



Université catholique de Louvain

**Recherche d'un Higgs invisible produit par
interaction e^+e^- à des énergies centre de
masse allant jusqu'à 209 GeV dans
l'expérience *ALEPH***

Christophe Delaere (*delaere@fynu.ucl.ac.be*)

Mémoire présenté en juin 2001 en vue de l'obtention
du diplôme de licence en sciences physiques.

Promoteur : Vincent Lemaitre.

Remerciements

Je ne voudrais manquer l'opportunité qui m'est ici offerte de remercier tous ceux qui, par leur présence, leurs conseils ou leur soutien m'ont aidé à mener à bien cette merveilleuse aventure.

Je pense tout particulièrement à Vincent Lemaitre, mon promoteur, toujours présent et disponible ainsi qu'à Olivier van der Aa, qui m'a beaucoup aidé par les nombreuses discussions que nous avons pu avoir.

Merci également à Guenther Dissertori et Rick Cavanaugh, du projet Alpha++.

Cette année n'aurait pas été non plus la même sans toute la fine équipe FYNU-CRC ; qu'ils en soient tous remerciés, même si je ne peux tous les citer ici.

Enfin, comment pourrais-je ne pas remercier mes parents qui, depuis le début de mes études, ont su me pousser, me soutenir, m'écouter, me supporter parfois ? Ils ont su me communiquer la passion des sciences, et je leur en serai toujours reconnaissant. Merci !

Christophe Delaere.

La vocation, c'est le bonheur d'avoir pour métier sa passion.
Stendhal.

Préface

Alors que la communauté scientifique vient de célébrer la clôture officielle du LEP, le grand collisionneur d'électrons et positrons du CERN, l'activité scientifique continue au sein des 4 expériences. La quantité de données accumulées pendant 10 ans représente une mine de données à analyser, et il faudra encore plusieurs années avant de pouvoir considérer en avoir retiré tout le potentiel.

La possibilité nouvelle d'analyser les données de l'expérience *ALEPH* à Louvain-La-Neuve, ainsi que la complémentarité des projets de développement au niveau du logiciel avec les activités déjà menées au sein du groupe "Hautes Energies" de l'unité de physique nucléaire est à l'origine de ce travail.

Ce mémoire de fin d'études, présenté en vue de l'obtention du diplôme de licencié en sciences physiques se compose de deux parties. La première rend compte du travail effectué dans le cadre du développement et de l'évaluation du nouvel environnement d'analyse orienté objet développé au sein de l'expérience *ALEPH*. La seconde partie expose quant à elle la recherche d'un Higgs invisible produit par interaction e^+e^- à des énergies centre de masse allant jusqu'à 209 GeV , mettant à profit les outils élaborés dans la première partie.

Table des matières

1	L'expérience <i>ALEPH</i>	1
1.1	Le détecteur	1
1.2	Le système de déclenchement	3
1.3	Reconstruction des événements	4
2	Développement d'<i>ALPHA++</i>	7
2.1	Introduction	7
2.2	Reconstruction de Jets	9
2.2.1	Principes fondamentaux	9
2.2.2	Réalisation concrète	12
2.3	Sélection des leptons	13
2.3.1	Principes fondamentaux	13
2.3.2	Réalisation concrète	16
2.4	Accès aux données générales	19
2.5	Autres développements	21
2.6	Comparaison <i>ALPHA - ALPHA++</i>	22
3	Recherche du Higgs invisible	25
3.1	Introduction	25
3.2	Le mécanisme de Higgs	26
3.3	Production du boson de Higgs au LEP	28
3.4	Le mode de désintégration invisible	30
3.5	Définitions et notations	31
3.6	Signal et bruits de fond	31
3.7	Données et échantillons Monte Carlo	35
3.8	Sélection hadronique à 189 <i>GeV</i>	40
3.8.1	Optimisation des coupures	40
3.8.2	Construction du réseau de neurones	45
3.8.3	Erreurs systématiques et biais	51
3.8.4	Résultats	54
3.8.5	Etude des topologies sélectionnées	54

3.9	Analyse à 206 <i>GeV</i>	59
3.10	Combinaison des deux analyses	70
A	Le driver <i>ALPHA++</i>	77
B	Le mécanisme de brisure de symétrie	79
C	Code source singleW commenté	81
D	Code source acoplanar	89

Chapitre 1

L'expérience *ALEPH*

1.1 Le détecteur

Le but de ce chapitre n'est pas de décrire de façon complète et détaillée le détecteur *ALEPH*, une telle description du détecteur peut être trouvée dans les références [1], [2] et [3]. Certains éléments sont cependant nécessaires à une bonne compréhension de la suite.

L'anneau du LEP a un diamètre moyen de $8486m$. C'est au point d'interaction '4' que l'anneau traverse l'expérience *ALEPH*. Construit pour mesurer les événements dans tout l'angle solide à un taux inférieur à 1Hz, le détecteur est structuré en couches cylindriques concentriques autour du tube à faisceau. Un champ magnétique de $1.5T$ engendré par un aimant supraconducteur courbe les trajectoires des particules chargées. On trouve successivement, en se rapprochant du centre (voir figure 1.1), différents sous-détecteurs.

deux épaisseurs de chambres à muons (MCAL) détectent les muons qui ont traversé le fer.

un calorimètre hadronique (HCAL) constitue le support de fer doux, pesant à lui seul 2580 tonnes. Il est complètement équipé de chambres à streamers. Le calorimètre hadronique permet une bonne identification des π . La résolution en énergie pour des pions d'incidence normale est de $\Delta E/E = 0.85/\sqrt{(E(GeV))}$

un calorimètre électromagnétique (ECAL) permettant la mesure des particules chargées avec une grande précision angulaire. Ce calorimètre est constitué de 3 fois 12 modules, situés respectivement dans le "tonneau" et dans les deux "bouchons", de façon à couvrir 92% de l'angle solide. Chaque module contient 45 feuilles de plomb séparées par des chambres proportionnelles. La grande granularité permet une détermination du centroïde avec une résolution angulaire donnée par $\Delta\theta(mrad)/\sin\theta = 0.32 + 2.7/\sqrt{(E(GeV))}$ et une bonne

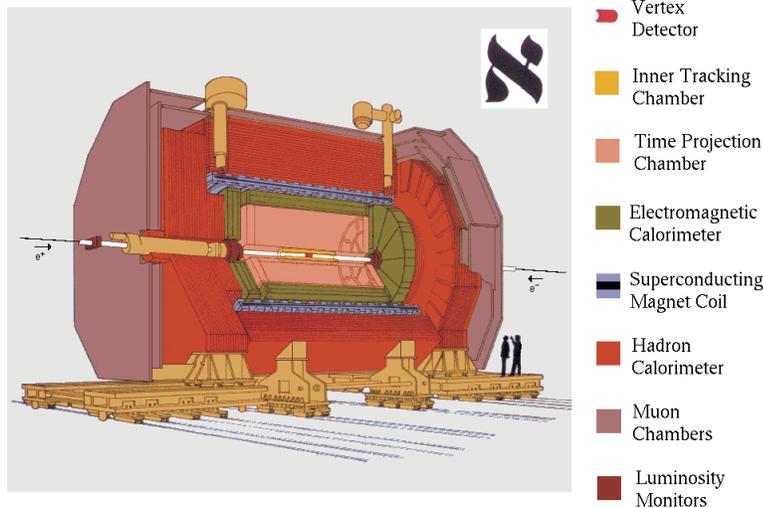


FIG. 1.1 – Le détecteur ALEPH.

identification des électrons et photons, ce également dans les régions à forte activité. La résolution en énergie est $\Delta E/E = 0.017 + 0.19/\sqrt{E(GeV)}$

une chambre proportionnelle (TPC) fournissant un ensemble de mesures tridimensionnelles des traces allant jusqu'à 21 points et reconstruisant l'impulsion avec un précision $\Delta p/p^2 = 1.2 \cdot 10^{-3} (GeV/c)^{-1}$ à $45 GeV$. Elle fournit jusqu'à 360 mesures de perte d'énergie, ce qui facilite l'identification des particules à basse énergie. Chaque extrémité de la TPC est à la masse alors qu'il y a au centre une membrane de mylar recouvert de graphite, maintenue à $-26 kV$.

une chambre à fils axiale (ITC) fournissant 8 coordonnées avec une précision de $100 \mu m$ dans le plan perpendiculaire au champ magnétique, ainsi qu'un signal de déclenchement. La chambre fournit l'unique information de tracking au premier niveau du système de déclenchement.

un détecteur à silicium (VDET) fournissant pour chaque trace une paire de points avec une précision de $10 \mu m$ permettant d'interpoler la position d'éventuels vertex secondaires.

A cela viennent s'ajouter les calorimètres de luminosité (LCAL et SICAL), utilisés pour mesurer avec précision la luminosité pendant la prise de données.

Le système de coordonnées utilisé tant pour la localisation des différentes parties du détecteur que pour le traitement et l'analyse des objets détectés est présenté figure 1.2.[4] Il trouve son origine au point théorique d'interaction, milieu de la section droite entre les deux quadrupôles les plus proches. L'axe z est dirigé dans la direction nominale des électrons, et pointe donc vers Bellegarde. Il fait un angle

de $+3.5875 \text{ mrad}$ avec l'horizon. L'axe x pointe vers le centre du LEP, tout en restant horizontal. L'axe y pointe quant à lui vers le haut, en étant orthogonal aux précédents.

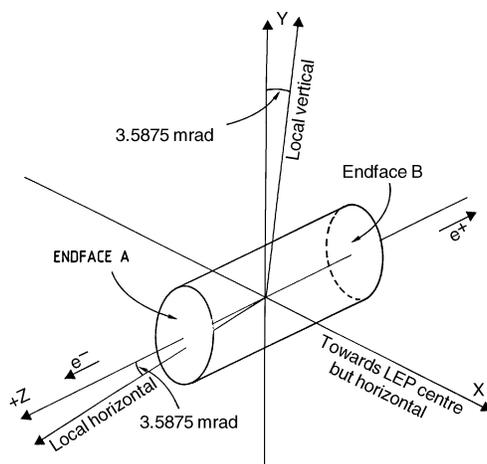


FIG. 1.2 – Le Système de référence d'ALEPH (ARS)

Les coordonnées sphériques (r, θ, ϕ) sont définies comme suit

$$x = r \sin \theta \cos \phi,$$

$$y = r \sin \theta \sin \phi,$$

$$z = r \cos \theta.$$

1.2 Le système de déclenchement

Le système de déclenchement est organisé en trois niveaux. Le niveau 1 décide s'il faut ou non lire la totalité des canaux. Le niveau 2 confirme la décision du niveau 1 en remplaçant les données de l'ITC par les données plus précises de la TPC, disponibles $50 \mu s$ après l'interaction. Un niveau logiciel est alors utilisé pour rejeter les bruits de fond tels que les interactions des particules avec le gaz résiduel dans le tube à faisceau, la chambre à vide ou les collimateurs. Les principaux signaux de déclenchement sont¹

- plus de 5.5 GeV dans le tonneau ECAL, 4.5 GeV dans chacun des bouchons ou 1.6 GeV en coïncidence dans les deux bouchons.

¹Les coupures dépendent légèrement de l'énergie. Les valeurs présentées ici sont celles utilisées en novembre 1996 à 172 GeV .

TAB. 1.1 – *ALEPH* en quelques chiffres

Circonférence du LEP	26.66 km
Energie maximale du faisceau	105 MeV
Temps entre deux collisions	$22\mu s$
Luminosité	$10^{32} cm^{-2} s^{-1}$
Rayon du tonneau au HCAL	468 cm
Champ magnétique à 5000 A	1.5T
Volume de gaz de la TPC	$43 m^3$
Nombre de fils dans l'ITC	960
Nombre de pistes du VDET	1920 pistes/module, 48 modules
Précision de la mesure de luminosité	$\sim 0.2\%$
Taux maximal à la sortie du système de déclenchement	1 Hz

– plus de $1.2 GeV$ dans un module ECAL, en coïncidence avec une trace dans l'ITC.

– une particule pénétrant le HCAL en coïncidence avec une trace dans l'ITC.

Un certain nombre de masques additionnels permettent d'atteindre une efficacité proche de 1.0 pour la toute grande majorité des topologies susceptibles d'être étudiées. Ces efficacités sont de plus connues à mieux d'1%.

1.3 Reconstruction des événements

Les principaux aspects de la reconstruction des événements à *ALEPH* sont la reconstruction des traces dans la TPC et l'ITC, la recherche de dépôts groupés dans les calorimètres et l'association de ces deux données pour permettre la mesure de l'énergie et l'identification des particules neutres et chargées.

La reconstruction des traces se fait en regroupant des points compatibles avec une hélice prenant son origine à l'extérieur de la TPC et se propageant vers l'intérieur. Quatre points sont requis pour définir une trace. Les erreurs de diffusion sont alors prises en compte avant de rechercher des signaux associés dans l'ITC. Les données provenant des deux sous-détecteurs sont prises en compte pour ajuster les paramètres de l'hélice ainsi obtenue.

La reconstruction de l'énergie dans les calorimètres se base sur la reconnaissance de "clusters" topologiques, composés de dépôts d'énergie adjacents. Un tel groupe sera validé si il contient un signal excédant $90 MeV$. Les signaux inférieurs à $30 MeV$ sont simplement ignorés. Les traces sont extrapolées jusqu'aux calorimètres et associées à tout cluster rencontré, définissant ainsi un dépôt chargé. On

appelle alors dépôts neutres les clusters restants.

La grande granularité des calorimètres permet en outre l'identification des particules. L'identification des électrons repose sur la mesure du dépôt d'énergie dans le calorimètre ECAL et de la perte d'énergie dans la TPC. L'identification dans le calorimètre est basée sur deux variables décrivant la topologie (longitudinale et transversale) du dépôt. Ces variables permettent d'atteindre une efficacité de 98%. La mesure de perte d'énergie dans la TPC est basée sur la combinaison des mesures individuelles le long de la trajectoire, tronquées à 60%. La comparaison de cette grandeur avec la valeur attendue améliore l'efficacité et la réjection des pions entre 2 et 20 GeV . Les muons sont quant à eux caractérisés par une trace traversant l'entièreté du détecteur avant d'atteindre les chambres à muons. Toute trace de plus d'un GeV est extrapolée jusqu'à l'extérieur du détecteur, et un cône trois fois plus grand que l'incertitude sur le déplacement moyen dû aux diffusions multiples est défini à la sortie du calorimètre hadronique. Une trace est considérée comme étant un muon si plus de 9 plans ont été déclenchés, dont au minimum 4 parmi les 10 derniers. Selon les analyses, des coupures plus ou moins strictes sont cependant appliquées. Typiquement, pour une analyse pour laquelle il est important d'avoir une grande efficacité, on atteint ainsi 99% d'efficacité, avec une pollution par des pions de 1.5%.

Parmi l'ensemble d'objets reconstruits, une attention toute particulière doit être apportée à ce que l'on appelle "flux d'énergie" (Energy flow, souvent noté Eflw). Le flux d'énergie est un objet dont toutes les composantes sont obtenues en combinant de façon optimale les données fournies par les différents composants du détecteur. Pour un événement à haute multiplicité, 55% de l'énergie est emportée par les particules chargées. En se limitant à l'information contenue dans les traces, on perd ainsi près de la moitié de l'énergie de l'événement, d'où l'importance accrue d'une telle entité. Pour chaque événement, l'amplitude et la direction du flux d'énergie sont reconstruites en 2 grandes étapes.

Dans un premier temps, appelé "cleaning", les canaux du calorimètre avec du bruit électronique sont retirés. La détection de ces canaux se base sur des critères d'auto-cohérence ainsi que sur la sélection des canaux systématiquement actifs sur plusieurs événements successifs. Cette phase permet ainsi de réduire l'énergie totale maximale dans des événements sélectionnés au hasard (pour lesquels aucun signal ne devrait être présent) de 30 MeV à 3.5 GeV .

La seconde phase consiste à associer l'information des traces et des dépôts calorimétriques.

- Les bonnes traces chargées sont comptées comme de l'"énergie chargée".
- Les particules neutres à long temps de vie se désintégrant en deux particules chargées sont conservées si elles pointent vers le vertex avec la même tolérance que pour les particules chargées.
- Les électrons et les muons sont comptés comme de l'"énergie chargée". Les

photons sont comptés comme de l'“énergie neutre”. A ce niveau, la calibration du calorimètre pour ces différentes particules est utilisée. Un algorithme de “brem-recovery” est utilisé pour tenir compte de l'émission éventuelle d'un photon par un électron.

- Les objets hadroniques restants sont alors également considérés comme de l'“énergie neutre”.

La résolution en énergie du détecteur, sans cet algorithme, serait de $\sigma(E) = 1.2\sqrt{E(\text{GeV})}$. Cette démarche permet d'obtenir une valeur de $\sigma(E) = (0.59 \pm 0.03)\sqrt{E(\text{GeV})} + (0.6 \pm 0.3)$. Une vision qualitative des performances de cet algorithme est obtenue par la figure 1.3, obtenue avec les données acquises au pic du Z. On y retrouve la distribution de la masse invariante totale en fonction du nombre de flux d'énergie pour les événements comptant plus de deux traces.

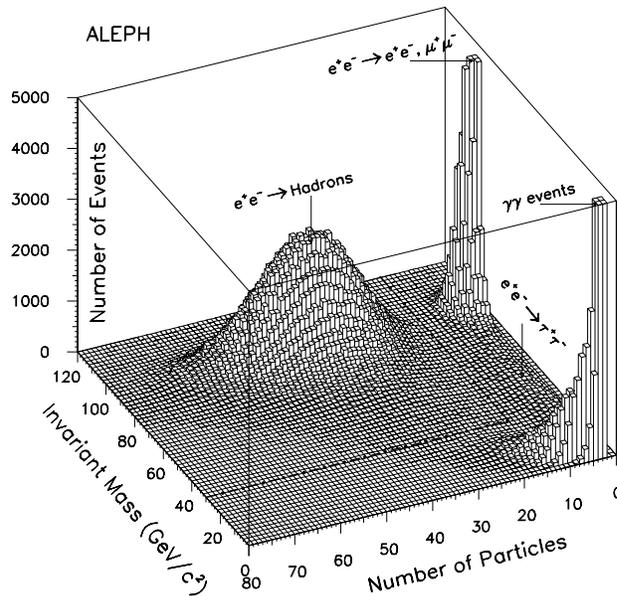


FIG. 1.3 – Masse invariante en fonction du nombre de particules (nombre de flux d'énergie) pour tous les événements comportant au moins 2 traces chargées, pour une partie des données de 1992. Les pics à basse multiplicité ont été tronqués.

Chapitre 2

Développement de l'environnement d'analyse *ALPHA++*

2.1 Introduction

Si la conception et le design d'une expérience sont de première importance et conditionnent fortement tant la faisabilité des objectifs que la qualité des résultats, il en est de même pour l'analyse des données produites. La tendance actuelle à une explosion de la quantité des données conduit de plus l'expérimentateur à développer des environnements d'analyse permettant un travail à la fois rapide, efficace et précis. Du soin apporté à la réalisation de cet environnement dépendra aussi la qualité des résultats scientifiques.

L'environnement d'analyse développé pour l'expérience *ALEPH*, encore appelé *ALPHA*¹, a pour objectif de simplifier la conception de programmes d'analyse en Fortran. *ALPHA* fournit un accès aisé aux variables physiques (telles que l'impulsion, l'énergie, ...) afin que le physicien ne doive pas se préoccuper de la structure des données *ALEPH*, à savoir une base de données tabulaire de type BOS[6]. De plus, un ensemble étendu de routines d'analyse est mis à disposition du programmeur. Ces routines donnent par exemple accès à l'aspect cinématique, à la topologie de l'événement, permettent d'identifier les diverses particules ou de localiser les vertex secondaires, etc.

L'évolution des langages de programmation, des processeurs et des compilateurs nous pousse à nous tourner vers un environnement d'analyse orienté objet. C'est l'objectif majeur du projet *ALPHA++*. Il est en effet nécessaire de préserver les données recueillies par les diverses expériences menées auprès du LEP. Leur disparition faute de compilateur Fortran adapté serait pour le moins regrettable. De plus, ce projet constitue également un banc d'essai pour les nouveaux outils d'analyse

¹*ALEPH* Physics Analysis Package

qui sont et seront utilisés au LHC.

Comme expliqué dans le cahier des charges d'ALPHA++ [7], les principaux objectifs de ce projet sont :

1. Convertir les données *ALPHA* (préférentiellement des données LEP2) du style "base BOS" en objets persistants et les écrire dans une base de données Objectivity.
2. Ecrire une version de l'environnement d'analyse d'*ALEPH* dans un langage orienté objet (C++) basé sur une base de donnée Objectivity.
3. Tester le logiciel développé par la collaboration *RD45* et le projet de développement *ANAPHE/LHC++*, qui évalue les options de stockage de données et d'analyse pour les expériences au LHC.
4. Fournir de l'expérience pour le projet d'archivage des données du LEP.
5. Donner l'opportunité aux physiciens d'étudier la programmation le design et l'analyse "orienté objet".

Dans cet environnement, on identifie un driver et une partie d'analyse. La première partie contient les routines d'initialisation, d'exécution et de terminaison, et assure l'accès aux données stockées dans la base de données Objectivity ou dans l'ancienne base de données EPIO. C'est également à ce niveau que les fichiers de configuration sont accédés. La seconde partie exploite la structure en classe du C++ pour mettre en œuvre les deux principaux objets d'analyse : les objets vectoriels et les "vertex", reflétant ainsi la structure d'analyse déjà familière aux utilisateurs d'*ALPHA*. Tous les objets concrets (photons, traces de particules chargées, jets, ...) sont représentés par des classes dérivant de ces deux entités. Une classe nommée "AlphaBank" sert de container pour tous les objets ; l'utilisateur manipule des pointeurs vers l'AlphaBank, ce qui rend l'accès aux objets très aisé. Finalement, les algorithmes sont des méthodes de la classe "AlephCollection" qui dérive du vecteur standard STL².

Dans le cadre de ce mémoire, nous avons pris part au développement d'un ensemble de classes dans la branche "traces" du projet, et d'un certain nombre d'algorithmes de sélection. Nous avons ainsi implémenté :

- La classe muon et l'algorithme de sélection des muons.
- La classe électron, et l'algorithme de sélection des électrons.
- La classe tau et l'algorithme de production et sélection des taus.
- Les algorithmes standards de reconstruction des jets.
- La simulation et la gestion affinée du trigger.
- Le contrôle de la stabilité de la luminosité.
- Plusieurs routines topologiques (sphéricité, planarité, etc.).

²Standard Template Library

La version finale simplifiée du diagramme de classes, basé sur le standard UML, est présenté aux figures 2.1 et 2.2. Elle a été réalisée par “reverse ingenering” à l’aide du logiciel ROSE[8]. Sur ces figures, les classes grisées sont les classes nouvellement implémentées. L’application représente un volume trop important pour être joint à ce travail. L’entièreté du code peut être obtenue sur CVS à l’adresse suivante :

`/afs/cern.ch/aleph/project/alpha++` .

En annexe A, on trouvera également le diagramme correspondant pour le driver.

Pour tester la fiabilité du nouvel environnement, une comparaison détaillée des versions *ALPHA* et *ALPHA++* de l’analyse de sélection hadronique d’un *W* isolé [9] a été menée. La routine de sélection d’un *W* isolé a été développée en traduisant la version existant en Fortran vers l’environnement *ALPHA++*. Cette sélection nécessite l’identification d’électrons et de taus, et fait usage des algorithmes de jets. En plus de l’identification des particules, le programme génère un ntuple PAW contenant diverses variables discriminantes telles que l’énergie des leptons, l’impulsion manquante ou l’impulsion des jets.

La comparaison directe, événement par événement des ntuples produits par les deux versions du programme montre que le résultat peut être considéré comme parfaitement équivalent, la seule différence ($\sim 10^{-7}$) pouvant être attribuée à la représentation machine des nombres flottants.

2.2 Reconstruction de Jets

2.2.1 Principes fondamentaux

Un événement typique tel qu’en voit le détecteur *ALEPH* est relativement complexe, comportant un grand nombre de particules distribuées sur l’entièreté de l’angle solide couvert. Ces particules sont soit isolées, soit groupées en faisceaux, qu’il convient d’appeler jets. Ces jets de particules trouvent leur origine dans l’hadronisation des quarks produits lors de l’interaction e^+e^- , ou traduisent la présence d’un gluon isolé. Bien souvent, on cherche à reconstruire le parton original de ce jet de particules, et tout particulièrement, à établir son impulsion ainsi que sa masse invariante. On s’intéresse aussi parfois à l’identification de la paire $q\bar{q}$ à l’origine des jets observés (on parle ainsi de b-tagging).

La tâche principale du programme d’analyse est donc d’isoler le jet, et de reconstruire le quadrivecteur impulsion global. L’approche adoptée est la suivante. Lorsque deux traces, deux dépôts calorimétriques, ou de façon plus générale deux flux d’énergie sont proches l’un de l’autre, leurs quadrivecteurs sont combinés de façon à obtenir le quadrivecteur d’un prototype de jet. Cette procédure est itérée jusqu’à ce que la distance minimale entre deux objets soit supérieure à une limite donnée. Pour ce faire, il convient de définir

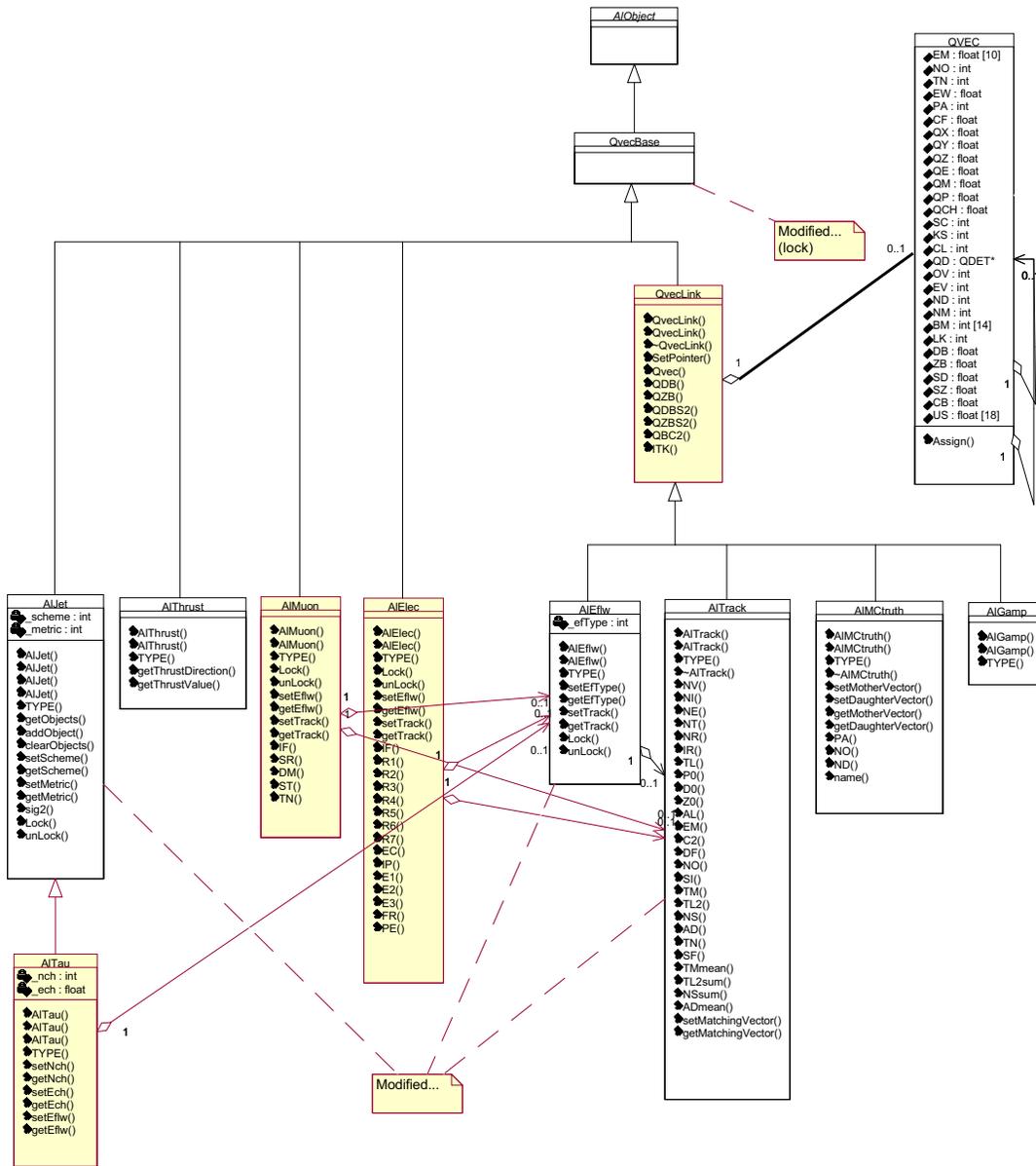


FIG. 2.1 – Diagramme UML simplifié de la nouvelle version d'ALPHA++. Partie contenant les différents objets vectoriels. Les objets dérivent de QvecBase, commun à tous les objets vectoriels, via Qveclink si un lien existe avec la base de données (pour les objets élémentaires contenus dans celle-ci). Les classes grisées sont les classes nouvellement implémentées.

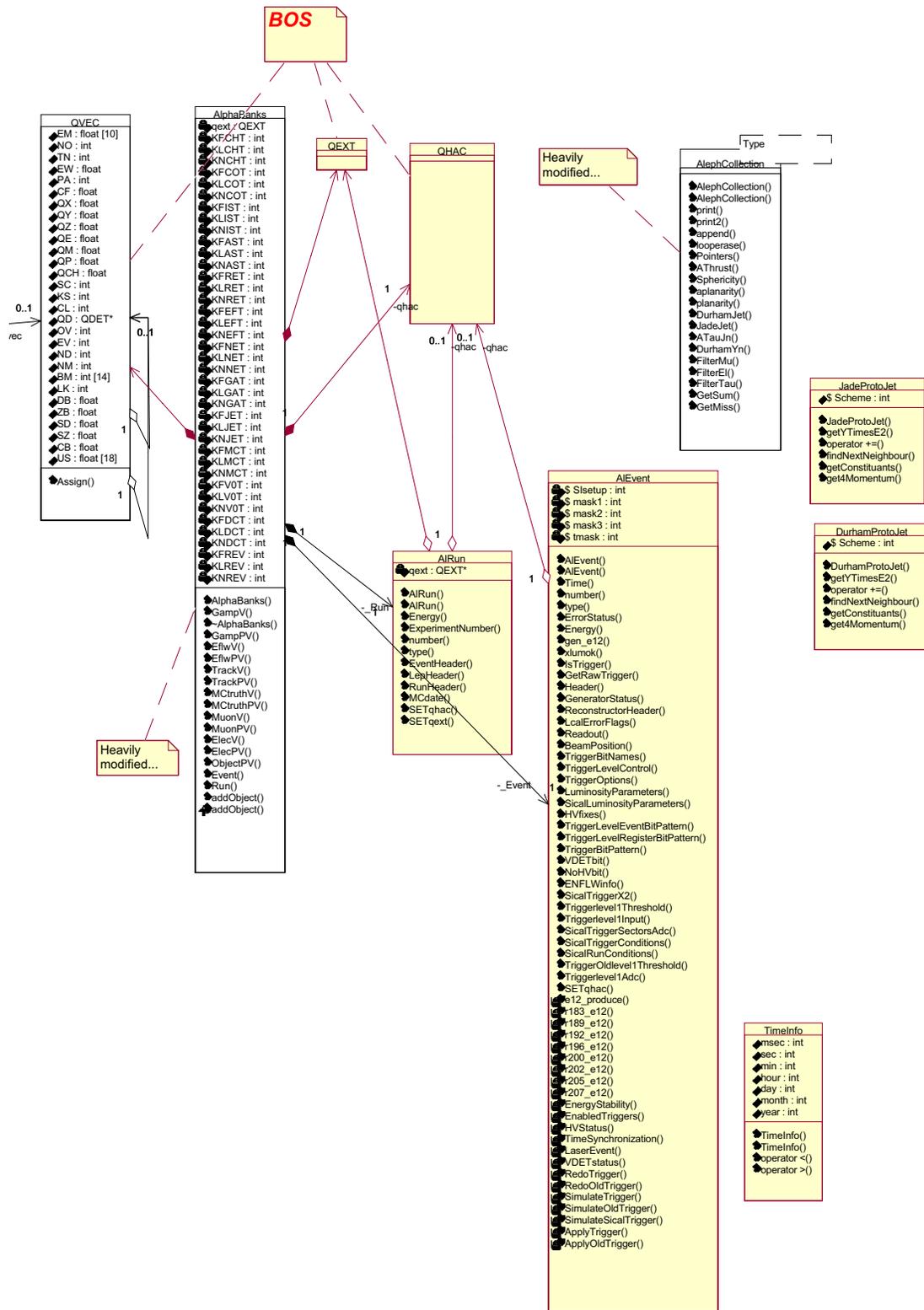


FIG. 2.2 – Diagramme UML simplifié de la nouvelle version d'ALPHA++. Partie contenant la copie de la base de données (AlphaBank), l'information sur le run et l'événement ainsi que les algorithmes (AlephCollection).

le **schéma de recombinaison** , c'est à dire la règle choisie pour l'addition de deux quadrivecteurs.

la **métrie** , c'est à dire la façon de mesurer la distance entre deux objets. C'est ce critère qui permettra de décider si deux traces sont suffisamment proches pour être groupées.

