

ACIS Extract
An ACIS Source Extraction Package

14 February 2022

(corresponds to *ACIS Extract* code version 2022feb14)

Patrick Broos, Leisa Townsley, Konstantin Getman, Franz Bauer

Department of Astronomy and Astrophysics,
Penn State University

Contents

1	Introduction	6
1.1	XMM Point Sources	7
2	What's New?	8
2.1	AE Version 2022feb14 ¹	8
2.2	AE Version 2021feb4 ²	8
2.3	AE Version 2019jul29	9
2.4	AE Version 2018jun14	9
2.5	AE Version 2018may10	10
2.6	AE Version 2018feb06	10
2.7	AE Version 2016sep22	11
2.8	AE Version 2016may25	11
2.8.1	Off-axis	12
2.8.2	On-axis, I-array aimpoint	12
2.8.3	Judge the fidelity of the PSF models yourself	13
2.8.4	Technical Details of this release	14
3	Getting Started—a two-ObsID example	16
3.1	Create a New Directory for Your AE Work	16
3.2	Inputs to AE	16
3.3	Add Point Sources to AE	18
3.4	Basic Extraction	19
3.5	Review Your Catalog and Extractions	20
3.6	Spectral Fitting	20
3.7	Collating the Results	20
4	Citation	24
5	Algorithms	25
5.1	Point Spread Function Images	25
5.1.1	HRMA Model	25
5.1.2	Post-HRMA Blurring	25
5.1.3	PSF Cropping and Aperture Correction	26
5.2	Extraction and Mask Regions	27
5.3	Source Positions	27
5.4	Source Spectra, Single Observation	28
5.5	Background Spectra, Single Observation	28
5.6	Multi-ObsId Spectra	28
5.6.1	Source Spectrum	29
5.6.2	Background Spectrum	29
5.6.3	Background Scaling	30
5.7	Grouping of Spectra	31
5.8	RMFs	32
5.9	ARFs	32
5.10	Broad Band Photometry	33
5.10.1	Source and background counts with limits.	33
5.10.2	Background scaling, net counts with limits	33
5.10.3	Source validity/significance	34
5.10.4	Photon Flux	36
5.10.5	Characterizations of the background subtracted OBSERVED spectrum	36

¹<http://personal.psu.edu/psb6/TARA/code/>

²<http://personal.psu.edu/psb6/TARA/code/>

5.10.6	Characterizations of the background subtracted INCIDENT spectrum	37
5.11	Light Curves	37
5.12	Automated Spectral Fitting	39
5.12.1	Background Spectra and the C-statistic	40
5.13	Discarding Observations	45
5.13.1	Overlap Rejection	45
5.13.2	Optimization for Source Validity (PROB_NO_SOURCE statistic)	45
5.13.3	Optimization for Source Position	46
5.13.4	Optimization for Photometry and Spectral Fitting	46
5.13.5	Optimized merges may hide source variability	46
5.14	Diffuse Sources	46
5.14.1	The slippery concept of a region's area on the sky	46
5.14.2	Multiple observations should be combined in surface brightness units	47
5.14.3	Analysis of background	47
6	Installation	48
6.1	Unix Shells	48
6.2	Tips on Minimizing Execution Times	50
6.2.1	Overhead When Launching Data Processing Tools	50
6.2.2	Parallel Processing	50
6.2.3	Filesystem Performance	50
6.2.4	NFS Performance Issues	51
6.2.5	Filesystem Congestion from Other Software	51
6.2.6	Algorithmic Shortcuts	51
7	Using <i>ACIS Extract</i>	52
7.1	Recipes	52
7.1.1	Point Sources	52
7.1.2	Diffuse Sources	56
7.1.3	<i>XMM</i> Point Sources	58
7.2	Source Management with the <i>ae_source_manager</i> Tool	59
7.3	General Parameters	60
7.4	Default Parameters	61
7.5	The <i>ae_make_catalog</i> Tool	62
7.6	The <i>ae_standard_extraction</i> Tool	63
7.6.1	Model-based Background Regions	63
7.6.2	Circular Background Regions with Circular Masking	64
7.6.3	Circular Background Regions with Model-based Masking	65
7.7	Adjust BACKSCAL Stage	67
7.7.1	Usage and Inputs	67
7.8	MERGE_OBSERVATIONS Stage	69
7.8.1	Actions	69
7.8.2	Usage and Inputs	71
7.8.3	Data Products	72
7.9	CHECK_POSITIONS Stage	75
7.9.1	Actions	75
7.9.2	Usage and Inputs	75
7.9.3	Data Products	76
7.9.4	Choosing Among Position Estimates	76
7.9.5	PSF and Neighborhood Image Binning	76
7.10	SHOW_REGIONS Stage	78
7.10.1	Actions	78
7.10.2	Usage and Inputs	78
7.10.3	A Complementary Visualization Technique	79

7.11	FIT_SPECTRA Stage	80
7.11.1	Actions	80
7.11.2	Usage and Inputs	81
7.11.3	Troubleshooting	83
7.11.4	Data Products	84
7.12	COLLATED_FILENAME Stage	85
7.12.1	Usage and Inputs	85
7.13	Caveats	86
7.14	Accessory Tools	88
7.14.1	Verify the astrometric alignment of individual ObsIDs	88
7.14.2	Convert source labels to source names	88
7.14.3	Constructing Exposure Maps	89
7.14.4	Grouping	90
7.14.5	Time-resolved extractions	90
7.14.6	Comparing event data with a PSF model	90
7.14.7	PSF “Hook” Feature	92
7.14.8	Printing spectra	92
7.14.9	Reviewing spectral fits	93
7.14.10	Catalog matching	94
7.14.11	Adding source properties	94
7.14.12	Performing a spectral fit using an existing script	94
7.14.13	Scanning for Residual Afterglow Events	94
7.14.14	Table generation	94
A	Running Multiple Instances of <i>ACIS Extract</i>	96
B	Comparison of Wachter and <i>cplinear</i> Background Models	96
C	Correction for Photometric Bias	101
D	Standardized Directory Trees and Data Products	102
D.1	Input to AE	102
D.2	Point Source Output from AE	103
D.3	Diffuse Source Output from AE	106
E	Low-level AE Stages	107
E.0.1	Source Lists and Catalogs	107
E.1	CONSTRUCT_REGIONS Stage	107
E.1.1	Actions	107
E.1.2	Usage and Inputs	108
E.1.3	Data Products	109
E.2	EXTRACT_EVENTS Stage	110
E.2.1	Actions	110
E.2.2	Usage and Inputs	110
E.2.3	Data Products	111
E.3	EXTRACT_SPECTRA Stage	112
E.3.1	Actions	112
E.3.2	Usage and Inputs	112
E.3.3	Data Products	114
E.4	TIMING Stage	115
E.4.1	Actions	115
E.4.2	Usage and Inputs	115
E.4.3	Data Products	115
E.5	EXTRACT_BACKGROUND Stage	117
E.5.1	Actions	117

E.5.2	Usage and Inputs	117
E.5.3	Data Products	118
E.6	<i>ae_better_backgrounds</i> Tool	119
E.6.1	Usage and Inputs	119

F Additional Release History 121

F.1	AE Version 2016feb1	121
F.2	AE Version 2015jul7	121
F.3	AE Version 2014sep10	122
F.4	AE Version 2014feb17	122
F.5	AE Version 2014feb5	123
F.6	AE Version 2013mar6	123
F.7	AE Version 2012nov1	123
F.8	AE Version 2012apr05	124
F.9	AE Version 2012-03-19	124
F.10	AE Version 2011-04-06	125
F.11	AE Version 2011-03-15	125
F.12	AE Version 2010-02-26	126
F.13	AE Version 2009-12-15	126
F.14	AE Version 2009-12-01	127
F.15	AE Version 2009-08-09	127
F.16	AE Version 2009-06-22	128
F.17	AE Version 2009-01-27	129
F.18	AE Version 2008-06-09	130
F.19	AE Version 2008-04-29	131
F.20	AE Version 2008-03-03	131
F.21	February, 2008 (AE Version 3.175)	132
F.22	January, 2008 (AE Version 3.173b)	132
F.23	January, 2008 (AE Version 3.172)	132
F.24	April, 2007 (AE Version 3.135)	133
F.25	March, 2007 (AE Version 3.128)	133
F.26	February, 2007 (AE Version 3.125)	134
F.27	October, 2006 (AE Version 3.107)	134
F.28	September, 2006 (AE Version 3.101)	135
F.29	August, 2006 (AE Version 3.100)	135
F.30	January, 2006 (AE Version 3.96)	136
F.31	December, 2005 (AE Version 3.94)	136
F.32	November, 2005 (AE Version 3.91)	136
F.33	October, 2005 (AE Version 3.89)	136
F.34	July, 2005 (AE Version 3.79)	136
F.35	January, 2005 (AE Version 3.65)	137
F.36	October, 2004 (AE Version 3.56)	137
F.37	September, 2004 (AE Version 3.51)	137
F.38	August, 2004 (AE Version 3.42)	138
F.39	June, 2004 (AE Version 3.34)	138
F.40	February, 2004 (AE Version 3.19)	139
F.41	November, 2003 (AE Version 3.5)	140
F.42	September, 2003 (AE Version 3.0)	141
F.43	May, 2003 (AE version 2.22)	142
F.44	10 February, 2003	142
F.45	9 January, 2003	142
F.46	Initial Release, October 2002	142

1 Introduction

ACIS Extract (AE) is a software package that provides innovative and automated solutions to the varied challenges found in the analysis of X-ray data taken by the ACIS instrument on NASA’s *Chandra* observatory (Broos et al., 2010, 2012; Broos & Townsley, 2016). AE is written in the *IDL* language. What does AE do? Well, suppose you have Level 2 event data from one or multiple ACIS observations, and you have a catalog of proposed point sources. AE can perform virtually all the data processing and analysis tasks that lie between your Level 2 ACIS data and publishable L^AT_EX tables of point-like and diffuse source properties and spectral models.

The Point Source Catalog

Source detection is not part of AE. However, source position estimates and source significance statistics produced by AE can be used to improve a proposed catalog of point sources.

The Extraction Regions

AE constructs two sets of regions for the point sources: The source extraction region are contours of the local PSF enclosing whatever PSF fraction is desired. Crowded fields are handled by shrinking the extractions regions so they do not overlap. The “mask regions”, which completely cover the sources, are used to construct a background data set.

Extraction of Event Data and Backgrounds

Source and scaled local background spectra are extracted.

Calibration Products

ARFs and RMFs are constructed for each source. The ARFs are corrected to account for the light missed by the finite extraction regions.

Timing Analysis

Source variability is quantified. Four data products are produced for each source:

- A scatter plot of event arrival times and energies, plus the observed and uniform cumulative distribution functions used to quantify variability (Figure 18).
- A “grouped light curve”, i.e. source flux measured in a set of independent, unequally sized time bins with constant significance (Figure 9, Figure 10).
- The median energy of events in each time bin.
- An “adaptively smoothed light curve”, i.e. source flux measured in a set of overlapping (i.e. correlated), unequally sized time intervals with constant significance.

BEWARE: these time series make no distinction between events produced by the source and background events. Variation in the aperture background rate among ObsIDs can lead to a spurious indication of source variability. Such background variation can occur because the aperture size can vary among ObsIDs (e.g. due to variations in off-axis angle), because background from neighboring sources can vary among ObsIDs, and because FI and BI devices have different levels of instrumental background.

Photometry

Source photometry is performed over any desired set of energy bands.

Spectral Modeling

Source spectra are grouped and spectral models are fit with *XSPEC*.

Collation of Results

Source properties, statistics, photometry, and spectral models are collated into a large FITS file for analysis, generation of L^AT_EX tables, etc.

Visual Review

Source extraction region files help the observer review the observation for spurious and missed sources. A variety of interactive plots are produced showing various source properties across the catalog.

Multiple Observations

The source catalog can be extracted from any number of observations (at different aim points or roll angles). AE combines the extraction data products prior to position estimation, photometry, timing analysis, and spectral modeling.

Diffuse Sources

If you can define extraction and background regions for diffuse sources, then AE can extract and analyze them in much the same way as is done for point sources.

The implementation strategy for AE is to use *CIAO* tools for specific computations whenever possible, and *DS9* for display and editing of extraction regions. Section 5 of this manual describes the algorithms used to compute various data products, with an emphasis towards scientific concerns. Section 6 and section 7 describe how to install and use the AE software.

The authors happily acknowledge that AE is derived from the ideas, software, and testing efforts of David Alexander, George Chartas, Eric Feigelson, Nicolas Grosso, Scott Koch, Bertil Olsson, Yohko Tsuboi, and the authors of various *CIAO* threads and manuals. Support for this effort was provided by NASA contract NAS8-38252 to Gordon Garmire, the ACIS Principal Investigator

Please contact Patrick Broos³ if you have problems, questions, or suggestions for improving these tools. Please join the *ACIS Extract* email list⁴ to receive announcements of AE releases.

Note that several other tools for extracting *Chandra* data have been developed:

- *XAssist*⁵
- *yaxx*⁶
- *psextract*⁷
- *speceextract*⁸
- the Level 3 pipeline at the CXC

1.1 XMM Point Sources

AE can also perform simple extractions of point sources observed with the EPIC instrument on *XMM* (§7.1.3).

³<mailto:patrick.broos@icloud.com>

⁴<http://lists.psu.edu/cgi-bin/wa?A0=L-ASTRO-ACIS-EXTRACT>

⁵<http://xassist.pha.jhu.edu/xassist/index.jsp>

⁶<http://cxc.harvard.edu/contrib/yaxx/>

⁷<http://cxc.harvard.edu/ciao/ahelp/psextract.html>

⁸<http://cxc.harvard.edu/ciao3.3/ahelp/speceextract.html>

2 What's New?

These release notes are a good way to discover capabilities that have been added since the last time you used AE.

2.1 AE Version 2022feb14⁹

- Due to a bug in *CIAO* version 4.14 (related to the “[opt null=xx]” syntax, HelpDesk ticket #023895), *AE* versions prior to this one are not compatible with *CIAO* version 4.14.
- *AE* versions prior to this one are not compatible with machines that use the Apple M1 processor. See bug report at the Google Group *idl-pvwave*¹⁰.
- The algorithm that chooses vertexes for the *cplinear* background model (§5.12) has been redesigned.
- This release includes the most recent versions of the scripts and recipe documents¹¹ (§7.1) that we use locally for our own *AE* processing. These are continuously improved as we use them in our own ACIS projects.

2.2 AE Version 2021feb4¹²

- CALDB 4.9.0¹³ introduced a new class of “bad CCD pixels”—the “framestore shadow rows” at the bottom of every CCD. Standard L1-to-L2 processing using CALDB 4.9.0 or later now excludes all events from those rows. However the *Chandra* archive has not yet been fully reprocessed, so some older observations in the archive retain those rows.

