TELLIE PCA Automation

Report

May 10, 2021

Michal Rigan
mrigan@snolab.ca

University of Sussex

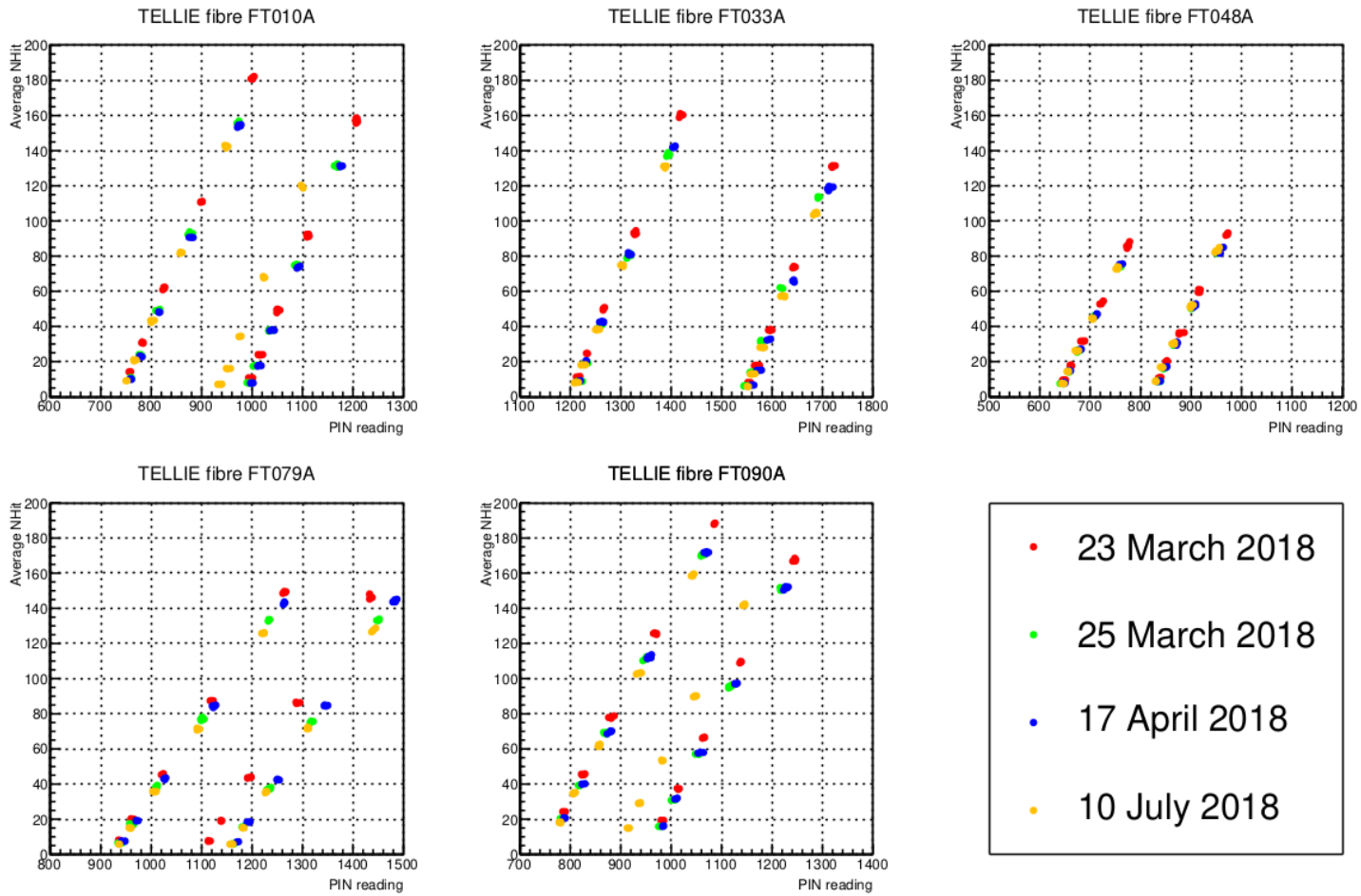
Automation - Why

- TELLIE PCA takes a lot of time - both manpower and detector time ($<$16h data taking, 2 days processing)
- the frequency can be tuned so that we are satisfied with the deadtime
- was the plan since TELLIE was designed
- other detector aspects are being automated (shifting)
- (automation, if done properly, is good)