Dans le cadre de ce mémoire, les métriques Jade et Durham ont été implémentées, alors que deux schémas de recombinaison (appelés E et E0) sont disponibles.

La métrie la plus simple est la métrie de Jade. Elle est définie par

$$y = \frac{2E_1 E_2 \left(1 - \frac{\vec{p}_1 \cdot \vec{p}_2}{|\vec{p}_1| |\vec{p}_2|}\right)}{E_{vis}^2}. \quad (2.1)$$

C'est en fait la masse invariante, en supposant que les particules filles ont une masse nulle, divisée par l'énergie totale visible de l'événement, le tout élevé au carré.

Afin d'améliorer la reconstruction des gluons mous émis à grands angles, que l'on veut incorporer au jet principal, on définit la métrie de Durham.

$$y = \frac{2 (\min(E_1, E_2))^2 \left(1 - \frac{\vec{p}_1 \cdot \vec{p}_2}{|\vec{p}_1| |\vec{p}_2|}\right)}{E_{vis}^2}. \quad (2.2)$$

Comme on le voit, le produit des énergies est remplacé par le minimum des énergies au carré. Si une des particules est très peu énergétique, y sera très petit, ce qui favorisera l'incorporation de la trace associée au jet.

Comme nous l'avons déjà dit, il faut aussi préciser la façon d'additionner deux quadrivecteurs. Dans le schéma de recombinaison ' E ', on additionne simplement les deux quadrivecteurs, composante par composante. Dans le schéma ' E_0 ', par contre, le trivecteur impulsion sommé est renormalisé de façon à obtenir une masse invariante totale nulle. Le choix de l'algorithme visera à améliorer, dans un cas donné, la correspondance entre les hadrons du jet et le parton original.

2.2.2 Réalisation concrète

Nous avons réalisé le design des classes traitant les jets de façon similaire à ce qui est fait au sein de l'environnement ORCA de la collaboration CMS. Les algorithmes se basent sur deux classes internes (DurhamProtoJet et JadeProtoJet). C'est dans ces classes que sont définis la métrie et le schéma de recombinaison (ce dernier prend la forme naturelle d'un opérateur d'addition $+=$). Trois algorithmes sont mis à disposition de l'utilisateur.

1. **DurhamJets** produit des jets avec la métrie de Durham
2. **JadeJets** produit des jets avec la métrie de Jade

3. **DurhamYn** produit une liste des valeurs critiques Y_{cut} pour lesquelles la topologie de l'événement passe de n à $n + 1$ jets, en utilisant la métrique de Durham.

Nous avons ici un bel exemple de la simplicité de la syntaxe que permet d'atteindre le C++, en respectant l'intuition et en agissant sur des objets physiques. Soit *bank* une référence à l'*AlphaBank* courante. Nous écrivons :

```
AlephCollection<Eflws*> eflws = bank.EflwsPV();
AlephCollection<AlJet> J1 = eflws.DurhamJets(ycut,scheme,Evis);
AlephCollection<AlJet> J2 = eflws.JadeJets(ycut,scheme,Evis);
vector<float> Yn = eflws.DurhamYn(scheme, Evis);
```

Après avoir obtenu un vecteur de pointeurs vers des objets de type flux d'énergie ("Energy flow"), ces quelques lignes produisent deux vecteurs de jets (J1 et J2) respectivement avec les métriques Durham et Jade, ainsi qu'un vecteur Yn des valeurs critiques pour lesquelles la topologie de l'événement passe de n à $n + 1$ jets. Il est important de noter que les algorithmes de production des jets peuvent agir sur tout type d'objet de la branche *traces*, que ce soit des flux d'énergie comme ici, des traces, ou encore des dépôts d'énergie dans un calorimètre. Les paramètres sont :

ycut	valeur limite pour la distance entre 2 traces au sein d'un jet
scheme	schéma de recombinaison : 0 pour E0, 1 pour E
Evis	énergie totale. 0 \Rightarrow calcul du total sur le vecteur.

Par défaut, les jets sont reconstruits suivant le schéma de recombinaison E, en calculant l'énergie visible sur le vecteur d'objet fourni.

2.3 Sélection des leptons

2.3.1 Principes fondamentaux

Le détecteur a été conçu pour détecter et identifier les électrons, muons et taus. La reconnaissance de chacune de ces trois entités repose sur la grande spécificité des signaux laissés par chacune d'elles.

La détection des muons se base sur la recherche d'un signal dans les chambres à muons ainsi que sur le tracking effectué par le calorimètre hadronique (en utilisant à cette fin l'information retournée par les 23 plans de streamers qui le constituent). Toute trace suffisamment énergétique est extrapolée en faisant l'hypothèse que c'est un muon à travers le calorimètre hadronique, en utilisant une carte précise du champ magnétique. Un cône est alors "ouvert" autour de cette trace. L'estimation se fait sur base du nombre de plans HCAL actifs le long de la trace, ainsi que sur base de la multiplicité des signaux dans HCAL. Trois quantités sont définies pour

cela : N_{fir} est le nombre de plans actifs le long de la trajectoire, N_{exp} est le nombre de plans attendus le long de la trajectoire et N_{10} est le nombre de plans actifs parmi les 10 derniers susceptibles de l'être. Ces quantités sont reprises à la figure 2.3.

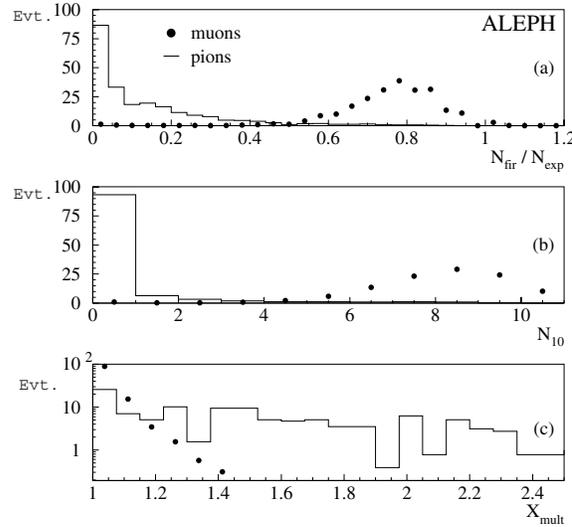


FIG. 2.3 – Quantités intervenant dans l'identification des muons.

Un candidat muon est validé à trois conditions :

- La trace associée doit répondre à certains critères de qualité : une impulsion suffisante, un nombre conséquent de hits dans la TPC, une origine proche du point d'interaction théorique et une direction suffisamment éloignée du faisceau.
- Cette trace doit avoir $N_{fir} / N_{exp} > 0.4$, $N_{10} > 4$ et avoir une multiplicité de signaux dans HCAL inférieure à 1.4. Elle est alors dite "pénétrante".
- On impose un signal dans une chambre à muons en accord avec cette trace pénétrante.

L'identification des électrons est un peu plus complexe. Pour identifier un électron, on peut se baser sur la topologie du dépôt d'énergie dans le calorimètre, ainsi que sur la perte d'énergie par unité de longueur mesurée par la TPC. Plus précise, la première de ces deux mesures sera privilégiée [10], sauf si l'électron aboutit dans une région insensible ou saturée du calorimètre électromagnétique. Différents estimateurs sont construits lors de la reconstruction des événements. Ils expriment, en unités d'écart type, la différence entre une grandeur et la valeur attendue pour un électron. Nous avons ainsi :

- **R1** (balance énergétique) = $\frac{E-p}{\sigma_1}$
- **R2** (profil transverse) = $\frac{E_A/P - \langle E_A/P \rangle}{\sigma_2}$

- **R3** (profil longitudinal) : $\frac{a-\langle a \rangle}{\sigma_3}$
- **R4** (profil longitudinal) : $\frac{b-\langle b \rangle}{\sigma_4}$
- **R5** (ionisation spécifique) : $\frac{I-\langle I \rangle}{\sigma_5}$

où E et p sont respectivement l'énergie et l'impulsion mesurés, où E_4 est l'énergie déposée dans les 4 tours calorimétriques les plus proches de la trace, et où σ_i est l'écart type attendu de la variable i . Les paramètres a et b sont les paramètres qui reproduisent au mieux le dépôt d'énergie par unité de profondeur, exprimé en longueurs de radiations :

$$\frac{dE}{dS} = S^{1/b-1} e^{-a/bS} .$$

La figure 2.4 représente la distribution pour R2 ($R_{transverse}$), R3 ($R_{longitudinal}$) et R5 (R_I).

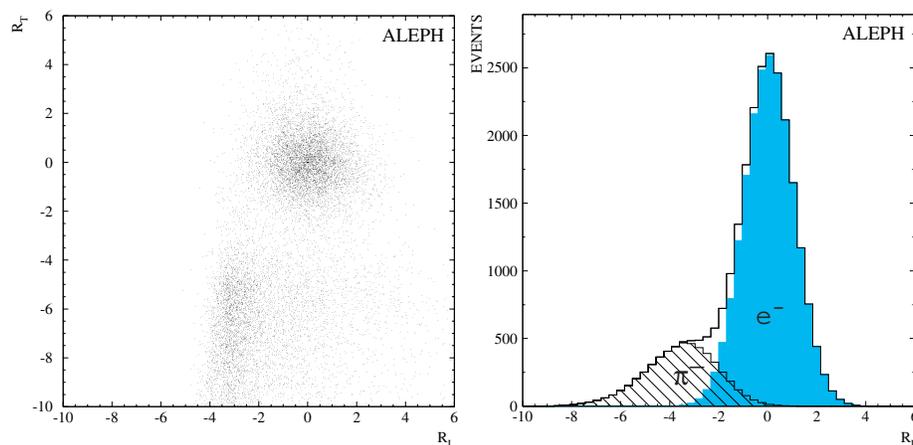


FIG. 2.4 – Quantités intervenant dans l'identification des électrons. La figure de gauche reprend $R_{transverse}$ et $R_{longitudinal}$. On y voit les électrons (proches de (0,0)) et la contamination de pions. La figure de droite reprend la distribution de R_I après des coupures strictes sur $R_{transverse}$ et $R_{longitudinal}$. On y voit la contamination de π restant.

Une coupure forte sur ces quantités ($-1.5 < R_2$, $-2 < R_3 < 3$ et $-3 < R_5$) permet d'obtenir une grande pureté (99%) pour une efficacité de 60%. Un électron isolé est sélectionné en exigeant $R_2 > -3$. On a ainsi une efficacité de 99%.

Un candidat électron sera validé à trois conditions :

- La trace doit répondre aux caractéristiques de perte d'énergie (dans le calorimètre ou la TPC) d'un électron.
- La trace associée doit répondre à certains critères de qualité : une impulsion suffisante, un nombre conséquent de hits dans la TPC, une origine proche

du point d'interaction théorique et une direction suffisamment éloignée du faisceau.

- La trace ne doit pas déjà être identifiée comme un muon.

C'est l'identification des taus qui est la plus difficile et la plus sujette à erreur. Le tau se désintègre en effet avant de pouvoir être détecté, de sorte que ce sont les produits de désintégration qui sont étudiés. Les principaux modes de désintégrations sont[11] :

- $\mu^\mp \bar{\nu}_\mu \nu_\tau (\gamma)$ (17.37%)
- $e^\mp \bar{\nu}_e \nu_\tau (\gamma)$ (19.56%)
- $h^\pm, \geq 0$ part. neutres, $\geq 0 K_L^0, \nu_\tau$ (49.51%) (“1 prong”)
- $h^\mp h^\mp h^\pm, \geq 0$ part. neutres, ν_τ (15.18%) (“3 prong”)

ici, h est soit un π soit un K.

Les deux premiers cas sont traités indirectement par l'identification des électrons et muons. On recherchera donc une ou trois traces chargées accompagnées d'activité calorimétrique, formant un mini-jet de masse invariante proche de celle du tau. Pour cela, on utilise la métrique de jade avec un y_{cut} (voir l'équation 2.1) donné par

$$y_{cut} \approx \left(\frac{6}{E_{LEP}} \right)^2$$

et le schéma de recombinaison E.

Un candidat tau sera validé à trois conditions :

- L'algorithme de Jade doit produire un jet de 1 ou 3 traces chargées.
- Les traces associées doivent répondre à certains critères de qualité : une impulsion suffisante, un nombre conséquent de hits dans la TPC, une origine proche du point d'interaction théorique et une direction suffisamment éloignée du tube à faisceau.
- L'énergie totale des particules chargées doit être au-delà d'un seuil choisi.

2.3.2 Réalisation concrète

Pour identifier et manipuler les leptons, trois classes ont été implémentées : AlMuon, AlElec et AlTau. Les classes correspondant à l'électron et au muon donnent accès aux données contenues dans la base de données utilisées pour les sélectionner. Elles possèdent également un pointeur vers l'“Energy flow” et la trace associée. La classe tau dérive quant à elle de la classe jet (AlJet).

Il est possible de sélectionner séparément électrons, muons et taus. Dans les trois cas, le procédé est le même(voir figure 2.5) :

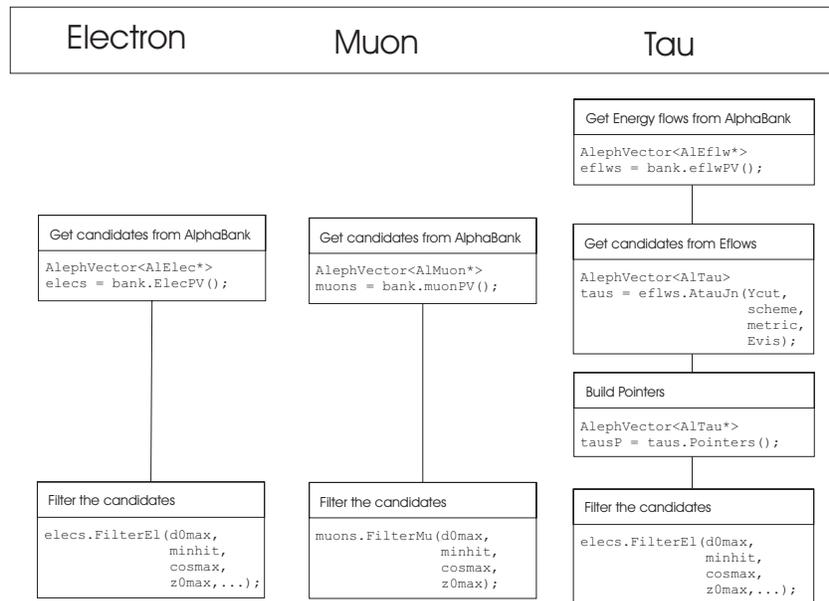


FIG. 2.5 – Procédure de sélection pour les électrons, muons et taus.

obtenir un liste de candidats

Les informations d'identification des électrons et muons (estimateurs, qualité,...) se trouvent respectivement dans les banques EIDT et MUID. La première phase consiste à obtenir un vecteur de candidats. Pour les électrons (les muons) il s'agit d'un vecteur reprenant tous les objets possédant une entrée dans la banque alpha EIDT (MUID). Le quadrivecteur s'obtient via la trace associée. Pour les taus, il s'agit d'un vecteur généré par l'utilisateur. En effet, si les candidats électrons et taus sont générés au niveau de l'*AlphaBank*, les candidats taus sont générés à la demande de l'utilisateur par la méthode *AtauJn()*.

La liste des électrons muons et taus est donc obtenue comme suit :

- `AlephCollection<AlMuon*> muons = bank.muonPV();`
- `AlephCollection<AlElec*> elecs = bank.elecPV();`
- `AlephCollection<AlEflw*> eflws = bank.eflwPV();`
`vector<float> taidcuts = theCardsReader()->TAIDcuts();`
`// parameters are: Ycut,scheme,metric,Evis`
`AlephCollection<AlTau> taus = eflws.ATauJn(taidcuts,Evis);`
`AlephCollection<AlTau*> tausP = taus.Pointers();`

Notons l'étape supplémentaire pour le tau, où l'utilisateur doit construire un vecteur de pointeurs à partir du vecteur de taus produit par *AtauJn()*.

Les paramètres standards pour *AtauJn()* sont :

Paramètre	Valeur	Description
Ycut	$\simeq 0.001$	coupure utilisée par l'algorithme de jet.
Scheme	0	Schéma de recombinaison : 0 pour E0 ou 1 pour E
Metric	1	1 pour Jade, 0 pour Durham.
Evis	QELEP	fixé ici à l'énergie centre de masse. Si 0 est choisi, l'énergie visible est utilisée.

La façon de l'utiliser est donc : $\text{AtauJn}(y_{\text{cut}}, 1, 1, E_{LEP})$, avec $y_{\text{cut}} = \left(\frac{5.8}{E_{LEP}}\right)^2$.

filtrer les candidats

La seconde phase consiste à filtrer les candidats avec des coupures plus précises dépendantes de l'analyse.

```
vector<float> muidcuts = theCardsReader()->MUIDcuts();
vector<float> elidcuts = theCardsReader()->ELIDcuts();
vector<float> taidcuts = theCardsReader()->TAIDcuts();
muons.FilterMu(muidcuts) ; // d0max,pmin,minhit,cosmax,z0max
//parameters are: almup,d0max,pmin,minhit,cosmax,
//                z0max,nwirem,r3cut,r2cut0l, r2cut1l,
//                r2cut2l,r2cuth,r5cutl,r5cuth
elects.FilterEl(almup,elidcuts);
tausP.FilterTau(taidcuts) ; // pmin,d0max,minhit,cosmax,z0max
```

Les vecteurs de paramètres sont obtenus via le driver - comme dans l'exemple - et contiennent alors le contenu des cartes de paramètres MUID, TAID et ELID. Ils peuvent également être remplis par l'utilisateur. Si une carte est absente, le driver fournit un vecteur de paramètres par défaut.

Un vecteur de muons, éventuellement vide, doit être donné pour l'algorithme de filtrage des électrons. Il est utilisé pour garantir qu'aucun électron n'a déjà été marqué comme muon. Si un vecteur vide est donné, il revient éventuellement à l'utilisateur de faire ce test.

Les paramètres par défaut sont ceux définis dans *ALPHA* :

Paramètre	Valeur	Description
d0max	0.5 (2 for τ)	distance d'approche minimale du faisceau
Pmin	1	impulsion minimale du lepton
Minhit	4	nombre minimal de hits dans la TPC
Cosmax	0.95	cosinus maximal de l'angle trace-faisceau
Z0max	10	distance minimale au point d'interaction en z
Nwirem	40	nombre de fils utiles pour le calcul dE/dx
R3cut	1000	limite sup. pour le paramètre r3 de dE/dx
R2cut0l	-3	limite inf. pour r2 dans une région propre de HCAL
R2cut1l	-7	limite inf. dans une région insensible
R2cut2l	-5	limite inf. dans une région surchargée
R2cuth	1000	limite supérieure pour r2.
R5cutl	-0.5	limite inférieure pour r5.
R5cuth	1000	limite supérieure pour r5.

2.4 Accès aux données générales

En plus des objets physiques évoqués jusqu'ici, l'analyse d'un événement dans le détecteur nécessite un certain nombre d'informations annexes, contenues de façon éparsée dans différentes banques de données. Pour accéder à ces informations, deux classes - `AlRun` et `AlEvent` - ont été implémentées. L'information ainsi disponible regroupe l'en-tête de chaque run et de chaque événement, ainsi que le statut des hautes tensions, du trigger et des différents sous-détecteurs. En particulier, la méthode "xlumok" se base sur une grande partie de ces informations pour décider si un événement peut être gardé pour l'analyse. Pour cela, les éléments suivants sont pris en compte :

- L'état de configuration du SiCAL
- Les différents systèmes de déclenchement actifs
- Les hautes tensions actives (correspondantes aux sous-détecteurs)
- La synchronisation des différents sous-détecteurs et de l'acquisition
- L'état du Vdet (après 1997) : alimentation, DAQ, etc.

En plus de cela, un certain nombre d'événements sont éliminés, de façon à tenir compte des dysfonctionnements qui ont échappé à l'acquisition mais ont été constatés par la suite. Il est enfin possible de ne pas considérer les événements obtenus pendant les périodes d'instabilité de l'énergie du faisceau, correspondant au passage d'une énergie à une autre, en cours d'acquisition. Ces périodes sont particulièrement nombreuses en 2000. Ces périodes correspondent en effet à ce qui est appelé "miniramps", périodes pendant lesquelles l'accélérateur était poussé dans ses derniers retranchements pour tenter d'atteindre des énergies records.

La méthode mise à disposition de l'utilisateur s'utilise comme suit.

```
// initialize the event related quantities
ALEvent myevent = bb.Event();
// Select event according to luminosity
if (!myevent.xlumok(QELEP)) {cout << "L";return;}
```

La méthode `xlumok` prend deux paramètres :

- L'énergie dans le centre de masse (QELEP) est un paramètre obligatoire.
- Un second argument (le mode) est facultatif. Mode 0(défaut) : sélection standard ; mode1 : admet également les miniramps ; mode2 : sélectionne uniquement les miniramps.

Un autre ensemble de routines intégré à l'objet `ALEvent` est chargé de la simulation du système de déclenchement, et de l'utilisation de l'information recueillie pour accepter ou non un événement. Dans le schéma classique de simulation, le système de déclenchement est simulé, de sorte que tant les événements Monte Carlo que les données disposent de l'information correspondante. Cependant, il est nécessaire d'éliminer les événements simulés qui ne déclencheraient pas l'acquisition. De plus, certaines tâches d'analyse se concentrent sur des topologies particulières pour lesquelles les efficacités des systèmes de déclenchement classiques sont mal connues. Pour ces analyses, une routine permet de recalculer un masque de déclenchement plus restreint et mieux compris.

Les méthodes développées s'utilisent comme suit :

```
// initialize the event related quantities
ALEvent myevent = bb.Event();
// Select event according to trigger for non-MINI data.
if (!myevent.IsTrigger(0)) {cout << "T" << ++Nbad;return;}
// Look at the trigger word
cout << "Trigger: " << myevent.GetRawTrigger() << endl;
```

L'unique argument de la méthode `IsTrigger` dépend du format du fichier de données à traiter. `IsTrigger(1)` est plus rapide lorsque l'on tourne sur des données de type MINI, alors que `IsTrigger(0)` est obligatoire dans les autres cas.

Enfin, différentes informations générales sont accessibles tant pour le run que pour l'événement. Pour l'événement, nous avons :

Time() renvoie la date et l'heure

number() renvoie le numéro unique de l'événement dans le run.

type() renvoie le type d'événement (MC,data,...)

ErrorStatus() renvoie un éventuel marqueur d'erreur.

Energy() renvoie, si l'information existe à ce niveau, l'énergie de la collision pour cet événement.

Pour le run, nous avons de même,

number() renvoie le numéro unique du run.

type() renvoie le type de run.

ExperimentNumber() renvoie le numéro de l'expérience en cours

Energy() renvoie, si l'information existe à ce niveau, l'énergie de la collision pour ce run.

2.5 Autres développements

De façon annexe, d'autres changements moins importants ont été apportés au code. Il s'agit notamment d'un travail de restructuration, d'une amélioration du traitement des copies d'objets, ou d'ajout de méthodes utiles. Citons par exemple dans *AlephCollection* :

1. La méthode *Pointers()* déjà mentionnée, qui retourne un vecteur de pointeurs à partir d'un vecteur d'objets.
2. La méthode *GetSum()* qui retourne le quadrivecteur somme associé à une classe d'objets de type "trace".
3. La méthode *GetMiss()* qui retourne l'impulsion manquante associée à une classe d'objets de type "trace".
4. Les méthodes *Sphericity()*, *aplanarity()* et *planarity()* qui retournent des paramètres topologiques.
5. La possibilité de trier un *AlephCollection* selon différents critères.

En outre, il est maintenant possible de verrouiller un objet, de façon à l'exclure de la suite de l'analyse. Cette fonctionnalité présente dans *ALPHA* avait été abandonnée au profit de la possibilité propre au vecteur STL d'effacer un élément. Il a été montré que le verrouillage d'un objet restait une fonctionnalité complémentaire permettant un gain en temps de programmation, en concision et en lisibilité du code.

Pour compléter les simulations Monte Carlo, la routine *gen_e12*, également implémentée dans le cadre de ce travail, permet de simuler l'énergie déposée dans un cône de 12 degrés autour du tube à faisceau par le faisceau lui-même³. Cette routine reproduit les distributions observées en fonction de l'énergie dans un ensemble d'événements déclenchés aléatoirement pendant la prise de données.

Enfin, un certain nombre de modifications ont été apportées au driver, ce afin de compléter l'ensemble de cartes de paramètres acceptées par *ALPHA++*. Les nouvelles cartes sont :

- MUID : paramètres pour la sélection des muons

³Les programmes de simulations ne prennent en effet pas en compte le faisceau.

- ELID : paramètres pour la sélection des électrons
- TAID : paramètres pour la sélection des taus
- SEED : choix de la graine du générateur de nombres aléatoires
- USER : définition de cartes utilisateur

L'accès à ces données se fait via un pointeur vers le `AlephCardReader` (classe du driver). Une méthode spécifique à chaque carte renvoie un vecteur de réels pour les coupures leptoniques ou un entier long pour le générateur de nombres aléatoires. La méthode `theCardsReader()->getUserCards()` renvoie l'objet suivant :

```
vector< pair< string, vector<double> > >
```

où la string est le nom de la carte utilisateur, et le vecteur de doubles le vecteur des paramètres éventuels. Un nombre arbitraire de cartes utilisateurs peut être défini.

Un générateur de nombres aléatoires a également été incorporé.

2.6 Comparaison détaillée des environnements *ALPHA* et *ALPHA++*

Pour tester les différents outils développés, la sélection hadronique d'un W isolé a été menée en traduisant l'existante version du code[9] d'*ALPHA* à *ALPHA++*. Les principales parties techniques et/ou physiques de la routine utilisateur peuvent être trouvées en annexe C. Notons que ce code a été écrit avant que la totalité des développements aient été accomplis. C'est ainsi que la syntaxe des algorithmes leptoniques est sensiblement différente de ce qui vient d'être présenté. Les principales quantités reconstruites sont : l'énergie dans un cône de 12° autour du faisceau, la masse invariante totale, l'impulsion manquante, l'énergie des électrons, muons et taus reconstruits, l'acolinéarité et l'acoplanarité, ainsi que les quadrivecteurs énergie-impulsion des jets reconstruits.

Jusqu'ici, seules de grossières estimations de la cohérence entre *ALPHA* et *ALPHA++* avaient été réalisées[12][13]. L'objectif était ici de comparer les résultats d'une analyse complète utilisant tous les algorithmes existants[14]. Pour arriver à ce résultat, les deux versions de l'analyse "W isolé" ont été exécutées sur le même fichier de données (*WW CC03 Monte Carlo sample*) et chaque quantité reconstruite a été comparée événement par événement.

Pour illustrer le degré de compatibilité des résultats, la figure 2.7 montre la distribution des différences relatives entre la masse invariante totale reconstruite avec *ALPHA* et *ALPHA++* (histogramme du haut). L'histogramme du bas de la même figure montre la distribution des différences relatives entre l'énergie du jet le plus énergétique obtenu après avoir verrouillé les objets constituant le candidat tau.

Ces résultats peuvent donc être considérés comme exactement équivalents, la seule différence observée étant attribuée à la représentation machine des nombres flottants (une différence relative de l'ordre de 10^{-7}). La structure qui apparaît dans ce graphique est comprise en considérant la figure 2.6. On y comprend en effet que l'erreur s'exprime comme un nombre entier de fois la précision du nombre flottant, cette précision diminuant par pas de deux lorsque la valeur absolue augmente. Le nombre d'entrées dans chaque histogramme représente le nombre d'événements pour lesquels la quantité tracée fait sens.

Comme le montre la figure 2.8, de tels résultats ont été obtenus pour les différentes variables leptoniques.

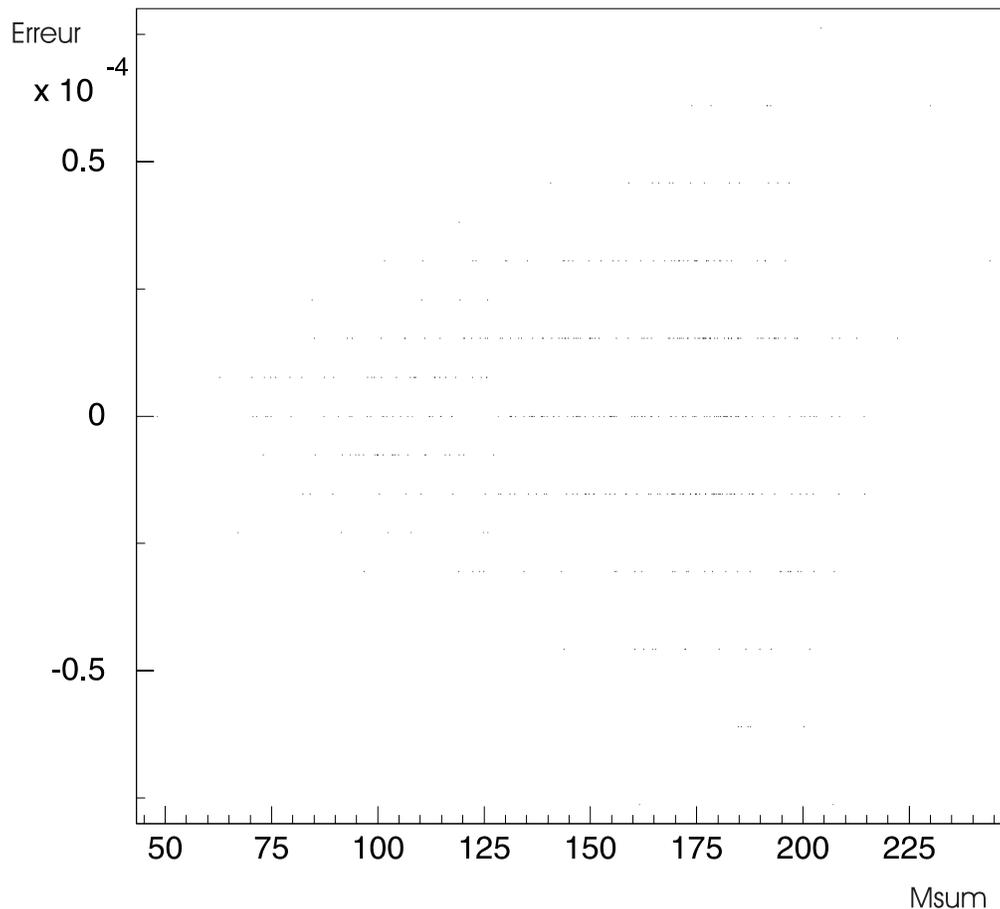


FIG. 2.6 – Différence entre *ALPHA* et *ALPHA++* pour la masse invariante totale exprimée en fonction de la masse invariante.

