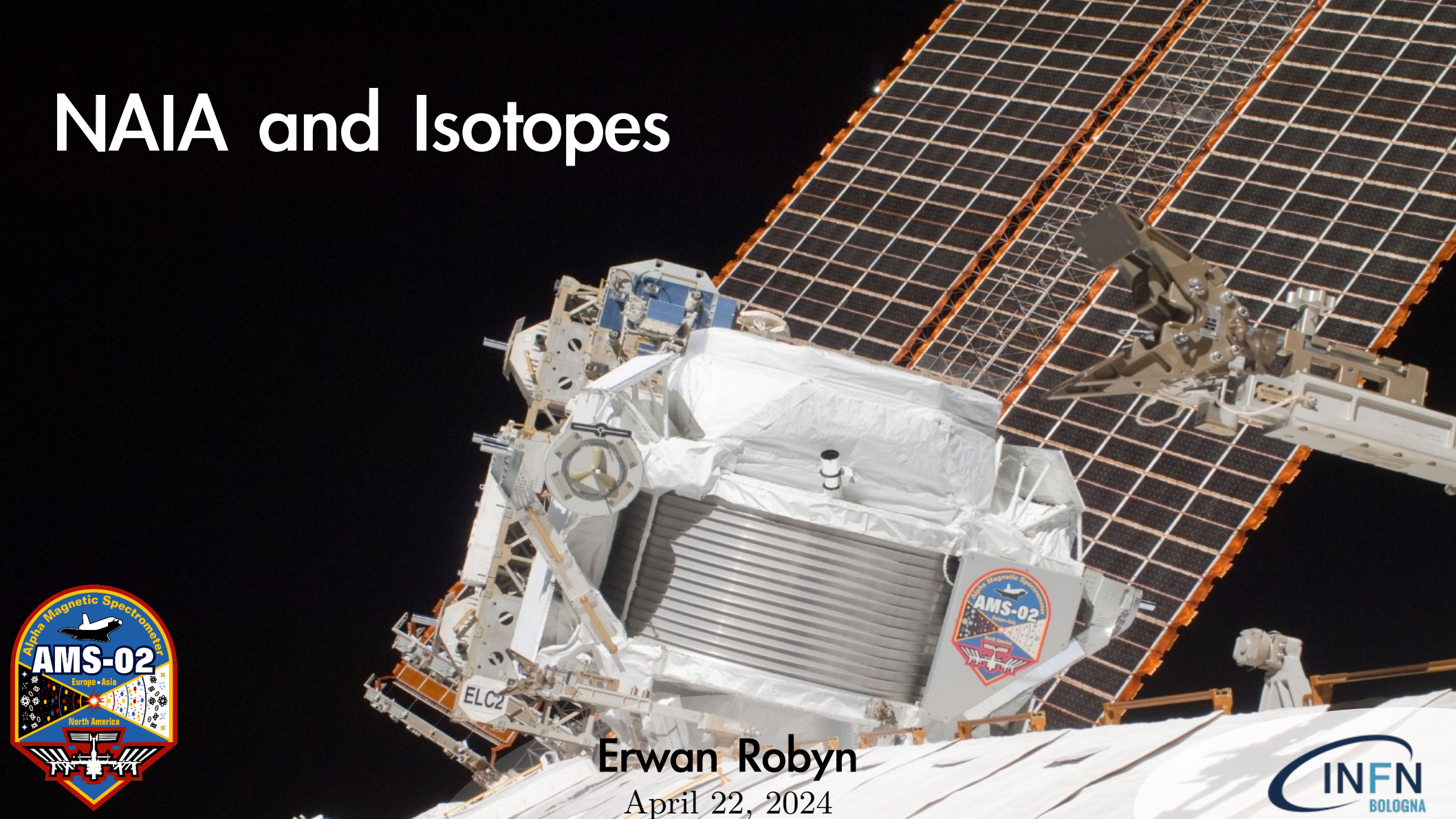
NAIA and Isotopes

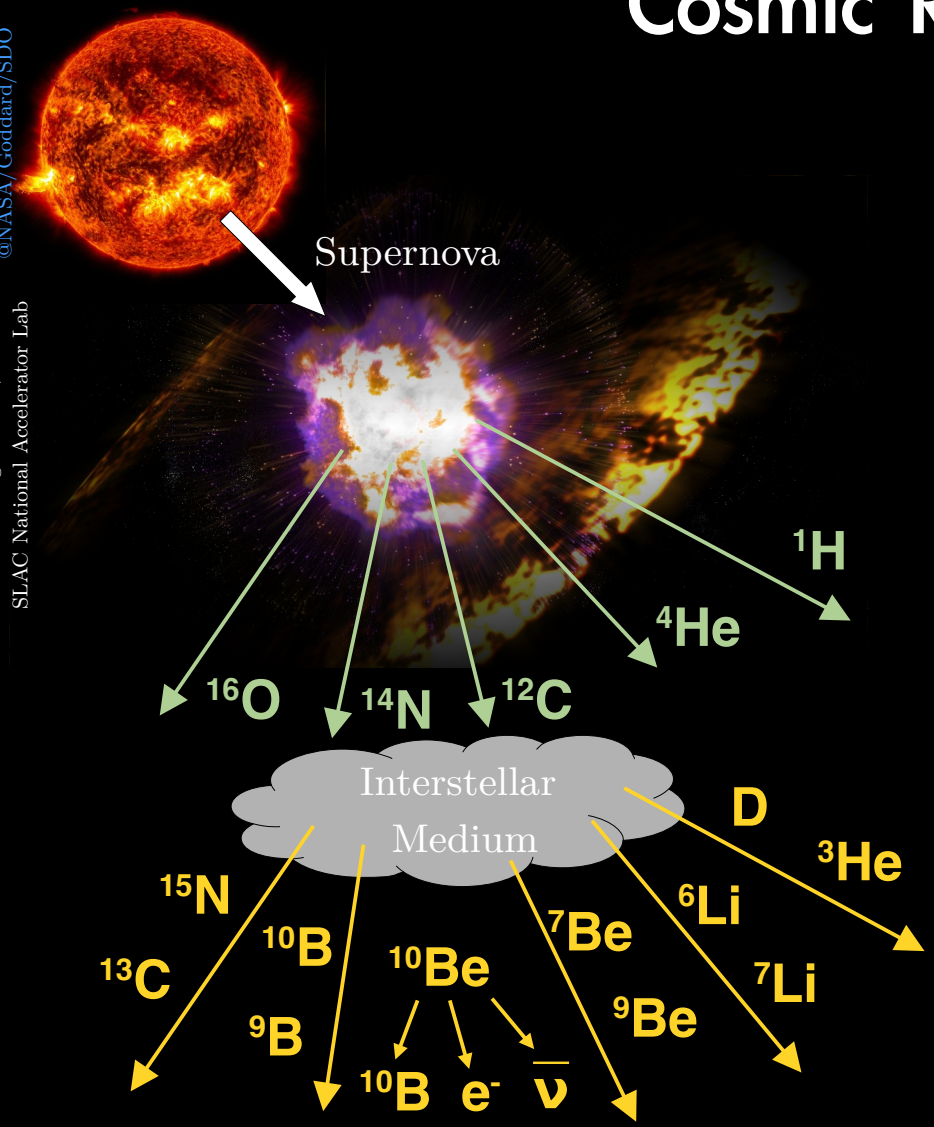


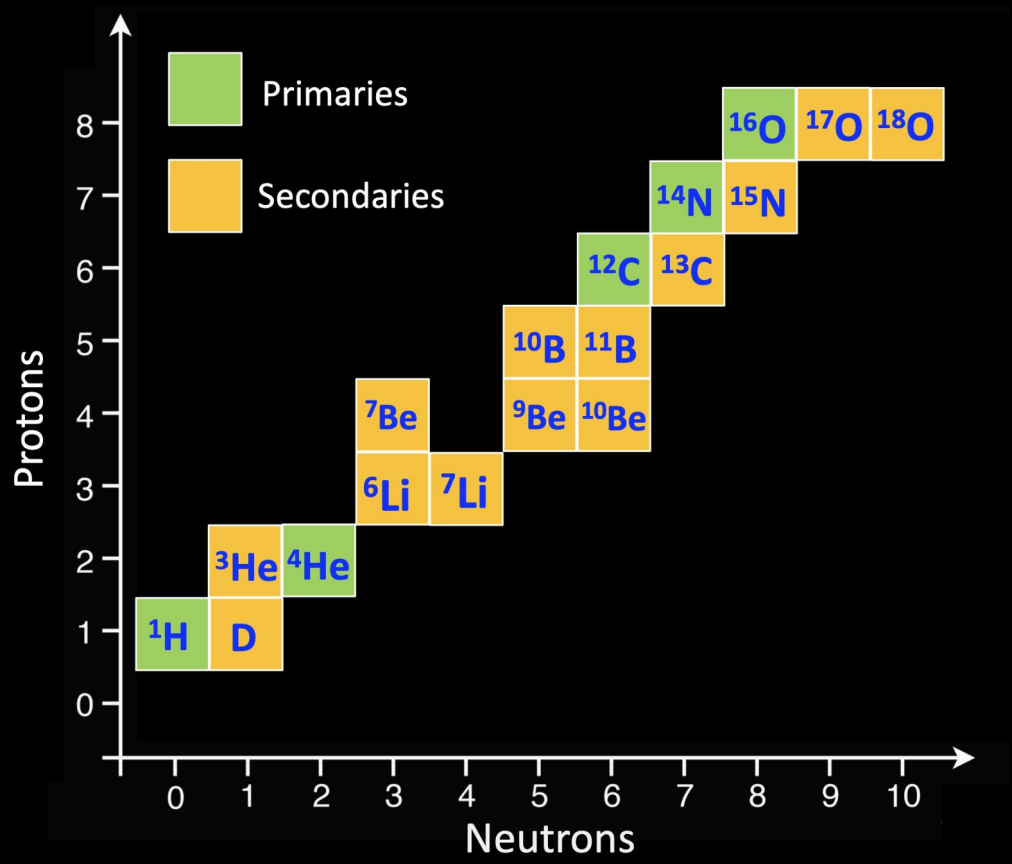
Erwan Robyn

April 22, 2024

INFN BOLOGNA

Cosmic Rays Isotopes



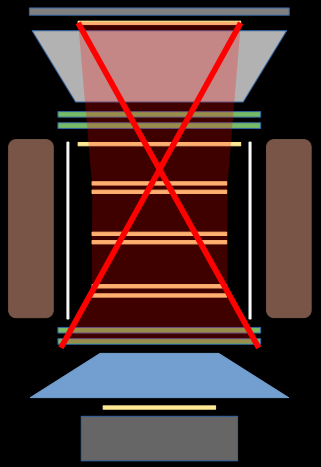


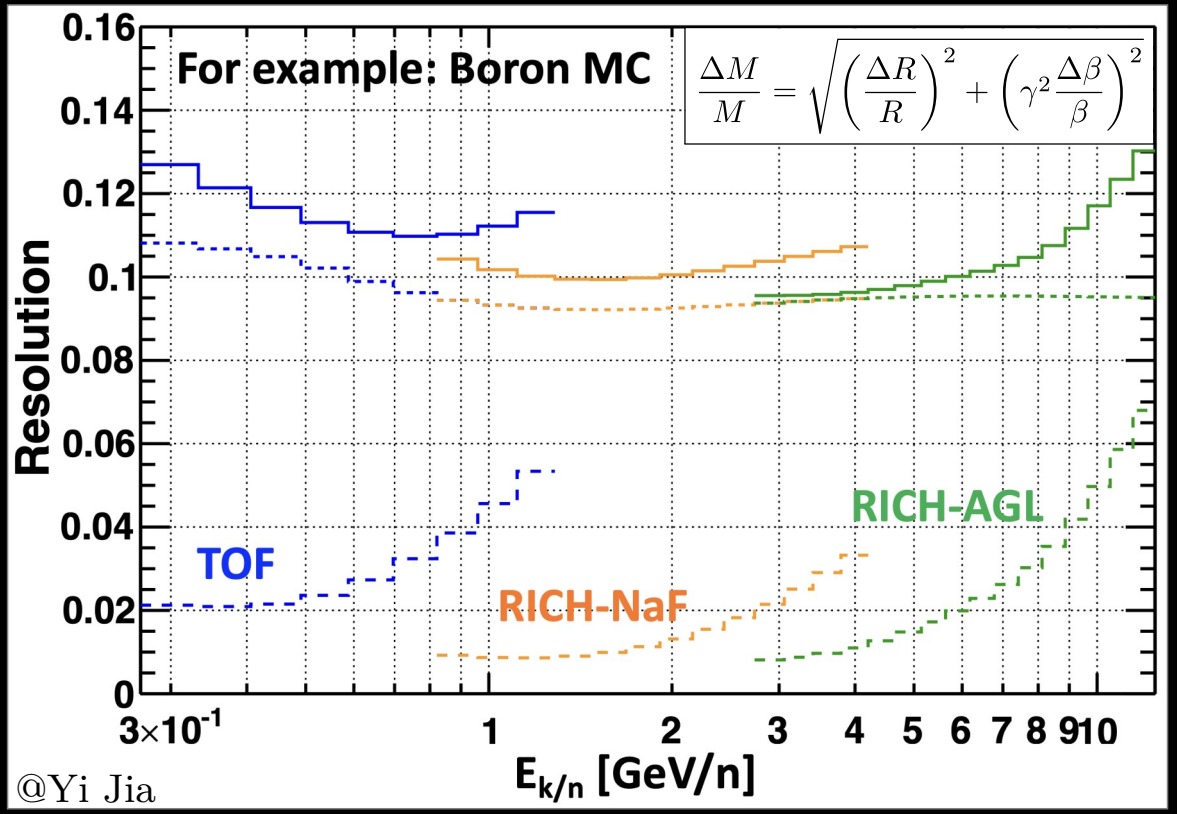
Three Different Geometries

To cover three different beta ranges:

$$M = \frac{ZR}{\beta\gamma}$$

L1Inner + ToF



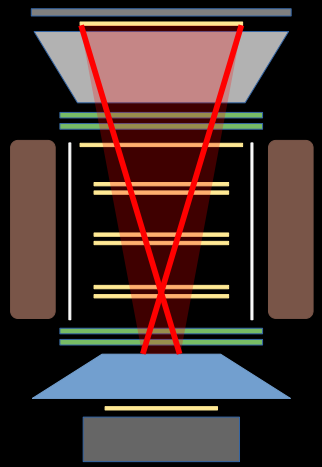


$\frac{\Delta M}{M}$

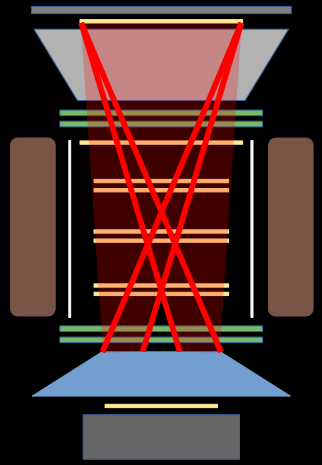
$\frac{\Delta R}{R}$

$\gamma^2 \frac{\Delta\beta}{\beta}$

L1Inner + NaF



L1Inner + Agl



Event Selections

Use NAIA v1.1.0: { ISS: B1236 ; MC: B1306

Standard nuclei selections on Inner Tracker track and InnerL1 (L1, UToF, and Inner) charges.

Use L1 hit for charge measurements but the Inner Rigidity!

InnerL1 + ToF :

- Exclude ToF edge paddles
- Coo Chi2< 5 (data), <10 (MC)
- Time Chi2< 10 (data), < 20 (MC)

InnerL1 + RICH :

- Good & clean
- $P_{Kolmogorov} > 0.01$
- $Z-1 < Q_{RICH} < Z+2$

InnerL1 + RICH NaF :