Monitoring

- there won't be a person looking at the data-taking in real time (as was the case for dedicated runs until now)
- need a method to monitor the light intensity fired by TELLIE
- don't have immediate access to live NHit data in Orca (this is done by Event builder, and then later available on data stream)
- have access to PIN readouts - internal TELLIE reading, corresponds to light intensity, already available to Orca
- need to know PIN to NHit correlation and stability

PIN vs NHit



reasonably stable over time, only likely to change with changes to detector or TELLIE hw

Automation - How

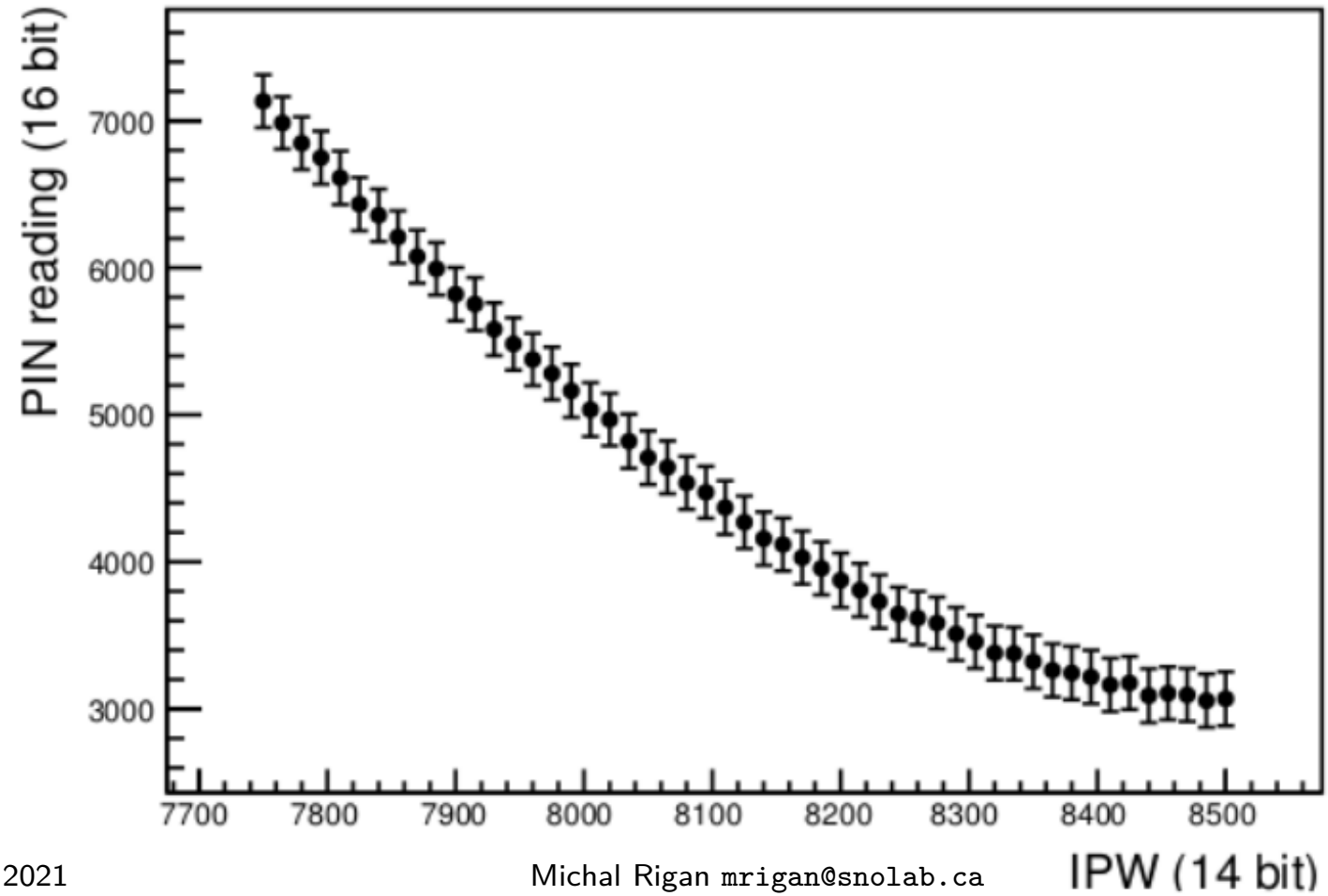
- data-taking auto
- processing + analysis auto

