Strategy for the total energy reconstruction in the calorimeter

Francesca and Lorenzo

Energy reconstruction

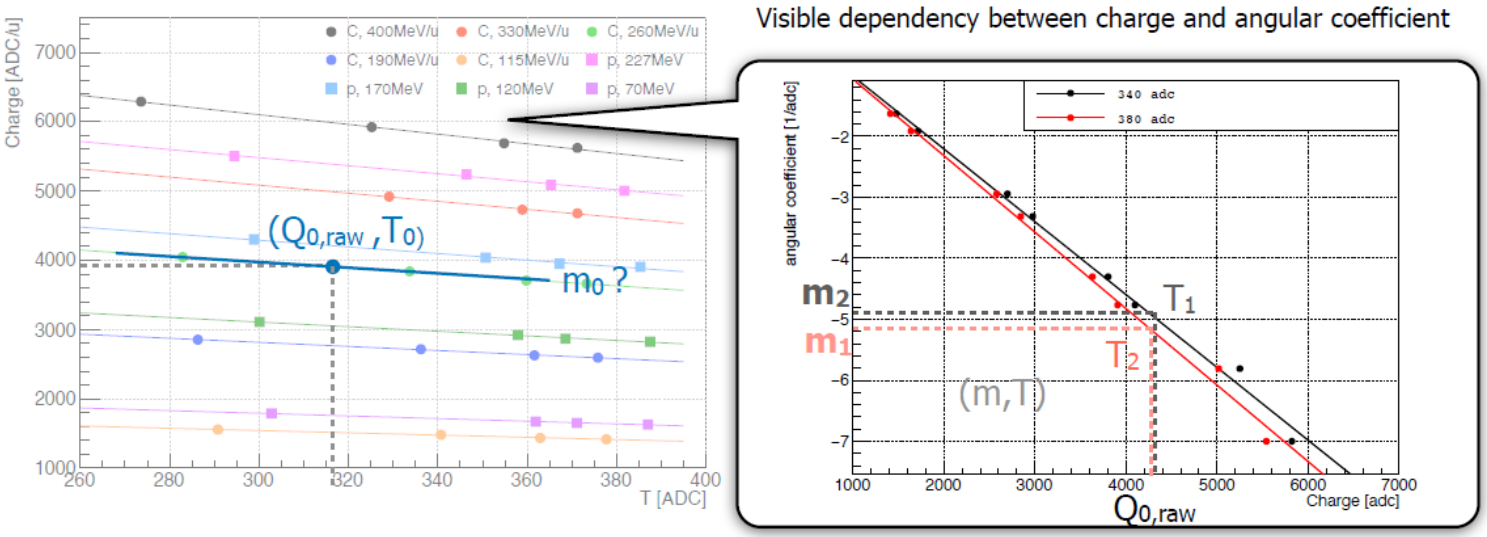
- For each event the raw charge deposited in a crystal is converted in energy according to the following steps:

1. Temperature correction: charge -> charge_tcorr
2. Crystal equalizations: charge_tcorr -> charge_tcorr_eq
3. Range correction: charge_tcorr_eq -> charge_tcorr_eq_pcorr
4. Conversion from charge to energy: charge_tcorr_eq_pcorr -> energy

Temperature correction

- For the moment the temperature reading is not integrated in the DAQ but read from an external file. We will modify as soon as the integration will be ready.

```
//-------------------------------------------+-----------------------------------
Double_t TACAactNtuHit::GetTemperatureCorrection(Double_t charge, Int_t  crysId)
{
  TACAparCal* parcal = (TACAparCal*) fpParCal->Object();


  Double_t T0 = parcal->GetTemperatureCry(crysId);

  Double_t m1 = fTcorr1->Eval(charge);
  Double_t m2 = fTcorr2->Eval(charge);

  Double_t m0 = m1 + ((m2-m1)/(fT2-fT1))*(T0-fT1);

  Double_t delta = (fT1 - T0) * m0;

  Double_t charge_tcorr = charge + delta;

 // return charge; //ricordarsi di cambiare quando avremo la calibrazione!!!!!!

  return charge_tcorr;

}
```