- NaF geometry
- $N_{PMT} > 10$
- $N_{pe}(ring)/N_{pe}(total) > 0.45$

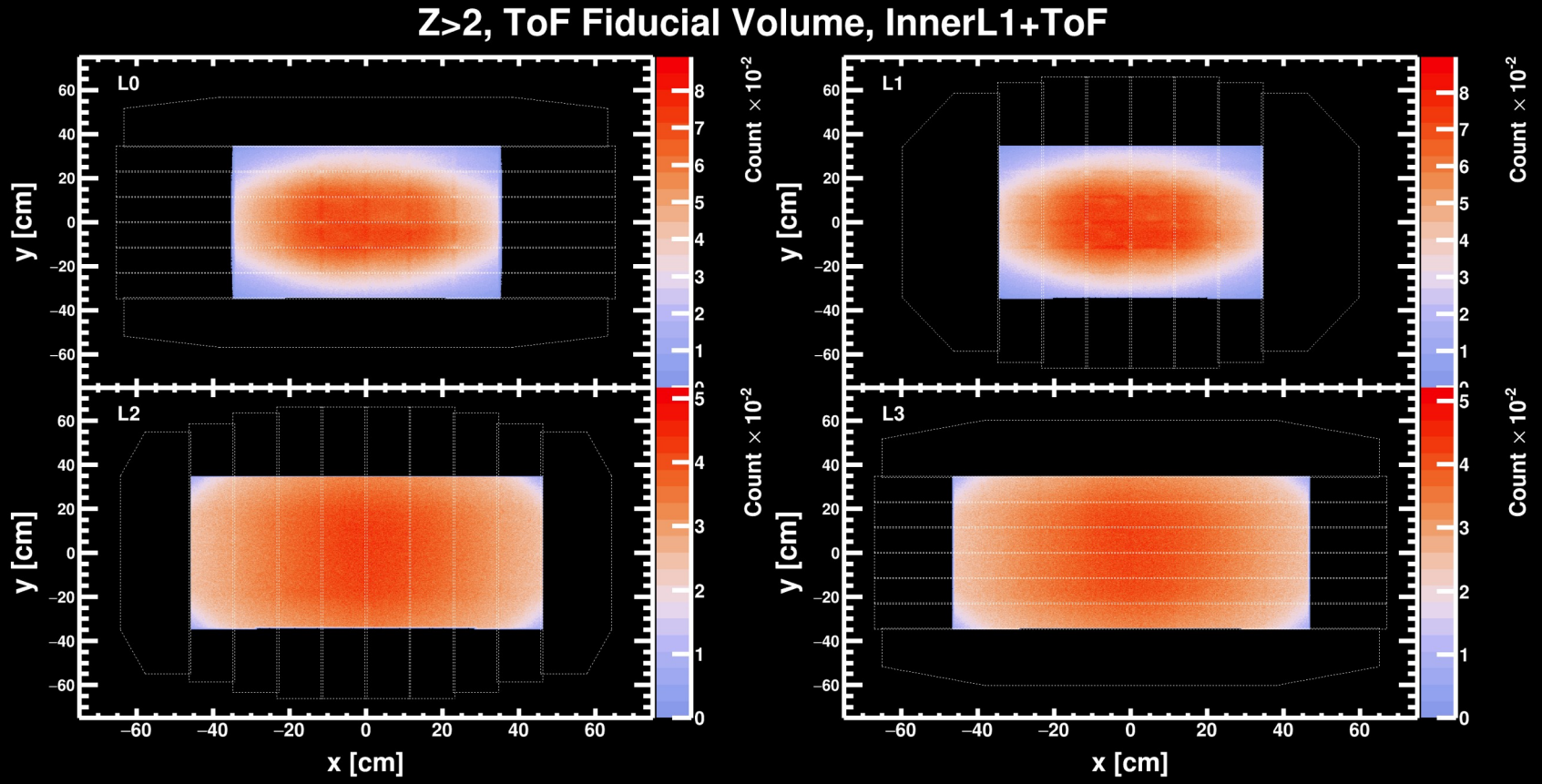
InnerL1 + RICH Agl :

- AGL geometry
- $N_{PMT} > 2$
- $N_{pe}(ring)/N_{pe}(total) > 0.4$
- Good Rich Tiles

ToF Fiducial Volume

ToF fired paddle ID or hits positions are missing in NAIA.

For now use the inter/extrapolation of the tracker track to discard the trapezoidal paddles.



RICH Beta Tuning

A new tuning for the CIEMAT beta is available but not implemented yet in the AMS Software.

Could be stored in NAIA under a new RICH beta type: `Rich::BetaType::TUNED`

The tuning event by event relies on several informations about the event and its ring:

Available in NAIA:

```
float beta;       // RichBaseData::GetBeta()[Rich::BetaType::CIEMAT]
Rad rad;          // RichBaseData::IsNaF()
unsigned int run; // HeaderData::Run

float rich_x;     // RichBaseData::m_beta[0].Pos[0]
float rich_y;     // RichBaseData::m_beta[0].Pos[1]
float rich_theta; // RichBaseData::m_beta[0].Theta
float rich_phi;   // RichBaseData::m_beta[0].Phi


unsigned int charge_z; // Use the Inner Tracker charge as reference
                       // for the tuning
```

Not Available in NAIA:

```
// number of hits on the ring reflected by the mirror
int rich_usedm = ring->getReflectedHits();

// total number of hits on the ring
// (not exactly what is stored in NAIA ring->getUsedHits())
int rich_hit = ring->getHits();

// where RichRingR *ring=pev->pRichRing(irich);
```

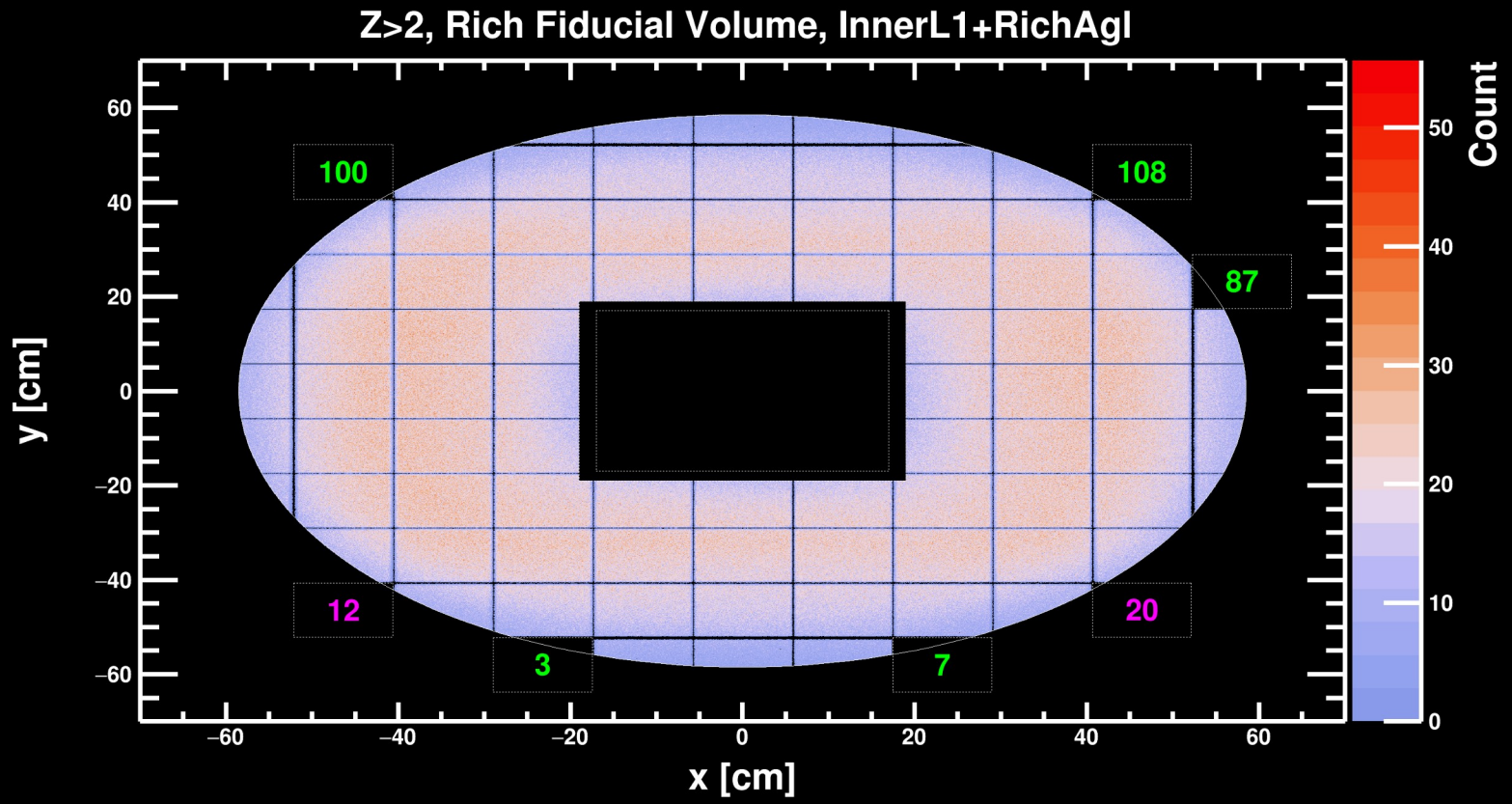
https://gitlab.cern.ch/ams-isotope-analysis/rich-additional-correction

