

SIMULACIÓN EN PYTHIA 6 DE LA PRODUCCIÓN DE SQUARKS Y
GLUINOS VIA NEUTRALINOS, TAUS Y STAUS A ENERGÍAS DE
CENTRO DE MASA DE 7, 9, 11 Y 14 TeV EN EL EXPERIMENTO CMS
DEL LABORATORIO CERN

TRABAJO DE GRADO

Para optar el título de:

Físico

LUIS HERNANDO ECHEVERRI POTOSI

Universidad de Nariño
Facultad de Ciencias Exactas y Naturales
Departamento de Física
Noviembre 2015

SIMULACIÓN EN PYTHIA 6 DE LA PRODUCCIÓN DE SQUARKS Y
GLUINOS VIA NEUTRALINOS, TAUS Y STAUS A ENERGÍAS DE
CENTRO DE MASA DE 7, 9, 11 Y 14 TeV EN EL EXPERIMENTO CMS
DEL LABORATORIO CERN

LUIS HERNANDO ECHEVERRI POTOSI

TRABAJO DE GRADO

Director:

JAIME BETANCOURT MINGANQUER
M.Sc fisica

Universidad de Nariño
Facultad de Ciencias Exactas y Naturales
Departamento de Física
Noviembre 2015

©2015 - LUIS HERNANDO ECHEVERRI POTOSI

“Las ideas y conclusiones aportadas en la tesis de grado son responsabilidad exclusiva de los autores”

Artículo 1. del acuerdo No. 324 del 11 de Octubre de 1966, emanado por el Honorable Consejo Directivo de la Universidad de Nariño.

Todos los derechos reservados.

Nota de Aceptación

Jaime Alfredo Betancourt

Director

Juan Florez

Jurado

Jose Luis Palomino

Jurado

San Juan de Pasto, Noviembre 2015

Agradecimientos

A mi MAMÁ por que simplemente me ha dado todo.

A mi amigo y Maestro David Martinez quien me a guiado en el camino de las altas energías.

Al profesor Jaime por darme la oportunidad de pertenecer al grupo de investigación en Altas energías de la Universidad de Nariño y brindarme todas sus valiosas enseñanzas, por todas las correcciones y aportes del trabajo.

A mis compañeros y amigos por compartir tan grande experiencia.

A mi madre por supuesto

Tabla de Contenido

1	Introducción	1
2	SUPERSIMETRIA	4
2.1	Modelo Estándar	4
2.2	Supersimetria	6
2.3	Rompimiento de supersimetria	8
2.4	Parametros MSUGRA	9
2.5	CMS	12
3	Simulación del canal	14
3.1	Canal de estudio	14
3.2	Phytia 6	16
3.3	Root	17
3.3.1	Marcos de trabajo	17
3.3.2	Programación orienta a objetos (POO)	18
3.3.3	Procesos	18
4	Análisis y resultados	21
4.1	Canal de decaimiento	21
4.2	Momento transversal faltante	22
4.3	Energía transversal faltante	23
4.4	Separación angular	24
4.5	Energia particulas Vs energía centro de masa	26
5	Conclusiones y Recomendaciones	29
5.1	Conclusiones	29
5.2	Recomendaciones	30
	ANEXOS	31
A	Código canal de decaimiento	31
	ANEXOS	40
B	Energía transversal faltante	40
	Bibliografía	42

Lista de Figuras

1.1	Canal de estudio	2
2.1	Partículas del modelo estándar	6
2.2	Partículas SUSY	8
2.3	Puntos MSUGRA	10
2.4	Puntos de referencia MSUGRA	11
2.5	Sistema coordinado	12
2.6	CMS	13
3.1	Canal de estudio	14
4.1	Canal por grupos	21
4.2	Momento transversal g_1	22
4.3	Momento transversal g_2	23
4.4	Energia Transversal Faltante	24
4.5	Separación angular grupo 1	25
4.6	Separacion angular grupo 1	25
4.7	Separación angular grupo 2	26
4.8	Separacion angular grupo 2	26
4.9	Grupo 1, fuente de este trabajo	27
4.10	Grupo 2, fuente de este trabajo	28

Lista de Tablas

2.1	Fermiones del modelo estándar	5
2.2	Mediadores modelo estándar	6
2.3	Partículas ME y partículas SUSY	7
3.1	ID de partículas	15
3.2	Procesos	19
3.3	Parámetros mSUGRA	19
3.4	Partículas que decaerán	20
4.1	Energía grupo 1	27
4.2	Energía grupo 2	28

Glosario

CMS:	Compact Muon Solenoid, Solenoide compacto de muones
ECAL:	Electromagnetic Calorimeters, Calorímetro electromagnético
GUT:	Great Unification Theory, teoría de la gran unificación
HCAL:	Hadron calorimeters, Calorímetro hadronico
LHC:	Large Hadron Collider, Gran colisionador de hadrones
LSP:	Lightest Supersimetry Particle, partícula supersimetrica mas liviana.
MSSM:	Minimal Supersymmetric Standard Model, modelo estándar mínimo supersimetrico
MSUGRA:	Minimal Supergravity, Mínima supergravedad
MUON BARREL:	Detector de muones, formado por varias capas de detectores o multi-capas
NA49:	Gran espectrómetro de aceptancia de trayectoria del SPS del CERN
PARIDAD-R:	Numero cuántico en modelos supersimetricos, que permite identificar las sparticulas y las partículas
SUSY:	Supersymmetry, supersimetría
TRACKER:	Detector de trayectorias

Resumen

El modelo mínimo supersimétrico (MSSM) es un esquema para introducir supersimetría (SUSY) al modelo estándar (ME). Asignando a cada partícula del ME un compañero supersimétrico o s-partícula, una para cada grado de libertad.

En este trabajo se realizará una simulación computacional para colisión protón-protón con energías de centro de masa de 7, 9, 11 y 14 TeV, bajo condiciones experimentales del COMPACT MUON SOLENOID (CMS) para el estudio de un canal en particular con decaimiento en estados finales de taus, con momento transversal, separación angular y energía transversal faltante. Para su estudio se usará los programas Pythia 6 y Root.

Abstract

The minimal supersymmetric standard model (MSSM) is a scheme to introduce supersymmetry (SUSY) to the standard model (SM). Assigning each particle of the SM, a supersymmetric partner or s-particle, one for each degree of freedom.

In this paper a computational simulation of proton-proton collision energies for four center of mass 7, 9, 11 and 14 TeV shall be on, under experimental conditions of the Compact Muon Solenoid (CMS) for the study of a particular channel with decay in final states of tau, with transverse moment. angular separation and missing transverse energy. To study the Pythia 6 programs were used and Root.

Capítulo 1

Introducción

El modelo estándar (ME) de partículas describe apropiadamente una gran variedad de observaciones en física de altas energías.¹ En los tres años de operación, (2010 a 2012) el Large Hadron Collider (LHC) de la European organization for nuclear research (CERN) y el experimento Compact Muon Solenoid (CMS) han realizado una labor destacable. Recolectando un total de $25 fb^{-1}$ de datos a una energía de centro de masa de 8 TeV, la más alta energía alcanzada hasta el momento en un acelerador de partículas.²

Todo esto es posible gracias al avance tecnológico que la humanidad a desarrollado hasta el momento como consecuencia de esto se tiene un mayor conocimiento del modelo estándar y sus componentes. Pero a pesar de los muchos éxitos de esta teoría en las ultimas décadas, este modelo tiene varias limitaciones desde el punto de vista experimental y teórico. Por ejemplo las correcciones cuadráticamente divergentes a la masa del Higgs, no puede explicar la abundancia de materia oscura en el universo y no proporciona la unificación de todas las interacciones. [1]

Por lo tanto es necesario la presencia de física mas allá del modelo estándar, Supersymmetry (SUSY) es una de las teorías mas opcionadas para hacer esto pues incluye todas las interacciones, no tiene problema con la jerarquía de masas, tiene partículas candidatas a materia oscura y trabaja con cinco parámetros libres, tres de masa, uno para el signo y un ángulo, a diferencia del ME que adicional a los problemas ya mencionados se le suma el uso de 17 o 18 parámetros libres. [2]

Uno de los objetivos de este trabajo es simular un canal de decaimiento específico bajo las condiciones necesarias para usar SUSY para ello usamos el software Pythia 6 el cual

¹Colaboradores CMS. Search for new physics in the multijet and missing transverse momentum final state in proton-proton collisions at $\sqrt{s} = 8$ TeV, Journal of high energy physics, Berlin, 2014, p.1.

²Keith A. Ulmer. Future Sensitivity Studies for Supersymmetry Search at CMS at 14 TeV, California, Editorial, 2013, p. 1.

ofrece un generador de eventos de física de altas energías, enfatizando en la producción de multipartículas, en colisiones de partículas elementales, con interacciones fuertes tales como e^+e^- , pp y e^-p . El programa genera eventos completos con tanto detalle como sea posible bajo los modelos desarrollados e implementados que no cubre el ME, en esta simulación se usará una colisión protón-protón, en el marco de trabajo del experimento CMS,³ usando energías de centro de masa de 7, 9, 11, y 14 TeV, en el canal de decaimiento mostrado en la figura 1.1.

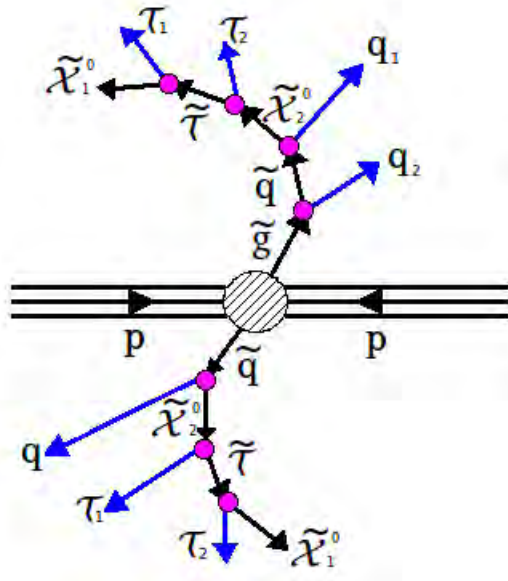


Figura 1.1: Canal de estudio, imagen tomada de R. Cavanaugh, Inclusive SUSY Search at CMS whit Emphasis on Detector Systematics

Este trabajo inicia con una breve introducción del ME, el detector CMS del LHC del laboratorio CERN y SUSY en el capítulo uno, en el capítulo dos se da una descripción mas detallada del ME, SUSY, rompimiento de simetría, parámetros MSUGRA y especificaciones del detector CMS. En el capítulo tres se especifica el canal a simular junto con los programas usados, condiciones y procesos necesarios para este fin. En el capítulo cuatro se hace un análisis de los resultados obtenidos con base en las gráficas de momento transversal faltante, energía transversal faltante y separación angular, en el capítulo cinco se consignan las

³Torbjörn Sjöstrand. Pyhia 6.4 physics and manual, Lund, editorial, 2006, p.4,

conclusiones y recomendaciones pertinentes y finalmente en el apéndice los códigos de los programas que se utilizaron en esta simulación.

