

PRODUCCION DE S-QUARKS Y GLUINOS EN EL EXPERIMENTO CMS

LINA HUERTAS GUATIVA

UNIVERSIDAD DE NARIÑO
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE FÍSICA
PROGRAMA DE FÍSICA
SAN JUAN DE PASTO
2009

PRODUCCION DE S-QUARKS Y GLUINOS EN EL EXPERIMENTO CMS

LINA HUERTAS GUATIVA

TRABAJO DE GRADO PARA OPTAR EL TITULO DE FÍSICO

DIRECTOR:
JAIME BETANCOURTH, MSc.

CODIRECTOR:
JUAN CARLOS SANABRIA, PhD.

UNIVERSIDAD DE NARIÑO
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE FÍSICA
PROGRAMA DE FÍSICA
SAN JUAN DE PASTO
2009

Nota de aceptación

Director

Jurado

Jurado

San Juan de Pasto, Noviembre de 2009

“Las ideas y conclusiones aportadas en la tesis de grado son responsabilidad exclusiva del autor”

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

AGRADECIMIENTOS

Agradezco la financiación de COLCIENCIAS, a la facultad de ciencias de la Universidad de los Andes y al departamento de Física de la Universidad de Nariño.

Agradezco a los profesores Juan Carlos Sanabria, Carlos Avila, Bernardo Gomez y Jaime Betancourth, por su tiempo, paciencia y ayuda en el desarrollo de este trabajo.

Al programa de Física de la Universidad de Nariño, a mis compañeros, amigos y profesores.

TABLA DE CONTENIDO

	Pag.
INTRODUCCIÓN	18
1. SUPERSIMETRIA Y LA PRODUCCION DE S-QUARKS Y GLUINOS	20
1.1 EL MODELO ESTÁNDAR	20
1.2 LA SUPERSIMETRIA	22
1.2.1 Modelo Mínimo Supersimetrico	23
1.2.2 Paridad-R y LSP	24
1.2.3 Rompimiento de la Supersimetria	24
1.2.4 mSUGRA	24
1.3 PRODUCCION Y DECAIMIENTO DE S-QUARKS Y GLUINOS	25
1.3.1 Produccion de \tilde{q} y \tilde{g}	25
1.3.2 Decaimientos de \tilde{q} y \tilde{g}	26
1.4 BÚSQUEDA DE SEÑALES DE S-QUARKS Y GLUINOS	27
2. SIMULACIÓN DE LA SEÑAL EXPERIMENTAL DE PRODUCCION DE S-QUARKS Y GLUINOS	30
2.1 PYTHIA	30
2.2 ROOT	30
2.3 SIMULACIÓN IMPLEMENTADA	30
2.3.1 Procesos a Simular	30
3. RESULTADOS	33
3.1 LAS COLISIONES EN EL LHC	33
3.2 ANÁLISIS DEL CANAL	34

3.3 SEÑAL DEL Z^0	37
3.4 SEÑAL DEL $\tilde{\chi}_1^0$	40
3.5 SEÑAL DE LOS JETS	42
4. CONCLUSIONES	43
5. RECOMENDACIONES	44
BIBLIOGRAFÍA	45

LISTA DE CUADROS

	Pág.
Cuadro 1.1 Partículas elementales del Modelo Estándar	20
Cuadro 1.2 Mediadores del Modelo Estándar	21
Cuadro 1.3 Partículas de MSSM	23
Cuadro 2.1 Subprocesos que permiten la simulación de la producción de s-quarks y gluinos	31
Cuadro 2.2 Decaimientos activados para simular el canal mostrado en la figura 1.6	31
Cuadro 2.3 Parámetros SUSY escogidos para el análisis	32

LISTA DE FIGURAS

	Pág.
Figura 1. Canal de decaimiento escogido	19
Figura 1.1 Dos quarks a altas energías se aniquilan en un boson W . El boson W decae en dos quarks de altas energías los cuales se hadronizan emitiendo muchos gluones (lineas en espiral) que decaen a mas quarks. La región gris es donde los quarks forman hadrones, las partículas que son observadas en un detector. Los hadrones finales, colimaran a lo largo de la dirección de sus quarks padres o gluones formando jets. Habrá al menos dos jets de los dos quarks que decayeron del W . Los gluones que decayeron de los quarks deben también producir jets	22
Figura 1.2 Diagramas de Feynman para la producción $\tilde{g}\tilde{g}$ en colisiones hadronicas	26
Figura 1.3 Diagramas de Feynman para la producción $\tilde{q}\tilde{q}$ en colisiones hadronicas	27
Figura 1.4 Diagramas de Feynman para la producción $\tilde{q}\tilde{g}$ en colisiones hadronicas	28
Figura 1.5 Diagramas de Feynman para decaimiento de neutralinos	28
Figura 1.6 Canal usado en este análisis	29
Figura 3.1 Momentum transverso evento subyacente	33
Figura 3.2 Marco de referencia centro de masa parton-parton	34
Figura 3.3 Canal dividido por canales de decaimiento	35
Figura 3.4 Ángulo de separación entro los s-quarks y gluinos	35
Figura 3.5 Separación angular entre partículas del grupo 1	36
Figura 3.6 Separación angular entre partículas del grupo 2	37
Figura 3.7 Masa invariante de los dos pares de muones	38
Figura 3.8 Masa invariante de dimuones para el background y para la señal (rojo). Las líneas verticales denotan los requerimientos de masa impuestos	39
Figura 3.9 Ángulo que separa al par de muones	39
Figura 3.10 $\Delta\phi$ entre los dos leptones para el bacground	40
Figura 3.11 Energía Transversal Faltante	41
Figura 3.12 (a) Energía Transversal Faltante , la linea vertical indica los requerimientos de E_T^{miss} y (b) $M_{\mu\mu}$ después del corte sobre E_T^{miss} ; para el background y la señal (rojo) ¹⁵	41
Figura 3.13 Momentum transverso para los jets del canal	42

LISTA DE ANEXOS

	Pág.
Anexo A. Códigos utilizados para la simulación	46
Anexo A.1. Marco de referencia centro de masa partonico	46
Anexo A.2. Ángulos de separación entre partículas del grupo1	49
Anexo A.3. Ángulos de separación entre partículas del grupo 2	52
Anexo A.4. Masa invariante y separación angular disminuones	55
Anexo A.5. Energía Transversal Faltante	57

GLOSARIO

- BACKGROUND:** Señal no deseada proveniente del Modelo Estándar o de efectos del detector que podría simular la señal supersimétrica.
- CMS:** Solenoide compacto de muones, es un detector de partículas que se encuentra en el LHC.
- CORTES CINEMÁTICOS:** Filtros que sirven para separar señal supersimétrica del background.
- EVENTO SUBYACENTE:** Los remanentes de los protones en una colisión proton-proton a grandes energías.
- GUT:** Teoría de Gran Unificación.
- JET:** Un chorro de partículas, hadrones, que se mueven a lo largo del quark original.
- LHC:** El gran colisionador de hadrones, que hará colisionar haces de protones a energías de 14 TeV en el CERN, en Suiza.
- LSP:** Partícula supersimétrica mas liviana.
- MSSM:** El Modelo Mínimo Supersimétrico.
- MSUGRA:** Modelo de supergravedad.
- PYTHIA:** Software, que permite simular eventos de altas energías.
- ROOT:** Software, que permite analizar datos experimentales.
- SUSY:** Supersimetría.

RESUMEN

La simulación de una de las señales experimentales de la producción de s-quarks y gluinos en colisiones protón-protón a energía de centro de masa de 14TeV, usando el software PYTHIA, es presentada. El estudio consiste en analizar la señal experimental que deja el decaimiento de un s-quark y un gluino vía neutralinos y bosones Z que a su vez decaen en un par de muones de signo opuesto. Los muones están acompañados por tres jets y energía transversal faltante. Con este trabajo se quieren estudiar las características cinemáticas de las señales detectables en el estado final. Los resultados obtenidos ayudan a establecer criterios para la búsqueda de este canal en el experimento CMS y determinar cortes cinemáticos que permitan separar la señal supersimétrica del "background" proveniente del modelo estándar.

ABSTRACT

The simulation of one of the experimental signals of the production of s-quarks and gluinos in proton-proton collisions at a center of mass energy of 14 TeV using the PYTHIA software. The final state signal that is analyzed, results from the decay of an s-quark and a gluino via neutralinos and Z bosons, the Z bosons decay in opposite sign muons. We also find in the final state 3 jets and missing transverse energy. With this work I want to study the kinematic features of the detectable signals in the final state. The conclusions of this study will help in the search of this channel in the CMS experiment and set kinematic cuts which allow a separation between supersymmetric signal and the background coming from the standard model.

INTRODUCCIÓN

En los últimos años se ha comprobado la eficacia del Modelo Estándar (ME) como una teoría que ha predicho la existencia de partículas y de sus propiedades, este modelo describe tres de las cuatro fuerzas fundamentales: electromagnética, débil y fuerte. Pese a su éxito se sabe que el modelo falla en la descripción de algunos fenómenos y presenta problemas en su extensión al régimen de altas energías. Extensiones del ME tratan de explicar estas inconsistencias y agregar simplicidad y "belleza" al modelo; una de tales extensiones es la Supersimetría (SUSY).

SUSY predice la existencia de otras partículas además de las del ME; por ejemplo, para cada boson del ME debe existir un supercompañero fermionico y de igual forma para cada fermion un supercompañero bosonico. Ninguna partícula SUSY ha sido encontrada todavía, si los compañeros supersimetricos existen deberían tener masas en el rango de TeV. El experimento CMS, del acelerador LHC en el laboratorio CERN, busca estas partículas en colisiones protón-protón a energías de 14 TeV, esto abre la posibilidad para el descubrimiento de nueva física.

En este trabajo se realiza una búsqueda de señales de producción de s-quarks y gluinos en colisiones protón-protón a energía centro de masa de 14 TeV. La señal experimental que se usa para esta búsqueda esta compuesta por eventos con muones, jets y energía transversal faltante. La búsqueda es realizada mediante simulaciones computacionales, usando el programa PYTHIA, donde se incluirá las características de detección del experimento CMS.

Hay muchos canales de decaimiento para estas partículas, resultando en diferentes señales experimentales; por lo que se utiliza el canal con señal experimental mas sencilla, en este caso corresponde al decaimiento de s-quark y gluino vía neutralinos y bosones Z que decaen en un par de muones de signo opuesto; por lo tanto, el objetivo principal de este estudio, es analizar las características cinemáticas de las señales detectables en el estado final. Este canal en particular es mostrado en la figura 1.

Hay un gran numero de procesos del ME y efectos del detector que producen eventos que se asemeja a la señal supersimetrica, a esto se le conoce como *background*. En este estudio no se tendrá en cuenta el background, ya que se sale del alcance de este trabajo, solamente se establecen cortes que ayudarían a separar la señal supersimetrica del background en trabajos posteriores.

Este trabajo esta organizado de la siguiente manera: comenzando el capitulo 1 presenta una breve explicación de la estructura del ME y SUSY, y luego se analiza la producción de partículas supersimetricas, s-quarks y gluinos, y la estrategia para su búsqueda. El software que se utiliza para la búsqueda de estas señales sera descrito en el capitulo 2; y el análisis y los resultados de la simulación son presentados en el capitulo 3. Por ultimo el capitulo 4 presenta las conclusiones que se llegaron con este trabajo.

1 SUPERSIMETRIA Y LA PRODUCCIÓN DE S-QUARKS Y GLUINOS

En la física de partículas el modelo mas exitoso hasta el momento ha sido el Modelo Estándar, ciertamente ningún experimento hasta el momento ha sido capaz de contradecir alguna de sus predicciones. Sin embargo, el ME es una teoría complicada y contiene algunas inconsistencias, creyendo que hay una teoría mas haya de este modelo, de la cual este hace parte, una de estas teorías es SUSY.

Tanto el ME como SUSY son modelos basados en herramientas matemáticas muy elaboradas. Este trabajo no pretende entrar en detalle con su formulación teórica sino mas bien analizar sus resultados experimentales, por lo que en este capitulo se presenta solamente una breve descripción de la estructura de ME, SUSY y de la producción y decaimiento de estas partículas adicionales predichas por SUSY, s-quarks y gluinos.

1.1. EL MODELO ESTÁNDAR

El Modelo Estándar (ME) es el modelo mas preciso para entender las partículas y sus interacciones. En resumen, el ME es una teoría cuántica de campos la cual describe la unificación de las interacciones electromagnéticas y débiles en un sector electrodebil, y contiene un sector de Cromodinamica Cuántica (QCD) para la descripción de las interacciones fuertes. Sin embargo, el ME tiene algunos problemas como se describirán mas adelante.

El sector fermionico del ME consiste de quarks y leptones como se muestra en la cuadro 1.1 y esta organizado en 3 familias (generaciones) con idénticas propiedades excepto por la masa. En adición, la cuadro 1.2 muestra los diferentes bosones gauge disponibles en el marco del ME, los cuales son los responsables de las principales fuerzas presentes en la naturaleza; 8 gluones sin masa para las interacciones fuertes, un foton sin masa para la interacción electromagnética y 3 bosones débiles masivos para la interacción débil. La gravedad es la fuerza que no esta incluida en el ME y esto es una de las principales motivaciones para la búsqueda de otras teorías.¹

Cuadro 1.1: Partículas elementales del Modelo Estándar

Generación	I	II	III
Quarks	up (u) down (d)	charm (c) strange (s)	top (t) bottom (b)
Leptones	electrón (e) neutrino electrónico (ν_e)	muon (μ) neutrino muonico (ν_μ)	tau (τ) neutrino taunico (ν_τ)

Otra partícula importante del ME es el boson de Higgs, este es introducido por el mecanismo de Higgs, este mecanismo rompe espontáneamente la simetría del sector electrodebil a través del valor esperado en el vacío del campo de Higgs y como resultado

¹MORII, T., LIM, C.S., MUKHERJEE, S.N., The Physics of the Standard Model and Beyond, 2004.

Cuadro 1.2: Mediadores del Modelo Estándar

Mediador	Símbolo	Interacción
Gluon	g	Fuerte
Bosones Vectoriales	W^\pm, Z	Débil
Foton	γ	Electromagnética

los bosones gauge W^\pm y Z se convierten en masivos. No solamente los bosones gauge adquieren masa mediante el rompimiento de simetría, sino también todos los fermiones del ME.

