Higgs2zz.cc

```
}

void Higgs2zz::saveFastJet( LCCollection* col_Jets , LCCollection* col_rec1 , LCCollection* col_rec2  ) {

 int ncol = col_Jets->getNumberOfElements();
 //std::cout << "nFastJet = " << ncol << std::endl;

 // Save the number of reconstructed particles for each jet
 m_fastjet_npfo1 = col_rec1->getNumberOfElements();
 m_fastjet_npfo2 = col_rec2->getNumberOfElements();

 m_nfastjet = ncol;
 for( int i = 0; i < ncol; i++) {

  ReconstructedParticle *reco_jet = dynamic_cast<EVENT::ReconstructedParticle *>(col_Jets->getElementAt(i));
  m_fastjet_px.push_back( reco_jet->getMomentum()[0] );
  m_fastjet_py.push_back( reco_jet->getMomentum()[1] );
  m_fastjet_pz.push_back( reco_jet->getMomentum()[2] );

  std::cout << "EjetMin = " << EjetMin << std::endl;
  //if( reco_jet->getEnergy() > EjetMin )
    m_fastjet_e.push_back( reco_jet->getEnergy() );
 }

 if( m_nfastjet == 2 ) {

  TLorentzVector jet1, jet2;

  ReconstructedParticle *reco_jet1 = dynamic_cast<EVENT::ReconstructedParticle *>(col_Jets->getElementAt(0));
  ReconstructedParticle *reco_jet2 = dynamic_cast<EVENT::ReconstructedParticle *>(col_Jets->getElementAt(1));

  jet1[0] = reco_jet1->getMomentum()[0];
  jet1[1] = reco_jet1->getMomentum()[1];
  jet1[2] = reco_jet1->getMomentum()[2];
  jet1[3] = reco_jet1->getEnergy();
```

test

-- **Higgs2zz.cc** 56% (1045,0) (C++/l Abbrev)----------

```
Higgs2zz::Higgs2zz() : Processor("Higgs2zz") {

 registerProcessorParameter( "TreeOutputFile" , "Output Filename" , FileName , std::string("default.root"));

 registerProcessorParameter( "TreeName" , "ROOT tree name" , TreeName , std::string("tree"));

 registerProcessorParameter( "Ecms" , "ECMS value in GeV." , Ecms , 240.0);

 registerProcessorParameter( "EcmsOption" , "Use ECMS (1) or with Gaussian Blur (0)." , EcmsOption , 1);

 registerProcessorParameter( "OverwriteFile" , "Re-create (1) or Update (0)" , OverWrite , 1 );

 registerProcessorParameter( "LeptonEMax" , "Maximum energy of leptons in GeV" , LeptonEMax , 100.0);

 registerProcessorParameter( "LeptonEMin" , "Minimum energy of leptons in GeV" , LeptonEMin , 10.0);

 registerProcessorParameter( "LeptonPtMin" , "Minimum Pt of leptons in GeV" , LeptonPtMin , 0.0);

 registerProcessorParameter( "EjetMin" , "Minimum energy of leptons in GeV" , EjetMin , 5.0);
}
```

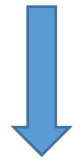
```
// output rootfile
TFile *m_file;
TTree *m_tree;

TH1F* h_evtflw;
TH1F* h_mc_init_plist;
TH1F* h_mc_higgs_dlist;

// paramter setting
int    OverWrite;      // tree overwrite opti
int    EcmsOption;
double Ecms;
double LeptonEMax;
double LeptonEMin;
double LeptonPtMin;
double EjetMin;

TLorentzVector P4_Ecms;
```