Automation: data-taking

- Profiling (less frequently):
  - need calibration curves (IPW - PIN)
  - need tuning curves (PIN - NHit)
- need to decide how we want to take the data:
  - automate current dedicated running
  - continuous running alongside Physics

Data-taking: calibration curves

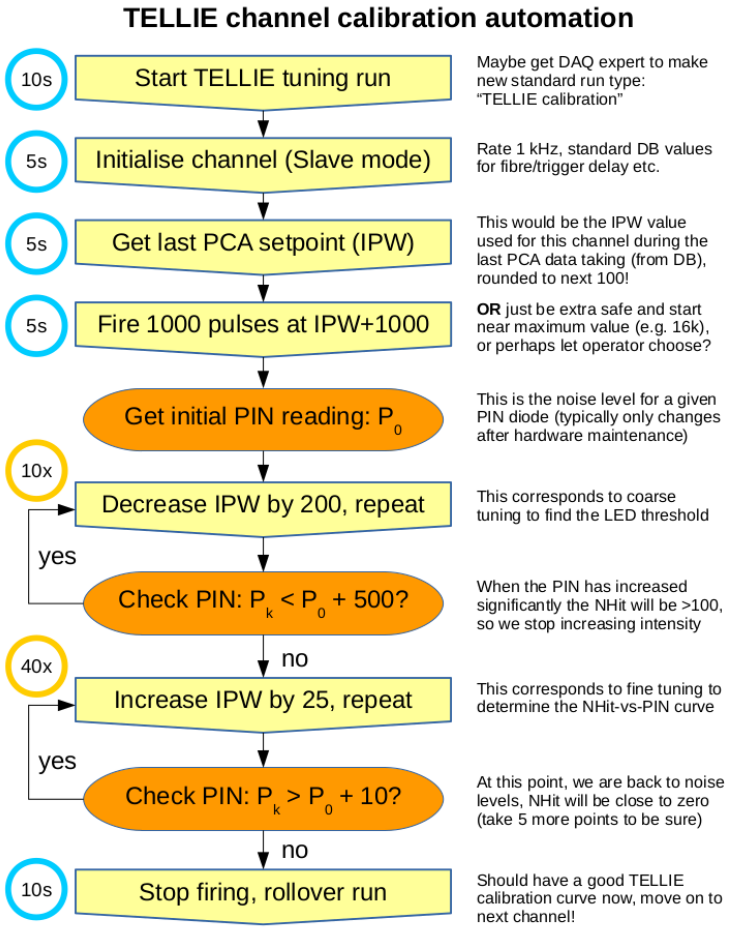
Calibration curves scan the fibre to obtain the PIN to IPW data (we set the IPW, PIN is our internal intensity check)



Data-taking: calibration curves

- PIN reading is inversely proportional to the IPW setpoint (this determines the pulse intensity)
- up until emission threshold where the PIN becomes flat
- LED intensity tends to drift over time with the same IPW, so the PIN readings should be trusted rather than IPW

Data-taking: calibration curves



Potential issues / suggestions:

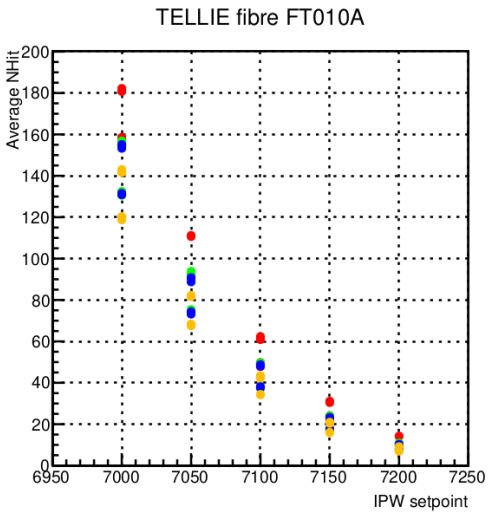
- Would be better to check NHit instead of PIN, if possible
- Need safety interlock: stop firing immediately if NHit > 1000
- Is the time estimate (5 min/channel) realistic?
- Are there any other issues one might encounter?