En la naturaleza existen 4 interacciones fundamentales, la interacción electromagnética, débil, fuerte y gravitacional. Entre estas interacciones, la interacción gravitacional es extremadamente débil comparada con las otras interacciones, por esto no es descrita por el ME.².

En la interacción fuerte los quarks permanecen juntos en hadrones tales como protones y neutrones. Las partículas mediadoras de esta interacción son los gluones, estas afectan todo lo que tiene carga de color, la cual es llevada por los quarks y gluones por si mismos. Los quarks y gluones nunca son observados como partículas libres. Si un quark en un protón tiende a escapar (posiblemente debido a una colisión a gran energía con otro quark), la energía potencial de la fuerza fuerte sera tan grande que un nuevo par de quark-antiquark aparecerá fuera del vacío, uno que se une al quark que escapo formando un meson y otro permanece con los remanentes del protón. Al final, todo lo que se observa son hadrones sin color. Este proceso es llamado *hadronizacion* o *fragmentación*. Los quarks en altas energías producirán tantos nuevos quarks y antiquarks como ellos escapan, como se muestra en la figura 1.1. Por lo tanto en un detector, un quark a altas energías es visto como un chorro de partículas o *jet* colimado de hadrones moviéndose a lo largo de la dirección del quark original.

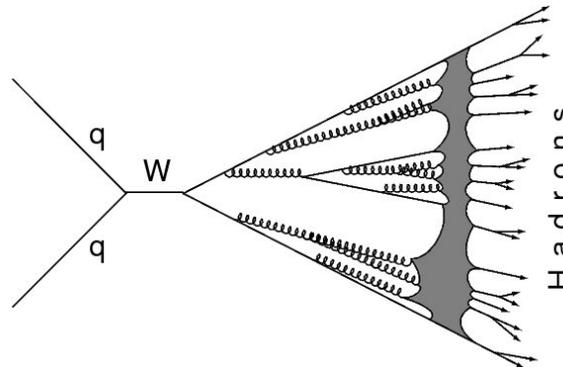
La interacción débil es mediada por tres tipos de partículas llamadas Z^0 , W^+ y W^- , que interactúan con todas las partículas que tienen un tipo de carga conocida como isospin débil, esta incluye todos los quarks y leptones y también los Z^0 y W^\pm por si mismos.

Todas las partículas anteriormente descritas (fermiones y bosones) han sido descubiertos en experimentos. Desafortunadamente el higgs no ha sido encontrada todavía, quizás porque es muy pesada; este es uno de los problemas mas urgentes en la física de partículas.

Uno de los principales problemas del modelo estándar tiene que ver con la masa del campo de Higgs. El esquema de renormalizacion para la masa de esta particula incluye loops fermionicos que producen integrales cuadráticamente divergentes. Si bien estas divergencias se pueden renormalizar, su comportamiento altamente patológico es insatisfactorio a muy altas energías.

²HALZEN, F, MARTIN, A.D., Quarks and Leptons, New York: John Wiley & Sons, 1984.

Figura 1.1: Dos quarks a altas energías se aniquilan en un boson W . El boson W decae en dos quarks de altas energías los cuales se hadronizan emitiendo muchos gluones (líneas en espiral) que decaen a mas quarks. La región gris es donde los quarks forman hadrones, las partículas que son observadas en un detector. Los hadrones finales, colimaran a lo largo de la dirección de sus quarks padres o gluones formando jets. Habrá al menos dos jets de los dos quarks que decayeron del W . Los gluones que decayeron de los quarks deben también producir jets.



El ME tiene serios problemas teóricos, empezando del hecho que la gravedad no es acomodada en la teoría. El ME tiene al menos 19 parámetros libres, tales como acoplamientos, masas y mezclas; los cuales no pueden predecirse pero deben ser medidos experimentalmente.

El ME también deja muchas preguntas sin resolver, tales como porque hay tres generaciones?, como entendemos las oscilaciones del neutrino y los neutrinos masivos?, porque la carga eléctrica del protón y la del electrón son exactamente iguales y opuestas? o si el mecanismo de Higgs es realmente el proceso a través del cual se rompe la simetría electrodébil y establece el origen de las masas. Además, el modelo no puede explicar cual es el mecanismo que produce la materia-antimateria observada en el universo o cual es la relación entre las fuerzas fuerte y electrodébil.

1.2. LA SUPERSIMETRIA

Es necesario tener una extensión del ME consistente con todos los datos experimentales y predicciones de nueva física a una escala sobre 1 TeV. Uno de los candidatos mas favorecidos para nueva física es SUSY.

La supersimetría es un tema de bastante interés para los físicos. No solo por su elegancia, sino porque también se intuye que tendrá gran protagonismo en nuevos descubrimientos en física de partículas. A continuación se presenta una descripción no rigurosa de la supersimetría con el fin de entender un poco mas su mecanismo de funcionamiento.

SUSY es una simetría propuesta entre fermiones y bosones. Los operadores que generan las transformaciones, transforman estados de spin $|S\rangle$ en estados de spin $|S \pm 1/2\rangle$.

Cuadro 1.3: Partículas de MSSM.

Partículas ME			S-partículas		
Nombre	Símbolo	Spin	Nombre	Símbolo	Spin
quark	q	1/2	s-quark	\tilde{q}	0
lepton	ℓ	1/2	slepton	$\tilde{\ell}$	0
gluon	g	1	gluino	\tilde{g}	1/2
bosones W	W^\pm	1	winos	\tilde{W}^\pm	1/2
boson Z	Z^0	1	zino	\tilde{Z}^0	1/2
boson Higgs	H	0	higgsinos	\tilde{H}	1/2
graviton	G	2	gravitino	\tilde{G}	3/2

El hecho de que no se hayan observado partículas SUSY implica directamente que SUSY debe ser una simetría rota, al menos para esta escala de energía, porque de lo contrario ya se habrían podido observar partículas supersimétricas correspondientes a las partículas conocidas.³

1.2.1. Modelo Mínimo Supersimétrico. El Modelo Mínimo Supersimétrico MSSM (the Minimal Supersymmetric Standard Model) es un esquema para introducir supersimetría al ME que agrega pocas nuevas partículas. Cada partícula del ME recibe un compañero supersimétrico o s-partícula, una por cada grado de libertad.

Las partículas incluidas en el MSSM son listadas en la cuadro 1.3. The MSSM contiene un mínimo número de nuevas partículas. Las partículas supersimétricas son comúnmente llamadas s-partículas (*supersymmetric particles*). Los compañeros fermiónicos (s-fermiones) tienen un prefijo *s* (partícula escalar, por ejemplo s-quarks, s-leptones...), mientras que los supercompañeros de los bosones (gauginos) tiene el sufijo *ino* (por ejemplo gluino, wino...)⁴

El *gluino* de spin 1/2 es el compañero supersimétrico del gluon, el mediador de la fuerza fuerte. El compañero supersimétrico del quark es llamado *s-quark*, el s-quark tiene las mismas interacciones como el quark, solamente que el s-quark es un boson con spin 0. Hay dos grados de libertad para el quark, el número de grados de libertad para el s-quark debe ser el mismo que su compañero, el quark. Los compañeros supersimétricos de los bosones gauge *W*, *Z*, y Higgs son llamados *gauginos* y *higgsinos*.⁵

Las s-partículas del MSSM mostradas en la cuadro 1.3 representan los auto-estados gauge en la teoría. Los auto-estados de masa observados experimentalmente están formados por la mezcla de estos auto-estados, combinaciones de fermiones supersimétricos con los

³LYON, Adam, A Search for Squarks and Gluinos using the Jets and Missing Energy Signature at D0, University of Maryland, 1997.

⁴MOHR, Nickas. Neutralino Reconstruction in Dilepton Final States with the CMS Experiment. Rheinisch-Westfälischen Technischen Hochschule Aachen, 2008.

⁵GOFORTH, Mark. Search for Squarks and Gluinos with D0 Detector. The Florida State University, Department of Physics, 1995.

mismos números cuánticos. Se tienen gauginos neutrales, llamados *neutralinos* ($\tilde{\chi}_{1-4}^0$), los cuales son estados mezclados de las partículas gauge neutrales (wino, bino, higgsinos neutrales). Los cuatro auto-estados de masa gauginos cargados son llamados *charginos* ($\tilde{\chi}_{1,2}^\pm$) y son obtenidos por la mezcla de gauginos cargados (winos cargados, higgsinos cargados). Estas partículas son ordenadas por su masa, indicado por el índice de la partícula, donde la masa incrementa con el índice.

1.2.2. Paridad- R y LSP. En modelos supersimétricos un nuevo número cuántico es introducido, el cual permite distinguir partículas del ME y supersimétricas. Esta es llamada la paridad- R y se define como:

$$R = (-1)^{3B+L+2S} \quad (1.1)$$

donde B y L se refieren a los números barionicos y leptonicos de la partícula, respectivamente y S es el spin. Dado que la diferencia en spin entre las partículas SUSY y las del ME es de $1/2$, las partículas del ME tienen $R = 1$ mientras que $R = -1$ para las s-partículas. Su conservación tendrá tres importantes implicaciones:

- S-partículas pueden ser producidas solamente en pares, si el estado inicial esta formado de materia ordinaria.
- La partícula supersimétrica mas liviana, LSP (*Lightest Supersymmetric Particle*) es estable; dado que esta tiene $R = -1$ no puede decaer en partículas del ME.
- S-partículas finalmente decaen en partículas del ME y LSP.

La LSP es el neutralino mas liviano, $\tilde{\chi}_1^0$. Este debe ser neutral porque las LSPs cargadas tendrían que haberse visto en física atómica y a nivel cosmológico. La LSP solamente interactúa débilmente, como los neutrinos, produciendo energía faltante (E_T^{miss}) en un detector, la cual puede ser usada como una señal experimental para SUSY.⁶

1.2.3. Rompimiento de la Supersimetría. En una teoría supersimétrica sin romper, los supercompañeros tendrán masas degeneradas con los compañeros del ME. Este obviamente no es el caso, de otra manera estas partículas se habrían observado. Por lo tanto algún mecanismo debe romper SUSY. Hasta el momento, no se ha llegado a un consenso sobre cual debe ser este método. Existen alternativas de rompimiento de la supersimetría entre las que se encuentran los modelos de mínima supergravedad (mSUGRA). Estos métodos de rompimiento son los mas utilizados para estudios de SUSY.

1.2.4. mSUGRA. Una reducción del número de parámetros libres es posible si se asume que los parámetros no son todos independientes. En el modelo de supergravedad, el campo gravitacional es supersimétrico. El compañero del graviton es llamado *gravitino*.

⁶Al ser pesada, la LSP es un buen candidato de materia oscura.

Aparte de utilizar este mecanismo, al utilizar mSUGRA se hacen otras suposiciones basadas en teorías de gran unificación (GUT, *Grand Unification Theory*), para así poder justificar la asignación de clases de masas y acoplamientos iguales en la escala GUT. Fundamentalmente, estas suposiciones permiten reducir el número de parámetros necesarios para poder caracterizar la supersimetría de aproximadamente 105 a 5. Estos son:

- m_0 : masa común para todos los escalares en la teoría: s-fermiones y escalares higgs de acuerdo con la escala GUT.
- $m_{1/2}$: masa común para todos los fermiones en la teoría: fermiones del ME, susy fermiones (gauginos) y higgsinos de acuerdo con la escala GUT.
- A_0 : acoplamiento trilineal (refiriéndose a que el acoplamiento se hace entre 3 campos).
- $\tan\beta$: razón entre los valores esperados en el vacío de los dobletes de Higgs. Se ha observado que su valor está entre 5 y 40.
- μ : es el parámetro de masa para los higgsinos. Se asume que su valor es real y que su magnitud está dada por otros parámetros, por lo que solo se especifica su signo.

1.3. PRODUCCIÓN Y DECAIMIENTO DE S-QUARKS Y GLUINOS

El experimento CMS del CERN, es el mejor lugar para la búsqueda de producción directa de nuevas partículas todavía no observadas. Si las partículas supersimétricas existen, se espera que se revelen vía exceso de estados finales con multi-jets + E_T^{miss} (Energía Transversal Faltante) + multi-leptones, comparados con lo que se espera del ME. De toda la variedad de partículas que el nuevo marco del MSSM introduce, dos tipos de ellas son de especial relevancia para este estudio, s-quarks y gluinos. En esta sección se discutirá sobre cómo producirlos, sus decaimientos y la estrategia para la búsqueda de estas partículas.⁷

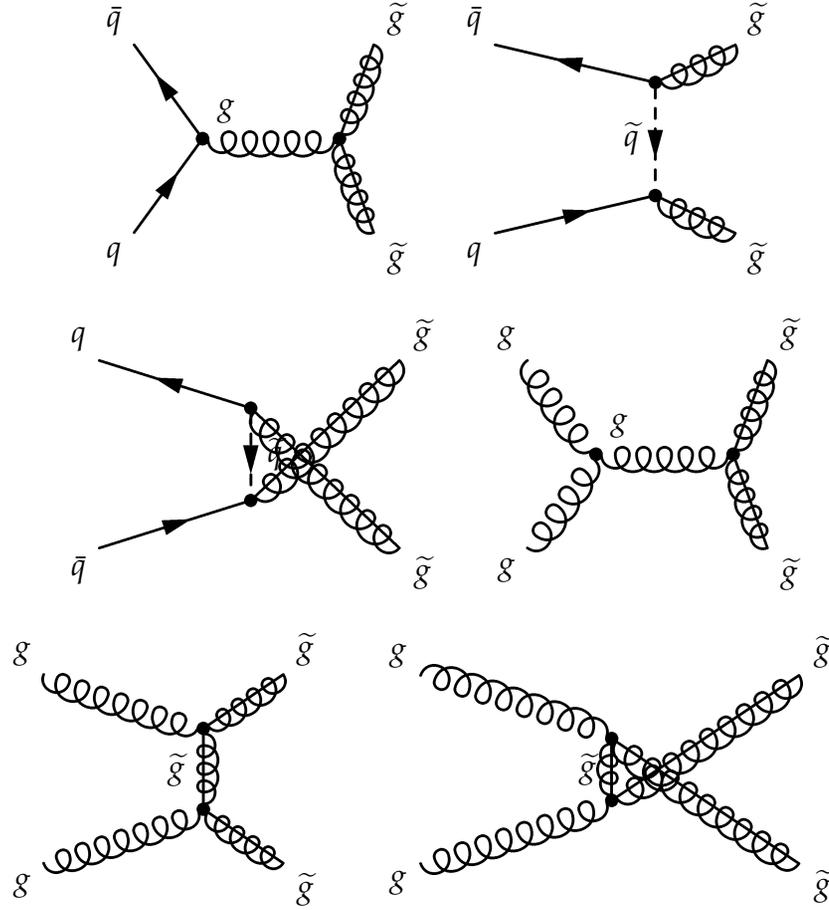
1.3.1. Producción \tilde{q} y \tilde{g} . En una colisión protón-protón a energías del LHC las s-partículas van a ser producidas como un resultado de interacciones parton-parton. Algunos ejemplos de la producción de s-quarks y gluinos son mostrados en las figuras 1.2, 1.3 y 1.4.