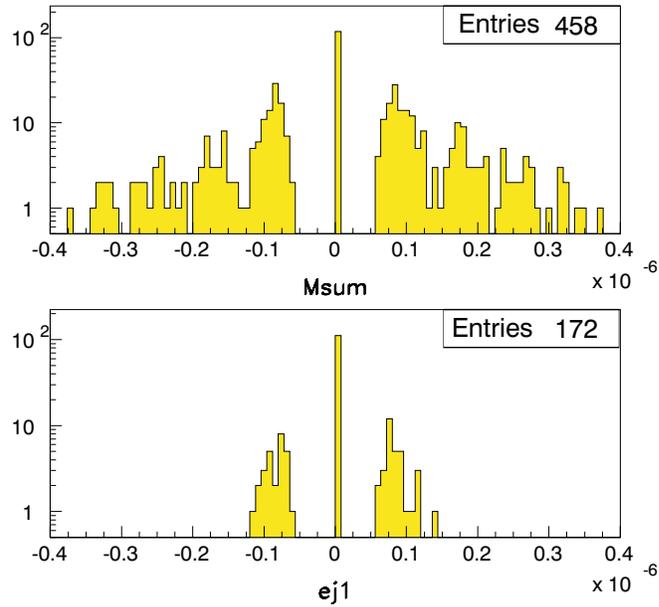


FIG. 2.7 – Différence relative entre *ALPHA* et *ALPHA++* pour la masse invariante totale et l'énergie du jet le plus énergétique. Cette dernière est calculée après avoir verrouillé les objets appartenant au candidat tau.

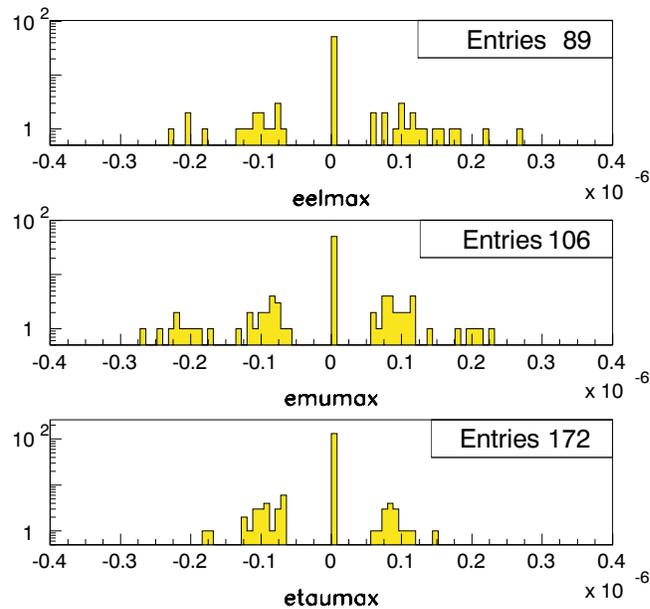


FIG. 2.8 – Différence relative entre *ALPHA* et *ALPHA++* pour l'énergie du plus énergétique muon, électron ou tau .

Chapitre 3

Recherche d'un boson de Higgs invisible auprès de l'expérience *ALEPH*

3.1 Introduction

Bien que le mécanisme de Higgs[15] ait été imaginé il y a 40 ans de cela, ce n'est que depuis peu que la conjonction de l'expérience et de la théorie limitent substantiellement le domaine de masse accessible au boson de Higgs, tout au moins dans le cadre du modèle standard des interactions fondamentales. Les mesures de précision[16] et les récentes tentatives d'observation directe au LEP[17] permettent notamment d'établir une densité de probabilité de présence du boson de Higgs, dans le cadre de ce modèle (figure 3.1)[18].

Ce que l'on appelle secteur de Higgs est une théorie de jauge découlant de la symétrie d'un champ scalaire permettant d'expliquer de façon séduisante la masse des différentes particules du modèle standard, des termes explicites de masses brisant la symétrie étant interdits. Ce mécanisme permet ainsi le passage d'une théorie unifiée de l'interaction électrofaible à une théorie effective à basse énergie - ce incluant le processus de brisure de symétrie. Notons cependant que le champ scalaire introduit peut être vu comme une solution ad hoc au problème des masses dans le modèle standard car il n'est pas (encore) justifié par l'observation. Cependant, bien que d'autres approches soient aussi envisagées[19], le mécanisme de Higgs reste le scénario le plus apprécié, que ce soit dans le cadre du modèle standard, où le boson de Higgs est unique, ou dans les différentes approches "au delà du modèle standard" (supersymétries, dimensions supplémentaires macroscopiques, etc.).

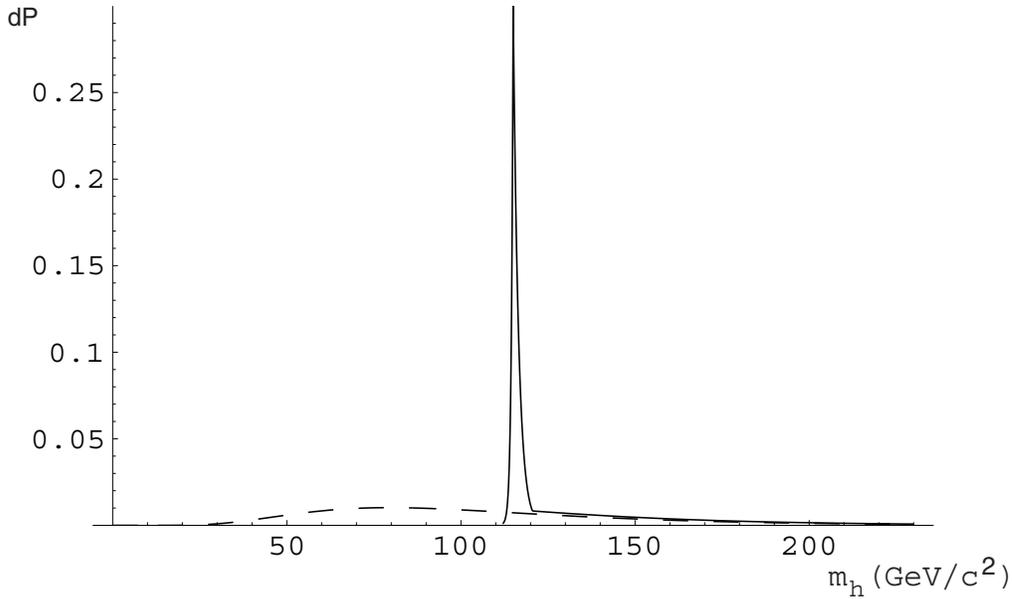


FIG. 3.1 – Densité de probabilité de présence d'un boson de Higgs, en utilisant les informations indirectes (trait interrompu) et en intégrant les résultats de recherches directes (trait plein)

3.2 Le mécanisme de Higgs

Le mécanisme de Higgs est donc un mécanisme de brisure de symétrie faisant intervenir un (ou plusieurs) champ scalaire pour faire apparaître des termes de masse aux particules. Le développement présenté en annexe B permet de comprendre, dans le cas simple abélien d'un champ scalaire complexe, le principe d'un tel mécanisme de brisure de symétrie.

Dans le cadre du modèle standard, théorie non abélienne, la situation est très similaire. Si le champ de Higgs acquiert une valeur non nulle dans le vide, les couplages aux bosons et fermions ressortent directement. Par exemple, considérons une famille consistant en un doublet de quarks gauchers et en un doublet de leptons $q_L = (u_L d_L)$, $l_L = (\nu_L e_L)$ ainsi que le singlet droitier correspondant u_R , d_R et e_R . Nous avons

$$\mathcal{L} = (D_\mu \phi)^* (D^\mu \phi) - g_d \bar{q}_L \phi d_R - g_u \bar{q}_L \phi^c u_R - g_l \bar{l}_L \phi e_R. \quad (3.1)$$

avec le champ conjugué de charge $\phi^c = i\tau_2 \phi^*$, et la dérivée covariante

$$D^\mu = \partial^\mu + ig \frac{\vec{\tau}}{2} \cdot \vec{W}^\mu + ig' \frac{Y}{2} B^\mu. \quad (3.2)$$

Pour le champ de Higgs, $Y = 1$ alors que les τ_i sont les matrices de Pauli, normées par $Tr(\tau_i\tau_j) = 2\delta_{ij}$. Les matrices d'isospin sont données par $\vec{T} = \vec{\tau}/2$ et la charge électrique est $Q = T_3 + Y/2$. Le champ de Higgs réel est obtenu en perturbant le champ autour d'un minimum (le vide), choisi sur la droite réelle,

$$\phi \rightarrow \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ v + h(x) \end{pmatrix}, \quad (3.3)$$

nous obtenons les masses des particules

$$m_W = gv/2 \quad (3.4)$$

$$m_f = g_f v / \sqrt{2} \quad (3.5)$$

$$v = (\sqrt{2}G_F)^{-1/2} = 246 \text{ GeV} \quad (3.6)$$

où g est la constante de couplage du groupe SU(2), g' la constante du groupe U(1). Une étude plus approfondie de l'expression obtenue permet également d'obtenir les différents couplages du champ ϕ^0 . Ces couplages aux fermions sont proportionnels à la masse des fermions, et le rapport de branchement est donc proportionnel à m_f^2 . La largeur partielle de désintégration en une paire quelconque de fermion est

$$\Gamma(\phi^0 \rightarrow f\bar{f}) = \frac{N_c g^2 m_f^2}{32\pi m_W^2} \beta^3 m_{\phi^0}, \quad (3.7)$$

où N_c dénote le nombre de couleurs (1 pour les leptons et 3 pour les quarks), et $\beta^2 = 1 - 4m_f^2/m_{\phi^0}^2$.

De ces résultats, il est important de retenir les choses suivantes :

1. On voit apparaître un boson *physique massif*.
2. La théorie ne fixe pas la masse de ce boson.
3. Le couplage aux particules est connu, et proportionnel à la masse dans le cas des fermions.

Dans le cadre du modèle standard donc, les différents rapports de branchement du Higgs sont ceux présentés à la figure 3.2. Aux énergies accessibles au LEP, le mode dominant correspond à la paire de quarks la plus lourde accessible : la paire $b\bar{b}$.

Cependant, comme le répètent à plusieurs reprises les auteurs du Higgs Hunter's Guide[20], peu de ceux qui ont étudié la physique dans le secteur de Higgs s'attendent à rencontrer un unique boson de Higgs neutre, quelle que soit sa masse. Parmi les théories concurrentes, une place de choix est occupée par les différents modèles supersymétriques. Pour établir les signatures de bosons de Higgs dans de tels modèles, il est nécessaire de considérer les multiples modes de désintégration des neutralinos et charginos, produits de désintégration du Higgs.

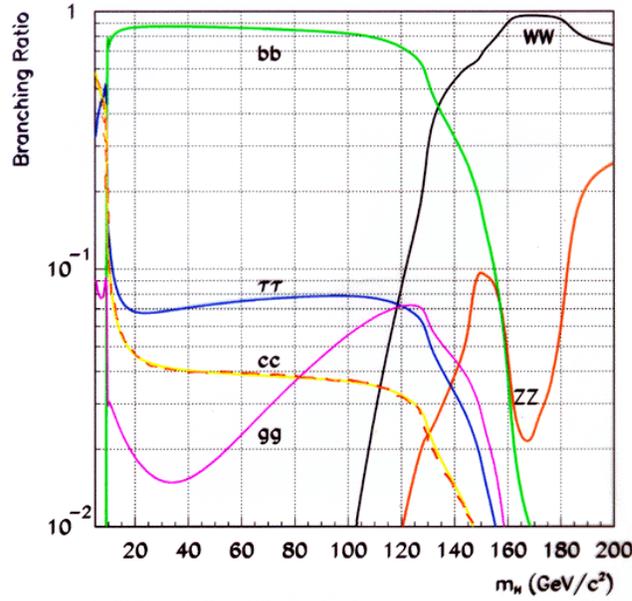
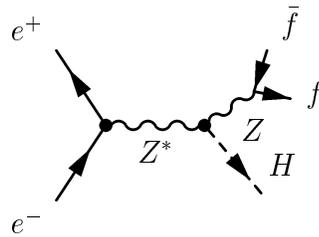


FIG. 3.2 – Rapport de branchement des divers modes de désintégration du boson de Higgs en fonction de sa masse dans le modèle standard minimal

3.3 Production du boson de Higgs au LEP

Le mode dominant la production du Higgs au LEP consiste en la production en association avec un Z, dont on observe la désintégration hadronique (ce mode, comptant pour 70% des désintégrations du Z, a été suggéré pour la première fois par J. Bjorken en 1977)[21].



La section efficace pour un tel mécanisme de production est donnée, au tree-level, par l'expression[22, 23]

$$\sigma_{ZM} = \frac{G_F^2 m_Z^4}{96\pi s} (1 + (1 - 4\sin^2 \theta_W)^2) \lambda^{1/2} \frac{\lambda + 12m_Z^2/s}{(1 - m_Z^2/s)^2}. \quad (3.8)$$

Dans cette expression, on retrouve dans le préfacteur les couplages HZ^0Z^0 et Zee . Vient alors le facteur d'espace de phase :

$$\lambda \equiv \left(1 - \left(\frac{m_H^2}{s}\right) - \left(\frac{m_Z^2}{s}\right)\right)^2 - \frac{4m_H^2 m_Z^2}{s^2} \quad (3.9)$$

traduisant la cinématique dans la désintégration en deux corps. Le terme $(1 - m_Z^2/s)^2$ vient du propagateur du Z , alors que le terme $\lambda + 12m_Z^2/s$ s'obtient lors du calcul de la trace, et correspond aux deux termes dans la somme sur les polarisation du Z .

La section efficace attendue dans le cadre du modèle standard pour la production associée de H et Z est représentée à la figure 3.3 Elle est obtenue en utilisant le générateur Monte Carlo HZHA02 [24] (incluant les corrections radiatives) et comparée à la valeur au tree-level telle qu'obtenue en utilisant la formule (3.8). Il est intéressant de noter que la différence avec la valeur au tree-level est de 10% tant que l'on reste loin du seuil. Près de celui-ci, elle augmente, notamment parce que la largeur du Z est négligée dans l'expression 3.8. Précisons encore que l'incertitude théorique de HZHA est estimée à 5%.

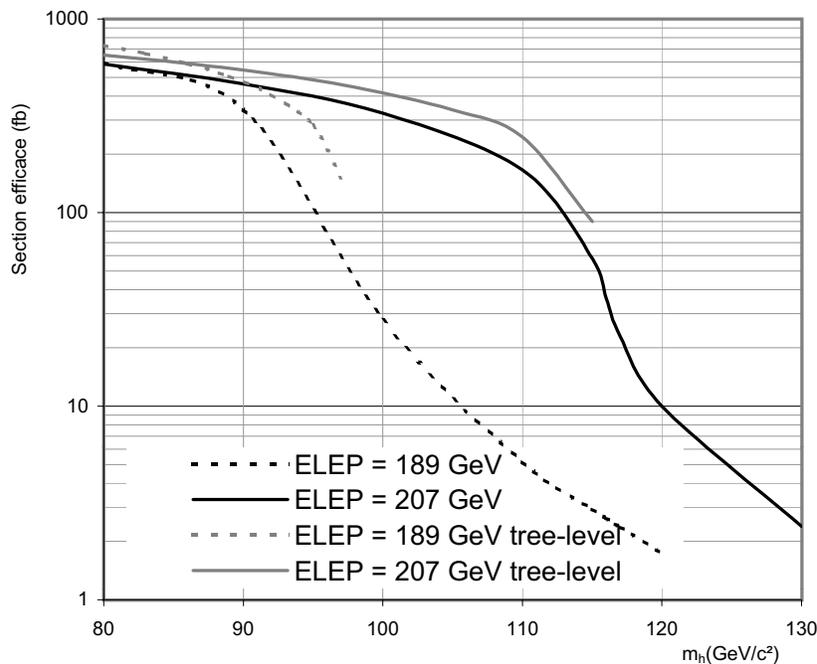


FIG. 3.3 – section efficace attendue dans le cadre du modèle standard pour la production associée de H et Z. Les courbes correspondent à la section efficace obtenue au tree-level (gris) et à l'aide du générateur Monte Carlo HZHA02.

3.4 Le mode de désintégration invisible

Dans le cadre du modèle supersymétrique minimal (MSSM), on est amené à introduire deux doublets d'isospin de Higgs plutôt qu'un. La brisure de symétrie fait apparaître parmi ces 8 degrés de liberté 3 bosons de Goldstone, absorbés par les bosons de jauge comme précédemment, et 5 bosons de Higgs (h , H^+ , H^- , H et A). Le plus léger boson de Higgs pair sous opération de CP (h) est susceptible de se désintégrer de façon invisible en une paire du plus léger neutralino

$$h \rightarrow \chi\chi,$$

si la R-parité¹ est conservée et si ce neutralino est suffisamment léger. La section efficace de la réaction $e^+e^- \rightarrow hZ$, où le boson de Higgs se désintègre en des états finaux invisibles, peut s'exprimer sous la forme

$$\xi^2 \sigma(e^+e^- \rightarrow HZ),$$

où ξ^2 représente un facteur dépendant du modèle, toujours inférieur à 1 et qui réduit la prédiction par rapport à la production d'un Higgs du modèle standard. La présence d'un tel boson de Higgs est presque totalement exclue avec une masse proche de la limite du LEP2 par la recherche directe de neutralinos[25]. Cependant, ces limites sont établies dans le cadre du modèle de grande unification (GUT) et de nouvelles régions de l'espace de phase sont accessibles si l'on choisit un autre modèle.

De tels modes invisibles sont par ailleurs présents dans le cadre d'autres théories ; tels par exemple sont certains des modèles à axions ou majorons, dans lesquels le boson de Higgs est susceptible de se désintégrer respectivement en axions ou en majorons stériles[26].

Un autre mode invisible pourrait également être la désintégration d'un Higgs du modèle standard en deux gravitons. Si ce mode de désintégration est de commune acceptation hautement peu probable, il pourrait se révéler être le mode dominant dans le cadre des théories à dimensions extra-spatiales macroscopiques (LEDS).

En dehors de tout contexte théorique, la recherche de modes invisibles de désintégration est ainsi un sujet susceptible de contraindre les modèles, sinon de mettre en évidence le tant convoité boson.

De précédentes études de ce processus à des énergies moins élevées ([27] à [30]) ont permis d'exclure à 95% toute masse m_h inférieure à $94.5 GeV/c^2$, pour une valeur de $\xi^2 = 1$. Le principal objectif de ce travail est de refaire une telle analyse pour une énergie centre de masse de $207 GeV$, et de tester l'hypothèse du Higgs. Par ailleurs, nous désirons reproduire (et ainsi conforter) les résultats déjà obtenus à $189 GeV$.

¹La R-parité qui est définie par $R = (-1)^{(B-L)+2S}$ vaut 1 pour la matière standard et -1 pour les partenaires supersymétriques.

3.5 Définitions et notations

Afin de classer et décrire les différents processus rencontrés au LEP, ainsi que pour éviter toute ambiguïté dans les analyses, il convient de définir quelques notions importantes.

Une bonne trace est définie comme ayant au moins 4 hits dans la TPC, et une valeur de $|\cos \theta| < 0.95$. De plus, elle doit trouver son origine dans un cylindre long de 10 cm et de 2 cm de rayon centré au point d'interaction nominal et parallèle au tube à faisceau (les particules créées lors de la désintégration hadronique devraient trouver leur origine au point d'interaction principal). Seules les particules chargées suffisamment éloignées du faisceau sont sélectionnées, car les mesures dans les calorimètres LCAL et SICAL sont moins précises, et le rayon de courbure moins bien mesuré.

Un jet est par définition, dans un contexte théorique, un ensemble de particules résultant de l'hadronisation en vol d'un quark ou d'un gluon. Dans un contexte expérimental, ce sera un ensemble de particules topologiquement proches et regroupées par un algorithme de sélection. Sauf mention contraire, l'algorithme utilisé correspond à la métrique de Durham et au schéma de recombinaison E. Par extension, le jet désigne le quadrivecteur énergie-impulsion reconstruit.

L'**acolinéarité** est par définition l'angle que font entre eux les deux jets. Si cet angle est de 180° , on parlera de jets *colinéaires*.

L'**acoplanarité** est quant à elle l'angle que font entre eux les deux jets dans le plan (r, ϕ) . Si cet angle est de 180° , on parlera d'événement *coplanaire*.

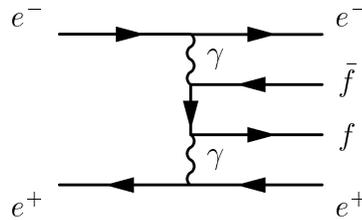
3.6 Signal et bruits de fond

Le processus étudié dans le cadre de ce mémoire rend compte de la désintégration invisible d'un boson de Higgs produit en association avec un Z, dont on observe la désintégration hadronique (ce mode compte pour 70% des désintégrations du Z).

Pour dresser la liste des principaux bruits de fond à envisager, il est utile dans un premier temps de considérer les grandes classes de signaux rencontrés au LEP. Ces grandes classes sont au nombre de 3 : les processus " $\gamma\gamma$ ", les processus "à 2 fermions" et les processus "à 4 fermions".

processus " $\gamma\gamma$ "

Ces processus résultent de l'interaction de deux photons émis respectivement par l'électron et le positon.



Il s'agit d'un cas (très) particulier d'interaction à 4 fermions, mais les photons doivent avoir une faible impulsion, de par la forme en $1/q^2$ de leur propagateur. Ainsi, les deux leptons vont se perdre dans le tube à faisceau et ne sont pas détectés.

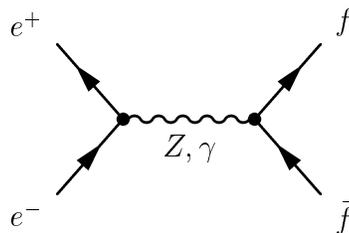
Les caractéristiques permettant d'identifier ces événements sont :

- La masse invariante $M_{f\bar{f}}$ est faible, car seule une fraction de l'énergie incidente est emportée par les photons.
- L'impulsion manquante $P_{t_{miss}}$, définie comme l'impulsion manquante dans le plan (r, ϕ) , est faible car le fermion et son antiparticule ont approximativement la même impulsion transverse.
- Suite à la grande asymétrie des impulsions des photons, il y a acolinéarité.
- Puisque $P_{t_{miss}}$ est quasi nulle, il y a également coplanarité.

Bien que la signature de ces événements soit très différente de la signature attendue pour le signal, ces événements sont simulés car la section efficace est très grande. Notons au passage la difficulté de calculer ces processus, pour lesquels la singularité correspondant à un angle de diffusion nulle des leptons exige de tenir compte de leur masse, alors que les énergies mises en jeu sont de l'ordre du GeV .

processus “à 2 fermions”

C'est le cas le plus simple d'interaction directe e^+e^- .

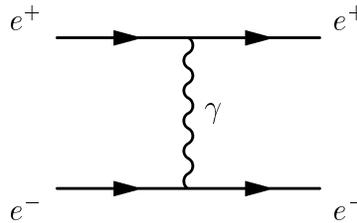


Les caractéristiques rencontrées sont généralement les suivantes :

- La masse invariante $M_{f\bar{f}}$ est importante (proche de $200 GeV/c^2$)
- L'impulsion manquante est idéalement nulle.
- Puisque il n'y a pas d'impulsion manquante, il y a colinéarité.
- Puisque il n'y a pas de $P_{t_{miss}}$, il y a coplanarité.

Cependant, la présence de bremstrahlung² dans la voie d'entrée introduit un important retour radiatif du Z^0 , se traduisant par des événements avec une masse invariante proche de $91 \text{ GeV}/c^2$ et une grande acolinéarité. La production directe d'une paire $q\bar{q}$ au retour radiatif du Z^0 constitue un important bruit de fond, dans la mesure où le photon initial n'est pas détecté.

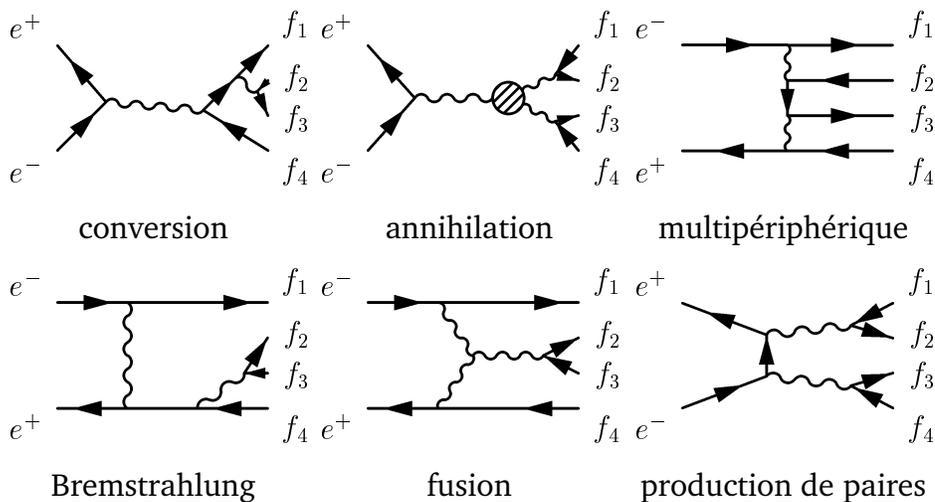
Dans le cas particulier où $f = e$, le diagramme suivant joue un rôle dominant.



Ce processus, dit Bhabha, est exploité pour estimer la luminosité.

processus “à 4 fermions”

Plusieurs diagrammes sont regroupés sous cette appellation. Tous sont caractérisés par la présence de 4 fermions dans la voie de sortie.



Il s'agit de l'annihilation, de la conversion, des événements multipériphériques, du bremstrahlung, de la production de paires et des processus non abéliens (fusion par “couplage de jauge trilinéaire”).

²rayonnement de freinage

TAB. 3.1 – Principaux bruits de fond observables, accompagnés d'un diagramme de Feynmann caractéristique.

processus	diagramme	processus	diagramme
$q\bar{q}$		WW	
$W e \nu$		ZZ	
$Z e e$		$Z \nu \nu$	
$\gamma\gamma$			

conclusion

De cette discussion, il ressort que les principaux bruits de fond qu'il faudra éliminer sont :

- La production de paires W^+W^- .
- La production d'un W isolé.
- La production d'une paire de Z^0 .
- La production d'un Z isolé (processus Zee ou $Z\nu\nu$).

Ces différents processus sont repris dans la table 3.1, accompagnés d'un diagramme représentatif (souvent choisi arbitrairement parmi d'autres).

TAB. 3.2 – Répartition de la luminosité en 2000.

E (GeV)	nombre de runs	luminosité (pb^{-1}) (+ miniramps)
199.76	10	0.844(0.844) \pm 0.015
201.77	9	0.705(0.706) \pm 0.014
202.67	19	1.760(1.764) \pm 0.022
203.79	81	8.325(8.357) \pm 0.048
205.03	869	71.559(72.054) \pm 0.140
206.31	677	65.525(65.873) \pm 0.135
206.61	702	60.810(61.224) \pm 0.130
207.95	400	7.318(7.674) \pm 0.046

3.7 Données et échantillons Monte Carlo

L'analyse porte d'une part sur les données collectées à 189 GeV en 1998, d'autre part sur les données à 204 – 209 GeV de 2000. Les données de 1999, acquises à une énergie intermédiaire avec une faible luminosité ne sont pas analysées ici.

En 1998, une luminosité de $174.20 \pm 0.20 pb^{-1}$ a été accumulée à une énergie de 189 GeV, répartie sur 1355 runs. Le but de l'analyse à 189 GeV est de valider la méthode d'analyse en reproduisant (améliorant) les performances de l'analyse officielle.

En 2000, la luminosité accumulée à une énergie supérieure à 200 GeV est de $216.846 pb^{-1}$ ($218.496 pb^{-1}$ en acceptant les miniramps). Elle se répartit comme indiqué dans le tableau 3.2. Ces données sont résumées par la figure 3.4. La limite supérieure en énergie, atteinte pour quelques événements seulement, est de 209 GeV. Dans cette analyse, seules les données à plus de 203 GeV sont analysées. L'objectif est ici d'obtenir la meilleure limite sur la masse du boson de Higgs, pour $\xi^2 = 1$.

La simulation des événements dans *ALEPH* comporte deux parties distinctes : tout d'abord la génération du quadrivecteur énergie-impulsion de particules en fonction du processus étudié, ensuite la simulation de la propagation de ces particules dans le détecteur.

La propagation des particules se base sur les outils GEANT/GHEISHA. Les dépôts d'énergie produits sont dégradés de façon à reproduire la résolution spatiale appropriée. On obtient ainsi des "données brutes" contenant la même information que les données réelles. Les signaux analogiques et digitaux sont alors simulés à leur tour de façon à reproduire la réponse du système de déclenchement et à déterminer l'efficacité globale du détecteur.

Les processus $q\bar{q}$ sont générés par KORALZ[31].

Les différents processus à 4 fermions sont simulés séparément. Bien que cela signifie que l'on ne tient pas compte des effets d'interférence entre ces différents

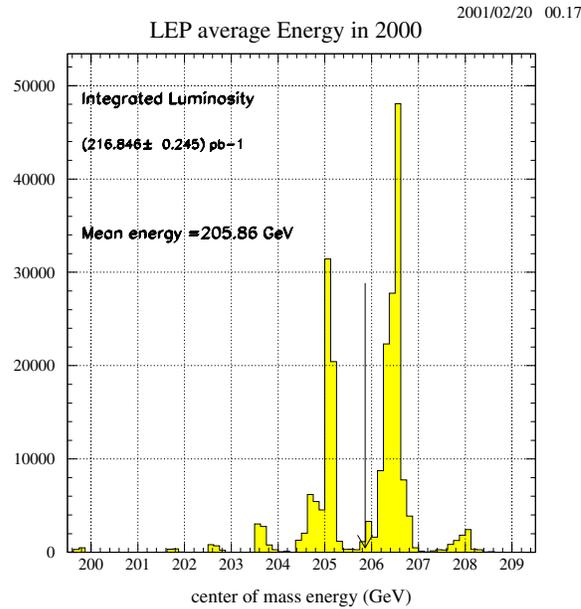


FIG. 3.4 – Répartition de la luminosité en 2000, en fonction de l'énergie centre de masse.

processus, cette approche a pour but d'augmenter la statistique pour les signaux non dominants, vrai bruit de fond de cette analyse.

La production de paires de W est simulée avec KORALW[32]. PYTHIA[33] est utilisé pour générer les autres processus à quatre fermions, tels que ZZ , Zee et $W_{e\nu}$. Les événements $Z\nu\nu$ sont produits par un générateur dédié.

