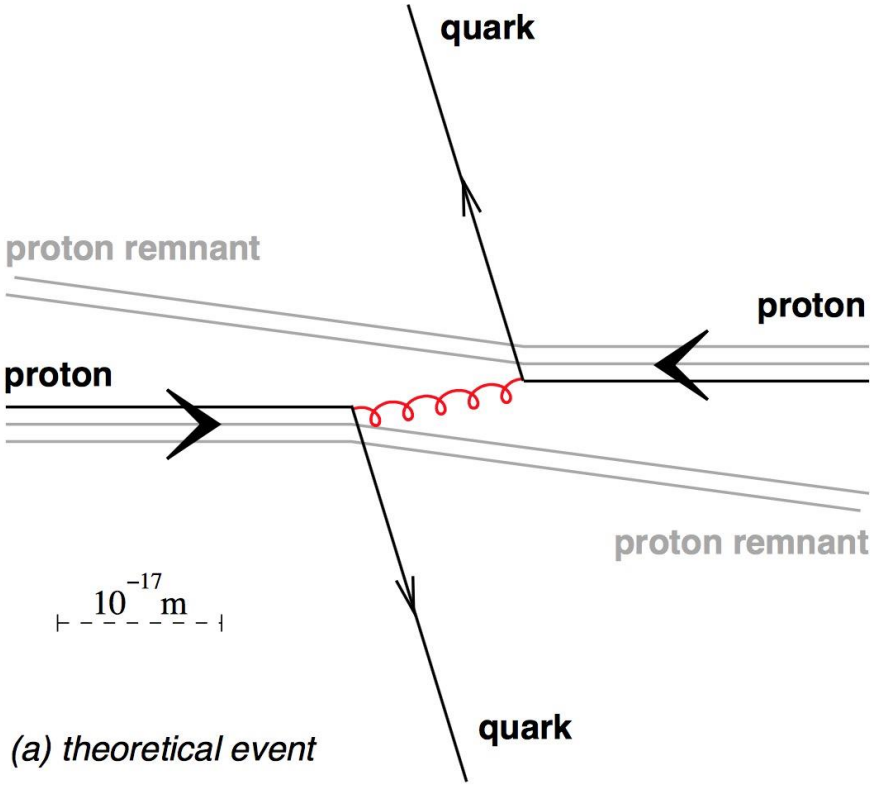
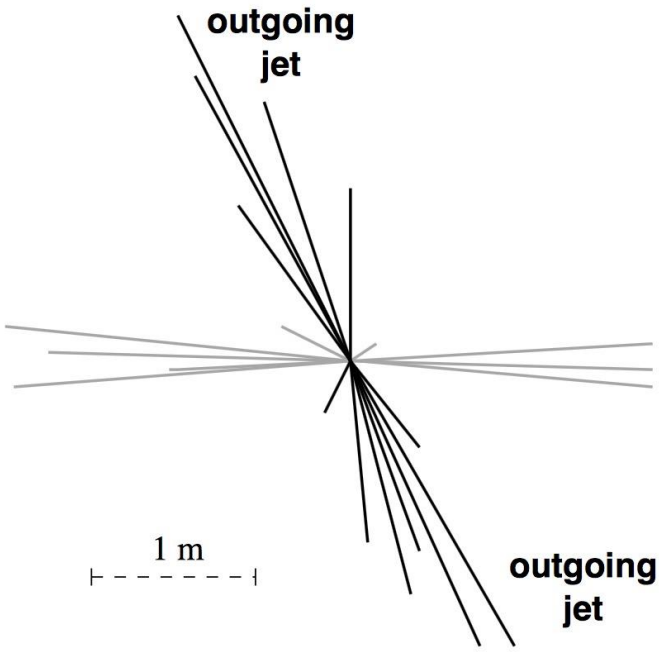
$\mu^+\mu^-$ Recoil Mass Measurement