Capítulo 2

SUPERSIMETRIA

2.1 Modelo Estándar

Inicialmente se creía que la materia conocida estaba compuesta por pequeñas partes a las cuales se les dio el nombre de átomos pero a medida que la tecnología crecía se descubrió que el átomo estaba compuesto por partes mas pequeñas a las cuales se les llamo neutrones, protones y electrones los dos primeros se encontraban juntos en lo que se denomino núcleo y los electrones girando alrededor de este, pero estas no fueron las únicas partículas que se descubrieron pues cada vez se encontraban mas y mas partículas nuevas, con características y comportamientos desconocidos esto hacia necesario buscar una forma de entender mejor y ordenar estas partículas, así el primer modelo para esto fue el modelo del quark el cual fue postulado primero por Murray Gell-Mann y de forma independiente y simultanea por George Zweig, en este modelo se planteaba la existencia de nuevas partículas los quarks¹. Estos tienen carga fraccionaria, una nueva característica llamada sabor a la cual se le dio los nombres de up, down, charm, strange, top y botom, y para no violar el principio de exclusión de Pauli tienen una característica adicional el color el cual no esta relacionado con el que los seres humanos percibimos con nuestros ojos solamente es una característica adicional. Estos colores son rojo, verde y azul, entonces si un quark es rojo diremos que tiene un valor de 1 en su rojes 0 en su verdura y 0 en su azules, un quark verde tiene 1 en su verdura, 0 en su rojes y 0 en su azules, y de forma similar para un quark azul y para los antiquarks que son las antipartículas de los quarks tenemos un valor de -1 en el color así un antiquark rojo tiene un valor de -1 en su rojes 0 en su verdura y 0 en su azules, de forma análoga ocurrirá para los otros dos antiquarks restantes. De esta manera tenemos tres colores rojo

¹Brayan R Martin, An Introduction and Particle Physics, Jhon Wiley & Sons,2006, p 4, 5 y 6

, verde y azul, seis sabores up, down, charm, strange, top y botom resultando dieciocho quarks, que con sus respectivas antipartículas los antiquarks dan un total de treinta y seis partículas. Ahora era necesario entender sus interacciones y tratar de unificar todo en un solo modelo y es aquí donde el modelo estándar (ME) entra en escena y establece que el universo material esta compuesto de partículas materiales, conocidas como fermiones. Los cuales están organizados por el ME como se muestra en la tabla 2.1.

Fermiones	1 ^{er} Generación	2 ^{da} Generación	3 ^{er} Generación
	Quarks		
	up (u)	charm (c)	top (t)
	down (d)	strange (s)	bottom (b)
	Leptones		
	electrón (e)	muon (μ)	tau (τ)
	neutrino electrónico (ν_e)	neutrino muonico (ν_μ)	neutrino taunico (ν_τ)

Tabla 2.1: Fermiones del modelo estándar

Como se observa en la tabla 2.1 los fermiones están compuestos por seis leptones y seis quarks. Las generaciones hacen referencia a la masa de las partículas así las partículas de la primera generación son mas livianas que las de la segunda y las de la segunda generación mas livianas que las de la tercera generación[3].

Los fermiones interactúan a través de cuatro fuerzas fundamentales estas son denominadas como fundamentales por que con base en ellas se producen las demás fuerzas conocidas entre ellas tenemos (fuerza de fricción, fuerza elástica, fuerza eléctrica, etc). Haciendo una jerarquía entre las interacciones fundamentales con base en su magnitud tenemos que de menor a mayor son: fuerza gravitacional, fuerza débil, fuerza electromagnética y fuerza fuerte. Estas fuerzas interactúan a través de partículas mediadoras llamadas bosones los cuales poseen spin entero, y cada una de ellas es la encargada de mediar cierta interacción, como el ME no tiene en cuenta la interacción gravitacional en la tabla 2.2 se muestra únicamente las partículas mediadoras y su respectiva interacción para el ME.

Bosones	Mediador	Símbolo	Interacción
	Gluon	g	Fuerte
	Bosones vectoriales	W^{\pm}, Z	Débil
	Fotón	γ	Electromagnética

Tabla 2.2: Mediadores modelo estándar

Así tanto partículas como mediadores que intervienen en el modelo estándar se pueden agrupar como se muestra en la figura 2.1

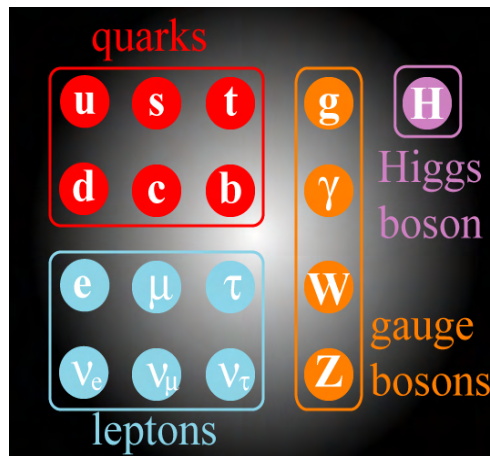


Figura 2.1: Partículas del Modelo estándar, imagen tomada de Carlos Arturo Avila, El Experimento CMS del Acelerador LHC

2.2 Supersimetria

Supersimetria (SUSY) provee un marco de trabajo para construir nuevos modelos para física más allá del ME. SUSY es una teoría cuántica de campos la cual trata de solucionar la mayor cantidad de problemas que presenta el ME. Para ello toma en cuenta la interacción gravitacional, convirtiéndola en una teoría completa, soluciona el problema de la jerarquía de masas, esta dentro de la escala de energía de la teoría de la gran unificación (GUT), trabaja con cinco parámetros libres, introduce un nuevo numero cuántico la R-paridad la

cual es conservada, brindando así una partícula estable al modelo. Donde R-paridad esta relacionada por el numero barionico, leptonico y de espin según la ecuación 2.1

$$R = (-1)^{3B+L+2S} \quad (2.1)$$

Donde R simboliza la R-paridad, la cual sirve principalmente para diferenciar las partículas del ME de las partículas supersimetricas esto teniendo en cuenta su valor, ya que si $R = 1$ es una partícula del ME, pero si $R = -1$ es una partícula SUSY, B representa el numero bariónico, L el numero leptonico y S el numero de spin, garantizando que la light supersimetry particle (LSP) sea estable, que para el Minimal Supersymmetric Standard Model (MSSM) (el MSSM es el marco de trabajo que permite introducir SUSY al ME) es el neutralino χ_1^0 de esta manera se provee un candidato no barionico a materia oscura en el universo los cuales son una mezcla de gauginos neutrales y higgsinos.²

SUSY hace una relación fermión-bosón y bosón-fermión, para transformar partículas del ME en partículas SUSY pues estas solo difieren en su numero cuántico de espin para realizar esto la función del operador es restar 1/2 al spin de las partículas del ME dando como resultado partículas SUSY como se muestra en la tabla 2.3

Partícula	Spin	Spartícula	Spin
quark Q	1/2	squark \tilde{Q}	0
lepton l	1/2	slepton \tilde{l}	0
foton γ	1	Fotino $\tilde{\gamma}$	1/2
gluon G	1	gluino \tilde{G}	1/2
W^\pm	1	wino \tilde{W}^\pm	1/2
Z^0	1	zino \tilde{Z}^0	1/2

Tabla 2.3: Partículas ME y partículas SUSY

Por lo tanto para diferenciar partículas del ME de partículas SUSY se antepuso el prefijo s a los fermiones y el sufijo ino a los gluones, ejemplo un quark del ME se convierte en un

²Andrew Ferstl, Keint A. Olive, Variations of the Neutralino Elastic Cross-Section whit CP Violating Phases, 1999, p. 3

squarks en SUSY y un gluón del ME se convierte en un gluino en SUSY, finalmente para distinguir partículas SUSY de partículas del ME se coloca sobre la letra que las representa el simbolo “ \sim ”, ejemplo a un quark del ME con símbolo q en SUSY le corresponde el simbolo \tilde{q} y para un gluón γ del ME le corresponde en SUSY el gluino $\tilde{\gamma}$.

Así el resumen de partículas SUSY es como se muestra en la figura 2.2

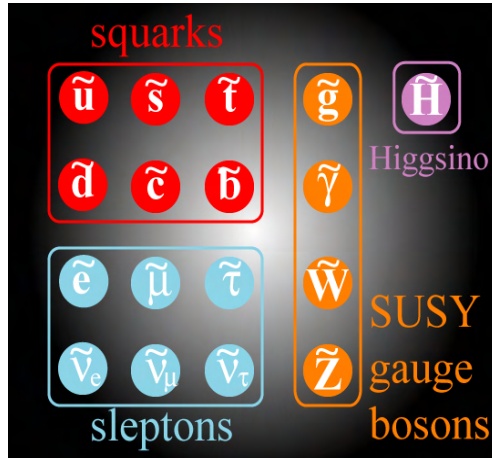


Figura 2.2: Partículas SUSY, imagen tomada de Carlos Arturo Avila, El Experimento CMS del Acelerador LHC

2.3 Rompimiento de supersimetria

Como partículas (SUSY) no han sido descubiertas todavía, estas deben tener masas mucho mayores que las del (MS). Por lo tanto SUSY debe ser rota, dicho rompimiento se puede presentar de dos formas la primera como rompimiento explicito y la segunda como rompimiento espontáneo este ultimo es el mas prometedor teóricamente por que preserva renormalización en el (MS), pero en la practica el rompimiento de SUSY es realizado introduciendo términos que rompan la simetría sin generar nuevas divergencias como el modelo de mínima supergravedad (mSUGRA).³

³Sprengrer Daniel, Search for Supersymmetry in Opposite-sgn Dilepton Final State wiht the CMS Experiment, 2012, p 8.

2.4 Parametros MSUGRA

El modelo de mínima supergravedad (MSUGRA) provee una explicación dinámica de la ruptura de simetría electrodébil que divide la fuerza nuclear débil del electromagnetismo y da masa a los bosones W^\pm y Z^0 . En el ME esto se hace dando masa cuadrado negativa para el campo del Higgs. El cual puede ser considerado artificial. En MSUGRA la ruptura de SUSY naturalmente dispara la ruptura de la simetría electrodébil y conduce a predicciones de masa de sparticulas que estan en el rango de energía de los 100 GeV-TeV.⁴

Los parámetros supersimetricos de espacio en MSUGRA son usualmente estudiados en términos de la masa escalar común m_0 , la masa común del gaugino $m_{1/2}$, la razón del valor esperado del Higgs $\tan \beta$, y el signo del parámetro de masa del Higgsino μ el cual es el termino de masa del superpotencial Higgs cuya magnitud pero no el signo es determinado por las condiciones apropiadas para la ruptura de simetría electrodébil. Por otro lado el quinto parámetro libre el acoplamiento escalar trilinear común A_0 por lo general es cero de ahí que en ocasiones se dice que MSUGRA depende únicamente de cuatro parámetros libres.