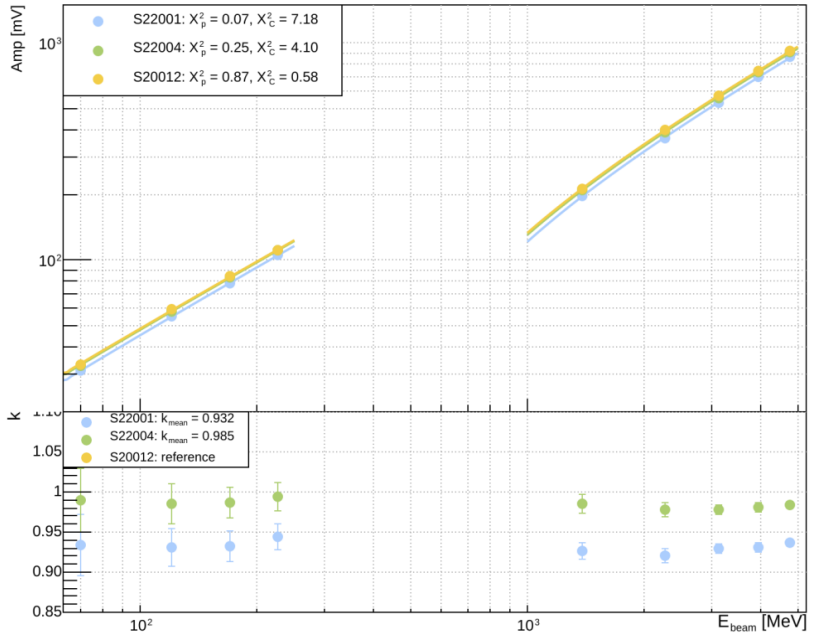


Visible dependency between charge and angular coefficient

$$Q_0 = Q_{0,raw} + m_0 \cdot (T_{work} - T_0)$$

Crystal equalizations

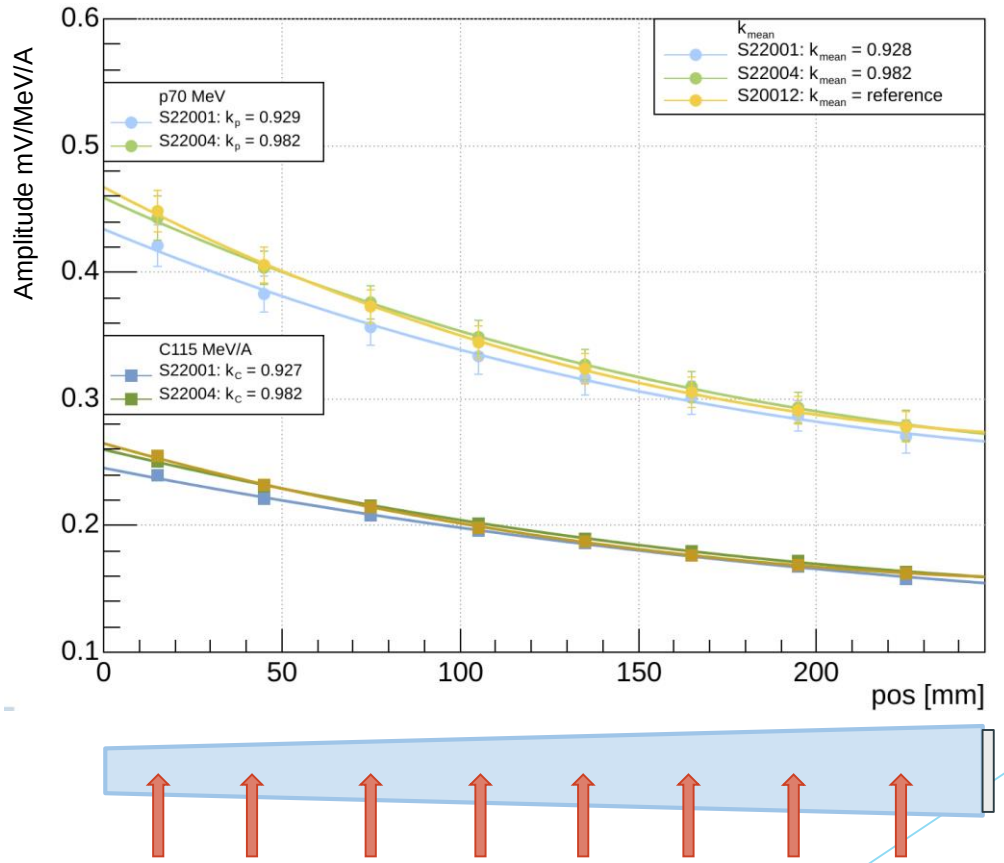
- The correction factor for equalization is presently taken from a calibration file contained in each experimental campaign folder (TACA_Temperature_Calibration_perCry.cal)
- We have the intercalibration factors for three crystals




```
//-----------------------------------------+-----------------------------------
Double_t TACAactNtuHit::GetEqualisationCorrection(Double_t charge_tcorr, Int_t  crysId)
{
  TACAparCal* parcal = (TACAparCal*) fpParCal->Object();


  Double_t Equalis0 = parcal->getCalibrationMap()->GetEqualiseCry(crysId);
  Double_t charge_equalis = charge_tcorr*Equalis0;

  return charge_equalis;

}
```