La producción gluino-gluino va a ser observada cuando se produzca una aniquilación quark-antiquark o una interacción gluon-gluon. En el caso de la producción s-quark-s-quark, las interacciones podrían ser por la interacción quark-quark, quark-antiquark o gluon-gluon. Para producir s-quarks-gluinos se necesita la interacción entre quarks y gluones.⁸

⁷PORTELL, Xavier. Search for Gluino and Squark production in Multi-jets plus Missing Transverse Energy Final States at the Tevatron using the CDF detector. Universidad Autónoma de Barcelona, Instituto de Física de Altas Energías, 2007.

⁸OCAMPO, Alberto Andres. Search for Squarks and Gluinos events with muons, Jets and Missing Transverse Energy in pp collisions using the CMS detector. Ph.D Thesis Proposal, Universidad de los Andes, Departamento de Física, 2008.

Figura 1.2: Diagramas de Feynman para la producción $\tilde{g}\tilde{g}$ en colisiones hadronicas



1.3.2. Decaimientos \tilde{q} y \tilde{g} . La mayoría de canales de decaimiento de s-quarks y gluinos envuelven charginos y neutralinos. Estas partículas también decaerán hasta que el LSP es alcanzado. Los modos de decaimiento preferidos para los s-quarks son:

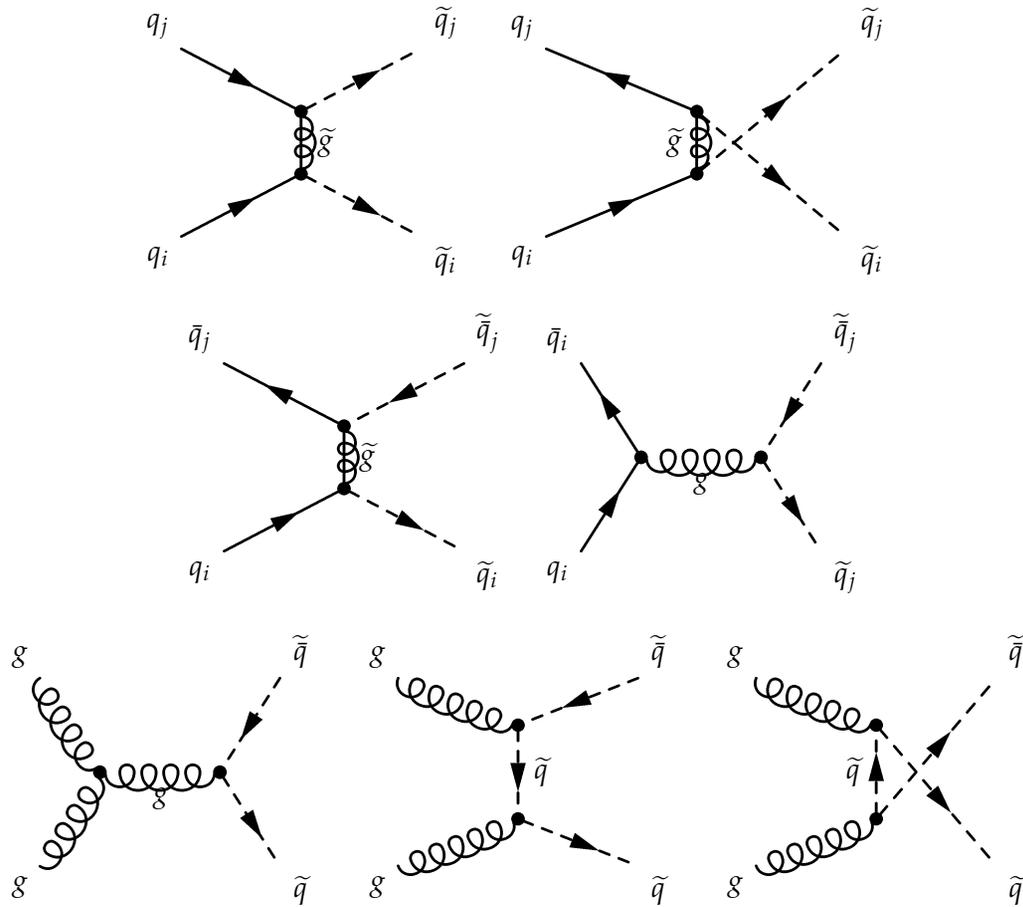
$$\tilde{q} \rightarrow q\tilde{g} \quad \tilde{q} \rightarrow q\tilde{\chi}_i^0 \quad \tilde{q} \rightarrow q'\tilde{\chi}_i^\pm \quad (1.2)$$

En el caso del gluino, su decaimiento se hace a través de s-quarks:

$$\tilde{g} \rightarrow \tilde{q}\tilde{q} \quad \tilde{g} \rightarrow qq'\tilde{\chi}_i^0 \quad \tilde{g} \rightarrow qq'\tilde{\chi}_i^\pm \quad (1.3)$$

Los neutralinos decaerán en neutralinos mas livianos hasta llegar a $\tilde{\chi}_1^0$, como se muestra en la figura 1.5. El decaimiento del neutralino puede ser mediado por la producción de bosones Z. Los bosones Z pueden decaer en un par de leptones de signo opuesto, como por ejemplo muones.

Figura 1.3: Diagramas de Feynman para la producción $\tilde{q}\tilde{q}$ en colisiones hadronicas



1.4. BÚSQUEDA DE SEÑALES DE S-QUARKS Y GLUINOS

Debido a la conservación de la paridad- R , las s -partículas deben ser producidas en pares. Cada s -partícula debe decaer en otra s -partícula, hasta que finalmente se llegue a la LSP, la cual debe ser estable dado que no hay s -partículas más livianas. En el estado final de todo evento en el cual se produjo materia supersimétrica deben haber dos LSPs. Las LSPs son masivas, neutras y solo interactúan débilmente, es decir, se comportan como neutrinos muy masivos. En la gran mayoría de los modelos SUSY, las LSPs son neutralinos $\tilde{\chi}_1^0$, que no dejarán huella en los detectores, pero sí un gran faltante de energía transversal (E_T^{miss}). Esta es la principal señal experimental de producción de partículas supersimétricas.

Buenos canales experimentales para ver la producción de s -quarks y gluinos son aquellos en los cuales estas partículas decaen por la producción en cadena de neutralinos y Z s. Esta clase de decaimientos resultará en señales con jets, energía transversal faltante y muones. Por lo tanto el canal que será usado en este trabajo es el que se muestra en la figura 1.6.

Figura 1.4: Diagramas de Feynman para la producción $\tilde{q}\tilde{g}$ en colisiones hadronicas

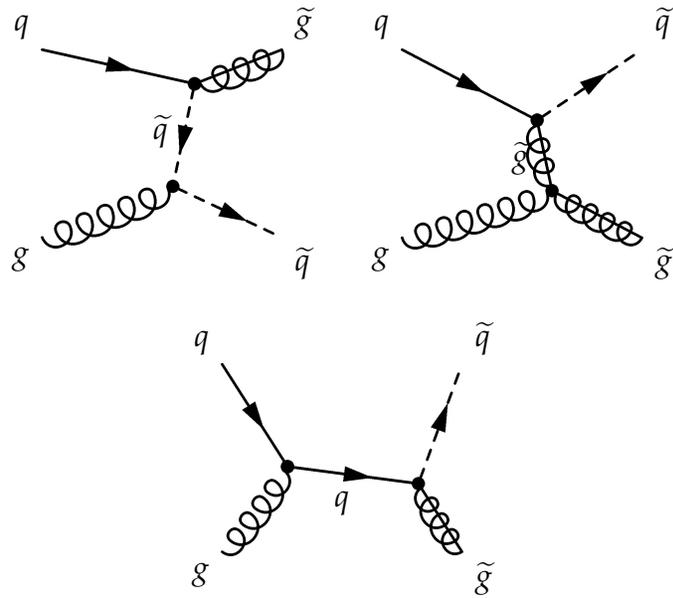
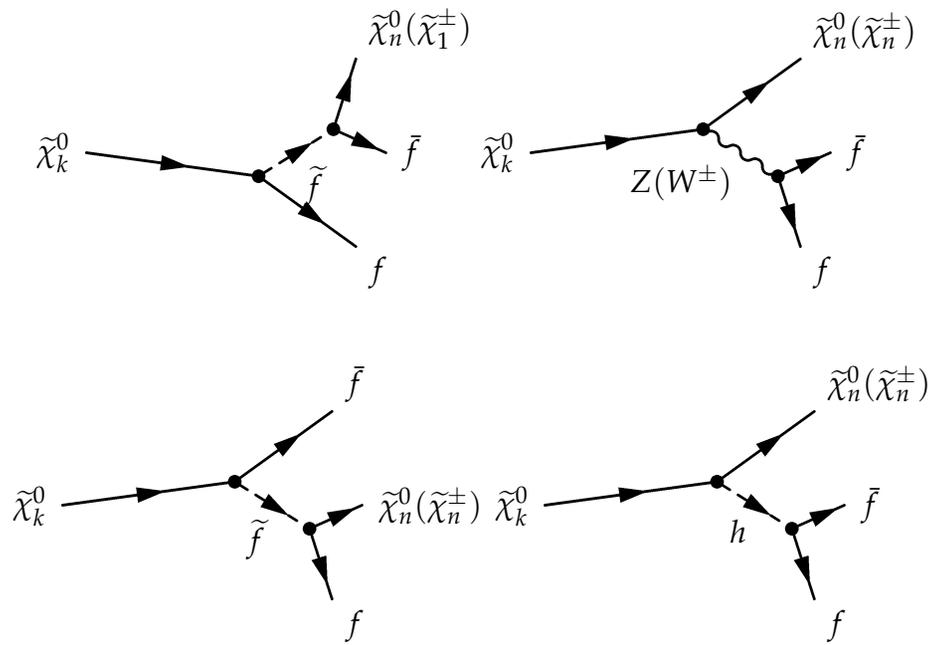


Figura 1.5: Diagramas de Feynman para decaimiento de neutralinos



2 SIMULACIÓN DE LA SEÑAL EXPERIMENTAL DE PRODUCCIÓN DE S-QUARKS Y GLUINOS

Como se menciono anteriormente, uno de los objetivos de este trabajo es simular la producción y decaimientos de los s-quarks y gluinos, específicamente el canal que se describió en el capítulo anterior, para esto se utiliza el software PYTHIA. En este capítulo se describirá las características generales que se necesitan para realizar la simulación.

2.1. PYTHIA

El simulador PYTHIA es un generador de eventos basado en el método de Monte Carlo. Esta compuesto por un conjunto de librerías escritas para ser corridas mediante el lenguaje Fortran 77. Dadas las partículas iniciales y la energía con la que se quieren colisionar, este programa esta en capacidad de generar la mayoría de subprocesos que podrían producirse a partir de esta colisión.⁹

PYTHIA es un programa que es usado para generar eventos de física de altas energías. El programa esta diseñado para generar eventos completos, como los observados experimentalmente.

2.2. ROOT

Este es un paquete de software para analizar datos desarrollado en el CERN (e inspirado en las actividades de física de altas energías). Este programa es rico en un conjunto de herramientas para la adquisición de datos, procesamiento y análisis, implementado como librerías C++¹⁰.

Ya que ROOT tiene una interfaz en PYTHIA, es posible reproducir los histogramas en ROOT con los eventos generados por PYTHIA, esto no requerirá programar en Fortran sino mas bien en C++. Esta interfaz se llama *TPythia6*, la cual contiene todas las características y subprocesos que tiene PYTHIA.

2.3. SIMULACIÓN IMPLEMENTADA

Como se menciono anteriormente se deben simular colisiones protón-protón a energía centro de masa de 14 TeV; y además de esto que PYTHIA solamente simule el canal que se menciono en el capítulo anterior.

2.3.1. Procesos a Simular. Lo primero que se debe seleccionar es el proceso físico que producirá un par s-quark-gluino por medio de una interacción parton-parton. PYTHIA contiene una gran variedad de procesos SUSY, que son activados individualmente con la

⁹SJOSTRAND, LONNBLAND, MRENNNA, SKANDS. Pythia 6.4 Physics and Manual. hep-ph, 0603175, 2006.

¹⁰ROOT-An Object Oriented Data Analysis Framework, Users Guide 5.20, 2008.

opción MSUB(ISUB)=1, donde el numero ISUB del proceso a simular se lo puede ver en las tablas del manual ¹¹. En este caso los procesos son los mostrados en la cuadro 2.1.

Cuadro 2.1: Subprocesos que permiten la simulación de la producción de s-quarks y gluinos

Subproceso	Numero (ISUB)
$f_i g \rightarrow \tilde{q}_{iL} \tilde{g}$	258
$f_i g \rightarrow \tilde{q}_{iR} \tilde{g}$	259

Por lo tanto al realizar una interacción protón-protón que en realidad es una interacción parton-parton, se producirán solamente s-quarks y gluinos.

Además de esto se seleccionan los subprocesos que se quieren estudiar. Debido a que cada evento generado produce cientos de partículas, no todas correspondientes al estado final que se espera, y como se quiere simular un canal en particular, se debe seleccionar algunos tipos de decaimientos para estudiar solamente las partículas que nos interesan.

Primero que todo se debe desactivar todos los modos de decaimiento, de esta forma solamente se habilitara los decaimientos deseados. MDME(IDC,1)=i es la orden que activa o desactiva un canal de decaimiento individual para una partícula; donde IDC es el numero del canal de decaimiento, hay en total 4352 canales de decaimiento de todas las partículas incluidas en PYTHIA, este numero se lo puede obtener con la orden PYLIST(12) la cual imprime una lista completa de todas las partículas y sus posibles decaimientos. La variable i puede tomar el valor entre -1 y 5 (ver manual), entre algunos de estos valores los de interés son 0 y 1, los cuales desactivan y activan el canal respectivamente. Los canales de decaimiento que nos interesan para simular en este trabajo son mostrados en el cuadro 2.2.