Los puntos claves de este modelo son:

- La combinación de las masas bosonicas para la masa escalar común del m_0 .
- Las masas fermionicas para la masa común del gaugino $m_{1/2}$.
- Los acoplamientos escalares trilineales A_0 para la escala GUT

En adición a la unificación de las cuplas gauge, como consecuencia todo el MSSM puede ser descrito solo por cinco parámetros adicionales, para el ME en lugar de más de cien parámetros como en el MSSM general. [4]

Si SUSY es descubierta en el LHC se espera encontrar una gran cantidad de energía transversal faltante (ET_{miss}), Jets y unas grandes tazas de producción de squarks y/o gluinos, las secciones transversales para la producción de un squark (excluyendo al stop) o un gluino se

⁴<http://cerncourier.com/cws/article/cern/28918>

muestran en la figura 2.3 la cual muestra la producción de sección transversal en la región del plano m_0 vs $m_{1/2}$ con decaimientos principales de \tilde{q} y \tilde{g}

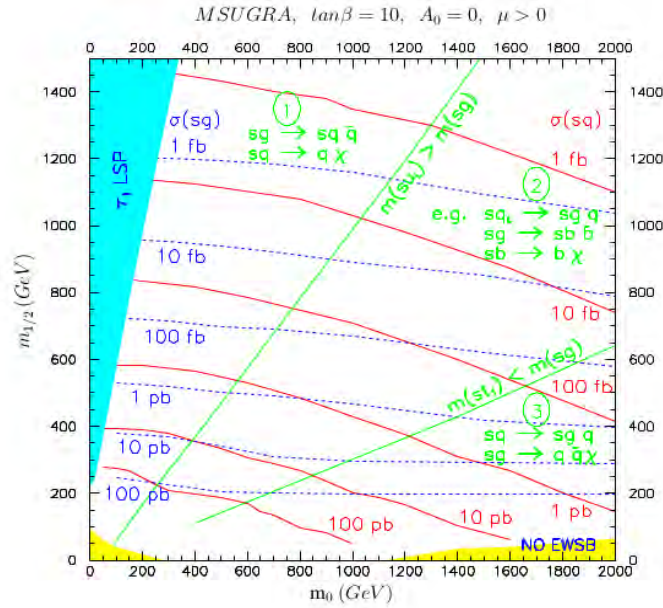


Figura 2.3: Puntos MSUGRA, imagen tomada de CMS Physics Technical Desing Report, Volumen II

En esta figura podemos observar tres regiones delimitadas por líneas diagonales de color verde las cuales ofrecen ciertas características y condiciones de decaimiento.

- Región 1: En esta región los gluinos son mas pesados que cualquiera de los squarks que se produzcan. Las cadenas de decaimiento que producirán las sparticulas se espera sean

$$\tilde{g} \rightarrow \tilde{q} \bar{q}, \quad \bar{q} \rightarrow q \chi \quad (2.2)$$

- Región 2: En esta región algunos quarks son mas pesados, otros son mas ligeros que

el gluino. Resultando cadenas de decaimiento mas complicadas, algunas de ellas son:

$$\tilde{q}_L \rightarrow \tilde{g}q, \quad \tilde{g} \rightarrow \tilde{b}\bar{b}, \quad \tilde{b} \rightarrow b\chi. \quad (2.3)$$

como los \tilde{q}_L de las dos primeras generaciones se esperan sean los mas pesados mientras que \tilde{b}_1 y \tilde{t}_1 los mas livianos.

- Región 3: En esta región los gluinos son mas livianos que los s-quarks y el modo de decaimiento es

$$\tilde{q} \rightarrow \tilde{g}q, \quad \tilde{g} \rightarrow q\bar{q}\chi. \quad (2.4)$$

En la ultima región el gluino da lugar a una descomposición en tres cuerpos mediada por un squark virtual.

En la gráfica 2.4 se representan las tres regiones y los valores de cada uno de los parámetros MSUGRA, los cuales se dividen en dos grandes grupos el primero es el de Low mass LM masa baja que inicia en LM1 y llega hasta LM10 y el otro el de High mass HM masa alta que inicia en HM1 y llega hasta HM4⁵

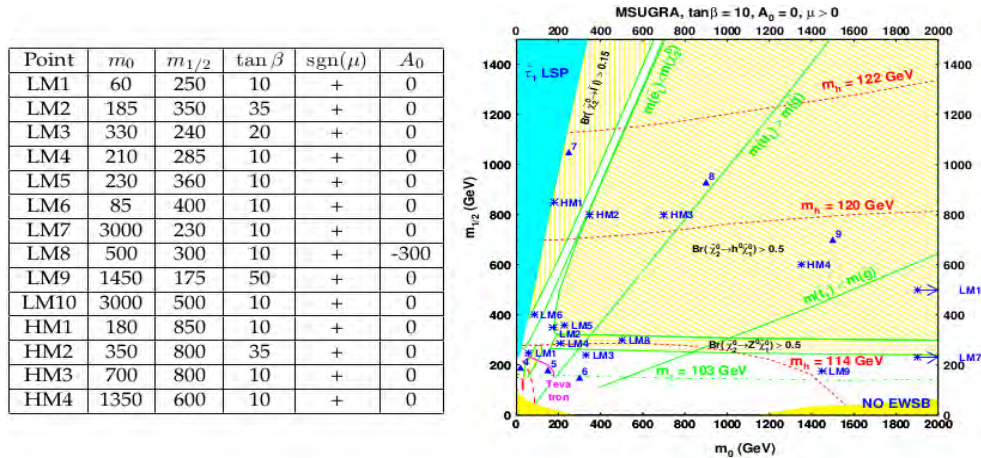


Figura 2.4: Izquierda Puntos de referencia MSUGRA. Derecha Puntos MSUGRA en el plano m_0 vs $m_{1/2}$, tomada de CMS Physics Technical Desing Report, Volumen II

⁵CMS Collaboration, CMS Physics Technical Desing Report, 2006, p 393-399

2.5 CMS

CMS son las siglas en ingles de (Compact muon solenoid, Solenoide compacto de muones) el cual es un detector multiproposito del Large Hadron Collider (LHC) ubicado en el CERN (de sus siglas en francés Conseil Européen pour la Recherche Nucléaire, es decir Consejo Europeo para la Investigación Nuclear). El sistema coordenado del detector esta definido en el centro del mismo con el eje z , en la dirección de avance de las particulas, el ángulo polar θ es medido con respecto al eje z y el ángulo azimutal ϕ (medido en radianes) en el plano perpendicular al plano formado por el eje z y la medidad el ángulo θ ⁶, como se observa en la figura 2.5

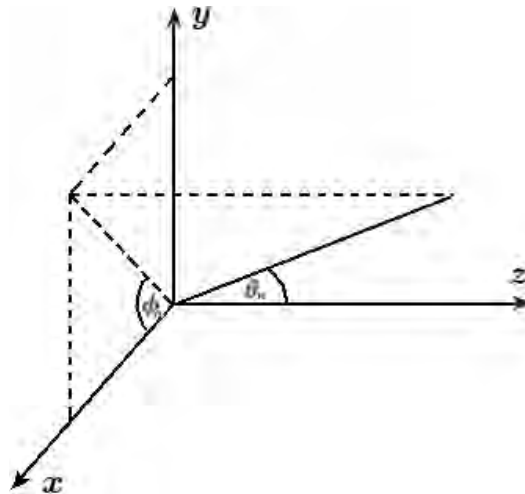


Figura 2.5: Sistema coordenado, imagen tomada de CMS Physics Technical Desing Report, Volumen II

El CMS es un detector multiproposito que investiga nuevas teorías físicas entre ellas tenemos dimensiones extra, partículas candidatas a materia oscura, Boson de Higgs y supersimetría, este detector tiene un diseño multicapas y un gran solenoide superconductor que le provee un campo magnético de $\sim 4 \text{ Teslas}$ cada capa esta diseñada para detectar un tipo específico de partículas.[5]

⁶CMS Collaboration, Search for new physics in the multijet and missing transverse momentum final state in proton-proton collision at $\sqrt{s} = 8 \text{ TeV}$, 2014, p 1

La estructura multicapas del CMS se puede observar en la figura 2.6

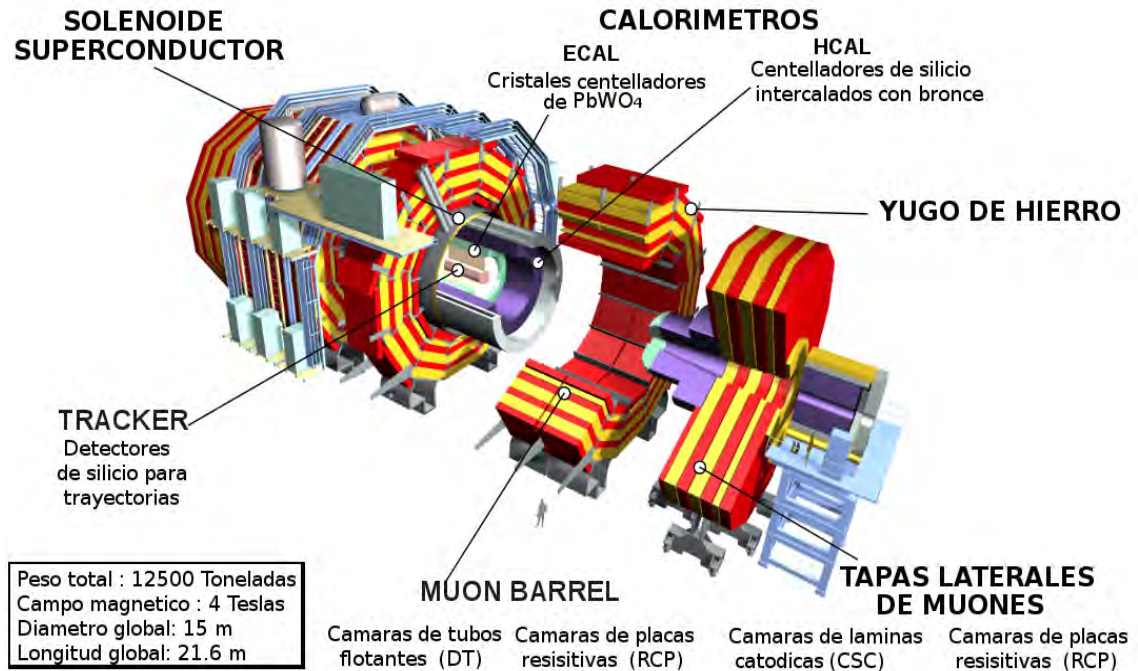


Figura 2.6: CMS, imagen tomada de CMS Grid Computing CERN.