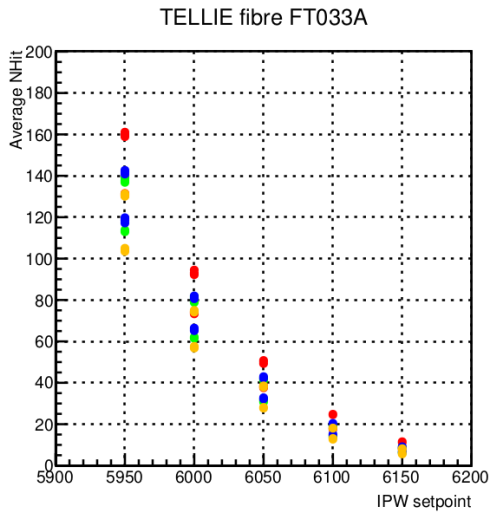
Data-taking: tuning curves

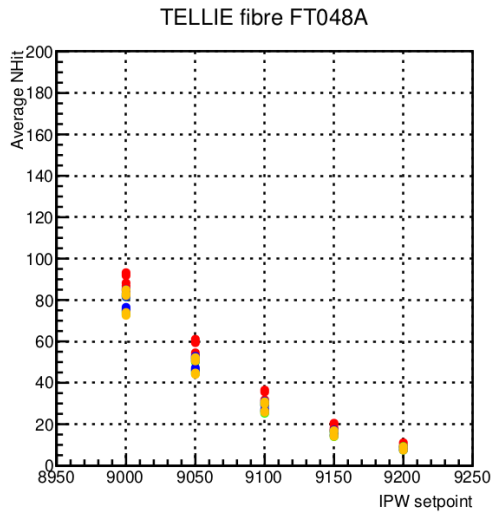
The average EXTA-NHit can be extracted from data by nearline processor, creating PIN-NHit plot. This can be fitted and several parameters can be extracted:

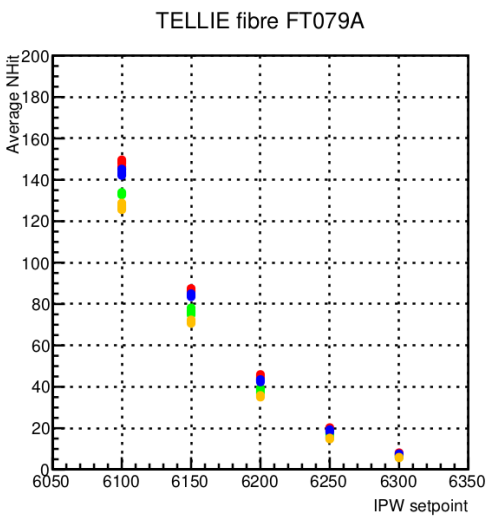
- LED emission threshold (lowest IPW producing 0 NHit)
- PCA intensity setpoint (IPW corresponding to 45 NHit)
- Internal noise level (PIN value below the LED emission threshold)
- Fit parameters to match external NHit to PIN reading ($y = ax + b$)