Cuadro 2.2: Decaimientos activados para simular el canal mostrado en la figura 1.6

Decaimiento	IDC
$\tilde{g} \rightarrow \tilde{q} + q$	1976-1971
$\tilde{q}_L \rightarrow \tilde{\chi}_2^0 + q$	1596, 1641, 1665, 1710
$\tilde{q}_R \rightarrow \tilde{\chi}_2^0 + q$	3661, 3710, 3738, 3787
$\tilde{\chi}_2^0 \rightarrow \tilde{\chi}_1^0 + Z^0$	2337, 2409, 2408
$Z^0 \rightarrow \mu^+ \mu^-$	184

Una vez activados solamente estos decaimientos, PYTHIA simulara el canal mostrado en la figura 1.6.

En éste trabajo no se simulara background, ya que su estudio requeriría un análisis mas extenso y se sale del alcance de este trabajo.

¹¹SJOSTRAND, LONNBLAND, MRENNNA, SKANDS. Pythia 6.4 Physics and Manual. hep-ph, 0603175, 2006, p430.

Por ultimo en la producción de partículas supersimétricas hay que especificar los parámetros SUSY, que se explicaron en el capítulo anterior, m_0 , $m_{1/2}$, A_0 , $\tan(\beta)$, $\text{signo}(\mu)$; los valores de estos 5 parámetros fueron tomados de estudios en artículos consultados¹², en estos estudios se realizan análisis para sectores de masas bajas y altas de s-fermiones y gauginos, para este trabajo se utilizan los valores para LM2 (Low Mass), estos son mostrados en el cuadro 2.3.

Cuadro 2.3: Parámetros SUSY escogidos para el análisis.

m_0	$m_{1/2}$	A_0	$\tan(\beta)$	$\text{signo}(\mu)$
185	350	0	35	+

Luego que ya se han especificado los parámetros mencionados anteriormente, se inicializa el programa para la generación de eventos. Esta orden es:

CALL PYINIT(FRAME,BEAM,TARGET,WIN)

Donde FRAME es una variable usada para especificar el marco del experimento que en esta caso es CMS, lo que quiere decir que se colisionara un haz con momentum en la dirección $+z$ con otra haz con momentum en la dirección $-z$. BEAM y TARGET son las variables que especifican que tipo de partículas colisionaran, que para este caso son protones "p". Y por ultimo WIN la cual es relacionada a la energía del sistema, y depende del marco del experimento, para el caso del CMS es de 14 TeV (14000 GeV).

Ya con esto queda completamente simulado el canal a estudiar, por ultimo hay que generar los histogramas, que se estudiaran en el siguiente capítulo.

¹²CAVANUGH, R. Inclusive SUSY Searches at CMS with Emphasis on Detector Systematics, University of Florida, 2006.

3 RESULTADOS

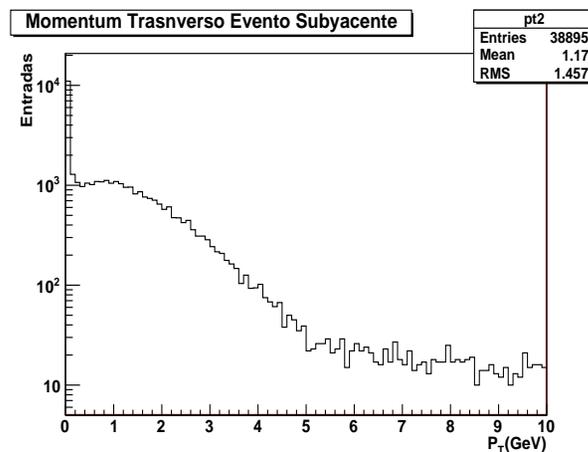
Después de haber especificado todos los parámetros necesarios para la simulación, se obtiene los histogramas que ayudaran en el análisis. Los códigos resultantes se muestran en el apéndice A.

Como se menciona en el capítulo 1, el canal a estudiar es aquel en el que un par s-quark-gluino decaen a estados finales de leptones que reconstruyen bosones Z^0 , LSP y jets, además también se estudio las señales que este tipo de partículas dejan a su paso por el detector y la estrategia para su búsqueda. En este capítulo se describe el análisis para estudiar este tipo de canal, además se presenta los resultados para los cuales la evidencia de supersimetrica podría ser obtenida en el CMS.

3.1. LAS COLISIONES EN EL LHC

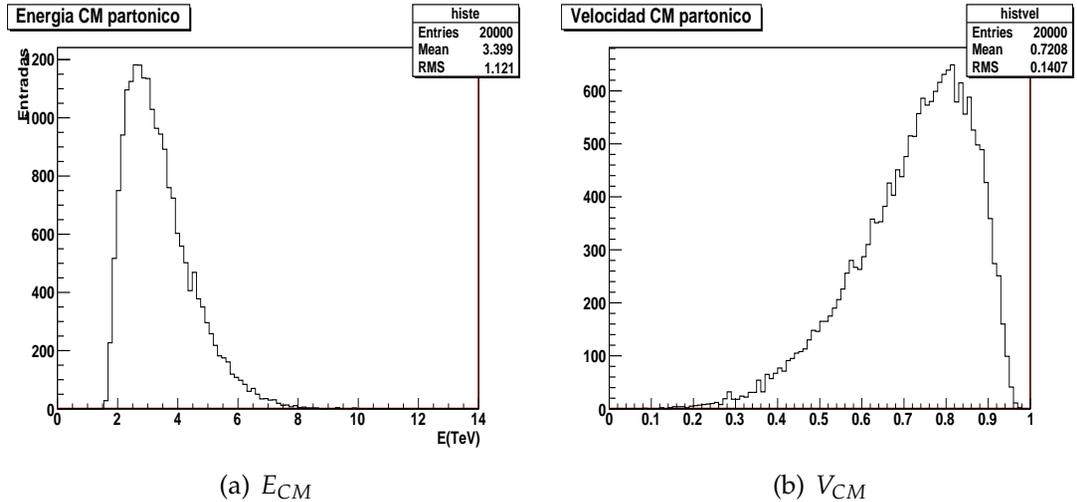
Un protón consiste de tres quarks de valencia (uud), mas un "mar" de quarks y gluones. Cuando dos protones son llevados en una colisión a energía del LHC, es desconocido cual de los constituyentes en realidad colisionan y cuales fueron puramente espectadores del evento. Los constituyentes individuales llevan solamente una fracción desconocida de la energía del protón total. Esto significa que la energía real centro de masa en una colisión quark-quark, quark-gluon o gluon-gluon, es mucho menor que la energía centro de masa total del protón. El resto de la energía queda en los remanentes de la colisión, esto es lo que se llama el *evento subyacente*, estas partículas se fragmentaran y producirán piones y gamas, que luego deberán tenerse en cuenta a la hora de calcular la energía faltante. En la figura 3.1 se muestra el momentum transverso del evento subyacente generado en la simulación, en el cual se puede observar que su $p_T \approx 1$ GeV, lo que constituye eventos de bajo p_T que no llegarían a contaminar significativamente la señal supersimetrica.

Figura 3.1: Momentum transverso evento subyacente



En la producción de s-quarks y gluinos (ver capítulo 1), se realiza por medio de una interacción quark-gluon. En este estudio se encuentra que al colisionar un par de protones a $E_{CM} = 14\text{TeV}$, la colisión parton-parton se hace a $E_{CM} \approx 3\text{TeV}$ como se muestra en la figura 3.2; ciertamente este valor es mucho menor de los 14 TeV de la colisión inicial, así que 3 TeV es la energía con la que se producirán nuevas partículas, es decir, al aniquilarse un par de partones esta energía se convertirá en nuevas partículas masivas. Además la V_{CM} en sistema partonico es $V_{CM} \approx 0,7$.

Figura 3.2: Marco de referencia centro de masa parton-parton.



3.2. ANÁLISIS DEL CANAL

Como se dijo anteriormente el análisis se realiza por medio de un canal vía bosones Zs que decaen en un par de muones, jets y neutralinos. Así que estos estados finales son objeto de este estudio.

Para el análisis de este canal se lo dividirá en dos grupos, con el fin de estudiar las partículas finales por separado, dependiendo si son producidas por el decaimiento en cadena de un s-quark o un gluino. Este método es usado en muchos análisis, es un algoritmo que permite la separación de cadenas de decaimiento de las s-partículas también llamado *Algoritmo Hemisferio*¹³. En la figura 3.3 se muestra la separación en grupos del canal.

Para producir un s-quark-gluino se hace por medio de una colisión parton-parton, que como se vio en el capítulo 1 es una colisión quark-gluon. Al simular este canal en PYTHIA se encuentra que al producir esta interacción, los s-quarks y gluinos resultantes tienen una separación angular aproximadamente entre los 30 - 160 grados; por lo que la probabilidad de producir s-quark-gluino con un ángulo de separación menor de 30 grados es muy poca, como se muestra en la figura 3.4.

¹³CMS Physics Technical Design Report, Volume II: Physics Performance, CERN/LHCC 2006-021, p410.

Figura 3.3: Canal dividido por cadenas de decaimiento.

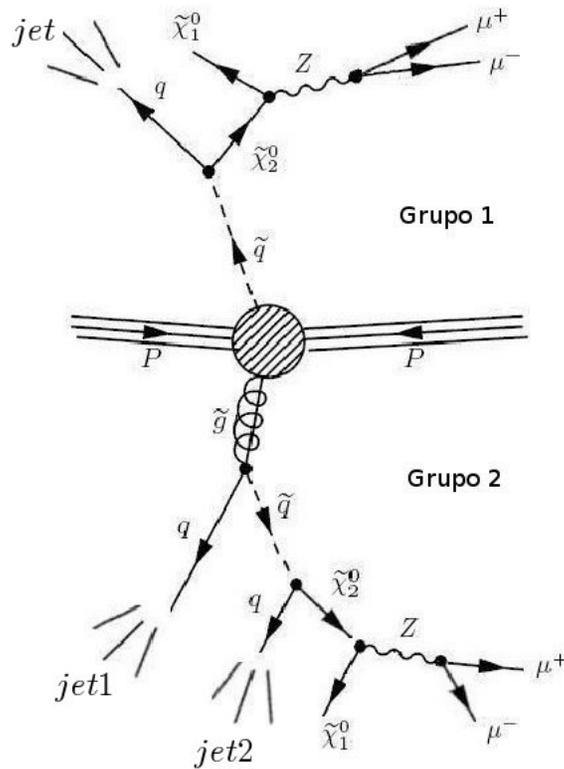
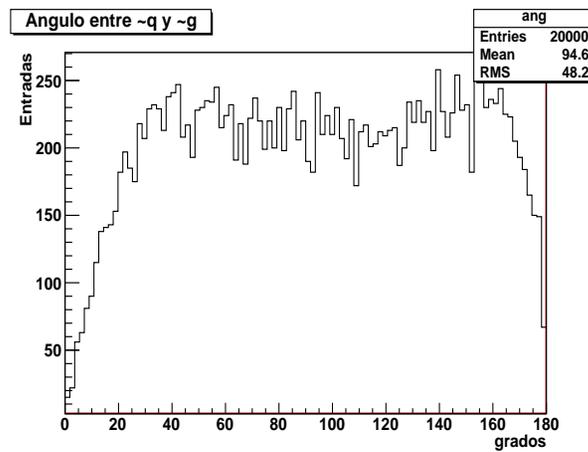


Figura 3.4: Ángulo de separación entre los s-quarks y gluinos.

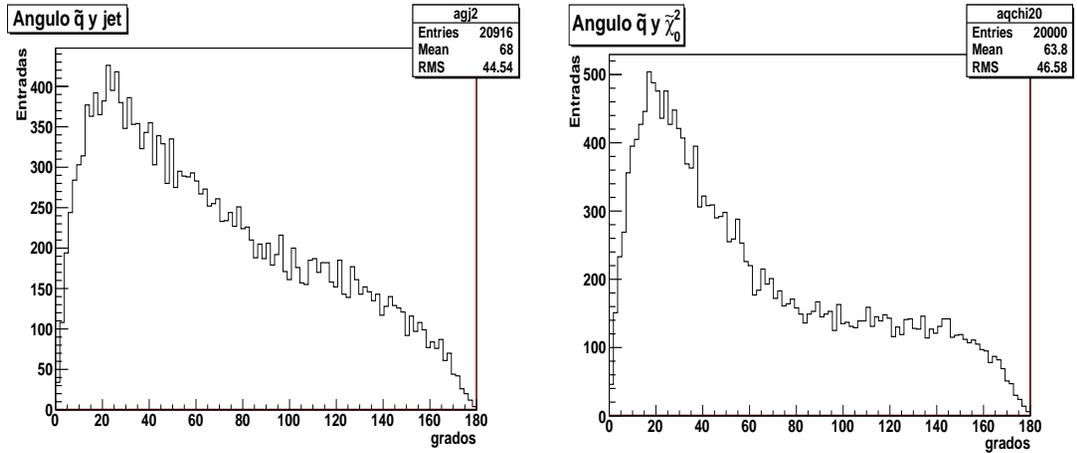


Se puede observar en la figura 3.5; la separación angular para las partículas que pertenecen al grupo 1. Los histogramas, representan el ángulo que separa al s-quark del jet, al s-quark del boson Z^0 , el ángulo de separación entre el s-quark y el segundo neutralino mas liviano $\tilde{\chi}_2^0$ y entre el s-quark y $\tilde{\chi}_1^0$ (LSP). Estos histogramas son bastante similares,

en donde la mayoría de eventos muestran ángulos muy pequeños alrededor de 20-40 grados.