Después de la colisión protón-protón se genera una gran cantidad de partículas que viajan en todas las direcciones y entregan su energía en los diferentes detectores el primero de ellos es el tracking system (sistema de trayectorias) el cual se encarga de reconstruir las trayectorias de las partículas cargadas que a su vez permiten medir su momento, el siguiente sistema es el ECAL (calorímetro electromagnético) este sistema mide la energía de electrones, positrones y fotones, seguido a este está el HCAL (Calorímetro hadrónico) el cual mide la energía de los hadrones, el siguiente dispositivo no es un detector, es el solenoide superconductor que se encarga de generar un campo magnético de $\sim 4 \text{ Teslas}$ obligando a las partículas a tomar trayectorias opuestas dependiendo únicamente de la carga que estas tienen, finalmente tenemos los detectores de muones los cuales se encargan de identificar muones y medir su momento. [6]

programa Root que permite acceder a librerías de programación, análisis de datos e histogramas. Para tener acceso a todas estas herramientas es necesario desarrollar un script donde se especifique una a una las ordenes necesarias para la interfase entre Pythia y Root, teniendo especial cuidado en el orden de los comandos que usan estos programas, como se muestra en el apéndice A se inicia declarando todas las librerías de Pythia, C++ y Root que serán utilizadas en el programa. Seguido a esto se declaran las clases y objetos que usa Pythia para identificar las partículas, tales como nombre, primer hijo, ultimo hijo, momento, masa, tiempo de vida y numero de eventos, para facilitar y estandarizar esta información a cada parámetro se le asigna un numero por ejemplo para el nombre de la partícula se usa el código asignado en el Particle data group (PDG), así a cada partícula le corresponde un único numero para identificarla con excepción de las antipartículas a las cuales les corresponde el mismo numero de las partículas, en este caso en especifico los códigos usados se muestran en la tabla 3.1

Partículas	ID
q y \bar{q}	1-6
τ	15
$\tilde{\tau}$	1000015
\tilde{q}	1000001-1000006
\tilde{g}	1000021
$\tilde{\chi}_1^0$	1000022

Tabla 3.1: ID de partículas

Después de esto se consignan los comandos de Root para crear los histogramas necesarios para el estudio de este canal de decaimiento; se retoman los comandos para Pythia que permiten escoger los procesos mas probables de decaimiento después de la colisión proton-proton, activar interacciones, fragmentaciones, tipo de aproximación SUGRA, valores específicos de los cinco parámetros MSUGRA, desactivación y activación de los decaimientos de cada una de las partículas producidas en la simulación información disponible en el PDG.

En la siguiente etapa del código se especifica las partículas estables, el tipo de colisión y la energía con la que se efectuara la simulación, a continuación se declaran las variables locales para energía, momento, separación angular y demás variables necesarias para los diferentes cálculos. Usando gran parte de las declaraciones hechas con anterioridad se usa los comandos necesarios para especificar el orden y la procedencia de cada una de las partículas del canal de decaimiento, evitando filtraciones de partículas provenientes de otros decaimientos. Finalmente se llenan los histogramas de momento transverso, separación angular y energía transversal faltante.

3.2 Phytia 6

El programa Pythia 6 es frecuentemente usado para la generación de física de altas energías. El cual se enfatiza en la producción de multipartículas para la colisión entre partículas elementales en especial se enfoca en interacciones $e^+ e^-$, $p^+ p^+$ y $e^- p^+$. El programa puede ser utilizado de las siguientes formas:

- Comparar datos experimentales con datos simulados por el programa.
- Correcciones del detector o para estudiar física de altas energías.
- Explorar posibles experimentos que se llevan a cabo o se desarrollaran a futuro.

Phytia 6 esta diseñado para generar eventos completos, con tanto detalle como los ocurridos experimentalmente en el detector, dentro de los limites de la comprensión actual de física de altas energías. La variabilidad natural generada por la mecánica cuántica es reemplazada usando métodos de Monte Carlo para obtener apropiadamente la cantidad de fluctuaciones y un buen comportamiento estadístico. especialmente se enfoca en el desarrollo e implementación de teoría mas allá del ME. Tiene una interfase con ROOT la cual le permite comunicarse con este programa para obtener las gráficas de las diferentes variables de estudio. [7]

3.3 Root

ROOT se desarrollo para el experimento NA49 en el CERN, pues este experimento generaba una gran cantidad de datos, alrededor de 10 Terabytes por corrida. Esta taza de datos proporciono un entorno ideal para desarrollar y poner a prueba un sistema de análisis de datos para futuras generaciones. Root es un software libre para análisis de datos en física de altas energías el cual se desarrollo enfocado en programación orientada a objetos (POO) y en marcos de trabajo (frameworks).

3.3.1 Marcos de trabajo

Los marcos de trabajo son herramientas de programación que se usan con mucha frecuencia por lo tanto ya se encuentran desarrolladas en los programas de tal manera que los usuarios únicamente hacen uso de estos recursos en su beneficio, de esta manera se enfocan en su tarea o problema en especifico y no en diseñar una y otra vez las herramientas necesarias, adicional que en ocasiones no siempre es posible diseñar estas herramientas ya sea por su complejidad o por la cantidad de recursos y tiempo necesario para desarrollarlas. las ventajas mas sobresalientes de los marcos de trabajo son:

- Se escribirá una cantidad mínima de código pues el programador usara códigos existentes que realizan tareas tales como análisis e histogramas.
- El código sera mas consistente y robusto, así el código heredado de un marco de trabajo estará listo para ser probado e integrado con el resto del marco de trabajo
- El código sera mas consistente y modular, la reutilización de código provee consistencia y capacidades comunes entre programas, por lo cual no hay necesidad de reescribir código desde cero. Los marcos de trabajo permiten fraccionar un programa en pequeñas partes.
- Más enfoque en áreas especializadas, Los usuarios pueden concentrarse en un problema en particular sin ser expertos en escribir interfases de usuario, gráficos o la

creación de redes para usar marcos de trabajo que proveen estas herramientas.[8]

3.3.2 Programación orienta a objetos (POO)

La programación orienta a objetos (POO) disminuye el trabajo del programador en cuanto a estar pendiente de todos los procesos del programa pues los objetos son mas autónomos no se necesita reescribir todo el código cuando se necesita hacer ciertas modificaciones lo cual es imprescindible en simulaciones de altas energías por sus constantes cambios lo cual induce a cambios en el código y gracias a la POO, no se tiene que reescribir todo desde cero solo se cambia la parte implicada ahorrando tiempo y recursos. Las características mas importantes de la POO son:

- La encapsulación hace cumplir la abstracción de datos y aumenta la oportunidad para su reutilización.
- Sub-clases y herencia permiten ampliar y modificar objetos
- Jerarquía de clases y jerarquía de contención proporcionan un mecanismo flexible para el modelamiento de objetos del mundo real y las relaciones entre ellos.
- La complejidad se reduce por que hay poco crecimiento en la parte global del programa, pues cada proceso se encuentra dentro de cada objeto en lugar de estar disperso en todo el programa en forma de variables globales.
- Los objetos pueden ir y venir a su antojo, pero la estructura básica del programa sigue siendo relativamente estática, aumentando la posibilidad de reutilización del diseño base. [9]

3.3.3 Procesos

El programa Pythia 6 maneja diferentes formas de trabajo entre ellas tenemos la parte automática donde el programa genera todo tipo de partículas después de la colisión protón protón, pero en este trabajo estamos interesados en un canal específico de decaimiento por lo cual es

necesario hacerlo de forma manual que es la segunda forma de trabajo de este programa lo cual implica. Desactivar todos los procesos que no aportan al canal de decaimiento y activar únicamente los que si lo hacen.

Se inicia con la colisión protón-protón a una de las energía de centro de masa que serán usadas, en el marco de trabajo del CMS donde colisionaran 5.5×10^6 protones que viajan en una dirección y la misma cantidad de protones viajando en dirección opuesta, en este punto se debe tener especial cuidado pues esta colisión genera diferentes procesos de decaimiento de partículas y en este trabajo se busca un proceso en particular por lo tanto es necesario usar un comando en específico dado por $MSUB(ISUB)=1$ donde ISUB es el numero del proceso que se desea, para mas información sobre los posibles procesos revisar el manual. ¹ Para nuestra simulación los procesos necesarios están dados por

Procesos	numero ISUB
$f_i g \rightarrow \tilde{q}_{iL} \tilde{g}$	258
$f_i g \rightarrow \tilde{q}_{iR} \tilde{g}$	259

Tabla 3.2: Procesos

Después de esto es necesario especificar los parámetros MSUGRA con los cuales se va a trabajar

Punto	m_0	$m_{1/2}$	$\tan\beta$	signo μ	A_0
LM2	185	350	35	+	0

Tabla 3.3: Parámetros mSUGRA

Una vez activados los procesos y parámetros MSUGRA con los que se va a trabajar procedemos a especificar las partículas que decaerán en nuestro canal de estudio ya que solo estas partículas nos permitirán construir el canal de decaimiento con los requerimientos necesarios. Para una lista completa de partículas y decaimientos consultar el PDG ² el cual no solo

¹Torbjörn Sjöstrand, Steffen Mrenna, Peter Skand, Pythia 6.4 Physics and Manual 2006, pag 156

²K.A. Olive et al. (Particle Data Group), Chin. Phys. C, 38, 090001 (2014).

muestra esta información si no también la información mas relevante de cada una de las partículas conocidas en física.

En la tabla 3.4 se muestra la partícula el numero que la representa y los códigos de decaimiento necesarios para construir el canal de este estudio.

Partícula	ID	decaimiento
\tilde{g}	1000021	1975-2162
\tilde{d}	1000001	1592-1637
\tilde{u}	1000002	1637-1661
\tilde{s}	1000003	1661-1706
\tilde{c}	1000004	1706-1730
\tilde{b}	1000005	1730-1775
\tilde{t}	1000006	1775-1802
$\tilde{\chi}_2^0$	1000023	2331-2594
$\tilde{\tau}$	1000015	1917-1948

Tabla 3.4: Partículas que decaerán

Capítulo 4

Análisis y resultados

4.1 Canal de decaimiento

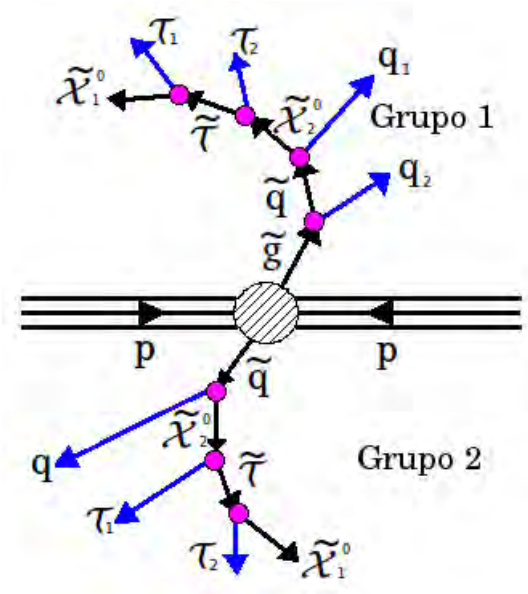


Figura 4.1: Canal de estudio en grupos, imagen tomada de R. Cavanaugh, Inclusive SUSY Search at CMS whit Emphasis on Detector Systematics

Como se muestra en la figura 4.1 el canal de decaimiento escogido se puede trabajar en dos grupos dicha separación se llama algoritmo hemisferio ¹ y permite analizar por separado las cadenas de decaimiento. Así en este trabajo designamos a la parte superior como grupo 1 y a la parte inferior como grupo 2.