template job file

```
 <global>
  <parameter name="LCIOInputFiles">
$slcio_in_list
  </parameter>
  <!-- limit the number of processed records (run+evt): -->
  <parameter name="MaxRecordNumber"> $max_event_num </parameter>
  <parameter name="SkipNEvents" value="0" />
  <parameter name="SupressCheck" value="false" />
  <parameter name="GearXMLFile">
$gear_xml
  </parameter>
  <parameter name="Verbosity" options="DEBUG0-4,MESSAGE0-4,WARNING0-4,ERROR0-4,SILENT"> DEBUG  </parameter>
 </global>



<processor name="MyHiggs2zz" type="Higgs2zz">

  <parameter name="OverwriteFile" type="int"> 1 </parameter>
  <parameter name="Ecms" type="double"> $ecms </parameter>
  <parameter name="EcmsOption" type="int"> 1 </parameter>
  <parameter name="TreeName" type="string"> tree </parameter>
  <parameter name="LeptonEMax" type="double"> 100.0 </parameter>
  <parameter name="LeptonEMin" type="double"> 10.0 </parameter>
  <parameter name="LeptonPtMin" type="double"> 0.0 </parameter>
  <parameter name="TreeOutputFile" type="string">
$output_rootfile
  </parameter>

  <parameter name="EjetMin" type="double"> 3.0 </parameter>
</processor>



<processor name="MyFastJetClustering" type="FastJetProcessor">
  <parameter name="recParticleIn" type="string" lcioOutType="ReconstructedParticle"> WithoutIsoLeps </parameter>
  <parameter name="jetOut" type="string" lcioOutType="ReconstructedParticle">FastJets </parameter>
  <parameter name="recParticleOut1" type="string" lcioOutType="ReconstructedParticle"> FJPList1 </parameter>
  <parameter name="recParticleOut2" type="string" lcioOutType="ReconstructedParticle"> FJPList2 </parameter>
  <parameter name="algorithm" type="StringVec">ee_kt_algorithm  </parameter>
  <parameter name="recombinationScheme" type="string"> E_scheme </parameter>
--:---  template_jobfile.xml   5% (50,59)    (XML)----------
```

```
    Col_MC     = evt->getCollection( "MCParticle"     );
    checkGenMCInfo( Col_MC );

    //Col_Jets   = evt->getCollection( "RefinedJets"    );  // LCFIplus
    Col_FastJet = evt->getCollection( "FastJets"      );   // FastJet
    Col_FJPList1 = evt->getCollection( "FJPList1" );       // FastJet, List of rec. particle list for jet1
    Col_FJPList2 = evt->getCollection( "FJPList2" );       // FastJet, List of rec. particle list for jet2
    Col_Leps    = evt->getCollection( "IsoLeps"       );
    Col_WoLeps  = evt->getCollection( "WithoutIsoLeps" );
    Col_Reco    = evt->getCollection( "ArborPFOs"     );

    h_evtflw->Fill(1); // Does collection exist or not

    if( buildHiggsToZZ() ) {

     m_n_col_reco = Col_Reco->getNumberOfElements();

     // Save variables
     saveNeutral( Col_Reco );
     savePhotons( Col_Reco );
     selectCharged( Col_Reco );

     saveLeptonInfo( P4_MuonPlus, P4_MuonMinus );
     saveJetInfo( P4_Jet );
     saveVariables();

     saveFastJet( Col_FastJet , Col_FJPList1, Col_FJPList2 ); // save FastJet info.

     // Save MC Truth info.
     saveMCTruthInfo( Col_MC );

     // Fill to the tree
     m_tree->Fill();
    }
   }
   catch (lcio::DataNotAvailableException err) { }
  }

}


bool Higgs2zz::buildHiggsToZZ() {
--- Higgs2zz.cc    18% (422,86)   (C++/l Abbrev)----------
```

modification example is in this function



if the cut (npfo/minE) should be applied at jet selection condition, that's should be done inside of "seletJets"

```
bool Higgs2zz::buildHiggsToZZ() {

 // Select Lepton pairs
 selectLeptons( Col_Leps );
 if( m_n_muon_plus==0 || m_n_muon_minus==0 || m_n_lepton!=2 ) return false;

 h_evtflw->Fill(2); // N_{MuonP} > 0 && N_{MuonM} > 0

 // Select Jet pairs
 selectJets( Col_FastJet );
 if( m_n_jet!=2 ) return false;

 h_evtflw->Fill(3); // N_{Jet} == 2

 return true;
}
```