(must subscribe to the ams-isotope e-group to get access to the repository)

Good Impact Point

```
bool NAIA::RichBaseData::RingGeomTest ( int geom = 0 ) const
```

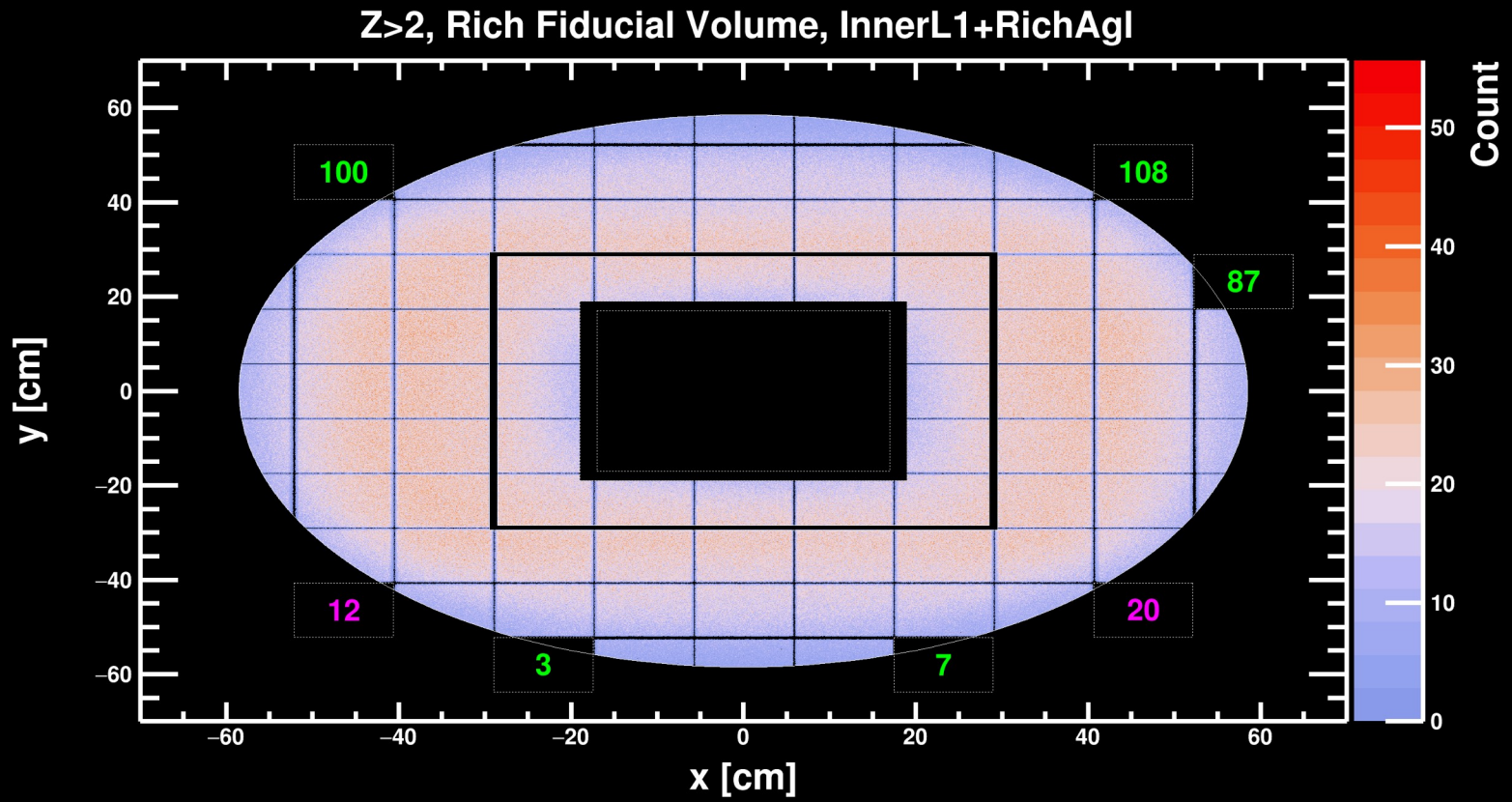
Two **new bad tiles** and the outer aerogel border slightly tightened
(from 59.16 to 58.5 cm)



Good Impact Point

`NAIA::RichBaseData::RingGeomTest( 1 )`

Removes an additional region corresponding to some edges between Aerogel tiles.
This region is also removed in the beta tuning model.