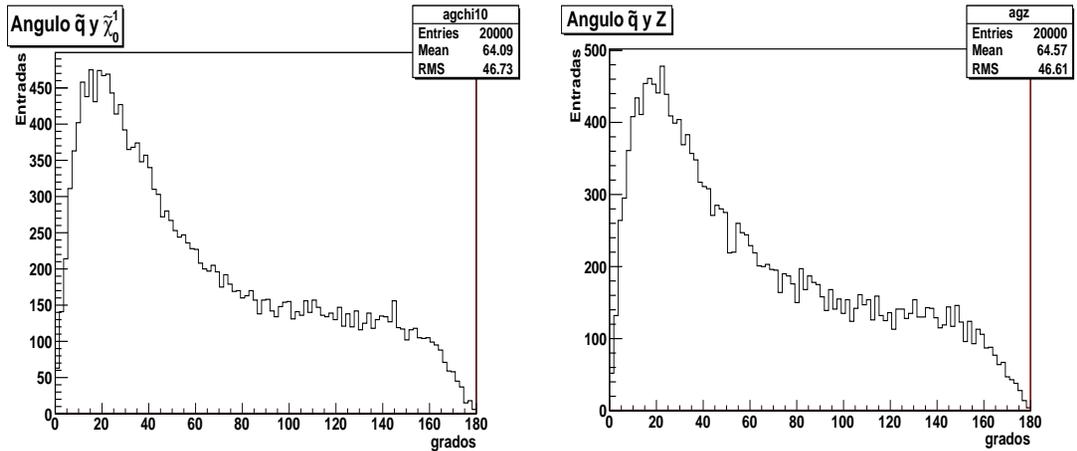
Mientras que en la figuras 3.6; están las separaciones angulares para las partículas de grupo 2. En esta figura, se encuentran los histogramas para el ángulo de separación entre el gluino y el primer jet, entre el gluino y el segundo jet, entre el gluino y el boson Z^0 y entre el gluino y el $\tilde{\chi}_1^0$.

Figura 3.5: Separación angular entre partículas del grupo 1



(a) Separación angular entre s-quark y jet

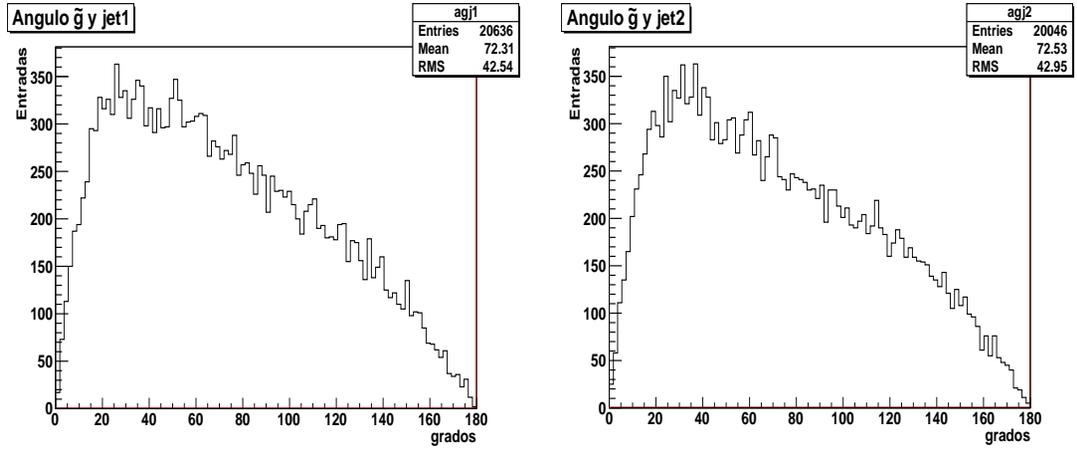
(b) Separación angular entre s-quark y $\tilde{\chi}_2^0$



(c) Separación angular entre s-quark y $\tilde{\chi}_1^0$

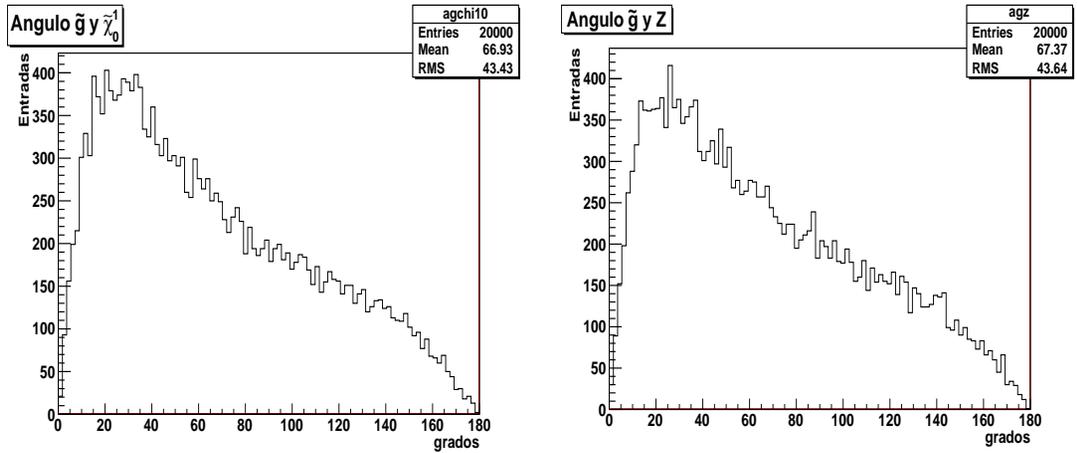
(d) Separación angular entre s-quark y Z^0

Figura 3.6: Separación angular entre partículas del grupo 2



(a) Separación angular entre gluino y jet 1

(b) Separación angular entre gluino y jet 2



(c) Separación angular entre gluino y $\tilde{\chi}_1^0$

(d) Separación angular entre gluino y Z^0

3.3. SEÑAL DEL Z^0

La detección de procesos susy dejando estados finales con Z^0 , puede ser detectado en el CMS usando los decaimientos del Z^0 en un par de leptones de signo opuesto. Los principales backgrounds del ME vienen de la producción de uno o mas bosones Z^0 y jets, de aquí la importancia de estudiar este tipo de señales.

Para determinar si los muones se produjeron a partir del boson Z^0 , se reconstruye la masa invariante del par de muones $M_{\mu\mu}$, como se muestra en la siguiente ecuación:

$$M_{\mu\mu} = \sqrt{(E_{\mu^+} + E_{\mu^-})^2 - (p_{\mu^+} + p_{\mu^-})^2} \quad (3.1)$$

Ya que todos los muones en este caso decayeron de un Z^0 , el valor calculado a partir de los datos experimentales para $M_{\mu\mu}$ debe estar alrededor de 91,18 GeV, ya que este es el valor de la masa del Z^0 .

En la figura 3.7 se muestra la masa invariante obtenida a partir del par de muones en estado final, calculada para los dos grupos.

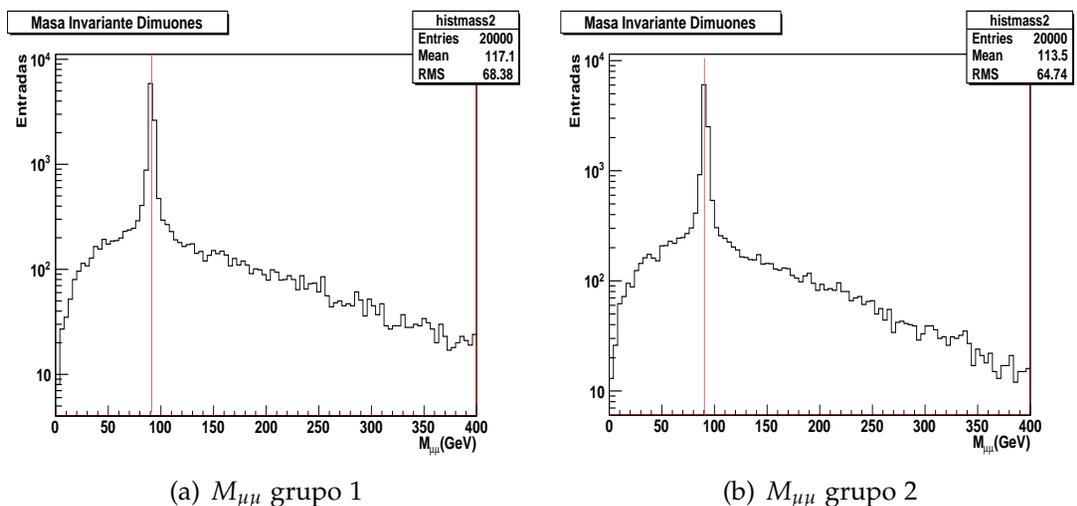
Como se dijo anteriormente se puede observar una representación alrededor de un pico, cuyo valor pertenece a la masa del boson Z^0 , la línea roja representa este valor experimental.

Estos histogramas presentan el mismo comportamiento con los que se muestran en el análisis hecho en *CMS Physics Technichal Design Report*¹⁴, en este estudio se gráfica la masa invariante para la señal y para el background, como se muestra en la figura 3.8. Para este tipo de análisis se tiene en cuenta el background lo que ayuda a determinar un corte mas efectivo.

Las líneas verticales mostradas en la figura 3.8, representan los requerimientos en la masa invariante para un par de leptones de signo opuesto, que para este análisis es $81\text{GeV} < M_{\ell\ell} < 96,5\text{GeV}$. Este intervalo es requerido para reducir todo el background que podría simular la señal.

El ángulo que separa los dos muones que reconstruyen el Z^0 se muestra en la figura 3.9. Aquí se gráfica la separación angular tanto para el grupo 1 a la izquierda y para el grupo 2 a la derecha, los dos histogramas son bastante similares, los cuales muestran una mayor producción entre los 30 - 60 grados aproximadamente, que corresponden a un rango de 0.5 - 1 radianes, bastante similar al mostrado en la figura 3.10. En esta ultima se muestra el ángulo $\Delta\phi$ entre los dos leptones que reconstruyen la masa del Z^0 , obtenido en el artículo *CMS Physics Technichal Design Report*¹⁵, aquí se gráfica para la señal y el background; por lo tanto el requerimiento para el ángulo de separación de los muones es de 2.65 rad.

Figura 3.7: Masa invariante de los dos pares de muones.



¹⁴CMS Physics Technichal Design Report, Volume II: Physics Performance, CERN/LHCC 2006-021, p434.

¹⁵CMS Physics Technichal Design Report, Volume II: Physics Performance, CERN/LHCC 2006-021, p434.

Figura 3.8: Masa invariante de dimuones para el background y para la señal (rojo). Las líneas verticales denotan los requerimientos de masa impuestos¹⁴

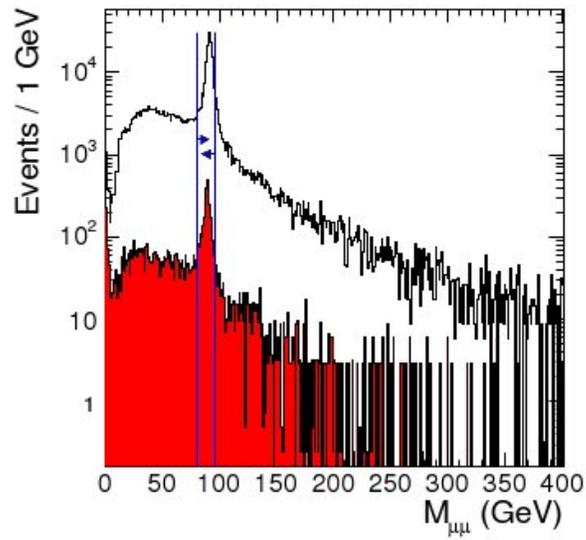


Figura 3.9: Ángulo que separa al par de muones.

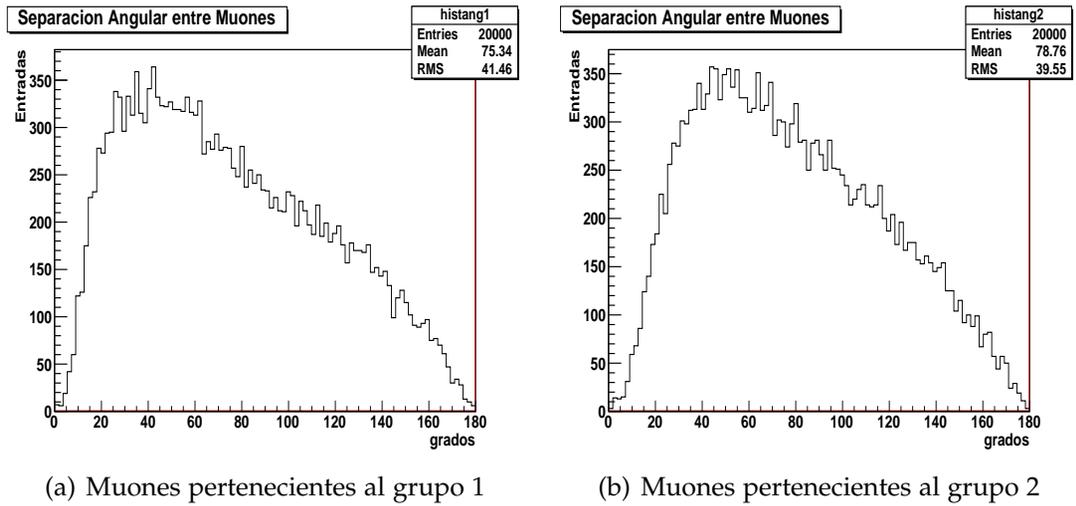
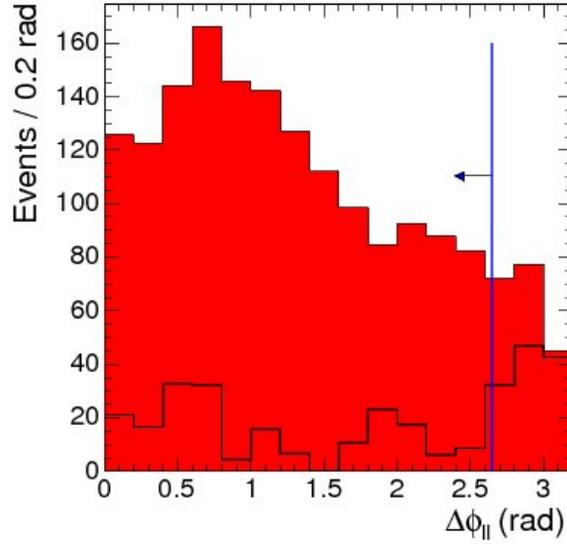


Figura 3.10: $\Delta\phi$ entre los dos leptones para el background (línea negra) y para la señal (rojo). La línea vertical indica los requerimientos para $\Delta\phi$.¹⁴



3.4. SEÑAL DEL $\tilde{\chi}_1^0$

Como se ha dicho en los capítulos anteriores una de las señales experimentales más importantes que revelarían la existencia de materia supersimétrica es la energía transversal faltante, E_T^{miss} , que dejarían los neutralinos más livianos, LSP, a su paso por el detector.

