

# Mesure de l'asymétrie avant-arrière $A_{FB}$ avec le détecteur CMS

Exercices dirigés du cours *Physique des particules* - PHYSF416

Année académique 2017-2018

- 1 Présentation
- 2 Introduction théorique
- 3 Le détecteur CMS
- 4 Les outils de base
- 5 Introduction au C++ et à ROOT
- 6 Exercice 1: calculer l'asymétrie  $A_{FB}$  théorique
- 7 Exercice 2: simulations Monte Carlo

# Section 1

## Présentation

## Déroulement des exercices:

- 4 séances de 3h
  - ▶ Les mardis 07-14-21-28/11 de 13h à 16h
  - ▶ (Mardi 05/12 de 13h à 16h pour une séance Q&A)
- Répartition du travail:
  - ▶ Essentiellement en classe
  - ▶ A terminer chez vous

## Présentation de vos résultats:

- Rapport écrit
  - ▶ Introduction du contexte théorique et expérimental
  - ▶ Sélection des événements, histogrammes pertinents
  - ▶ Résultats commentés

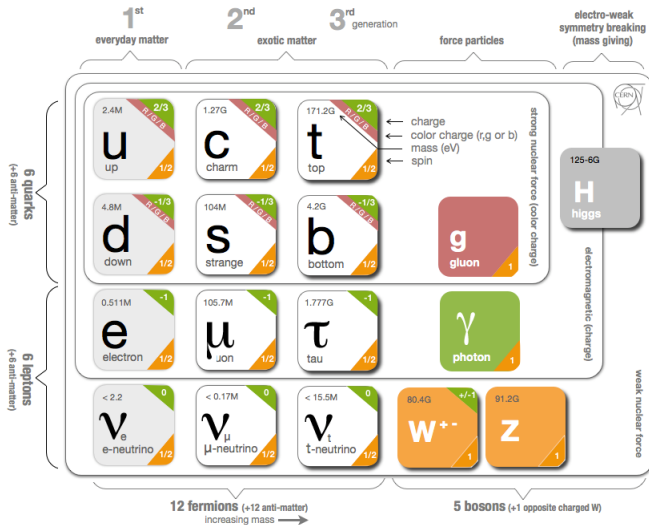
## Encadrement

- David Vannerom: david.vannerom@ulb.ac.be
- Reza Goldouzian: reza.goldouzian@cern.ch
- Diego Beghin: Diego.Beghin@ulb.ac.be

## Section 2

### Introduction théorique

# Le Modèle Standard (SM)

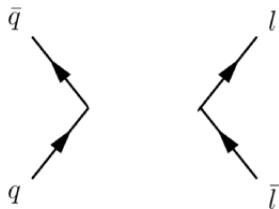


# Le processus Drell-Yan (DY)

DY  $\equiv$  annihilation d'une paire de quarks pour donner une paire de leptons chargés:

$$q\bar{q} \rightarrow l\bar{l}$$

Quels sont les diagrammes possibles?



# Le processus Drell-Yan (DY)

DY  $\equiv$  annihilation d'une paire de quarks pour donner une paire de leptons chargés:

$$q\bar{q} \rightarrow l\bar{l}$$

Quels sont les diagrammes **importants**?

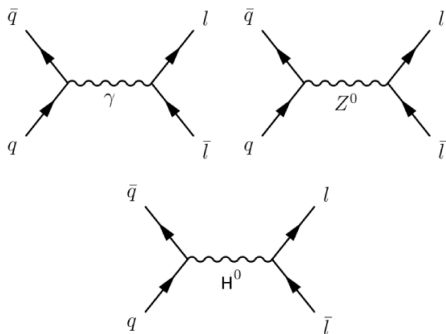




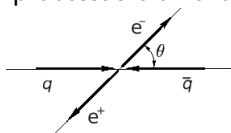
Diagramme de Feynman = Nombre complexe (élément de matrice  $\mathcal{M}$ ):

$$\mathcal{M}_\gamma = \begin{array}{c} \begin{array}{ccc} \bar{q} & & l \\ \swarrow & & \nearrow \\ & \gamma & \\ \nwarrow & & \searrow \\ q & & \bar{l} \end{array} \\ \\ \begin{array}{ccc} \bar{q} & & l \\ \swarrow & & \nearrow \\ & Z^0 & \\ \nwarrow & & \searrow \\ q & & \bar{l} \end{array} \end{array}$$