¹CMS Collaboration, CMS Physics Technical Desing Report, 2006, p 400

4.2 Momento transversal faltante

Para detectar las partículas que no dejan trazas en los detectores del CMS se usa el momento transversal faltante el cual se define como el aparente imbalance de la componente del momento en el plano perpendicular al eje de avance de la fuente de partículas.²

El método para determinar el momento transversal faltante para colisión de hadrones esta basada en la información que brinda el calorímetro. En el CMS es calculada como la suma del vector negativo de la energía transversa depositada en las torres calorimétricas. Por esta razón se utiliza la letra E_T^{miss} en lugar de P_T^{miss} para el momento transversal faltante, y esta dado por la ecuación³

$$\vec{E}_T^{miss} = - \sum_n (E_n \sin \theta_n \cos \phi_n \hat{i} + E_n \sin \theta_n \sin \phi_n \hat{j}) \quad (4.1)$$

En esta simulación el momento transversal faltante para los neutralinos $\tilde{\chi}_1^0$ y $\tilde{\chi}_2^0$ los cuales no dejan trayectoria o traza en el detector pues no poseen carga eléctrica e interactúan débilmente con el detector, tanto para el grupo uno como para el grupo dos se muestran en las gráficas 4.2 y 4.3 respectivamente.

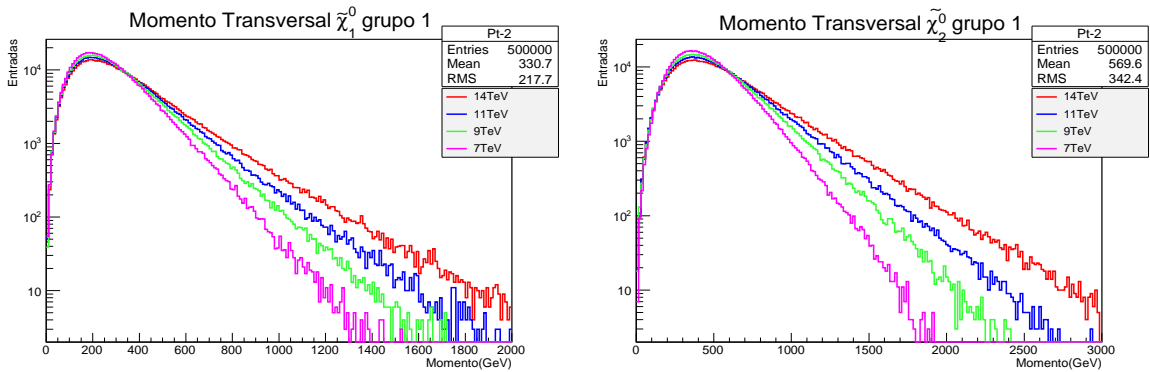


Figura 4.2: Izquierda Momento transversal faltante $\tilde{\chi}_1^0$ derecha Momento transversal faltante $\tilde{\chi}_2^0$ grupo 1, fuente de este trabajo

²CMS Collaboration, Missing Transverse Energy Performance in Minimum-Bias and Jet Events from Proton-Proton Collision at $\sqrt{s} = 7 \text{ TeV}$, 2010, pag 1

³CMS Collaboration. Performance of Track-Corrected Missing E_T in CMS, 2009, pag 1

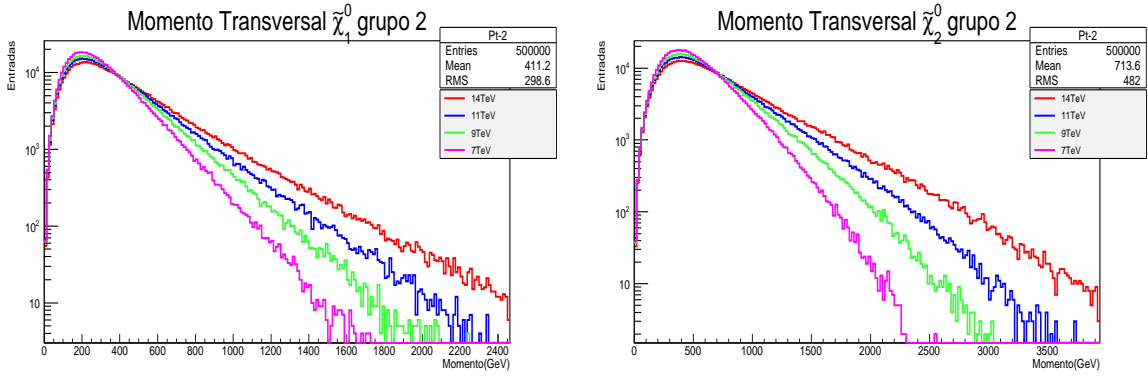


Figura 4.3: Izquierda Momento transversal faltante $\tilde{\chi}_1^0$ derecha Momento transversal faltante $\tilde{\chi}_2^0$ grupo 2, fuente de este trabajo

4.3 Energía transversal faltante

La energía transversal faltante es la magnitud del momento transversal y esta expresada por la ecuación 4.2 ⁴

$$E_t^{miss} = \sqrt{\left(-\sum_n E_n \sin \theta_n \cos \phi_n\right)^2 + \left(-\sum_n E_n \sin \theta_n \sin \phi_n\right)^2} \quad (4.2)$$

La presencia de energía transversal faltante E_T^{miss} es muy importante pues permite discriminar decaimientos leptonicos, física mas allá del ME entre otros, en este caso en específico SUSY pues un valor mayor a 450 GeV en la energía transversal faltante ⁵ sugiere la existencia de partículas supersimetricas. En este trabajo se obtuvo una energía trasversal faltante $E_T^{miss} \approx 924$ GeV como se observa en la gráfica 4.4 esto nos indica que el canal simulado es un canal supersimétrico.

⁴H. Pi, P. Avery, D. Green, J. Rohlf, C. Tully, Measurement of missing trasnverse energy whit the CMS detector at the LHC, 2006, pag 2 y 4.

⁵The CMS Collaboration, Search for supersymmetry in pp collisions at $\sqrt{s} = 8$ TeV in events with a single lepton, large jet multiplicity, and multiple b jets, 2014, pag 5,6 y 7

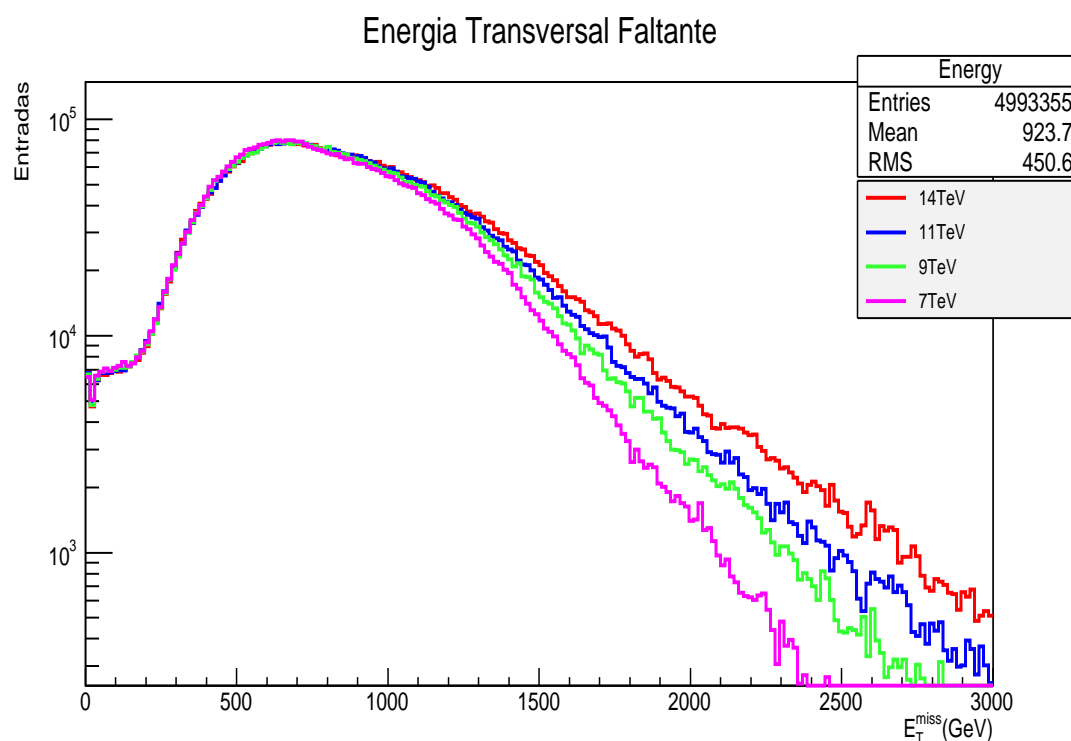


Figura 4.4: Energía Transversal Faltante canal de estudio, fuente de este trabajo

4.4 Separación angular

La separación angular es una variable de bajo nivel la cual nos permite confirmar si la partícula obtenida en la simulación después de los vértices es producto del decaimiento de la partícula predecesora, puesto que la simulación se hace con tanta similitud al experimento que algunas partículas después de los vértices pueden ser las que se buscan pero no proceden del decaimiento de la partícula antes del vértice, así tenemos que si la separación angular entre partículas supera un ángulo de 180° dichas partículas no corresponden al vértice que se busca y por lo tanto tampoco corresponden a nuestro canal de estudio.