When exposure maps are constructed, framestore shadow rows must be treated as either “live” or “dead” area, to match how they were classified during L1-to-L2 event processing. The *CIAO* tool making that decision is *mkinstmap*. Unfortunately, *mkinstmap* now requires a “BPMASK” qualifier to its *detsubsys* parameter in order to operate correctly:

```
detsubsys="ACIS-x;BPMASK=0x3F9FFF"  
OR  
detsubsys="ACIS-x;BPMASK=0X03FFFF"
```

where “x” is the CCD number (0–9). The *AE* tool *ae_make_emap* (7.14.3) now calls *mkinstmap* in that way. See the document Caveats about ACIS Frame Store Shadow Bad Pixels¹⁴

If you have your own script for building exposure maps, you should revise it accordingly. If you use a *CIAO* contributed script for building exposure maps, you should contact HelpDesk to find out if it has been updated to include the proper **BPMASK** qualifier.

- *AE* is compatible with *CIAO* version 4.13.

NOTE THAT CIAO VERSIONS 4.11 AND EARLIER ARE NOT RECOMMENDED BY THE CXC, BECAUSE THEY ARE INCOMPATIBLE WITH IMPORTANT CALIBRATION IMPROVEMENTS PROVIDED BY CALDB VERSION 4.9.0 AND BEYOND.

- *AE* is NOT compatible with aspect files modified by the *CIAO* tool *asp_offaxis_corr* (introduced in *CIAO* v4.13).
- Note that *AE* requires *MARX*¹⁵ version 5.3 to 5.5.1 (Section 6). Later versions are NOT acceptable.
- *AE* now requires *TARA* version 2021jan6¹⁶.

⁹<http://personal.psu.edu/psb6/TARA/code/>

¹⁰<https://groups.google.com/g/idl-pvwave/c/Jq5OM4ATtgE>

¹¹<http://personal.psu.edu/psb6/TARA/procedures/>

¹²<http://personal.psu.edu/psb6/TARA/code/>

¹³https://cxc.harvard.edu/caldb/downloads/Release_notes/CALDB_v4.9.0.html

¹⁴https://cxc.cfa.harvard.edu/ciao4.14/caveats/acis_shadow_badpix.html

¹⁵<https://space.mit.edu/ASC/MARX/inbrief/news.html>

¹⁶<http://personal.psu.edu/psb6/TARA/code/>

- *AE* has always required that your unix shells define the alias *ciao*. The form of that alias depends on whether you install¹⁷ *CIAO* with the traditional *ciao-install* script, or via the *conda* package manager (Section 6.1).
- For technical reasons, the TIMING stage no longer constructs barycentric corrections to event timestamps.
- Section 6.2 (Tips on Minimizing Execution Times) has been expanded.

2.3 AE Version 2019jul29

- **As the *Chandra* archive grows, more and more ACIS studies involve source extractions that do not share identical calibrations.** Front-illuminated and back-illuminated ACIS detectors have moderately different Effective Area curves. ACIS-only and zeroth-order HETG configurations have very different Effective Area curves. Even observations taken on the same detector in the same configuration suffer from time-dependent Effective Area curves due to time-varying contamination on the ACIS Optical Blocking Filters. These calibration variations among source extractions introduce variable biases into certain uncalibrated source properties (e.g., “count rate”, “hardness ratio”, mean or median event energy) and introduce source-dependent calibration errors into other source properties (e.g., Photon Flux, Energy Flux, inter-ObsID variability metrics).
 - As an uncalibrated property of the observed events, Median Energy (ENERG_PCT50_OBSERVED, §5.10) depends on the shape of the Effective Area curves for the extractions of the source.
 - Because Mean Effective Area (MEAN_ARF, §5.10.4) depends on the shape of the Effective Area curves for the extractions of the source, Photon Flux (FLUX2, §5.10.4) and Energy Flux (§7.14.14) exhibit source-dependent calibration errors.
 - The accuracy of inter-ObsID variability metrics (MERGE_KS and MERG_CHI, §7.8) depends on the consistency of the Effective Area curves for the extractions of the source. Sources observed with both the ACIS-only and zeroth-order HETG configurations will often produce spurious indications of inter-ObsID variability (because those configurations have very different Effective Area curves). Inter-ObsID variability metrics are proper *p*-values only when all observations of the source have identical Effective Area curves. When Effective Area shape is variable, variability of a source can be proven only via spectral fitting of each extraction.
- AE now requires the “2019jul29” or later version of the *TARA* package¹⁸.
- This version of *AE* is compatible with *DS9* version 8.0, and with *DS9* version 7.5, but is not compatible with *DS9* version 7.6 (i.e. the version that is distributed with *CIAO* 4.10).
- The naming convention for AE fitting scripts (§5.12) has changed.
- Event timestamps with barycentric correction are now stored in files named *source_barycentric.evt*.
- This release includes the most recent versions of the scripts and recipe documents¹⁹ (§7.1) that we use locally for our own *AE* processing. These are continuously improved as we use them in our own ACIS projects.

2.4 AE Version 2018jun14

- This release includes the most recent versions of the scripts and recipe documents²⁰ (§7.1) that we use locally for our own *AE* processing. These are continuously improved as we use them in our own ACIS projects.
- We have run a lot of point source extraction under *CIAO* version 4.10, without incident. So, I believe that *AE* is compatible with *CIAO* version 4.10.
- **Do not try to use the SHOW_REGIONS stage (§7.10) with version 7.6 of *DS9* (e.g. the version that is distributed with *CIAO* 4.10).** That version of *DS9* has some serious problems with binning event lists in multiple *DS9* frames.
- The 25-Jul-2016 or later version of Wayne Landsman’s *IDL* Astronomy Library²¹ (AstroLib) is now required.

¹⁷<https://cxc.harvard.edu/ciao4.12/download/>

¹⁸<http://personal.psu.edu/psb6/TARA/code/>

¹⁹<http://personal.psu.edu/psb6/TARA/procedures/>

²⁰<http://personal.psu.edu/psb6/TARA/procedures/>

²¹<http://idlastro.gsfc.nasa.gov/homepage.html>

2.5 AE Version 2018may10

- As the tool *ae_better_backgrounds* (§E.6) builds a background region for a source aperture, it tries to simultaneously achieve several goals, as described in §7.6.1. If the background region grows to the maximum size allowed (1 square arcminute) without achieving those goals, the algorithm must decide which sub-optimal background region it should adopt. In previous versions that maximum sized region was adopted, but often there was a smaller region that came closer to the goals. In this version, the best available region is adopted.
- A flaw in previous versions of *ae_better_backgrounds* sometimes produced background regions that were less compact than they should have been. Sources with strong exposure variations nearby (e.g. sources near chip gaps) were most at risk. This version has eliminated that problem.
- I recently discovered and fixed bug in the *AE* fitting scripts that use two-component models (tbabs_2vapec.xcm, tbvarabs_2vapec.xcm, windtabs_2vapec.xcm). Prior to this release, the NH-corrected flux calculations for the second component (F2c... values) were calculated without setting NH to zero, so F2c... values were probably equal to the apparent flux values (F2...).
- The *ae_spectra_viewer* tool (§7.14.9) has several new features and optional inputs.
- I cannot guarantee that *AE* will run with *IDL* versions older than 8.5.1.
- This release includes the most recent versions of the scripts and recipe documents²² (§7.1) that we use locally for our own *AE* processing. These are continuously improved as we use them in our own ACIS projects.

2.6 AE Version 2018feb06

• Comments on the Validation of Source Candidates

For sources with multiple observations, Broos et al. (2010, Section 6.2) point out that a subset of the observations may produce higher quality measurements of source properties than the full set of observations would. This is often true when the observations have very different off-axis angles, and thus very different angular resolutions and degrees of crowding. In our own work, we continue to build spectra and photometry from a set of observations that favors the photometric signal-to-noise ratio (§5.13) and to estimate the source position from a set of observations that favors small position uncertainty.

In our previous studies, source *validation* followed the same strategy; the validity of a source (in each energy band) was calculated from the set of observations in which the source was most valid (via the MERGE_FOR_PB option to the MERGE stage). Broos et al. (2010, Section 6.2) point out that this selection process increases sensitivity to variable sources at the cost of an increased false detection rate arising from the additional number of data sets that are searched.

Recently, while reducing a *Chandra* target with a large number of overlapping ObsIDs, we found this source validation strategy produced an unreasonable number of sources validated by a subset of the ObsIDs, since the number of possible ways to combine N ObsIDs grows rapidly with N . Thus, we have recently adopted a different strategy to balance false detection rate and sensitivity to variable sources. We choose to evaluate source validity on a **pre-defined**, small set of ObsID combinations. Those combinations do not depend on characteristics of the data; they consist simply of (1) each ObsID by itself and (2) ObsIDs with similar off-axis angles. Empirically-determined off-axis angle ranges are 0–3 arcmin, 2–6 arcmin, 5–9 arcmin, 8–14 arcmin, and >13 arcmin. These ranges overlap to avoid missing sources at the boundaries.

In this and future *AE* versions, a warning message will be printed when the MERGE_FOR_PB option is used, to encourage observers to think about which combinations of ObsIDs should be allowed to validate their source candidates.

- In multi-ObsID targets, we have found that backgrounds constructed by the *ae_better_backgrounds* tool (§7.6.1) can be biased downward (biasing source significances upward) when a neighbor’s aperture contains no significant point source signal. Such an empty aperture tends to be excluded from the background region when it exhibits random positive fluctuations of NET_CNTS, and tends to be included when it exhibits negative fluctuations of

²²<http://personal.psu.edu/psb6/TARA/procedures/>

NET_CNTS. The *ae.better_backgrounds* tool now ignores models of neighboring sources that are very weak. This can, of course, bias the backgrounds in the other direction, allowing small levels of power from neighboring sources to sneak into the background region. Balancing these two forms of selection bias is more an art than a science, in my opinion. The default significance threshold used to ignore neighbors can be changed via the optional *ae.better_backgrounds* parameter NEIGHBOR_INVALID_THRESHOLD.

- We have added a third variability metric (MERG_CHI) to the MERGE stage (§ 7.8)—the p-value for the χ^2 statistic computed on the single-ObsID (background-subtracted) photon fluxes.
- We recommend installing the “2017oct21” or later version of the *TARA* package²³ to avoid a bug (in *median_with_ci.pro*) that may degrade the accuracy of the tools *match_xy*, *ae.interObsID_astrometry*, and *evaluate_astrometry*.
- The syntax for calling to tool *ae.make_emap* has significantly changed (7.14.3).
- Calls to the *CIAO* tool *dmimgpick* have been revised (as needed) to work around the bug described in HelpDesk Ticket #020605.
- All calls to IDL’s buggy *binomial.pro* routine have been replaced with an implementation obtained from Numerical Recipes (*binomial_nr.pro*).
- This release includes the most recent versions of the (complex) scripts and recipe documents that we use locally for our own AE processing (§7.1). These are continuously improved as we use them in our own ACIS projects. Several tools used in these recipes (including *recon_detect* and *ae.interObsID_astrometry*) have been significantly revised.

2.7 AE Version 2016sep22

- In the fall of 2015 the AstroLib²⁴ (along with other libraries of FITS tools) stopped supporting IEEE 754 special values (e.g. NaN, Infinity) in FITS header keywords. It turns out that the FITS standard does not allow those values in headers—a disappointing design decision in my opinion. *AE* has used these special values in header keywords since its first release. I’ve worked around this crisis using the new wrapper routines *psb_xaddpar* and *psb_xpar*. This version of *AE* should be compatible with any AstroLib version after 22-Sep-2011.

***AE* versions prior to this release MUST USE AstroLib versions prior to 22-Sep-2015! If you have already run *AE* with AstroLib version 22-Sep-2015 or later, then your analysis may have hidden damage; contact me for assistance.**

- The “padding” added to neighborhood images before reconstruction (to deal with the “wrapping” that occurs in convolutions) has been changed from zeros to the background level in the image; this reduces reconstruction artifacts.
- PSF images are now constructed using only the set of CCDs that are present in the observation. This should fix some edge effects we see at the boundaries between FI/BI detectors.program
- This release includes the most recent versions of the scripts and recipe documents that we use locally for our own AE processing (§7.1). These are continuously improved as we use them in our own ACIS projects.
- As described in the 2015jul7 release notes, ***AE* versions prior to 2015jul7 are not compatible with *CIAO* versions 4.7 and later.**

2.8 AE Version 2016may25

We wish to make you aware of two problems in the point spread function (PSF) images built by all previous ACIS Extract (AE) versions that employed *MARX* 5.0, 5.1, 5.2, or 5.3 (i.e. AE releases 2014feb5, 2014feb17, 2014sep10, 2015jul7, 2016feb1, and 2016feb1). One of these problems has been resolved by the release of *MARX* 5.3, and a preliminary fix for the other problem is available in this release of AE.

²³<http://personal.psu.edu/psb6/TARA/code//>

²⁴<http://idlastro.gsfc.nasa.gov/homepage.html>

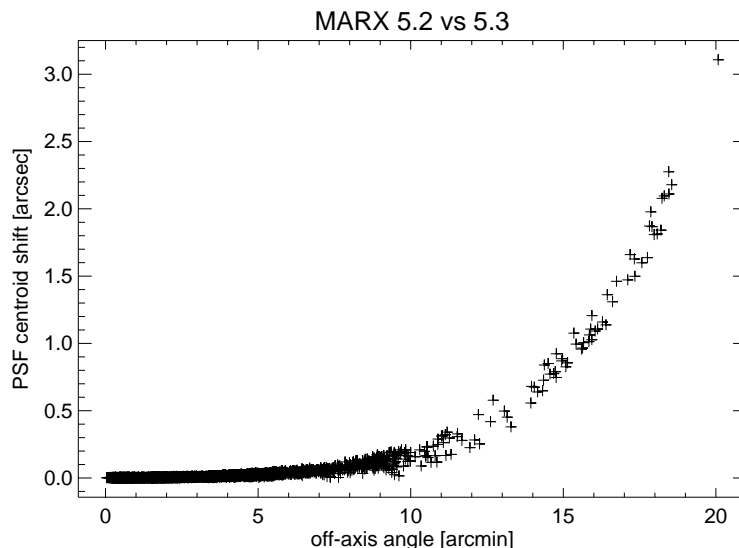
Note that installing *MARX* on OS-X (the Mac operating system) is currently a complex task that deserves your careful attention. On many Macs, *MARX* will install without incident but will not operate correctly with certain inputs. The problem seems to be bugs in Apple's latest C compiler (called *clang*). See <http://space.mit.edu/cxc/marx/inbrief/install.html#mac-os-x> for details and instructions or contact marx-help@space.mit.edu. Note that this issue arises whether *MARX* is installed by hand or with the *CIAO* tool *install_marx*.

2.8.1 Off-axis

The first problem is a *MARX* bug introduced in version 5.0 that produces a "hole" in off-axis PSFs; this problem was resolved in *MARX* 5.3²⁵. Although the example far off-axis image²⁶ linked from Chandra Electronic Announcement #142²⁷ looks scary, we expect the impact of this PSF distortion is going to be small for most AE users.