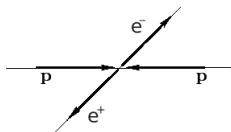
La section efficace  $\sigma$  d'un processus faisant intervenir plusieurs diagrammes est définie par:

$$\begin{aligned} \sigma &= \left| \sum_{i=1}^{N_{\text{processus}}} \mathcal{M}_i \right|^2 \\ &= |\mathcal{M}_\gamma + \mathcal{M}_Z|^2 \\ &= |\mathcal{M}_\gamma|^2 + |\mathcal{M}_Z|^2 + \mathcal{M}_\gamma \mathcal{M}_Z^* + \mathcal{M}_\gamma^* \mathcal{M}_Z \\ &= \text{Photon-only} + \text{Z-only} + \text{Interference} \end{aligned}$$

Avant de discuter le résultat du calcul des sections efficaces, un peu de cinématique. Distinguons le processus élémentaire DY:



de la collision proton-proton:



- $\sqrt{s}$  = énergie dans le centre de masse  $p - p$
- $\sqrt{s'}$  = énergie dans le centre de masse  $q - \bar{q}$
- $\theta$  = angle entre la direction du quark initial et du lepton final

Voici l'expression des sections efficaces différentielles:

Photon-only:

$$\frac{d\sigma_\gamma}{d\Omega} = \frac{e^4}{(4\pi)^2} Q_q^2 Q_l^2 \frac{1}{8s'} [(1 + \cos\theta)^2 + (1 - \cos\theta)^2]$$

Z-only:

$$\frac{d\sigma_Z}{d\Omega} = \frac{e^4}{(4\pi)^2} Q_q^2 Q_l^2 \frac{1}{8s'} |\mathcal{R}|^2 [c_{1,Z}(1 + \cos\theta)^2 + c_{2,Z}(1 - \cos\theta)^2]$$

Interférence:

$$\frac{d\sigma_{int}}{d\Omega} = \frac{e^4}{(4\pi)^2} Q_q^2 Q_l^2 \frac{1}{8s'} \text{Re}(\mathcal{R}) [c_{1,int}(1 + \cos\theta)^2 + c_{2,int}(1 - \cos\theta)^2]$$

Voici l'expression des sections efficaces différentielles:

Photon-only:

$$\frac{d\sigma_\gamma}{d\Omega} = \frac{e^4}{(4\pi)^2} Q_q^2 Q_l^2 \frac{1}{8s'} [(1 + \cos\theta)^2 + (1 - \cos\theta)^2]$$

Z-only **ASYMETRIQUE EN  $\theta$** :  $\sigma(\pi - \theta) \neq \sigma(\theta)$ :

$$\frac{d\sigma_Z}{d\Omega} = \frac{e^4}{(4\pi)^2} Q_q^2 Q_l^2 \frac{1}{8s'} |\mathcal{R}|^2 [c_{1,Z}(1 + \cos\theta)^2 + c_{2,Z}(1 - \cos\theta)^2]$$

Interférence **ASYMETRIQUE EN  $\theta$** :  $\sigma(\pi - \theta) \neq \sigma(\theta)$ :

$$\frac{d\sigma_{int}}{d\Omega} = \frac{e^4}{(4\pi)^2} Q_q^2 Q_l^2 \frac{1}{8s'} \text{Re}(\mathcal{R}) [c_{1,int}(1 + \cos\theta)^2 + c_{2,int}(1 - \cos\theta)^2]$$