La mayoría de los detectores de colisiones hadrónicas han sido diseñados para cubrir tanto ángulo sólido como sea posible con el calorímetro. La principal motivación fue proveer un método para identificar la presencia de uno o más neutrinos energéticos u otras partículas estables que interactúen débilmente con la materia en el estado final a través de aparente energía faltante. Esta medida de la energía transversal faltante, es una poderosa herramienta para ayudar en la detección de señales de física envueltas en una o más partículas que interactúen débilmente con la materia.

La energía transversal faltante es reconstruida de la suma vectorial de las energías transversas de todas las celdas en el calorímetro, el vector opuesto a este vector total de momentum visible es denotado por el vector de energía faltante; así que la E_T^{miss} es la magnitud negativa de la suma vectorial de las energías transversales en las torres calorimétricas y definida como¹⁶:

$$\vec{E}_T^{miss} = - \sum_n \{ E_n \sin \theta_n \cos \phi_n \hat{i} + E_n \sin \theta_n \sin \phi_n \hat{j} \} \quad (3.2)$$

Para encontrar este valor se debe tener en cuenta todas las partículas en estado final. E_T^{miss} para este canal es mostrado en la figura 3.11. El valor promedio de la energía transversal faltante para este estudio es aproximadamente $E_T^{miss} \approx 241 \text{ GeV}$.

¹⁶HEYNINCK, Jan. The CMS Top Quark Physics, p94, 2008.

Para el análisis hecho en artículo *CMS Physics Technichal Design Report*¹⁷, la gráfica de E_T^{miss} es mostrada en la fig 3.12 (a). En este caso el análisis se hace para la señal y el background, encontrando que la E_T^{miss} debe ser mas grande que 230 GeV para reducir todo el background. Las masas reconstruidas para el par $\mu^-\mu^+$ sin el corte en la masa del Z^0 pero después de el corte sobre E_T^{miss} es mostrado en la fig 3.12 (b); un claro pico de Z^0 de la señal es observado.

Figura 3.11: Energía Transversal Faltante

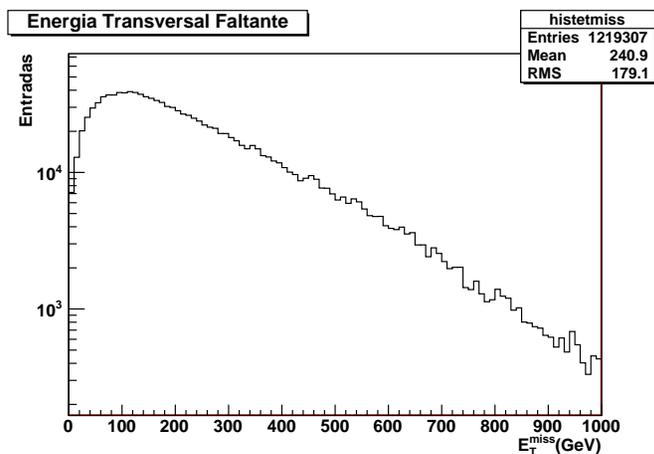
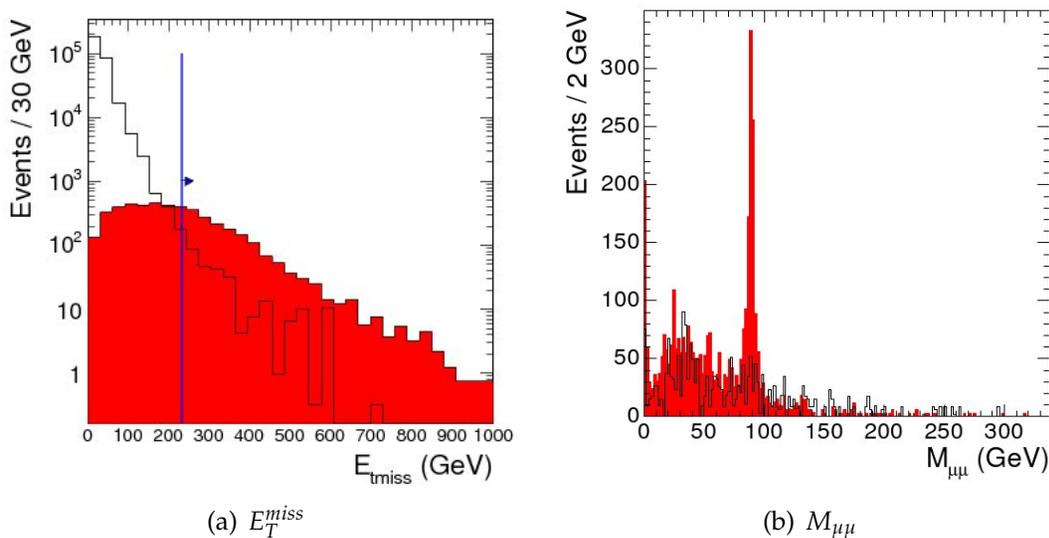


Figura 3.12: (a) Energía Transversal Faltante , la línea vertical indica los requerimientos de E_T^{miss} y (b) $M_{\mu\mu}$ después del corte sobre E_T^{miss} ; para el background y la señal (rojo)¹⁵



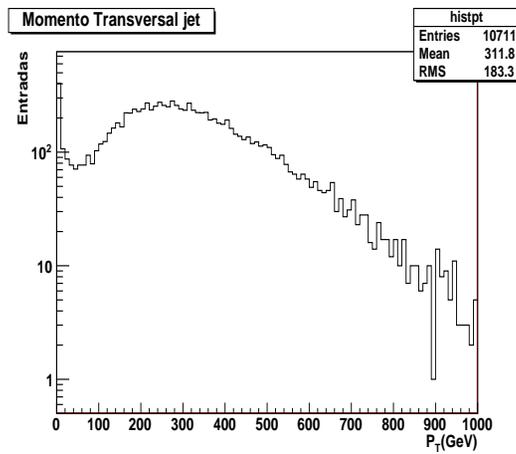
¹⁷CMS Physics Technichal Design Report, Volume II: Physics Performance, CERN/LHCC 2006-021, p434.

3.5. SEÑAL DE LOS JETS

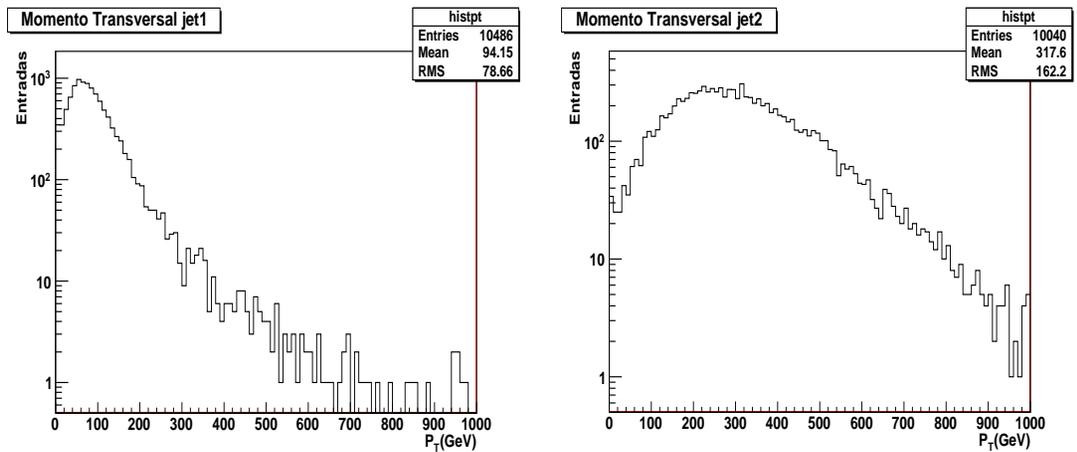
Una estrategia general para la búsqueda de señales susy sería la selección de eventos con gran energía transversal faltante y partículas reconstruidas con gran momentum transverso. En el LHC estos objetos son predominantemente jets.

En la figura 3.13, se muestra la distribución para el momento transverso de los jets, de estas graficas se puede observar que los jets con mayor momento pertenecen a aquellos que vienen del decaimiento de un s-quark, $\tilde{q} \rightarrow q\chi_2^0$, esto para los dos grupos. Para el grupo 1 el jet tiene un momentum $p_T \approx 312\text{GeV}$, mientras tanto para el grupo 2 como se dijo anteriormente el jet con mayor momentum es el jet 2, que proviene del decaimiento de un s-quark y tiene un momentum de $p_T \approx 318\text{GeV}$.

Figura 3.13: Momentum transverso para los jets del canal.



(a) Jet del grupo 1



(b) Jet1 grupo 2

(c) Jet2 grupo 2

4 CONCLUSIONES

- El evento subyacente, como se explico anteriormente son los remanentes de los protones fracturados que se fragmentaran y podrían llegar a formar jets, sin embargo, estos no llegarían a contaminar significativamente la señal detectable, debido a que constituyen jets de bajo p_T ($p_T \approx 1\text{GeV}$).
- Entre los cortes que servirán como filtro en señales que sean contaminadas con eventos de background sin interés, se debe escoger un intervalo de valores aceptable para la masa invariante del par de muones provenientes del Z^0 , este rango de valores estará alrededor de la masa del boson Z^0 como se muestra en la figura 3.8, pero para este analisis no hay criterios para escoger con exactitud este intervalo debido a que este simulación no presenta background..
- Para el ángulo entre los dos muones que reconstruyen el Z^0 se puede ver que hay mayor producción entre 40- 60 grados aproximadamente, debido a que en este estudio no se simula background no se puede determinar con exactitud los valores angulares para los cuales la señal estará libre de background..
- De igual manera la energia transversal faltante estará en el rango de aproximadamente los 240 GeV, pero en este caso tampoco se simula el background así que no se puede determinar un corte exacto para separar la señal supersimetrica..
- Los objetos con mayor momentum producidos en este canal de estudio son los jets, específicamente aquellos que provienen del decaimiento de un s-quark $\tilde{q} \rightarrow q\tilde{\chi}_2^0$, que como se puede observar en las graficas para el grupo 1 el jet que viene de esta clase de decaimiento tiene un momentum aproximado de $p_T \approx 312\text{GeV}$. De igual forma para el grupo 2 el jet con mayor momentum proviene del decaimiento de un s-quark y es el jet que en este caso se ha llamado jet2, con un valor aproximado de $p_T \approx 317\text{GeV}$. Estos jets que son el producto de la misma clase de decaimiento, pero que vienen de una cadena de decaimiento diferente, tiene aproximadamente igual momentum.

5 RECOMENDACIONES

Seria interesante comparar resultados para diferentes regiones de masa, parámetros de m_{Sugra} (ver Capitulo 2), las regiones de masas bajas las cuales serán las primeras en estudiarse en el LHC y las regiones para puntos de masas grandes que serian las ultimas en alcanzar en el LHC; en este caso se trabajo con los parámetros para LM2, pero podría realizarse un estudio comparando resultados para diferentes parámetros.

Realizar la búsqueda de s-quarks y gluinos en el mismo canal de decaimiento, pero sin restringir la simulación a un canal en particular, sino mas bien permitir toda clase de decaimientos, así se obtendrán señales contaminadas con bastante background, con lo que se podría obtener cortes cinemáticos específicos.

BIBLIOGRAFIA

- CAVANUGH, R. Inclusive Susy Searches at CMS with emphasis on detector systematics, University of Florida, 2006.
- CMS Physics Technical Design Report, Volume II: Physics Performance, CERN/LHCC 2006-021.
- DANKERS, REINER. The Physics Performance of and Level 2 Trigger for the Inner Detector of ATLAS.
- GENIK, RICHARD. The Search for Di-Lepton Signatures from Squarks and Gluinos in $p\bar{p}$ Collisions at $\sqrt{s} = 1,8$ TeV. Michigan State University, Department of Physics and Astronomy, 1998.
- GOFORTH, MARK. Search for Squarks and Gluinos with D0 Detector. The Florida State University, Department of Physics, 1995.
- HALZEN, F., MARTIN, A.D., Quarks and Leptons, New York: John Wiley & Sons, 1984.
- HEYNINCK, JAN. The CMS Top Quark Physics, 2008.
- LYON, ADAM, A Search for Squarks and Gluinos using the Jets and Missing Energy Signature at D0, University of Maryland, 1997.
- MOHR, NICKAS. Neutralino Reconstruction in Dilepton Final States with the CMS Experiment. Rheinisch-Westfälischen Technischen Hochschule Aachen, 2008.
- MORII, T., LIM, C.S., MUKHERJEE, S.N., The Physics of the Standard Model and Beyond, 2004.
- OCAMPO, ALBERTO ANDRES. Search for Squarks and Gluinos events with muons, Jets and Missing Transverse Energy in pp collisions using the CMS detector. Ph.D Thesis Proposal, Universidad de los Andes, Departamento de Física, 2008.
- PORTELL, XAVIER. Search for Gluino and Squark production in Multi-jets plus Missing Transverse Energy Final States at the Tevatron using the CDF detector. Universidad Autónoma de Barcelona, Instituto de Física de Altas Energías, 2007.
- ROOT- An Object Oriented Data Analysis Framework, Users Guide 5.20, 2008.
- SJOSTRAND, LONNBLAND, MRENNNA, SKANDS. Pythia 6.4 Physics and Manual. hep-ph, 0603175, 2006.