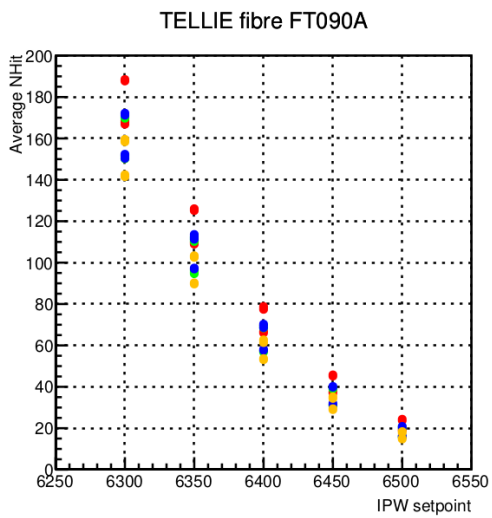
Data-taking: tuning curves

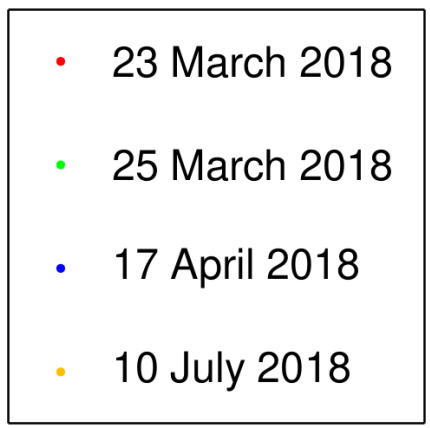












Data-taking: dedicated

If we decide to follow the dedicated way = 1 kHz, 200 000 pulses:

Take PCA dataset (dedicated TELLIE run)

1. Start "TELLIE" run, initialise a channel
2. Select IPW setting corresponding to PCA intensity setpoint (from database)
3. Fire 1000 pulses at 1 kHz (tuning run), do not roll over run
4. Use fit parameters from database to extrapolate NHit from PIN (y = ax + b)
   - If EXTA-NHit is too high (>45), increase IPW by 10. Repeat from step 3.
   - If EXTA-NHit is too low (<40), decrease IPW by 10. Repeat from step 3.
5. When the extrapolated NHit is in the desired range (40-45), save IPW setpoint.
6. Restart into "TELLIE_PCA" run, initialise same channel.
7. Fire 200'000 pulses at 1 kHz (PCA run) with the obtained IPW setpoint.
8. Restart into "TELLIE" run and repeat for next channel/fibre from step 1.

Data-taking: dedicated

- this is what is currently done manually by operators
- entire dataset = 95 fibres takes 12-16h to complete
- automation would greatly facilitate this

Data-taking: continuous

If we decide to follow the continuous method:

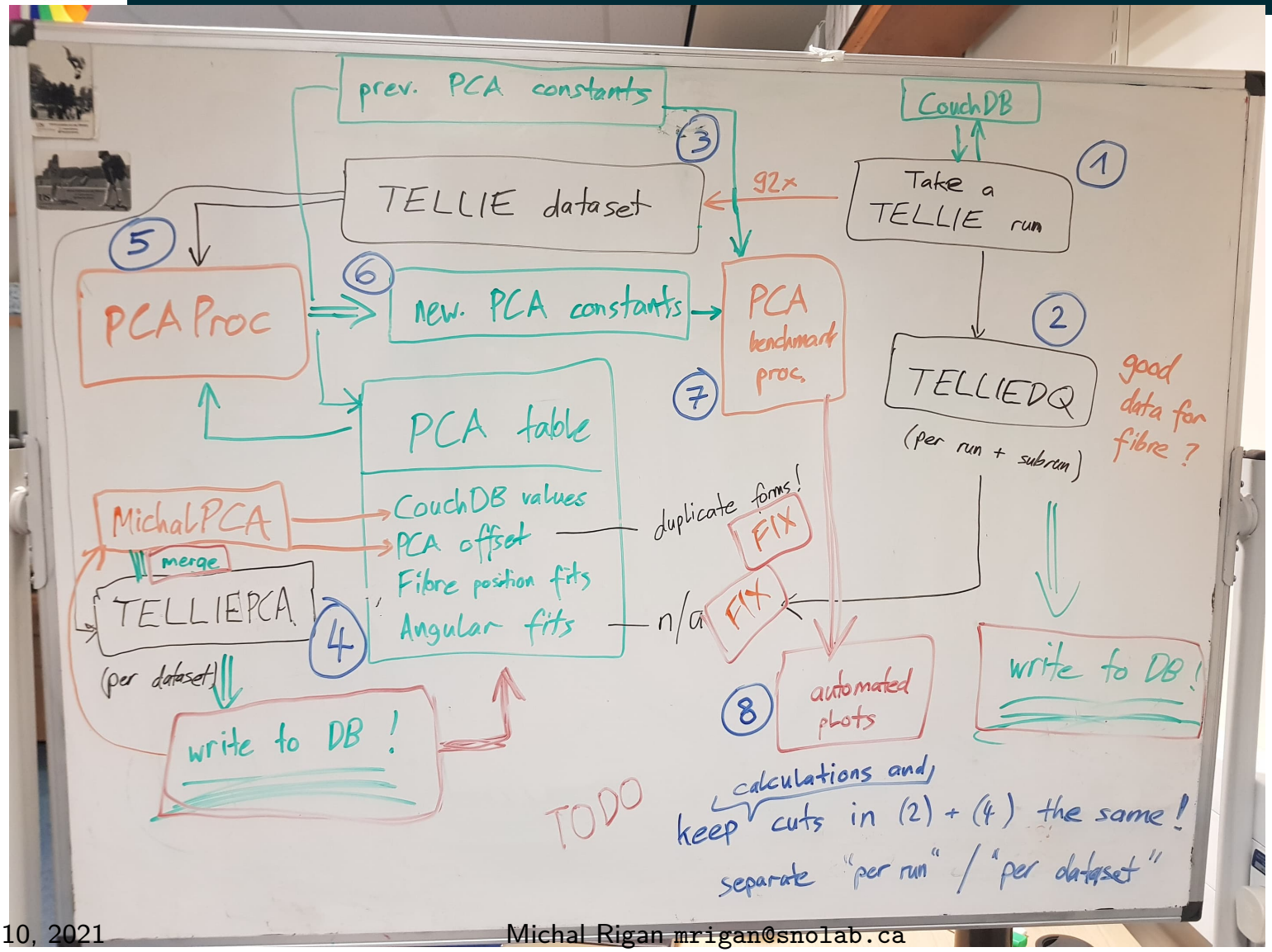
Take PCA dataset (continuously during physics run)