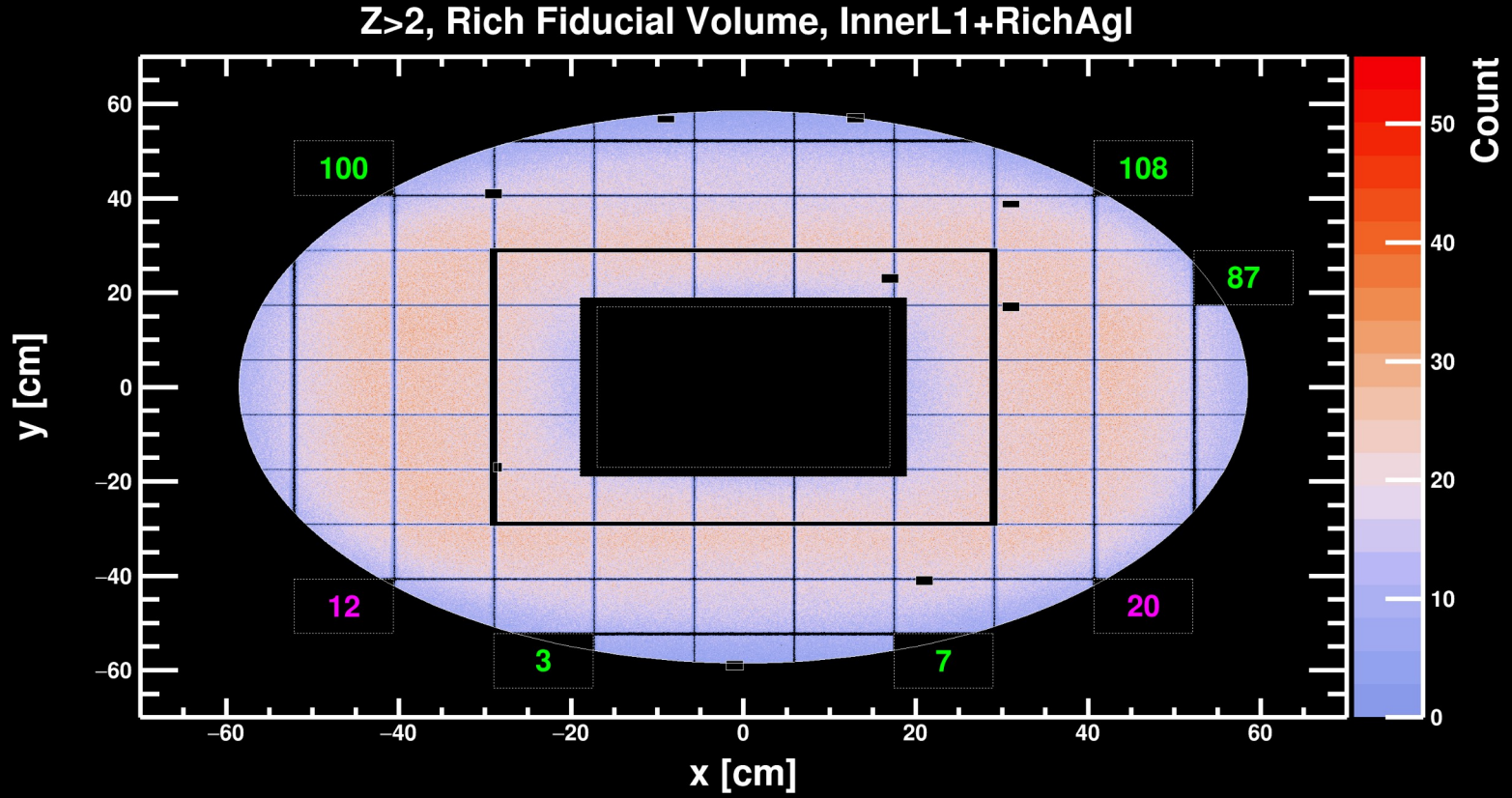


Good Impact Point

`NAIA::RichBaseData::RingGeomTest( 2 )`

Removes nine additional bad aerogel regions.

The exclusion of these regions is suggested on the Twiki.