A 189 GeV , nous disposons en outre de simulations obtenues par le générateur GRACE4f[34], plus précis que PYTHIA (voir la figure 3.5 pour une comparaison des deux générateurs). Si les sections efficaces totales sont en bon accord, PYTHIA suppose une masse nulle pour l'électron, et doit donc traiter approximativement la singularité présente lorsque l'angle de diffusion de l'électron est nul, surestimant ainsi $W_{e\nu}$ d'un facteur 1.2. PYTHIA ne prend de plus pas en compte les diagrammes multipériphériques, contribuant de façon conséquente à faible masse invariante. Dans cette analyse, cette contribution est éliminée par les coupures de sélection, mais il est nécessaire d'appliquer le facteur correctif, de façon à adopter la normalisation de GRACE4f lorsque l'on utilise PYTHIA (pour les données à plus de 203 GeV).

Finalement, PHOT02[35] est utilisé pour simuler les processus à deux photons "tagged". De façon à tenir compte des événements $\gamma\gamma$ pour lesquels les leptons ne sont pas détectés (événements dits "untagged"), un facteur 5 est appliqué sur la simulation. On reproduit ainsi approximativement la distribution observée dans les données. Il s'agit ici d'une approximation relativement grossière, car les topologies

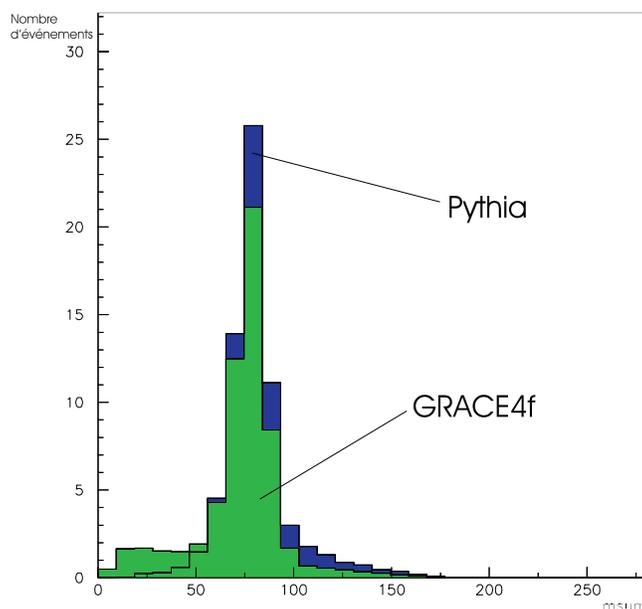


FIG. 3.5 – Comparaison des générateurs PYTHIA et GRACE4f pour le canal $W e \nu$.

“tagged” et “untagged” peuvent différer légèrement l’une de l’autre. En général, les Analyses *ALEPH* demandent deux coupures fortes pour éliminer la totalité de ces événements.

Le signal est produit par le générateur dédié HZHA. A 189 GeV , le signal a été simulé pour des masses allant de 65 GeV à 100 GeV , par pas de 5 GeV , ainsi qu’à 88 GeV et 92 GeV . A plus de 203 GeV , le signal est simulé pour des masses de 85 GeV à 105 GeV par pas de 5 GeV , ainsi qu’à 102 GeV , 108 GeV , 112 GeV et 114 GeV . A haute énergie, tant le signal que les différents bruits de fond sont simulés de 204 GeV à 208 GeV .

La figure 3.6 montre le nombre attendu d’événements en fonction de la masse invariante, pour une luminosité intégrée de 174.20 pb^{-1} à une énergie centre de masse de 208.6 GeV . Dans ce graphe, qui constitue un échantillon de contrôle, les différentes contributions au bruit de fond ont été additionnées l’une à l’autre alors que le signal attendu est surimposé. Un facteur 5 (discuté plus haut) est appliqué aux processus $\gamma\gamma$. Les pré-coupures appliquées sont réparties en deux ensembles :

1. – class 16 (événements de haute multiplicité)
 - $|\cos\theta_{miss}| > 0.95$
 - $m_{sum} > 5\text{GeV}$
 - $E_{wedge} < 20\%$ de E_{LEP} .
 - acolinéarité < 175

2. – class 16 (événements de haute multiplicité)
 - $|\cos\theta_{miss}| > 0.95$
 - $m_{sum} > 5\text{GeV}$
 - $E_{vis} < 90\%$ de E_{LEP} .
 - acoplanarité < 175
 - $Pt_{miss} < 4\%$ de E_{LEP}

Un événement est conservé s'il satisfait un des ces deux ensembles de pré-coupures. Ces quantités sont décrites à la section 3.8.1.

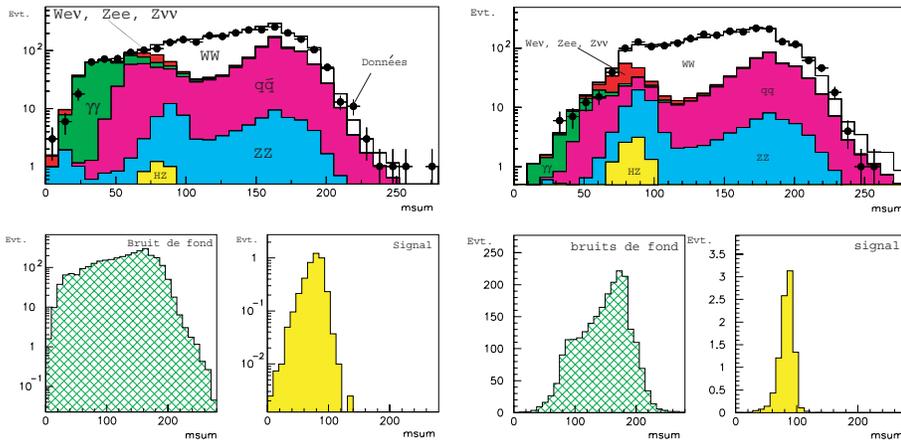


FIG. 3.6 – Nombre attendu d'événements en fonction de la masse invariante à 188.6 GeV et entre 204 et 208 GeV . Les contributions au bruit de fond sont cumulées, alors que le signal (boson de Higgs de $100\text{ GeV}/c^2$) est surimposé. Les points correspondent aux données collectées en 1998 et 2000 respectivement. Les pré-coupures mentionnées dans le texte sont appliquées.

Les échantillons utilisés ainsi que la quantité d'événements simulés à 189 GeV et entre 204 et 208 GeV sont repris respectivement tables 3.3 et 3.4.

TAB. 3.3 – échantillons MC utilisés à 189 GeV. Un échantillon de signal est également indiqué.

Processus	Générateur	section efficace (pb)	nombre produit
$u\bar{u}(\gamma)$	KORL08	20.299	145000
$d\bar{d}(\gamma)$	KORL08	20.018	145000
$c\bar{c}(\gamma)$	KORL08	20.331	145000
$s\bar{s}(\gamma)$	KORL08	20.055	145000
$b\bar{b}(\gamma)$	KORL08	19.604	145000
WW	KORALW	16.595	150000
$W e \nu$	GRACE4f	0.691	15000
ZZ	PYTHIA	2.76	60000
$Z e e$	PYTHIA	99.11	1000000
$Z \nu \nu$	ZNUNU	0.0113	10000
$\gamma\gamma \rightarrow u\bar{u}, d\bar{d}$	PHOTO	487	400000
$\gamma\gamma \rightarrow c\bar{c}$	PHOTO	93.2	200000
$\gamma\gamma \rightarrow s\bar{s}$	PHOTO	23.9	40000
$\gamma\gamma \rightarrow b\bar{b}$	PHOTO	0.626	12000
$H(100 \text{ GeV})Z$	HZHA	0.0278	10 x 2000

TAB. 3.4 – échantillons MC utilisés à 207 GeV. La section efficace est considérée à 206 GeV, et le nombre produit est le nombre total de 204 GeV à 208 GeV.

Processus	Générateur	section efficace (pb)	nombre produit
$u\bar{u}(\gamma)$	KORL08	16.473	700000
$d\bar{d}(\gamma)$	KORL08	15.998	700000
$c\bar{c}(\gamma)$	KORL08	16.486	700000
$s\bar{s}(\gamma)$	KORL08	16.003	700000
$b\bar{b}(\gamma)$	KORL08	15.649	700000
WW	KORALW	17.541	250000
$W e \nu$	PYTHIA	0.884	100000
ZZ	PYTHIA	2.81	250000
$Z e e$	PYTHIA	98.72	3000000
$Z \nu \nu$	ZNUNU	0.018	100000
$\gamma\gamma \rightarrow u\bar{u}, d\bar{d}$	PHOTO	484	1250000
$\gamma\gamma \rightarrow c\bar{c}$	PHOTO	94.4	250000
$\gamma\gamma \rightarrow s\bar{s}$	PHOTO	23.4	75000
$\gamma\gamma \rightarrow b\bar{b}$	PHOTO	0.641	15000
$H(100 \text{ GeV})Z$	HZHA	0.320263	10 x 5000

3.8 Sélection hadronique à 189 GeV

3.8.1 Optimisation des coupures

Bien que notre objectif premier soit d'étudier les données obtenues en 2000, à des énergies supérieures à 200 GeV, nous commençons par analyser les données à 189 GeV afin de valider et comparer notre analyse à des analyses précédentes.

La topologie étudiée consiste en une paire de jets avec une masse invariante proche de la masse du Z et une grande énergie manquante. Les échantillons Monte Carlo des bruits de fond discutés dans la section précédente (les processus du modèle standard $\gamma\gamma$, W^+W^- , $W\nu$, ZZ , Zee , $Z\nu\nu$ et $q\bar{q}$) ainsi que du signal ont été utilisés pour déterminer les efficacités respectives. La figure 3.6 montre qu'une variable discriminante évidente est la masse invariante. Cependant, le très petit nombre d'événements attendus rend nécessaire l'usage de coupures plus poussées.

Le programme de sélection élaboré dans le cadre d'ALPHA++ pour la recherche d'un Higgs invisible est similaire à l'analyse "single W" décrite dans la première partie de ce mémoire. Les principales variables calculées sont :

- le nombre et l'énergie des traces
- l'énergie dans un cône de 12° autour du tube de faisceau
- l'énergie dans un secteur de 30° en ϕ autour de P_{miss}
- les quadrivecteurs P_{miss} et P_{sum}
- l'acoplanarité et l'acolinéarité
- y_{12} et y_{13} (les coupures pour lesquelles la topologie passe respectivement de 1 à 2 jets et de 2 à 3 jets).
- Les quadrivecteurs des leptons les plus énergétiques anti-parallèles à P_{miss}
- L'énergie de la trace la plus isolée
- plusieurs variables leptoniques obtenues en forçant 3 jets.

Le code source correspondant peut être trouvé en annexe D.

Comme il a déjà été mentionné, la topologie étudiée consiste en une paire de jets acoplanaires (provenant de la désintégration du Z associé). De façon à sélectionner ces 2 jets, 3 sont forcés, et l'angle minimal entre 2 des 3 jets doit être inférieur à 52°. De même, la masse invariante de 2 des 3 jets ne peut être supérieure à 30 GeV/c². Une coupure supérieure (174°) est appliquée sur l'acoplanarité.

Le pic du Z est localisé en sélectionnant un intervalle en m_{sum} et E_{sum} : $m_{sum} \in [0.19E_{LEP}, 0.69E_{LEP}]$ et $E_{sum} \in [0.31E_{LEP}, 0.60E_{LEP}]$. Les coupures inférieures correspondantes constituent les premières coupures anti- $\gamma\gamma$.

Ces événements $\gamma\gamma$ ainsi que les événements $q\bar{q}(\gamma)$ sont très gênants. Outre le fait d'avoir une grande section efficace, ces processus sont en effet moins bien simulés que les autres processus du modèle standard. Les processus $\gamma\gamma$ sont éliminés en exigeant au moins 10 traces chargées, comptant pour 16% de l'énergie centre de masse. L'impulsion transverse totale doit être supérieure à 2 % de l'énergie centre

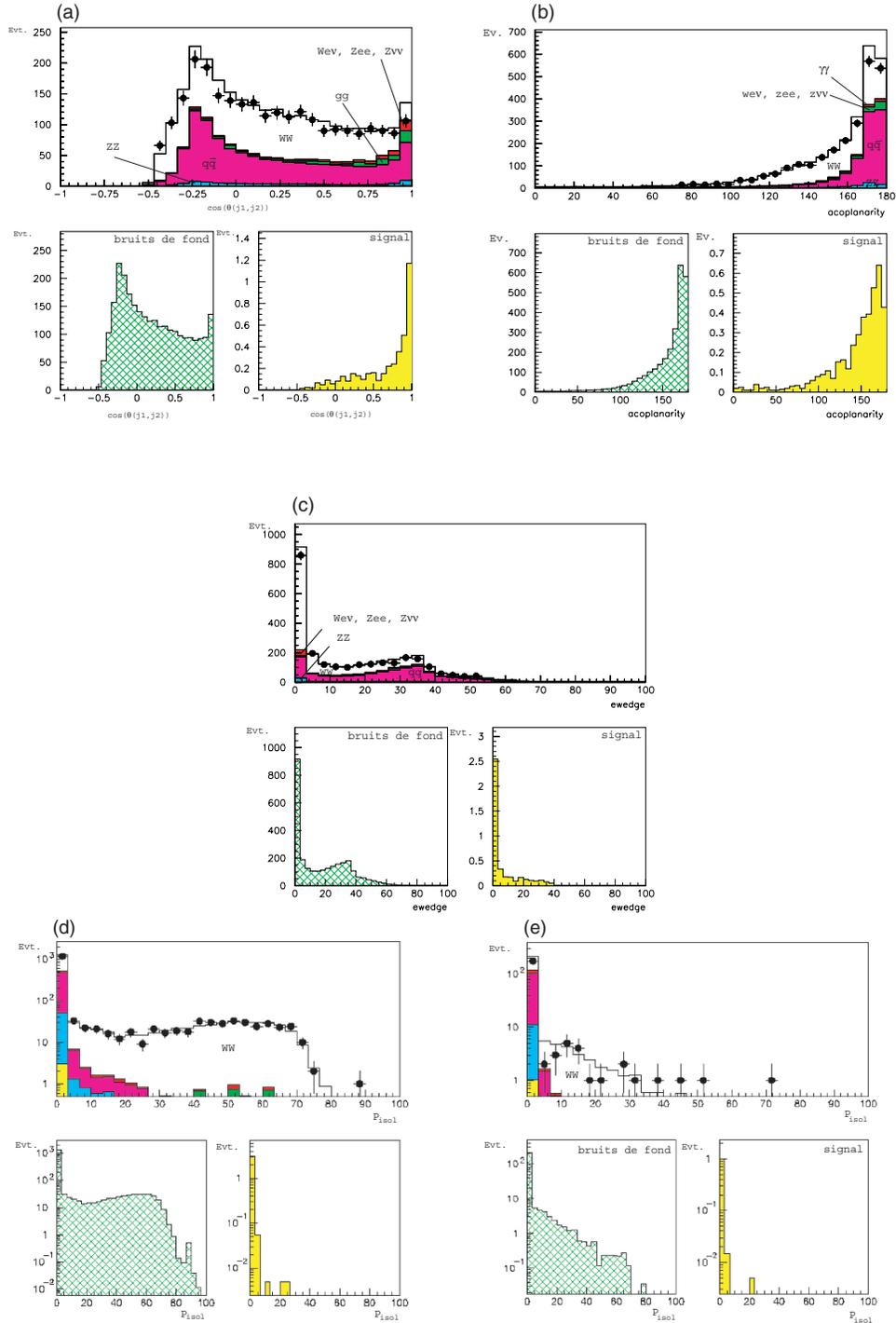


FIG. 3.7 – Quelques quantités significatives intervenant dans les coupures de sélection. (a) $\cos(\theta(j_1, j_2))$ après les premières coupures anti- $\gamma\gamma$ ($m_{sum} > 0.19E_{LEP}$, $E_{sum} > 0.31E_{LEP}$). (b) acoplanarité après les premières coupures anti- $\gamma\gamma$. (c) Energie dans le wedge après les premières coupures anti- $\gamma\gamma$. (d) impulsion de la trace la plus isolée après toutes les coupures anti- $\gamma\gamma$. (e) impulsion de la trace la plus isolée après toutes les coupures anti- $\gamma\gamma$ et après avoir utilisé les algorithmes de reconnaissance des électrons et muons.

de masse. On impose également une impulsion manquante faisant un grand angle avec l'axe du faisceau pour rejeter les événements avec des particules non détectées à de petits angles. Les processus $q\bar{q}(\gamma)$ sont éliminés en exigeant que l'énergie déposée dans LCAL et SICAL, plus l'énergie neutre dans un cône de 12 degrés autour du tube à faisceau soit inférieure à 5% de l'énergie centre de masse. Par ailleurs, l'énergie dans un secteur de 30 degrés (en ϕ) autour de P_{miss} (le "wedge") doit être inférieure à 5% de l'énergie centre de masse. Enfin, l'acolinéarité doit être supérieure à 90° .

Pour rejeter les événements $e^+e^- \rightarrow WW \rightarrow q\bar{q}l\nu_l$, aucun lepton identifié ne peut avoir une impulsion supérieure à $17\text{ GeV}/c$. La particule la plus isolée ne peut pas être un lepton ou avoir une énergie supérieure à 6.7 GeV .

Aucune coupure n'est appliquée sur la masse manquante, car cela introduirait une dépendance en l'hypothèse de masse du boson de Higgs.

Ces coupures sont optimisées par minimisation de la limite supérieure attendue à 95% de niveau de confiance sur la section efficace en l'absence de signal, telle que prédite par les simulations Monte Carlo, pour une masse $m_h = 100\text{ GeV}$.

Dans cette analyse, l'approche adoptée est appelée "modified frequentist confidence level method"[36]. Considérons le cas de n canaux indépendants de recherche ; on peut définir une statistique X discriminant les situations avec signal des situations avec pur bruit de fond. Le "rapport de vraisemblance" constitue un choix optimal pour cette statistique. Si le signal estimé dans le canal i est s_i , le bruit de fond b_i et le nombre de candidats observé d_i , le rapport de vraisemblance peut être écrit comme :

$$X = \prod_{i=1}^n X_i, \quad (3.10)$$

avec

$$X_i = \frac{e^{-(s_i+b_i)}(s_i + b_i)^{d_i}}{d_i!} \bigg/ \frac{e^{-b_i}b_i^{d_i}}{d_i!}. \quad (3.11)$$

Le niveau de confiance pour exclure la possibilité de présence simultanée du signal et du bruit de fond est

$$CL_{s+b} = P_{s+b}(X \leq X_{obs}), \quad (3.12)$$

c'est à dire la probabilité que la statistique soit inférieure ou égale à la valeur observée pour les données. Cette probabilité est la somme de distributions de Poisson

$$P_{s+b}(X \leq X_{obs}) = \sum_{X(\{d'_i\}) \leq X(\{d_i\})} \prod_{i=1}^n \frac{e^{-(s_i+b_i)}(s_i + b_i)^{d'_i}}{d'_i!}, \quad (3.13)$$

où $X(\{d_i\})$ est la statistique calculée pour les candidats observés $\{d_i\}$, et où la somme se fait sur toutes les situations possibles $\{d'_i\}$ pour lesquelles la statistique est inférieure ou égale à la statistique observée.

De façon à s'affranchir d'éventuelles fluctuations du bruit de fond, on calcule également le niveau de confiance pour le bruit de fond seul :

$$CL_b = P_b(X \leq X_{obs}), \quad (3.14)$$

où la somme de probabilité suppose la présence uniquement du bruit de fond :

$$P_b(X \leq X_{obs}) = \sum_{X(\{d'_i\}) \leq X(\{d_i\})} \prod_{i=1}^n \frac{e^{-b_i} b_i^{d'_i}}{d'_i!}. \quad (3.15)$$

Cette quantité est suggérée pour quantifier la confiance en une éventuelle découverte, quantifiant la probabilité que les processus de bruit de fond donnent moins de candidats qu'observé. Le niveau de confiance est alors obtenu par

$$CL_s = CL_{s+b}/CL_b. \quad (3.16)$$

On dira avoir atteint 95% C.L. si $CL = 1 - CL_s$ est supérieur à 0.95. Lors de l'optimisation, chaque entrée de l'histogramme représentant la masse manquante est considérée comme un canal indépendant. Le choix du "binning" de ce graphique est quelque peu arbitraire. Le choix adopté peut cependant être contrôlé a posteriori. La figure 3.8 présente ainsi la limite attendue $\langle CL_s \rangle$, valeur attendue de CL_s en l'absence de signal, en fonction du binning. On observe ainsi que pour un faible binning, les potentialités de la méthode ne sont pas bien exploitées. Pour des valeurs plus importantes de binning, on risque par ailleurs une perte de signification, liée au manque de statistique Monte Carlo. Le choix adopté s'avère bien être un bon compromis. Cette façon de procéder conduit à des coupures plus lâches que dans une analyse n'utilisant que le nombre de signaux, bruits de fond et données passant les coupures, car l'information disponible dans un histogramme permet de mieux discerner signal et bruit de fond.

Ces calculs sont facilités par l'usage du programme ECL, développé par Thomas Junk, pour le compte de l'expérience OPAL[36]. Ce programme calcule, pour une distribution donnée, le rapport de vraisemblance (3.10) ainsi que les probabilités (3.13) et (3.15), où la distribution des expériences possibles est évaluée à l'aide d'une PDF, qui peut être représentée comme une liste des situations possibles j pour les différents canaux $i : (X_i^j, p_i^j)$. On évite ainsi l'usage d'un calcul Monte Carlo lent et peu précis. Pour tenir compte des systématiques, la prescription de Cousins et Highlands est suivie : Dans les expressions (3.13) et (3.15), différentes quantités sont redéfinies en convoluant sur le signal et le bruit de fond :

$$(P_{s+b})_i \rightarrow \frac{\int_0^\infty ds' \int_0^\infty db' \frac{e^{-\left((s'-s_i)^2/2\sigma_{s_i}^2 + (b'-b_i)^2/2\sigma_{b_i}^2\right)} e^{-(s'+b')}(s'+b')^{d'_i}}{2\pi\sigma_{s_i}\sigma_{b_i}} \frac{d'_i!}{d'_i!}}{\int_0^\infty ds' \int_0^\infty db' \frac{e^{-\left((s'-s_i)^2/2\sigma_{s_i}^2 + (b'-b_i)^2/2\sigma_{b_i}^2\right)}}{2\pi\sigma_{s_i}\sigma_{b_i}}}, \quad (3.17)$$

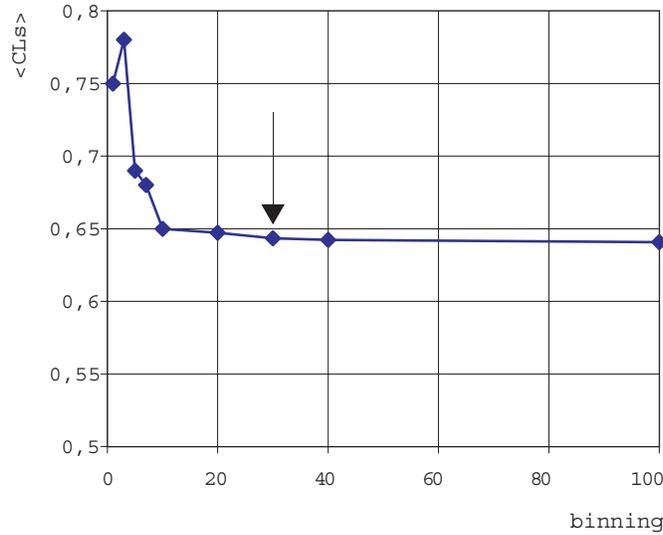


FIG. 3.8 – limite attendue $\langle CL_s \rangle$ en fonction du binning

$$(P_b)_i \rightarrow \frac{\int_0^\infty ds' \int_0^\infty db' \frac{e^{-\left(\frac{(b'-b_i)^2}{2\sigma_{b_i}^2}\right)} e^{-b' d'_i}}{2\pi\sigma_{b_i}} \frac{d'_i!}{d'_i!}}{\int_0^\infty ds' \int_0^\infty db' \frac{e^{-\left(\frac{(b'-b_i)^2}{2\sigma_{b_i}^2}\right)}}{2\pi\sigma_{b_i}}}, \quad (3.18)$$

avec la nouvelle statistique définie par

$$X_i(d'_i) \rightarrow \frac{\int_0^\infty ds' \int_0^\infty db' \frac{e^{-\left(\frac{(s'-s_i)^2}{2\sigma_{s_i}^2} + \frac{(b'-b_i)^2}{2\sigma_{b_i}^2}\right)}}{2\pi\sigma_{s_i}\sigma_{b_i}} X_i(d'_i)}{\int_0^\infty ds' \int_0^\infty db' \frac{e^{-\left(\frac{(s'-s_i)^2}{2\sigma_{s_i}^2} + \frac{(b'-b_i)^2}{2\sigma_{b_i}^2}\right)}}{2\pi\sigma_{s_i}\sigma_{b_i}}}. \quad (3.19)$$

Ces intégrales sont traitées numériquement par le programme.

Comme il a déjà été dit, les coupures sont optimisées itérativement de façon à maximiser la limite attendue sur la masse, pour l'hypothèse de masse la plus haute possible.

En partant de coupures grossières, un plot de la masse manquante est construit. Chaque bin est alors considéré comme une variable indépendante pour le calcul de $\langle CL_s \rangle$. On itère alors en modifiant une ou plusieurs coupures à chaque pas.

Cette procédure est répétée pour les différentes variables jusqu'à atteindre un minimum de $\langle CL_s \rangle$.

De cette optimisation, il ressort notamment qu'aucune coupure n'est appliquée sur y_{12} , paramètre permettant a priori de caractériser le nombre de jets, et donc de sélectionner les événements d'intérêt. La figure 3.9 montre par ailleurs qu'une telle coupure ne serait pas très discriminante.

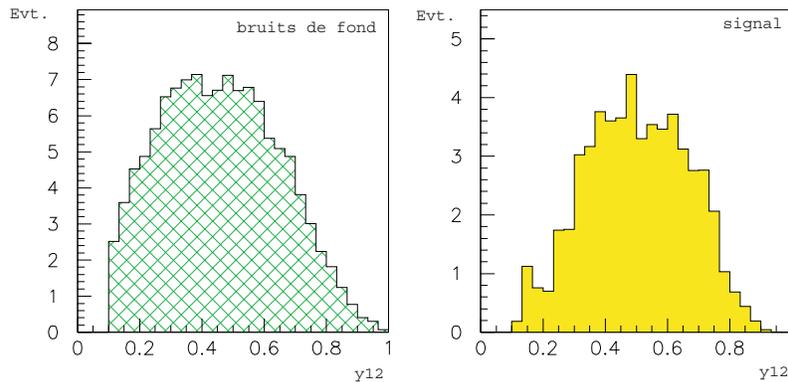


FIG. 3.9 – Distribution de y_{12} pour un signal de $90 \text{ GeV}/c^2$ et pour le bruit de fond.

La table 3.5 reprend les différentes coupures après optimisation. Par ailleurs, la table 3.6 reprend l'effet de ces différentes coupures sur les différents signaux. Notons que l'efficacité n'est pas fonction de l'hypothèse de masse pour le Higgs. Ceci est une conséquence du choix des coupures. Après ces coupures, on attend 53 événements de bruit de fond. Le signal attendu dépend fortement de l'hypothèse de masse, ainsi que de l'hypothèse sur ξ^2 . Pour ($m_h = 95 \text{ GeV}$, $\xi^2 = 1$), on attend 9 événements de signal. Dans les données, on observe 47 candidats.

3.8.2 Construction du réseau de neurones

De façon à tenir compte des éventuelles corrélations entre les variables discriminantes, un réseau de neurones a été construit dans le cadre de MLPfit[37] (intégré depuis peu à PAW). Une telle approche permet de rendre l'analyse plus performante.

Un réseau de neurones (figure 3.10) est une fonction non linéaire dans $f : \mathbb{R}^{n_1} \rightarrow \mathbb{R}^{n_2}$. Techniquement, c'est une structure composée d'un certain nombre de niveaux appelés "couches neuronales". La première couche constitue l'entrée du réseau, et reçoit les variables de départ, alors que la dernière couche produit le résultat. Le réseau utilisé ici ne comporte qu'un unique neurone de sortie.

A chaque neurone i est associée une variable x_i^n qui est une combinaison des

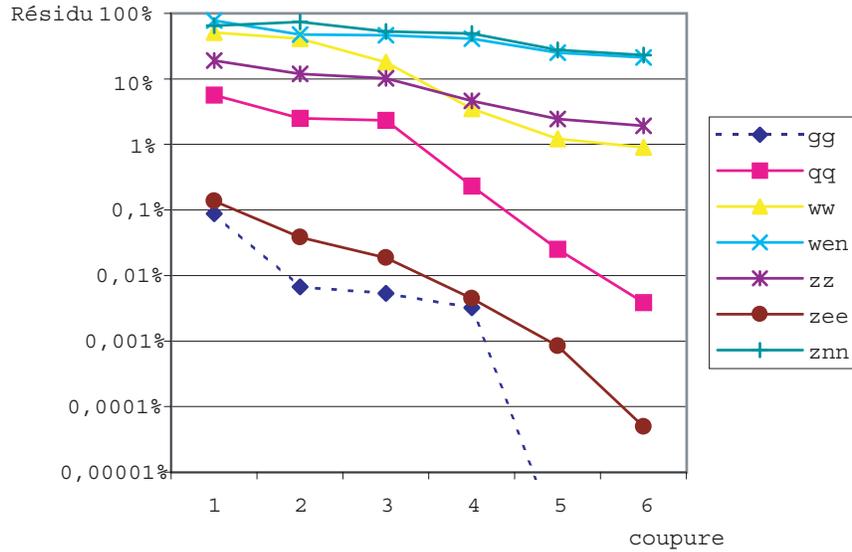
TAB. 3.5 – Coupures après optimisation à 189 GeV.

coupures anti-$\gamma\gamma$	
• nombre de particules chargées	> 9
• énergie des particules chargées	> 16%
• impulsion transverse manquante (P_{miss}^t)	> 0.02 E_{LEP}
• $ \cos(\overbrace{P_{miss}, z}) $	< 0.81
coupures anti-WW	
• impulsion maximale des leptons	< 0.09 E_{LEP}
• impulsion de la trace la plus isolée	< 6.7 GeV
• la trace la plus isolée ne peut être un muon	
coupures anti-$q\bar{q}$	
• masse invariante	$\in [0.19 E_{LEP}, 0.69 E_{LEP}]$
• énergie visible	$\in [0.31 E_{LEP}, 0.60 E_{LEP}]$
coupures topologiques	
• acoplanarité	< 174°
• acolinéarité	> 70°
3 jets sont forcés	
• cosinus du plus petit angle entre 2 jets	> 0.61
• masse invariante minimale de 2 jets	< 30 GeV
• les 3 jets doivent comporter une trace chargée	
coupures additionnelles anti-$\gamma\gamma$ et anti-retour radiatif	
• LCAL et SICAL énergie, plus énergie neutre dans un cône de 12 degrés autour du tube à faisceau	< 0.05 E_{LEP}
• énergie dans un secteur de 30 degrés (en ϕ) autour de P_{miss}^t	< 0.05 E_{LEP}

TAB. 3.6 – Effet des coupures à 189 GeV sur les bruits de fond. Les incertitudes sont prises égales à $1/\sqrt{N}$, où N est le nombre d'événements Monte Carlo passant les coupures.