En las siguientes gráficas se muestra la separación angular de cada par de partículas después del vértice en el canal de estudio, donde se observa que ninguna de ellas supera los 180° , pues la máxima separación angular es aproximadamente de 98° la cual ocurre entre q_1 y $\tilde{\chi}_2^0$ del grupo 1

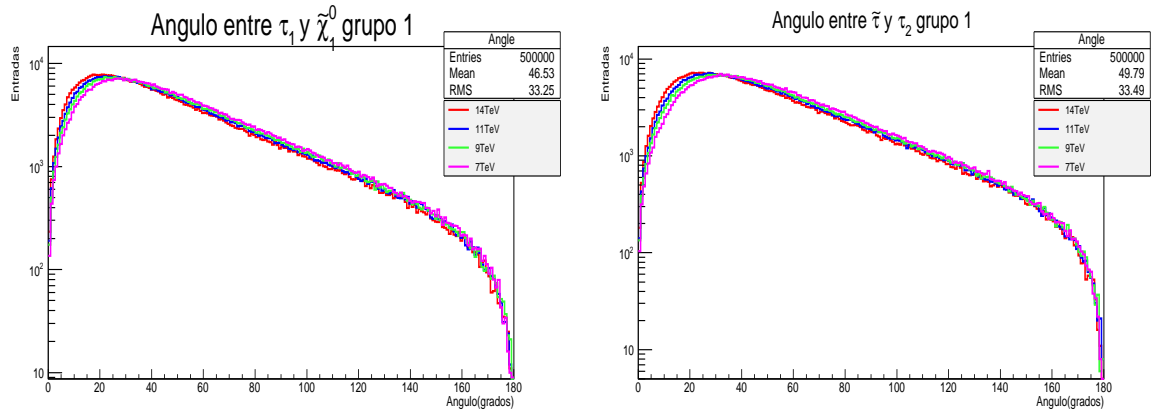


Figura 4.5: Izquierda separación angular entre τ_1 y $\tilde{\chi}_1^0$ grupo 1. Derecha separación angular entre $\tilde{\tau}$ y τ_2 . grupo 1, fuente de este trabajo

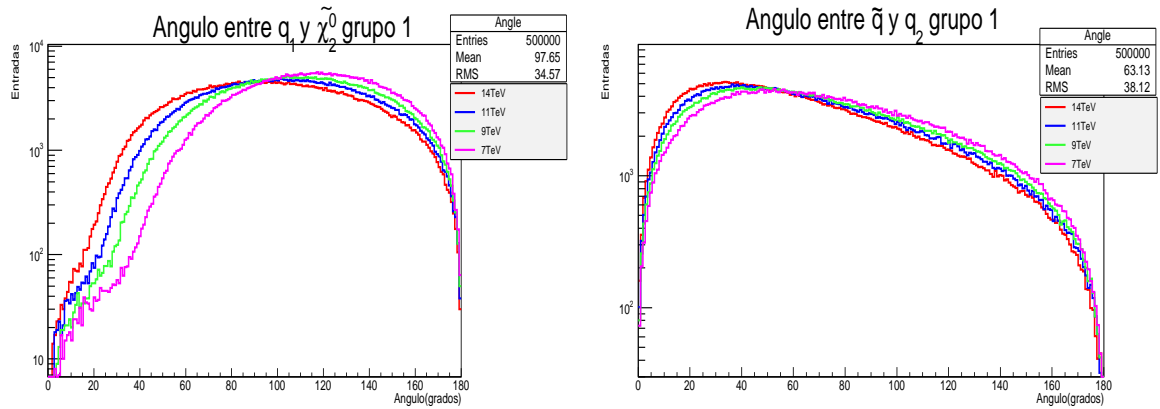


Figura 4.6: Izquierda separación angular entre q_1 y $\tilde{\chi}_2^0$ grupo 1 Derecha separación angular entre \tilde{q} y q_2 grupo 1, fuente de este trabajo

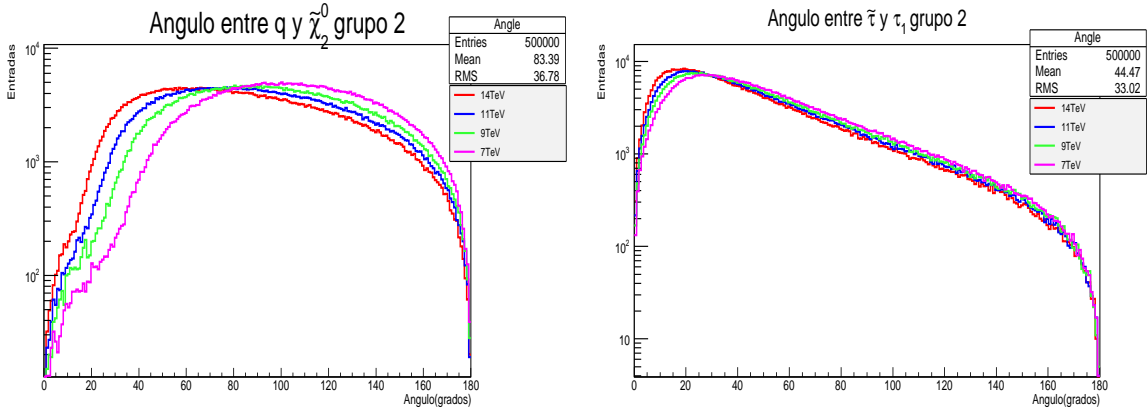


Figura 4.7: Izquierda separación angular entre q y $\tilde{\chi}_2^0$ grupo 2 Derecha separación angular entre $\tilde{\tau}$ y τ_1 grupo 2, fuente de este trabajo

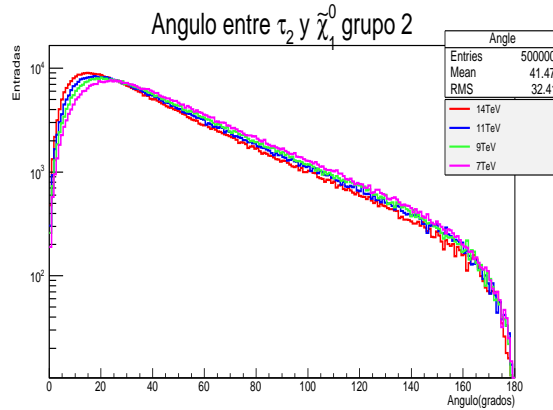


Figura 4.8: Separación angular entre τ_2 y $\tilde{\chi}_1^0$ grupo 2, fuente de este trabajo

4.5 Energía partículas Vs energía centro de masa

Las tablas 4.1, 4.2 y las gráficas 4.9 y 4.10 muestran los valores de energía de cada partícula comparada con el valor de energía a la cual se hizo la simulación, es decir la energía de centro de masa, donde se aprecia una gran diferencia en la energía de una partícula del ME y una partícula SUSY tanto para el grupo uno como para el grupo dos

Grupo 1

Partícula	7 TeV	9 TeV	11 TeV	14 TeV
$\tilde{\chi}_1^0$	309.4 GeV	328 GeV	345.4 GeV	369 GeV
τ_1	90.98 GeV	96.57 GeV	101.6 GeV	108.6 GeV
$\tilde{\tau}$	400.5 GeV	424.7 GeV	447.1 GeV	478.3 GeV
τ_2	136 GeV	144.4 GeV	151.9 GeV	162.6 GeV
$\tilde{\chi}_2^0$	542.5 GeV	575.6 GeV	605.8 GeV	648.3 GeV
q_1	412.3 GeV	436 GeV	459.9 GeV	490.7 GeV
\tilde{q}	957.4 GeV	1015 GeV	1069 GeV	1143 GeV
q_2	116.9 GeV	124.9 GeV	131.5 GeV	140.8 GeV

Tabla 4.1: Energía grupo 1

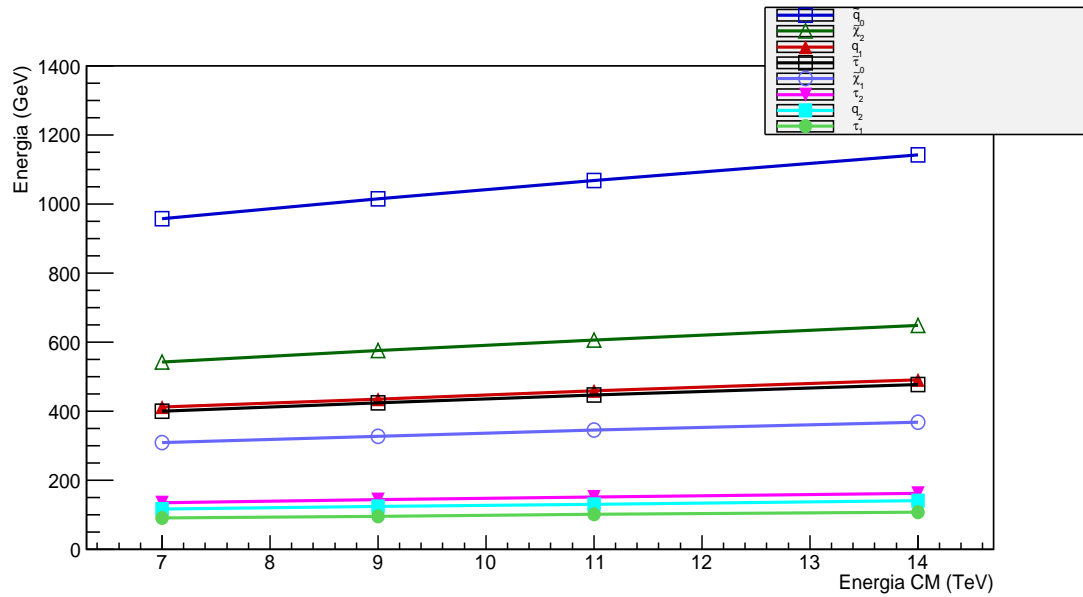


Figura 4.9: Grupo 1, fuente de este trabajo

Grupo 2

Partículas	7 TeV	9 TeV	11 TeV	14 TeV
q	445.4 GeV	483.9 GeV	521.8 GeV	574.7 GeV
$\tilde{\chi}_2^0$	599 GeV	654.1 GeV	706.8 GeV	782.7 GeV
τ_1	150 GeV	164.1 GeV	177.3 GeV	196.2 GeV
$\tilde{\tau}$	442 GeV	482.5 GeV	521.4 GeV	578 GeV
τ_2	100.6 GeV	109.7 GeV	118.3 GeV	131.1 GeV
$\tilde{\chi}_1^0$	341.5 GeV	372.7 GeV	402.7 GeV	445.2 GeV