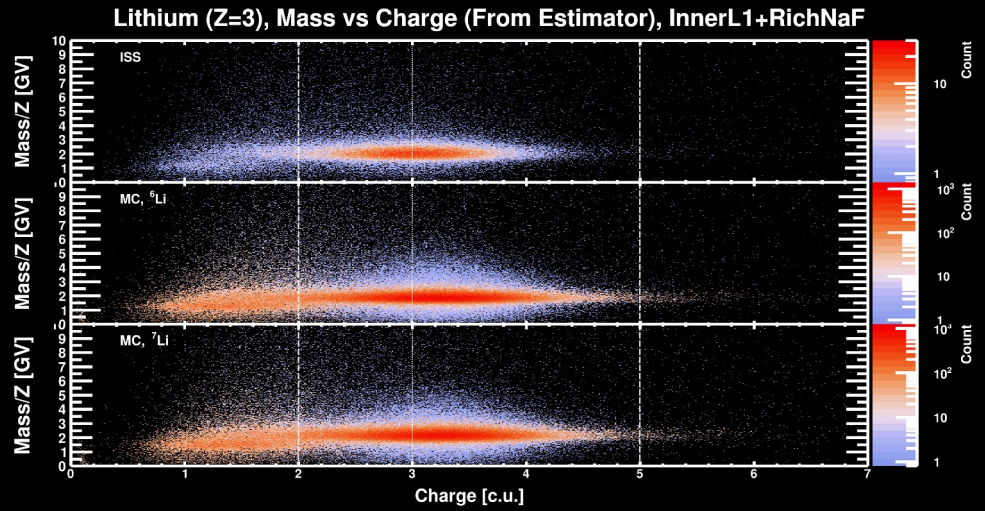
RICH variables in NAIA

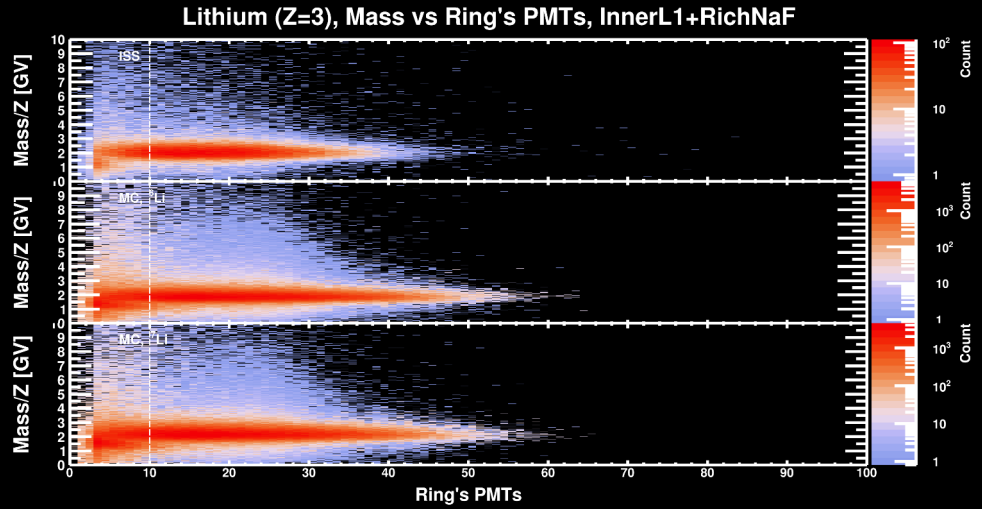
Checks on the different RICH variables stored in NAIA.

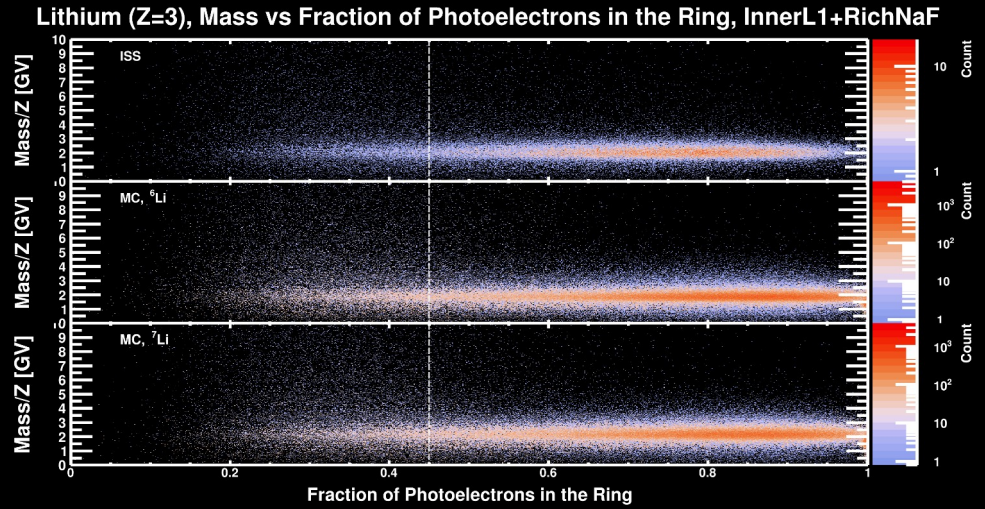
Misleading RICH variables naming/description

#95 · created 1 month ago by Erwan Robyn · 1.2.0

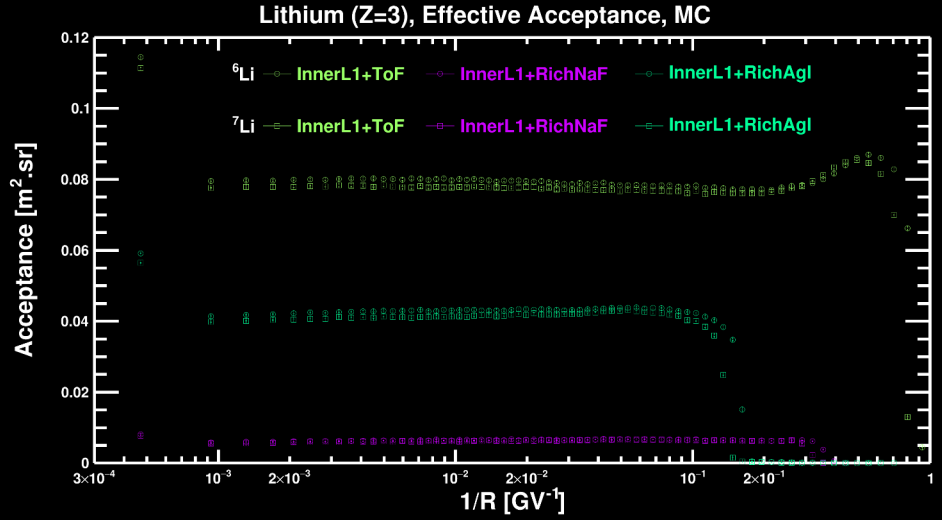
Bug | Data model | Good first issue | To Do