ANEXOS

Anexo A. Códigos Utilizados para la Simulación

Anexo A.1 Marco de Referencia Centro de Masa Partonico El código que se utiliza para generar la simulación en una colisión proton-proton a energía de 14 TeV y producir un canal de decaimiento específico (ver Capítulo 1) es el siguiente

```
//=====
//  LIBRERIAS
//=====
#ifndef __CINT__
# include <stdlib.h>
# include <ostream.h>
# include <TR00T.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
}

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;

//=====
//  HISTOGRAMAS
//=====

TPythia6 Pythia;
TFile* file = new TFile("stree1.root","RECREATE","stree1.root", compress);
TH1F *energy = new TH1F("histe","Energia CM partonico", 100, 0, 14);
TH1F *vel = new TH1F("histvel","Velocidad CM partonico", 100, 0, 1);
```

```

//=====
//      SUBPROCESOS
//=====
Pythia.SetMSEL(0); //subprocesos deseados deben ser activados.
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);//seleccionando multiiples interacciones
Pythia.SetMSTP(111,0);//fragmentacion on
Pythia.SetMSTP(82, 3); //multiple interaction "vary impact param"
Pythia.SetPARP(82, 2.41); //cut-off pt for multiple interaction
Pythia.SetMRPY(1, 88158204); //random seed number
// AJustar Parametros SUSY en escenario SUGRA
Pythia.SetIMSS(1,1); //Simulacion general en MSSM
Pythia.SetIMSS(1,2); //aproximacion SUGRA
Pythia.SetIMSS(9,1);
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(4,100);//mu
Pythia.SetRMSS(16,0); //A0 coupling

//=====
//      CANALES DE DECAIMIENTO
//=====
//Desabilitar todos los canales de decaimiento
for (int i=1; i<4352; i++) {
    Pythia.SetMDME(i,1,0);
}

//Habilitar Canales Deseados
//Forzar ~g-> ~q + q
Pythia.SetMDME(1976,1,1); //~g-> ~d_L + dbar
Pythia.SetMDME(1977,1,1); //~g-> ~d_Lbar + dbar
Pythia.SetMDME(1978,1,1); //~g-> ~d_R + dbar
Pythia.SetMDME(1979,1,1); //~g-> ~d_Rbar + d
Pythia.SetMDME(1980,1,1); //~g-> ~u_L + ubar
Pythia.SetMDME(1981,1,1); //~g-> ~u_Lbar + u
Pythia.SetMDME(1982,1,1); //~g-> ~u_R + ubar
Pythia.SetMDME(1983,1,1); //~g-> ~u_Rbar + u
Pythia.SetMDME(1984,1,1); //~g-> ~s_L + sbar
Pythia.SetMDME(1985,1,1); //~g-> ~s_Lbar + s
Pythia.SetMDME(1986,1,1); //~g-> ~s_R + sbar
Pythia.SetMDME(1987,1,1); //~g-> ~s_Rbar + s
Pythia.SetMDME(1988,1,1); //~g-> ~c_L + cbar
Pythia.SetMDME(1989,1,1); //~g-> ~c_Lbar + c
Pythia.SetMDME(1990,1,1); //~g-> ~c_R + cbar
Pythia.SetMDME(1991,1,1); //~g-> ~c_Rbar + c

//Forzar ~q->~chi_20+q
Pythia.SetMDME(1596,1,1); //~d_L->~chi_20+d
Pythia.SetMDME(1641,1,1); //~u_L->~chi_20+u
Pythia.SetMDME(1665,1,1); //~s_L->~chi_20+s
Pythia.SetMDME(1710,1,1); //~c_L->~chi_20+c
Pythia.SetMDME(3661,1,1); //~d_R->~chi_20+d
Pythia.SetMDME(3710,1,1); //~u_R->~chi_20+u
Pythia.SetMDME(3738,1,1); //~s_R->~chi_20+s
Pythia.SetMDME(3787,1,1); //~c_R->~chi_20+c

//Forzar ~chi20-> ~chi10 + Z0
Pythia.SetMDME(2337,1,1); //~chi20-> ~chi_10 + Z0
Pythia.SetMDME(2409,1,0); //~chi20-> ~chi_10 + Z0
Pythia.SetMDME(2408,1,0); //~chi20-> ~chi_10 + Z0

//Forzar Z0 -> mu+ + mu-
Pythia.SetMDME(184,1,1); // Z0-> mu- + mu+

//=====

```

```

// INICIALIZAR COLISION
//=====
Pythia.Initialize("CMS", "p", "p", 14000.0);
TStopwatch ioTime;
ioTime.Stop();

//=====
// GENERANDO EVENTOS
//=====
//Definiendo variables
Double_t p, E, M, e, E1, E2, m1, m2, px1, px2, py1, pt1, pt2, py2, pz1, pz2, p1, p2, cos12, masa1,
masa2, ang, angt, etaq, etag, thetaq, thetag, phiq, phig;
int aqg1 = 0;int aqg12=0;
Double_t pi = 4*atan(1.);

//Loop de eventos
for ( i=0; i<NEvents; i++ )
{
    if ( i%1 == 0 ) cout << "Event No.: " << i << endl;
    Pythia.GenerateEvent();

    //Loop sobre todas las particulas producidas
    for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
    {
        //Solamente s-quarks y gluinos iniciales
        if (Pythia.GetK(Pythia.GetK(npart,3),2)!=0) continue;

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

        if (Pythia.GetK(npart,2)==1000001 || Pythia.GetK(npart,2)==-1000001 || Pythia.GetK(npart,2)==
            1000002 || Pythia.GetK(npart,2)==-1000002 || Pythia.GetK(npart,2)==1000003 ||
            Pythia.GetK(npart,2)==-1000003 || Pythia.GetK(npart,2)==1000004 || Pythia.GetK(npart,2)==
            -1000004 || Pythia.GetK(npart,2)==2000001 || Pythia.GetK(npart,2)==-2000001 ||
            Pythia.GetK(npart,2)==2000002 || Pythia.GetK(npart,2)==-2000002 || Pythia.GetK(npart,2)==
            2000003 || Pythia.GetK(npart,2)==-2000003 || Pythia.GetK(npart,2)==2000004 ||
            Pythia.GetK(npart,2)==-2000004)
        {
            px1 = Pythia.GetP(npart,1);
            py1 = Pythia.GetP(npart,2);
            pz1 = Pythia.GetP(npart,3);
            E1 = Pythia.GetP(npart,4);
            p1 = sqrt(px1*px1+py1*py1+pz1*pz1);
        }

        if (Pythia.GetK(npart,2)==1000021)
        {
            px2 = Pythia.GetP(npart,1);
            py2 = Pythia.GetP(npart,2);
            pz2 = Pythia.GetP(npart,3);
            E2 = Pythia.GetP(npart,4);
            p2 = sqrt(px2*px2+py2*py2+pz2*pz2);
        }

        if(aqg1==1 || aqg12==1)
        {
            Double_t Ecm=sqrt(((E1+E2)*(E1+E2))+((p1+p2)*(p1+p2)))*0.001;
            Double_t Vcm=((p1+p2)/(E1+E2));
            energy->Fill(Ecm,1.);
            vel->Fill(Vcm,1.);
        }

        // Create "MCEvent" object
        ioTime.Start(kFALSE);
        // Write the "whole" event into ROOT file

```

```

    int runNumber = 777;
    int eventNumber = i;
    // delete this "event"
    ioTime.Stop();

    } //loop particulas

} //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");
energy->GetXaxis()->SetTitle("E(TeV)");
energy->GetYaxis()->SetTitle("Entradas");
energy->Write();
vel->Write();
}

```

Los parámetros que se especifican en este código son los mismos para todos los demás, lo que cambia son las ordenes para llenar determinados histogramas, así que solamente se mostraran el loop de eventos para los siguientes códigos.

Anexo A.2 Ángulos de Separación entre partículas del grupo 1

```

//=====
// HISTOGRAMAS
//=====
TH1F *ang1 = new TH1F("aqchi20","Angulo #tilde{q} y #tilde{#chi}_{0}^{2}", 100, 0, 180);
TH1F *ang2 = new TH1F("agj2","Angulo entre #tilde{#chi}_{0}^{2} y #tilde{#chi}_{0}^{1}", 100, 0, 180);
TH1F *ang3 = new TH1F("agz","Angulo #tilde{q} y Z", 100, 0, 180);
TH1F *ang4 = new TH1F("agchi10","Angulo #tilde{q} y #tilde{#chi}_{0}^{1}", 100, 0, 180);
TH1F *ang5 = new TH1F("agj1","Angulo #tilde{q} y jet", 100, 0, 180);

//=====
// GENERANDO EVENTOS
//=====
//Definiendo variables
Double_t pi = 4*atan(1);
Double_t px1, px4, px5, py1, py4, py5, pz1, pz4, pz5, p1, p4, p5, ang;
int aqj1 = 0;
int aqj12=0;
int aqj2 = 0;
int aqj22=0;

//Loop de eventos
for ( i=0; i<NEvents; i++ )
{
    if ( i%1 == 0 ) cout << "Event No.: " << i << endl;
    Pythia.GenerateEvent();

    //Loop de particulas
    for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
    {
        if (Pythia.GetK(npart,2)!=1000001 && Pythia.GetK(npart,2)!=-1000001 && Pythia.GetK(npart,2)!=
1000002 && Pythia.GetK(npart,2)!=-1000002 && Pythia.GetK(npart,2)!=1000003 &&
Pythia.GetK(npart,2)!=-1000003 && Pythia.GetK(npart,2)!=1000004 && Pythia.GetK(npart,2)!=
-1000004 && Pythia.GetK(npart,2)!=2000001 && Pythia.GetK(npart,2)!=-2000001 &&
Pythia.GetK(npart,2)!=2000002 && Pythia.GetK(npart,2)!=-2000002 && Pythia.GetK(npart,2)!=
2000003 && Pythia.GetK(npart,2)!=-2000003 && Pythia.GetK(npart,2)!=2000004 &&
Pythia.GetK(npart,2)!=-2000004 && Pythia.GetK(npart,2)!=1 && Pythia.GetK(npart,2)!=-1 &&
Pythia.GetK(npart,2)!=2 && Pythia.GetK(npart,2)!=-2 && Pythia.GetK(npart,2)!=3 &&

```

```

        Pythia.GetK(nparti,2)!=-3 && Pythia.GetK(nparti,2)!=4 && Pythia.GetK(nparti,2)!=-4 &&
        Pythia.GetK(nparti,2)!=1000022 && Pythia.GetK(nparti,2)!=23) continue;

//s-quarks
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)
    {
        if (Pythia.GetK(Pythia.GetK(nparti,3),2) != 0) continue;
        px1 = Pythia.GetP(nparti,1);
        py1 = Pythia.GetP(nparti,2);
        pz1 = Pythia.GetP(nparti,3);
        p1 = sqrt(px1*px1+py1*py1+pz1*pz1);
        aqz = 1;
        aqchi10 = 1;
        aqchi20 = 1;
        aqj2 = 1;
    }

//Z
if (Pythia.GetK(nparti,2) == 23)
{
    if (Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),3),2)!=0) continue;
    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);
    aqz2 = 1;
}

//~chi_10
if (Pythia.GetK(nparti,2)==1000022)
    {
        if (Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),3),2)!=0) continue;
        px3 = Pythia.GetP(nparti,1);
        py3 = Pythia.GetP(nparti,2);
        pz3 = Pythia.GetP(nparti,3);
        E3 = Pythia.GetP(nparti,4);
        p3 = sqrt(px3*px3+py3*py3+pz3*pz3);
        aqchi102 = 1;
        achi10chi20=1;
    }

//~chi_20
if (Pythia.GetK(nparti,2)==1000023)
    {
        if (Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),2)!=0) continue;
        px4 = Pythia.GetP(nparti,1);
        py4 = Pythia.GetP(nparti,2);
        pz4 = Pythia.GetP(nparti,3);
        p4 = sqrt(px4*px4+py4*py4+pz4*pz4);
        aqchi202 = 1;
        achi10chi202=1;
    }

//q
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)
    {
        if (Pythia.GetK(Pythia.GetK(nparti,3),2)!=1000001 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=
-1000001 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=1000002 &&
Pythia.GetK(Pythia.GetK(nparti,3),2)!=-1000002 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=
1000003 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=-1000003 &&
Pythia.GetK(Pythia.GetK(nparti,3),2)!=1000004 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=

```

```

-1000004 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=2000001 &&
Pythia.GetK(Pythia.GetK(nparti,3),2)!=-2000001 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=
2000002 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=-2000002 &&
Pythia.GetK(Pythia.GetK(nparti,3),2)!=2000003 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=
-2000003 && Pythia.GetK(Pythia.GetK(nparti,3),2)!=2000004 &&
Pythia.GetK(Pythia.GetK(nparti,3),2)!=-2000004)
    continue;

    if(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),2)!=0)continue;
    px5 = Pythia.GetP(nparti,1);
    py5 = Pythia.GetP(nparti,2);
    pz5 = Pythia.GetP(nparti,3);
    p5 = sqrt(px5*px5+py5*py5+pz5*pz5);
    aqj22 = 1;
}

    if (aqz==1 || aqz2==1)
    {
ang=acos((px1*px2+py1*py2+pz1*pz2)/(p1*p2))*180/pi;
ang3->Fill(ang,1.);
    }

    if (aqchi10==1 || aqchi102==1)
    {
ang=acos((px1*px3+py1*py3+pz1*pz3)/(p1*p3))*180/pi;
ang4->Fill(ang,1.);
    }

    if (aqchi20==1 || aqchi202==1)
    {
ang=acos((px1*px4+py1*py4+pz1*pz4)/(p1*p4))*180/pi;
ang1->Fill(ang,1.);
    }

    if (achi10chi20==1 || achi10chi202==1)
    {
ang=acos((px3*px4+py3*py4+pz3*pz4)/(p3*p4))*180/pi;
ang2->Fill(ang,1.);
    }

    if (aqj2==1 || aqj22==1)
    {
ang=acos((px1*px5+py1*py5+pz1*pz5)/(p1*p5))*180/pi;
ang5->Fill(ang,1.);
    }

// Create "MCEvent" object
ioTime.Start(kFALSE);
// Write the "whole" event into ROOT file
int runNumber = 777;
int eventNumber = i;
// delete this "event"
ioTime.Stop();

} //loop particulas

} //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");

ang1->GetXaxis()->SetTitle("grados");
ang1->GetYaxis()->SetTitle("Entradas");
ang2->GetXaxis()->SetTitle("grados");
ang2->GetYaxis()->SetTitle("Entradas");
ang3->GetXaxis()->SetTitle("grados");
ang3->GetYaxis()->SetTitle("Entradas");
ang4->GetXaxis()->SetTitle("grados");
ang4->GetYaxis()->SetTitle("Entradas");

```

```

ang5->GetXaxis()->SetTitle("grados");
ang5->GetYaxis()->SetTitle("Entradas");

ang1->Write();
ang2->Write();
ang3->Write();
ang4->Write();
ang5->Write();
}