One consequence of this PSF distortion is a shift to the centroid of the PSF, which would affect source positions that are estimated using the PSF (e.g. AE's "correlation" position estimates). The plot below shows that centroid shift between AE PSFs made with *MARX* 5.2 (bug present) and 5.3 (bug fixed). The vast majority of your sources are probably less than 10' off-axis, where the centroid shift is <0.2".

A second consequence of this PSF distortion is that off-axis AE extraction apertures can become heart-shaped, particularly when apertures are smaller than the nominal 90% enclosed fraction.



2.8.2 On-axis, I-array aimpoint

The second problem is that, for data taken at the I-array aimpoint, the AE versions listed above have been producing on-axis PSFs that are narrower than the data. This AE release produces PSFs that are better calibrated near the I-array aimpoint. **Observations taken at the S-array aimpoint are not affected by this problem, and this release does not alter AE PSFs for S-array aimpoint observations.**

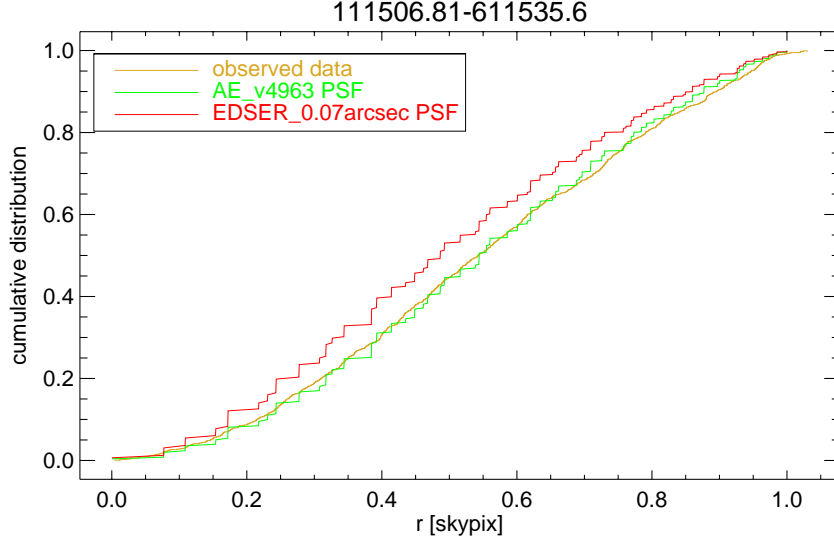
The plot below compares cumulative radial distributions ("encircled energy") of a Chandra-ACIS source (0.5:8 keV) near the I-array aimpoint (yellow), the PSF model (1.5 keV) produced by the AE versions listed above (red, too narrow), and the PSF model (1.5 keV) produced by the AE version released today (green).

In the plot above the data and PSFs are truncated at only 1.0 sky pixel radius, to reduce the effects of background in the data. The plot below extends the cumulative distributions (of the PSFs) out to a radius that encloses nearly 100% of the light. For each PSF the approximate radius that encloses 50% of the light is marked with a blue vertical line.

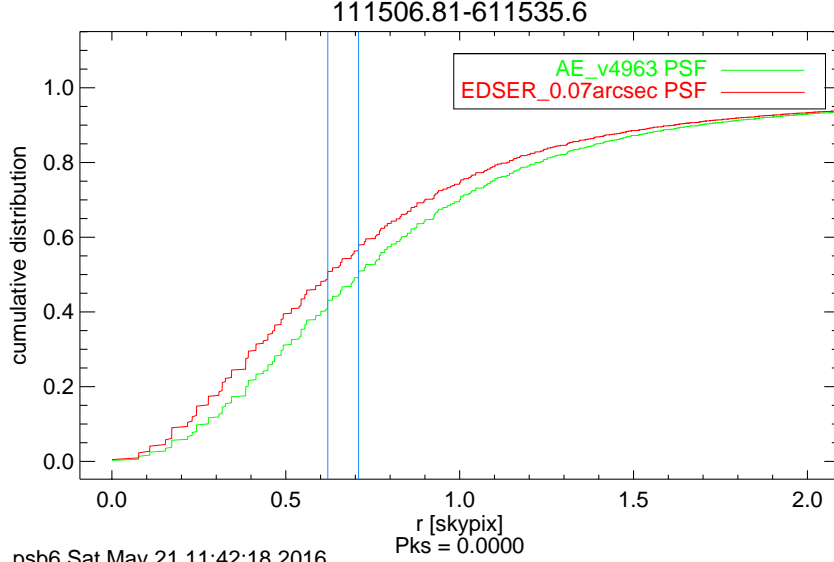
²⁵<http://space.mit.edu/ASC/MARX/>

²⁶<https://github.com/Chandra-MARX/marx/pull/21>

²⁷http://cxc.harvard.edu/announcements/group_embed.html?place=msg/chandra-announce/wbUoGzIlWWk/CTLjBoc9AwAJ



psb6 Sat May 21 11:42:14 2016



psb6 Sat May 21 11:42:18 2016

Today's AE release produces the green PSF above by calling *MARX* with non-standard parameter values (for observations at the I-array aimpoint). Those parameters were tuned by us to match the observed PSF for a few of our sources near the I-array aimpoint. **Our sources are not ideal calibration targets, and this tuning is not an official recommendation from the CXC or the *MARX* team.**

The CXC science thread Using *MARX* to Simulate an Existing Observation²⁸ and the corresponding *CIAO* tool *simulate_psf* are currently under review. If we discover that the recipe for calling *MARX* in this release of AE can be improved, we will revise AE accordingly, and of course if *MARX* is revised we will try to revise AE accordingly.

2.8.3 Judge the fidelity of the PSF models yourself

You can compare your own sources to the PSF images produced by AE very easily, using the AE call below. Simply list the names of the sources of interest in a file, 'review.srclist' below. Specify an energy filter for the observed events, ENERGY_RANGE below. Specify the radius (in sky pixels) of the circular aperture in which the comparison will be performed. (Large radii will increase contamination by background.)

²⁸http://cxc.harvard.edu/ciao/threads/marx_sim/index.html

The *ae_radial_profile* tool operates on "merged" (potentially multi-ObsID) AE products. Supply a `MERGE_NAME` below if you are working with a "named merge" (a merge that lives in a directory below the source directory).

```
idl
.run ae
ae_radial_profile, report, /PLOT, SRCLIST_FILENAME='review.srclist', ENERGY_RANGE=[0.5,8], $
SRC_RADIUS=1.0, MERGE_NAME=''
```

2.8.4 Technical Details of this release

For future reference, previous releases of AE using *MARX* 5.0 or greater have called *MARX* in the same way for S-array and I-array aimpoints:

- AspectBlur=0.07 (arcsec) in the call to marx.
- pix_adj=(keyword PIX_ADJ from FITS header) (normally "EDSER") in the call to *marx2fits*.

This release calls *MARX* differently for the two possible observation aimpoints.

For S-array aimpoint:

- AspectBlur=0.07 (arcsec) in the call to marx.
- pix_adj=EDSER in the call to *marx2fits*.

For I-array aimpoint:

- AspectBlur=0.07 (arcsec) in the call to marx.
- pix_adj=NONE in the call to *marx2fits*.

These *MARX* recipes are appropriate only for event data produced using the EDSER option in *acis_process_events*. AE will now produce an error message if your event data were processed differently.

Notes on older AE releases can be found in Appendix [F](#).

3 Getting Started—a two-ObsID example

AE addresses a wide range of issues in the extraction and analysis of point sources in ACIS observations, embraces difficult observational environments such as multiple observations and crowded fields, and provides a lot of visualization feedback to the observer. As one might expect, §7 of this manual, *Using ACIS Extract*, is somewhat long since it tries to thoroughly describe every option. §7.1 includes a reference to my own fairly complex “recipe” for extracting multiple ACIS observations of a field. This recipe is **not** the place to start if you have little experience with ACIS analysis because it tries hard to do a very careful job of the extraction.

If AE or ACIS analysis in general are new to you, then rest assured that a basic extraction of an ACIS source list from a couple of ObsIDs can be performed quite easily (as described in the remainder of this Section) once you’ve gathered the various data products needed as inputs to *CIAO*. **Using AE is complex only if you choose to make your ACIS analysis complex!**

Before starting an AE extraction, carefully follow the instructions for installing AE in §6. Common installation **mistakes** include:

- Your *IDL* installation is older than the required version.
- Your *IDL* Astronomy Library (from Wayne Landsman) is older than the required version.
- Your account is configured such that shells spawned by *IDL* are not in the *csh* family.
- The required Unix shell aliases *ciao* and *heasoft* are not defined, or they generate output to STDOUT.
- Your Unix shell is configured such that the aliases *ciao* and *heasoft* are missing in the **non-interactive** shells used by AE.
- Your *CIAO* installation is broken, or the *CIAO* version is not compatible with AE; the *CIAO* contributed scripts are not installed.

3.1 Create a New Directory for Your AE Work

```
mkdir extract
cd    extract/
```

3.2 Inputs to AE

In later steps of this extraction example we make use of some high-level AE tools (in the file *ae_recipe.pro*) that require that we establish a standardized directory structure holding the usual *CIAO* data products that any extraction tool/recipe would require as input. Appendix D shows an example directory structure for these AE inputs. The best way to set this up is to create Unix symbolic links pointing to your various *CIAO* input data products, as described below.

1. Choose names for your observations, typically the observation ID numbers (e.g. “1875” and “1876”).
2. In the *extract/* directory, make “observation directories” (using the naming convention *obs<name>*) to hold (symbolic links to) the inputs AE requires for each of your ObsIDs. For example:

```
mkdir obs1875 obs1876
```

3. In each observation directory make a symbolic link named *spectral.evt* to a suitable Level 2 event list (§7.3). For example:

```
cd obs1875/
ln -s ../../1875/primary/acis*evt2.fits spectral.evt

cd obs1876/
ln -s ../../1876/primary/acis*evt2.fits spectral.evt
```

4. In each observation directory make a symbolic link named *obs.emap* to a *CIAO* exposure map that corresponds to the field of view of the event data you are giving to AE (§7.3). For example:

```
ln -s ../../1875/full_field.emap obs.emap
```

```
ln -s ../../1876/full_field.emap obs.emap
```

The CXC provides threads and scripts for constructing exposure maps. We use the tool *ae_make_emap* (§7.14).

5. In each observation directory make a symbolic link named *asphist* to a directory containing aspect histogram files for each active CCD. The aspect histogram files must be named *ccd0.asphist*, *ccd1.asphist*, ... Aspect histograms are required by the *CIAO* tool *mkarf*, and are a by-product of constructing an exposure map (§E.3). In the following example, a symbolic link is made to such a directory (named “asphist”) that is left over from using the tool *ae_make_emap* (§7.14) to construct an exposure map.

```
ln -s ../../1875/asphist asphist
```

```
ln -s ../../1876/asphist asphist
```

If you build your exposure map another way, then simply change the symlinks above to point to whatever directory contains your aspect histograms. The aspect histogram files must be named *ccd0.asphist*, *ccd1.asphist*, ...

6. In each observation directory make a symbolic link named *obs.asol* to an *CIAO* aspect file²⁹ covering the time range of your observation³⁰. You will have located this file in order to make the aspect histograms above. For example:

```
ln -s ../../1875/primary/pcad*asol1.fits obs.asol
```

```
ln -s ../../1876/primary/pcad*asol1.fits obs.asol
```

7. If you wish the *CIAO* tools *mkarf*/*mkwarf* to apply the “Dead Area calibration”³² then in each observation directory make a symbolic link named *obs.pbkfile* to the observation’s “parameter block file”. For example:

```
ln -s ../../1875/secondary/acis*pbk0.fits obs.pbkfile
```

```
ln -s ../../1876/secondary/acis*pbk0.fits obs.pbkfile
```

8. In each observation directory make a symbolic link named *obs.mskfile* to the observation’s “mask file”³³, which is used by *mkarf* and *mkwarf*. For example:

```
ln -s ../../1875/secondary/acis*msk1.fits obs.mskfile
```

```
ln -s ../../1876/secondary/acis*msk1.fits obs.mskfile
```

9. In each observation directory build the standard *CIAO* parameter file *ardlib.par*, properly configured for your observation. There is a *CIAO* thread³⁴ that discusses the *CIAO* script *acis_set_ardlib*. For example:

²⁹<http://asc.harvard.edu/ciao/why/asol.html>

³⁰ *MARX* accepts only a single aspect file, not a “stack” of files. If your observation has multiple aspect files, you must concatenate them using *dmmerge*, as discussed in the *CIAO* Data Products Guide³¹. The aspect files must be combined in time order; the string of numbers in the file name (e.g. “pcadf111767021N001_asol1.fits”) refers to the start time of the period for which the aspect solution is valid.

³²<http://cxc.harvard.edu/ciao3.4/why/acisdeadarea.html>

³³http://cxc.harvard.edu/ciao/threads/intro_data/index.html

³⁴<http://cxc.harvard.edu/ciao/threads/badpix/>

```

punlearn ardlib
acis_set_ardlib ../../1875/primary/acis*bpix1.fits absolutepath=yes

cp `paccess ardlib w` ./ardlib.par

plist ./ardlib.par | more

```

10. In the `extract/` directory, make a directory named `point_sources`. The `extract/` directory should now have subdirectories `point_sources/`, `obs1875/`, and `obs1876` (Appendix D).

```

cd ...../extract/
mkdir point_sources

```

Move to `point_sources/`. **This is where all IDL sessions that run AE commands will be launched, and where AE will write most of its data products!** For later convenience, create a symbolic link to the *XSPEC* scripts supplied with AE.

```

cd    point_sources/
idl
    file_link, file_which('xspec_scripts'), '.'
    exit

```

Identify point sources by whatever means you choose (e.g. `wavdetect`, human examination of the data, etc.). Source detection is of course a complex subject, and is beyond the scope of this manual. Read into *IDL* the celestial coordinates of the sources in units of decimal degrees; the specific *IDL* commands to do that depend on how you have recorded your source coordinates; two examples are shown below. Be sure to use adequate precision for the coordinates; any ASCII representation of the coordinates should use 6 digits to the right of the decimal, and I/O in *IDL* should be done in *DOUBLE* precision.

```

idl |& tee -a AE_simple_run.log

; For a FITS catalog from wavdetect:
bt = mrdfits('wavdetect_src.fits', 1)
RA = bt.RA
DEC = bt.DEC

; OR, for an ASCII file (e.g. a \DSnine\ region file) saved in decimal celestial coordinates:
readfmt, 'catalog.reg', '8x,D,D', RA, DEC
; The 8x part of the format specification should be adjusted to
; skip over whatever prefix appears in each row, e.g. "ellipse".
; The "D" format codes force RA and DEC to be DOUBLE precision vectors.

```

For all AE runs we strongly recommend that you start *IDL* in such a way that all screen output is saved to a log file, e.g. via the Unix command *tee* as shown above. Such a log file will record the *CIAO* commands AE runs on your data, and is very helpful for investigating problems.