1. Follow similar procedure to above, but fire at a reduced rate of 50 Hz while ensuring zero overlap with PulseGT (alternate between PGT+EXTA triggers, i.e. the combined rate would be 100 Hz).
2. Continuously update IPW setpoint (e.g. every 1000 pulses / 20s) to ensure the desired PCA intensity setpoint (NHit ~40, PIN ~1150 in Fig. 1) are maintained despite potential intensity shifts.
3. Only switch to the next fibre if the previous run rolled over by itself.

Data-taking: continuous

- normal physics run would contain 180 000 TELLIE pulses
- PulseGT won't steal EXTA triggers (other triggers still can, but the precise timing can be used to flag data)
- full PCA set would require 95 completed physics runs ($\sim$ 4 days)
- Processing follows when full set is taken
- intervention only required when things go wrong
- this should increase the detector livetime and reduce need for TELLIE operators
- obviously requires a lot of work, and most importantly ORCA knowledge

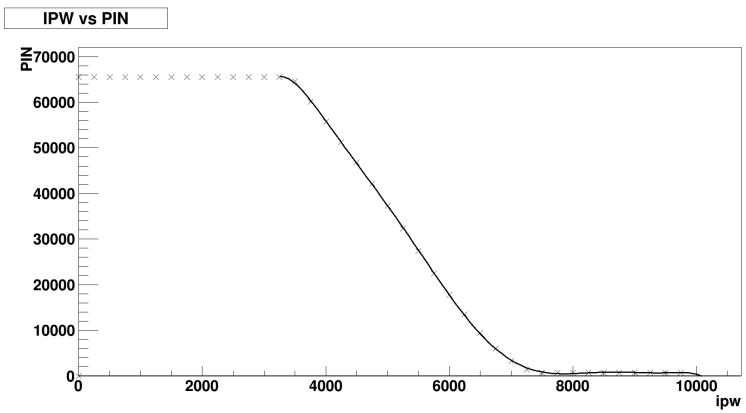
Automation: processing



Automation: processing

- complex
- all the required bits are available
- need to unify settings and cuts
- needs database interface
- need monitoring

Automation: what was done




```
{
    "type": "TUNING",
    "index": "",
    "timestamp": "2019-08-028T04:45:52.40Z",
    "version": 0,
    "run_range": [1, 300000],
    "pass": 0,
    "channel": 2,
    "ipw": 6350
}
```

- a start was made using teststand and fake tellie server
- the calibration curves script written
- polynomial fit to PIN response
- data upload to database
- really just a baby step...