Bruits de fond et données passant les coupures.

coupures	$\gamma\gamma$	$q\bar{q}$	WW	$W\ell\nu$	ZZ	Zee	$Z\nu\nu$	Données
précoupures	0.088%	6%	51%	77%	19%	0.137%	65%	2806
anti- $\gamma\gamma$	0.007%	2.5%	41%	48%	12%	0.038%	74%	1649
anti-WW	0,08%	5%	23%	75%	16%	0.074%	63%	1917
anti- $q\bar{q}$	0.02%	1.4%	8%	63%	7%	0.05%	61%	606
topologiques	0.014%	0.34%	5.5%	43%	4%	0.009%	36%	307
additionnelles	0.021%	1%	29%	57%	8%	0.017%	48%	1152
total	0%	0.004%	1%	21%	2%	0%	23%	47
incertitude	-	0.001%	0.02%	0.2%	0.1%	-	0.1%	-

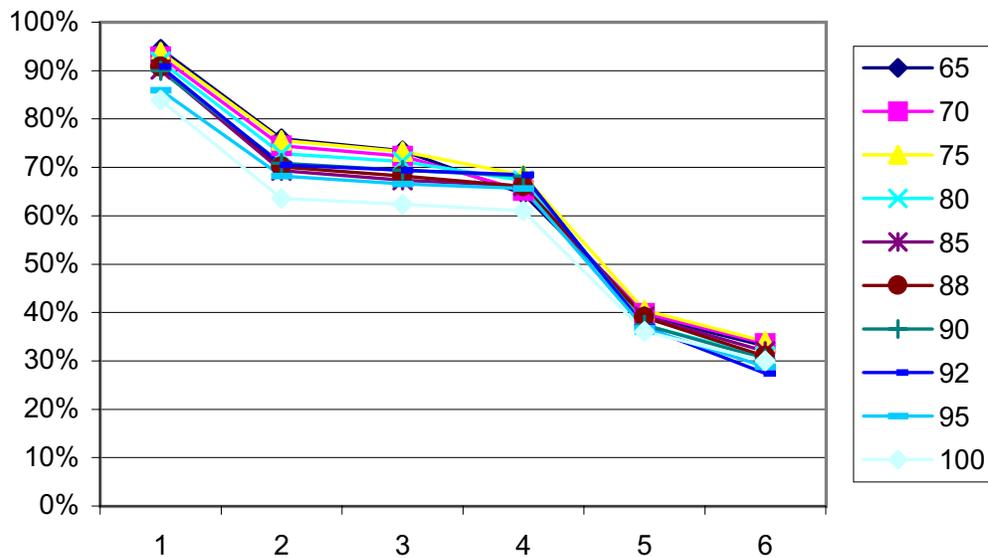


Bruits de fond passant les coupures (cumulatif).

TAB. 3.7 – Effet des coupures à 189 GeV sur le signal. Les incertitudes sont prises égales à $1/\sqrt{N}$, où N est le nombre d'événements Monte Carlo passant les coupures.

Signal passant les coupures. (cumulatif)

coupures	m_h									
	65	70	75	80	85	88	90	92	95	100
précoupures	94%	93%	94%	92%	90%	91%	90%	91%	86%	84%
anti- $\gamma\gamma$	76%	74%	76%	73%	69%	70%	71%	71%	68%	64%
anti-WW	73%	72%	73%	71%	67%	68%	69%	69%	67%	62%
anti- $q\bar{q}$	65%	65%	68%	67%	66%	66%	68%	68%	66%	61%
topologiques	39%	40%	40%	37%	39%	39%	37%	37%	37%	36%
additionnelles	33%	34%	34%	31%	32%	31%	30%	27%	29%	30%
incertitude	4%	4%	4%	4%	4%	4%	4%	4%	4%	4%



Signal passant les coupures (cumulatives) pour différentes masse du Higgs.

variables associées aux neurones de la couche précédente :

$$x_n^i = F(w_{ij}^{n-1} x_j^{n-1}),$$

avec $i = 1, N_n$ et $j = 1, N_{n-1}$ (une sommation sur les indices répétés est implicite) où N_n est le nombre de neurones sur la couche n , w_{ij}^n est un poids et F est la fonction neuronale. La fonction neuronale est selon les cas une fonction linéaire ou une sigmoïde $F(x) = 1/(1 + e^{-x})$. Les différents paramètres sont ajustés par rétro-propagation des erreurs lors d'une phase d'apprentissage. Soit tout d'abord la fonction de coût :

$$E = \frac{1}{2} \left((\zeta_1 - x_1^p)^2 + (\zeta_2 - x_2^p)^2 \right)$$

où ζ_i est la valeur idéale de la sortie (généralement 1 pour le signal et 0 pour le bruit de fond). Cette fonction rend compte de la distance entre les valeurs désirées (ζ_i) et calculées (x_i^p). Cette fonction est minimisée itérativement sur les poids.

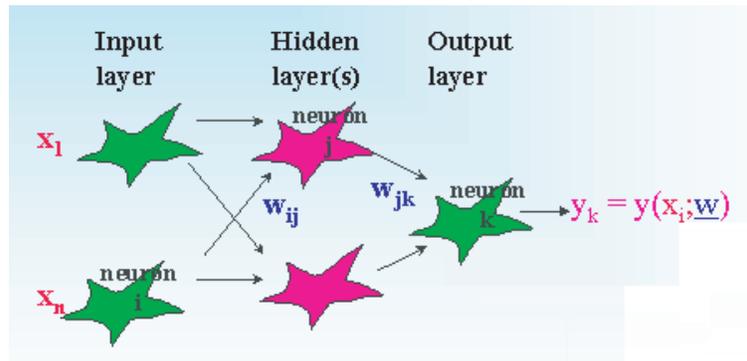


FIG. 3.10 – Structure d'un réseau de neurones.

L'architecture d'un réseau de neurones est donnée par le nombre de couches et le nombre de neurones dans chacune de ces couches. Le choix de l'architecture d'un réseau de neurones est quelque peu arbitraire. Dans le cadre de ce mémoire, différentes solutions ont été envisagées. Cinq types d'entrées ont été considérées :

1. masse invariante totale, énergie visible, impulsion de la trace la plus isolée et impulsion du plus énergétique lepton. (4 arguments)
2. acoplanarité, acolinéarité, y_{12} , angle minimum entre 2 jets et masse invariante minimale de deux jets parmi 3. (5 arguments)
3. masse invariante totale, énergie visible, acoplanarité, acolinéarité, y_{12} , angle minimum entre 2 jets et masse invariante minimale de deux jets parmi 3. (7 arguments)

4. masse invariante totale, énergie visible, acoplanarité, acolinéarité, angle minimum entre 2 jets et masse invariante minimale de deux jets parmi 3. (6 arguments)
5. masse invariante totale, énergie visible, acoplanarité, acolinéarité (4 arguments)

Pour chacun de ces cas, différentes architectures ont été considérées. Le choix s'est opéré en optimisant la limite sur la masse du Higgs comme précédemment. L'architecture retenue est 4-8-1 (4 neurones d'entrée : la masse invariante totale, l'énergie visible, l'acoplanarité et l'acolinéarité, une couche cachée de 8 neurones, et un unique neurone de sortie). Il a été entraîné pour produire 1 pour le signal et 0 pour les principaux bruits de fond passant les coupures (ZZ , $Z\nu\nu$, $W e\nu$ et WW). La figure 3.11 montre les courbes d'apprentissage de ce réseau de neurones pour une masse simulée m_h de 85 GeV^2 . On y voit la courbe d'apprentissage, valeur de la fonction de coût pour le sous-échantillon sur lequel est effectuée la minimisation, et la courbe de test, valeur de cette même fonction de coût pour un échantillon indépendant. L'apprentissage est arrêté lorsque ces deux courbes s'écartent l'une de l'autre, signe d'un risque de surdétermination du système. La figure 3.12 montre la sortie du réseau de neurones pour une masse de $100 \text{ GeV}/c^2$. Un tel réseau a été construit et entraîné pour chacune des 10 hypothèses de masse.

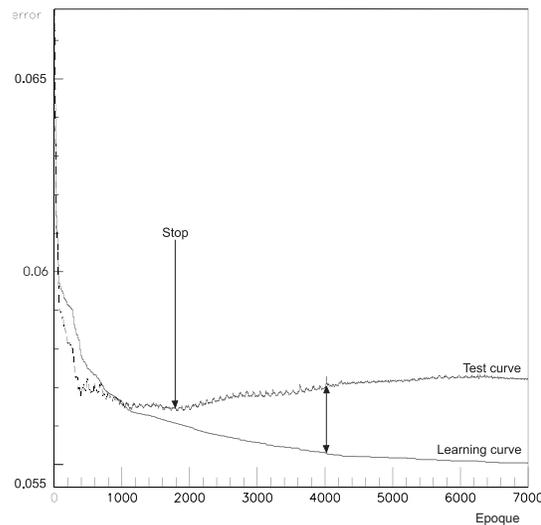


FIG. 3.11 – Courbe d'apprentissage du réseau de neurones pour une masse simulée m_h de $85 \text{ GeV}/c^2$.

Pour calculer la valeur finale du niveau de confiance, un diagramme bidimensionnel est construit, reprenant le nombre d'événements en fonction de la sortie du réseau de neurones et de la masse manquante (figure 3.13).

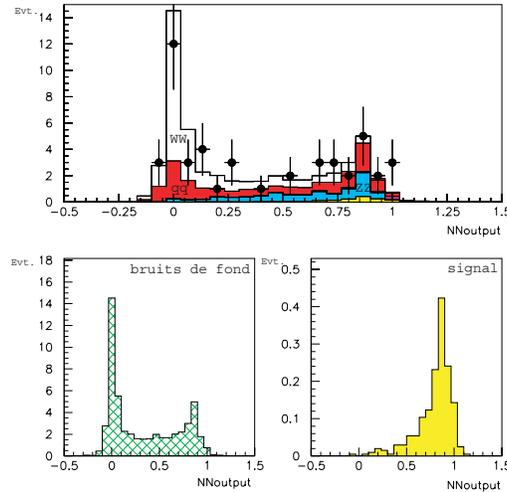


FIG. 3.12 – Sortie du réseau de neurones pour un boson de Higgs de $100 \text{ GeV}/c^2$.

Chaque entrée de ce diagramme est considérée comme un canal indépendant pour le calcul. Cette approche se révèle plus efficace que l'approche traditionnelle consistant à appliquer une coupure supplémentaire sur la sortie du réseau de neurones, en ce sens que la limite attendue est meilleure.

3.8.3 Erreurs systématiques et biais

Lors de l'obtention du résultat final, le programme ECL permet alors de tenir compte de des erreurs systématiques. Les différentes sources d'erreur systématiques que nous avons considérées sont :

- les imprécisions des différents générateurs Monte Carlo
- les incertitudes sur la calibration en énergie du détecteur
- les incertitudes sur la calibration angulaire du détecteur
- l'incertitude introduite par la simulation de E12 (énergie dans un cône de 12° autour du tube à faisceau)

De précédentes études[38] ont montré que l'erreur due au détecteur est inférieure aux erreurs statistiques. Les moteurs Monte Carlo utilisés ici sont par ailleurs de très bonne précision, de sorte que la principale erreur provient de la simulation WW par KORALW. Nous estimons ainsi les incertitudes à 5% sur le signal et 15% sur le bruit de fond. Bien qu'une analyse poussée soit nécessaire si l'on désire une estimation précise de ces erreurs systématiques, la méthode d'analyse s'avère particulièrement stable par rapport à ces dernières (la limite attendue pour CL_s ne varie que de 1% en considérant 30% d'incertitude sur le bruit de fond. Cet aspect n'est de plus pas critique en l'absence d'un signal dans les données.

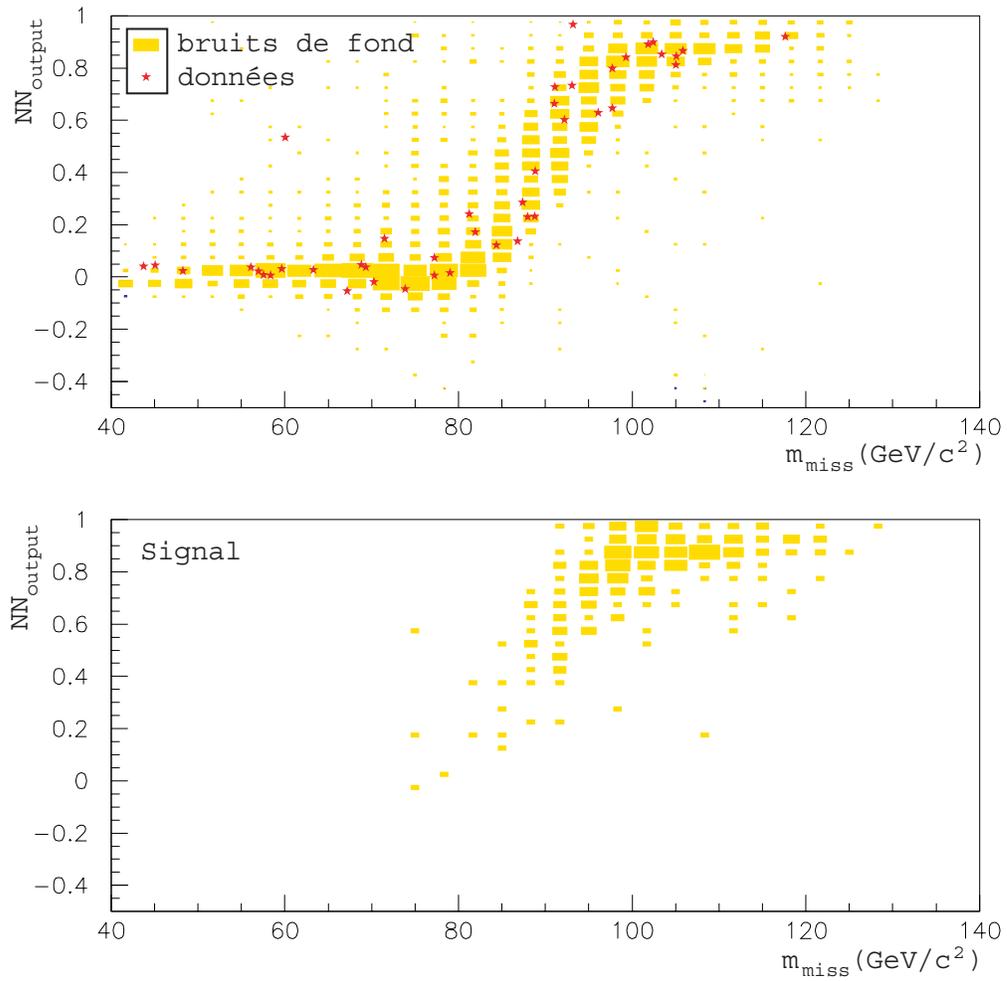


FIG. 3.13 – Représentation bidimensionnelle reprenant le nombre d'événements en fonction de la sortie du réseau de neurones et de la masse manquante pour un boson de Higgs de $100 \text{ GeV}/c^2$ à une énergie de 189 GeV .

Notons que le fait de ne pas utiliser l'identification des quarks b supprime une importante source d'erreurs systématiques.

Il est important de considérer également les différents risques de biais dans l'analyse, afin de s'assurer de leur maîtrise.

Une source de biais potentiel vient de l'absence d'échantillon de contrôle lors de l'optimisation des coupures. La limitation statistique nous a en effet poussé à utiliser les mêmes échantillons pour optimiser les coupures et pour estimer la limite finale. On risque ainsi d'obtenir un biais dû à un ajustement critique des coupures.

Cependant, l'optimisation sur une distribution de m_{miss} conduit à des coupures relativement lâches. La limite obtenue présente par ailleurs une bonne stabilité par rapport aux coupures autour du minimum atteint.

Une mesure de la différence des distributions de la sortie du réseau de neurones pour l'échantillon d'apprentissage et l'échantillon de test nous est fournie par le test de Kolmogorov. La valeur de 3% de probabilité de similitude confirme le danger à utiliser l'échantillon d'apprentissage pour l'analyse des données. Dans cette analyse, seul l'échantillon de test a été utilisé pour produire les résultats, ce qui nous met à l'abri de cette source d'erreurs.

Par ailleurs, l'information Monte Carlo des événements simulés passant les coupures a été contrôlée. Les événements de bruits de fond correspondent aux topologies attendues, de sorte que l'incertitude prévue sur la simulation reste acceptable. Par exemple, les événements WW conservés correspondent aux événements $e^+e^- \rightarrow WW \rightarrow q\bar{q}\tau\nu_\tau$ avec un τ peu énergétique, pouvant être assimilé à un gluon mou par l'algorithme de Durham lors de la reconstruction des jets. La figure 3.14 présente ainsi l'énergie et l'angle que fait le tau avec les deux jets avant et après sélection.

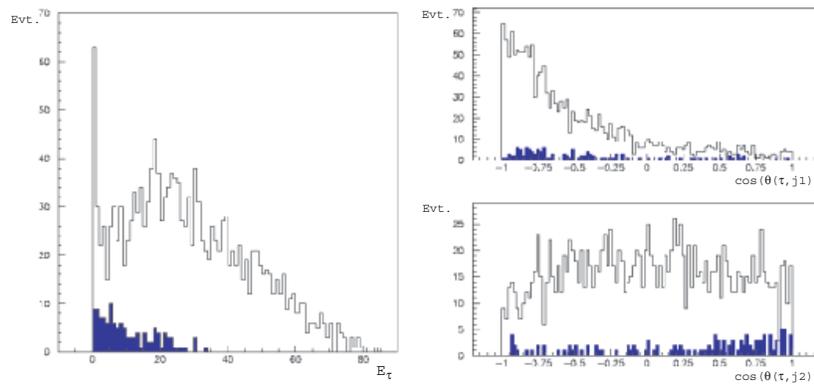


FIG. 3.14 – Energie du tau et angle avec chacun des jets dans les processus $e^+e^- \rightarrow WW \rightarrow q\bar{q}\tau\nu_\tau$ avant (blanc) et après (noir) sélection. On conserve les tau peu énergétiques, alors que la distribution angulaire est relativement uniforme.

3.8.4 Résultats

$\langle CL_s \rangle$ et CL_s ont été évalués pour les 10 hypothèses de masse, ainsi que pour 10 valeurs de ξ^2 (de 0.1 à 1 par pas de 0.1).

Pour $\xi^2 = 1$, la limite obtenue à 95% C.L. sur la masse du boson de Higgs est de $95.2 \text{ GeV}/c^2$. La limite attendue étant de $95.2 \text{ GeV}/c^2$. Ce résultat peut être comparé à celui obtenu par de précédents travaux à ALEPH à 189 GeV ($95.4 \text{ GeV}/c^2$, pour une limite attendue de $94.4 \text{ GeV}/c^2$.) en utilisant une reconnaissance des b-jets pour raffiner l'analyse, et en prenant en compte les canaux leptoniques du Z. L'analyse réalisée est donc tout aussi performante (si pas plus) que l'analyse officielle ALEPH.

La figure 3.15a montre l'évolution du niveau de confiance en fonction de la masse du candidat. On y voit la limite obtenue à 95% C.L. pour $\xi^2 = 1$. La figure 3.15b représente la valeur obtenue pour CL_b ; on note un déficit d'événements par rapport au modèle standard, maximum pour une hypothèse de masse proche de 85 GeV . Ceci se répercute sur la limite obtenue dans ces régions, meilleure qu'attendue. Si aucun signal n'est présent, il faut s'attendre à ce que cette limite soit réduite si la statistique augmente.

Les résultats peuvent également être interprétés comme une zone d'exclusion dans le plan (m_h, ξ^2) . Cette zone d'exclusion est représentée à la figure 3.16.

3.8.5 Etude des topologies sélectionnées

Les coupures ayant été optimisées de façon systématique et sans biais visuel éventuel, il est intéressant de regarder les différentes topologies sélectionnées avec un bon rapport signal/bruit.

Différentes topologies sont retenues. On peut grossièrement les classer comme suit :

1. les topologies à grande impulsion transverse
2. les topologies à deux jets
3. les topologies à trois jets

Sur les 12 événements passant les coupures et ayant une valeur de sortie du réseau de neurones supérieure à 0.8, nous avons 10 événements à 2 jets, 1 événement à "grande" impulsion transverse et 1 événement à trois jets apparents. Il est important de remarquer que l'on ne doit pas attendre que le réseau de neurones puisse discriminer ces événements, vu les variables utilisées. Par ailleurs, aucune de ces topologies n'est incompatible avec l'hypothèse de signal, ce qui justifie a posteriori qu'elles n'aient pas été retenues dans le choix des variables d'entrées du réseau de neurones. Ces événements ont le meilleur rapport signal sur bruit (atteignant 0.2).

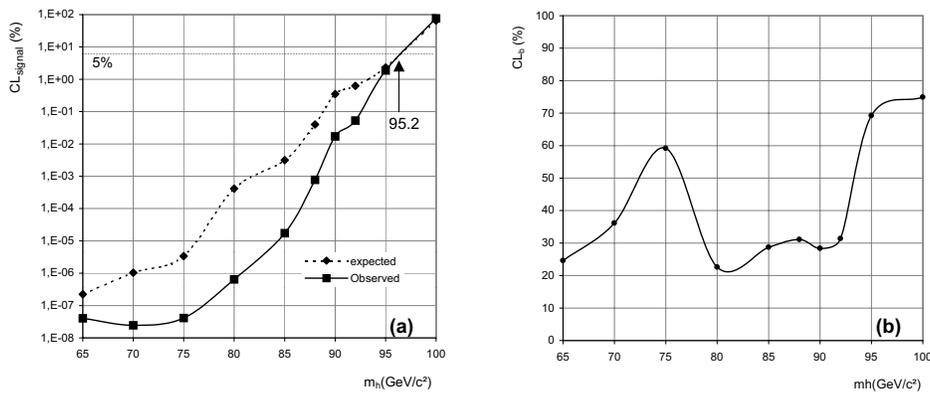


FIG. 3.15 – Niveau de confiance sur le signal, CL_s , (a) et sur le bruit de fond uniquement, CL_b , (b) obtenus à 189 GeV.

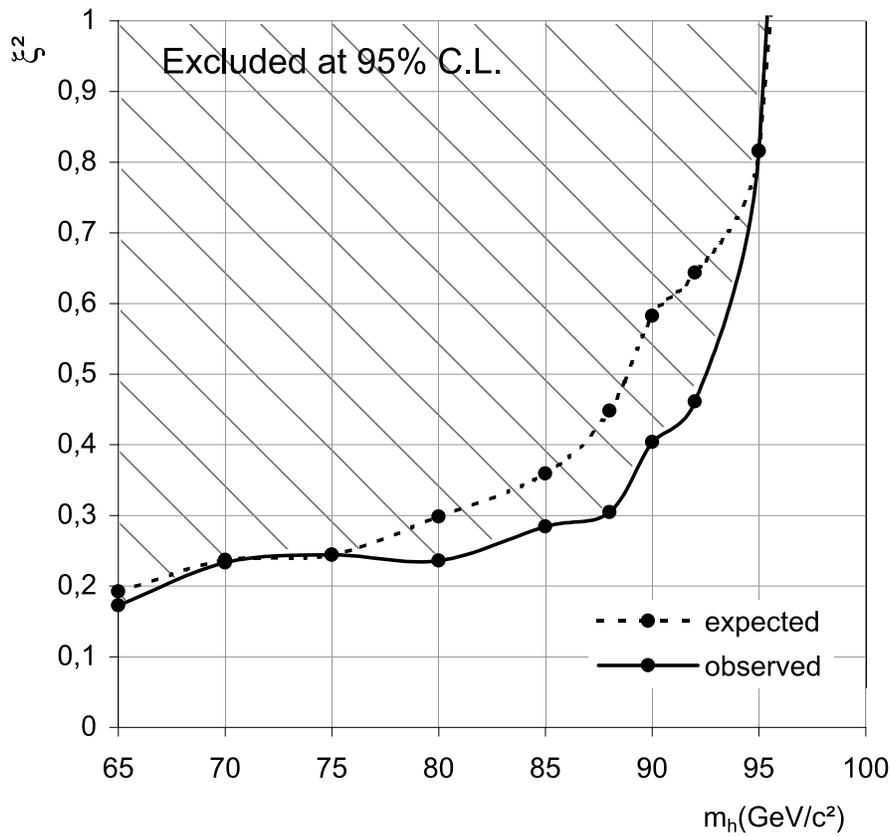
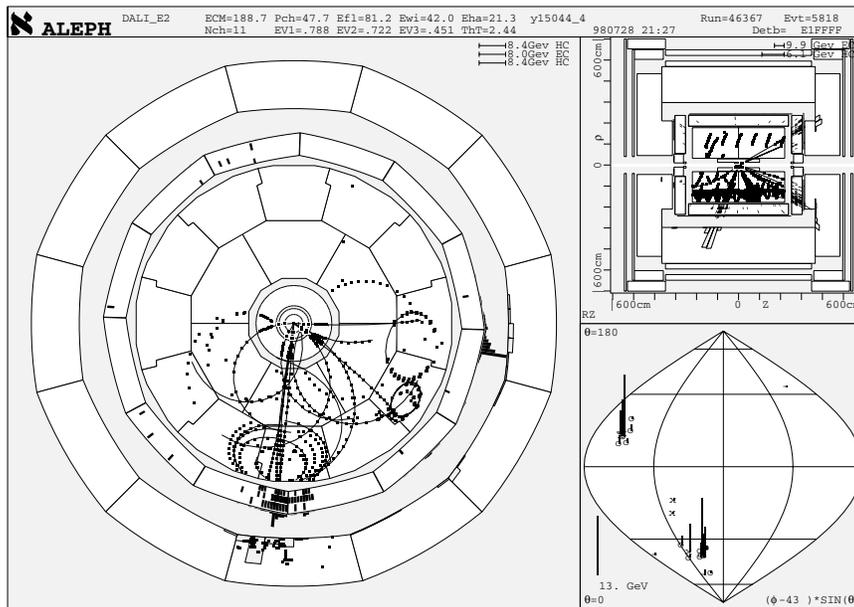


FIG. 3.16 – Région exclue à 95% C.L. dans le plan (m_h, ξ^2) (analyse à 189 GeV).

l'événement à fort Pt_{miss}

NNoutput	1.03
s/b	0.2
Pt_{miss}	44 GeV/c
y_{12}	0.55

FIG. 3.17 – Événement à fort Pt_{miss}

Bien que ce type d'événement ne corresponde pas à la topologie type recherchée (un Higgs près du seuil ne devant pas emporter une grande impulsion transverse), il n'est pas en désaccord avec le signal, d'autant plus si l'on s'intéresse à des masses plus faibles.

La figure 3.18 montre la distribution de pt_{miss} attendue pour différentes masses du boson de Higgs. Remarquons que le pic de signal pour $m_H = 65 \text{ GeV}$ se trouve précisément à $44 \text{ GeV}/C^2$. Une coupure sur Pt_{miss} est donc dangereuse.

Par ailleurs, la représentation angulaire met bien en évidence deux jets, ce qui n'est pas clair sur les autre vues.

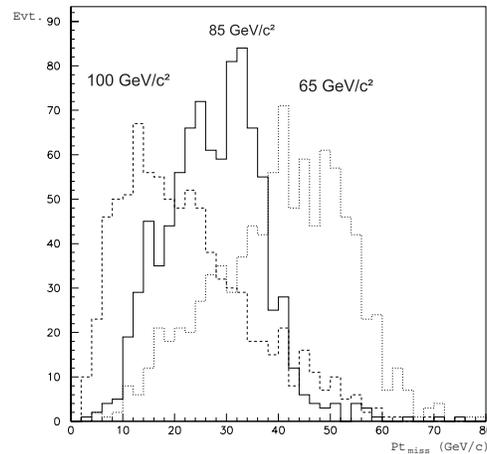


FIG. 3.18 – Impulsion transverse manquante pour 3 hypothèses de signal

L'événement à 3 jets

NNoutput	1.006
s/b	0.2
y_{23}	0.011
mjet	$14 \text{ GeV}/c^2$
intjet	28.7°

L'événement *semble* ici correspondre clairement à une topologie à trois jets. Les coupures sensées rejeter ces topologies sont $mjet < 33$ et $intjets > 0.61$. De toute évidence, nous sommes loin de ces limites.

En regardant de plus près cet événement, on note également que le troisième jet est relativement mou. Une mesure de la séparation des jets en tenant compte de la métrique de Durham (utilisée dans l'analyse) est donnée par y_{23} , valeur de y_{cut} pour laquelle la topologie passe de 2 à 3 jets. Une faible valeur de y_{23} nous indique que ce troisième jet est en effet très rapidement regroupé avec le jet énergétique le plus proche. Il peut, par exemple, s'agir ici d'un gluon mou émis à grand angle.

les événements à 2 jets

Pour illustrer la topologie attendue, considérons l'événement représenté à la figure 3.20.

NNoutput	0.95
s/b	0.18

On observe clairement deux jets relativement colinéaires (acolinéarité de 174°) mais acoplanaires (acoplanarité de 172°).

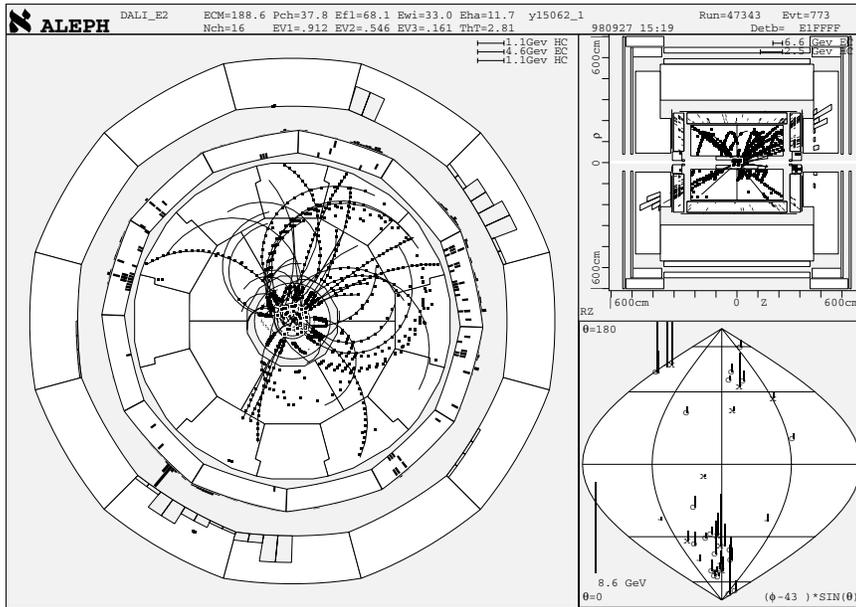


FIG. 3.19 – Evénement à trois jets

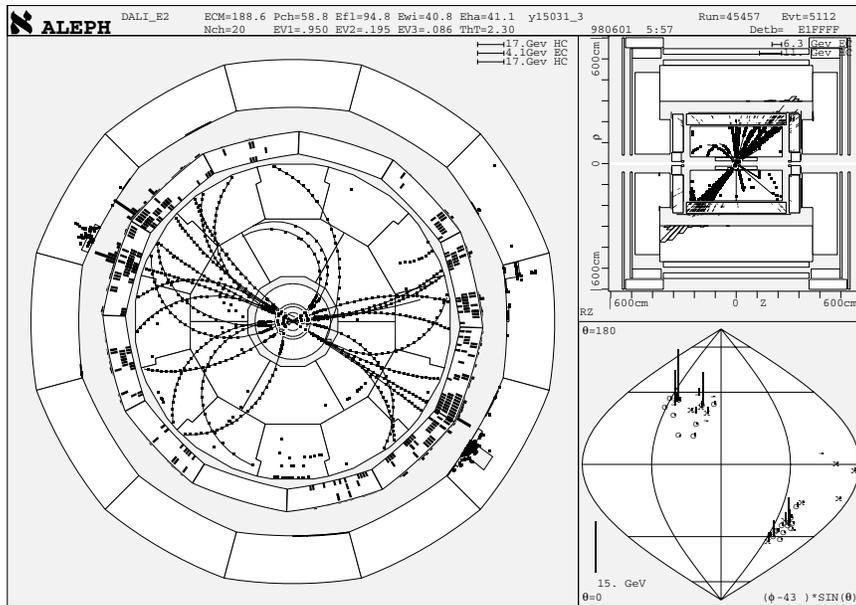


FIG. 3.20 – Evénement à deux jets

3.9 Analyse à 206 GeV

L'approche adoptée ici est similaire à celle qui a été suivie à 189 GeV. De façon à améliorer encore la limite supérieure attendue à 95% de niveau de confiance sur la section efficace en l'absence de signal, telle que prédite par les simulations Monte Carlo, on définit la masse invisible comme

$$m_{invis} = \sqrt{\left(E_{LEP} - \frac{m_Z E_{sum}}{M_{sum}}\right)^2 - \left(\frac{m_Z P_{miss}}{M_{sum}}\right)^2}$$

où E_{LEP} est l'énergie dans le centre de masse, E_{sum} et M_{sum} sont respectivement l'énergie et la masse visibles, m_Z est la masse du Z et P_{miss} l'impulsion transverse manquante.