Tabla 4.2: Energía grupo 2

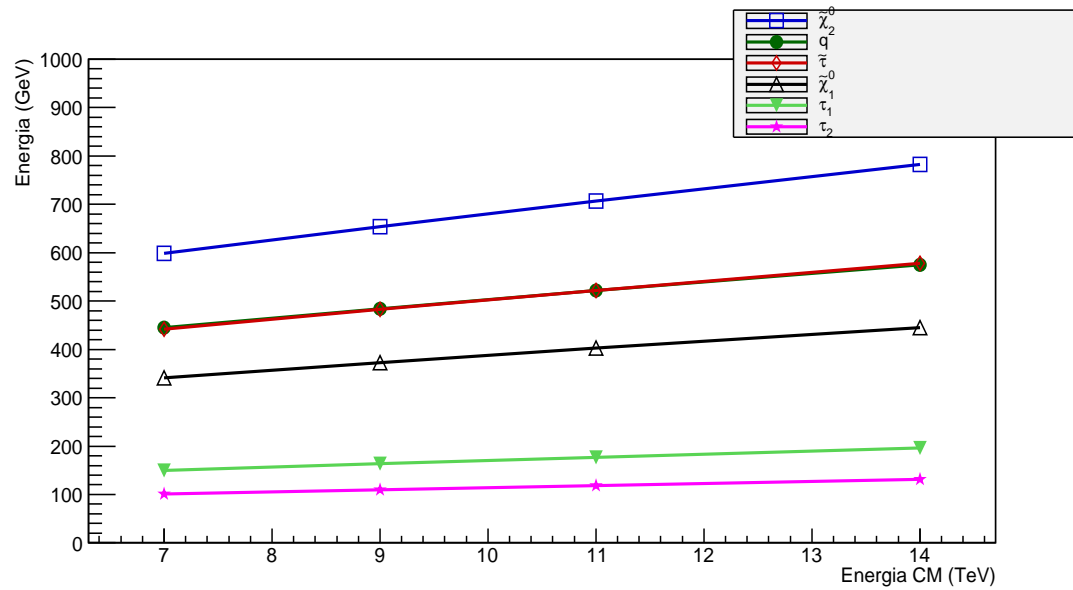


Figura 4.10: Grupo 2, fuente de este trabajo

Capítulo 5

Conclusiones y Recomendaciones

5.1 Conclusiones

- Las variables de bajo nivel separación angular y energía transversal faltante son el primer filtro para confirmar que la simulación es la adecuada, en este trabajo se trabajo con cuatro energías de centro de masa y en todos los casos la separación angular confirmo y la energía transversal faltante que se trata de un proceso supersimétrico
- Otra forma de corroborar que cada una de las partículas simuladas en el canal proviene del vértice correspondiente es que al sumar la energía de las dos partículas que provienen del mismo vértice su resultado debe ser la energía de la partícula madre, por ejemplo si tomamos del grupo 1 las dos ultimas partículas $\tau_1 = 90.98 \text{ GeV}$ y $\tilde{\chi}_1^0 = 309.40 \text{ GeV}$ al sumarlas se espera el valor de la partícula madre $\tilde{\tau} = 400.5 \text{ GeV}$ esto es:

$$\begin{aligned}\tau_1 + \tilde{\chi}_1^0 &\approx \tilde{\tau} \\ 90.98 \text{ GeV} + 309.40 \text{ GeV} &= 400.38 \text{ GeV}\end{aligned}$$

donde es claro ver que la suma 400.38 GeV es aproximada a 400.5 GeV de la partícula madre

- La simulación muestra que existe una gran diferencia entre la energía de una partícula supersimetrica y una partícula del modelo estándar,

5.2 Recomendaciones

- Hacer un estudio con simulación de background ya que una señal supersimétrica es muy sensible a ruido
- Hacer una comparación con los datos que se han obtenido hasta el momento en el LHC para el experimento CMS y cuando este alcance su máxima energía y luminosidad.

Apéndice A

Código canal de decaimiento

```
//=====
//      Histogramas Angulo, Energia y Momento      =
//      Autor Luis H. Echeverry Potosi              =
//      Simulación Pythia 6. Canal Estado Final Taus =
//=====

# ifndef __CINT__
# include <stdlib.h>
# include <TROOT.h>
# include <TRint.h>
# include <TApplication.h>
# include <TFile.h>
# include <TMCParticle.h>
# include <TObjArray.h>
# include <TPythia6.h>
# include <TTree.h>
# include <TClonesArray.h>
# include <TH2.h>
# include <TStyle.h>
# include <TCanvas.h>
# include "Riostream.h"
# include "TNetFile.h"
# include "TRandom.h"
# include "TBranch.h"
# include "TStopwatch.h"
# endif
```

```

void loadLibraries()
{
#ifdef __CINT__
// Load the Event Generator abstraction library, Pythia 6
// library, and the Pythia 6 interface library.
gSystem->Load("libEG");
gSystem->Load("usr/local/lib/root/libPythia6");
gSystem->Load("usr/local/lib/root/libEGPythia6");
#endif
}

struct pythia_particle {
Int_t status;          // status of particle
Int_t pdg_id;          // flavour code
Int_t parent;          // parrent's id
Int_t firstChild;      // id of first child
Int_t lastChild;       // id of last child
Float_t momentum[4];   // X,Y,Z,energy momenta [GeV/c]
Float_t mass;          // Mass [Gev/c^2]
Float_t vertex[4];     // X,Y,Z vertex [mm];
Float_t lifetime;      // proper lifetime [mm/c]
};

int P2ATest(int nEvent=5, int compress=1);
int P2ATest(int nEvent, int compress)
{
TStopwatch fulltime;
Int_t i;
Int_t NEvents;
Int_t startdecay;
Int_t enddecay;
NEvents = nEvent;
// TFileIter MCFFile
TFile* file = new TFile("Sq-q-11TeV-g1.root", "RECREATE",
"Sq-q-11TeV-g1.root", compress);
TPythia6 Pythia;
TTree *tree = new TTree("T", "tree");
TH1F *ener1 = new TH1F("Energy1", "Energy ~q 11 TeV group 1"
, 200, 500, 1800);
TH1F *ener2 = new TH1F("Energy2", "Energy q 11 TeV group 1"
, 200, -10, 400);
TH1F *momentol = new TH1F("Pt-1", "Momentum Transverse ~q 11

```

```

TeV group 1",200,-50,1800);
TH1F *momento2 = new TH1F("Pt-2","Momentum Transverse q 11
TeV group 1",200,-10,400);
TH1F *angu = new TH1F("Angle","Angle q y ~q 11 TeV group 1"
, 200, 0, 180);
TH1F *ang2 = new TH1F("histang2","Angle tau y chi", 300, 0,
3.14);
//=====
//                                PROCESOS                                =
//=====
Pythia.SetMSEL(0);                // subprocessos deben ser activados.
                                   // permitir produccion squark-gluino
Pythia.SetMSUB(258,1);            // f_ig-> ~q_iL~g
Pythia.SetMSUB(259,1);            // f_ig-> ~q_iR~g
Pythia.SetMSTP(61, 1);            // ISR "on"
Pythia.SetMSTP(71, 1);            // FSR "on"
Pythia.SetMSTP(81,0); // seleccion multiiples interacciones
Pythia.SetMSTP(111,1);            // fragmentacion on
Pythia.SetMSTP(82, 3);            // multiple interaction
Pythia.SetPARP(82, 2.41); //cut-off pt multiple interaction
Pythia.SetMRPY(1, 88158204);      //random seed number
Pythia.SetIMSS(1,2);              //aproximacion SUGRA
Pythia.SetMSTJ(11, 3); // seleccion funcion de fragmentacion
Pythia.SetMSTJ(22, 2);            // decay those unstable particles
Pythia.SetMSTJ(1, 1); // parton hadronizacion y produccion
                                   de particulas finales estables
Pythia.SetPARJ(50+4, -0.07); //peterson parameter for charm
Pythia.SetPARJ(50+5, -0.006); //peterson parameter for bottom
Pythia.SetPARJ(50+5, -0.000001); //peterson parameter for top
//=====
//                                PARAMETROS MSUGRA                                =
//=====
Pythia.SetRMSS(8,185);            // masa escalar m0 (squark)
Pythia.SetRMSS(1,350);            // m1/2 masa gaugino
Pythia.SetRMSS(5,35);            // tanbeta (chargino y neutralino)
Pythia.SetRMSS(4,10);            // signo mu
Pythia.SetRMSS(16,0);            // A0 coupling
//=====
//                                CANALES DE DECAIMIENTO                                =
//=====

```

```

for (int i=1; i<4352; i++){ // desabilita todos los canales
Pythia.SetMDME(i,1,0);
}
// Primero desabilita todo
for (int i=1975; i<2162; i++) {
Pythia.SetMDME(i,1,0);
}
// Forzar  $\tilde{g} \rightarrow \tilde{q} + q$ 
Pythia.SetMDME(1976,1,1); //  $\tilde{g} \rightarrow \tilde{d}_L + dbar$ 
Pythia.SetMDME(1977,1,1); //  $\tilde{g} \rightarrow \tilde{d}_{Lbar} + dbar$ 
Pythia.SetMDME(1978,1,1); //  $\tilde{g} \rightarrow \tilde{d}_R + dbar$ 
Pythia.SetMDME(1979,1,1); //  $\tilde{g} \rightarrow \tilde{d}_{Rbar} + d$ 
Pythia.SetMDME(1980,1,1); //  $\tilde{g} \rightarrow \tilde{u}_L + ubar$ 
Pythia.SetMDME(1981,1,1); //  $\tilde{g} \rightarrow \tilde{u}_{Lbar} + u$ 
Pythia.SetMDME(1982,1,1); //  $\tilde{g} \rightarrow \tilde{u}_R + ubar$ 
Pythia.SetMDME(1983,1,1); //  $\tilde{g} \rightarrow \tilde{u}_{Rbar} + u$ 
Pythia.SetMDME(1984,1,1); //  $\tilde{g} \rightarrow \tilde{s}_L + sbar$ 
Pythia.SetMDME(1985,1,1); //  $\tilde{g} \rightarrow \tilde{s}_{Lbar} + s$ 
Pythia.SetMDME(1986,1,1); //  $\tilde{g} \rightarrow \tilde{s}_R + sbar$ 
Pythia.SetMDME(1987,1,1); //  $\tilde{g} \rightarrow \tilde{s}_{Rbar} + s$ 
Pythia.SetMDME(1988,1,1); //  $\tilde{g} \rightarrow \tilde{c}_L + cbar$ 
Pythia.SetMDME(1989,1,1); //  $\tilde{g} \rightarrow \tilde{c}_{Lbar} + c$ 
Pythia.SetMDME(1990,1,1); //  $\tilde{g} \rightarrow \tilde{c}_R + cbar$ 
Pythia.SetMDME(1991,1,1); //  $\tilde{g} \rightarrow \tilde{c}_{Rbar} + c$ 
// forzar  $\tilde{q}_{\{L\}} \rightarrow \chi_{20} + q$ 
// first disable all
for (int i=1592; i<1802; i++) {
Pythia.SetMDME(i,1,0);
}
// then enable squark channel
Pythia.SetMDME(1596,1,1); //  $\tilde{d}_L \rightarrow \tilde{\chi}_{20} + d$ 
Pythia.SetMDME(1641,1,1); //  $\tilde{u}_L \rightarrow \tilde{\chi}_{20} + u$ 
Pythia.SetMDME(1665,1,1); //  $\tilde{s}_L \rightarrow \tilde{\chi}_{20} + s$ 
Pythia.SetMDME(1710,1,1); //  $\tilde{c}_L \rightarrow \tilde{\chi}_{20} + c$ 
// forzar  $\tilde{q}_{\{R\}} \rightarrow \chi_{20} + q$ 
for (int i=3657; i<3888; i++) {
Pythia.SetMDME(i,1,0);
}
Pythia.SetMDME(3661,1,1); //  $\tilde{d}_R \rightarrow \tilde{\chi}_{20} + d$ 
Pythia.SetMDME(3710,1,1); //  $\tilde{u}_R \rightarrow \tilde{\chi}_{20} + u$ 

