [arXiv:hep-ph/0703088](#).
- [9] J. Alwall, P. C. Schuster, and N. Toro, *Simplified models for a first characterization of new physics at the LHC*, *Phys. Rev. D* **79** (2009) 075020, [arXiv:0810.3921 \[hep-ph\]](#).
- [10] D. Alves et al., *Simplified models for LHC new physics searches*, *J. Phys. G* **39** (2012) 105005, [arXiv:1105.2838 \[hep-ph\]](#).



- [11] M. D’Alfonso, *Joint ATLAS+CMS Simplified Models talk*, Presentation at Implications of LHC results for TeV-scale physics meeting, 2012. Slides: <http://indico.cern.ch/getFile.py/access?contribId=4&sessionId=4&resId=0&materialId=slides&confId=162621>.
- [12] S. Mrenna, *Simplified models in ATLAS*, Presentation at Implications of LHC results for TeV-scale physics: WG2 meeting, 2011. Slides: <http://indico.cern.ch/getFile.py/access?contribId=5&sessionId=1&resId=0&materialId=slides&confId=157244>.
- [13] Z. Marshall, *Simplified models in CMS*, Presentation at Implications of LHC results for TeV-scale physics: WG2 meeting, 2011. Slides: <http://indico.cern.ch/getFile.py/access?contribId=5&sessionId=1&resId=0&materialId=slides&confId=157244>.
- [14] H. Okawa and for the ATLAS Collaboration, *Interpretations of SUSY Searches in ATLAS with Simplified Models*, [arXiv:1110.0282](#) [hep-ex].
- [15] CMS Collaboration, *Search for supersymmetry in final states with missing transverse energy and 0, 1, 2, or  $\geq 3$  b-quark jets in 7 TeV pp collisions using the variable  $\alpha_T$* , *JHEP* **1** (2013) 77, [arXiv:1210.8115](#) [hep-ex].
- [16] CMS Collaboration, *Search for supersymmetry in hadronic final states with missing transverse energy using the variables  $\alpha_T$  and b-quark multiplicity in pp collisions at 8 TeV*, [arXiv:1303.2985](#) [hep-ex].
- [17] M. M. Mühlleitner, A. Djouadi, and M. Spira, *Decays of Supersymmetric Particles — the Program SUSY-HIT*, *Acta Physica Polonica B* **38** (2007) 635, [arXiv:hep-ph/0609292](#).
- [18] A. Djouadi, J.-L. Kneur, and G. Moultaka, *SuSpect: A Fortran code for the Supersymmetric and Higgs particle spectrum in the MSSM*, *Comp. Phys. Comm.* **176** (2007) 426, [arXiv:hep-ph/0211331](#).
- [19] M. Mühlleitner, A. Djouadi, and Y. Mambrini, *SDECAY: a Fortran code for the decays of the supersymmetric particles in the MSSM*, *Comp. Phys. Comm.* **168** (2005) 46, [arXiv:hep-ph/0311167](#).
- [20] A. Djouadi, J. Kalinowski, and M. Spira, *HDECAY: a program for Higgs boson decays in the Standard Model and its supersymmetric extension*, *Comp. Phys. Comm.* **108** (1998) 56, [arXiv:hep-ph/9704448](#).
- [21] W. Beenakker, R. Höpker, and M. Spira, *PROSPINO: A Program for the Production of Supersymmetric Particles in Next-to-leading Order QCD*, [arXiv:hep-ph/9611232](#).

- [22] W. Beenakker, R. Höpker, M. Spira, and P. Zerwas, *Prospino 2.1*, <http://www.thphys.uni-heidelberg.de/~plehn/index.php?show=prospino>, 2012.
- [23] B. C. Allanach *et al.*, *SUSY Les Houches Accord 2, Computer Physics Communications* **180** (2009) 8, [arXiv:0801.0045](https://arxiv.org/abs/0801.0045) [hep-ph].
- [24] R. Brun and F. Rademakers, *ROOT - An Object Oriented Data Analysis Framework, Nucl. Inst. & Meth. in Phys. Res. A* **389** (1997) 81. See also <http://root.cern.ch/>.
- [25] A. Buckley, *PySLHA: a Pythonic interface to SUSY Les Houches Accord data*, [arXiv:1305.4194](https://arxiv.org/abs/1305.4194) [hep-ph].
- [26] P. M. Nadolsky, H.-L. Lai, Q.-H. Cao, J. Huston, J. Pumplin, D. Stump, W.-K. Tung, and C.-P. Yuan, *Implications of CTEQ global analysis for collider observables, Phys.Rev. D* **78** (2008) , [arXiv:0802.0007](https://arxiv.org/abs/0802.0007) [hep-ph].
- [27] B. C. Allanach, *SOFTSUSY: A program for calculating supersymmetric spectra, Comp. Phys. Comm.* **143** (2002) 305, [arXiv:hep-ph/0104145](https://arxiv.org/abs/hep-ph/0104145).
- [28] D. Feldman and P. Sandick, *Well-Mixed Dark Matter and the Higgs*, [arXiv:1303.0329](https://arxiv.org/abs/1303.0329) [hep-ph].

# Appendix A

## Example SASS program

A example program using the toolchain SASS to apply the T1 simplified model limits of CMS on a cMSSM parameter point. For more detail see section 4.2.

```
import Model, sms, run

# Create a model
m = Model.Model(name='example', directory='./')

# There is builtin logging to a model specific file.
m.log.write('This is a example')

# Set our desired cMSSM point
run.setCMSSMPoint(model=m,
                  m0=200,
                  m12=300,
                  A0=0,
                  tanb=10,
                  musign=+1)

# calculate particle masses and their decays
run.runSUSYHIT(model=m,
               workdir=<path to SUSY-HIT installation>)

# Calculate the cross section.
run.runProspino(model=m, binary='T1.run',
               workdir=<path to Prospino installation>)
```

```

# Calculate the theory predicted cross section
theo = sms.T1.theoSigma(model=m)

# Retrieve the experimental value.
exp = sms.T1.expSigma(model=m,
                       datafile='../data/CMS limits/sms.root')

# Compare theory with experiment.
if theo > exp:
    m.log.write('Excluded by T1.')
else:
    m.log.write('Not excluded by T1.')

# Store the results
m.sigmas['T1'] = (theo, exp)
m.save()
m.writeSLHAFile()

```