Chen Zhenxing (PKU&IHEP)

pp collision



(a) theoretical event

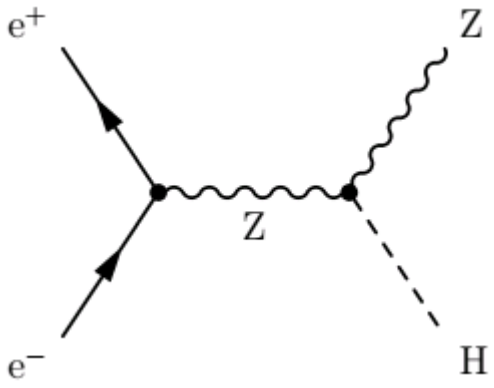


(b) schematic view of real event

Parton collision, CMS energy is not definite

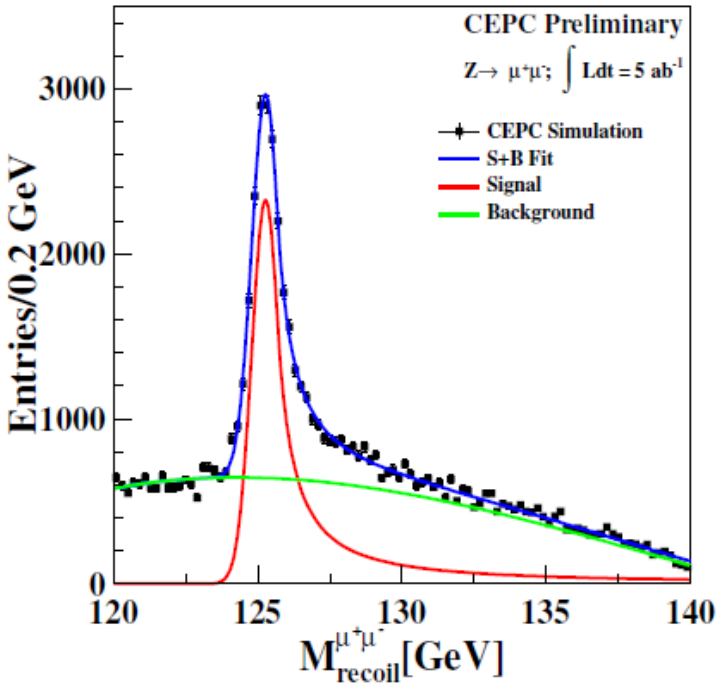
e^+e^- collision

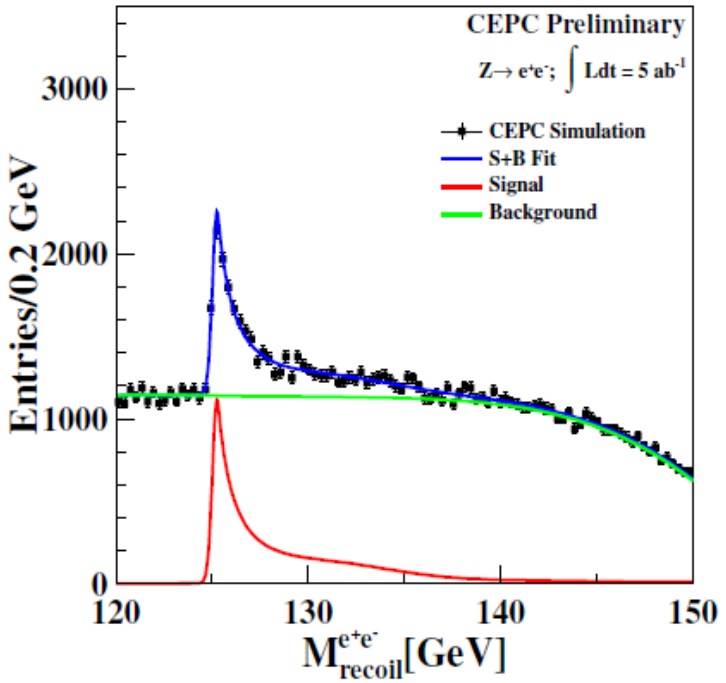
-recoil mass method



Z: 10% l^+l^-
20% invisible
70% qq

$$m^2_{\text{recoil}} = (\sqrt{s} - E_{f\bar{f}})^2 - p^2_{f\bar{f}} = s - 2E_{f\bar{f}}\sqrt{s} + m^2_{f\bar{f}}$$





strategy

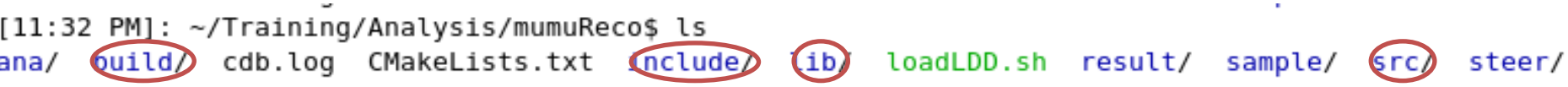
1. Tag a $\mu^+\mu^-$ pair whose invariant mass is nearest to Z boson

   …….

2. The recoil mass forms a peak at the position of Higgs

Get ready

$ cd /home/ihep/Training
$ source env_ilcsoft.sh
$ cd Analysis/mumuReco
$ ls



build: not exists for a new package, remove it
$ rm –fr build

Source code

$ cd src
$ vi HiggsRecoil.cc

```
#include <HiggsRecoil.hh>
#include <EVENT/LCCollection.h>
#include <IMPL/LCCollectionVec.h>
#include <EVENT/LCFloatVec.h>
#include <EVENT/MCParticle.h>
#include <EVENT/ReconstructedParticle.h>
#include <IMPL/MCParticleImpl.h>
#include <values.h>
#include <string>
#include <iostream>
#include <EVENT/LCFloatVec.h>
#include <EVENT/LCParameters.h>
#include <stdexcept>
#include <TFile.h>
#include <TTree.h>
#include <TVector3.h>
#include <TRandom.h>
#include <Rtypes.h>
#include <sstream>
#include <cmath>
#include <vector>
#include <TMath.h>
#include "TLorentzVector.h"
```

include head files

Source code

```
HiggsRecoil::HiggsRecoil()
        : Processor("HiggsRecoil"),
        _output(0)
{
        _description = "Print MC Truth" ;

        _treeFileName="MCTruth.root";
        registerProcessorParameter( "TreeOutputFile" ,
                        "The name of the file to which the ROOT tree will be written" ,
                        _treeFileName ,
                        _treeFileName);

        _colName="MCParticle";
        registerProcessorParameter( "MCObjects" ,
                        "The name of the PFOs" ,
                        _colName ,
                        _colName);

        _treeName="MCPart";
        registerProcessorParameter( "TreeName" ,
                        "The name of the ROOT tree" ,
                        _treeName ,
                        _treeName);

        _leptonID = 13;
        registerProcessorParameter( "LeptonIDTag" ,
                        "Lepton ID that will be used in this analysis." ,
                        _leptonID ,
                        _leptonID);

        _overwrite=0;
        registerProcessorParameter( "OverwriteFile" ,
                        "If zero an already existing file will not be overwritten." ,
                        _overwrite ,
                        _overwrite);
```

Source code

```
void HiggsRecoil::init() {

        printParameters();

        TFile *tree_file=new TFile(_treeFileName.c_str(),(_overwrite ? "RECREATE" : "UPDATE"));

        if (!tree_file->IsOpen()) {
                delete tree_file;
                tree_file=new TFile(_treeFileName.c_str(),"NEW");
        }

        _outputTree = new TTree(_treeName.c_str(),_treeName.c_str());
        _outputTree->SetAutoSave(32*1024*1024);  // autosave every 32MB
        _outputTree->Branch("EventNr", &_eventNr, "EventNr/I");
        _outputTree->Branch("Num", &_Num, "Num/I");

        _outputTree->Branch("ZdaughterPID",&_ZdaughterPID,"ZDauPID/I");
        _outputTree->Branch("HdaughterPID",&_HdaughterPID,"HDauPID/I");

        _outputTree->Branch("Zindex",&_Zindex,"Zindex/I");
        _outputTree->Branch("Windex",&_Windex,"Windex/I");

        _outputTree->Branch("PzD1", _PzD1, "PzD1[4]/F");
        _outputTree->Branch("PzD2", _PzD2, "PzD2[4]/F");

        _outputTree->Branch("PhD1", _PhD1, "PhD1[4]/F");
        _outputTree->Branch("PhD2", _PhD2, "PhD2[4]/F");

        _outputTree->Branch("Pz",_Pz,"Pz[4]/F");
        _outputTree->Branch("Mz", &_Mz, "Mz/F");
        _outputTree->Branch("Ph",_Ph,"Ph[4]/F");
        _outputTree->Branch("NHDaug", &_NHDaug, "NHDaug/I");

        _outputTree->Branch("NNeutrino", &_NNeutrino, "NNeutrino/I");
        _outputTree->Branch("ENeutrino", &_ENeutrino, "ENeutrino/F");
```

Source code

include/HiggsReco.hh

```
std::string _treeFileName;
std::string _treeName;
std::string _colName;
std::string _colAdcVals;

int _overwrite, _leptonID;
float _cmsE;
TTree *_outputTree;

int _NMuP, _NMuM, _NChP, _NChM, _NCh, _NLep, _NLepCh;
float _P_MuP[4], _P_MuM[4], _P_DL[4];
int _EventType;
float _InvMass, _RecoilMass;

int _PID1, _PID2;
float _PL1[4], _PL2[4], _RPL1[4], _RPL2[4], _SM[4], _P_allCharged[4], _P_allNeutral[4], _P_Higgs[4], _P_allReco[4], _P_Photon[4], _P_allLepton[4];
float _Hmass;
int _Num;
int _NHDaug;
int _HdaughterPID;
int _ZdaughterPID;
int _Zindex;
int _Windex;
float _Pz[4], _Ph[4], _PzD1[4], _PzD2[4], _PhD1[4], _PhD2[4], _RPzD1[4], _RPzD2[4], _RPhD1[4], _RPhD2[4];
float _P[4], _SumP[4], _VisP[4], _MissP[4];
int _PID, _NFMCP, _MotherFlag, _NNeutrino;
float _ENeutrino, _DiPhMass, _DiPhMassCorr;
float _CosTheta, _Phi, _Charge;
float _Mz, _Mrecoil, _MzReco, _MhReco, _MrecoilReco;
float _Pt_Z, _Pt_photon, _cosZ, _acol, _acop, _DeltaPt;
```

Source code

$LCIO/include/EVENT

```
void HiggsRecoil::processEvent( LCEvent * evtP )
{

        if (evtP)
        {
                try
                {
                        LCCollection* col_MCP = evtP->getCollection( _colName ) ;
                        LCCollection* col_RecoP = evtP->getCollection( "ArborPFOs" );

                        int _nMCP=col_MCP->getNumberOfElements();
                        int _nRecoP = col_RecoP->getNumberOfElements();

                        _eventNr=evtP->getEventNumber();

                        int tmpPID = 0;
                        int NParent = 0;
                        int NDaughter = 0;
                        float MCPEn = 0;

                        _Hmass = 0;
                        _Mz = 0;
                        _NMuP = 0;
                        _NMuM = 0;
                        _NLep = 0;
                        _NLepCh = 0;
                        _NChP = 0;
                        _NChM = 0;
```

```
for(int j = 0; j < _nRecoP; j++)
{
        ReconstructedParticle *a_RecoP = dynamic_cast<EVENT::ReconstructedParticle *>(col_RecoP->getElementAt(j));
        if(a_RecoP->getCharge()==0) continue;
        RecoPID = a_RecoP->getType();
        RecoE = a_RecoP->getEnergy();
        RecoP[0] = a_RecoP->getMomentum()[0];
        RecoP[1] = a_RecoP->getMomentum()[1];
        RecoP[2] = a_RecoP->getMomentum()[2];

        TLorentzVector currP(RecoP[0], RecoP[1], RecoP[2], RecoE);

        if(RecoE>2.0) _NCh++;


        for(int s = 0; s < 4; s++)
        {
                _P_allCharged[s] += currP[s];
        }

        if( RecoE > 2)
        {
                if(abs(RecoPID) == 11 || abs(RecoPID) == 13) // loop all reconstructed particles to look for         isolated leptons...
                {
                   double coneE=0.;
                   for(int i1 = 0; i1 < _nRecoP; i1++)
                   {
                      ReconstructedParticle *b_RecoP = dynamic_cast<EVENT::ReconstructedParticle *>(col_RecoP->getElementAt(i1));
                      TLorentzVector tmp(b_RecoP->getMomentum()[0],b_RecoP->getMomentum()[1],b_RecoP->getMomentum()[2],b_RecoP->getEnergy());
                      if(currP.DeltaR(tmp)<0.15)
                          coneE+=b_RecoP->getEnergy();
                   }
                  if( coneE/RecoE < 1.1 ) // ratio
                  {
                      _NLep++;
                      _NLepCh =+ a_RecoP->getCharge();
                      _P_allLepton[3] += a_RecoP->getEnergy();
                      _P_allLepton[0] += a_RecoP->getMomentum()[0];
                      _P_allLepton[1] += a_RecoP->getMomentum()[1];
                      _P_allLepton[2] += a_RecoP->getMomentum()[2];
                  }
                  if(abs(RecoPID) == _leptonID )        //Put by hand... guess enough
                  {
                       if(RecoPID == _leptonID )        //Got swapped...gosh!
                       {
                               FourMom_MuonM.push_back(currP);
                       }
                       else
                       {
                               FourMom_MuonP.push_back(currP);
                       }
                  }
                  else if( a_RecoP->getCharge() > 0.5 )
                  {
                       P_ChP.push_back(currP);
                  }
                  else if( a_RecoP->getCharge() < -0.5 )
                  {
                       P_ChM.push_back(currP);

                  }
                }
        }
}
```

Source code

```
09:32 AM]: ~/Training$ cd $LCIO/include/IMPL
09:32 AM]: ~/ilcsoft/v01-17-05/lcio/v02-04-03/include/IMPL$ ls
AccessChecked.h       LCEventImpl.h           LCParametersImpl.h  MCParticleImpl.h             SimCalorimeterHitImpl.h  TrackerHitImpl.h          TrackerRawDataImpl.h
CalorimeterHitImpl.h  LCFlagImpl.h            LCRelationImpl.h    ParticleIDImpl.h             SimTrackerHitImpl.h      TrackerHitPlaneImpl.h     TrackImpl.h
ClusterImpl.h         LCGenericObjectImpl.h   LCRunHeaderImpl.h   RawCalorimeterHitImpl.h      TPCHitImpl.h             TrackerHitZCylinderImpl.h TrackStateImpl.h
LCCollectionVec.h     LCIOExceptionHandler.h  LCTOOLS.h           ReconstructedParticleImpl.h  TrackerDataImpl.h        TrackerPulseImpl.h        VertexImpl.h
```

```
/** Type of reconstructed particle.
 *  Check/set collection parameterrs ReconstructedParticleTypeNames and
 *  ReconstructedParticleTypeValues.
 */
virtual int getType() const ;

/** Convenient method - same as (getParticles().size() > 0 ).
 */
virtual bool isCompound() const { return getParticles().size() > 0 ; }

/** The magnitude of the reconstructed particle's momentum,
 */
virtual const double* getMomentum() const ;

/** Energy of the  reconstructed particle.
 */
virtual double getEnergy() const ;

/** Covariance matrix of the reconstructed particle's 4vector (10 parameters).
 */
virtual const EVENT::FloatVec & getCovMatrix() const ;

/** Mass of the  reconstructed particle, set independently from four vector quantities.
 */
virtual double getMass() const ;

/** Charge of the reconstructed particle.
 */
virtual float getCharge() const ;
```

Source code

```
if( NCandiP > 0 && NCandiM > 0 )
{
        for(int p = 0; p < NCandiP; p++)
        {
                P_P = CandiP[p];

                for(int m = 0; m < NCandiM; m++)
                {
                        P_M = CandiM[m];

                        currInvMass = (P_P + P_M).M();

                        if(fabs(currInvMass - 91.2) < MinZThrDis)
                        {
                                MinZThrDis = fabs(currInvMass - 91.2);
                                _InvMass = currInvMass;
                                currRecoilMass = (P_T - P_P - P_M).M();
                                _RecoilMass = currRecoilMass;
                                for(int s = 0; s < 4; s++)
                                {
                                        _P_MuP[s] = P_P[s];
                                        _P_MuM[s] = P_M[s];
                                        _P_DL[s] = _P_MuP[s] + _P_MuM[s];
                                }
                                _acop = fabs(P_P.Phi()-P_M.Phi());
                                _acol = P_P.Angle(P_M.Vect())*180./3.1415926;
                                _Pt_Z = sqrt(_P_DL[0]*_P_DL[0]+_P_DL[1]*_P_DL[1]);
                                _DeltaPt = _Pt_Z - _Pt_photon;
                                _cosZ = _P_DL[2]/sqrt(_P_DL[0]*_P_DL[0]+_P_DL[1]*_P_DL[1]+_P_DL[2]*_P_DL[2]);
                        }
                }
        }
}
```

Source code

```
for(int i = 0; i < _nMCP; i++)
{
        MCParticle *a1_MCP = dynamic_cast<EVENT::MCParticle *>(col_MCP->getElementAt(i));
        tmpPID = a1_MCP->getPDG();
        NParent = a1_MCP->getParents().size();
        NDaughter = a1_MCP->getDaughters().size();
        MCPEn = a1_MCP->getEnergy();

        if(NParent == 0 && NDaughter == 1 && abs(tmpPID) < 20 ) //Including all Z decay
        {
                _ZdaughterPID = abs(tmpPID);
                _Pz[3] += MCPEn;
                _Pz[0] += a1_MCP->getMomentum()[0];
                _Pz[1] += a1_MCP->getMomentum()[1];
                _Pz[2] += a1_MCP->getMomentum()[2];
                CurrPMomentum = a1_MCP->getMomentum();

                if(_ZdaughterPID < 6 || _ZdaughterPID == 13 || _ZdaughterPID == 11 || _ZdaughterPID == 15)
                {
                        if(tmpPID > 0)
                        {
                                _PzD1[0] = a1_MCP->getMomentum()[0];
                                _PzD1[1] = a1_MCP->getMomentum()[1];
                                _PzD1[2] = a1_MCP->getMomentum()[2];
                                _PzD1[3] = MCPEn;
                        }
                        else if(tmpPID < 0)
                        {
                                _PzD2[3] = MCPEn;
                                _PzD2[0] = a1_MCP->getMomentum()[0];
                                _PzD2[1] = a1_MCP->getMomentum()[1];
                                _PzD2[2] = a1_MCP->getMomentum()[2];
                        }
                }
        }
```

Source code

```
void HiggsRecoil::end()
{

        if (_outputTree) {

                TFile *tree_file = _outputTree->GetCurrentFile(); //just in case we switched to a new file
                tree_file->Write();
                delete tree_file;
        }

}
```

Compile

```
$ cd ..
$ mkdir build
$ HFcmake
$ make install
$ cd ..
$ source loadLDD.sh
```

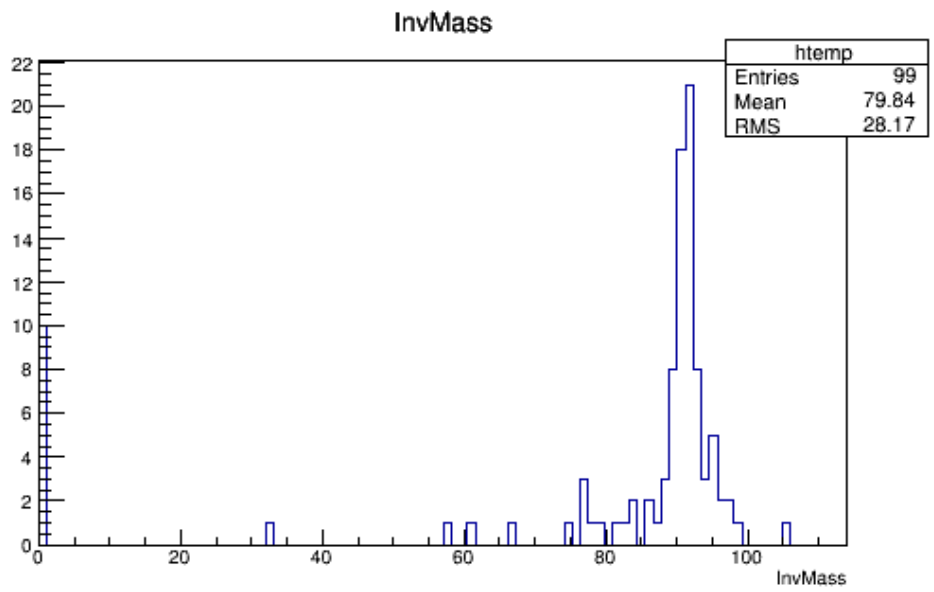
Run

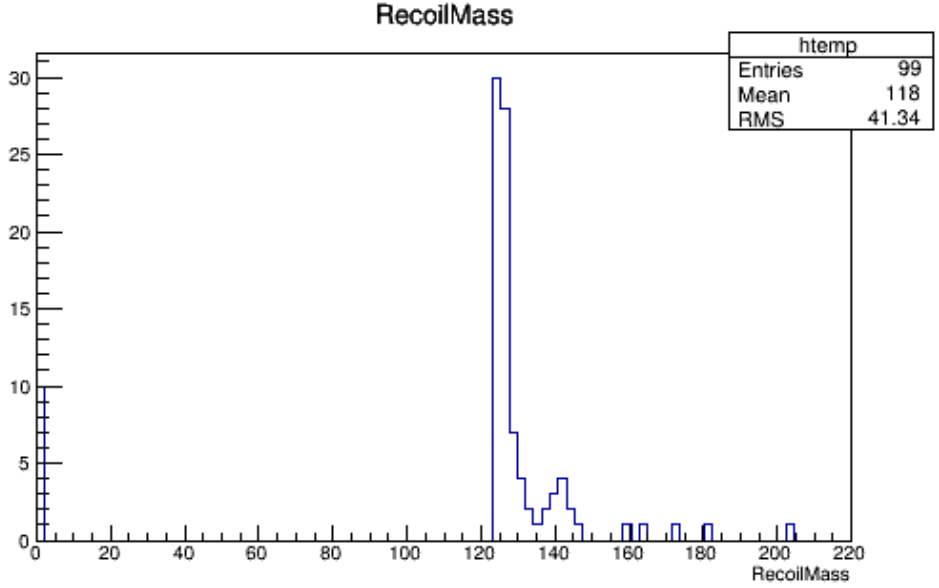
$ cd steer
$ vi Higgs.steer
$ Marlin Higgs.steer

```
.begin Global  --------------------------------------

LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_1.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_2.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_3.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_4.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_5.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_6.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_7.slcio
LCIOInputFiles ../sample/signal/E250-CDR_ws.Pe2e2h.eU.pU.dst_8.slcio


ActiveProcessors MyHiggsRecoil
MaxRecordNumber 100

.end   ----------------------------------------------

.begin MyHiggsRecoil
ProcessorType HiggsRecoil

#       The name of the PFOs
#        type:  [string]
#        default: MCParticle
        MCObjects   MCParticle

    OverwriteFile   0

    TreeName   MCPart

    LeptonIDTag 13

    TreeOutputFile   ../result/signal/signal.root
.end ------------------------------------------------
```

Result

$ cd ../result/signal
$ root –l signal.root
$ TBrowser a





Event selection

Category	$Signal$	ZZ	WW	$ZZorWW$
Total	35110	5499688	44418522	17864598
$N_{\mu^+} \geq 1, N_{\mu^-} \geq 1$	32043	647605	291356	726875
$120GeV < M_{recoil} < 160GeV$	31112	99524	104257	131815
$80GeV < M_{\mu^+\mu^-} < 100GeV$	28516	39771	29618	42234
$P_{T\mu^+\mu^-} > 20GeV$	26771	31251	25983	32393
$\lvert\phi_{\mu^+} - \phi_{\mu^-}\rvert < 175°$	25982	29573	24545	31490
$BDT > -0.05$	21758	13759	6153	8475

Category	Z	W	$ZorW$	$Z(2f)$
Total	7788916	17508000	1246941	406992159
$N_{\mu^+} \geq 1, N_{\mu^-} \geq 1$	761948	106	0	4875619
$120GeV < M_{recoil} < 160GeV$	151515	20	0	531803
$80GeV < M_{\mu^+\mu^-} < 100GeV$	17764	0	0	320791
$P_{T\mu^+\mu^-} > 20GeV$	12984	0	0	59267
$\lvert\phi_{\mu^+} - \phi_{\mu^-}\rvert < 175°$	12361	0	0	45522
$BDT > -0.05$	4133	0	0	15992

Fit

$ cd ../../ana/fit
$ root –l fit.cxx