```

Anexo A.3 Ángulos de Separación entre partículas del grupo 2

```

//=====
//      HISTOGRAMAS
//=====
TH1F *ang1 = new TH1F("agj1","Angulo #tilde{g} y jet1", 100, 0, 180);
TH1F *ang2 = new TH1F("agj2","Angulo #tilde{g} y jet2", 100, 0, 180);
TH1F *ang3 = new TH1F("agz","Angulo #tilde{g} y Z", 100, 0, 180);
TH1F *ang4 = new TH1F("agchi10","Angulo #tilde{g} y #tilde{#chi}_{0}^{1}", 100, 0, 180);
TH1F *ang5 = new TH1F("achi10chi20","Angulo #tilde{#chi}_{0}^{2} y #tilde{#chi}_{0}^{1}", 100, 0, 180);
TH1F *ang6 = new TH1F("agchi20","Angulo #tilde{g} y #tilde{#chi}_{0}^{2}", 100, 0, 180);

//=====
//      GENERANDO EVENTOS
//=====
//Definiendo variables
Double_t pi = 4*atan(1);
Double_t E1, E2, E3, E4, E5, E6, E7, E8, E9, m1, m2, m3, m4, m5, m6, m7, m8, m9, px1, px2, px10,px3, px4,
px5, px6, px7, px8, px9, py1, py2, py3, py4, py5, py6, py7, py8, py9,py10, pz10,pz1, pz2, pz3, pz4, pz5,
pz6, pz7, pz8, pz9, p1,p10, p2, p3, p4, p5, p6, p7, p8, p9, cos12,masa1,masa2,ang;
int agj1 = 0;
int agj12=0;
int agj2 = 0;
int agj22=0;
int agz = 0;
int agz2=0;
int agchi10 = 0;int agchi20=0; int agchi202=0;
int agchi102=0;int achi10chi20=0;int achi10chi202=0;

//Loop de Eventos
for ( i=0; i<NEvents; i++ )
{
    if ( i%1 == 0 ) cout << "Event No.: " << i << endl;
    Pythia.GenerateEvent();

    //Loop de Particulas
    for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
    {
        if (Pythia.GetK(npart,2)!=1000022 && Pythia.GetK(npart,2)!=1000021 && Pythia.GetK(npart,2)!=1 &&
Pythia.GetK(npart,2)!=-1 & Pythia.GetK(npart,2)!=2 && Pythia.GetK(npart,2)!=-2 &&
Pythia.GetK(npart,2)!=3 && Pythia.GetK(npart,2)!=-3 && Pythia.GetK(npart,2)!=4 &&
Pythia.GetK(npart,2)!=-4 && Pythia.GetK(npart,2)!=23) continue;

        // Gluinos
        if (Pythia.GetK(npart,2) == 1000021)
        {
            px1 = Pythia.GetP(npart,1);
            py1 = Pythia.GetP(npart,2);
            pz1 = Pythia.GetP(npart,3);
            E1 = Pythia.GetP(npart,4);
            p1 = sqrt(px1*px1+py1*py1+pz1*pz1);
            agz = 1;
            agchi10 = 1;
            agj1 = 1;
            agj2 = 1;
            agchi20=1;
        }
    }
}

```

```

}

//Z
if (Pythia.GetK(nparti,2) == 23)
{
  if(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),3),2)!=1000021) continue;
  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);
  agz2 = 1;
}

//~chi_10
if (Pythia.GetK(nparti,2)==1000022)
{
  if (Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),3),2)!=1000021) continue;
  px3 = Pythia.GetP(nparti,1);
  py3 = Pythia.GetP(nparti,2);
  pz3 = Pythia.GetP(nparti,3);
  E3 = Pythia.GetP(nparti,4);
  p3 = sqrt(px3*px3+py3*py3+pz3*pz3);
  agchi102 = 1;
  achii0chi20=1;
}

//~chi_20
if (Pythia.GetK(nparti,2)==1000023)
{
  if (Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),2)!=1000021) continue;
  px4 = Pythia.GetP(nparti,1);
  py4 = Pythia.GetP(nparti,2);
  pz4 = Pythia.GetP(nparti,3);
  p4 = sqrt(px4*px4+py4*py4+pz4*pz4);
  agchi102 = 1;
  achii0chi20=1;
  agchi202=1;
}

//q
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)
{
  if(Pythia.GetK(Pythia.GetK(nparti,3),2)==1000021)
  {
    px5 = Pythia.GetP(nparti,1);
    py5 = Pythia.GetP(nparti,2);
    pz5 = Pythia.GetP(nparti,3);
    E5 = Pythia.GetP(nparti,4);
    p5 = sqrt(px5*px5+py5*py5+pz5*pz5);
    agj12 = 1;
  }

  if (Pythia.GetK(Pythia.GetK(nparti,3),2)==1000001 || Pythia.GetK(Pythia.GetK(nparti,3),2)==
      -1000001 || Pythia.GetK(Pythia.GetK(nparti,3),2)==1000002 ||
      Pythia.GetK(Pythia.GetK(nparti,3),2)==-1000002 || Pythia.GetK(Pythia.GetK(nparti,3),2)==
      1000003 || Pythia.GetK(Pythia.GetK(nparti,3),2)==-1000003 ||
      Pythia.GetK(Pythia.GetK(nparti,3),2)==1000004 || Pythia.GetK(Pythia.GetK(nparti,3),2)==
      -1000004 || Pythia.GetK(Pythia.GetK(nparti,3),2)==2000001 ||
      Pythia.GetK(Pythia.GetK(nparti,3),2)==-2000001 || Pythia.GetK(Pythia.GetK(nparti,3),2)==
      2000002 || Pythia.GetK(Pythia.GetK(nparti,3),2)==-2000002 ||
      Pythia.GetK(Pythia.GetK(nparti,3),2)==2000003 || Pythia.GetK(Pythia.GetK(nparti,3),2)==
      -2000003 || Pythia.GetK(Pythia.GetK(nparti,3),2)==2000004 ||
      Pythia.GetK(Pythia.GetK(nparti,3),2)==-2000004)
  {
    if(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),2)==1000021)
    {
      px6 = Pythia.GetP(nparti,1);
    }
  }
}

```

```

    py6 = Pythia.GetP(nparti,2);
    pz6 = Pythia.GetP(nparti,3);
    E6 = Pythia.GetP(nparti,4);
    p6 = sqrt(px6*px6+py6*py6+pz6*pz6);
    agj22 = 1;
  }
}

    if (agj1==1 || agj12==1)
    {
ang=acos((px1*px5+py1*py5+pz1*pz5)/(p1*p5))*180/pi;
ang1->Fill(ang,1.);
    }

    if (agj2==1 || agj22==1)
    {
ang=acos((px1*px6+py1*py6+pz1*pz6)/(p1*p6))*180/pi;
ang2->Fill(ang,1.);
    }

    if (agz==1 || agz2==1)
    {
ang=acos((px1*px2+py1*py2+pz1*pz2)/(p1*p2))*180/pi;
ang3->Fill(ang,1.);
    }

    if (agchi10==1 || agchi102==1)
    {
ang=acos((px1*px3+py1*py3+pz1*pz3)/(p1*p3))*180/pi;
ang4->Fill(ang,1.);
    }

    if (achi10chi20==1 || achi10chi202==1)
    {
ang=acos((px10*px3+py10*py3+pz10*pz3)/(p10*p3))*180/pi;
ang5->Fill(ang,1.);
    }

    if (agchi20==1 || agchi202==1)
    {
ang=acos((px4*px1+py4*py1+pz4*pz1)/(p4*p1))*180/pi;
ang6->Fill(ang,1.);
    }

// Create "MCEvent" object
ioTime.Start(kFALSE);
// Write the "whole" event into ROOT file
int runNumber = 777;
int eventNumber = i;
// delete this "event"
ioTime.Stop();

} //loop particulas

} //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");

ang1->GetXaxis()->SetTitle("grados");
ang1->GetYaxis()->SetTitle("Entradas");
ang2->GetXaxis()->SetTitle("grados");
ang2->GetYaxis()->SetTitle("Entradas");
ang3->GetXaxis()->SetTitle("grados");
ang3->GetYaxis()->SetTitle("Entradas");
ang4->GetXaxis()->SetTitle("grados");
ang4->GetYaxis()->SetTitle("Entradas");

```

```

ang5->GetXaxis()->SetTitle("grados");
ang5->GetYaxis()->SetTitle("Entradas");
ang6->GetXaxis()->SetTitle("grados");
ang6->GetYaxis()->SetTitle("Entradas");

ang1->Write();
ang2->Write();
ang3->Write();
ang4->Write();
ang5->Write();
ang6->Write();
}

```

Anexo A.4 Masa invariante y Separación Angular dimuones

```

//=====
//      HISTOGRAMAS
//=====
TH1F *mass1 = new TH1F("histmass1","Masa Invariante Dimuones", 100, 0, 400);
TH1F *mass2 = new TH1F("histmass2","Masa Invariante Dimuones", 100, 0, 400);
TH1F *ang1 = new TH1F("histang1","Separacion Angular entre Muones", 100, 0, 180);
TH1F *ang2 = new TH1F("histang2","Separacion Angular entre Muones", 100, 0, 180);

//=====
// GENERANDO DE EVENTOS
//=====
//Definiendo variables
Double_t E1, E2, E3, E4, px1, px2, px3, px4, py1, py2, py3, py4, pz1, pz2, pz3, pz4, p1, p2, p3, p4,
        masainv, ang;
Double_t pi = 4*atan(1.);
int agmm = 0;
int agmm2 = 0;
int aqmm = 0;
int aqmm2=0;

//Loop de Eventos
for ( i=0; i<NEvents; i++ )
{
    if ( i%1 == 0 ) cout << "Event No.: " << i << endl;
    Pythia.GenerateEvent();

    //Loop de Particulas
    for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
    {
        if (Pythia.GetK(npart,2) != 13 && Pythia.GetK(npart,2) != -13)continue;

        //Solamente mu-
        if (Pythia.GetK(npart,2) == 13)
        {
            //mu- que pertenecen al grupo 1
            if(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(npart,3),3),3),3),3),2)==0)
            {
                px1 = Pythia.GetP(npart,1);
                py1 = Pythia.GetP(npart,2);
                pz1 = Pythia.GetP(npart,3);
                E1 = Pythia.GetP(npart,4);
                p1 = sqrt(px1*px1+py1*py1+pz1*pz1);
                aqmm=1;
            }

            //mu- que pertenecen al grupo 2
            if(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(npart,3),3),3),3),3),2)==1000021)
            {
                px2 = Pythia.GetP(npart,1);
                py2 = Pythia.GetP(npart,2);
                pz2 = Pythia.GetP(npart,3);
            }
        }
    }
}

```

```

E2 = Pythia.GetP(nparti,4);
p2 = sqrt(px2*px2+py2*py2+pz2*pz2);
agmm=1;
}
}

//Solamente mu+
if (Pythia.GetK(nparti,2) == -13)
{
//mu+ que pertenecen al grupo 1
if(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),3),3),2)==0)
{
px3 = Pythia.GetP(nparti,1);
py3 = Pythia.GetP(nparti,2);
pz3 = Pythia.GetP(nparti,3);
E3 = Pythia.GetP(nparti,4);
p3 = sqrt(px3*px3+py3*py3+pz3*pz3);
aqmm2=1;
}

//mu+ que pertenecen al grupo 2
if(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(Pythia.GetK(nparti,3),3),3),3),2)==1000021)
{
px4 = Pythia.GetP(nparti,1);
py4 = Pythia.GetP(nparti,2);
pz4 = Pythia.GetP(nparti,3);
E4 = Pythia.GetP(nparti,4);
p4 = sqrt(px4*px4+py4*py4+pz4*pz4);
agmm2=1;
}
}

//muones del grupo 1
if(aqmm==1 || aqmm2==1)
{
ang=acos((px1*px3+py1*py3+pz1*pz3)/(p1*p3))*180/pi;
masainv=sqrt(((E1+E3)*(E1+E3))-((px1+px3)*(px1+px3))-((py1+py3)*(py1+py3))-((pz1+pz3)*(pz1+pz3)));
ang1->Fill(ang,1.);
mass1->Fill(masainv,1.);
}
//muones del grupo 2
if(agmm==1 || agmm2==1)
{
ang=acos((px2*px4+py2*py4+pz2*pz4)/(p2*p4))*180/pi;
masainv=sqrt(((E2+E4)*(E2+E4))-((px2+px4)*(px2+px4))-((py2+py4)*(py2+py4))-((pz2+pz4)*(pz2+pz4)));
ang2->Fill(ang,1.);
mass2->Fill(masainv,1.);
}

// Create "MCEvent" object
ioTime.Start(kFALSE);
// Write the "whole" event into ROOT file
int runNumber = 777;
int eventNumber = i;
// delete this "event"
ioTime.Stop();

} //loop particulas

} //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");

mass1->GetXaxis()->SetTitle("M_{#mu#mu}(GeV)");
mass1->GetYaxis()->SetTitle("Entradas");
mass2->GetXaxis()->SetTitle("M_{#mu#mu}(GeV)");
mass2->GetYaxis()->SetTitle("Entradas");
ang1->GetXaxis()->SetTitle("grados");

```

```

ang1->GetYaxis()->SetTitle("Entradas");
ang2->GetXaxis()->SetTitle("grados");
ang2->GetYaxis()->SetTitle("Entradas");
ang1->Write();
ang2->Write();
mass1->Write();
mass2->Write();

}

```

Anexo A.5 Energia Transversal Faltante

```

//=====
//      HISTOGRAMAS
//=====
TH1F *etmiss = new TH1F("histetmiss","Energia Transversal Faltante", 100, 0, 1000);

//=====
//      GENERANDO EVENTOS
//=====
//Definiendo variables
Double_t pi = 4*atan(1.);
Double_t Etmis = 0;
Double_t etmx = 0;
Double_t etmy = 0;

//Loop de eventos
for ( i=0; i<NEvents; i++ )
{
    if ( i%1 == 0 ) cout << "Event No.: " << i << endl;
    Pythia.GenerateEvent();

    //Loop de particulas
    for (Int_t nparti=1; nparti<=Pythia.GetN(); nparti++)
    {
        //solamente particulas finales
        if (Pythia.GetK(nparti,1) != 1) continue;
        if (Pythia.GetK(nparti,2) == 1000022) 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,3);
        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);
        Etmis = sqrt(etmx*etmx+etmy*etmy);
        etmiss->Fill(Etmis,1.);

        // Create "MCEvent" object
        ioTime.Start(kFALSE);
        // Write the "whole" event into ROOT file
        int runNumber = 777;
        int eventNumber = i;
        // delete this "event"
        ioTime.Stop();

    }//loop particulas

}

//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");

```

```
etmiss->GetXaxis()->SetTitle("E_{T}^{miss}(GeV)");  
etmiss->GetYaxis()->SetTitle("Entradas");  
etmiss->Write();  
}
```