On peut séparer la section efficace totale en deux parties:

- Avant (Forward):  $\sigma_F = \sigma_{\theta < \pi/2}$ 
  - ▶ Evénements avec le lepton final dans **la même direction** que le quark initial
- Arrière (Backward):  $\sigma_B = \sigma_{\theta > \pi/2}$ 
  - ▶ Evénements avec le lepton final dans **la direction opposée** à celle du quark initial

Et définir l'asymétrie avant-arrière:

$$A_{FB} = \frac{\sigma_F - \sigma_B}{\sigma_F + \sigma_B}$$

# L'asymétrie avant-arrière ( $A_{FB}$ )

Si on exprime  $A_{FB}$  en fonction des variables du problème, on obtient:

$$A_{FB} = \frac{3}{8} \frac{c_2}{c_1}$$

où:

$$c_1 = 1 + 2 \operatorname{Re}(R) g_{VI} g_{Vq} + |R|^2 (g_{VI}^2 + g_{AI}^2) (g_{Vq}^2 + g_{Aq}^2)$$

$$g_{AI,q} = -I_{WI,q}^3$$

$$c_2 = 4 \operatorname{Re}(R) g_{AI} g_{Aq} + 8 |R|^2 g_{VI} g_{AI} g_{Vq} g_{Aq}$$

$$g_{VI,q} = I_{WI,q}^3 - 2 Q_{l,q} \sin^2 \theta_W$$

$$R = \frac{1}{Q_l Q_q \sin^2 2\theta_W} \frac{s'}{s' - M_Z^2 + i s' \Gamma_Z / M_Z}$$

	$Q$	$I_W^3$
$e$	-1	-1/2
$u$	2/3	1/2
$d$	-1/3	-1/2

**Est-ce que  $A_{FB}$  est une constante?**

# L'asymétrie avant-arrière ( $A_{FB}$ )

Si on exprime  $A_{FB}$  en fonction des variables du problème, on obtient:

$$A_{FB} = \frac{3}{8} \frac{c_2}{c_1}$$

où:

$$c_1 = 1 + 2 \operatorname{Re}(R) g_{Vl} g_{Vq} + |R|^2 (g_{Vl}^2 + g_{Al}^2) (g_{Vq}^2 + g_{Aq}^2)$$

$$g_{Al,q} = -I_{Wl,q}^3$$

$$c_2 = 4 \operatorname{Re}(R) g_{Al} g_{Aq} + 8 |R|^2 g_{Vl} g_{Al} g_{Vq} g_{Aq}$$

$$g_{Vl,q} = I_{Wl,q}^3 - 2 Q_{l,q} \sin^2 \theta_W$$

$$R = \frac{1}{Q_l Q_q \sin^2 2\theta_W} \frac{s'}{s' - M_Z^2 + i s' \Gamma_Z / M_Z}$$

	$Q$	$I_W^3$
$e$	-1	-1/2
$u$	2/3	1/2
$d$	-1/3	-1/2

$\Rightarrow A_{FB}$  dépend de  $s'$ , de l'énergie de la collision!

Le but de ces séances d'exercices sera de:

- **Calculer** la valeur théorique de  $A_{FB}$
- **Calculer**  $A_{FB}$  en utilisant des simulations numériques
- **Mesurer**  $A_{FB}$  dans les données du détecteur CMS
- **Comparer** la théorie à la mesure et interpréter les résultats