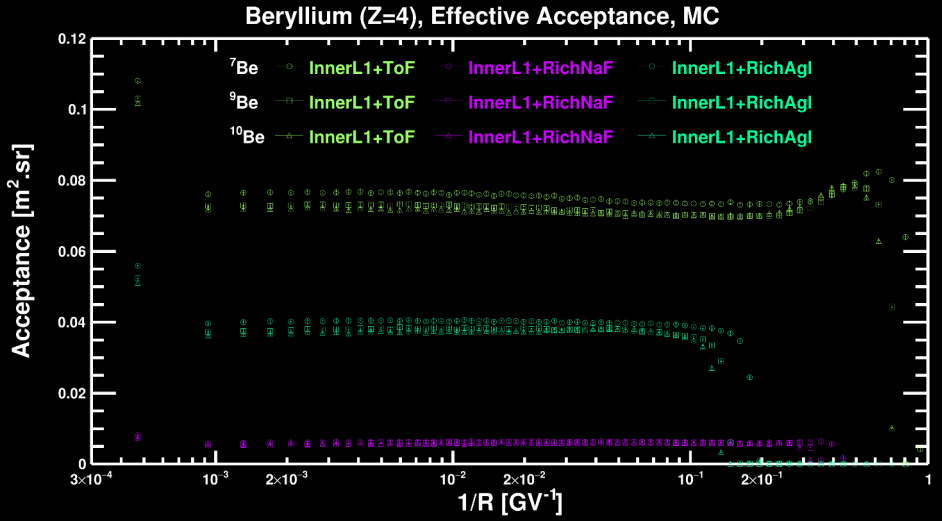


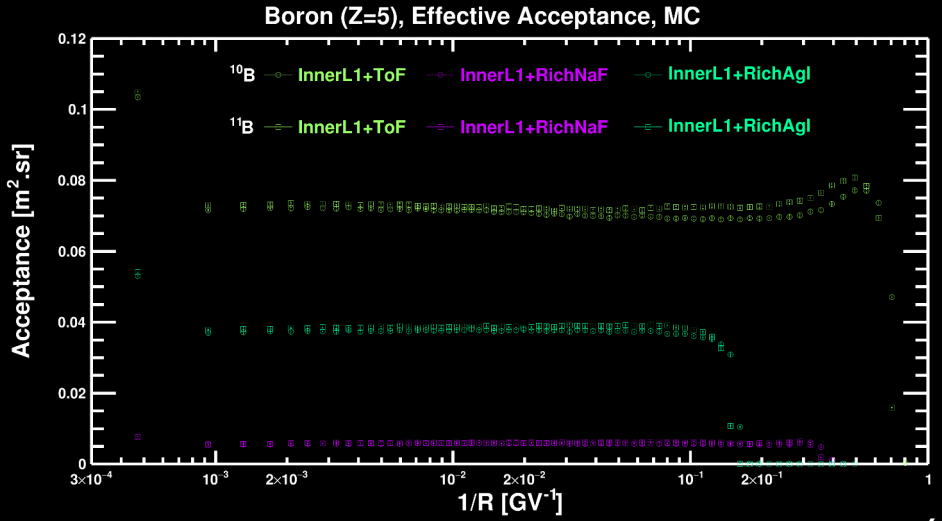




Accceptance and Efficiencies







Reduced NAIA Dataset

I am producing an Heavy Ion Skimmed Dataset of NAIA v1.1.0.

It contains all the event which pass loose Charge Greater Than 2 masks:

```
bool UTOF   = event.CheckMask(NAIA::Category::ChargeGT2_Tof);
bool UTOFSt = event.CheckMask(NAIA::Category::ChargeGT2_Tof_St);
bool TRCK   = event.CheckMask(NAIA::Category::ChargeGT2_Trk);
bool TRCKSt = event.CheckMask(NAIA::Category::ChargeGT2_Trk_St);

bool charge_selection = ( (UTOF) || (UTOFSt) || (TRCK) || (TRCKSt) );
```

5 to 8% of the events survive to this selections.
The use of this reduce dataset for the analysis speed up the selections process by a factor of ~10.

The production is not fully done (nor is NAIA v1.1.0's one) but the objective is to skim all v1.1.0.

The dataset can be found here:
/storage/gpfs_ams/ams/users/erobyn/IonSkimmed/v1.1.0

Future

New NAIA variables:

- ToF fired paddle ID or hits positions.
- RICH number of reflected hits (and maybe more variables about reflected rings)
- New RICH beta obtained with the most recent tuning

Monte-Carlo version B1308 available:

It includes paddle-by-paddle correction for MC. This improve the agreement between data and MC at high energy.

NAIA:

- Should I report in some way the corrupted NAIA files ?
- NAIA v1.1.0_bugfix ? NAIA v1.2.0 ?

> ⚠ Warning
>
> In order for this to work in NAIA we overload the `->` operator to hide this "read-on-demand" behavior. It is required that you always use `->` to access the data members and methods of a container.

Would it be possible to raise an error if → is not used ? Or prevent the usage of the .

Back Up

Three different selections

Common :

- Tracker InnerL1 fiducial volume
- Physical trigger
- beta > 0.4, NToF Hit >= 3
- No 2nd Track
  - 2nd Track Rigidity < 0.5GV or ntrack==1 or (Inner X Hit < 3 && Inner Y Hit < 5)

- Track:
  - InnerNHitY>=5 && L2&(L3|L4)&(L5|L6)&(L7|L8)
  - L1XY Hit
  - InnerNormChisY<10

- Charge selections:
  - Z-0.45 < Q_{Inner} < Z+0.45
  - $Q_{Inner,\ RMS}$<0.55
  - Z-0.6 < Q_{UToF} < Z+1.5
  - Z-0.46-0.16(Z-3) < Q_{L1} < Z+0.65
  - Good L1 charge status
  - Q_{LToF} > Z-0.6

Use L1 hit for charge but the Inner Rigidity

Common :

- Good & clean
- $P_{Kolmogorov}$ > 0.01

- Z-1< Q_{RICH} < Z+2

InnerL1 + ToF :

- Exclude ToF edge paddles
- Coo Chi2< 5 (data), <10 (MC)
- Time Chi2< 10 (data), < 20 (MC)

InnerL1 + RICH NaF :

- NaF geometry
- N_{PMT} > 10
- $N_{pe}(ring)/N_{pe}(total)$ > 0.45

InnerL1 + RICH Agl :

- AGL geometry
- N_{PMT} > 2
- $N_{pe}(ring)/N_{pe}(total)$ > 0.4
- Good Rich Tiles