Les différentes coupures sont optimisées en utilisant chaque entrée de l'histogramme de la masse invisible (à 189 GeV, la masse manquante était considérée).

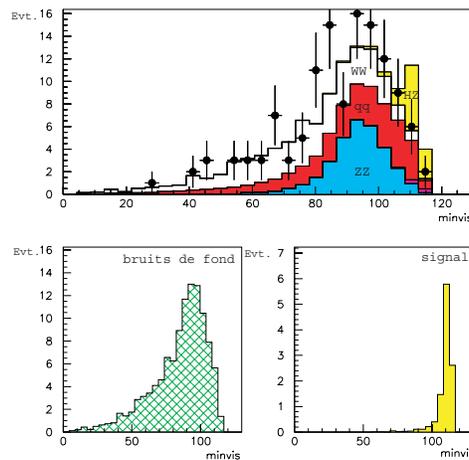


FIG. 3.21 – Distribution de la masse invisible pour un signal de $112 \text{ GeV}/c^2$ et le bruit de fond.

La figure 3.21 reprend la distribution de m_{invis} pour un Higgs de $112 \text{ GeV}/c^2$, comparée au bruit de fond attendu après toutes les coupures mentionnées ci-après.

Les coupures finales obtenues sont présentées dans la table 3.8. Notons la similitude avec les coupures obtenues à 189 GeV. La table 3.9 reprend l'effet de ces coupures sur les différents signaux. Quelques graphiques de contrôle, à diverses étapes de la sélection, sont repris à la figure 3.23. Après ces coupures, on attend 119 événements de bruit de fond. Le signal attendu dépend fortement de l'hypothèse de masse et de la valeur de ξ^2 . Pour ($m_h = 112 \text{ GeV}$, $\xi^2 = 1$) on s'attend à 10.5 événements. On observe 134 candidats dans les données. La figure 3.22 montre par ailleurs que ce léger excès n'est pas localisé à une énergie particulière.

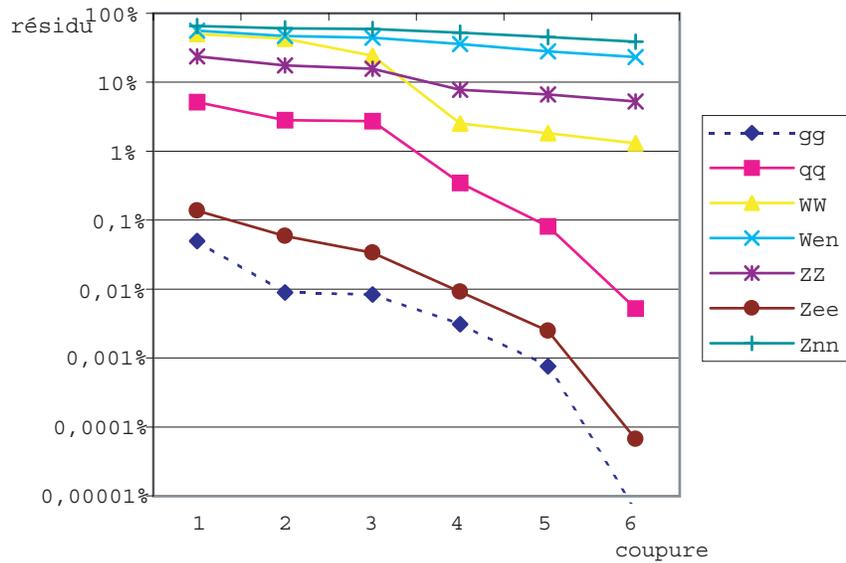
TAB. 3.8 – Coupures après optimisation à 206 GeV.

coupures anti-$\gamma\gamma$	
• nombre de particules chargées	> 9
• énergie des particules chargées	> 10%
• impulsion transverse manquante ($P_{t_{miss}}$)	> 3.5 % de E_{LEP}
• $ \cos(\overbrace{P_{miss}, z}) $	< 0.9
coupures anti-WW	
• impulsion maximale des leptons	< 7% de E_{LEP}
• impulsion de la trace la plus isolée	< 3% de E_{LEP}
• la trace la plus isolée ne peut être un muon ou un électron	
coupures anti-$q\bar{q}$	
• masse invariante	$\in [25\% \text{ de } E_{LEP}, 51\% \text{ de } E_{LEP}]$
• énergie visible	$\in [30\% \text{ de } E_{LEP}, 55\% \text{ de } E_{LEP}]$
coupures topologiques	
• acoplanarité	< 173°
• acolinéarité	> 95°
• valeur de y_{cut} lors du passage de 3 à 2 jets (y_{12})	> 0.1
3 jets sont forcés	
• masse invariante minimale de 2 jets parmi 3	< 44 GeV
• les 3 jets doivent comporter une trace chargée	
coupures additionnelles anti-$\gamma\gamma$ et retour radiatif	
• LCAL et SICAL énergie, plus énergie neutre dans un cône de 12 degrés autour du tube à faisceau	< 2.1% de E_{LEP}
• énergie dans un secteur de 30 degrés (en ϕ) autour de $P_{t_{miss}}$	< 5% de E_{LEP}

TAB. 3.9 – Effet des coupures à 204 GeV et plus sur les bruits de fond. Les incertitudes sont prises égales à $1/\sqrt{N}$, où N est le nombre d'événements Monte Carlo passant les coupures.

Bruits de fond et données passant les coupures.

coupures	$\gamma\gamma$	$q\bar{q}$	WW	$W e\nu$	ZZ	Zee	$Z\nu\nu$	Données
précoupures	0.05%	5%	50%	56%	23%	0.137%	66%	3314
anti- $\gamma\gamma$	0.009%	2.8%	43%	47%	18%	0.06%	61%	2301
anti-WW	0.04%	5%	29%	53%	21%	0.075%	64%	2414
anti- $q\bar{q}$	0.01%	1%	4%	43%	9%	0.04%	58%	530
topologiques	0.009%	0.57%	13%	38%	9%	0.02%	55%	864
additionnelles	0.011%	0.89%	28%	40%	11%	0.015%	48%	1431
total	0%	0.005%	1%	23%	5%	0.0001%	39%	134
incertitude	-	0.0004%	0.02%	0.2%	0.05%	0.0001%	0.4%	-

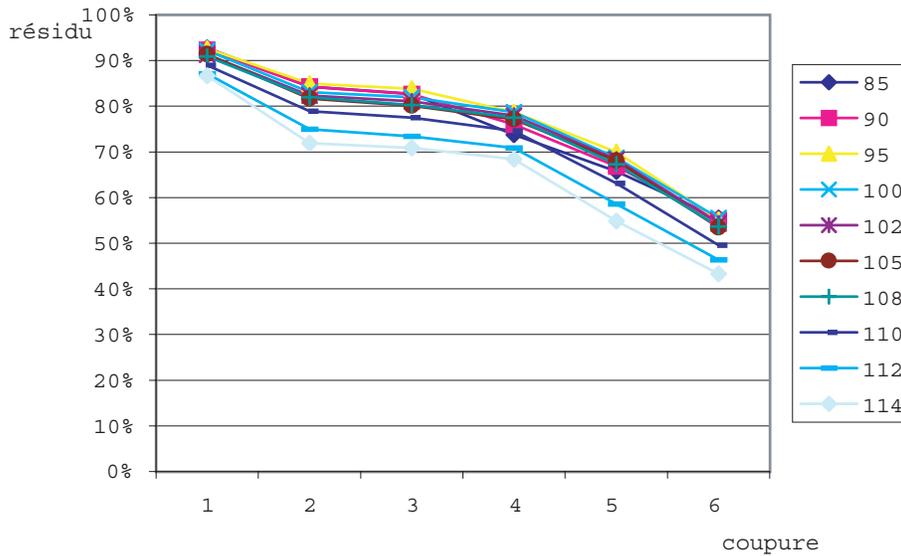


Bruits de fond passant les coupures (cumulatif).

TAB. 3.10 – Effet des coupures à 204 GeV et plus sur le signal. Les incertitudes sont prises égales à $1/\sqrt{N}$, où N est le nombre d'événements Monte Carlo passant les coupures.

Signal passant les coupures. (cumulatif)

coupures	m_h									
	85	90	95	100	102	105	108	110	112	114
précoupures	93%	92%	93%	92%	91%	91%	90%	89%	87%	87%
anti- $\gamma\gamma$	84%	84%	85%	83%	82%	82%	82%	79%	75%	72%
anti-WW	83%	83%	84%	82%	81%	80%	80%	78%	73%	71%
anti- $q\bar{q}$	74%	76%	79%	79%	78%	77%	77%	75%	71%	68%
topologiques	66%	67%	70%	69%	68%	68%	67%	63%	59%	55%
additionnelles	56%	55%	55%	56%	54%	53%	54%	50%	46%	43%
incertitude	2%	2%	2%	2%	2%	2%	2%	2%	2%	3%



Signal passant les coupures (cumulatives) pour différentes masse du Higgs.

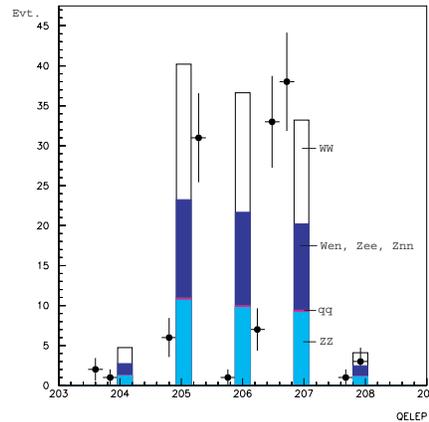


FIG. 3.22 – Distribution du nombre d'événements attendus (histogrammes) et des données retenues par les coupures en fonction de l'énergie centre de masse. Les contributions au bruit de fond sont (de bas en haut) : ZZ, $W_{e\nu}$, et WW. Notons encore la très faible quantité de processus $q\bar{q}$ (invisibles sur ce plot) passant les coupures.

Un réseau de neurones a été construit de façon à tenir compte des corrélations entre variables, et à obtenir une meilleure limite sur la masse. Tout comme précédemment, la structure adoptée est 4-8-1 (4 neurones d'entrée : la masse invariante totale, l'énergie visible, l'acoplanarité et l'acolinéarité, une couche cachée de 8 neurones, et un unique neurone de sortie). Le réseau a été entraîné pour produire 1 pour le signal et 0 pour les principaux bruits de fond passant les coupures (ZZ, $Z\nu\nu$, $W_{e\nu}$ et WW). La figure 3.23 montre la sortie de ce réseau de neurones pour une masse m_h simulée de $114 \text{ GeV}/c^2$. Un tel réseau a été construit et entraîné pour chacune des 10 hypothèses de masse.

Pour calculer la valeur finale du niveau de confiance, un diagramme bidimensionnel (figure 3.24) est construit, reprenant le nombre d'événements en fonction de la masse manquante. Chaque entrée de ce diagramme est considérée comme un canal indépendant pour le calcul.

Cette analyse nous permet de rejeter tout boson de Higgs invisible de masse inférieure à 112.15 GeV (pour $\xi^2 = 1$), la limite attendue étant de 112.19 GeV . Les résultats obtenus sont résumés à la figure 3.25. On notera, quel que soit l'hypothèse de masse, l'importante valeur de CL_b , traduisant l'excès global observé dans les données. Ils peuvent également être interprétés comme une zone d'exclusion dans le plan (m_h, ξ^2) . Cette zone d'exclusion est présentée à la figure 3.26.

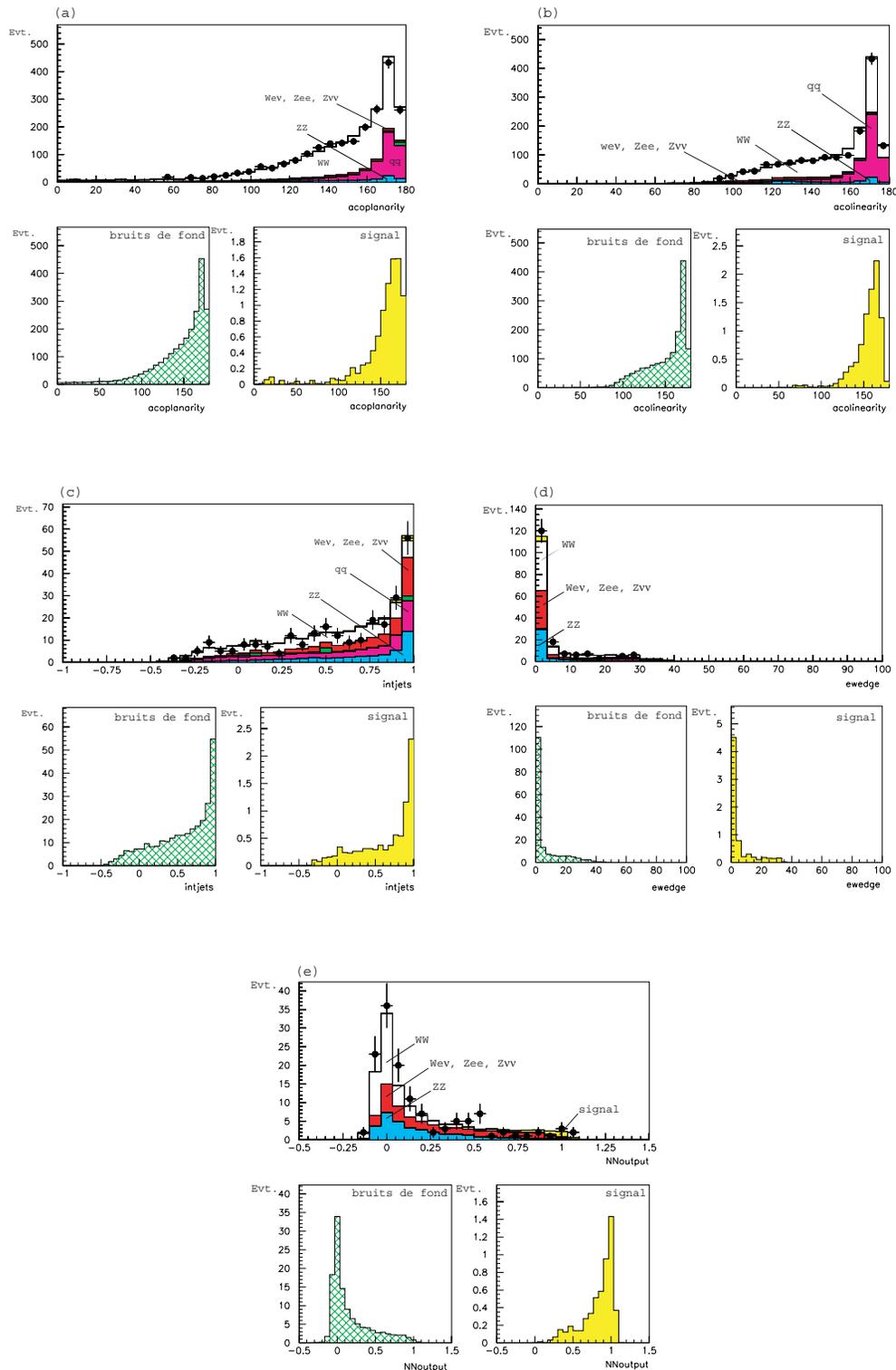


FIG. 3.23 – Différentes grandeurs au cours de la sélection (coupures cumulatives) : (a) acoplanarité après les coupures anti- $\gamma\gamma$; (b) acolinéarité après les coupures anti-WW; (c) angle minimal entre 2 jets parmi 3 après les coupures anti- $q\bar{q}$; (d) énergie dans le “wedge” après les coupures topologiques; (e) distribution de sortie pour le réseau de neurones pour un signal de $114 \text{ GeV}/c^2$ après toutes les coupures.

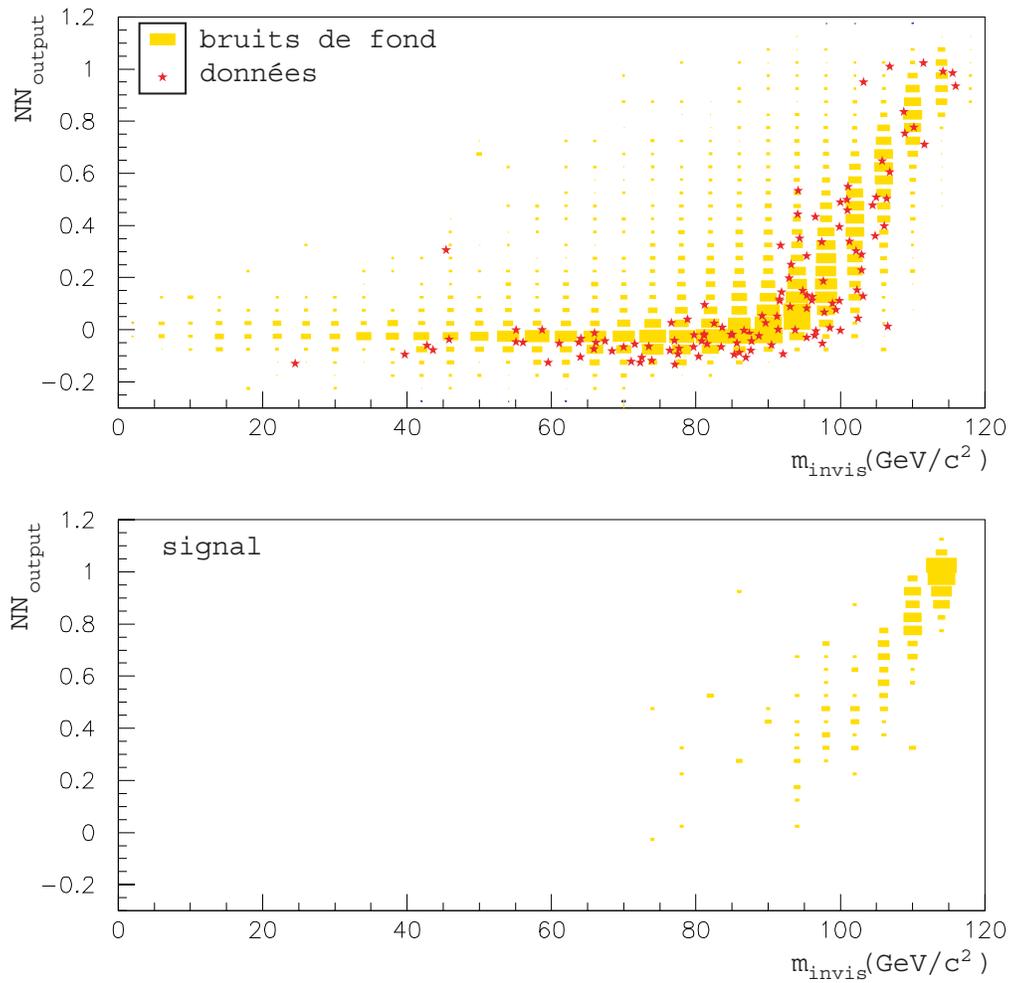


FIG. 3.24 – Représentation bidimensionnelle reprenant le nombre d'événements en fonction de la sortie du réseau de neurones et de la masse manquante pour un boson de Higgs de $114 \text{ GeV}/c^2$ à une énergie de 206 GeV .

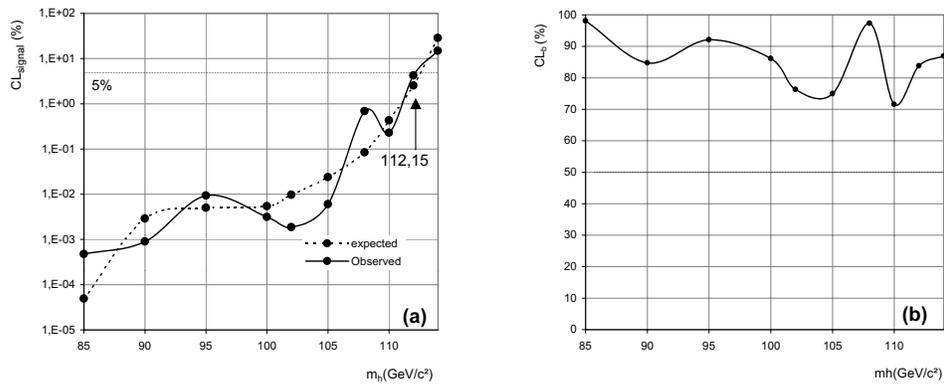


FIG. 3.25 – Niveau de confiance sur le signal, CL_s , (a) et sur le bruit de fond uniquement, CL_b , (b) obtenus à plus de 204 GeV .

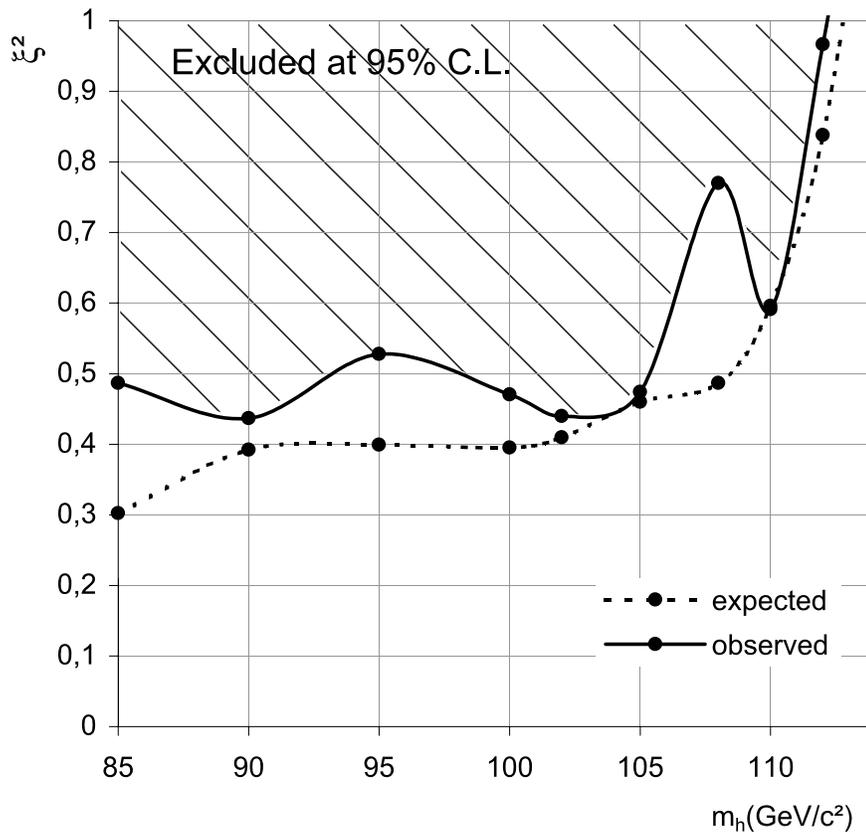


FIG. 3.26 – Région exclue à 95% C.L. dans le plan (m_h, ξ^2) (analyse à plus de 204 GeV).

TAB. 3.11 – Quelques candidats à $m_h = 114 \text{ GeV}$

candidat	Energie	Acolin.	Acopl.	M_{sum} (GeV/c^2)	M_{invis} (GeV/c^2)	s/b	s	b
1	206.8	161°	159°	94.8	113	2.85	1.85	0.275
2	207.7	158°	164°	92.56	114.965	2.85	1.85	0.275
3	205.2	160°	168°	85.6	112.2	1.85	0.816	0.304
4	207.0	153°	172°	98	111.1	0.34	0.975	0.3395
5	205.2	147°	170°	69.3	109.95	0.34	0.975	0.3395

La sortie du réseau de neurones pour un candidat de $114 \text{ GeV}/c^2$ (figure 3.23) présente un excès intéressant pour les grandes valeurs de sortie. Cet excès est de plus confirmé sur le diagramme bidimensionnel reprenant le nombre d'événements en fonction de la sortie du réseau de neurones et de la masse manquante (figure 3.24). Les 5 meilleurs candidats sont présentés aux figures 3.27 à 3.31. On notera la grande valeur de s/b pour le premier. Comme attendu, ces événements présentent deux jets relativement colinéaires mais acoplanaires. La table 3.11 reprend quelques paramètres pour ces événements. Notons que les topologies rencontrées sont similaires à celles présentées à 189 GeV . L'analyse statistique montre cependant que l'excès n'est pas significatif.

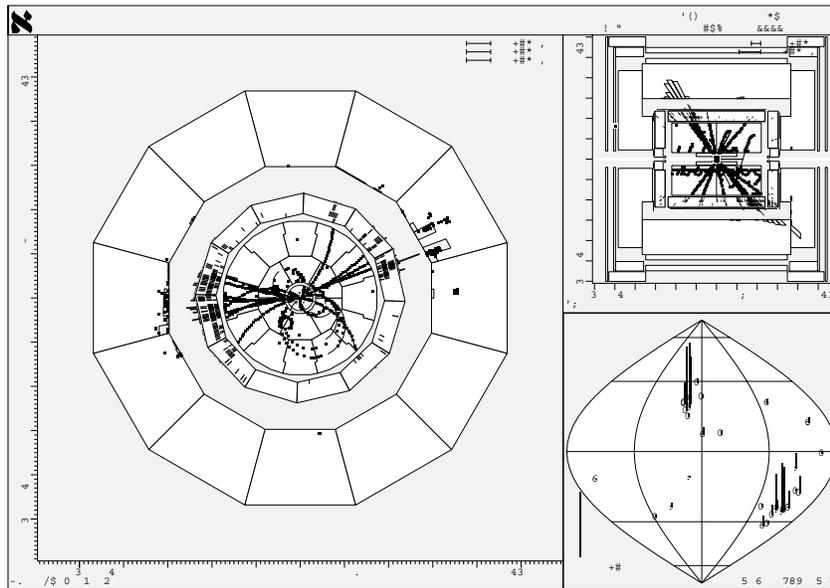


FIG. 3.27 – Candidat 1

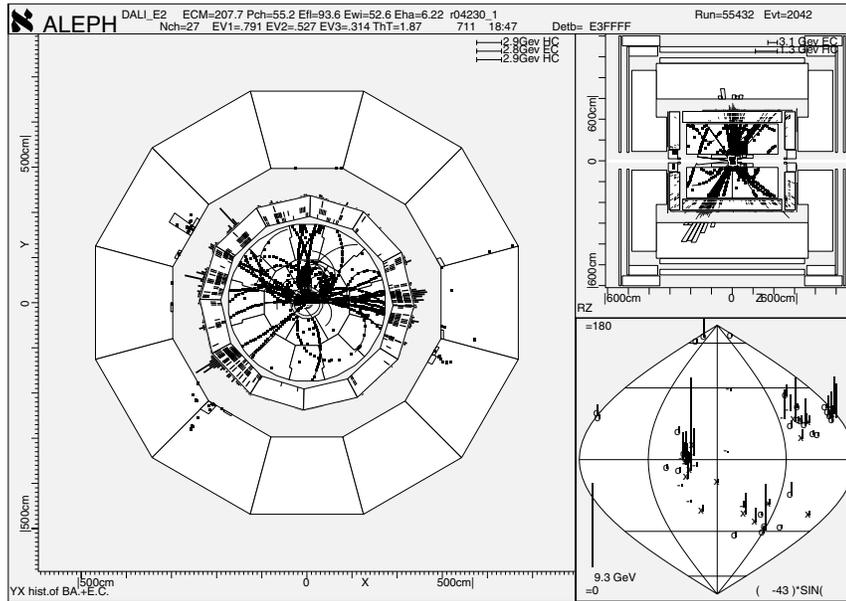


FIG. 3.27– Candidat2

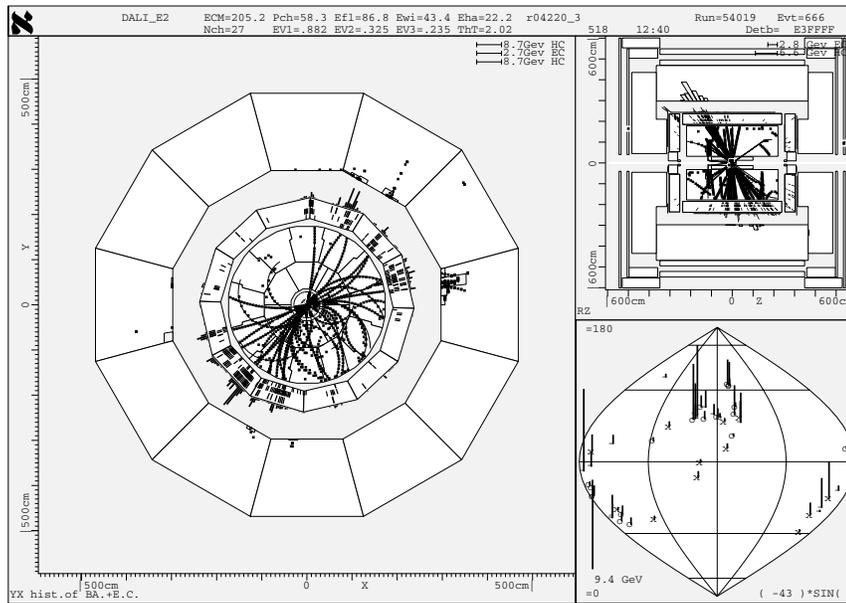


FIG. 3.28– Candidat3

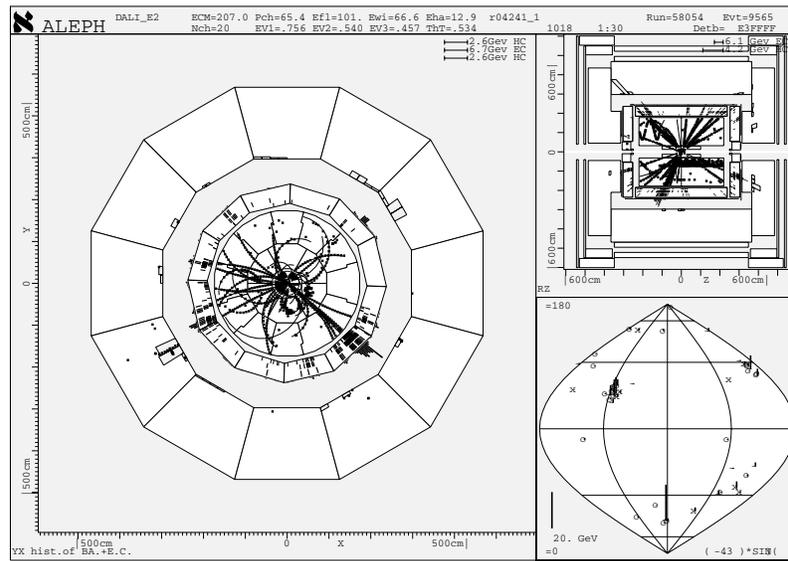


FIG. 3.29– Candidat4

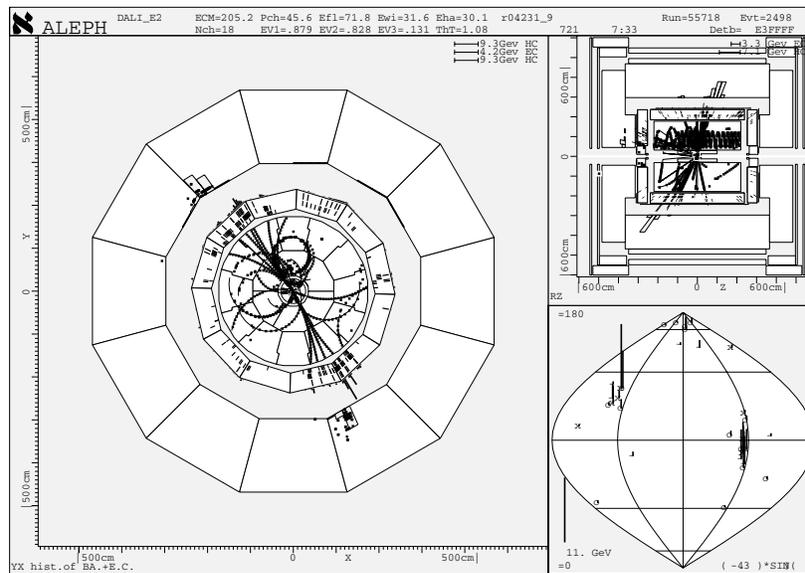


FIG. 3.30– Candidat5

3.10 Combinaison des deux analyses

En combinant les deux analyses, il est possible d'améliorer la limite dans la zone de masse du Higgs où il y a chevauchement. Ceci est principalement dû au gain de statistique. Pour cela, les diagrammes biparamétriques de $N_{event}(NN_{output}, m_{miss})$ obtenus dans chacune des deux analyses sont concaténés de façon à considérer simultanément l'entière des mesures pour le calcul des limites. Cette combinaison des analyses est possible pour les masses communes aux deux analyses (de 85 GeV à 100 GeV). La zone d'exclusion finale obtenue est présentée figure 3.32.