```

```

Pythia.SetMDME(3738,1,1); //~s_R->~chi_20+s
Pythia.SetMDME(3787,1,1); //~c_R->~chi_20+c
// forzar chi_20->~tau+tau
for(int i=2331;i<2594; i++){
Pythia.SetMDME(i,1,0);
}
Pythia.SetMDME(2421,1,1); //chi_20->~tau+tau
// forzar ~tau->chi_10+tau
for(int i=1917;i<1948; i++){
Pythia.SetMDME(i,1,0);
}
Pythia.SetMDME(1920,1,1); //~tau->chi_10+tau
//=====
//                               Particula en estado Final           =
//=====
// Pythia.SetMDCY(311,1,1); //chi_20 estables
Pythia.SetMDCY(15,1,0); //tau estables
Pythia.SetMDCY(310,1,0); // chi_10 estable
//=====
//                               Inicia la colision                   =
//=====
Pythia.Initialize("CMS", "p", "p", 11000.0);
TStopwatch ioTime;
ioTime.Stop();
//=====
//                               GENERADOR DE EVENTOS                 =
//=====
for ( i=0; i<NEvents; i++ ) {
// Double_t p,E,M,e;
int flag1 = 0;
int flag2 = 0;
Double_t E,E1,E2,E3,E4,m1,m2,m3,m4,masainv,agtt1,agtt2,
angul,angulo;
Double_t px,px1,px2,px3,px4,py,py1,py2,py3,py4,pz,pz1,pz2,
pz3,pz4;
Double_t p,p1,p2,p3,p4;
Double_t c = 1.;
Double_t pi = 4*atan(1.);
Double_t ET=0;
if ( i%1 == 0 ) cout << "Event No.: " << i << endl;

```

```

Pythia.GenerateEvent();
for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
{
//===== Particulas Madres =====

if(Pythia.GetK(Pythia.GetK(nparti,3),2)!=1000021)continue;
if(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),2)!=0)
continue;
if(Pythia.GetK(nparti,2)!= 1000001 && Pythia.GetK(nparti,2)
!=-1000001 &&
Pythia.GetK(nparti,2)!= 1000002 && Pythia.GetK(nparti,2)!=
-1000002 &&
Pythia.GetK(nparti,2)!= 1000003 && Pythia.GetK(nparti,2)!=
-1000003 &&
Pythia.GetK(nparti,2)!= 1000004 && Pythia.GetK(nparti,2)!=
-1000004 &&
Pythia.GetK(nparti,2)!= 2000001 && Pythia.GetK(nparti,2)!=
-2000001 &&
Pythia.GetK(nparti,2)!= 2000002 && Pythia.GetK(nparti,2)!=
-2000002 &&
Pythia.GetK(nparti,2)!= 2000003 && Pythia.GetK(nparti,2)!=
-2000003 &&
Pythia.GetK(nparti,2)!= 2000004 && Pythia.GetK(nparti,2)!=
-2000004 &&
Pythia.GetK(nparti,2)!=1000021);

//===== quiero ~q de 1Pa=~g, 2Pa=0 =====

if(Pythia.GetK(nparti,2)== 1000001 || Pythia.GetK(nparti,2)
== -1000001 ||
Pythia.GetK(nparti,2)==1000002 || Pythia.GetK(nparti,2)==
-1000002 ||
Pythia.GetK(nparti,2)==1000003 || Pythia.GetK(nparti,2)==
-1000003 ||
Pythia.GetK(nparti,2)==1000004 || Pythia.GetK(nparti,2)==
-1000004 ||
Pythia.GetK(nparti,2)==2000001 || Pythia.GetK(nparti,2)==
-2000001 ||
Pythia.GetK(nparti,2)==2000002 || Pythia.GetK(nparti,2)==
-2000002 ||

```



```

Pythia.GetK(nparti,2)==2000003 || Pythia.GetK(nparti,2)==
-2000003 ||
Pythia.GetK(nparti,2)==2000004 || Pythia.GetK(nparti,2)==
-2000004){
px1 = Pythia.GetP(nparti,1);
py1 = Pythia.GetP(nparti,2);
pz1 = Pythia.GetP(nparti,3);
E1 = Pythia.GetP(nparti,4);
p1 = sqrt(px1*px1+py1*py1+pz1*pz1);
flag1 = 1;
cout<<Pythia.GetK(nparti,2)<<"\t 1Pa ="<<Pythia.GetK(Pythia.
GetK(nparti,3),2);
cout<<"\t 2Pa "<<Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,
3),3),2) <<"\n";
momento1->Fill(p1,1.);
ener1->Fill(E1,1.);
}

//===== Quiero q de 1Pa=~g, 2Pa=0 =====

if(Pythia.GetK(nparti,2)== 1 || Pythia.GetK(nparti,2)==-1 ||
Pythia.GetK(nparti,2)== 2 || Pythia.GetK(nparti,2)==-2 ||
Pythia.GetK(nparti,2)== 3 || Pythia.GetK(nparti,2)==-3 ||
Pythia.GetK(nparti,2)== 4 || Pythia.GetK(nparti,2)==-4){
px2 = Pythia.GetP(nparti,1);
py2 = Pythia.GetP(nparti,2);
pz2 = Pythia.GetP(nparti,3);
E2 = Pythia.GetP(nparti,4);
p2 = sqrt(px2*px2+py2*py2+pz2*pz2);
flag2 = 1;
cout<<Pythia.GetK(nparti,2)<<"\t 1Pa ="<<Pythia.GetK(Pythia.
GetK(nparti,3),2);
cout<<"\t 2Pa "<<Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3)
,3),2) <<"\n";
momento2->Fill(p2,1.);
ener2->Fill(E2,1.);
}
} //end loop particle funciona

if(flag1==1 || flag2==1){

```

```

angul=acos((px1*px2+py1*py2+pz1*pz2)/(p1*p2));
Double_t angulo=angul*57.296;
angu->Fill(angulo,1.);
}

tree->Fill();
// Create "MCEvent" object
ioTime.Start(kFALSE);
// Write the "whole" event into ROOT file
int runNumber = 777;
int eventNumber = i;
// delete this "event"
ioTime.Stop();
} // end loop eventos

printf(" Full time: "); fulltime.Print();
printf(" I/O time:  "); ioTime.Print();
printf("\nUsage: root LoadPythia.C 'P2ATest.C(nEvent)'\n");
printf(" -- where nEvent - the total number of the events
to generaten");
//=====
//                               Nombres de Ejes y Lineas                               =
//=====
ener1->GetXaxis()->SetTitle("E_{ ~q} (GeV)");
ener1->GetYaxis()->SetTitle("Entries");
ener1->SetLineWidth(3);
ener2->GetXaxis()->SetTitle("E_{q_{2}} (GeV)");
ener2->GetYaxis()->SetTitle("Entries");
ener2->SetLineWidth(3);
momento1->GetXaxis()->SetTitle("Pt_{ ~q} (GeV)");
momento1->GetYaxis()->SetTitle("Entries");
momento1->SetLineWidth(3);
momento2->GetXaxis()->SetTitle("Pt_{ q_{2}} (GeV)");
momento2->GetYaxis()->SetTitle("Entries");
momento2->SetLineWidth(3);
angu->GetXaxis()->SetTitle("Angle (degree)");
angu->GetYaxis()->SetTitle("Entries");
angu->SetLineWidth(3);
ang2->GetXaxis()->SetTitle("#Delta#phi (rad)");
ang2->GetYaxis()->SetTitle("Entradas");

```

```
//=====
ener1->Write();
ener2->Write();
angu->Write();
ang2->Write();
momento1->Write();
momento2->Write();
}
```

Apéndice B

Energía transversal faltante

Como las condiciones y especificaciones para generar el canal de estudio son las mismas, en esta parte solo se indicaran las modificaciones para obtener la energía transversal faltante

```
//=====
// GENERATOR EVENTS      =
//=====
for ( i=0; i<NEvents; i++ ) {
Double_t Etmiss=0;
Double_t etmx=0;
Double_t etmy=0;
if ( i%1 == 0 ) cout << "Event No.: " << i << endl;
Pythia.GenerateEvent();
for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
{
//===== PADRES =====
if (Pythia.GetK(nparti,1) != 1)continue;
if (Pythia.GetK(nparti,2) == 1000021)continue;
Double_t px = Pythia.GetP(nparti,1);
Double_t py = Pythia.GetP(nparti,2);
Double_t pz = Pythia.GetP(nparti,3);
Double_t E = Pythia.GetP(nparti,4);
Double_t theta = acos(pz/(sqrt(px*px+py*py+pz*pz)));
Double_t phi = atan(py/px);
etmx += E*sin(theta)*cos(phi);
etmy += E*sin(theta)*sin(phi);
Etmiss = sqrt(etmx*etmx+etmy*etmy);
etmiss->Fill(Etmiss,1.);
}
tree->Fill();
// Pythia.Pylist(1); // list particle generate
// Create "MCEvent" object
ioTime.Start(kFALSE);
```

```
// Write the "whole" event into ROOT file
int runNumber = 777;
int eventNumber = i;
// delete this "event"
ioTime.Stop();
//end loop particulas
} // end loop eventos
printf(" Full time: "); fulltime.Print();
printf(" I/O time: "); ioTime.Print();
printf("nUsage: root LoadPythia.C 'P2ATest.C(nEvent)'n");
printf("where nEvent tot number the events to generaten");
etmiss->GetXaxis()->SetTitle("E_{T}^{miss} (GeV)");
etmiss->GetYaxis()->SetTitle("Entries");
etmiss->SetLineWidth(3);
// mass2->GetXaxis()->SetTitle("M_{#mu#mu} (GeV)");
// mass2->GetYaxis()->SetTitle("Entradas");
etmiss->Write();
}
```

Bibliografía

- [1] J. Poveda, en nombre de las colaboraciones ATLAS y CMS, Third generation super-partners: Results from ATLAS and CMS, Bloomington, 2014, p. 1.
- [2] Perkins, Donald H. Introduction to High Energy Physics 4ed, Cambridge University Press, '2000. p. 19
- [3] Halzen, F. Martin A.D Quarks and leptons, New York: Jhon Wiley & Sons, 1984.
- [4] Luisa Sabrina Stark, Petra Hafliger, Adrian Biland y Felicitas Pauss, New allowed MSUGRA parameter space from variations of the trilinear scalar coupling A_0 , Agosto 16 2006, p 3,4 y 5
- [5] CMS collaboration, The CMS experiment at the CERN LHC, 2008, p 3
- [6] Begoña de la Cruz, El bosón de Higgs en el acelerador de partículas LHC (Large Hadron Collider del CERN), 2012
- [7] Torbjörn Sjöstrand, Steben Mrenna, Peter Skand, Pythia 6.4 Physics and Manual
- [8] The ROOT team, ROOT An Object-Oriented Data Analysis Framework, 2009,
- [9] Moreno Francisco, Introducción a la OOP, grupo EIDOS 2000.