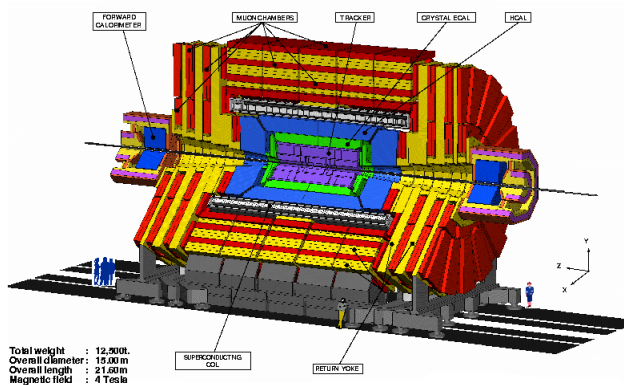


## Section 3

# Le détecteur CMS

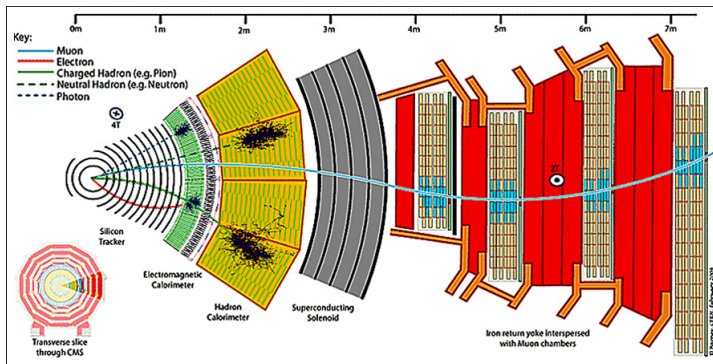
# Le détecteur CMS

CMS  $\equiv$  *Compact Muon Solenoid*



CMS: assemblage de couches (sous-détecteurs) remplissant chacun une tâche.

Voici une vue transverse de CMS:





Simulations			
Information accessible			
Génération des événements (Monte Carlo) Collisions proton-proton	Simulation des interactions particules-CMS Interaction particules-CMS	Reconstruction	Analyse
Information inaccessible	Information accessible		
Données			

Un peu de vocabulaire. Pour les simulations, on parle de:

- *Niveau généré*: niveau d'analyse avant l'interaction des particules avec le détecteur
- *Niveau reconstruit*: niveau d'analyse après la simulation de l'interaction des particules avec le détecteur et le processus de reconstruction



Parler de niveau généré pour les données n'a pas de sens puisqu'on a seulement accès aux informations après interaction des particules avec le détecteur!

## Section 4

### Les outils de base

Vous allez travailler sur une machine virtuelle. Pour y accéder, ouvrez *Virtual Box* et démarrez la session. Votre mot de passe est `labo;123`.

Pour ouvrir un fichier, `emacs` est conseillé:

```
emacs -nw file
```

Quelques astuces:

- Chercher un mot: `ctrl+s` puis tapez le mot dans le *minibuffer* en bas
- Sauver vos modifications: `ctrl+x` puis `ctrl+s` sans lâcher le `ctrl`
- Fermer le fichier: `ctrl+x` puis `ctrl+c` sans lâcher le `ctrl`

- `ls`: *list*, liste les fichiers présents dans le dossier courant
- `cd dir`: *change directory*, rentre dans le dossier `dir`
- `mkdir dir`: *make directory*, crée le dossier `dir`
- `mv file dir`: *move*, bouge le fichier `file` dans le dossier `dir`
- `mv file1 file2`: *move*, renomme le fichier `file1` en `file2`
- `cp file dir`: *copy*, copie le fichier `file` dans le dossier `dir`
- Un point `.` indique le dossier courant
- Deux points `..` indiquent le dossier parent

## Section 5

# Introduction au C++ et à ROOT



En C++, on peut définir des objets auxquels sont associées plusieurs méthodes. On utilise pour cela des classes:

## myObject.h

```
//include here what you need for your methods
#include <iostream>

class myObject{
public:
    //methods declaration here
    void PrintSomething();
    double GiveMeThisNumber(double a);
}; //This ; is MANDATORY
```

## myObject.C

```
#include "myObject.h"

//Define here the methods you declared in the .h

//Do not forget to write "myObject::"
void myObject::PrintSomething(){
    cout<<"Hello!"<<endl;
}

double myObject::GiveMeThisNumber(double a){
    cout<<a<<endl;
    return a;
}
```

La classe *myObject* est définie par:

- `myObject.h`: déclare les méthodes
- `myObject.C`: définit les méthodes pré-déclarées

Pour utiliser la classe *myObject* avec ROOT, suivez les commandes suivantes:

```
root -l
root [0] .L myObject.C++

root [1] myObject jack
root [2] jack.PrintSomething()
Hello!
root [3] jack.GiveMeThisNumber(3.876)
3.876
```

- root [0]: compilation de la classe
- root [1]: création d'un nouvel objet *jack* de la classe *myObject*
- root [2]: appel de la méthode `PrintSomething` de la classe
- root [3]: appel de la méthode `GiveMeThisNumber` de la classe

ROOT organise l'information sous forme d'arbres (on parle de *trees* ou de *ntuples*):

- 1 collision = 1 entrée
- 1 variable = 1 branche (e.g. énergie de l'électron, angle  $\phi$  du muon, etc.)

Exemple d'un tree contenant l'information de 3 évènements et 3 branches (X-pos, Y-pos et Energy):

Entry number	X-pos	Y-pos	Energy
1	13	0	17
2	24	-3	15
3	35	9	8

# Visualiser un ROOT tree

Ouvrez le fichier .root:

```
root -l myfile.root  
TBrowser myBrowser
```

- The TBrowser looks like this:

The screenshot shows the ROOT Object Browser interface. On the left, a tree structure is displayed with 'electron\_theta' selected. On the right, a histogram plot titled 'electron\_theta' is shown, with a 'htemp' table overlaid on it. The table contains the following data:

htemp	
Entries	6.135249e+07
Mean	1.573
RMS	0.9843

Annotations on the left side of the screenshot:

- ROOT file ("TFile") points to the 'root' folder in the tree.
- ROOT ntuple ("TTree") points to the 'electron\_theta' folder in the tree.
- Variables ("TLeaves") points to the list of variables under 'electron\_theta'.

Text overlaid on the histogram: "Double click to see a variable (this is slow!)"

## Section 6

Exercice 1: calculer l'asymétrie  $A_{FB}$  théorique

# Exercice 1: calculer l'asymétrie $A_{FB}$ théorique



Ouvrez la macro: `Compute_AFB.C`

Modifiez-là et complétez-là pour obtenir les histogrammes d'asymétrie avant-arrière dans les deux cas suivants:

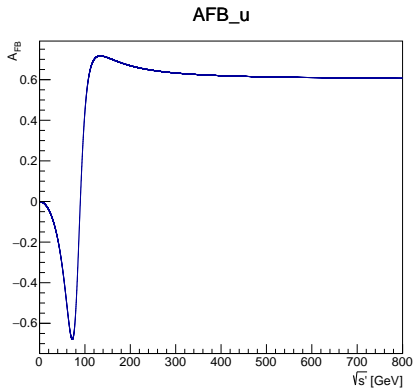
- $u\bar{u} \rightarrow e^+e^-$
- $d\bar{d} \rightarrow e^+e^-$

Plus d'info au début de la macro dans les commentaires. Pour lancer la macro:

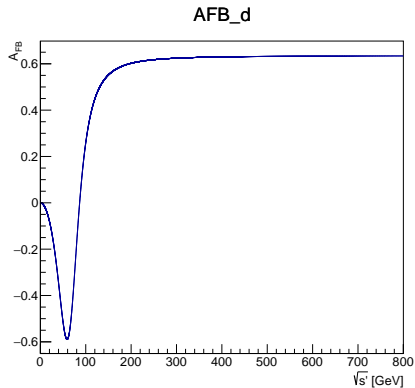
```
root Compute_AFB.C+
```

# Exercice 1: résultat

Vous devez obtenir ces deux histogrammes:



$$u\bar{u} \rightarrow e^+e^-$$



$$d\bar{d} \rightarrow e^+e^-$$

## Section 7

### Exercice 2: simulations Monte Carlo



Analyse de simulations MC à l'aide de la macro Analysis.C.