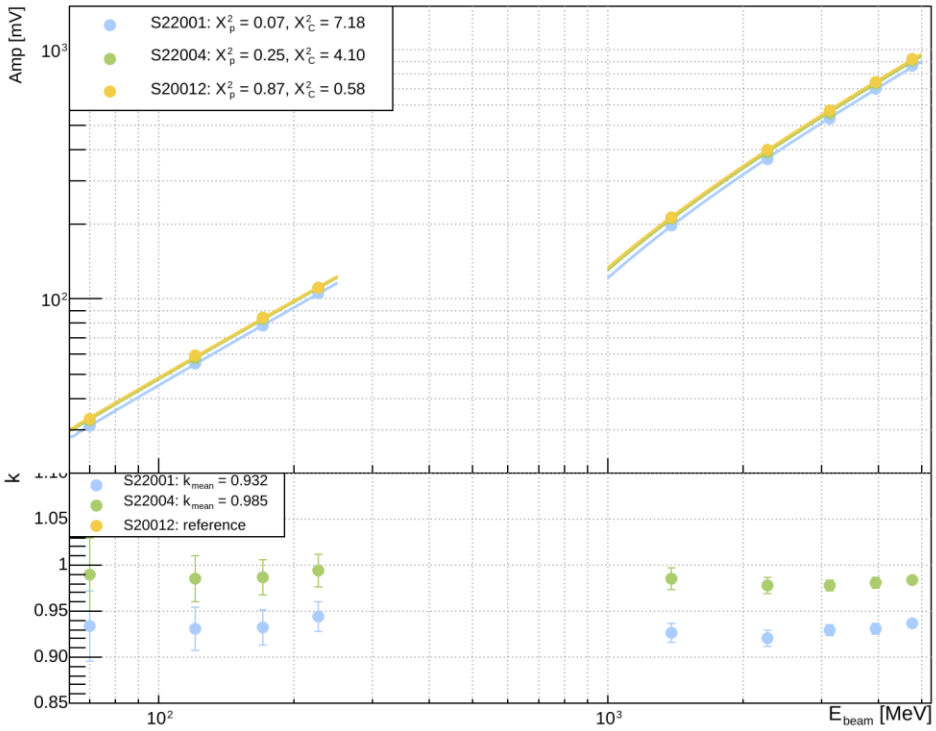
Range correction

- The energy/amplitude depends on the particle position inside the crystal
- The amplitude of the signal decreases when the beam is closer to the SiPM (optical photons absorption)
- Not implemented in Shoe



Conversion from charge to energy

- Linear calibration with p0 = 0 and p1 = 1
- E = p0 + p1*charge
- How to convert from charge to energy if we have not identified the Z of the particle? (Birks effect)
- Cluster reconstruction




```
//-----------------------------------------+-----------------------------------
Double_t TACAactNtuHit::GetEnergy(Double_t rawenergy, Int_t  crysId)
{
   TACAparCal* p_parcal = (TACAparCal*) fpParCal->Object();

   Double_t p0 = p_parcal->GetElossParam(crysId,0);
   Double_t p1 = p_parcal->GetElossParam(crysId,1);

   return p0 + p1 * rawenergy;


  //fake calibration (gtraini)  return raw value meanwhile
 // return rawenergy;
}
```

Conclusions

- Next steps:
  - Temperature reading integrated in the DAQ
  - Range correction?
  - Conversion from charge to energy? Can we obtain information from TOF?