A General Search for New Physics Using the ATLAS Detector at the Large Hadron Collider at CERN

Jason Boomsma
Argonne National Laboratory, Science Undergraduate
Laboratory Internship Program

Anderson University, Anderson, IN

Argonne National Laboratory
Argonne, IL

July 31, 2009

Participant: _____
Signature

Research Advisor: _____
Signature

ABSTRACT

A General Search for New Physics Using the ATLAS Detector at the Large Hadron Collider at CERN

Jason Boomsma (Anderson University, Anderson, IN 46012), Sergei Chekanov (Argonne National Laboratory, Argonne, IL 60439).

A general search is a relatively new idea in high energy physics. A general search analyzes all data that has been collected and attempts to find new particles over the full range of that data. The program InvMass has been developed to perform a general search, specifically through using the ATLAS detector at the Large Hadron Collider (LHC), a proton-proton collider, at CERN. The search to be performed by InvMass will look for statistically significant variations from the Standard Model's predictions. To do so, InvMass will find all possible and relevant particle groups contained in the data collected from the ATLAS detector. These particle groups will be analyzed, and the invariant mass and the total transverse momenta of the groups will be recorded in histograms. The generic code of InvMass allows it to be easily adapted for work with any particle types that may be found using the ATLAS detector. Through looking at the histograms InvMass will create, it will be possible to find signatures of new particles which have decayed into certain groups of particles. Since the LHC at CERN is not currently running, there is no real data to analyze using InvMass. However, InvMass was tested using simulation generated data and performed as expected. When the LHC begins running and the ATLAS detector has collected real data, InvMass will be used for its intended purpose.

INTRODUCTION

The laboratory at CERN houses the worlds largest and highest energy particle collider to date, the Large Hadron Collider (LHC). The LHC is a proton-proton collider which has not yet begun operation. The expected high energy levels of the collisions at the LHC will allow physicists to search for new particles that can corroborate or disprove current theories such as the Standard Model (SM).

One of the detectors along the LHC's beam line is the ATLAS detector. This is a general purpose detector that will be used to identify many types of known particles which will be produced when the LHC begins its operation.

One promising path to new discoveries at the LHC using the ATLAS detector is through a general search. The vast majority of particles yet to be found are thought to decay into groups of "daughter" particles. These daughter particles will have unique amounts of energy and momenta based on the mass of the "parent" particle. By going through each event that the ATLAS detector will record and grabbing each unique group of particles from the events, it will be possible to fill histograms with useful data about the particle groups. By analyzing these histograms, new parent particles may be found. A general search analyzes all the data collected and attempts to find new parent particles over the full range of that data.

To handle the scale of general search, a computer program can be used. Since it is possible to identify many different particles using the ATLAS detector, and since parent particles can decay into any number of daughter particles, the number of histograms to create and analyze becomes very large. For example, if it is possible to identify 7 types of particles and a search is made for parent particles that decay into 2, 3, or 4 daughter particles, 322 unique daughter groups must be investigated. If the search includes groups of 5 daughter particles as well, the

number jumps to 784 groups. To manually code every possible outcome would be difficult and time consuming, but a computer program can determine each possible grouping easily.

The idea for a general search is not new. A general search was preformed using the H1 detector at HERA, an electron-proton collider. The HERA search [3] broke particles into unique particle groups and analyzed the transverse momenta (PT), invariant mass (IM), and event distribution of the groups. The same type of search was run over old data that the Tevatron collider at Fermi National Laboratory had collected. Neither search provided conclusive results.

InvMass, a C++ based program, has been written to execute a general search using ATLAS detector data. InvMass will separate the particles found in each event into all possible groups of daughter particles. Using those groups it will fill corresponding histograms with useful information about the groups of particles; their event rate, IM and total PT. InvMass can also automatically analyze the histograms and output a summary of the peaks that may be found.

A GENERAL SEARCH

The LHC will reach higher energies than any other particle collider ever built when it begins operation. Using the detectors positioned around the beam line physicists hope to find many new, undiscovered particles which will provide a better understanding of the universe. Particles like the Higgs Boson, a possible SM facilitator particle for mass, have been predicted by physicists for years, and the new energy ranges that the LHC will attain may allow these particles to finally be observed. Table 1 shows a short summary of particles that will be searched for using the ATLAS detector at the LHC.

However, physicists are even more excited about the unknown. It is exciting to realize that new, unforeseen physics may be observed at the LHC. Exotic physics will be a large part of the searches being performed using the LHC data.

In both cases, a general search program would be helpful. Where a range is predicted for finding a certain particle, that range can be analyzed through the general search and perhaps the particle could be identified. For unforeseen physics, the broad view of the general search may unearth possibilities in ranges unlikely to be searched otherwise. A general search will be a first report of interesting areas, it will help physicists explore existing theories and perhaps it will even reveal new searches that should be performed.

PROGRAM: INVMASS

As stated, InvMass is a program that has been written to execute a general search. Written in C++, it makes use of the ROOT framework for data analysis. The architecture of InvMass is compartmentalized so that the code is easier to understand and edit. However, InvMass is just one step in the data analysis process. It requires an Athena based program to create the ROOT files it uses.

InvMass fits in at the third step of the data collection and analysis process for the ATLAS detector. As you can see in Figure 1, the ATLAS detector will collect data about the collisions it observes as either Event Summary Data (ESD), Analysis Object Data (AOD), or Derived Physics Data (DPD). In order to work interactively with the data the ATLAS detector will record, an Athena based program is needed, in this case PromptGamma. PromptGamma is an Athena program created by Sergei Chekanov which is used to create ROOT nTuples from the AOD the ATLAS detector collects. The nTuples organize the data by particle type and by individual particles. They are separated by event, so that each time protons collide within the detector it can be determined what the end product of the collision was. The data to be stored includes the momentum, energy, and charge of each particle found in the events. Once the nTuples are created by the Athena program, InvMass can be used. InvMass will execute a general search, and by

analyzing the ROOT nTuples it will output histograms which will be used to identify parent particles. Once those histograms are created, they can be analyzed further by other programs if desired.

ORGANIZATION

Performing a general search is a simple idea, but executing it well requires that complex code be used. There are four steps in InvMass that are used to execute the search. An overview of these steps will shed light on the overall architecture of InvMass and how it works. Figure 2 may be a helpful visual representation of the description.

The first step is the file loop. InvMass is written such that it can read multiple ROOT files each time that it is run. Therefore no matter how many ROOT nTuples are being analyzed, all of the data is outputted in one location. The file loop is located in the main function of the program and makes use of a class called Class Global to hold the locations of the files to run over.