Analysis.C fait appel à un *header*, Analysis.h où est déterminé quel fichier ROOT est utilisé. A partir de la ligne 106:

```
TChain * chain = new TChain("tree","");  
chain->Add("MC_DY.root");  
tree = chain;
```

Analysis.h liste aussi les variables incluses dans l'arbre, e.g.:

```
vector<float> *electron_pt;  
vector<float> *electron_energy;
```

Vous trouverez ce fichier, ainsi que la version à jour de Analysis.C et Analysis.h sur ma homepage:

<http://t2bweb.iihe.ac.be/~vannerom/PHYSF416/>

- Taille d'un vecteur en C++: `electron_pt->size()`
- Valeur de l'entrée  $i$  d'un vecteur: `electron_pt->at(i)`
- Remplir un histogramme: `Histo->Fill(variable,weight)`
- Somme d'histogrammes  $H = H + H1$  : `H- > Add(H1)`
- Différence d'histogrammes  $H = H - H1$  : `H- > Add(H1, -1)`
- Division d'histogrammes  $H = H/H1$  : `H- > Divide(H1)`

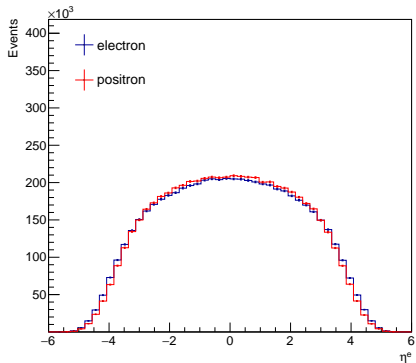
Modifiez la macro pour obtenir les histogrammes suivants au niveau généré:

- Impulsion transverse  $p_T$  des électrons ( $e^- + e^+$ )
- Pseudorapidité  $\eta$  des électrons
  - ▶ Superposez  $\eta^{e^-}$  et  $\eta^{e^+}$  pour:
    - ★  $M(e^+e^-) \in [0, Inf]$  (inclusif)
    - ★  $M(e^+e^-) \in [100, Inf]$
  - ▶ Discutez l'intérêt de ces distributions
- Angle azimuthal  $\phi$  des électrons

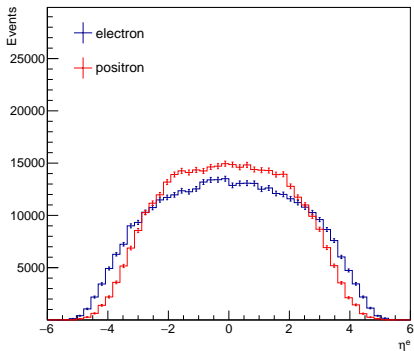
Commentez ces histogrammes (valeurs limites, forme (*shape*)).

Pour lancer la macro

- `root`
- `.L Analysis.C`
- `Analysis test`
- `test.Loop()`



Inclusif



$M(e^+e^-) > 100\text{GeV}$

Construisez ensuite le quadrivecteur de la paire  $e^+e^-$ . Obtenez les histogrammes suivants au niveau généré:

- Masse invariante  $M(e^+e^-)$  de la paire  $e^+e^-$
- Impulsion longitudinale  $p_Z(e^+e^-)$  de la paire  $e^+e^-$

Construisez également  $M(e^+e^-)$  au niveau reconstruit et superposez-le à l'histogramme au niveau généré.