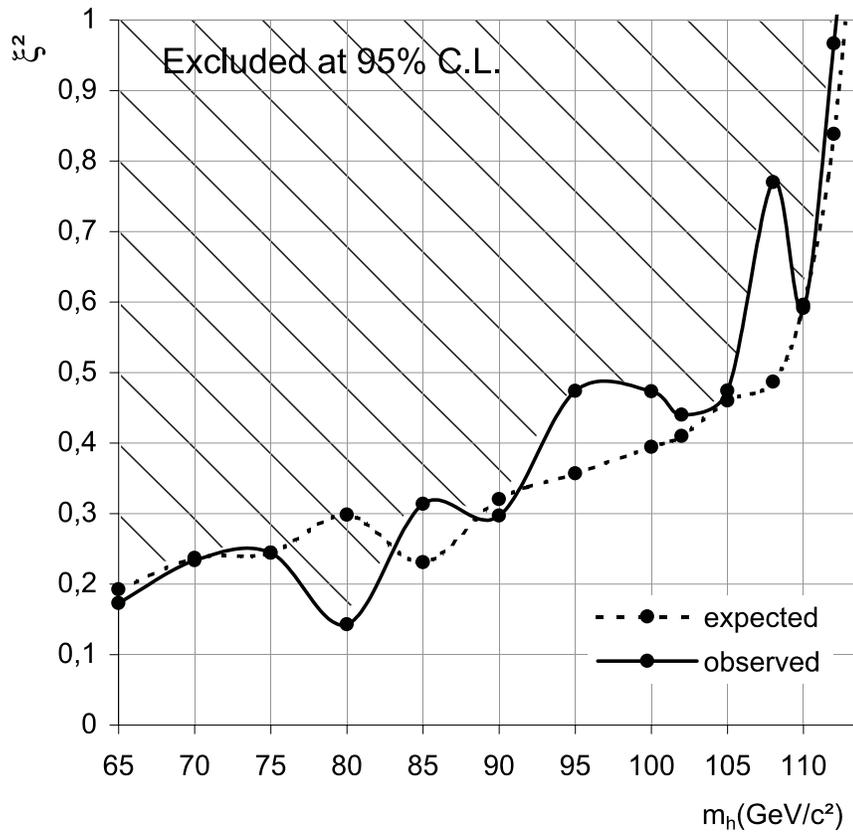


FIG. 3.32 – Région exclue à 95% C.L. dans le plan (m_h, ξ^2) (analyse combinée).

Conclusion

Une grande quantité de développements a été apportée à l'environnement d'analyse *ALPHA++*. Ces développements, portant sur la sélection des leptons, sur la reconstruction des jets mais aussi plus généralement sur la structure du logiciel ont rendu possible une analyse complète dans ce cadre.

Une comparaison poussée des algorithmes Fortran (*ALPHA*) avec leur équivalent C++ (*ALPHA++*) a été fructueusement menée, apportant au nouvel environnement le crédit nécessaire à la poursuite des développements.

Enfin, ces outils ont été mis en oeuvre pour l'étude d'un processus physique caractéristique intéressant (désintégration invisible d'un boson de Higgs). Les données à 189 GeV ont dans un premier temps été traitées de façon à valider la méthode et à confirmer les résultats publiés. Dans un second temps, les données de l'année 2000 ont été traitées. Cette analyse - la première menée dans le cadre d'*ALPHA++* - a permis d'exclure un tel boson de Higgs de masse inférieure à 112.15 GeV (pour $\xi^2 = 1$) à 95 % CL.

La première partie de ce travail a donné lieu à une note *ALEPH*[14], portant sur les algorithmes de sélection des leptons et de reconstruction des jets, ainsi que sur la comparaison à *ALPHA*. Une note similaire portant sur les autres développements techniques et une note présentant les résultats physiques obtenus dans le cadre de ce mémoire sont en préparation.

Bibliographie

- [1] *ALEPH* Collaboration, “*ALEPH* : A detector for electron-positron annihilation at LEP”, *Nucl. Inst. Meth.* **A294** (1990) 121.
- [2] *ALEPH* Collaboration, “Performance of the *ALEPH* detector at LEP”, *Nucl. Inst. Meth.* **A360** (1995) 481.
- [3] D. Creanza et al., “The new *ALEPH* vertex detector”, *Nucl. Inst. Meth.* **A409** (1998) 157.
- [4] B. Bloch-Dvaux and A. Putzer, Coordinate system, unit system, and naming conventions for detector modules in the *ALEPH* apparatus, ALEPH88-173, NOTE 88-21 (1988)
- [5] *ALEPH* Collaboration, “Observation of an excess in the search for the standard model Higgs boson at *ALEPH*”, *Phys. Lett.* **B495** (2000) 1-17.
- [6] Volker Blobel, The BOS system, dynamic memory management, DESY Internal report R1-99-01, DESY, Germany (1995)
- [7] Guenther Dissertori, The ALPHA++ Project.[web page] (1999); <URL :<http://alephwww.cern.ch/~disserto/objty/>>.
- [8] Rational Software Corporation, Rose Enterprise Edition.[web page] (2000); <URL :<http://www.rational.com>>.
- [9] *ALEPH* Collaboration, Single W Production at Energies up to $\sqrt{s} = 202\text{GeV}$ and Search for Anomalous Triple Gauge Boson Couplings, ALEPH/00-054, Moriond2000 (2000)
- [10] J. Badier, Longitudinal distribution of electromagnetic showers in the *ALEPH* electromagnetic calorimeter, ALEPH87-09/EMCAL 87-1 (1987)
- [11] D.E. Groom et al., “Review of particle physics”, *The European Physical Journal* **C15** (2000)
- [12] Hörndl M., *ALEPH* Collaboration, CERN Summer Student Project, “ALPHA++”, ALEPH99-097 (1999)
- [13] D. Wallin, *ALEPH* Collaboration, “Further development and testing of ALPHA++” Summer Student Report (2000).

- [14] C. Delaere, V. Lemaitre, V. Perez Reale and O. van der Aa, Comparison between an event selection program performed with *ALPHA* and *ALPHA++*, *ALEPH/2000-093* (2000)
- [15] P.W. Higgs, *Phys. Lett.* **12** (1964) 132 ; F. Englert and R. Brout, *Phys. Rev. Lett.* **13** (1964) ; G. Guralnik and C.K. Hagen, *Phys. Rev. Lett.* **13** (1964) 585.
- [16] The LEP and SLD Electroweak Working Group, A combination of Preliminary Electroweak Measurements and Constraints on the Standard Model, *CERN-EP/2000-16* (2000).
- [17] P. Igo-Kemenes, Report of the LEP Higgs Working Group to LEP Experiments Committee Meeting, CERN-Geneva, Nov. 3, 2000.
- [18] G. Degrassi, Where is the Higgs ?, *hep-th/0102137* (2001).
- [19] Mikhail Shaposhnikov and Peter Tinyakov, Extra dimensions as an alternative to Higgs mechanism ?, *hep-th/0102161* (2001)
- [20] J.F. Gunion, H.E. Haber, G. Kane, S. Dawson, The Higgs Hunter's Guide, Frontier in Physics. Addison-Wesley Publishing Company (1990) ISBN : 0-7382-0305-X
- [21] J.D. Bjorken, Weak interaction theory and neutral currents, Extracted from Proc. of Summer Inst. on Particle Physics, Stanford, Calif., Aug 2-13, SLAC-PUB-1866, Jan 1977. 84pp.
- [22] J. Ellis, M.K. Gaillard and D.V. Nanopoulos, *Nucl.Phys.* **B106** (1976) 292.
- [23] B.L. Ioffe and V.A. Khoze, *Sov.J.Part.Nucl.* **9** (1978) 50.
- [24] P. Janot, "The HZHA generator", dans "Physics at LEP2", Eds G. Altarelli, T. Sjostrand et F. Zwirner, CERN 96-01 (1996) Vol. 2., p. 309
- [25] Laurent Serin, on behalf of the ALEPH collaboration, Lower Limit on the mass of the neutralino (LSP) at LEP with the ALEPH detector, *hep-ex/0010049* (2000).
- [26] D. Chang and W.-Y. Keung, *phys. lett.* **B217** (1989) 238 ; S. Bertolini and A. Santamaria, *phys. lett.* **B220** (1989) 597.
- [27] ALEPH Collaboration, Search for invisible decays of the Higgs boson in e^+e^- collisions, *ALEPH/98-054* (1998)
- [28] ALEPH Collaboration, Search for invisible decays of the Higgs boson in e^+e^- collisions at a centre-of-mass energy up to 184 GeV , *ALEPH/99-008* (1999)
- [29] ALEPH Collaboration, Search for invisible decays of the Higgs boson in e^+e^- collisions at a centre-of-mass energy of 188.6 GeV , *ALEPH/99-069* (1999)
- [30] DELPHI Collaboration, A search for invisible Higgs boson produced in e^+e^- interactions up to $\sqrt{s} = 189\text{ GeV}$, CERN-EP-2000-051 (2000)

-
- [31] S. Jadach, B.F.L. Ward and Z. Was, *Comp. Phys. Comm.* **79** (1994) 503.
- [32] M. Skrzypek, S. Jadach, W. Placzek and Z. Wąs, *Comp. Phys. Comm.* **94** (1996) 216.
- [33] T. Sjöstrand, *Comp. Phys. Comm.* **82** (1994) 74 ;*ibid.*, CERN-TH 7112/93 (1993, revised August 1994).
- [34] J. Fujimoto *et al.*, *Comp. Phys. Comm.* **100** (1997) 128.
- [35] J.A.M. Vermaseren, *Proceedings of the IVth International Workshop on Gamma Gamma Interactions*, edited by G. Cochard and P. Kessler, Springer Verlag (1980), p.35 ; ALEPH Collaboration, *Phys. Lett.* **B313** (1993) 509.
- [36] Thomas Junk, *Confidence Level Computation for Combining Searches with Small Statistics*, hep-ex/9902006 (1999)
- [37] J. Schwindling, MLPfit : a tool for Multi-Layer Perceptrons. [web page] 2000 ; <URL :<http://schwind.home.cern.ch/schwind/MLPfit.html>>.
- [38] Jean-Baptiste de Vivie, *Recherche de bosons de Higgs à LEP2 et exploration de l'espace des paramètres du MSSM. Thèse présentée pour obtenir le grade de docteur en sciences de l'université Paris XI Orsay.* LAL 00-11 (2000)

Annexe A

Le driver *ALPHA++*

La classe principale intervenant dans le driver est l'*AlephSession*. Trois méthodes sont appelées successivement : *Initialize()*, *Run()* et *Terminate()*. Ces trois “états” sont propagés vers les différents managers :

1. l'*AlephDbManager* gère l'accès à la base de données, qu'elle soit BOS ou Objectivity, et opère le mapping dans l'*AlphaBank*.
2. l'*AlephIoManager* gère les opérations entrée/sortie, et accède notamment au fichier de configuration.
3. l'*AlephExManager* contient la boucle principale d'exécution (sur les runs et les événements) et appelle la méthode *UserEvent* pour chaque événement. C'est la classe *UserEvent* qui est implémentée par l'utilisateur, en utilisant les méthodes et objets de la partie “Analyse”, pour faire le travail d'analyse des données.

Selon que la base de données soit BOS ou Objectivity (choix opéré dans le fichier de configuration lu par l'*AlephIoManager*), une version particulière de l'*AlephDbManager* et de l'*AlephExManager* est utilisée. Notons que la session et les managers sont tous des singletons.

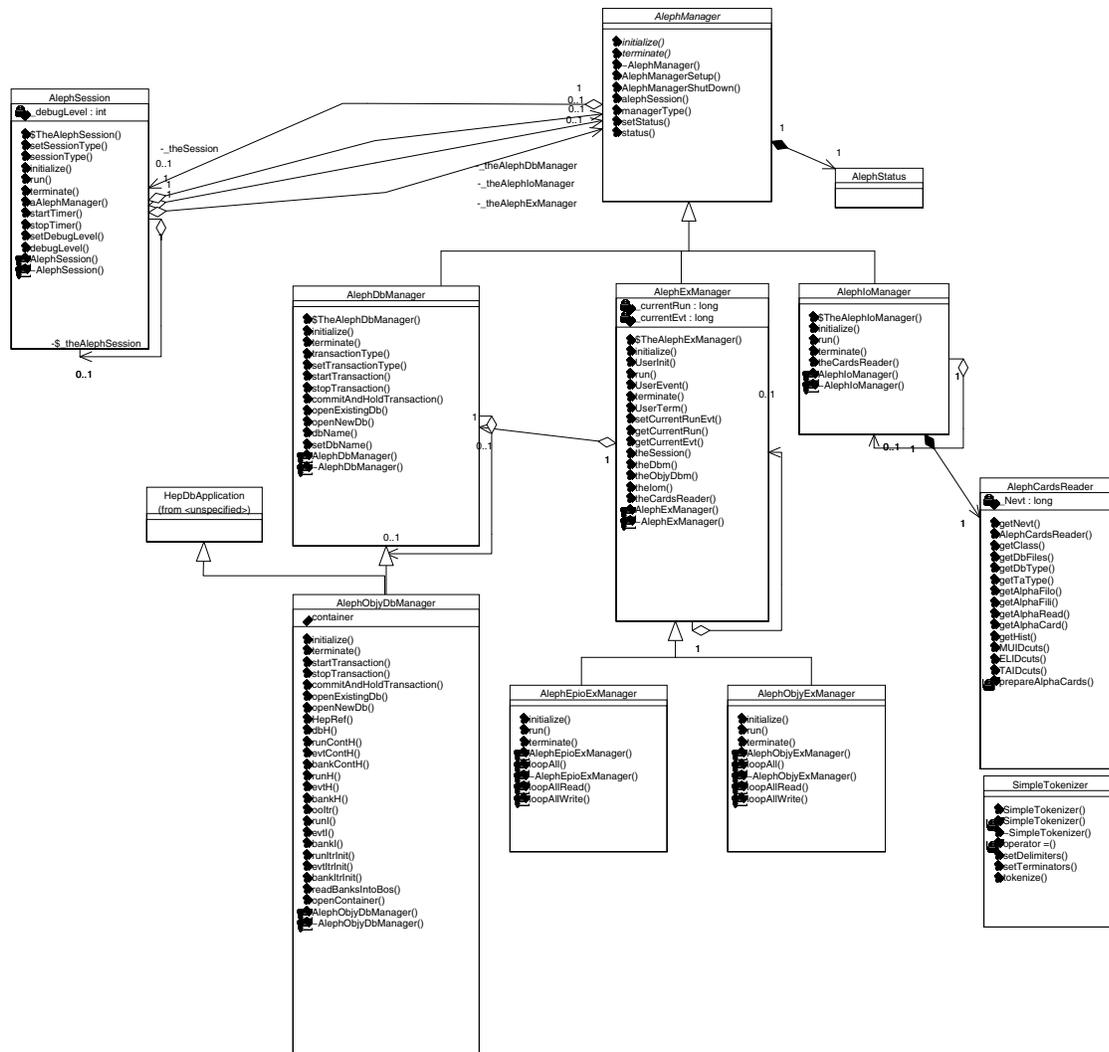


FIG. A.1 – Diagramme UML simplifié du driver ALPHA++. On y voit la AlephSession, contrôlant les différents AlephManagers, dérivant tous d'une classe de base AlephManager. A droite, l'AlephCardReader, qui lit le fichier de configuration.

Annexe B

Le mécanisme de brisure de symétrie

L'invariance de jauge locale sous le groupe $SU(2) \times U(1)$ du modèle standard requiert une masse nulle pour les bosons de jauge, car la présence de termes de masse viole la symétrie. En effet, tout terme du type $M^2 A_\mu A^\mu$ n'est pas invariant sous les transformations du type $A_\mu \rightarrow A_\mu - \partial_\mu \chi$ avec χ une fonction de l'espace-temps.

Puisque l'expérience met en évidence l'existence de bosons massifs, il est nécessaire d'introduire un mécanisme de brisure de symétrie, permettant de se ramener, à basse énergie, à une théorie effective comportant des bosons massifs.

Pour illustrer un tel mécanisme, considérons le cas simple abélien. Soit ϕ un champ scalaire complexe, et A^μ un boson de jauge non massif. Considérons une théorie décrite par le lagrangien

$$\mathcal{L} = (D_\mu \phi)^* (D^\mu \phi) + \mu^2 \phi^* \phi - \lambda (\phi^* \phi)^2 - \frac{1}{4} F^{\mu\nu} F_{\mu\nu}.$$

Dans cette expression, $\lambda > 0$ de façon à ce que le potentiel soit borné inférieurement et

$$F^{\mu\nu} = \partial^\mu A^\nu - \partial^\nu A^\mu.$$

L'invariance sous une transformation de jauge locale

$$\phi \rightarrow \phi' = e^{ig\chi(x)} \phi$$

$$A^\mu \rightarrow A'^\mu = A^\mu - \partial^\mu \chi(x)$$

est garantie en utilisant la dérivée covariante $D^\mu = \partial^\mu + igA^\mu$ en lieu et place de la dérivée partielle usuelle ∂^μ .

Le potentiel possède un minimum pour $\phi = v/\sqrt{2} = \sqrt{\mu/2\lambda}$. De façon à obtenir le spectre de la théorie, on développe le champ autour de ce minimum en écrivant

$$\phi = (v + h(x))/\sqrt{2}.$$

Par simple substitution dans \mathcal{L} , on obtient explicitement

$$\mathcal{L} = \frac{1}{2} \left((\partial_\mu - igA_\mu)(v+h)(\partial^\mu + igA^\mu)(v+h) \right) + \frac{1}{2} \mu^2 (v+h)^2 - \frac{1}{4} \lambda (v+h)^4 - \frac{1}{4} F^{\mu\nu} F_{\mu\nu}.$$

Le terme en $(g^2 v^2 / 2) A_\mu A^\mu$ est interprété comme un terme de masse pour le boson de jauge. Le terme $-\lambda v^2 h^2$ est quant à lui un terme de masse pour le boson scalaire. Il y a en plus des termes d'interaction en h^3 , h^4 , hAA et $h^2 AA$.

La théorie avec un boson scalaire complexe a ainsi été réinterprétée comme une théorie avec un boson scalaire réel et un boson de jauge massif, car le potentiel scalaire prend son minimum pour une valeur de ϕ non nulle.

Annexe C

Code source singleW commenté

```
void AlephEventManager::UserEvent(AlphaBanks& bb)
{
```

Il convient en premier lieu de recevoir les vecteurs d'objets de l'AlphaBank, reçue comme argument de la routine UserEvent (par référence).

```
// get vectors of pointers to the objects used in the analysis
AlephCollection<AlTrack *> altp=bb.TrackPV();
AlephCollection<AlMuon *> almup=bb.MuonPV();
AlephCollection<AlElec *> alelp=bb.ElecPV();
AlephCollection<AlEflw *> alefp=bb.EflwPV();
```

Dans le cadre de cette analyse, une bonne trace est définie comme ayant au moins 4 hits dans la TPC, et une valeur de $|\cos\theta| < 0.95$. De plus, elle doit trouver son origine dans un cylindre long de 10 *cm* de 2 *cm* de rayon centré au point d'interaction nominal et parallèle au tube à faisceau (les particules créées lors de la désintégration hadronique devrait trouver son origine au point d'interaction principal). Seules les particules chargées suffisamment éloignées du tube de faisceau sont sélectionnées, car les mesures y sont moins précises. Toutes les traces qui ne satisfont pas ces critères sont rejetées en appliquant la méthode `looperase()` d'AlephCollection. Elles sont également verrouillées pour ne pas être utilisées lors de la reconstruction des leptons.

```
// Good quality charged Tracks selection
int Nchsel=0;
float Echsel=0.;
for (Ich=altp.begin(); Ich<altp.end(); Ich++)
{
    int Ntpc = (*Ich)->NT();
    float costhch = (*Ich)->QCT();
    float d0 = (*Ich) -> QDB();
    float z0 = (*Ich) -> QZB();
```

```

if ((Ntpc<4)|| (costhch>0.95)|| (d0>2.)|| (z0>10.)
    || (costhch<-.95)|| (d0<-2)|| (z0<-10))
{
    (*Ich)->Lock();    // Lock the tracks (for leptons algos)
    altp.looperase(Ich); // Erases elements in the altp vector
}
else
{
    Nchsel++;
    Echsel += (*Ich)->QE();
}
}

```

Le événements sont alors sélectionnés en fonction des bonnes traces. Cette coupure est appliquée pour s'affranchir des signaux $\gamma\gamma$. L'exigence est de 5 bonnes traces minimum, avec une énergie totale supérieure à 5 GeV .

```

// now select the event according to the good charged tracks
bool GoodEvent = false;
if ((Nchsel >= 5) && (Echsel >= 5.))
{
    // now we have a good event
    Ngood++;
    GoodEvent = true;
}
else
    cout << "bad event" << endl;

```

On sélectionne les électrons, muons et taus. Comme expliqué à la section 2.3.2, les électrons et muons ont déjà été obtenus au tout début de la routine, et doivent juste être filtrés. Les taus doivent aussi être produits. Notons que la syntaxe diffère légèrement de la version finale présentée ci-avant.

```

//filter muons. parameters are (d0max,pmin,minhit,cosmax,z0max)
almup.FilterMu(.5,9.43,4,.95,10);
//filter elecs parameters are
//      (almup,d0max,pmin,minhit,cosmax,z0max,
//      nwirem,r3cut,r2cut0l,r2cut1l,r2cut2l,
//      r2cuth,r5cutl,r5cuth)
alelp.FilterEl(almup,.5,(0.05*QELEP),4,.95,10,
               0,1000,-3,-7,-5,
               1000,-0.5,1000);
//Generate raw tau candidates.
//Parameters are (Ycut,Scheme,Metric,Qelep)
AlephCollection<AlTau> JJ = alefp.ATauJn(Yjade, 1,1,QELEP);
AlephCollection<AlTau*> altaup = JJ.Pointers();

```

```
//filter taus. Parameters are (psummin,d0max,minhit,cosmax,z0max)
altaup.FilterTau(psummin,2.,4,.95,10.);
```

La masse invariante totale et l'impulsion manquante sont calculés. Ces méthodes retournent un quadrivecteur, de sorte que le résultat peut être analysé en utilisant les méthodes “ALPHA-like” implémentées dans la classe QvecBase.

```
QvecBase Psum = alefp.GetSum();
float esum = Psum.QE();
float psum = Psum.QP();
float msum = Psum.QM();
QvecBase Pmiss = alefp.GetMiss(QELEP);
float cthmiss=Pmiss.QCT();
```

Les événements marqués “à deux photons” sont rejetés en exigeant qu’il n’y ait pas d’énergie déposée par un objet neutre dans un cône de 12° autour du tube à faisceau (e12). Cette mesure tient compte des signaux ECAL, HCAL, LCAL et SICAL. L’énergie dans un secteur de 30° en ϕ autour de l’impulsion transverse manquante et dans un cône de 12° autour du tube à faisceau sont également estimés.

```
// wedge energy in 30 degree
float ewedge=0;
if (Pmiss.QP() >0)
{
  for(If1 = alefp.begin(); If1 < alefp.end(); If1++)
    if((acos(min(max(cos(Pmiss.QPH()-((*If1)->QPH())),-1.),1.)))*QQRADP<30)
      ewedge+= (*If1)->QE();
}
else
  GoodEvent=false;
//energy in 12 deg cone centred aound beam pipe
float e12_av=0;
float econ = 0;
for(If1 = alefp.begin(); If1 < alefp.end(); If1++)
{
  int type = (*If1)->getEfType();
  float costheta = fabs((*If1)->QCT());
  const double c12=0.9781476;
  float nint=(*If1)->QCH();
  if((type == 7) ||
     (type == 8) ||
     ((nint ==0) && (costheta>0.9781476)))
    e12_av += (*If1)->QE();
  if(costheta > c12)
```

```

    econe += (*If1)-> QE();
}

```

L'étape suivante consiste à identifier l'électron et/ou le muon avec une énergie supérieure à 0.05 x QELEP le plus anti-parallèle à l'impulsion manquante.

```

// Select the most anti-parallel (to Pmiss)
// identified electron or muon
int imutag=0;
int ieltag=0;
float emumax=0;
float eelmax=0;
float cosjmiss;
AlMuon selmuon;
AlElec selelec;
AlTrack* theElTrack;
AlTrack* theMuTrack;
float cosjmin=1;
for (Imu=almup.begin(); Imu<almup.end(); Imu++)
    if(Pmiss.QP() > 0)
    {
        cosjmiss=(*Imu)->QCOSA(Pmiss);
        if(cosjmiss <= cosjmin)
        {
            cosjmin=cosjmiss;
            emumax= (*Imu)->QE();
            selmuon =(**Imu);
            theMuTrack = (**Imu).getTrack();
            imutag =1;
        }
    }
cosjmin=1;
for (Iel=alelp.begin(); Iel<alelp.end(); Iel++)
    if(Pmiss.QP() > 0)
    {
        cosjmiss=(*Iel)->QCOSA(Pmiss);
        if(cosjmiss <= cosjmin)
        {
            cosjmin=cosjmiss;
            eelmax= (*Iel)->QE();
            selelec =(**Iel);
            theElTrack = (**Iel).getTrack();
            ieltag =1;
        }
    }
}

```

La sélection du tau le plus anti-parallèle à l'impulsion transverse manquante se fait en exigeant qu'aucune des traces associées au tau ne soit marquée comme électron ou muon. L'identification est faite en utilisant l'information de la trace.

```
// Select the most anti-parallel (to Pmiss)
// identified tau with energy > 0.05*ELEP
// Do not select Taus which are already flagged as electron or muon
int itautag=0;
float etaumax=0;
ALTau* seltau;
TauI Itau;
cosjmin=1;
for (Itau=altaup.begin(); Itau<altaup.end(); Itau++)
{
  int nch=0;
  bool keep=true;
  nch>(*Itau)->getNch();
  if(imutag>0 || ieltag>0)
    for (ObI Ifl>(*Itau)->getObjects().begin();
         Ifl<(*Itau)->getObjects().end(); Ifl++)
      if((((ALeflw*)(*If1))->getTrack()==theElTrack ||
          (((ALeflw*)(*If1))->getTrack()==theMuTrack) keep=false;
  if(Pmiss.QP())>0 && (keep))
  {
    cosjmiss= (*Itau)->QCOSA(Pmiss);
    if(cosjmiss < cosjmin)
    {
      cosjmin=cosjmiss;
      etaumax>(*Itau)->QE();
      seltau>(*Itau);
      itautag=nch;
    }
  }
}
```

A partir d'ici, le calcul n'est fait que pour les bons événements. Deux jets sont produits à l'aide de l'algorithme Durham PE pour les flux d'énergie. En définissant $y_{cut} = -2$, on force deux jets. L'acolinéarité et l'acoplanarité de ces jets est calculée, ainsi que la suite des Y_n .

```
if(GoodEvent){
  // Compute acoplanarity and acolinearity
  // ( Make 2 jets using Durham PE for a fixed number of jet,
  // enter a negative values of Ycut = -number of jets! )
  int jetr;
```

```

float Ycut = -2.;// force 2 jets
int scheme = 1; // E-scheme
AlephCollection<AlJet> Jn = alefp.DurhamJet(Ycut, scheme, 0);
jetr = Jn.size();
float acopl=180.;
float acoln=90.;
if(jetr == 2)
{
    acopl= acos(min(max(
        cos(Jn[0].QPH()-Jn[1].QPH()),-1.),1.))*QQRADP;
    acoln= acos(min(max(
        Jn[1].QCOSA(Jn[0]),(float)-1),(float)1))*QQRADP;
}
// Calculate Yi variables
vector<float> Yn = alefp.DurhamYn(scheme,0.);

    Si un tau a été trouvé, l'acoplanarité, l'acolinéarité et la masse invariante sont
calculés après exclusion des flux d'énergie appartenant au tau. Notons l'usage du
verrouillage récursif : Lock(1).

// If there is a tau, compute msumww acoplww acolww without the tau
float msumww = 0.;
float acoplww = 180.;
float acolww = 90.;
AlephCollection<AlJet> Jn2;
if(itautag>0)
{
    // lock all tracks belonging to the Tau.
    seltau->Lock(1);
    // and calculate sumww and acollinww
    Jn2 = alefp.DurhamJet(Ycut, scheme, 0);
    // first set the Sort Criterium
    // the default value is 3, so that we may forget this line
    QvecBase::SortCriterium = 3;
    // sort the vector using the component 3 (e) of A4V
    sort(Jn2.begin(),Jn2.end());
    // compute the invariant mass of the 2 jets
    msumww = Jn2[0].QM2(Jn2[1]);
    jetr = Jn2.size();
    // compute acoplanarity and acolinearity
    if(jetr == 2)
    {
        acoplww=acos(min(max(
            cos(Jn2[0].QPH()-Jn2[1].QPH()),-1.),1.))*QQRADP;
        acolww=acos(min(max(

```

```
        Jn2[1].QCOSA(Jn2[0]),(float)-1),(float)1))*QQRADP;
    }
    seltau->unlock(1);
}
```

Finalement, les variables d'intérêt sont copiées, et le ntuple PAW est rempli.

```
// Prepare output...
myvars[0] = Nchsel;           //nch
myvars[1] = Echsel;          //ech
...
myvars[25] = Yn[0];          //y12
myvars[26] = Yn[1];          //y23
myvars[27] = Yn[2];          //y34
myvars[28] = msumww;         //msumww
myvars[29] = acoplww;        //acoplww
myvars[30] = acolww;         //acolww
// Fill ntuple
HFN(1000,myvars);
}
else
    cerr << "bad event" << endl;
// delete temporary array
delete [] myvars;
}
```


Annexe D

Code source acoplanar

```
////////////////////////////////////
// User analysis routines for acoplanar jets selection
// suited for inv. Higgs and single W selection
//
// Author : C. Delaere
//
// Date: 2000-09-10
// Updated 2001-14-03
//
////////////////////////////////////

#include <fstream>
#include "AlephExManager.h"
#include "hbook.h"
#include "AlephCollection.h"
#include <vector>

extern ofstream fout;

////////////////////////////////////
// User Init
////////////////////////////////////

static int Ngood = 0;
static int Ebeam = -1;
static int Nbad = 0;
static float mZ = 91.1882; // RPP2000

void AlephExManager::UserInit()
{
    // book an ntuple
    int ntags = 90;
    char mytags[90][8] = {"run      ", "event  ", "trig   ",
                        "nch    ", "echf   ", "qelep  ",
                        "e12f   ", "econef ", "ewedgef",
                        "msumf  ", "esumf  ", "ptsumf ",
                        "thrust ", "spher  ", "plan   ", "aplan  ",
                        "cthmiss", "ptmiss ", "mmiss  ", "minvis ",
                        "acopl  ", "acolin ", "y12    ", "y23    ",
                        "imutag ", "ieltag ", "िताुताग",
                        "emumax ", "xmumax ", "ymumax ", "zmumax ",
                        "eelmax ", "xelmax ", "yelmax ", "zelmax "};
}
```

```

    "etaumax","xtaumax","ytaumax","ztaumax",
    "ej1  ","pxj1  ","pyj1  ","pzj1  ",
    "ej2  ","pxj2  ","pyj2  ","pzj2  ",
    "ej1ww ","pxj1ww ","pyj1ww ","pzj1ww ",
    "ej2ww ","pxj2ww ","pyj2ww ","pzj2ww ",
    "msumww ","acoplww","acolww ",
    "pmaxf  ","intjets","mjets  ","nchtjet",
    "pisol  ","lepton  ",
    "f1     ","f2     ","f3     ","f4     ",
    "f1_E   ","f1_px  ","f1_py  ","f1_pz  ",
    "f2_E   ","f2_px  ","f2_py  ","f2_pz  ",
    "f3_E   ","f3_px  ","f3_py  ","f3_pz  ",
    "f4_E   ","f4_px  ","f4_py  ","f4_pz  ",
    "f1_Ech ","f2_Ech ","f3_Ech ","f4_Ech ",
    "Egamma ","presel " };

HBOOKN(1000,"Acoplanar",ntags," ",5000,mytags);
}

////////////////////////////////////
////////////////////////////////////

//number of stable charged particles
int nchCount(ALMCtruth* particle)
{
    int nch=0;
    AlephCollection<ALMCtruth*> daughters = particle->getDaughterVector();
    for(AlephCollection<ALMCtruth*>::iterator daughter = daughters.begin();
        daughter < daughters.end();daughter++)

        if( (*daughter)->getDaughterVector().size() != 0)
            nch += nchCount(*daughter);
        else
            nch += (int)fabs((*daughter)->QCH());
    return nch;
}

//number of stable charged particles
HepLorentzVector A4VchCount(ALMCtruth* particle)
{
    HepLorentzVector Ech;
    AlephCollection<ALMCtruth*> daughters = particle->getDaughterVector();
    for(AlephCollection<ALMCtruth*>::iterator daughter = daughters.begin();
        daughter < daughters.end();daughter++)

        if( (*daughter)->getDaughterVector().size() != 0)
            Ech += A4VchCount(*daughter);
        else
            Ech +=(*daughter)->A4V();
    return Ech;
}

// visible part of A4V
HepLorentzVector visible(ALMCtruth* particle)
{
    HepLorentzVector vis;
    AlephCollection<ALMCtruth*> daughters = particle->getDaughterVector();
    for(AlephCollection<ALMCtruth*>::iterator daughter = daughters.begin();
        daughter < daughters.end();daughter++)

        if( (*daughter)->getDaughterVector().size() != 0)
            vis += visible(*daughter);
        else
            if(!(((daughter)->name()[0]=='n')&&((daughter)->name()[1]=='u')))
                vis+=(*daughter)->A4V();
    return vis;
}

```

```

/////////////////////////////////////////////////////////////////
// User Event
/////////////////////////////////////////////////////////////////
void AlephEventManager::UserEvent(AlphaBanks& EventInfo)
{
// some definitions
typedef AlephCollection<AlObject *>::iterator ObI;
typedef AlephCollection<AlTrack *>::iterator TI;
typedef AlephCollection<AlMuon *>::iterator MuI;
typedef AlephCollection<AlElec *>::iterator ELI;
typedef AlephCollection<AlEflw *>::iterator FLI;
typedef AlephCollection<AlTau *>::iterator TauI;
float QQRADP=180./M_PI;
float QELEP;
TI Ich;
MuI Imu;
ELI Iel;
FLI Ifl;
// initialize the run/event related quantities
AlRun myrun = EventInfo.Run();
AlEvent myevent = EventInfo.Event();
if (myrun.Energy() != -1) Ebeam = myrun.Energy();
QELEP = (2*Ebeam)/1000.;
if (QELEP<=0)
{
cerr << "ERROR: QELEP <= 0" << endl;
exit(1);
}
// Select event according to Energy
if (QELEP<110) return;
// Select event according to trigger/luminosity
int rawtrigger = myevent.GetRawTrigger();
if (!myevent.xlumok(QELEP))
{
cout << "Event " << myevent.number() << " : Bad Luminosity." << endl;
return;
}
if (!myevent.IsTrigger(1))
{
cout << "Event " << myevent.number() << " : Bad Trigger. ";
cout << "Trigger word is " << rawtrigger << endl;
return;
}
// get vectors of pointers to the objects used in MC analysis
AlephCollection<AlMctruth*> MCparticles = EventInfo.MctruthPV();
// get vectors of pointers to the objects used in the analysis
// vectors directly obtained from AlphaBank
AlephCollection<AlTrack *> altp =EventInfo.TrackPV();
AlephCollection<AlMuon *> almup=EventInfo.MuonPV();
AlephCollection<AlElec *> alelp=EventInfo.ElecPV();
AlephCollection<AlEflw *> alefp=EventInfo.EflwPV();
// Good quality charged Tracks selection
int Nchsel=0;
float Echsel=0.;
for (Ich=altp.begin(); Ich<altp.end(); Ich++)
{
int Ntpc = (*Ich)->NT();
float costhch = (*Ich)->QCT();
float d0 = (*Ich) -> QDB();
float z0 = (*Ich) -> QZB();
}
}

```

```

    if ((Ntpc<4)|| (costhch>0.95)|| (d0>2.)|| (z0>10.)
        || (costhch<-.95)|| (d0<-2)|| (z0<-10))
    {
        (*Ich)->Lock();          // Lock the tracks (for leptons algos)
        altp.looperase(Ich);     // Erases elements in the altp vector
    }
    else
    {
        Nchsel++;
        Echsel += (*Ich)->QE();
    }
}

// now select the event according to the good charged tracks
if ((Nchsel >= 3) && (Echsel >= 2.)) Ngood++;
else return;
if ( myevent.ErrorStatus() != 1) return;
if ( alefp.size()==0 ) return;
// compute some topological quantities
QvecBase Psum = alefp.GetSum();
float esum = Psum.QE();
float ptsum = Psum.QPT();
float psum = Psum.QP();
float msum = Psum.QM();
QvecBase Pmiss = alefp.GetMiss(QELEP);
float cthmiss=Pmiss.QCT();
float mmiss=Pmiss.QM();
float pmiss=Pmiss.QP();
float minvis=pow(pow(QELEP-(mZ*esum/msum),2)-pow(mZ*pmiss/mmiss,2),.5);
float thethrust = alefp.AThrust().getThrustValue();
float sphericity = alefp.Sphericity();
float planarity = alefp.planarity();
float aplanarity = alefp.aplanarity();
// preselection cuts
if((Pmiss.QPT()<=0) || (fabs(Pmiss.QCT())>.95) || (msum<5)) return;
// wedge energy in 30 degree
float ewedge=0;
if (Pmiss.QP() >0)
{
    for(If1 = alefp.begin(); If1 < alefp.end(); If1++)
        if((acos(min(max(cos(Pmiss.QPH()-((*If1)->QPH()),-1.),1.)))*QQRADP<30)
            ewedge+= (*If1)->QE();
}
else
    return;
// preselection cuts
if ((ewedge>(.2*QELEP))&&(Pmiss.QPT()<(.02*QELEP))) return;
//energy in 12 deg cone centred aound beam pipe
float e12_av=0;
float econe =0;
const float c12=0.9781476;
for(If1 = alefp.begin(); If1 < alefp.end(); If1++)
{
    int type = (*If1)->getEfType();
    float costheta = fabs((*If1)->QCT());
    const double c12=0.9781476;
    float nint=(*If1)->QCH();
    if((type == 7) || (type == 8) || ((nint ==0) && (costheta > c12)))
        e12_av += (*If1)->QE();
    if(costheta > c12)
        econe += (*If1)-> QE();
}
}

```

```

if(myrun.number(<4000) e12_av += myevent.gen_e12(QELEP);

// Initialize leptons loops
//generate raw taus candidates
AlephCollection<ALTau> JJ = alefp.ATauJn(theCardsReader()->TAIDcuts(),QELEP);
AlephCollection<ALTau*> altaup = JJ.Pointers();

//filter muons
almup.FilterMu(theCardsReader()->MUIDcuts());
sort(almup.begin(),almup.end());

//filter electrons
alelp.FilterEl(almup,theCardsReader()->ELIDcuts());
sort(alelp.begin(),alelp.end());

//filter taus
altaup.FilterTau(theCardsReader()->TAIDcuts());

// compute max leptonic momentum
vector<float> pmax(2);
pmax[0] = (almup.size() ? almup[almup.size()-1]->QP() : 0.);
pmax[1] = (alelp.size() ? alelp[alelp.size()-1]->QP() : 0.);
float plepmax = *max_element(pmax.begin(),pmax.end());

// Select the most anti-parallel (to Pmiss) identified electron or muon
int imutag=0;
int ieltag=0;
HepLorentzVector A4Vmumax;
HepLorentzVector A4Velmax;
float eelmax=0;
float cosjmiss;
ALMuon selmuon;
ALElec selelec;
ALTrack* theElTrack;
ALTrack* theMuTrack;
float cosjmin=1;
for (Imu=almup.begin(); Imu<almup.end(); Imu++)
  if(Pmiss.QP() > 0)
  {
    cosjmiss=(*Imu)->QCOSA(Pmiss);
    if(cosjmiss <= cosjmin)
    {
      cosjmin=cosjmiss;
      A4Vmumax = (*Imu)->A4V();
      selmuon =(**Imu);
      theMuTrack = (*Imu) -> getTrack();
      imutag =1;
    }
  }
cosjmin=1;
for (Iel=alelp.begin(); Iel<alelp.end(); Iel++)
  if(Pmiss.QP() > 0)
  {
    cosjmiss=(*Iel)->QCOSA(Pmiss);
    if(cosjmiss <= cosjmin)
    {
      cosjmin=cosjmiss;
      A4Velmax = (*Iel)->A4V();
      selelec =(**Iel);
      theElTrack = (*Iel) -> getTrack();
      ieltag =1;
    }
  }

```

```

    }
// Select the most anti-parallel (to Pmiss)
// identified tau with energy > 0.05*ELEP
// Do not select Tau which are already identified as electron or muon...
int itautag=0;
HepLorentzVector A4Vtaumax;
ATau* seltau;
TauI Itau;
cosjmin=1;
for (Itau=altaup.begin(); Itau<altaup.end(); Itau++)
{
    int nch=0;
    bool keep=true;
    nch=(*Itau)->getNch();
    if(imutag>0 || ieltag>0)
        for (ObI Ifl=(*Itau)->getObjects().begin();Ifl<(*Itau)->getObjects().end();Ifl++)
            if((((ALEflw*)(*If1)->getTrack()==theElTrack ||
                ((ALEflw*)(*If1)->getTrack()==theMuTrack) keep=false;
    if(Pmiss.QP(>0 && (keep))
    {
        cosjmiss= (*Itau)->QCOSA(Pmiss);
        if(cosjmiss < cosjmin)
        {
            cosjmin=cosjmiss;
            A4Vtaumax = (*Itau)->A4V();
            seltau=(*Itau);
            itautag=nch;
        }
    }
}

int alefsize=0;
for (AlephCollection<AlEflw*::iterator i = alefp.begin(); i != alefp.end(); i++)
    if (!(i->isLocked())) alefsize++;
if (alefsize<3) return;

// Compute acoplanarity and acolinearity
// ( Make 2 jets using Durham PE for a fixed number of jet,
// enter a negative values of Ycut = -number of jets! )
cerr << "." ;
int jetr;
float Ycut = -2.;// force 2 jets
int scheme = 1; // E-scheme
AlephCollection<AlJet> Jn = alefp.DurhamJet(Ycut, scheme, 0);
jetr = Jn.size();
// first set the Sort Criterium ( component of the 4-vector )
// the default value is 3, so that we may forget this line
QvecBase::SortCriterium = 3;
// sort the vector using the component 3 (e) of A4V
sort(Jn.begin(),Jn.end());
float acopl=180.;
float acolin=90.;
float one=1.;
if(jetr == 2)
{
    acopl= acos(min(max(cos(Jn[0].QPH()-Jn[1].QPH()),-1.),1.))*QQRADP;
    acolin= acos(min(max(Jn[1].QCOSA(Jn[0]),-one),one))*QQRADP;
}

// Final preselection cuts
bool set1 = ((Pmiss.QPT(<=0) ||(fabs(Pmiss.QCT(>0.95) ||
            (msum<5) ||(ewedge>(0.2*ELEP) ||(acolin>175));

```

```

bool set2 = ((Pmiss.QPT()<=0) || (fabs(Pmiss.QCT())>.95) ||
             (esum>(.9*QELEP)) || (Pmiss.QPT(<(.02*QELEP)) || (acopl>175));
if (set1 && set2) return;

// Calculate Yi variables
vector<float> Yn = alefp.DurhamYn(scheme,0.);

// Make 3 jets using Durham PE
Ycut = -3.; // force 2 jets
scheme = 1; // E-scheme
AlephCollection<ALJet> Jn2 = alefp.DurhamJet(Ycut, scheme, 0);
vector<float> interjet(3);
vector<float> msumjet(3);
vector<int> nchtjet(3);
float interjetmin;
float msumjetmin;
float nchtjetmin;
if (Jn2.size()==3)
{
    interjet[0] = Jn2[0].QCOSA(Jn2[1]);
    msumjet[0] = Jn2[0].QM2(Jn2[1]);
    nchtjet[0] = 0;
    for(ObI iter=Jn2[0].getObjects().begin();iter<Jn2[0].getObjects().end();iter++)
        if (((ALEflw*)(*iter))->getEfType()<=3) nchtjet[0] ++;
    interjet[1] = Jn2[0].QCOSA(Jn2[2]);
    msumjet[1] = Jn2[0].QM2(Jn2[2]);
    nchtjet[0] = 1;
    for(ObI iter=Jn2[1].getObjects().begin();iter<Jn2[1].getObjects().end();iter++)
        if (((ALEflw*)(*iter))->getEfType()<=3) nchtjet[1] ++;
    interjet[2] = Jn2[1].QCOSA(Jn2[2]);
    msumjet[2] = Jn2[1].QM2(Jn2[2]);
    nchtjet[2] = 0;
    for(ObI iter=Jn2[2].getObjects().begin();iter<Jn2[2].getObjects().end();iter++)
        if (((ALEflw*)(*iter))->getEfType()<=3) nchtjet[2] ++;
    interjetmin = *max_element(interjet.begin(),interjet.end());
    msumjetmin = *min_element(msumjet.begin(),msumjet.end());
    nchtjetmin = *min_element(nchtjet.begin(),nchtjet.end());
}

// Find the most isolated charged track
float globalmax=0;
ALTrack* associated;
for ( TI track1 = altp.begin(); track1<altp.end(); track1++)
{
    float localmin = 2;
    for ( TI track2 = altp.begin(); track2<altp.end(); track2++)
        if (track2!=track1)
        {
            float now = (1-(*track2)->QCOSA(**track1));
            localmin = localmin < now ? localmin : now;
        }
    if (globalmax<localmin)
    {
        globalmax = localmin;
        associated = (*track1);
    }
}

// Check if it is a lepton...

int lepfound = 0;
for (MuI mui=almup.begin(); mui<almup.end();mui++)
    lepfound = (*mui)->getTrack()==associated ? 1 : 0;

```

```

if (lepfound==0)
for (Eli eli=alelp.begin(); eli<alelp.end();eli++)
  lepfound = (*eli)->getTrack()==associated ? 2 : 0;

// If there is a tau, compute msumww acoplww acolww without the tau
float msumww = 0.;
float acoplww = 180.;
float acolww = 90.;
bool validwvjets = false;
AlephCollection<AlJet> Jn3;
if(itautag>0)
{
  // lock all tracks belonging to the Tau.(use the recursive lock)
  seltau->Lock(1);
  // and calculate sumww and acolllnww
  int alefsize=0;
  for (AlephCollection<AlEflw*>::iterator i = alefp.begin(); i != alefp.end(); i++)
    if (!((*i)->isLocked())) alefsize++;
  if (alefsize>1)
  {
    Ycut = -2;
    scheme = 1;
    validwvjets=true;
    Jn3 = alefp.DurhamJet(Ycut, scheme, 0);
    // first set the Sort Criterium ( component of the 4-vector )
    // the default value is 3, so that we may forget this line
    QvecBase::SortCriterium = 3;
    // sort the vector using the component 3 (e) of A4V
    sort(Jn3.begin(),Jn3.end());
    // compute the invariant mass of the 2 jets
    msumww = Jn3[0].QM2(Jn3[1]);
    jetr = Jn3.size();
    // compute acoplanarity and acolinearity
    if(jetr == 2)
    {
      acoplww= acos(min(max(cos(Jn3[0].QPH()-Jn3[1].QPH()),-1.),1.))*QQRADP;
      acolww= acos(min(max(Jn3[1].QCOSA(Jn3[0]),(float)-1),(float)1))*QQRADP;
    }
  }
  else
  {
    msumww = 0;
    acoplww = 0;
    acolww = 0;
  }
  seltau->unLock(1);
}

// Compute MC information.
// We've to try and catch to handle some very special cases
float Egamma=0;
vector<HepLorentzVector> fermions;
vector<int> fermionscodes;
vector<float> TauChargedEnergy;
bool badMC = true;
if(myrun.number()<2000) try
{
  badMC = false;
  // look for primary particles
  for(AlephCollection<AlMCtruth*>::iterator MCpart=MCparticles.begin();
    MCpart<MCparticles.end();MCpart++)
    if((*MCpart)->NO()!=0) MCparticles.looperase(MCpart);
  // compute energy for gammas

```

```

for(AlephCollection<AlMCtruth*>::iterator MCpart=MCparticles.begin();
    MCpart<MCparticles.end();MCpart++)
if((*MCpart)->PA()==1)
{
    Egamma+=(*MCpart)->QE();
    MCparticles.looperase(MCpart);
}
// now, try to obtain the 4 (2) fermions
// First, look for Ws or Zs, the names are for the W case.
AlephCollection<AlMCtruth*>::iterator theW;
for(AlephCollection<AlMCtruth*>::iterator MCpart=MCparticles.begin();
    MCpart<MCparticles.end();MCpart++)
if (((*MCpart)->name()[0] == 'W') || ((*MCpart)->name()[0] == 'Z'))
{theW=MCpart;
    AlephCollection<AlMCtruth*> Wdaughter = (*theW)->getDaughterVector();
backsafe:
    for(AlephCollection<AlMCtruth*>::iterator MCprt=Wdaughter.begin();
        MCprt<Wdaughter.end();MCprt++)
    {
        // special case for Higgsstrahlung on a Z
        // look the decays products of the Z
        if ((*MCprt)->name()[0] == 'Z')
        {
            AlephCollection<AlMCtruth*>::iterator theZ=MCprt;
            AlephCollection<AlMCtruth*> Zdaughter = (*theZ)->getDaughterVector();
            for(AlephCollection<AlMCtruth*>::iterator MCZd=Zdaughter.begin();
                MCZd<Zdaughter.end();MCZd++)
            {
                if(Zdaughter.size() == 2)
                // if the particle is a tau, take only visible Energy
                // also store the decay type (e, mu, 1p, 3p)
                if (((*MCZd)->PA()==33) || ((*MCZd)->PA()==34))
                {
                    int thecode=0;
                    AlephCollection<AlMCtruth*>::iterator thetau=MCZd;
                    int nch = nchCount(*thetau);
                    HepLorentzVector tau = visible(*thetau);
                    thecode += (nch+2)*1000;
                    AlephCollection<AlMCtruth*> daughters=(*thetau)->getDaughterVector();
                    for(AlephCollection<AlMCtruth*>::iterator daughter=daughters.begin();
                        daughters.end();daughter++)
                    {
                        if (((*daughter)->PA()==2) || ((*daughter)->PA()==3)) thecode = 1000;
                        if (((*daughter)->PA()==5) || ((*daughter)->PA()==6)) thecode = 2000;
                    }
                    thecode += (*thetau)->PA();
                    fermions.push_back(tau);
                    fermionscodes.push_back(thecode);
                    TauChargedEnergy.push_back(A4VchCount(*thetau).e());
                }
                else
                {
                    fermions.push_back((*MCZd)->A4V());
                    fermionscodes.push_back((*MCZd)->PA());
                    TauChargedEnergy.push_back(0);
                }
            }
        }
        if(Zdaughter.size() > 2)
        {
            AlephCollection<AlJet> Jets = Zdaughter.DurhamJet(-2,1,0);
            if (Jets.size() == 2)
            {

```

```

        fermions.push_back((Jets[0]).A4V());
        fermions.push_back((Jets[1]).A4V());
        fermionscodes.push_back(0);
        fermionscodes.push_back(0);
        TauChargedEnergy.push_back(0);
        TauChargedEnergy.push_back(0);
    }
}
if(Zdaughter.size() == 0 ) {Wdaughter.looperase(MCprt);goto backsafe;}
}
else
{
    if(Wdaughter.size() == 2)
        // if the particle is a tau, take only visible Energy
        // also store the decay type (e, mu, 1p, 3p)
        if(((MCprt)->PA()==33)||((MCprt)->PA()==34))
        {
            int thecode=0;
            AlephCollection<AlMCtruth*>::iterator thetau=MCprt;
            int nch = nchCount(*thetau);
            HepLorentzVector tau = visible(*thetau);
            thecode += (nch+2)*1000;
            AlephCollection<AlMCtruth*> daughters=(*thetau)->getDaughterVector();
            for(AlephCollection<AlMCtruth*>::iterator daughter=daughters.begin();
                daughter<daughters.end();daughter++)
            {
                if (((*daughter)->PA()==2)||((*daughter)->PA()==3)) thecode = 1000;
                if (((*daughter)->PA()==5)||((*daughter)->PA()==6)) thecode = 2000;
            }
            thecode += (*thetau)->PA();
            fermions.push_back(tau);
            fermionscodes.push_back(thecode);
            TauChargedEnergy.push_back(A4VchCount(*thetau).e());
        }
        else
        {
            fermions.push_back((MCprt)->A4V());
            fermionscodes.push_back((MCprt)->PA());
            TauChargedEnergy.push_back(0);
        }
    }
}
if(Wdaughter.size() > 2)
{
    AlephCollection<AlJet> Jets = Wdaughter.DurhamJet(-2,1,0);
    if(Jets.size() == 2)
    {
        fermions.push_back((Jets[0]).A4V());
        fermions.push_back((Jets[1]).A4V());
        fermionscodes.push_back(0);
        fermionscodes.push_back(0);
        TauChargedEnergy.push_back(0);
        TauChargedEnergy.push_back(0);
    }
}
if(Wdaughter.size() == 0)
{
    AlephCollection<AlMCtruth*> Wfragments;
    for(AlephCollection<AlMCtruth*>::iterator Wfrag=theW;
        Wfrag<MCparticles.end();Wfrag++)
    {
        // prepare a vector to "jetarize"

```

```

// look for one of the possible quarks
int thefirstquark = (*Wfrag)->PA();
int current;
int quarksnum=0;
float totalE=0;
bool islast=false;
if ((thefirstquark > 100)&&(thefirstquark < 113))
{
    do
    {
        current = (*Wfrag)->PA();
        totalE += (*Wfrag)->QE();
        Wfragments.push_back(*Wfrag);
        MCparticles.looperase(Wfrag);
        Wfrag++;
        if ((current> 100)&&(current < 107)) quarksnum++;
        if ((current> 106)&&(current < 113)) quarksnum--;
        if (Wfrag==MCparticles.end()) break;
    }
    while( !((current> 100)&&(current < 113)&&
        (!quarksnum)&&(totalE>(.90*(theW->QE()))));
    break;
}
}
// build the 2 jets
if (Wfragments.size()<2) throw 2;
AlephCollection<AlJet> Jets = Wfragments.DurhamJet(-2,1,0);
if(Jets.size() == 2)
{
    fermions.push_back((Jets[0]).A4V());
    fermions.push_back((Jets[1]).A4V());
    fermionscodes.push_back(0);
    fermionscodes.push_back(0);
    TauChargedEnergy.push_back(0);
    TauChargedEnergy.push_back(0);
}
}
MCparticles.looperase(theW);
MCpart--;
}
// now considering remaining particles.
// should be 4, 2 or 0
if((MCparticles.size()!=0)&&(MCparticles.size()!=2)&&(MCparticles.size()!=4)) throw 0;
for(AlephCollection<AlMCtruth*>::iterator MCpart=MCparticles.begin();
    MCpart<MCparticles.end();MCpart++)
{
    fermions.push_back((*MCpart)->A4V());
    fermionscodes.push_back((*MCpart)->PA());
    TauChargedEnergy.push_back(0);
}
// test the result...
if((fermions.size() != 2)&&(fermions.size() != 4)) throw 1;
// if there are only 2 particles (qqbar production) add 2 null objects
if(fermions.size() == 2)
{
    fermions.push_back(HepLorentzVector());
    fermionscodes.push_back(-1);
    TauChargedEnergy.push_back(0);
    fermions.push_back(HepLorentzVector());
    fermionscodes.push_back(-1);
    TauChargedEnergy.push_back(0);
}
}

```

```

}
// handle errors
catch(int)
{
    cout << "Event " << myevent.number() << " : unable to find out suited fermions" << endl;
    badMC = true;
}
// normalize parameters
Echsel /= QELEP;
e12_av /= QELEP;
econe /= QELEP;
ewedge /= QELEP;
psum /= QELEP;
msum /= QELEP;
ptsum /= QELEP;
esum /= QELEP;
plepmax /= QELEP;

// Ntuple variable initialization
int nvars = 90;
float* myvars = new float[nvars];
for (int i=0;i<nvars; i++) myvars[i]=0.;

// Prepare output...
myvars[0] = myrun.number(); //run number
myvars[1] = myevent.number(); //event number
myvars[2] = rawtrigger; //raw trigger word
myvars[3] = Nchsel; //nch
myvars[4] = Echsel; //ech
myvars[5] = QELEP; //LEP energy
myvars[6] = e12_av; //forward E
myvars[7] = econe; //E in 12deg cone
myvars[8] = ewedge; //wedge Energy
myvars[9] = msum; //invariant mass
myvars[10] = esum; //invariant E
myvars[11] = ptsum; //invariant Pt
myvars[12] = the thrust; //thrust value
myvars[13] = sphericity; //sphericity
myvars[14] = planarity; //planarity
myvars[15] = aplanarity; //aplanarity
myvars[16] = cthmiss; //cthmiss
myvars[17] = Pmiss.QPT()/QELEP; //ptmiss
myvars[18] = Pmiss.QM(); //missing mass
myvars[19] = minvis; //invisible mass (if signal)
myvars[20] = acopl; //acoplanarity
myvars[21] = acoln; //acolinearity
myvars[22] = Yn[0]; //y12
myvars[23] = Yn[1]; //y23
myvars[24] = imutag; //muon
myvars[25] = ieltag; //electron
myvars[26] = itautag; //tau
myvars[27] = A4Vmumax.e(); //muon A4V
myvars[28] = A4Vmumax.x();
myvars[29] = A4Vmumax.y();
myvars[30] = A4Vmumax.z();
myvars[31] = A4Velmax.e(); //electron A4V
myvars[32] = A4Velmax.x();
myvars[33] = A4Velmax.y();
myvars[34] = A4Velmax.z();
myvars[35] = A4Vtaumax.e(); //tau A4V
myvars[36] = A4Vtaumax.x();
myvars[37] = A4Vtaumax.y();

```

```

myvars[38] = A4Vtaumax.z();
myvars[39] = Jn[1].QE();           // WW jets A4Vs
myvars[40] = Jn[1].QX();
myvars[41] = Jn[1].QY();
myvars[42] = Jn[1].QZ();
myvars[43] = Jn[0].QE();
myvars[44] = Jn[0].QX();
myvars[45] = Jn[0].QY();
myvars[46] = Jn[0].QZ();
if (validwjets)
{
  myvars[47] = Jn3[1].QE();       // WW jets A4Vs
  myvars[48] = Jn3[1].QX();
  myvars[49] = Jn3[1].QY();
  myvars[50] = Jn3[1].QZ();
  myvars[51] = Jn3[0].QE();
  myvars[52] = Jn3[0].QX();
  myvars[53] = Jn3[0].QY();
  myvars[54] = Jn3[0].QZ();
}
myvars[55] = msumww;             //WW invariant mass
myvars[56] = acoplww;           //WW acoplanarity
myvars[57] = acolww;           //WW acolinearity
myvars[58] = plepmax;          //max lep P
myvars[59] = interjetmin;      //min interjet cos
myvars[60] = msumjetmin;       //min invariant mass of 2 jets
myvars[61] = nchtjetmin;       //min number of charged tracks in jets
myvars[62] = associated->QP();  //P of the most isolated track.
myvars[63] = lepfound;         //is the most isolated track a lepton
if(!badMC)
{
  myvars[64] = fermionscodes[0]; // MC fermions
  myvars[65] = fermionscodes[1];
  myvars[66] = fermionscodes[2];
  myvars[67] = fermionscodes[3];
  myvars[68] = (float)fermions[0].e(); // MC A4V of each fermion
  myvars[69] = (float)fermions[0].x();
  myvars[70] = (float)fermions[0].y();
  myvars[71] = (float)fermions[0].z();
  myvars[72] = (float)fermions[1].e();
  myvars[73] = (float)fermions[1].x();
  myvars[74] = (float)fermions[1].y();
  myvars[75] = (float)fermions[1].z();
  myvars[76] = (float)fermions[2].e();
  myvars[77] = (float)fermions[2].x();
  myvars[78] = (float)fermions[2].y();
  myvars[79] = (float)fermions[2].z();
  myvars[80] = (float)fermions[3].e();
  myvars[81] = (float)fermions[3].x();
  myvars[82] = (float)fermions[3].y();
  myvars[83] = (float)fermions[3].z();
  myvars[84] = TauChargedEnergy[0];
  myvars[85] = TauChargedEnergy[1];
  myvars[86] = TauChargedEnergy[2];
  myvars[87] = TauChargedEnergy[3];
  myvars[88] = Egamma;
}
myvars[89] = ((!set1) + 2*(!set2));
// Fill ntuple
HFN(1000,myvars);
// delete temporary array
delete [] myvars;

```

```
}  
  
////////////////////////////////////  
////////////////////////////////////  
  
////////////////////////////////////  
// User Term  
////////////////////////////////////  
void AlephExManager::UserTerm()  
{  
    fout << endl << " USER TERM : Ngood = " << Ngood << endl;  
}
```