Within the file loop is the next step, the event loop. As previously stated, ROOT nTuples are organized by events. The event loop is nested in the file loop and uses a separate class, Class EventLoop, to access the individual events from the ROOT files. Within the event loop the particles from each event are accessed and stored, organized by the particle's type.

Next in line is the particle-type loop. Nested within the event loop, the particle-type loop is executed by Class InvMass. Within the event loop Class InvMass is passed TClonesArrays, arrays that are full of objects found in ROOT libraries, for each particle type. The particle-type loop collects each possible unique combination of these particle types (see Table 2).

Lastly, there is the particle loop. Nested within the particle-type loop, it is the deepest and final step of InvMass. Also executed by Class InvMass, the particle loop runs through the particle-type groups and finds each unique group of individual particles to analyze (Table 2). The

analysis of the general search is preformed within this final loop, and the IM and total PT of the particle groups is determined. The information is stored in histograms based on the particle-type group the particles are from. For example, the IM of all electron-electron pairs are stored in one histogram, and the total PT of all jet-jet-gamma groups are stored in another, separate histogram.

FEATURES

InvMass has many different features which can be accessed. The general search can be altered to fit different data sets and search ranges, while maintaining its generality. The search can perform basic cuts to maintain its integrity, and the output can be controlled to the specifications desired. All of the features can be changed with little to no alteration of the code, so the program will rarely need to be recompiled. This is possible through the use of a configuration file.

Data Sets and Search Ranges

InvMass can be easily altered to work for different data sets and search ranges. There are a few options that control these tools. InvMass can be altered to work for a specific data set based on the particle types the data contains. It can be modified to select which of those particle types to use in the search. Furthermore, InvMass can be told how many particles to place in the decay groups being searched.

To alter InvMass to fit a different data set is done through coding. It is the only change which requires the program to be recompiled. In this case, a different data set is the set of all data files stored in ROOT nTuples which contain data for a certain group of particle types. For example, if one data set contains information of electrons, jets, and photons, a different data set may contain information about electrons, photons, and muons. To alter InvMass to a specific data set requires an alteration of the code in the event loop. Within the event loop a vector is filled

with the TClonesArrays of each particle type. But for each different data set, the particle-types change in some way. The vector created and passed by the event loop must be altered to contain every particle type available and only those particle types which are available based on the ROOT nTuples being analyzed. Once this is changed, the program must be recompiled before it is used for that specific data set.

However, in order to modifying InvMass to choose the particle types to use does not require coding. The configuration file for InvMass is used to make this change, so the program does not need to be recompiled. Within the configuration file there is a list of particle names. This list must match exactly to the vector created in the event loop, in the correct order, or else the output data will be misrepresented – perhaps by electrons being labeled jets and vice versa. Once the list is correct and complete, the particle types to use in the search can be chosen simply by commenting out those which are not desired. When a particle type is commented out, it is removed from the vector created in the event loop before the particle type loop is performed.

Likewise, changing the number of particles to use in the decay groups is set in the configuration file. There is a value, numDecays, which can be set to any number 2 or greater. For example, if numDecays is set to 4 the program will than perform a search for all decay groups that contain 2, 3, or 4 particles. Any value can be chosen, there is no limit – the search can be run for decay groups of 2 to 5, 2 to 10, and even 2 to 100 if desired. Since the feature is set through the configuration file, the program does not need to be recompiled.

Performing Cuts

InvMass also has the capability to perform generic cuts to maintain the integrity of the search. There are three cuts available; a minimum cut on PT for individual particles, a maximum rapidity cut for individual particles, and overlap removal for particles that were counted in two

categories. The cuts are performed before the particle type loop is executed for higher efficiency. All cuts are set in the configuration file.

First there is the PT cut. It is executed on a particle type basis, so each type is given a minimum value. Each individual particle is checked against its type's minimum value. If the particle's PT is too low, the particle is removed from the TClonesArray it was stored in.

The rapidity cut is performed in a similar way. There is one major difference though – only one cut is performed on all particles, regardless of their type. So, there is only one value to set, the maximum pseudorapidity value. As with the PT cut, all individual particles are checked and if a particle has a pseudorapidity value greater than the value set, it is removed from the TClonesArray that it was stored in.

There is a third type of cut, overlap removal. This cut has a boolean option to turn it on and off, as well as specific options for how to perform it. When data is collected and stored, some particles can be tagged as two different types, for example as both a jet and an electron. Only one of the tags is correct, so the other should be removed in order to maintain data integrity. The problem for InvMass became how to perform a general cut using overlap removal without knowing what the particles being cut were. The program needed to make a decision about which of the tags should be kept. So, a precedence value was added for each particle type. The particle type which has the higher precedence value is the particle type the double counted particle is removed from. However, there must also be some value used to determine if a particle is being double counted. Therefore, the value of dR is used; particles must be greater than a minimum dR value apart or else they are the same particle. If they are the same particle, they are removed according to the precedence value from the associated TClonesArray.

Managing Output

InvMass's output can be controlled to the specifications desired. There are three main sections of output that can be modified; the output file name, the histogram bin and range details, and peak searching. All are fully modifiable through the configuration file.

The output file name is customizable. It is set as a relative path, so it can be placed anywhere relative to the executable of the program. By default it is set to write to the "out/" folder. The file name is set as a stub, without the file extension. The extension is added by InvMass, based on what the output is, either the ROOT file which holds the histograms or the text file which holds the summary of the peaks found by the peak search.

The histogram details are also fully customizable. There are two types of histograms created by InvMass; IM histograms and total PT histograms. Both types can be set to different specifications. The user determines the number of bins and the range covered by the histograms.

Finally, there is a peak searching tool. There is a boolean option to turn peak searching on and off. The peak search InvMass uses is done by an object of the TSpectrum class from ROOT. When the search is performed there are many different options to control how it determines which peaks are important enough to mark and which histograms are important enough to search. The ideal values for these search parameters have yet to be found.

TEST RUNS

Although there is no data being collected at the LHC right now it was still necessary to test InvMass. So, using Monte Carlo simulations of ATLAS data collection, PromptGamma was used to create ROOT nTuples of various data sets. The data sets were split into two runs; one for all of the nTuples containing 6 particle types and another for all of the nTuples containing 7 particle types.

Both runs were successful. All features were checked and manipulated, and they behaved

as expected. For the output shown here, both runs were set to the same configurations. The PT cut was set to 20 GeV for all particles, the rapidity cut was set to 3, and the dR value for overlap removal was set to 0.3. All particles types in the data sets were used, and decay groups of up to 4 particles were analyzed. The settings for peak searching were as follows: maximum peaks set to 3, sigma set to 8, average window set to 7, deconvolution iterations set to 3, resolution set to 1, and minimum number of events set to 100. The histograms were all created with 500 bins over the range of 0 to 500 GeV.

Run 1 (6 Particles)

Run 1 used ROOT nTuples with 6 particles. The nTuples were created by PromptGamma and included electrons, jets, photons, muons, Z bosons, and missing ET particles. The ROOT files contained some specific decays, such as Z-bosons to two electrons and the "graviton" to two photons. The decays planted in the files were seen perfectly in the histograms and the peaks were marked properly by the peak search function, as you can see in Figures 3 and 6. Figures 4, 5, 7, and 8 show some other output from the run. Figure 5 shows a perfect example of peak reflection, where a peak from a known decay is found in another graph – in this case, the graviton.

Run 2 (7 Particles)

Run 2 used ROOT nTuples with 7 particles. The nTuples were created by PromptGamma and included electrons, jets, photons, muons, Z bosons, taus, and missing ET particles. The ROOT files contained many of the same planted decays as run 1, and they were found and marked properly as seen in Figure 9. The Monte Carlo selection of tau particles was not functioning properly at the time of testing, so there were very few tau particles collected in the nTuples. Figures 10 and 12 show this lack of tau particles. The remaining Figures 11, 13, and 14 show sample output from the run, with Figure 11 showing a small peak reflection from the

graviton decay.

CONCLUSION

InvMass performs exactly as it was created to. Using generic algorithms it performs a general search for new particles over a wide search range. All possible particle groups are found and analyzed through the four steps of the program. The features can all be accessed easily through a configuration file, and the program is easily adaptable to work for any collection of particles that the ATLAS detector may record. Some work must be done to fully understand the peak search options and find the ideal settings, though the peak search is still fully functioning.

Performing a general search at the LHC using the ATLAS detector will be very useful. InvMass has shown that it can identify peaks corresponding to particles that have decayed, and it is quite possible that through using this program physicists working at the ATLAS detector will be able to find predicted particles and even unpredicted particles.

A general search does have its limitations. The broad range does not allow for a very precise search with detailed background removal. Nevertheless, it can be used as a first report and a starting point for more detailed searches, and can provide a broad view of decay patterns for further studies.

ACKNOWLEDGMENTS

REFERENCES

[1]   ATLAS Collaboration, "Other physics beyond the Standard Model," in *ATLAS Detector and Physics Performance Technical Design Report*, 25 May 1999, ch. 21, pp. 915-954.

[2]   B.C. Allanach, C.M. Harris, M.A. Parker, P. Richardson, and B.R. Webber, "Detecting exotic heavy leptons at the Large Hadron Collider," *Journal of High Energy Physics*, vol. 0108, pp. 51-65, Nov. 2001.

[3]   H1 Collaboration, "A general search for new phenomena in ep scattering at HERA," *Physics Letters B*, vol. 602, pp. 14-30, Nov. 2004.

[4]   K. Benslama for the ATLAS and CMS Collaborations, "Other Exotic Scenarios at the LHC," 2007, http://arxiv.org/abs/0708.1592v1.

[5]   M.U. Mozer, "LHC searches for high-mass resonances decaying to leptons or photons," Prepared for 16th International Workshop on Deep Inelastic Scattering and Related Subjects (DIS 2008), London, England, 7-11 Apr. 2008.

[6]   N. V. Krasnikov and V. A. Matveev, "Search for new physics at LHC," *Physics-Uspekhi*, vol. 47, pp. 643-670, 2004.
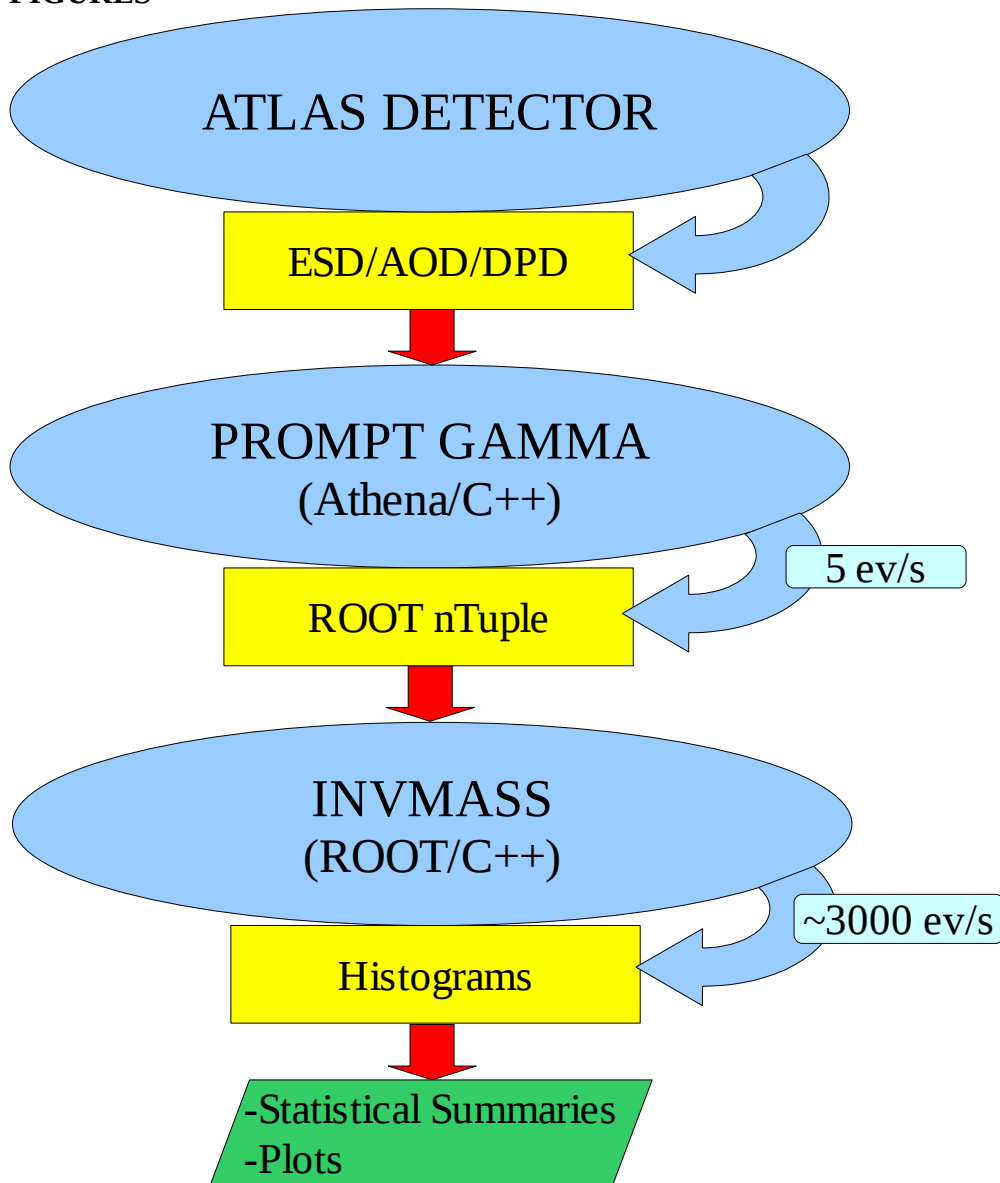
FIGURES

Figure 1: Flowchart for data analysis using ATLAS detector data. Yellow boxes are the output data.
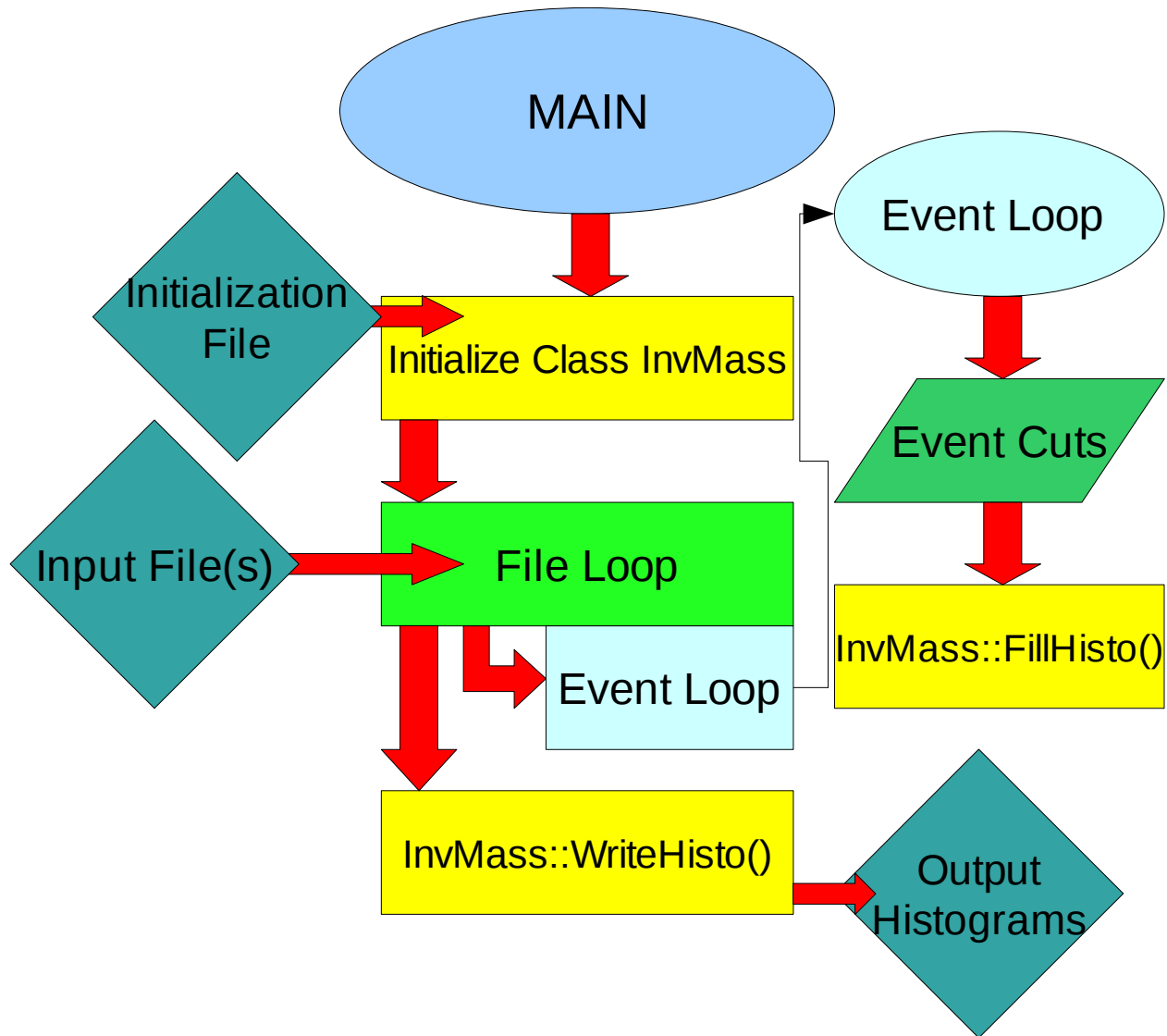
Figure 2: A diagram showing the basic architecture of the program InvMass, with input and output data labeled.
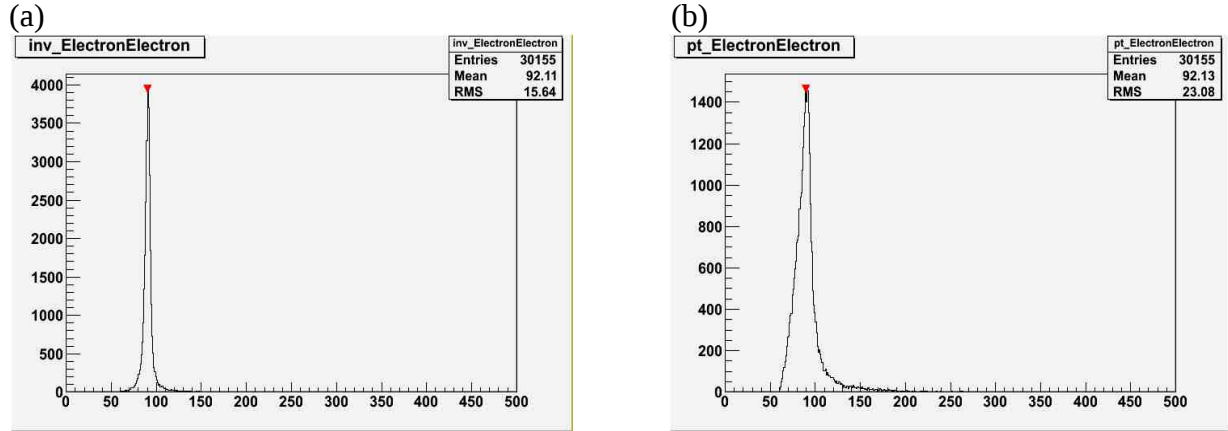
Figure 3: Histograms showing the 6 particle test run's electron-electron group's (a) invariant mass, and (b) total transverse momenta. Peak near 100 GeV is Z-boson decay to e+e-.
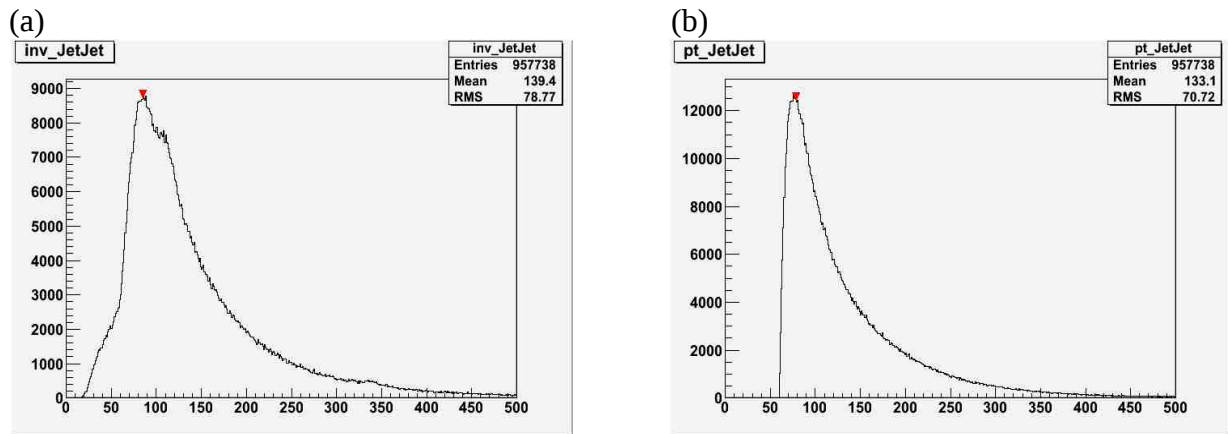