Pour lancer la macro

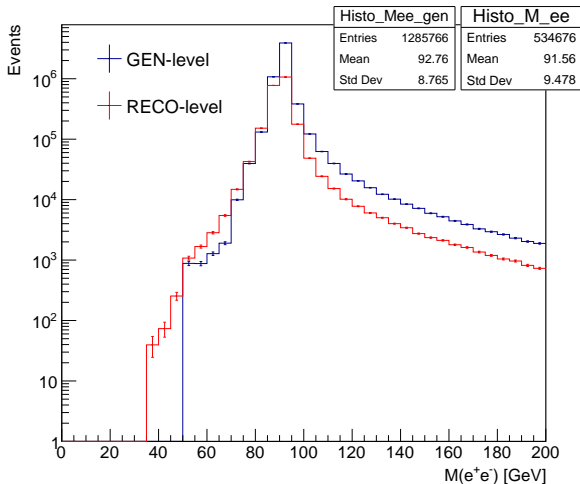
- `root`
- `.L Analysis.C`
- `Analysis test`
- `test.Loop()`

# La masse invariante $e^+e^-$ : résultat

Vous devez obtenir ces histogrammes:

Expliquez les différences:

- Nombre d'événements
- Largeur du pic
- Bornes

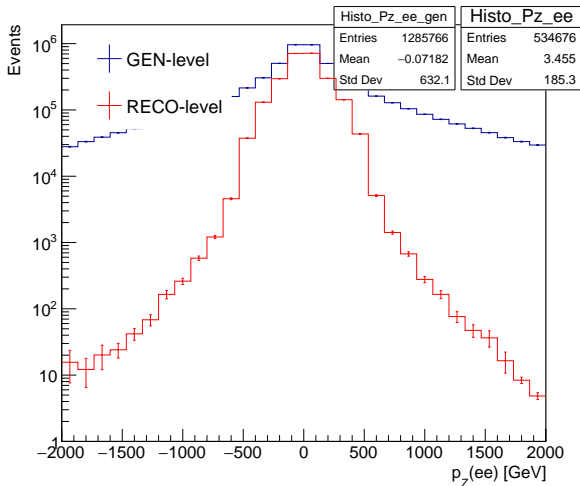


# L'impulsion longitudinale $e^+e^-$ : résultat

Vous devez obtenir ces histogrammes:

Expliquez les différences:

- Nombre d'événements
- Largeur du pic
- Bornes



Nous allons calculer  $A_{FB}$  de quatre façons différentes dans les simulations:

	GEN-GEN	GEN-RECO	RECO-GEN	RECO-RECO
Boost du quark	quark	quark	Z	Z
Electrons	GEN	RECO	GEN	RECO

La première partie du nom se réfère à l'information sur le boost longitudinal ( $p^z$ ) du quark:

- quark: on utilise la vraie direction du quark
- Z: on suppose que le quark et le boson Z (le système  $e^+e^-$ ) sont boostés dans la même direction

La seconde partie du nom se réfère au niveau de reconstruction de l'électron:

- GEN: on utilise les électrons au niveau généré
- RECO: on utilise les électrons au niveau reconstruit



# L'asymétrie avant-arrière

Voici la procédure GEN-GEN. Pour chaque événement:

- Déterminez la direction longitudinale du quark ( $\text{sign}(p^z)$ )
- Déterminez l'angle  $\theta$  de l'électron dans le système du CM
  - ▶ Construisez le système  $e^+e^-$ 
    - ★ TLorentzVector v\_ele, v\_pos;
    - ★ v\_ele.SetPtEtaPhiE(ele\_pt, ele\_eta, ele\_phi, ele\_E);
    - ★ v\_ele+v\_pos;
  - ▶ Boostez les particules dans le référentiel du centre-de-masse
    - ★ v\_ele.Boost(-(v\_ele+v\_pos).BoostVector());
  - ▶ Récupérez l'angle  $\theta$  de l'électron dans le CM
    - ★ v\_ele.Theta();
- Déduisez-en l'angle que fait l'électron avec le quark dans le référentiel du centre-de-masse
- En fonction de cet angle, remplissez un histogramme "forward"  $H_F$  ou "backward"  $H_B$
- Construisez l'histogramme d'asymétrie:  $H_{AFB} = \frac{H_F - H_B}{H_F + H_B}$

⇒ **Réitérez pour les autres cas!**

# L'asymétrie avant-arrière: résultats

Voici les quatre histogrammes que vous devez obtenir:

