Part 2: Developing Catherine's temperature dependent CTI-correction algorithm as contributed software

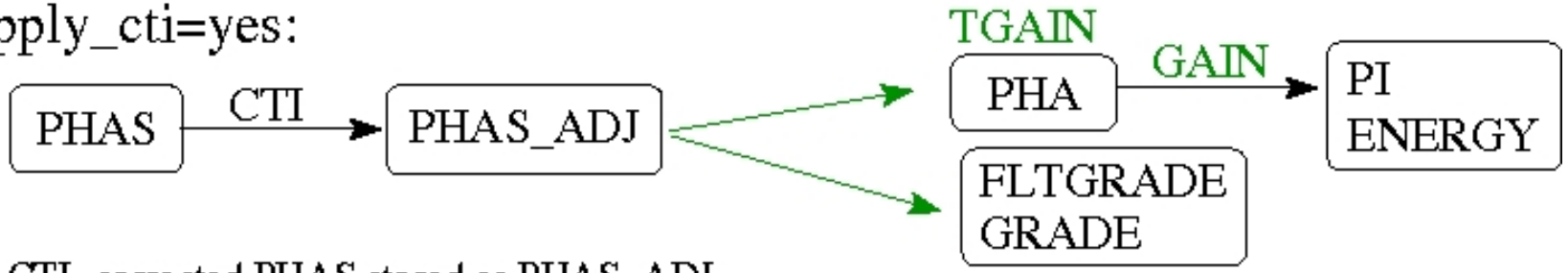
Jennifer Posson-Brown (SAO), Catherine Grant (MIT)
Glenn Allen (MIT), Paul Plucinsky (SAO)
& ACIS Team

Overview

- Our strategy for developing IDL code to work with acis_process_events
- Description of IDL code, testing, and how to use it
- Example with ECS data
- Future work

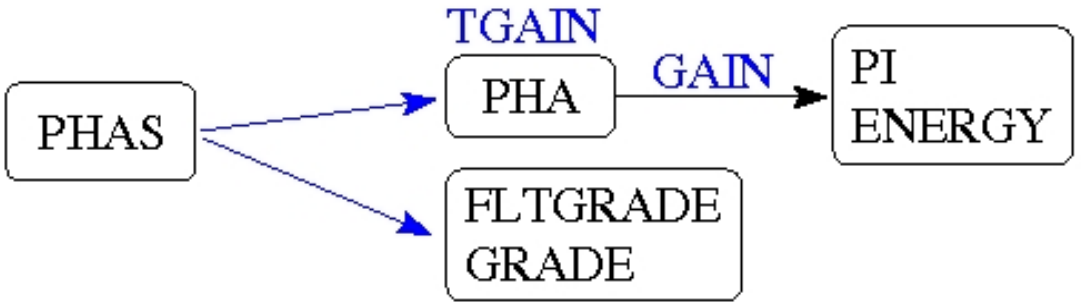
What acis_process_events does:

if apply_cti=yes:



* CTI–corrected PHAS stored as PHAS_ADJ
* original values retained in PHAS
* PHA, GRADE, FLTGRADE calculated from PHAS_ADJ
* TGAIN applied to PHA
* PI, ENERGY calculated from PHA and GAIN file

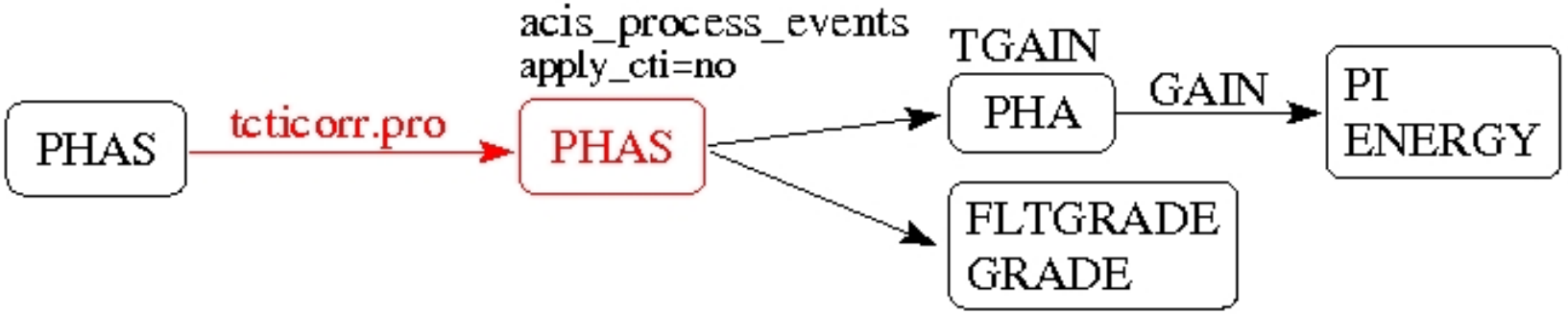
if apply_cti=no:



* PHA, GRADE, FLTGRADE calculated from PHAS
* TGAIN applied to PHA
* PI, ENERGY calculated from PHA and GAIN file

Note! TGAIN and GAIN files, and rules for calculating PHA and FLTGRADE, are different for CTI–corrected and uncorrected data.

Basic strategy:



1. Use IDL code to apply temperature–dependent CTI correction, updating PHAS column
2. Use acis_process_events with apply_cti=no to calculate PHA, FLTGRADE, and GRADE based on updated PHAS, apply TGAIN and GAIN

Problem: Rules for caculating PHA and FLTGRADE are different for CTI–corrected and uncorrected data.

Solution: IDL code rearranges PHAS in certain cases to ensure a consistent grading scheme with CTI–corrected data.

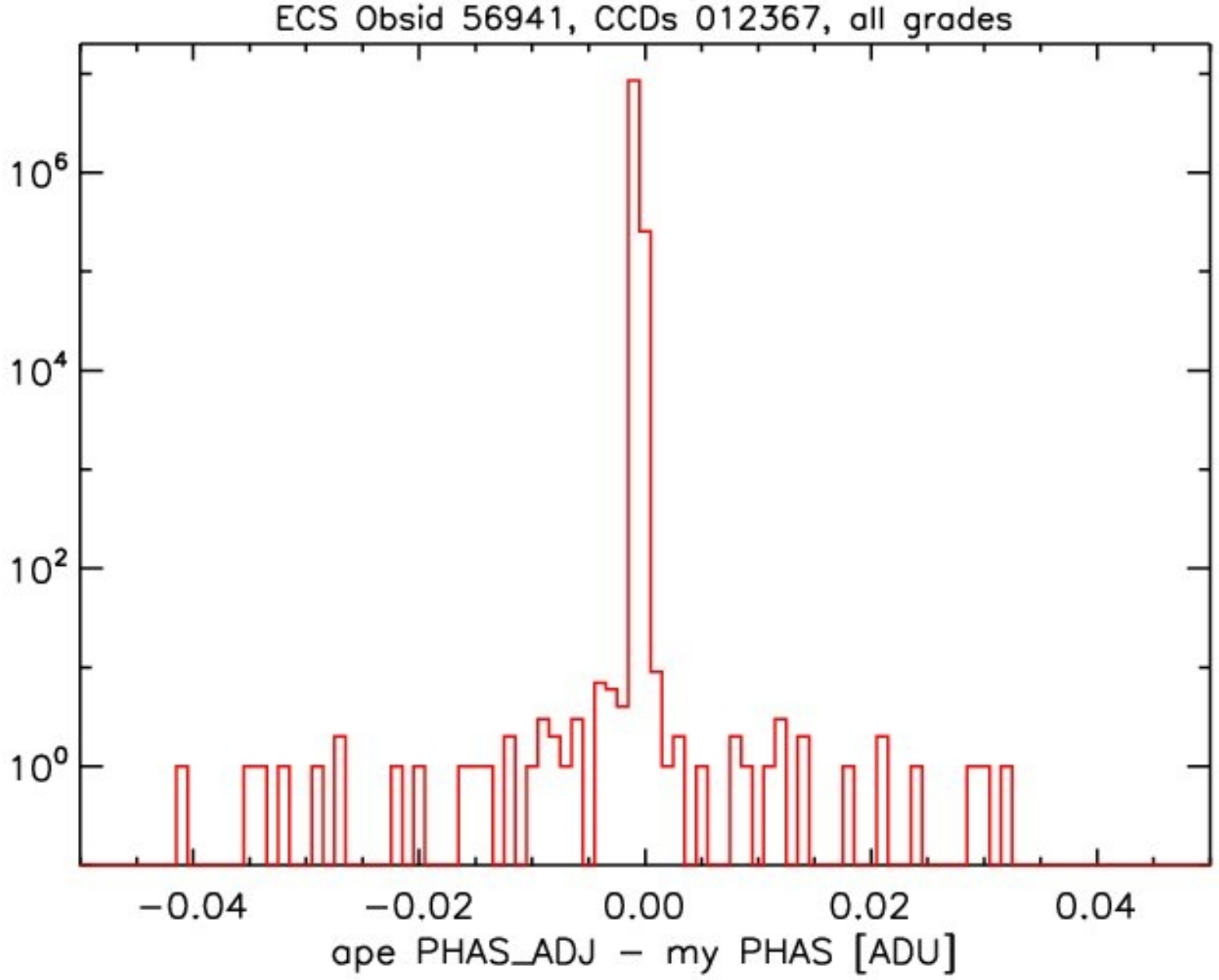
Problem: Different TGAIN and GAIN files are used for CTI–corrected and uncorrected data.

Solution: Specify appropriate files with acis_process_events parameters gainfile and tgainfile.

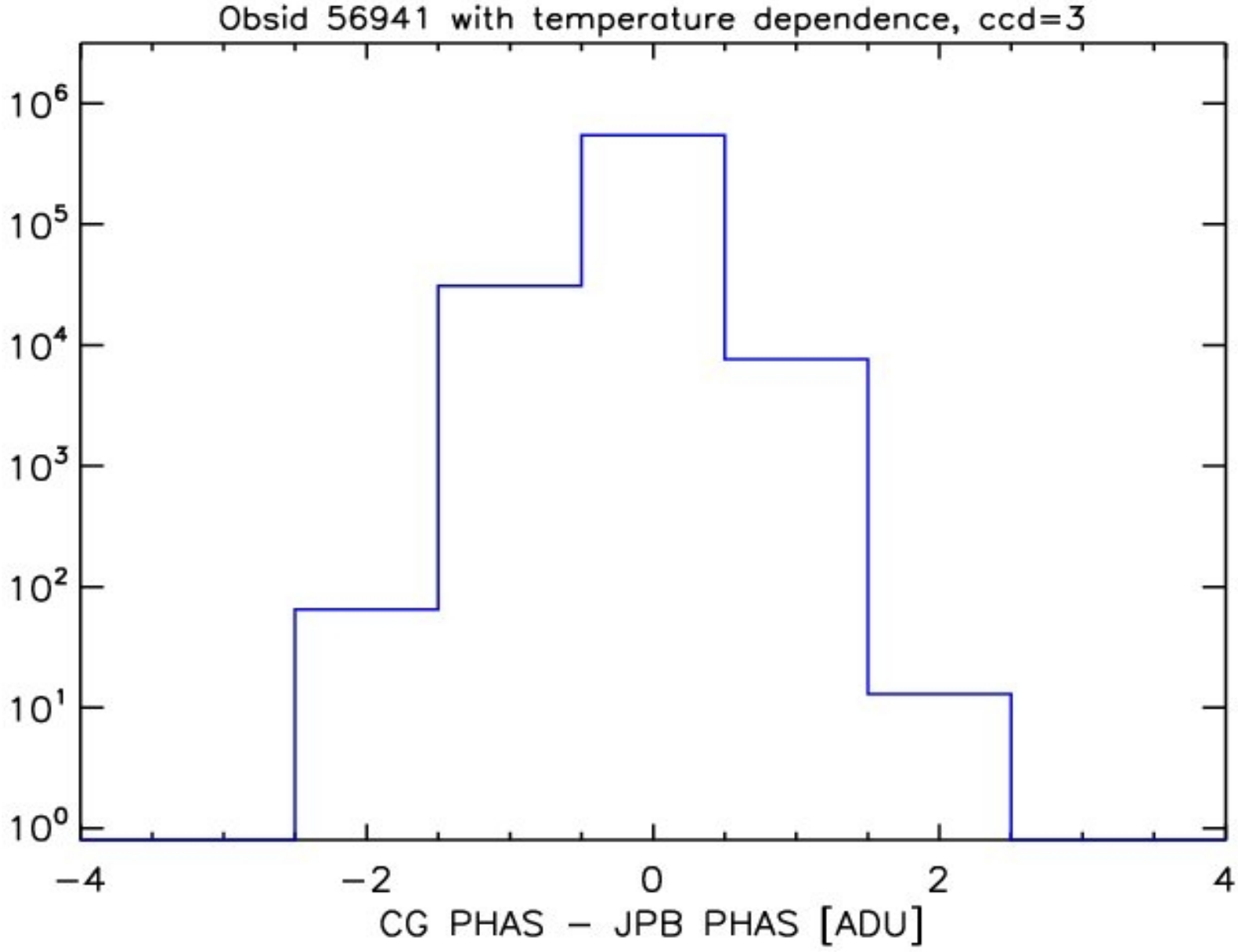
Description of tcticorr.pro

- Input files: level 1 event list, MTL file
- Parameters: % CTI change per degree values from Catherine's analysis. (1 per FI chip, 2 – parallel and serial – per BI chip)
- Gets focal plane temperature from MTL file for each event by matching exposure number.
- Applies CTI correction like acis_process_events (using CALDB CTI file) but with temperature-dependent correction factor applied to trapmap
- Rearranges PHAS in certain cases to ensure grading scheme consistent with CTI-corrected data.
- Output: level 1 event list with modified PHAS column

Test 1: tcticorr.pro should match acis_process_events when % CTI change per degree = 0



Test 2: tcticorr.pro should match Catherine's code when % CTI change per degree = values from her analysis:



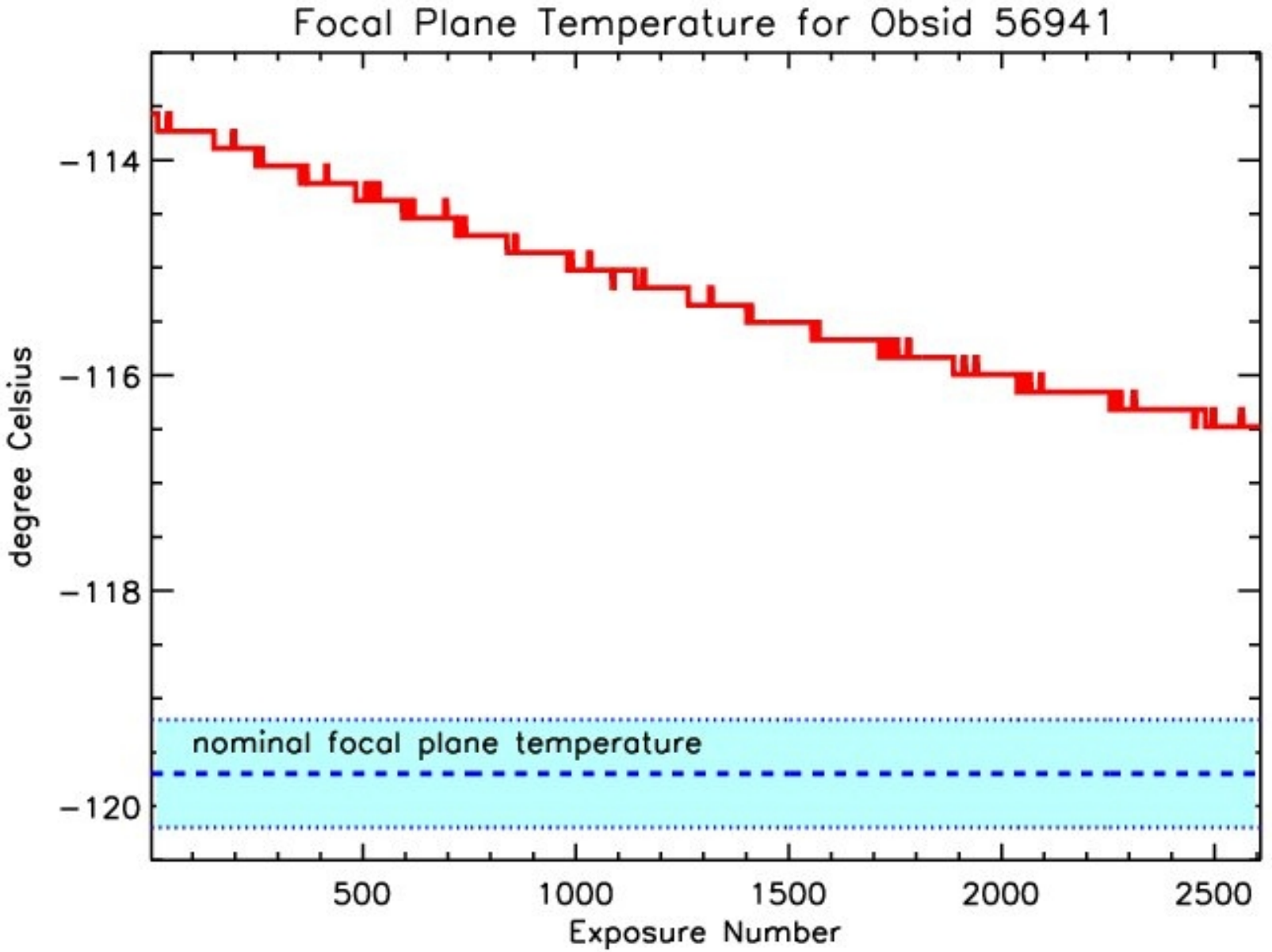
How to use

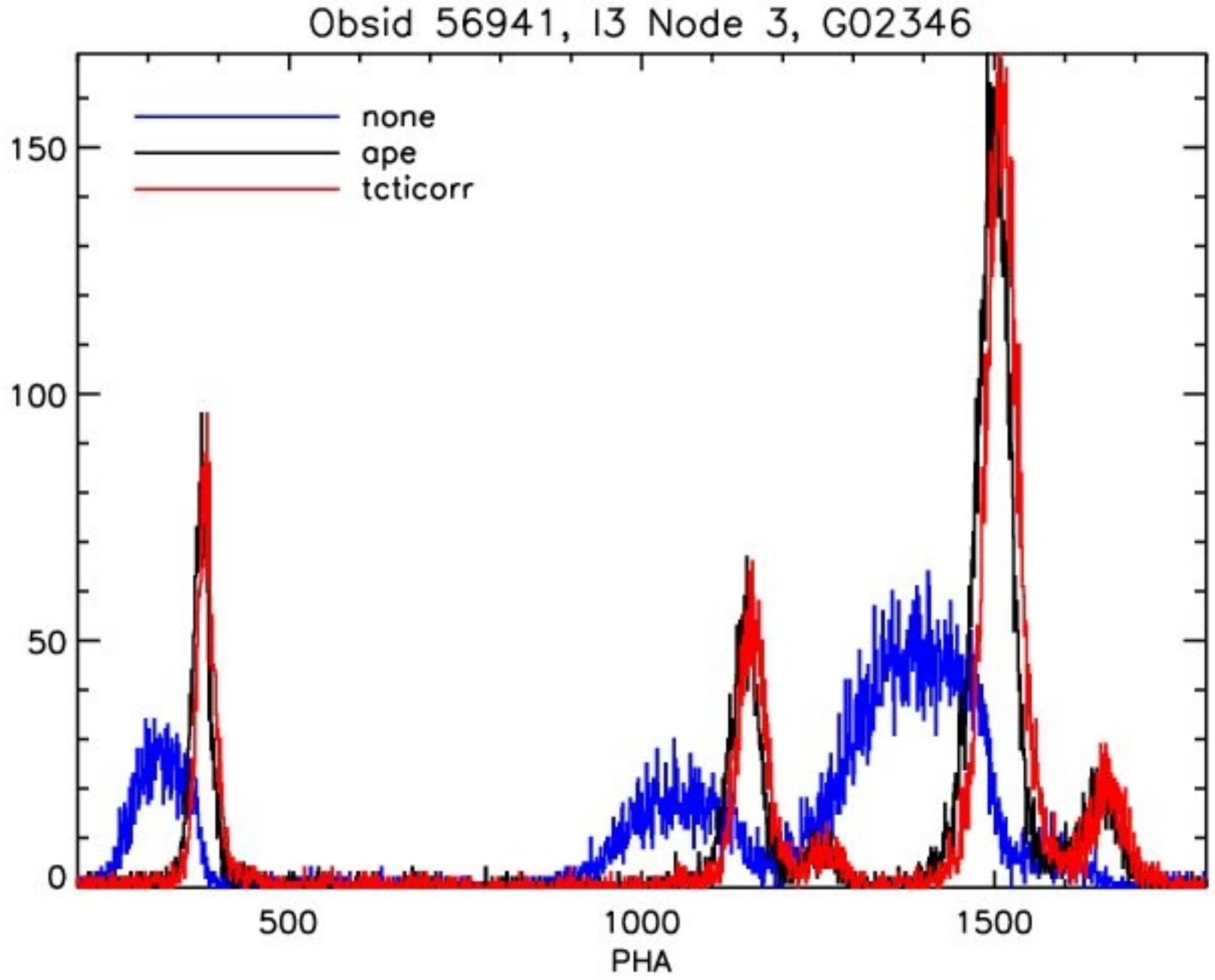
Given evt1 and MTL files from archive:

1. Run acis_process_events on evt1 file with default settings to ensure most recent CALDB products applied. Output: evt1_out1
2. Run tcticorr.pro on evt1_out1. Output: evt1_out2
3. Run acis_process_events on evt1_out2 with apply_cti=no, doevtgrade=yes to regrade. Specify gain and tgain files (file names read from header of evt1_out1). Edit eventdef parameter to remove PHAS column so CTI correction cannot be reapplied. Output: evt1_out3
4. Update CTI-related keywords in evt1_out3 header and add new one, CTITEMP, which gives % CTI change per degree values used.

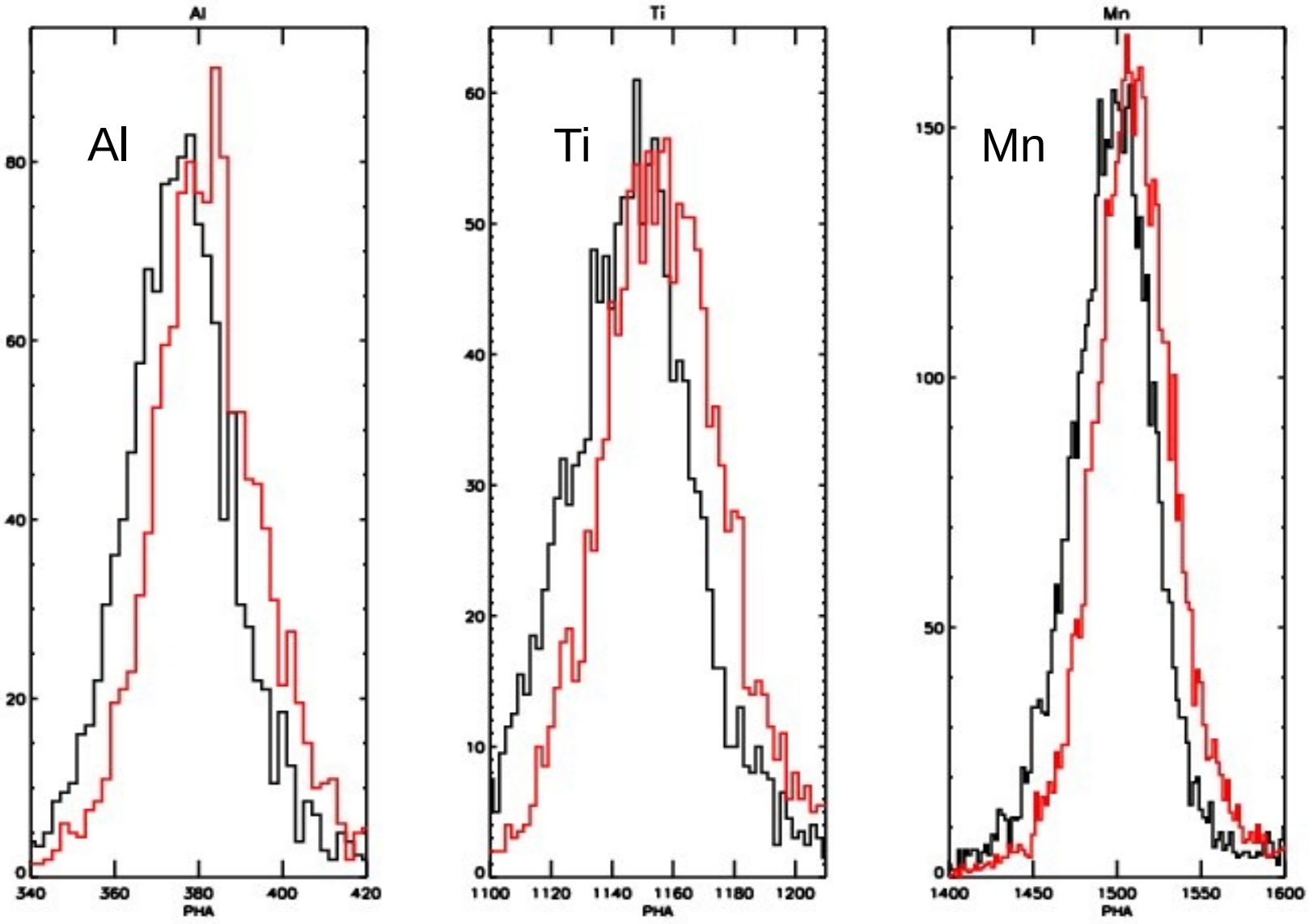
Note: Released version of code will execute steps 3 & 4.