Figure 4: Histograms showing the 6 particle run's jet-jet group's (a) invariant mass, and (b) total transverse momenta.
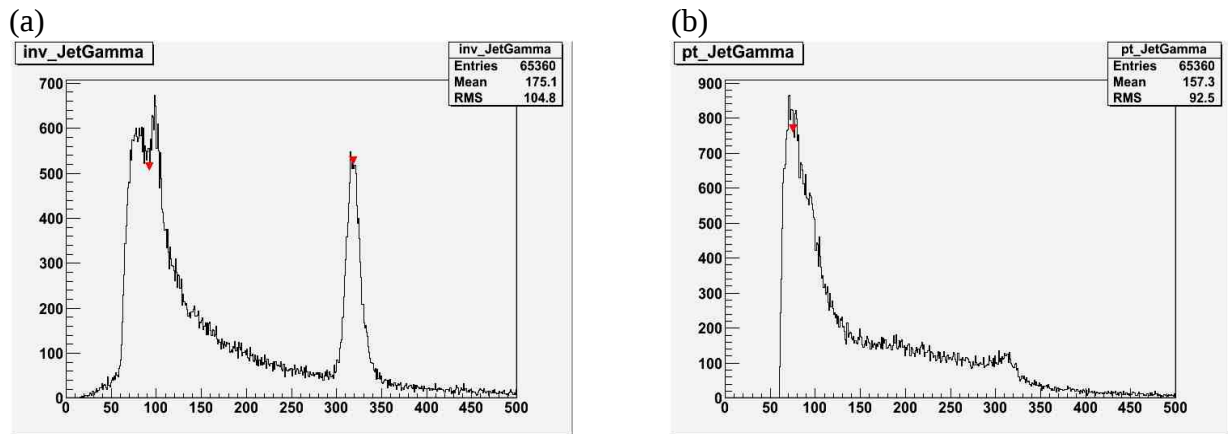

Figure 5: Histograms showing the 6 particle run's jet-photon group's (a) invariant mass, and (b) total transverse momenta. Peak near 300 GeV is a reflection of graviton decay to photon-photon.
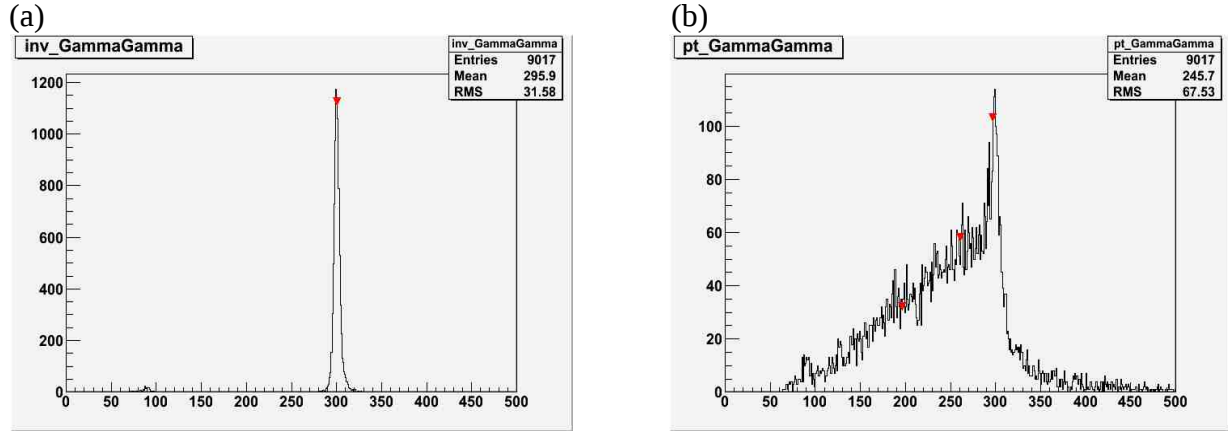
(a)

(b)

Figure 6: Histograms showing the 6 particle run's photon-photon group's (a) invariant mass, and (b) total transverse momenta. Peak near 300 GeV is graviton decay to photon-photon.
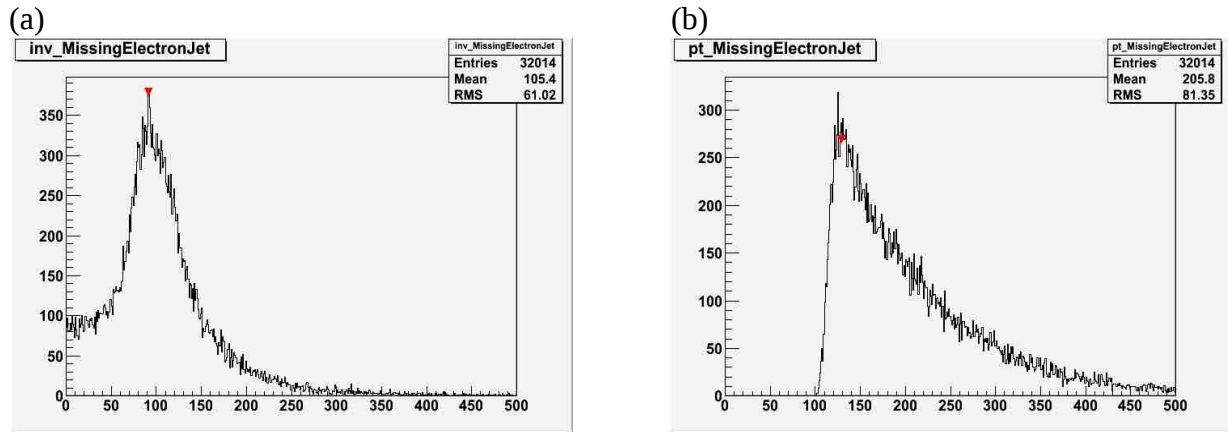
(a)

(b)

Figure 7: Histograms showing the 6 particle run's missingET-electron-jet group's (a) invariant mass, and (b) total transverse momenta.
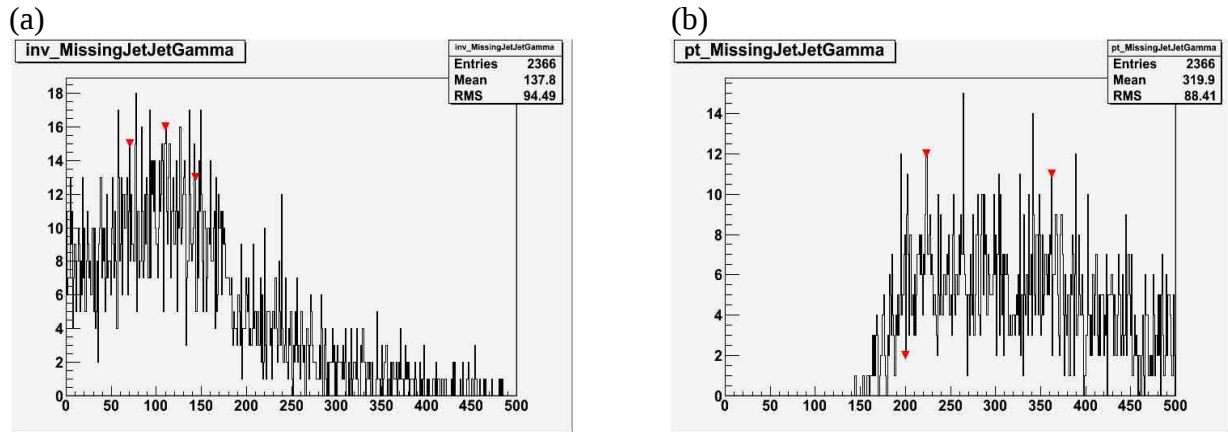
(a)

(b)

Figure 8: Histograms showing the 6 particle run's missingET-jet-jet-photon group's (a) invariant mass, and (b) total transverse momenta.
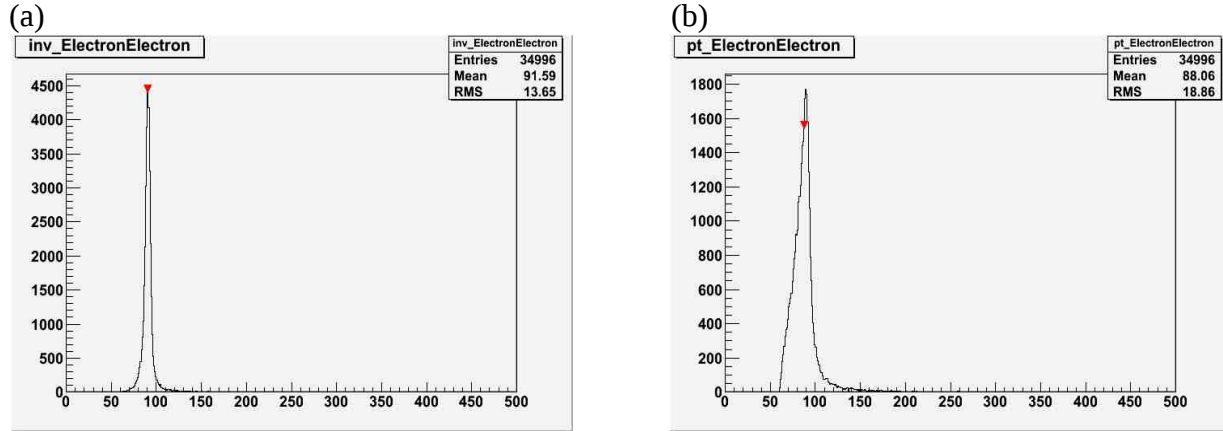
(a)

(b)

Figure 9: Histograms showing the 7 particle run's electron-electron group's (a) invariant mass, and (b) total transverse momenta. Peak near 100 GeV is Z-boson decay to e+e-.
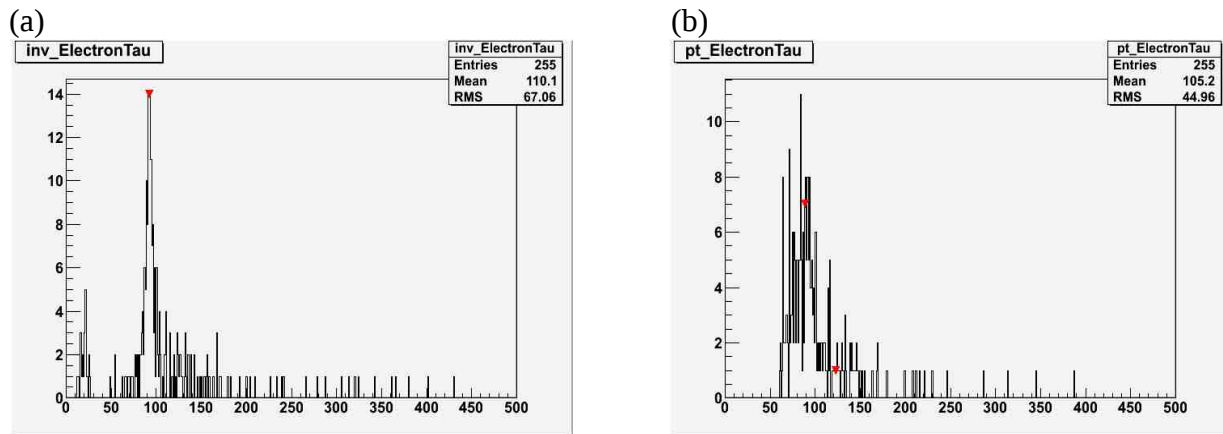
(a)

(b)

Figure 10: Histograms showing the 7 particle run's electron-tau group's (a) invariant mass, and (b) total transverse momenta. Low events due to lack of functioning tau selection.
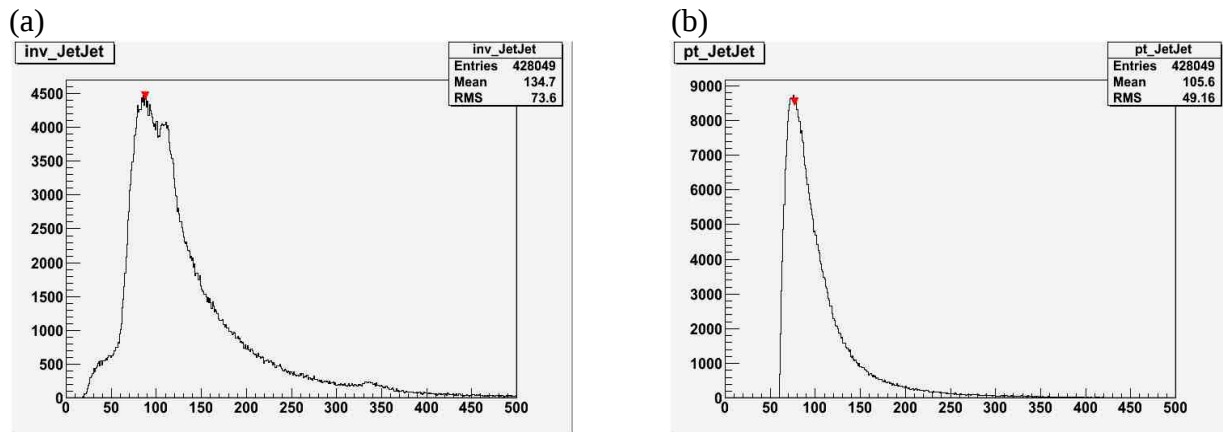
(a)

(b)

Figure 11: Histograms showing the 7 particle run's jet-jet group's (a) invariant mass, and (b) total transverse momenta. Bump near 350 GeV is a reflection of graviton decay to photon-photon.

(a)

(b)
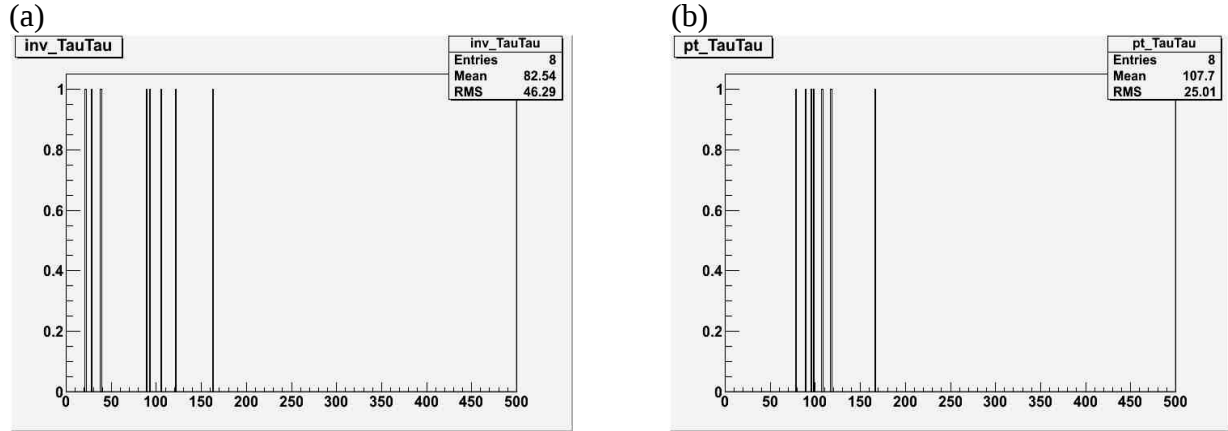
Figure 12: Histograms showing the 7 particle run's tau-tau group's (a) invariant mass, and (b) total transverse momenta. Low events due to lack of functioning tau selection.

(a)

(b)

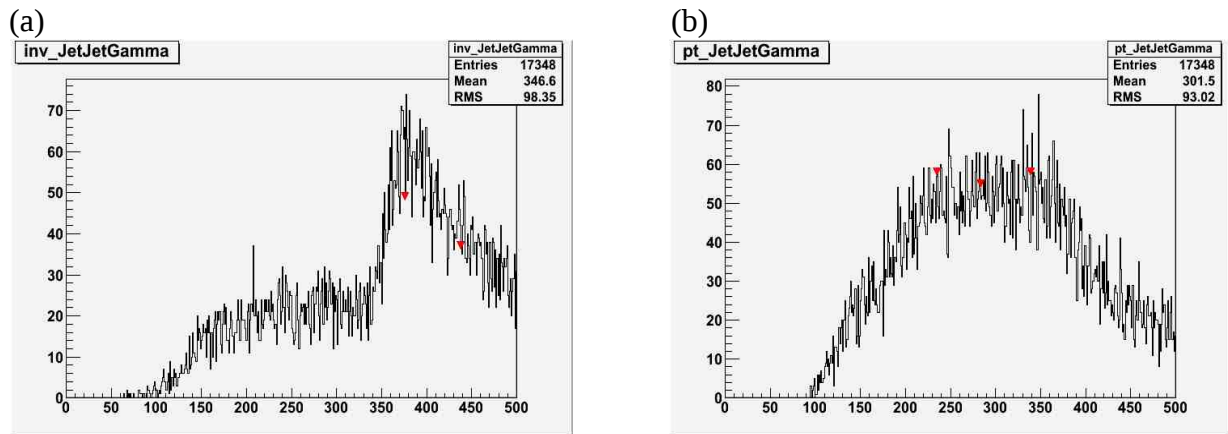
Figure 13: Histograms showing the 7 particle run's jet-jet-photon group's (a) invariant mass, and (b) total transverse momenta.
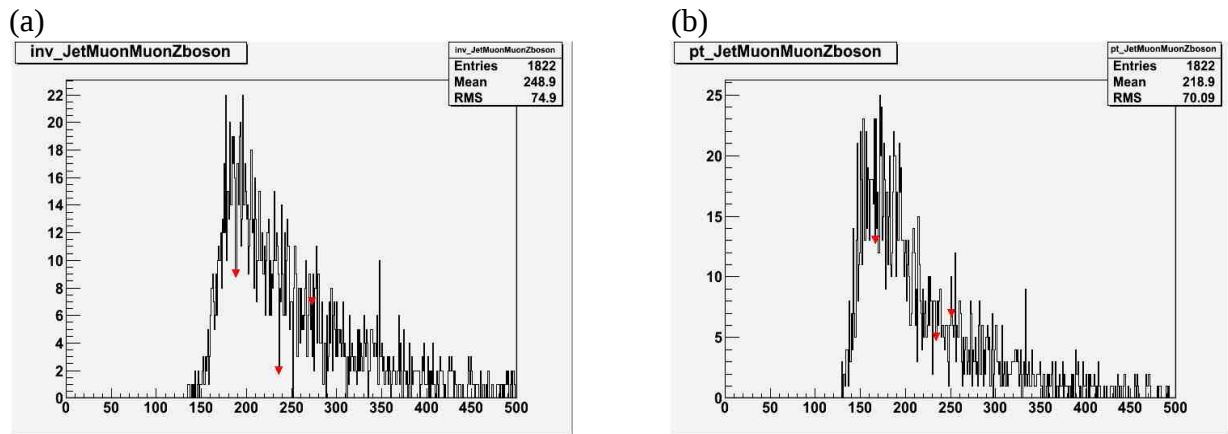
(a)

(b)

Figure 14: Histograms showing the 7 particle run's jet-muon-muon-zboson group's (a) invariant mass, and (b) total transverse momenta.

TABLES

Searches to be Performed at the LHC				
Particle	**Decay Group**	**Energy**	**Width**	**Reference**
Graviton	e+, e-	~1TeV	~10 GeV	[6]
W'	e, v	~4 TeV	---	[6]
New high mass resonances	e, e; m, m; p, p	>1 TeV	---	[5]
LeptoQuark (LQ)	e, j	1, 1.2, 1.3, 1.5 TeV	~50 GeV	[4]
New Excited lepton	e, j, j	.5, 1, 2, 3, 4 TeV	---	[4]
Exotic heavy Leptons:	l, l	175.41 Gev	7.64 GeV	[2]
		498.97 GeV	38.75 GeV	[2]
		767.70 GeV	70.70 GeV	[2]
		898.86 GeV	104.85 GeV	[2]
Excited Down Quark (d*)	a-l-l	1 TeV	37.3 GeV	[1]
	d-g		30.9 GeV	[1]
	u-W		4.26 GeV	[1]
	d-Z		1.98 GeV	[1]
	d-p		0.22 GeV	[1]
Excited Up Quark (u*)	a-l-l	1 TeV	37.3 GeV	[1]
	u-g		30.9 GeV	[1]
	d-W		4.26 GeV	[1]
	u-Z		1.32 GeV	[1]
	u-p		0.87 GeV	[1]
Excited Down Quark (d*)	a-l-l	3 TeV	102 GeV	[1]
	d-g		82.4 GeV	[1]
	u-W		13.1 GeV	[1]
	d-Z		6.07 GeV	[1]
	d-p		0.673 GeV	[1]
Excited Up Quark (u*)	a-l-l	3 TeV	102.1 GeV	[1]
	u-g		82.4 GeV	[1]
	d-W		13.01 GeV	[1]
	u-Z		4.05 GeV	[1]
	u-p		2.69 GeV	[1]
Excited Quark (q*)	j-p	1 TeV	50-300 GeV	[1]
Excited Quark (q*)	j-p	3 TeV	~200 GeV	[1]
LeptoQuark (LQ)	e-j (50%)	~1TeV	27 GeV	[1]
	e-e-j (25%)			[1]
	e-e-j-j (25%)			[1]
W' (charged boson)	WZ -> (l-v)-(l-l)	<2.5 TeV	---	[1]
Z' (neutral boson)	WW -> (j-j)-(e-v)	~2 TeV	---	[1]
more...				

Table 1: A short, incomplete summary of searches to be performed at the LHC when operation begins. References are to the papers describing the searches.

(a)

Particle Amounts by Type			
Particle Type	Electron	Jet	Muon
Number of Particles in Event	3	2	2

(b)

Particle-Type Groups of up to 3 Particle Decays	
Particle-Type Groups	**Unique Particle Groups**
Electron-Electron	e1-e2 ‖ e1-e3 ‖ e2-e3
Electron-Jet	e1-j1 ‖ e1-j2 ‖ e2-j1 ‖ e2-j2 ‖ e3-j1 ‖ e3-j2
Electron-Muon	e1-m1 ‖ e1-m2 ‖ e2-m1 e2-m2 ‖ e3-m1 ‖ e3-m2
Jet-Jet	j1-j2
Jet-Muon	j1-m1 ‖ j1-m2 ‖ j2-m1 ‖ j2-m2
Muon-Muon	m1-m2
Electron-Electron-Electron	e1-e2-e3
Electron-Electron-Jet	e1-e2-j1 ‖ e1-e2-j2 ‖ e1-e3-j1 e1-e3-j2 ‖ e2-e3-j1 ‖ e2-e3-j2
Electron-Electron-Muon	e1-e2-m1 ‖ e1-e2-m2 ‖ e1-e3-m1 e1-e3-m2 ‖ e2-e3-m1 ‖ e2-e3-m2
Electron-Jet-Jet	e1-j1-j2 ‖ e2-j1-j2 ‖ e3-j1-j2
Electron-Jet-Muon	e1-j1-m1 ‖ e1-j1-m2 ‖ e1-j2-m1 ‖ e1-j2-m2 e2-j1-m1 ‖ e2-j1-m2 ‖ e2-j2-m1 ‖ e2-j2-m2 e3-j1-m1 ‖ e3-j1-m2 ‖ e3-j2-m1 ‖ e3-j2-m2
Electron-Muon-Muon	e1-m1-m2 ‖ e2-m1-m2 ‖ e3-m1-m2
Jet-Jet-Jet	---
Jet-Jet-Muon	j1-j2-m1 ‖ j1-j2-m2
Jet-Muon-Muon	j1-m1-m2 ‖ j2-m1-m2
Muon-Muon-Muon	---

Table 2: (a) Particles found in the event. (b) Example of particle-type groups and the unique particle groups created within them for up to three particle decays.