Example: warm ECS observation

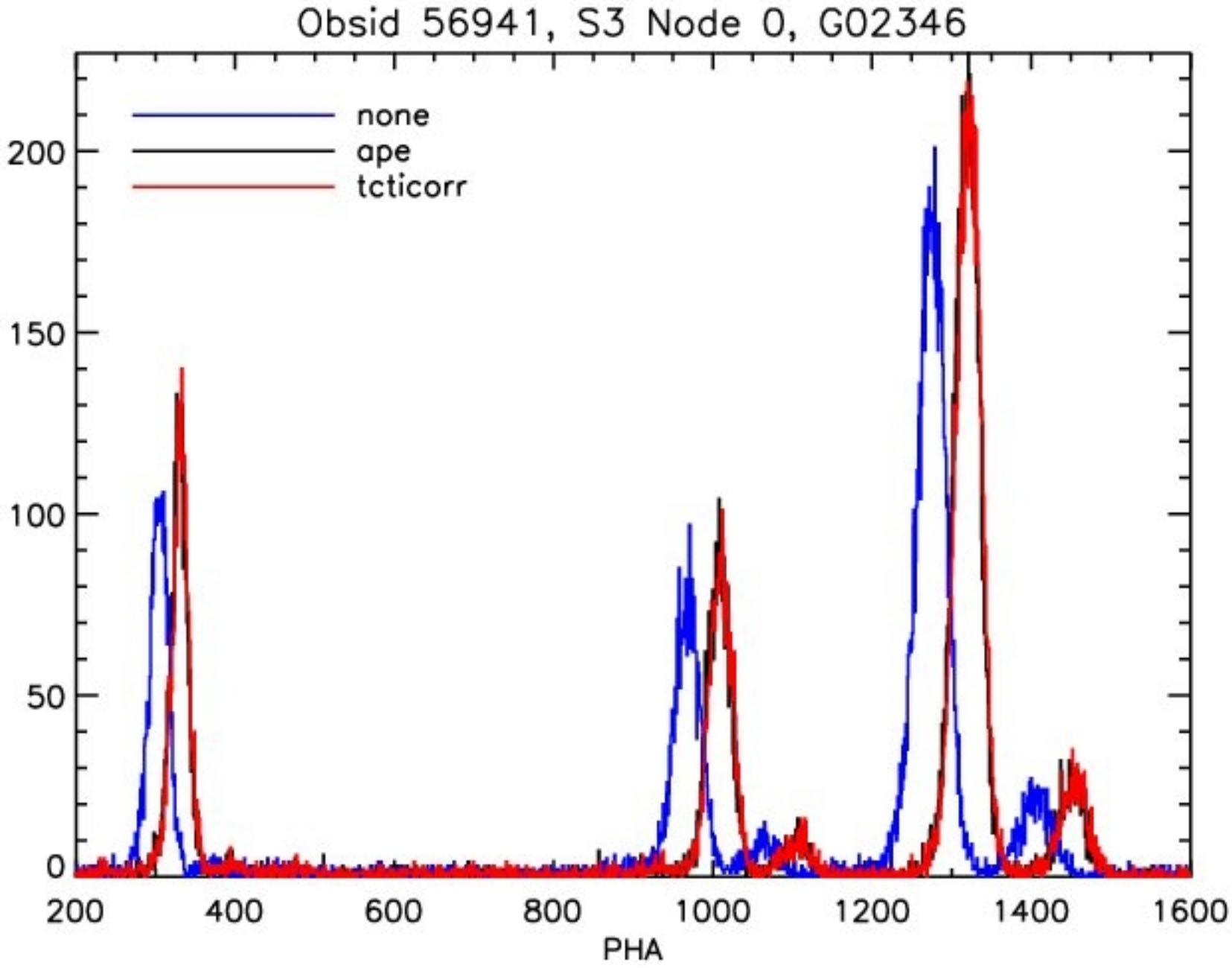



Obsid 56941, I3 Node 3, G02346
none
ape
tcticorr
150
100
50
0
500
1000
1500
PHA

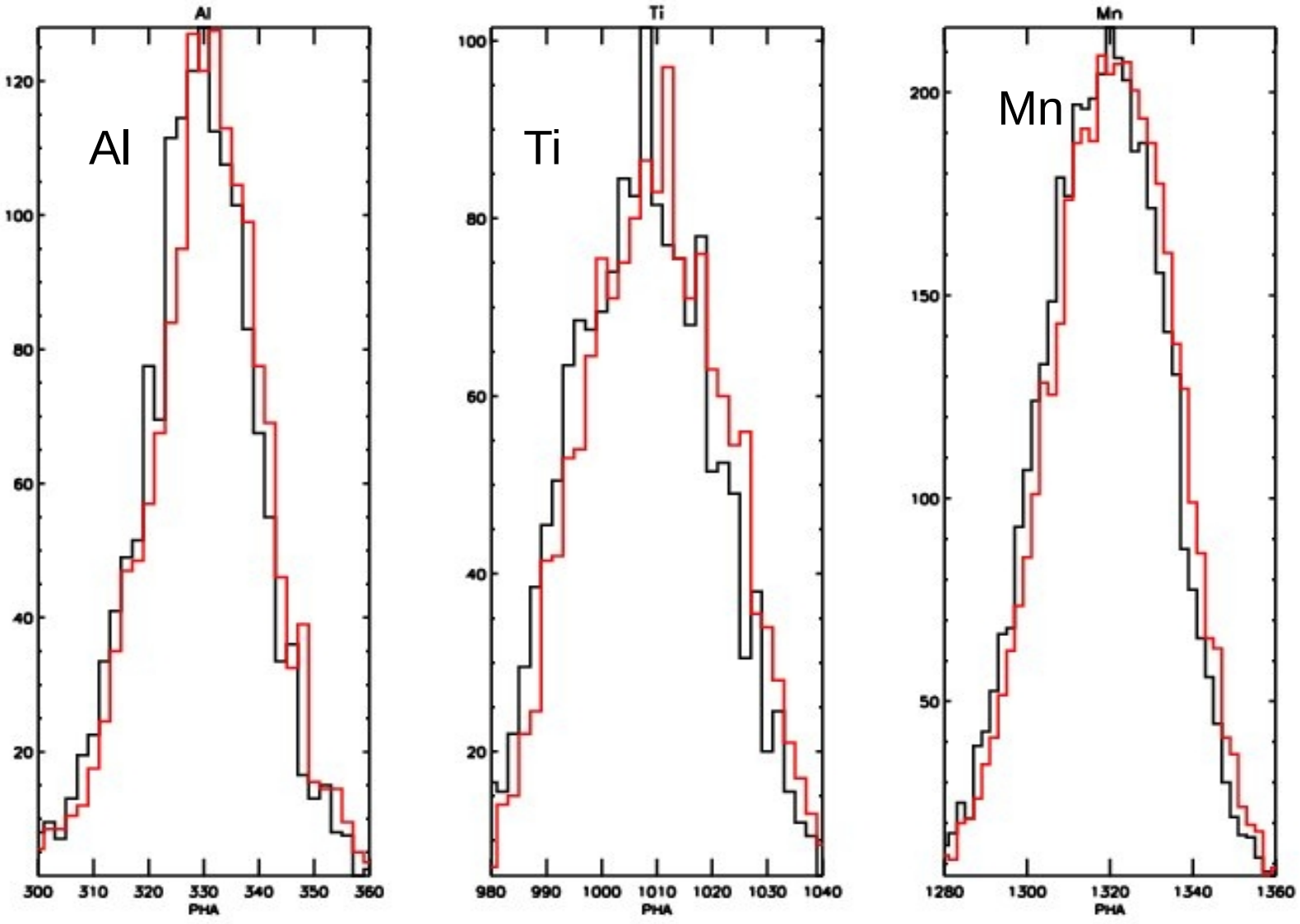
I3 Node 3: Close-up. Black=ape, red= tcitcorr



Shifts of approx 6 (Al Kα), 9 (Ti Kα), and 13 (Mn Kα) ADU


Obsid 56941, S3 Node 0, G02346
none
ape
tcticorr
200
150
100
50
0
200
400
600
800
1000
1200
1400
1600
PHA

S3, Node 0: Close-up. Black=ape, red=tcticorr



Shifts of approx. 1.2 (Al Kα), 2.0 (Ti Kα), and 2.1 (Mn Kα) ADU

Conclusions and Future Work

- We have implemented Catherine's temperature dependent CTI-correction algorithm as an IDL program for contributed software
- To Do:
  1. More testing: compare vs Catherine on BI chips; test on flight data
  2. Check that final output event list works with other CIAO tools
  3. Write thread and post on contributed software page
- We expect to release in ~2 months