3.3 Add Point Sources to AE

Once you have the celestial coordinates of your sources in *IDL* vectors (e.g. *RA* and *DEC*), add those sources to your AE workspace. We use an AE tool for this.

```

; Load all the AE code.
.run ae
.run

```

```

; Add sources.
ae_source_manager, /ADD, RA=RA, DEC=DEC, POSITION_TYPE='wavdetect'

; Sort the source catalog by RA; label sources by their sequence number.
ae_source_manager, /SORT_RA
ae_source_manager, /SET_LABEL_AS_SEQUENCE
end

```

The keyword `POSITION_TYPE` (required) should be a string (or string vector) describing the method you have used to estimate the source position. In the example above, source positions came from the *wavdetect* detection tool.

The *ae_source_manager* tool creates a directory for each source, and maintains a list of the sources in the ASCII file *all.srclist*. This tool can be used later to add, re-position, or remove sources as needed (§7.2).

3.4 Basic Extraction

With the various data products required by *CIAO* located in an observation directory as described above, and with sources created and listed in the source list *all.srclist*, a basic extraction of the sources from one observation can be accomplished with the following *IDL* commands (executed from the *point_sources* run directory):

```

.run
; Create extraction regions for each ObsID.
ae_make_catalog      , '1875', EVTFILE='spectral.evt', /SHOW
ae_make_catalog      , '1876', EVTFILE='spectral.evt', /SHOW

; Extract spectra and backgrounds.
ae_standard_extraction, '1875', EVTFILE='spectral.evt'
ae_standard_extraction, '1876', EVTFILE='spectral.evt'
end

; Repeat the three commands below until ae_adjust_backscal_range
; reports 'No adjustments are required.'
.run
ae_adjust_backscal_range, MIN_NUM_CTS=100

ae_standard_extraction, '1875', EVTFILE='spectral.evt', SRCLIST_FILENAME='rerun.srclist', $
                        EXTRACT_SPECTRA=0, TIMING=0, /REUSE_MASKING
ae_standard_extraction, '1876', EVTFILE='spectral.evt', SRCLIST_FILENAME='rerun.srclist', $
                        EXTRACT_SPECTRA=0, TIMING=0, /REUSE_MASKING
end

; Combine the extractions from both ObsIDs; perform photometry.
; ALWAYS RUN THIS COMMAND, even if you have only one obsid!
acis_extract, 'all.srclist', /MERGE_OBSERVATIONS

; Build a FITS table and region file summarizing the extractions.
acis_extract, 'all.srclist', COLLATED_FILENAME='all.fits', REGION_FILE='all.reg', $
                        LABEL_FILENAME='label.txt'

```

- The *ae_make_catalog* tool (§7.5) builds non-overlapping extraction regions for the sources. Those regions are nominally sized to enclose 90% of the local PSF (§5.9), but are iteratively reduced to avoid overlapping extractions in crowded areas.
- The tool *ae_standard_extraction* runs the `EXTRACT_SPECTRA` (§E.3), `EXTRACT_BACKGROUND`s (§E.5), and `TIMING` (§E.4, Figure 18) stages of AE.

- The cycle of calling the *ae_adjust_backscal_range* tool and re-extracting backgrounds with *ae_standard_extraction* adjusts the size of the background regions, as described in §7.7.
- The MERGE.OBSERVATIONS Stage is described in §7.8.
- In each source directory, you will find many data products (discussed in other sections and in Appendix D), including source and background spectra, RMF and ARF files, ready for spectral fitting.

3.5 Review Your Catalog and Extractions

We recommend that you visually review the extraction regions (polygons) which are shown in a *DS9* session spawned by *ae_make_catalog*. We also recommend that you visually review the background event list and exposure map (formed by masking the point sources), which are shown in a *DS9* session spawned by *ae_standard_extraction*.

Potentially valuable diagnostic plots are available from all these AE stages, as described in §7. For simplicity in this Getting Started tutorial, we omit generation of those plots.

The AE runs above will produce lots of output to the screen (archived in *AE_simple_run.log* in this example). The important calls to *CIAO* tools are shown. AE produces many messages about what it's doing, and the *CIAO* tools that AE calls sometimes produce messages. Reading through this log once can be educational. However, for routine AE runs we recommend that you use *egrep* to scan for lines in the log that are not routinely there. An shell command to get you started is shown below. The first *egrep* selects lines with the word “warning”, “error”, or “halted”. Since many routine, uninteresting messages contain those words, the second *egrep* tries to remove those uninteresting lines.

```
egrep -i 'WARNING|ERROR|halted' AE_simple_run.log | egrep -v 'DISPLAY variable|no in-band data|No HDUN
```

3.6 Spectral Fitting

Interesting sources with sufficient counts can be automatically fit with *XSPEC*, using *XSPEC* scripts supplied with AE (§5.12, §§7.11). Obviously, spectral fitting can be a complex undertaking, but the example call below shows how one would fit a set of point sources (listed in the file *xspec.srclist*) to a one-temperature thermal model with Solar abundances in both absorbing column and emitting plasma.

```
idl |& tee -a AE_fitting.log
  acis_extract, 'xspec.srclist', /FIT_SPECTRA, /CSTAT, CHANNEL_RANGE=[35,548], $
    MODEL_FILENAME='xspec_scripts/thermal/T:Solar,Solar.xcm'
```

In this example, the spectrum is not grouped, and the C-statistic is minimized in the fit. The energy band used in the fit is the PI channel range [35–548] (0.5–8 keV).

3.7 Collating the Results

All the source properties computed by AE plus the spectral fitting results³⁵ can be collated (§7.12) into a FITS table (with ~100 columns) via a command like:

```
acis_extract, 'all.srclist', COLLATED_FILENAME='all.collated', REGION_FILE='all.reg'
```

The resulting FITS binary table, *all.collated*, can of course be examined and manipulated with any of the usual FITS tools in *CIAO*, *HEASOFT*, *IDL*, etc.

The observer can derive from this FITS table whatever quantities are suitable for publication in L^AT_EX or machine-readable tables. For example in our own star formation studies we produce (§7.14) three machine-generated tables presenting basic source properties, thermal plasma spectral fits, and power law spectral fits; examples are shown in Figures 1, 2, and 3.

³⁵ Multiple fitting sessions can be applied to each source, e.g. thermal and power law spectral models. An interactive tool (*ae_spectra_viewer*) is available to help the observer designate the preferred fit for each source (§7.14).

Table 1. Primary Chandra Catalog: Basic Source Properties

Seq #	Source		Po sition			Extra cted Co unts					S ignif			Ch aracteri stics			E _{Exp} (ke V)
	C X O U J	(2)	γ _{zoom} (deg)	γ _{zoom} (deg)	Er r (%)	γ _{Net} (°)	Full	Net	B kgd	Hard	P S F	Log P a	An om	Var	ks		
1	181957.	56°-612004.	5	274.	989852	0.4	8.0	100.5	10.8	4.5	42.6	0.87	20.5	1.8	
2	181957.	57°-61328.	8	274.	990295	1.2	8.4	12.5	4.8	5.5	12.3	0.89	9-	9-	24.3	4.5	
3	181958.	77°-61440.	4	274.	994909	0.7	8.7	11.0	5.1	0.0	7.6	0.90	28.9	3.8	
4	181958.	94°-60924.	4	274.	995611	0.7	7.6	17.4	5.3	5.6	10.1	0.89	...	a	30.7	2.4	
5	181959.	04°-60350.	2	274.	996016	0.6	10.1	55.4	9.2	17.6	31.8	0.90	...	a	26.4	2.4	
6	182000.	05°-61742.	0	275.	000244	0.6	16.2	241.7	16.2	5.3	103.7	0.91	...	c	30.5	1.7	
7	182000.	05°-61112.	0	275.	002736	0.8	16.6	28.3	6.3	4.7	14.4	0.89	...	a	31.1	2.0	
8	182001.	62°-61600.	3	275.	006760	0.6	8.8	16.4	5.6	8.6	12.6	0.90	...	c	28.8	2.6	
9	182001.	73°-60529.	0	275.	0091417	0.7	8.5	24.5	6.0	5.5	0.0	0.90	...	b	28.9	1.0	
10	182002.	34°-60348.	0	275.	007272	0.6	9.5	61.7	9.1	10.3	7.9	0.90	...	a	28.0	1.2	
11	182003.	03°-60206.	7	275.	012661	0.6	10.1	121.4	12.2	16.6	18.1	0.91	...	a	26.1	1.4	
12	182003.	28°-61128.	9	275.	017840	0.6	6.3	19.8	5.2	2.2	8.6	0.90	...	a	32.3	1.7	
13	182004.	35°-61452.	6	275.	018964	1.1	7.6	7.2	3.9	3.8	7.8	0.90	...	a	30.6	5.6	
14	182004.	35°-60807.	1	275.	019140	0.4	6.6	29.9	6.3	3.1	15.1	0.90	...	a	30.5	2.0	

Note. — Table 1 is published in its entirety in the electronic edition of the *Astronomical Journal*. A portion is shown here for guidance regarding its form and content.

[illegible]

Table 3. X-ray Spectroscopy for Photometrically Selected Sources: Thermal Plasma Fits

Seq #	CXOU J (2)	Source ^a		Net Counts (3)	Signif ^b (4)	Spectral Fit ^b		log EM (cm ⁻³) (7)	log L _s			X-ray Luminosity ^c			Notes ^d (13)
						log N _H (cm ⁻²) (5)	kT (keV) (6)		(8)	(9)	(10)	log L _t	(11)	(12)	
1	181957.56–161204.5			100.5	8.9	-0.4 21.6 +0.2	-1.9 4.5	-0.2 54.3 +0.2	30.69	31.14	31.16	31.27	31.41
4	181958.94–160924.4			17.4	3.0	-0.4 22.4 +0.3	2.0	53.9 +0.4	29.46	30.29	30.45	30.35	30.89
6	182000.05–161242.0			241.7	14.4	-0.3 21.4 +0.2	-2.8 6.1 +7.1	54.4	30.88	31.30	31.31	31.44	31.53
7	182000.65–161122.0			28.3	4.1	-0.6 22.0 +0.2	-2.0 2.9	53.7	29.88	30.40	30.45	30.52	30.78
8	182001.62–161600.3			16.4	2.7	22.4	0.7	54.2 +0.3	29.50	29.63	29.89	29.87	31.20
9	182001.73–160529.0			24.5	3.7	20.0	-0.2 0.7 +0.2	-0.2 53.2 +0.2	30.17	28.91	28.91	30.19	30.20
11	182002.34–160348.0			61.7	6.4	20.0	-0.4 1.4 +0.5	-0.12 53.7 +0.09	30.51	30.00	30.00	30.63	30.64
12	182003.03–160206.7			121.4	9.4	-0.11 22.1 +0.03	0.5	-0.33 55.2 +0.07	30.77	30.10	30.23	30.86	32.04
15	182004.56–160807.1			29.9	4.4	-0.3 22.3 +0.2	-0.9 1.5 +1.5	-0.4 54.2 +0.8	29.90	30.41	30.56	30.53	31.15
16	182004.72–160937.4			19.4	3.4	22.6	-0.9 1.3 +3.8	54.4	...	30.33	30.61	30.39
18	182005.11–160756.9			64.2	6.9	-0.3 22.3 +0.2	-2.0 4.3	-0.2 54.3 +0.3	30.06	31.05	31.15	31.10	31.40
19	182005.17–161415.3			10.6	2.2	-0.3 22.9 +0.3	2.0	-0.4 54.2 +0.4	...	30.33	30.73	30.33
20	182005.43–161611.8			26.8	4.0	21.8 +0.2	0.5 +0.1	54.1 +0.2	30.12	29.13	29.21	30.16	31.00
24	182007.34–160718.9			16.7	3.1	20.0	-0.2 0.8 +0.3	-0.2 53.1 +0.2	30.00	28.80	28.80	30.03	30.04
25	182007.53–160751.9			134.8	10.6	-0.15 21.8 +0.08	-0.2 0.6 +0.1	-0.2 54.5	30.79	29.97	30.04	30.85	31.48

Note. — Table 3 is published in its entirety in the electronic edition of the Astrophysical Journal. A portion is shown here for guidance regarding its form and content.

^aFor convenience, columns 1–4 reproduce the source identification, net counts, and photometric significance data from Table 1.

^bAll fits used the “wabs(pec)” model in XSPEC and assumed 0.3Z abundances (Mitsuda et al. 2001; Feigelson et al. 2002). Columns 5 and 6 present the best-fit values for the column density and plasma temperature parameters. Column 7 presents the emission measure for the model spectrum, assuming a distance of 1.6 kpc. Quantities in italics were frozen in the fit. Uncertainties represent 90% confidence intervals. More significant digits are used for uncertain ties < 0.1 in order to avoid large rounding errors; for consistency, the same number of significant digits is used for both lower and upper uncertain ties. Uncertain ties are missing when XSPEC was unable to compute them or when their values were so large that the parameter is effectively unconstrained. Fits lacking uncertain ties, fits with large uncertain ties, and fits with frozen parameters should be viewed merely as splines to the data to obtain rough estimates of luminosities; the listed parameter values are unreliable.

^cX-ray luminosity is presented in columns 8–12: s = soft band (0.5–2 keV); h = hard band (2–8 keV); t = total band (0.5–8 keV). Absorption-corrected luminosity is presented in column 13: they are omitted when log N_H > 22.5 cm⁻² since the soft band emission is essentially unmeasurable.

^d2T means a two-temperature model was used; the second temperature is shown in parentheses. See spectra in Figure 12.

Figure 2: Example of machine-generated spectroscopy table

Table 4. X-ray Spectroscopy for Photometrically Selected Sources: Power Law Fits

Seq #	Source ^a		Net C counts (3)	Signif (4)	Spectral Fit ^b			X-ray Fluxes ^c				
	CXO U J (2)				log N _H (cm ⁻²) (5)	?	log N _T (7)	log L _s (8)	log L _h photons cm ⁻² s ⁻¹ (9)	log L _{h,c} photons cm ⁻² s ⁻¹ (10)	log L _t (11)	log L _{t,c} (12)
2	181957.67	-161328.8	12.5	2.3	23.1	1.1	-6.3	...	29.82	30.10	29.82	...
5	181959.04	-160350.2	55.4	5.7	21.9	1.6	-5.2	30.10	30.88	30.91	30.95	31.10
13	182004.28	-161128.9	19.8	3.4	21.6	-1.3 2.1	-5.6	29.78	30.12	30.14	30.28	30.46
40	182010.53	-161401.5	24.1	3.9	-1.0 21.9 +0.4	-1.0 1.5	-5.6	29.70	30.49	30.52	30.56	30.71
59	182013.22	-160856.5	10.3	2.3	22.2	0.1	-6.4	28.95	30.64	30.67	30.65	30.70
69	182015.48	-160706.0	10.1	2.3	22.8	5.4	-3.6	...	29.97	30.54	30.02	...
96	182017.83	-161014.5	12.8	2.7	-1.4 22.8 +0.5	1.2	-5.3	...	30.90	31.08	30.90	...
102	182018.19	-160937.0	56.5	6.6	-0.3 22.5 +0.2	-1.0 1.5	-4.9	...	31.07	31.18	31.08	...
112	182018.95	-160736.9	17.3	3.2	-0.7 22.3 +0.4	1.9	-5.4	29.33	30.41	30.50	30.45	30.76
121	182019.33	-161051.9	9.5	2.2	23.0	3.1	-4.4	...	30.32	30.78	30.33	...
153	182020.49	-160806.4	15.0	2.9	22.0 +0.5	1.6	-5.7	29.45	30.30	30.34	30.35	30.54
175	182021.08	-160738.2	20.2	3.6	22.0 +0.5	1.3	-5.8	29.47	30.50	30.54	30.54	30.67
179	182021.25	-160944.3	8.4	2.0	21.5 +0.8	0.6	-6.6	29.11	30.18	30.19	30.21	30.24
189	182021.59	-160703.3	11.3	2.5	22.5	1.6	-5.6	...	30.30	30.43	30.32	...
196	182021.74	-161317.4	10.4	2.3	22.0 +0.5	1.5	-6.0	29.24	30.16	30.20	30.21	30.37

Note. — Table 4 is published in its entirety in the electronic edition of the Astrophysical Journal. A portion is shown here for guidance regarding its form and content.

^aFor convenience, columns 1–4 reproduce the source identification, net counts, and photometric significance data from Table 1.

^bAll fits used the “wabs(overl) model in XSPEC. Column 5 and 6 present the best-fit values for the column density and power law photon index parameters. Column 7 presents the power law normalization for the model spectrum. Quantities in italics were frozen in the fit. Uncertain ties represent 90% confidence intervals. Uncertain ties are missing when XSPEC was unable to compute them or when their values were so large that the parameter is effectively unconstrained. Fits lacking uncertain ties, fits with large uncertain ties, and fits with frozen parameters should be viewed merely as splines to the data to obtain rough estimates of luminosities; the listed parameter values are unreliable.

^cX-ray luminosities are presented in columns 8–12: s = soft band (0.5–2 keV); h = hard band (2–8 keV); t = total band (0.5–8 keV). A absorption-corrected luminosity is subscripted with a c; they are omitted when log N_H > 22.5 cm⁻² since the soft band emission is essentially unmeasurable.

Figure 3: Example of machine-generated spectroscopy table

4 Citation

We request that observers make an appropriate citation in work that relies on AE. The primary AE citation is to our 2010 paper on ACIS data analysis. Citations to our entry in the Astrophysics Source Code Library³⁶ and the AE code archive in the Zenodo repository³⁷ would also be appreciated. If convenient, a mention of AE’s URL may be helpful to the reader. The L^AT_EX snippet below demonstrates all these citations.

```
Events were extracted using the
\anchor{http://personal.psu.edu/psb6/TARA/ae_users_guide.html}{\em ACIS Extract}}
(AE) software package
\footnote{
  The {\em ACIS Extract} software package and User's Guide are available at
  \url{http://personal.psu.edu/psb6/TARA/ae_users_guide.html}.
}
\citep{Broos2010,AE2012,AE2016}.
```

The references Broos2010³⁸ and AE2012³⁹ and AE2016⁴⁰ cited above are provided below, in a format compatible with Patrick Daly’s *natbib* package.

```
\bibitem[Broos et al.(2010)]{Broos2010} Broos, P.~S., Townsley, L.~K.,
Feigelson, E.~D., Getman, K.~V., Bauer, F.~E., \& Garmire, G.~P.\ 2010, \apj, 714, 1582
```

```
\bibitem[Broos et al.(2012)]{AE2012} Broos, P.~S., Townsley, L.~K., Getman, K.~V.,
Bauer, F.~E.\ 2012, Astrophysics Source Code Library, 1203.001
```

```
\bibitem[Broos \& Townsley(2016)]{AE2016} Broos, P.~S., \& Townsley, L.~K.\ 2016, ACIS Extract software pac
```

You should also, of course, make appropriate citations to the *CIAO*⁴¹ (Fruscione et al., 2006) and *MARX*⁴² (Davis et al., 2012) software, as requested by their institutions.

³⁶<http://ascl.net/1203.001>

³⁷<https://doi.org/10.5281/zenodo.781433>

³⁸<http://adsabs.harvard.edu/abs/2010ApJ...714.1582B>

³⁹<http://ascl.net/1203.001>

⁴⁰<https://doi.org/10.5281/zenodo.781433>

⁴¹<http://cxc.harvard.edu/ciao/>

⁴²<http://space.mit.edu/ASC/marx/index.html>

5 Algorithms

5.1 Point Spread Function Images

The point spread function (PSF) of the *Chandra*-ACIS system is obviously a fundamental calibration product useful in many ways for the analysis of point sources. AE makes use of the PSF in several ways, described elsewhere. Since the *Chandra*-ACIS PSF varies strongly with position on the detector and with energy, AE represents the PSF for each observation of each source individually by storing images (file *source.psf*) of the monochromatic PSF for several energy values that span the ACIS response.

5.1.1 HRMA Model

Several models of the HRMA component of the system PSF are available, including *ChaRT*, *SAOTrace*, and *MARX*. Routine use of *ChaRT* is infeasible in AE because it lacks a machine interface. As of January 2008, *SAOTrace* is in beta release and has limited operating system coverage.

AE currently⁴³ models the HRMA using the *MARX* ray-trace simulator. Simulations are run for each observation of each source at several monochromatic energies. *MARX* dithers the simulated source using the observation's aspect file, allowing accurate modeling of distortions caused by the PSF dithering over the boundaries of the ACIS CCDs. (*MARX* does not however model the bad pixel table.)

Note that application of the CXC thread “Improving the Astrometry of your Data: Correct for a Known Processing Offset” will induce an offset between your data and the *MARX* PSF. We do not understand the header magic involved here, but fortunately this thread is not applicable to data re-processed after 2004.

5.1.2 Post-HRMA Blurring

The PSF of a real source observed by ACIS differs strongly (on-axis) from the HRMA-alone PSF due to three blurring effects:

- ACIS pixels are large compared to the HRMA PSF, leading to significant quantization effects. These quantization effects depend on the event position estimator used by the *CIAO* tool *acis_process_events* tool (controlled by the *pix_adj* parameter).
- The reported positions of ACIS events are reconstructed from an aspect solution that measures the dither motion; this reconstruction has small errors.
- The *acis_process_events* tool can be configured to add uniform random noise to the ACIS event positions⁴⁴.

These post-HRMA blurring effects are very significant on-axis, as discussed in the [Proposers' Observatory Guide \(2011, §4.2.1, 4.4\)](#) and in presentations at the Chandra Calibration Workshop by Jerius⁴⁵, Marshall⁴⁶, Pease (page

⁴³ Prior to version 3.167, AE modeled the HRMA using the *mkpsf* tool, which has a number of problems:

- The CXC has always warned users that the accuracy of *mkpsf* results is compromised by the coarse spatial sampling of the PSF Library.
- We have found (and reported) that the PSF Library contains errors which produce artifacts in some off-axis PSF images.
- When processing observations at the I-aimpoint, PSFs cannot be reliably constructed for sources on the S-array. The following warning message is produced by *mkpsf*:

“The requested X,Y coords were outside the PSF library file range. The closest coordinates in the PSF Library file were used instead.”

This problem was reported to the CXC Helpdesk as ticket #5346. The root cause is thought to be that PSFs on some regions of S2 and S3 are missing from the CALDB. If *mkpsf* results are used it appears that such sources would be extracted using incorrect apertures and should be treated accordingly.

- When processing observations at the S-aimpoint, sources on I2 or I3 cannot be processed by AE because the CALDB PSF library does not contain the appropriate PSFs.

⁴⁴<http://cxc.harvard.edu/ciao/why/acispixrand.html>

⁴⁵http://cxc.harvard.edu/ccw/proceedings/03_proc/presentations/jerius/s025.html

⁴⁶http://cxc.harvard.edu/ccr/proceedings/05_proc/presentations/marshall3

8)⁴⁷, and Smith (page 3)⁴⁸.

As of September 2013, AE relies on *MARX* (version 5.0 or greater) to model these post-HRMA blurring effects. Observers are encouraged to use the AE tool *ae_radial_profile* (§7.14) to compare PSFs to their own observations.

5.1.3 PSF Cropping and Aperture Correction

The original and primary use of multi-energy PSFs in AE is to correct the mission’s spectral response calibration for the effects of the finite aperture used to extract the source. As of January 2008 neither the *CIAO* tools nor threads address the aperture correction issue.

As far as I can tell, both the HRMA and ACIS effective area calibrations implicitly assume an infinitely large detector and extraction aperture⁴⁹. I believe the HRMA simulations used to derive QE values count all rays exiting the HRMA, regardless of how far out in the wings of the source they fall. The ACIS QE was derived for flat illumination where there is no concept of an aperture.

AE attempts to imperfectly correct the ARF (§5.9) to account for the source events presumed to fall outside the extraction aperture, by measuring the fraction enclosed by that aperture on each of the mono-chromatic PSFs available. When the PSFs are constructed, AE tries to estimate how much total power the PSF contains, integrated to infinity⁵⁰, and then records how much of that power has been cropped away by the finite size of the PSF images themselves. *MARX* simulations used to generate PSFs are run with the ACIS readout streak disabled, since that “out of time” effect is handled by the CXC via a reduction in the source’s exposure time (via FITS keyword DTCOR).

One imperfection of our aperture correction arises from the fact that PSFs generated via *MARX* contain distortions caused by the dithered edges of the ACIS CCDs. For some uses of the PSF this is a good thing—*MARX* very nicely simulates the data one would observe on the real ACIS. For aperture correction however, the proper PSF to use would probably be one built assuming an infinite detector, since that is the assumption used in the HRMA and ACIS calibrations. We do not however have such a PSF available in AE. By using a PSF which exhibits chip edges, we should be under-estimating the PSF fraction of sources dithering off the detector, thereby over-correcting the ARF. The energy dependence of this over-correction has not been studied. The moral of this story is that calibration is difficult where ever the exposure map is varying over the scale of the PSF, and one should take a skeptical view of all properties derived for sources there.

The energy-dependence of the aperture correction—the correction to the shape of the ARF—can be significant. For example, a source far off-axis ($\theta = 8.7$ arcmin) extracted with a default aperture (~ 15 sky pixel radius) should have these corrections applied to its ARF:

Energy (keV)	PSF Fraction
0.277	0.91
1.496	0.90
4.510	0.85
6.400	0.77
8.600	0.63

A source nearly on-axis ($\theta = 0.4$ arcmin) extracted with a default aperture (~ 1.6 sky pixel radius) should have these corrections applied to its ARF:

⁴⁷http://cxc.harvard.edu/ccr/proceedings/05_proc/presentations/pease

⁴⁸http://cxc.harvard.edu/ccr/proceedings/05_proc/presentations/smith/pg03.html

⁴⁹ For example, the *ahelp* page for *mkarf* says “The ARF is computed assuming that the spectral extraction region is large enough to include the entire PSF (e.g. PSF fraction=1.0).”.

⁵⁰ When *MARX* is used the total PSF power can be estimated only for the portion of the sky falling on ACIS since rays missing ACIS are discarded. In earlier versions of AE, when *mkpsf* was used, estimating the total PSF power was very difficult since that tool does not provide cropping information; AE did attempt however to estimate cropping in the f1 and f3 PSF libraries using the f2 library which uses larger PSF footprints; see *What’s New* for February, 2004 (AE Version 3.19).

Energy (keV)	PSF Fraction
0.277	0.96
1.496	0.90
4.510	0.80
6.400	0.81
8.600	0.80

5.2 Extraction and Mask Regions

In the ideal case of an isolated point source with zero background, selection of an extraction region is simple—the larger the region the better. In real observations however, where sources just above the background are inevitably studied, selection of an extraction region is of course a trade off between maximizing the source’s signal and minimizing the background’s signal. Additionally, one may need to avoid overlapping PSFs, treat extended sources, or deal with complex background.

AE constructs polygon extraction regions which approximate contours of the *Chandra*-ACIS point spread function (PSF). A PSF contour is selected such that the fraction of the PSF enclosed approximates a target *PSF fraction* value specified by the observer. In a multi-observation reduction, an extraction region is constructed independently for each observation.

A circular *mask region* is also defined for each source in order to support the construction of a background data set (see §5.5) for each observation. The radius of the mask region is nominally chosen to be 1.1 times a radius that encloses 99% of the PSF.

5.3 Source Positions

Obviously, the extraction region will be most effective in maximizing a source’s signal and minimizing the background if it is positioned accurately on the source. AE can compute three types of source position estimates:

- The most straightforward position estimate provided is the mean position of events in a specified energy band that are found in the source’s extraction regions (§7.8). Obviously, this estimator will be biased if the extraction region(s) are offset from the true position; thus, iterative repositioning of the source is recommended. This estimator should also be biased if the local background is sloped (e.g. if the source sits in the wing of a bright neighbor). This estimator should also be biased if the PSF is asymmetric.

A standard error on the mean data position estimate is computed for each axis using the variances of the PSF and flat background within the extraction region of each observation. The resulting 2-D error ellipse is then approximated by the so-called “1 deviation root mean square” (dRMS or 1DRMS) error radius, $dRMS = \sqrt{\sigma_X^2 + \sigma_Y^2}$.⁵¹ The integral of the error ellipse inside this dRMS radius (i.e. the significance of this circular confidence region) varies from 63% for equal errors in X and Y (a circular ellipse) to 68% for a highly eccentric ellipse.

- When the PSF is asymmetric the mean data position estimate should be biased. Thus, AE provides a second position estimate by correlating the neighborhood around the source (not just the extracted counts) with the source’s PSF (§7.9). In a multi-observation reduction, a multi-ObsId data image and multi-ObsId PSF are constructed and correlated. This estimator should be biased if the local background is sloped (e.g. if the source sits in the wing of a bright neighbor). Currently no uncertainty estimate is available for the correlation position.

⁵¹

I don’t know a good astronomical reference for this practice, but see:

- Section 2.2 of “A Standardized Algorithm for the Determination of Position Errors by the Example of GPS with and without ‘Selective Availability’”, http://www.mar-it.de/NavGen/final_text3.pdf, Ingo Harre, published in “The International Hydrographic Review”, Vol. 2, No. 1 (New Series), June 2001,
- Section 4-10 of “Engineering and Design - Hydrographic Surveying”, engineering manual, US Army Corps of Engineers, http://www.publications.usace.army.mil/Portals/76/Publications/EngineerManuals/EM_1110-2-1003.pdf
- Section 4.5 of “Engineering and Design: Navstar Global Positioning System Surveying”, engineering manual, US Army Corps of Engineers, http://www.publications.usace.army.mil/Portals/76/Publications/EngineerManuals/EM_1110-1-1003.pdf
- Section 2.4.3 of “Principles and Practice of GPS Surveying”, Chris Rizos, University of New South Wales, School of Geomatic Engineering.

- Both the mean data and PSF correlation estimators above implicitly assume the observed data are explained by a **single** source, and then seek to estimate the best position for that source. In a crowded field, the wings of bright neighboring sources will tend to bias these estimators. For such situations, AE provides a third position estimate by performing image reconstruction on the neighboring field, and then estimating the position of any nearby peak in the reconstructed image (§7.9). Currently no uncertainty estimate is available for the reconstruction position.

The responsibility for choosing which position estimate should be adopted for each source is left with the observer.

5.4 Source Spectra, Single Observation

HEASARC/OGIP-compatible Type I source spectra⁵², over the PI channel range 0-1024 (0-15keV), are constructed using the *CIAO* tool *dmextract*. The spectra are NOT band-limited by AE; the observer is encouraged to delay energy filtering until the spectrum grouping and fitting process.

The BACKSCAL keyword (commonly used in fitting packages for background scaling) in the single-ObsId source spectrum is computed by estimating the integral of the exposure map (supplied by the observer) over the extraction region. This deviation from the common practice of using the simple geometric area of the extraction region for BACKSCAL was chosen because it is felt this method provides more accurate background scaling in cases where the exposure varies significantly over the extraction region. Such exposure variation would come from CCD edges or masked regions dithering into the extraction region.

The EXPOSURE, ONTIME, and LIVETIME keywords in the source spectrum are set equal to the EXPOSUR_n, ONTIMEn, and LIVTIMEn keywords, where CCD *n* contains the most extracted events. In some observations where the CCDs have significantly different time filters applied, this leads to more accurate absolute flux values than would be produced using the default EXPOSURE value which corresponds to the aim-point CCD. This potential inaccuracy is described on the CXC web site⁵³.

5.5 Background Spectra, Single Observation

When source crowding is not a concern, local background spectra are constructed by removing the point sources from the data (§7.6.2, 7.6.3) and then searching around each source for the smallest circular region that encompasses at least a minimum number of background counts specified by the observer. As in the source spectrum, the BACKSCAL keyword in the single-observation background spectrum is computed by estimating the integral of the exposure map over the background extraction region.

When crowded sources significantly contaminate each other’s extraction regions, more accurate local background spectra may be constructed from unmasked data by modeling the spatial distribution of the point source flux and then carefully constructing background regions which seek to subtract background components arising from neighboring sources (§7.6.1).

When an ObsID includes both front-illuminated (FI) and back-illuminated (BI) detectors (e.g. S3 and S4) background spectra can be corrupted. The issue is that AE defines background regions on the sky, with no regard to detector boundaries. Thus, a source lying near the S3/S4 boundary could end up with a bkg region that spans the two detectors (which have very different bkg spectra). The proper way to extract ObsIDs that mix FI and BI data is to simply split the ObsID into two parts, e.g. obs1234F holding the FI detectors and obs1234B holding the BI detectors. Each must have its own exposure map, showing only the appropriate detectors. Then each “virtual ObsID” should be extracted normally.

When a source has been observed on **both** FI and BI devices, AE’s photometric quantities should be accurate, since background are estimated separately for each ObsID. If spectral fitting is desired, the observer must decide whether to allow AE to combine all the spectra (the normal workflow) or whether to fit the two types of spectra separately (perhaps simultaneously) outside the context of AE.

5.6 Multi-ObsId Spectra

Most *Chandra* sources have been observed multiple times, either by design or as a result of scheduling constraints. Suppose a specific source was extracted from each of *N* observations, and AE has constructed the following quantities:

⁵²<http://heasarc.gsfc.nasa.gov/docs/xanadu/xspec/fits/files.html>

⁵³<http://cxc.harvard.edu/ciao/ahelp/times.html>

- N source spectra, $\{\mathbf{counts}_i^{\text{src}}, i = 1..N\}$, each with a measure of the source aperture “area”, $BACKSCAL_i^{\text{src}}$ (§5.4)
- N background spectra, $\{\mathbf{counts}_i^{\text{bkg}}, i = 1..N\}$, each with a measure of the background region “area”, $BACKSCAL_i^{\text{bkg}}$ (§5.5)

(In this section variables that are functions of energy are shown in **bold face**. Such quantities are stored as columns in FITS files, rather than as keywords.)

Certainly, the most statistically rigorous approach to analyzing these extractions—performing photometry and modeling the spectrum of the source—would be to use a procedure that works with these $2 \times N$ spectra directly. For example, if each of the extractions had plenty of counts then one could subtract the backgrounds to form net spectra with error bars, and then simultaneously fit those N spectra in *XSPEC*/*Sherpa* using the χ^2 statistic. However, extractions with large numbers of counts are very rare among *Chandra* sources. Thus, if you’re a fan of χ^2 fitting then in many cases the single-ObsId extractions will have to be combined in some way.

The C-statistic offers an alternative spectral modeling approach that could, in principle, work with the $2 \times N$ spectra directly. We have not had the courage to implement the data structures and fitting scripts that would be required to do this. Even if this approach was implemented, there would be no obvious way to visually compare the data and model. Imagine a source in a *Chandra* deep field that has more than 20 extractions, each with 0, 1, maybe 2 counts, plus 20 background spectra; how could one “see” that the derived model “fits” the data?

Thus, AE has always taken the less rigorous approach of merging the extractions of a source to form a “multi-ObsId” source spectrum, background spectrum, ARF, and RMF.

5.6.1 Source Spectrum

The integer-valued *multi-ObsId source spectrum* is formed by summing all the events falling in all the source extraction regions:

$$\mathbf{counts}_{\text{multi-ObsId}}^{\text{src}} = \mathbf{counts}_0^{\text{src}} + \mathbf{counts}_1^{\text{src}} + \dots \quad (1)$$

The HEASARC/OGIP FITS keyword *EXPOSURE* in the multi-ObsId spectrum is set to the sum of the *EXPOSURE* keywords in the N extractions.

5.6.2 Background Spectrum

Construction of a multi-ObsId background spectrum, and its scaling, is a more complex task that requires some discussion. We must somehow combine and scale N background spectra into a *multi-ObsId background spectrum* that estimates the background that is polluting the multi-ObsId source spectrum.

When spectral analysis is performed on background-subtracted (net) spectra using χ^2 , then eventually, in someone’s software, an estimate must be made for the number of background counts (in each spectral bin) expected to pollute the multi-ObsId source spectrum. The obvious method is to estimate separately the background polluting each source aperture (by scaling each background spectrum by its exposure ratio), and then sum those components:

$$\begin{aligned} \mathbf{scaledCounts}_{\text{multi-ObsId}}^{\text{bkg}} = & \mathbf{counts}_0^{\text{bkg}}(BACKSCAL_0^{\text{src}}/BACKSCAL_0^{\text{bkg}}) + \\ & \mathbf{counts}_1^{\text{bkg}}(BACKSCAL_1^{\text{src}}/BACKSCAL_1^{\text{bkg}}) + \dots \end{aligned} \quad (2)$$

However, χ^2 methods also require an estimate of the **uncertainty** on the background in order to assign an uncertainty to the net counts observed in each spectral bin; this is where combining extractions gets difficult. One can imagine computing an uncertainty on $\mathbf{scaledCounts}_{\text{multi-ObsId}}^{\text{bkg}}$ by estimating the uncertainty on each term in eq. 2 and then propagating those errors. The multi-ObsId spectrum would then be stored as a real-valued *RATE* column in a HEASARC/OGIP FITS file, and the errors would be stored as a *STAT_ERR* column. However, reasonably-sized ACIS background regions will produce background spectra with mostly zero or one count per channel. Assigning errors to those low-count channels, and then propagating those errors would be very foolish. Such bogus non-Gaussian error estimates would then be additionally propagated within *XSPEC*/*Sherpa* when the source spectrum is grouped.

An attractive alternative approach is to simply sum all the background spectra to form one multi-ObsId background spectrum:

$$\mathbf{counts}_{\text{multi-ObsId}}^{\text{bkg}} = \mathbf{counts}_0^{\text{bkg}} + \mathbf{counts}_1^{\text{bkg}} + \dots \quad (3)$$

just as we did with the source spectra (eq. 1). This multi-ObsId spectrum is stored as an integer-valued *COUNTS* column in a HEASARC/OGIP FITS file, and *XSPEC/Sherpa* are responsible for assigning uncertainties to each spectral bin (after grouping).

When the C-statistic is used for spectral modeling, the background spectrum **must** be represented as a set of detected counts, rather than a set of count rate estimates. Thus, for both fit statistics it appears that we must form the multi-ObsId background spectrum via eq. 3.

5.6.3 Background Scaling

An appropriate scaling must be assigned to the multi-ObsId background COUNTS spectrum (eq. 3) so that the background subtraction (χ^2 approach) or background modeling (C-stat approach) performed in the spectral analysis package corresponds to the background rate shown in eq. 2. Prior to February, 2009 AE calculated such a scaling independently on each channel of the background spectrum, resulting in an energy-dependent scaling:

$$\text{AREASCAL}_{\text{bkg}} = \frac{\text{counts}_{\text{multi-ObsId}}^{\text{bkg}}}{\text{scaledCounts}_{\text{multi-ObsId}}^{\text{bkg}}} \quad (4)$$

(The background scaling is represented in the multi-ObsId FITS spectra under the standard name *AREASCAL*, rather than the more common name *BACKSCAL*, so that observers have the choice of modeling the background spectrum rather than subtracting it.)

This energy-dependent scaling was merely an algebraic artifact to deal with the fact that the extractions being merged could have wildly different background scaling. For example, if the observer designed background regions to enclose a certain number of counts, then an off-axis background might be scaled by a factor of 4 and an on-axis background might be scaled by a factor of 40. In the multi-ObsId spectrum, a channel that (by chance) had counts only from the former would have an AREASCAL value of 4 and a channel that had counts only from the later would have an AREASCAL value of 40.

In the Fall of 2008 we realized that this design can lead to poor spectral fits when C-stat is used to model the background spectrum because the background scaling seems to be applied to the background model from which C-stat is calculated. When each channel has its own background scaling (via the AREASCAL column in AE spectra prior to February, 2009), then channels with large scaling values induce large spikes in the background model; apparently corrupting the fit. Such spikes occur when a channel of the multi-ObsId background has (by chance) data only from an extraction with a scaling much larger than the other extractions.

Even if one puts aside the phenomenon of these spikes, the revelation that every element of the AREASCAL vector (FITS column) in eq. 4 directly affects the background model in *XSPEC* raises an obvious theoretical issue, namely that eq. 4 is not defined for channels with **zero** counts!

In retrospect, it seems clear that there are theoretical objections to the very notion of combining background spectra that have different scalings (BACKSCAL values). In such a multi-ObsId spectrum, some extractions are over-represented and others are under-represented. Although one can contrive a scaling of the multi-ObsId spectrum (e.g. AE's historical AREASCAL column) that produces the correct background rate in each channel, information on the uncertainty in those rates has been lost. For example, imagine a channel in which ObsId #1 produced 25 counts with a scaling of 25 and ObsId #2 produced 10,000 counts with a scaling of 10,000, producing a scaled background of $1 + 1 = 2$. The Poisson uncertainty estimated (by *XSPEC/Sherpa*) for that value of 2 would be far too small, since it is based on the notion of a single observation with 10,025 counts. In fact, the 25 count extraction totally dominates the uncertainty on the scaled background.

From the perspective of the C-statistic, uncertainty information has similarly been corrupted in the example above. C-stat sees 10,025 indistinguishable background counts, equally contributing to the value of the C-statistic. The background model will be driven to closely follow the second ObsId's 10,000-count spectrum, mostly ignoring the 25-count ObsId, even though in reality the two ObsIds contribute equal background to the multi-ObsId source.

Thus, we have concluded that the only apparent way to fairly combine background spectra is to design the background extraction process so that all extractions of a given source require similar background scalings (BACKSCAL). AE's recipes have been modified to do this, and AE's extraction merging code (§7.8) now warns if the range of background scaling among the extractions is large. AE now defines the multi-ObsId background scaling to be the single

value (independent of energy) that produces the correctly scaled background over a broad energy band:

$$AREASCAL_{bkg} = \frac{\sum_{0.5:8keV} \text{counts}_{\text{multi-ObsId}}^{\text{bkg}}(i)}{\sum_{0.5:8keV} \text{scaledCounts}_{\text{multi-ObsId}}^{\text{bkg}}(i)} \quad (5)$$

(The background scaling is represented in the multi-ObsId FITS spectra under the standard name *AREASCAL*, rather than the more common name *BACKSCAL*, so that observers have the choice of modeling the background spectrum rather than subtracting it.)

5.7 Grouping of Spectra

AE groups the multi-ObsId source spectrum using an algorithm which we believe is superior to that in the grouping tools in *FTOOLS* or *CIAO* in several ways.

- The placement of the first and last grouping bin boundaries are directly controlled by the observer through the AE keyword `CHANNEL_RANGE`. For example if `CHANNEL_RANGE=[35,548]` is specified, then the first group would cover `PI=1:34` ($\sim <0.5$ keV) and the last group would cover `PI=549:1024` ($\sim >8$ keV). By subsequently “ignoring” the first and last groups during the fitting process the observer is able to control the energy range of the fit to a resolution of 1 PI channel, rather than a resolution of 1 group (which can be large).
- Instead of defining groups so they contain at least a minimum number of counts in the source spectrum, AE defines groups so that in the background-subtracted (net) spectrum each group achieves at least a specified significance (signal to noise ratio). In situations where the background level is high, e.g. in spectra of diffuse objects, this approach can produce higher quality grouping. A standard $1\text{-}\sigma$ Gaussian error estimate on the net counts in each group is computed using an analytical approximation to upper confidence intervals of a Poissonian distribution (Gehrels, 1986, equations 7 and 12) for both the source counts and background counts in the group.

Grouping always involves a trade off between the number of groups and the significance of the groups. AE’s management of this trade off across a wide range of source counts is controlled by four parameters consisting of a range of acceptable group significance (keyword `SNR_RANGE` in §7.11) and a range on the number of groups that is acceptable (keyword `NUM_GROUPS_RANGE`). Suppose for example `SNR_RANGE=[1,5]` and `NUM_GROUPS_RANGE=[6,250]`. AE starts by constructing groups with a significance requirement of 5 (SNR_RANGE_{max}). If the resulting number of groups is between 6 and 250 then those groups are accepted.

If instead (in this example) there are less than 6 ($NUM_GROUPS_RANGE_{min}$) groups then AE iteratively lowers the significance requirement and regroupes until there are more than 6 groups. If the significance requirement reaches 1 (SNR_RANGE_{min}) then whatever groups exist are accepted.

If instead (in this example) the initial grouping produced more than 250 groups ($NUM_GROUPS_RANGE_{max}$) then AE iteratively raises the significance requirement and regroupes until there are less than 250 groups.

- The standard grouping algorithms are asymmetric. Each group is built from left to right (low energy to high energy). After the start of the group is defined, the algorithm adds channels to the right until enough counts are found to meet the specified criterion. All groups will end on a non-zero channel; when the spectrum contains a significant number of empty channels many groups will begin with a string of empty channels. This asymmetry can lead to very harmful artifacts in the grouped spectrum. For example, imagine this extreme example:
 - Groups are defined by requiring 5 counts in each group.
 - Virtually all the source counts are above 3 keV, i.e. the source is very hard.
 - Due to Poisson luck, or due to ACIS afterglows, or due to a soft background spectrum there are 5 counts within a few channels of each other down at 0.5 keV, and the algorithm allows those to form a very narrow first group.
 - The second group will then be very wide, encompassing the wasteland of zeros from 0.5 keV to 3 keV.

The resulting grouped spectrum will arguably be a gross distortion of the truth: a high-flux narrow group followed by a low-flux wide channel. A χ^2 fit will take these misleading flux measurements at face value, and the model may be significantly biased.

AE’s grouping algorithm attempts to mitigate this problem by selecting group boundaries which fall mid-way in the run of empty channels (if any) that lie between events belonging to adjacent groups.

5.8 RMFs

If the event data have passed through the Townsley et al. CTI corrector⁵⁴, AE can be configured to select a response matrix file (RMF) for each source in each observation from a set of RMFs supplied with the CTI corrector.

If the data come from standard CXC processing, AE can be configured to construct an RMF using the *CIAO* tool *mkacisrmf* or *mkrmf*.

In a multi-observation reduction, a multi-ObsId RMF is constructed using the FTOOL *addrmf*, using the observation's EXPOSURE keywords as the weights.

5.9 ARFs

Construction of an ancillary response file (ARF) for each source in single observation consists of several steps.

- The *CIAO* tool *mkarf* is called to construct an ARF for each of the CCDs on which the source is observed. When the source spans multiple CCDs the ARFs are summed. The observer supplies appropriate aspect histogram files—these are typically obtained as a byproduct of exposure map construction tool (e.g. via the tool *ae_make_emap* in §7.14).
- The source extraction region is applied to each monochromatic PSF available (§5.1), and a PSF fraction is computed at each energy. The ARF is reduced by this PSF fraction curve (interpolated between sampled energies) to account for the source events that fell outside the extraction region.
- If desired, the observer can apply an externally-computed correction curve to the ARFs, e.g. a correction for a filter transmission or QE effect not yet modeled by the CALDB.

In a multi-observation reduction, a multi-ObsId ARF is constructed using the FTOOL *addarf*, using the observation's EXPOSURE keywords as the weights:

$$ARF = \frac{\sum_i EXPOSURE_i \times ARF_i}{\sum_i EXPOSURE_i} \quad (6)$$

A multi-ObsId source spectrum is constructed by summing the extracted spectra (eq. 1), and a multi-ObsId EXPOSURE value is defined as:

$$EXPOSURE = \sum_i EXPOSURE_i \quad (7)$$

To see that this is the appropriate way to combine the ARFs consider what happens inside the fitting program (e.g. *XSPEC*). The source model (*photons/s/cm²*) is multiplied by the multi-ObsId ARF (*cm²count/photon*) and by EXPOSURE (s) to produce a model spectrum (counts), shown on the left-hand side of eq. 9.

$$model \times ARF \times EXPOSURE = model \times \frac{\sum_i EXPOSURE_i \times ARF_i}{\sum_i EXPOSURE_i} \times EXPOSURE \quad (8)$$

$$= \sum_i model \times EXPOSURE_i \times ARF_i \quad (9)$$

By substituting the definition of the multi-ObsId ARF (eq. 6) and the definition of the multi-ObsId EXPOSURE (eq. 7) we can see that the model spectrum is precisely the sum of the data each ObsId is expected to produce (right-hand side of eq. 9).

It is interesting to consider the case where one of the extractions being merged has an ARF that is much smaller than the others (e.g. because it is in a chip gap, or because it has a very low PSF fraction). From equations 6 and 7 it's clear that including this sort of insignificant extraction in the merge can significantly change both the multi-ObsId ARF and the multi-ObsId EXPOSURE time. This seems very wrong at first glance, however, as shown in eq. 9, the two effects cancel each other within the fitting process.

⁵⁴<http://www.astro.psu.edu/users/townsley/cti/>

5.10 Broad Band Photometry

The multi-ObsId spectra and ARFs are used to perform background-subtracted photometry on the sources using a set of energy bands supplied by the observer. Starting with the following input data (**bold variables** are arrays with an element for each spectral channel):

- Integer observed PI spectral vectors from the source and background regions, **src_observed_counts** and **bkg_observed_counts**
- The ARF value in each of the spectral channels, **channel_arf**
- The scaling (a single quantity, not energy-dependent) of the extracted background spectrum required to model the background in the source aperture, $AREASCAL_{bkg}$, defined by Equation 5 in §5.6.
- The total exposure, *EXPOSURE*

the following photometry computations are performed. (Capitalized variables below are FITS columns in the photometry output file.)

5.10.1 Source and background counts with limits.

$$SRC_CNTS = \sum_{band} \mathbf{src_observed_counts}$$

$$BKG_CNTS = \sum_{band} \mathbf{bkg_observed_counts}$$

Standard 1- σ Gaussian confidence intervals for both SRC.CNTS and BKG.CNTS are computed using an analytical approximation to upper and lower confidence intervals of a Poissonian distribution (Gehrels, 1986, equations 7 and 12).

$$src_cnts_limit_up = SRC_CNTS + 1 + \sqrt{SRC_CNTS + 0.75}$$

$$bkg_cnts_limit_up = BKG_CNTS + 1 + \sqrt{BKG_CNTS + 0.75}$$

$$src_cnts_limit_low = SRC_CNTS * \left(1 - \frac{1}{9.0 * SRC_CNTS} - \frac{1}{3.0 * \sqrt{SRC_CNTS}}\right)^3$$

$$bkg_cnts_limit_low = BKG_CNTS * \left(1 - \frac{1}{9.0 * BKG_CNTS} - \frac{1}{3.0 * \sqrt{BKG_CNTS}}\right)^3$$

5.10.2 Background scaling, net counts with limits

$$\mathbf{bkg_counts_in_src_region} = \mathbf{bkg_observed_counts} / AREASCAL_{bkg}$$

$$\mathbf{net_counts} = \mathbf{src_observed_counts} - \mathbf{bkg_counts_in_src_region}$$

$$NET_CNTS = \sum_{band} \mathbf{net_counts}$$

$$BACKSCAL = \sum_{band} \mathbf{bkg_observed_counts} / \sum_{band} \mathbf{bkg_counts_in_src_region}$$

We propagate the SRC.CNTS and BKG.CNTS 1- σ confidence intervals through the equation $NET_CNTS = SRC_CNTS - BKG_CNTS / BACKSCAL$ using equation 1.31 in “A Practical Guide to Data Analysis for Physical Science Students”, L. Lyons, 1991 to get a confidence interval on NET.CNTS, conventionally stated as separate upper and lower 1- σ “errors” on NET.CNTS:

$$\begin{aligned}
NET_CNTS_SIGMA_UP &= \sqrt{(src_cnts_limit_up - SRC_CNTS)^2 + \left(\frac{bkg_cnts_limit_up - BKG_CNTS}{BACKSCAL}\right)^2} \\
NET_CNTS_SIGMA_LOW &= \sqrt{(src_cnts_limit_low - SRC_CNTS)^2 + \left(\frac{bkg_cnts_limit_low - BKG_CNTS}{BACKSCAL}\right)^2}
\end{aligned}$$

5.10.3 Source validity/significance

We compute a traditional “significance” as the photometric signal to noise ratio:

$$SRC_SIGNIF = \frac{NET_CNTS}{NET_CNTS_SIGMA_UP}$$

We also compute the “significance” of the observed SRC_CNTS as a disproof of the “null hypothesis” which is that there is no actual source, i.e. that all the observed counts are background. Weisskopf et al. (2007, Appendix A2) showed that the probability of observing SRC_CNTS or more counts in the source aperture and observing BKG_CNTS in the background region, conditioned on the assumption that there is no source (i.e. both apertures are sampling the same background population) can be found by integrating a binomial distribution. This calculation can be performed with the following call:

$$PROB_NO_SOURCE = binomial_nr(SRC_CNTS, SRC_CNTS + BKG_CNTS, \frac{1}{1 + BACKSCAL}) \quad (10)$$

The function *binomial_nr()* is a replacement for the function *binomial()* distributed with *IDL*, which is very unreliable (as of *IDL* version 8.5.1).

Since this expression takes into account the uncertainty one has in estimating the background level (i.e. the Poisson nature of BKG_CNTS), PROB_NO_SOURCE will increase as BKG_CNTS decreases, i.e. a source will be less significant (more consistent with the background) when the background is poorly estimated.

As one might expect, when the background is accurately estimated (BKG_CNTS is large) this expression approaches a simple integral of the Poisson distribution over the interval [SRC_CNTS, ∞] (Weisskopf et al., 2007, § A2), which is equal to:

$$1 - \sum_{i=0}^{SRC_CNTS-1} Poisson(i; bkg_counts_in_src_region) \quad (11)$$

Important Note: Changing the extraction region will usually change the source significances defined above since both SRC_CNTS and BKG_CNTS will change. The maximal significance is NOT obtained with the largest extraction region (PSF Fraction) because as you enlarge the extraction region you eventually start adding more background than signal and significance goes down.

Some observers choose to use these AE significance statistics in their source detection process—a liberal set of source candidates are nominated by various means, those candidates are extracted by AE, and the catalog is pruned by thresholding on the AE significance statistics. One might choose this approach based on the hope/expectation that AE’s significance statistics are more accurate than those computed by source detection codes because AE has a more complex process for estimating the background for each source (critical for these statistics). Or, one might choose this approach because some source candidates are nominated via processes that do not offer significance statistics (e.g. sources detected by-eye, sources detected in reconstructed images, or sources identified from other wavebands).

Note that this sort of procedure has strong analogies to the classic “cell detection” algorithm. Viewed in this way, AE can be seen to be evaluating the source significance statistic on only a limited number of cells—the AE extraction regions associated with the list of proposed sources. The procedure for nominating sources can be seen to be merely an efficiency technique—we avoid running AE at the many potential source locations where we have a low expectation that the AE significance statistics would be interesting.

Please note that it is not at all clear whether SRC_SIGNIF or PROB_NO_SOURCE threshold values used to prune the catalog can be used to analytically estimate the number of spurious sources produced by this hybrid procedure. Even though PROB_NO_SOURCE is a classic confidence level, there is no obvious “number of independent trials” that can be identified in the hybrid detection process outlined above. Laborious simulations would seem to offer the only hope of quantifying false detection rates for such a detection process. Such simulations may be nearly impossible when the source nomination process is not simple/automated.

Again viewed as a form of cell detection, the hybrid detection process outlined above might be expected to produce spurious sources which are spatially non-uniform because the cell size follows the *Chandra* 90% PSF size. Since the density of possible cells (extraction regions) is higher on-axis one might expect more spurious sources there (as observed in simulations with *CIAO’s celldetect*). **I think observers should be very cautious about performing studies of source spatial distribution that include weak sources near the limits of detection for ANY detection method that is used.**

5.10.4 Photon Flux

AE provides two estimates for the flux of photons incident on *Chandra*, which has units of photon /cm**2 /s.

$$\text{net_flux} = \text{net_counts}/\text{channel_arf}/\text{EXPOSURE}$$

$$\text{FLUX1} = \sum_{\text{band}} \text{net_flux} \quad (\text{photon}/\text{cm} * 2/\text{s})$$

Note that for weak sources the FLUX1 estimate can suffer from large **Poisson** errors with respect to true incident flux because events (either source or background) at energies where the ARF is very small have a large effect on FLUX1. For example, one observed event at 8 keV, where the ARF is tiny, makes a much larger contribution to FLUX1 than one event at 2 keV where the ARF is large.

$$\text{MEAN_ARF} = \frac{\sum_{\text{band}} \text{channel_arf}}{\sum_{\text{band}} 1}$$

$$\text{FLUX2} = \text{NET_CNTS}/\text{MEAN_ARF}/\text{EXPOSURE} \quad (\text{photon}/\text{cm} * 2/\text{s})$$

Note that the FLUX2 estimate suffers from a **systematic** error with respect to true incident flux because the scalar MEAN_ARF (a simple mean) is the correct effective area normalization only if the incident spectrum is flat. Some observers have found that summing the FLUX2 values computed over several adjacent narrow energy bands will produce a more accurate result than a single wide-band FLUX2 value.

BEWARE: The Mean ARF and the roughly-calibrated FLUX2 depend on the shape of the Effective Area curve, which varies between grating configurations, varies between ACIS detectors, and varies over time (§2.3). Obviously, spectral fitting should produce the most accurate flux estimates.

5.10.5 Characterizations of the background subtracted OBSERVED spectrum

AE provides estimates for the median (50% quantile), 25% quantile, and 75% quantile of the observed event energies over a variety of bands. These statistics are background-corrected, meaning they seek to characterize the observed spectrum of the astrophysical source if background were not present. AE implements an intuitive and straightforward background correction for standard quantiles, based on the observed cumulative distribution of the net spectrum, as shown in Figure 4. This method appears to be equivalent to that described by Hong et al. (2004, Appendix C), which was developed independently. AE does not yet attempt to estimate individual confidence intervals for these background-corrected quantiles; we expect that resampling techniques would be the best approach to that task.

$$\text{ENERG_PCT25_OBSERVED} = 25^{\text{th}} \text{percentile energy}$$

$$\text{ENERG_PCT50_OBSERVED} = 50^{\text{th}} \text{percentile energy}$$

$$\text{ENERG_PCT75_OBSERVED} = 75^{\text{th}} \text{percentile energy}$$

These are the energies below which 25%, 50%, and 75% of net_counts are found. The 50% percentile could be called a “median observed energy corrected for background”. The mean observed event energy, corrected for background, is also estimated.

$$\text{ENERG_MEAN_OBSERVED} = \sum_{\text{band}} [\text{channel_energy} * \text{net_counts}] / \sum_{\text{band}} \text{net_counts}$$

Note that the weighted mean calculation above is done on very narrow energy “channels” (defined by *CIAO*). Thus, the net_counts will generally change sign with high frequency (e.g. positive when a channel contains an extracted count but no background count, and negative when a channel contains a background count but no extracted count). When the background spectrum is well sampled and properly scaled, those negative net_counts properly correct ENERG_MEAN_OBSERVED for background. However, Poisson noise or improper background scaling can cause one end of the net spectrum to be mostly negative. That non-physical negative net spectrum can severely bias the calculation above, to the point of producing a ENERG_MEAN_OBSERVED value that lies outside the photometry energy band.

BEWARE: These uncalibrated properties depend on the shape of the Effective Area curve, which varies between grating configurations, varies between ACIS detectors, and varies over time (§2.3).

5.10.6 Characterizations of the background subtracted INCIDENT spectrum

Similarly, AE provides estimates for the median (50% quantile), 25% quantile, and 75% quantile of the energies of photons incident on *Chandra* over a variety of bands.

$$ENERG_PCT25_INCIDENT = 25^{th} \text{percentile energy}$$

$$ENERG_PCT50_INCIDENT = 50^{th} \text{percentile energy}$$

$$ENERG_PCT75_INCIDENT = 75^{th} \text{percentile energy}$$

These are the energies below which 25%, 50%, and 75% of net_flux (§5.10.4) are found. The 50% percentile could be called a “median incident energy corrected for background”. All the INCIDENT statistics can suffer from large **Poisson** errors as discussed for FLUX1 above. The mean incident event energy, corrected for background, is also estimated.

$$ENERG_MEAN_INCIDENT = \sum_{band} [\text{channel_energy} * \text{net_flux}] / \sum_{band} \text{net_flux}$$

Due to background subtraction, ENERG_MEAN_INCIDENT can fall outside the energy band!

5.11 Light Curves

Two light curves are constructed for each observation of each source. One is “grouped” (like spectra are grouped) to produce flux estimates in independent, unequally-sized time bins. The second is adaptively smoothed to produce flux estimates in overlapping (i.e. correlated), unequally-sized time intervals with constant significance.

Time variability of the source both within and among observations is quantified by comparing a uniform **flux** model to the distribution of time stamps for extracted events, and computing the 1-sample Kolmogorov–Smirnov statistic between the flux model and the observed distribution.

BEWARE: these data products currently make no distinction between events produced by the source and background events. Variation in the aperture background rate among ObsIDs can lead to a spurious indication of source variability. Such background variation can occur because the aperture size can vary among ObsIDs (e.g. due to variations in off-axis angle), because background from neighboring sources can vary among ObsIDs, and because FI and BI devices have different levels of instrumental background.

Time variability of the source among observations is quantified by calculating the p-value of the χ^2 statistic of the single-ObsID photon fluxes, under the null assumption that the flux is constant (at the weighted mean value).

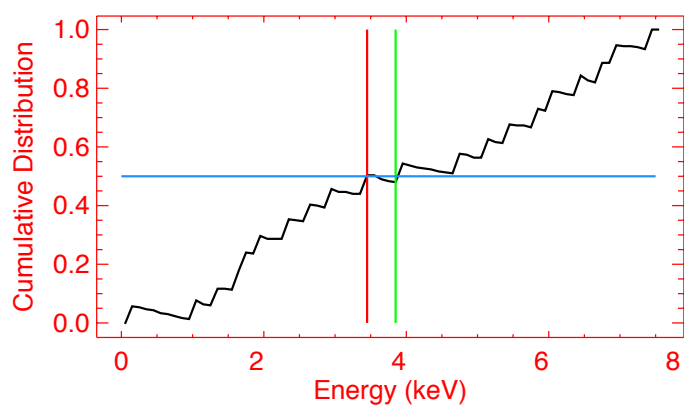


Figure 4: Example calculation of the background-corrected median energy statistic. Each large upward jump in the cumulative distribution of net counts (black) represents the 20 counts observed in a source aperture; each small downward jump represents the 100 counts observed in a background region, scaled down to match the source aperture size. The lowest energy (red line) and highest energy (green line) at which the 50th percentile (blue line) is reached

5.12 Automated Spectral Fitting

An excellent introduction to the concept of fitting X-ray spectra can be found in one of Keith Arnaud’s PowerPoint presentations⁵⁵.

The FIT.SPECTRA stage of AE (§7.11) can automate spectral fitting by running the *XSPEC* package⁵⁶ (Arnaud, 1996) on each source’s extraction data products. The *XSPEC* command interface is mediated through the Tcl scripting language, providing powerful programming features such as string manipulations, looping, conditions, and ASCII file I/O. An *XSPEC* script following certain interface requirements described in §7.11 must be supplied to AE.

Several such scripts, implementing simple spectral models, are distributed in the AE package. These carefully crafted scripts (originally developed by Konstantin Getman) fit the observed spectrum, estimate 90% confidence intervals on the fit parameters, calculate fluxes and estimate 90% flux confidence intervals over several energy bands, and produce various text and PostScript output. Both of *XSPEC*’s fit statistics, χ^2 and the C-statistic, are supported. The MODEL.CHANGES.FILENAME mechanism described in §7.11 can be used to override some script parameters without editing the script. If you choose to build your own *XSPEC* script that is compatible with AE, please consider providing it back to the AE community for use by your colleagues.

The following scripts are distributed with AE:

```
compact/2T:LMC,LMC.xcm
compact/2T:Solar,Solar.xcm
compact/2T:Solar,TT.xcm
compact/3T:Solar,Solar.xcm
compact/T+T:LMC,LMC.xcm
compact/T+T:Solar,Solar.xcm
compact/T:LMC,LMC.xcm
compact/T:Solar,Solar.xcm
compact/T:Solar,TT.xcm
powerlaw/tbabs_pow.xcm
powerlaw/tbvarabs_pow.xcm
bremss/tbvarabs_bremss.xcm
diffuse/diffuse_1apec.xcm
diffuse/diffuse_1pshock.xcm
diffuse/diffuse_3apec.xcm
diffuse/diffuse_3pshock.xcm
diffuse/diffuse_3vpshock.xcm
```

- The “compact” directory contains thermal plasma models suitable for point sources. In these script names, two terms after the colon (e.g. “Solar,TT”) specify the abundances used for the absorbing column and the emitting plasma. “TT” abundances (appropriate for low-mass young stars) are frozen to the following values, which are the abundances adopted by the XEST study (Güdel et al., 2007), relative to Anders & Grevesse (1989), scaled to Wilms et al. (2000), using the *tbabs* absorption model in *XSPEC*:

⁵⁵[http://lheawww.gsfc.nasa.gov/ kaa/india2003/lowresspec.ppt](http://lheawww.gsfc.nasa.gov/kaa/india2003/lowresspec.ppt)

⁵⁶<http://heasarc.gsfc.nasa.gov/docs/xanadu/xspec/>

He	1.0
C	1.0
N	1.0
O	0.75
Ne	1.17
Mg	0.39
Al	1.0
Si	0.57
S	0.55
Ar	0.78
Ca	0.29
Fe	0.35
Ni	0.32

The characters before the colon specify the form of the model (“T” = one plasma component; “2T”, “3T” = two/three components behind a common absorber; “T+T” = two components behind independent absorbers).

Although the MODEL.CHANGES.FILENAME mechanism described in §7.11 can not be used to thaw abundances, the /INTERACTIVE option can be used to allow the observer to fit with variable abundances within the AE context.

- The “powerlaw” and “bremss” directories contain powerlaw and bremsstrahlung models.
- The “diffuse” directory contains complex models suitable for extended Galactic sources.

Spectral fitting of weak sources remains a poorly understood task among the community of X-ray observers. If χ^2 is to be used there seems to be little consensus on the best trade off between group size and the number of groups. For both χ^2 and C-statistic methods there seems to be little consensus on the conditions necessary for the parameter **error** estimates produced by the delta-fit-statistic methods in *XSPEC* to be appropriate/reliable.

5.12.1 Background Spectra and the C-statistic

The C-statistic, an application of the Likelihood Ratio test, has long been recognized (Cash, 1979) as more appropriate for low-count data than the traditional χ^2 statistic. The concept of “subtracting” a background does not exist when using the C-statistic since the method involves computing the likelihood of the observed data set itself, the set of individual X-ray events. One can however consider the background data to be part of the generalized observation, form a model that predicts **all** the observed data (events from both the target and background regions), and calculate the C-statistic on the data and model. For some time now the C-statistic in *XSPEC* has provided this capability (Appendix B of the *XSPEC* manual⁵⁷). When a background spectrum is supplied and the C-statistic is selected, the source and background data are simultaneously modeled (Arnaud, see <http://heasarc.gsfc.nasa.gov/docs/xanadu/xspec/wstat.ps>; see <https://giacomov.github.io/Bias-in-profile-poisson-likelihood/>) using a method derived by Wachter et al. (1979).

Several observers using this Cstat-with-background capability have noted that seemingly undesirable fits are fairly common. For thermal models, typically the plasma temperature will be obviously too low, leading to very poor fits at high energies. (Examples of this problem will be shown later.)

We believe that these problems are the result of the under-constrained nature of the background model used by Wachter et al. (1979). That model represents the incident background flux in each spectral bin with an independent parameter, with no constraints. Thus, in an ungrouped ACIS spectrum with 500 channels the Wachter background model contains 500 free parameters. (The best-fit background parameter values are computed by algebraic manipulation (Wachter et al., 1979) and are thus hidden from *XSPEC*’s Cstat minimization search.)

In our view, such an extremely flexible model of the ACIS background is inappropriate. For many of our ACIS sources, both the extracted source spectrum and the background spectrum have relatively few counts (far fewer than the ~500 parameters in the Wachter background model); in any single channel both spectra will typically have only zero or one observed count. As it tries to follow the data, this flexible background model should be driven to zero (an unphysical result) in all channels that have no source or background events. One can imagine the more

⁵⁷<http://heasarc.gsfc.nasa.gov/docs/xanadu/xspec/manual/manual.html>

troubling tendency of the background model to try to “follow” the source counts, i.e. to spike upward at each channel where there is a source count. See <https://giacomov.github.io/Bias-in-profile-poisson-likelihood/> for a beautiful demonstration of this. This could allow the background model to siphon off flux that should be represented in the source model. More generally, one should expect that this flexible background would be very sensitive to the random (physically meaningless) alignment of the source and background counts, e.g. whether a source and background count of similar energy happened to fall in the same channel or in adjacent channels.

Motivated by these concerns, we decided to implement a smoother (more constrained) physics-free model for ACIS background spectra to replace the Wachter background model when the C-statistic is in use. We arbitrarily chose a model family consisting of continuous piecewise-linear functions with 2 to 10 vertexes; an example of such a model is shown in Figure 5. The model has 2 to 10 parameters representing the X-ray fluxes at the vertexes’s. The vertexes’s are placed on the energy scale so that they divide the energy range into intervals with approximately equal numbers of observed counts in the background spectrum (0.1 to 10 keV). Vertex energies are chosen to coincide with the energies of actual events in the background—this helps to prevent the vertex flux from being driven to the hard limit of zero during the fitting process. We use the name *cplinear* to refer to this model. Obviously, this smooth model would not be appropriate for very high quality background spectra that have significant of structure.

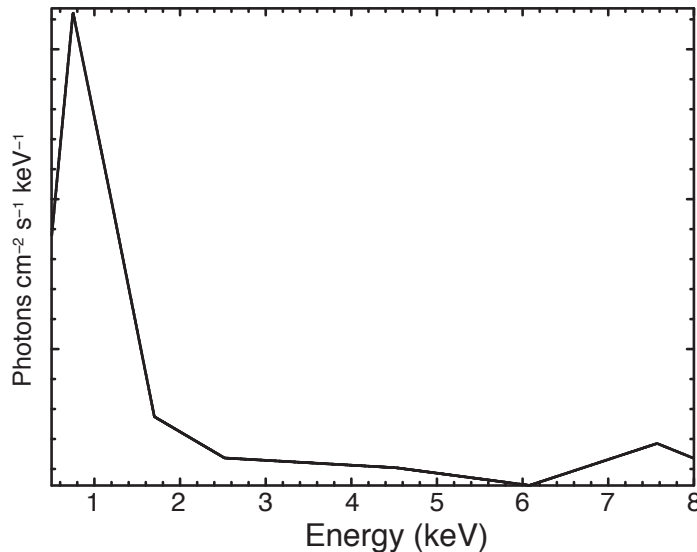


Figure 5: Example of a continuous piecewise-linear (*cplinear*) background model.

The AE fitting scripts configure *XSPEC* as shown in Figure 6. The background spectrum is compared (via the C-statistic) to a *cplinear* model passed through a flat ARF, rather than through the *Chandra* ARF. Thus we are in effect modeling the observed ACIS background, not any sort of incident astrophysical background. The source spectrum is compared (via the C-statistic) to the sum of two models:

- A copy of the *cplinear* model represents the background expected in the source extraction aperture.
- An astrophysical model (e.g. *tbabs * vapec*) passed through the *Chandra*-ACIS response represents the observed spectrum of the target.

XSPEC simultaneously fits these models, i.e. minimizes the sum of the C-statistic computed on the source spectrum (likelihood of the source spectrum given the source model) and the C-statistic computed on the background spectrum (likelihood of the background spectrum given the background model). It’s worth repeating that this strategy of simultaneously maximizing the Poisson likelihood of all the data (source spectrum and background spectrum) is NOT new; Wachter’s method also does this. We have simply adopted a more constrained background model, and are searching for its best-fit parameters within the context of *XSPEC*’s minimization machinery rather than solving for background parameters algebraically.

We have found that the *cplinear* model produces far fewer fits that are obviously poor. In Figure 7, below each example of a poor fit produced by the Wachter background method (rows 1 and 3) is shown an arguably better fit to

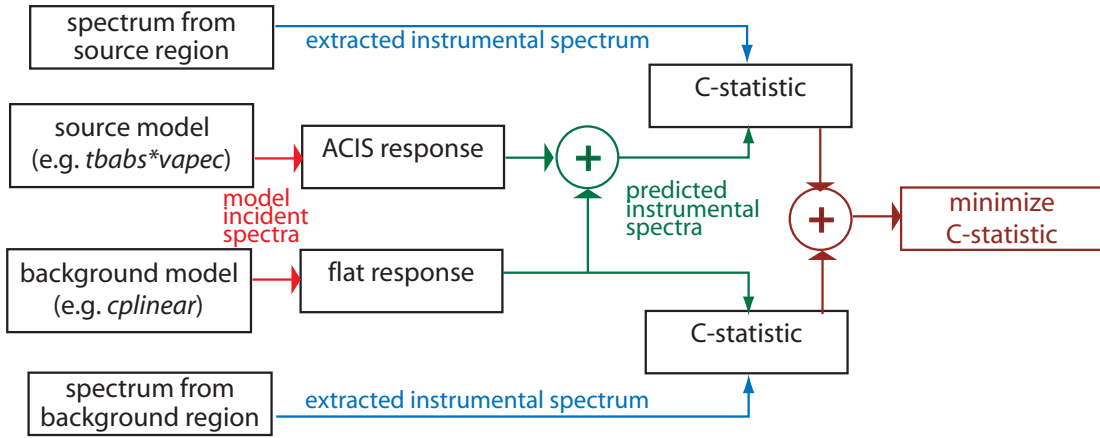


Figure 6: Configuration of maximum likelihood model for source and background spectra. The green plus sign represents the addition of two predicted instrumental spectra (vectors); the red plus sign represents the addition of two C-statistic values (corresponding to multiplication of the likelihood of the source and background data).

the same source produced by the *cplinear* method (rows 2 and 4). Appendix B compares the Wachter and *cplinear* models applied to real ACIS spectra.

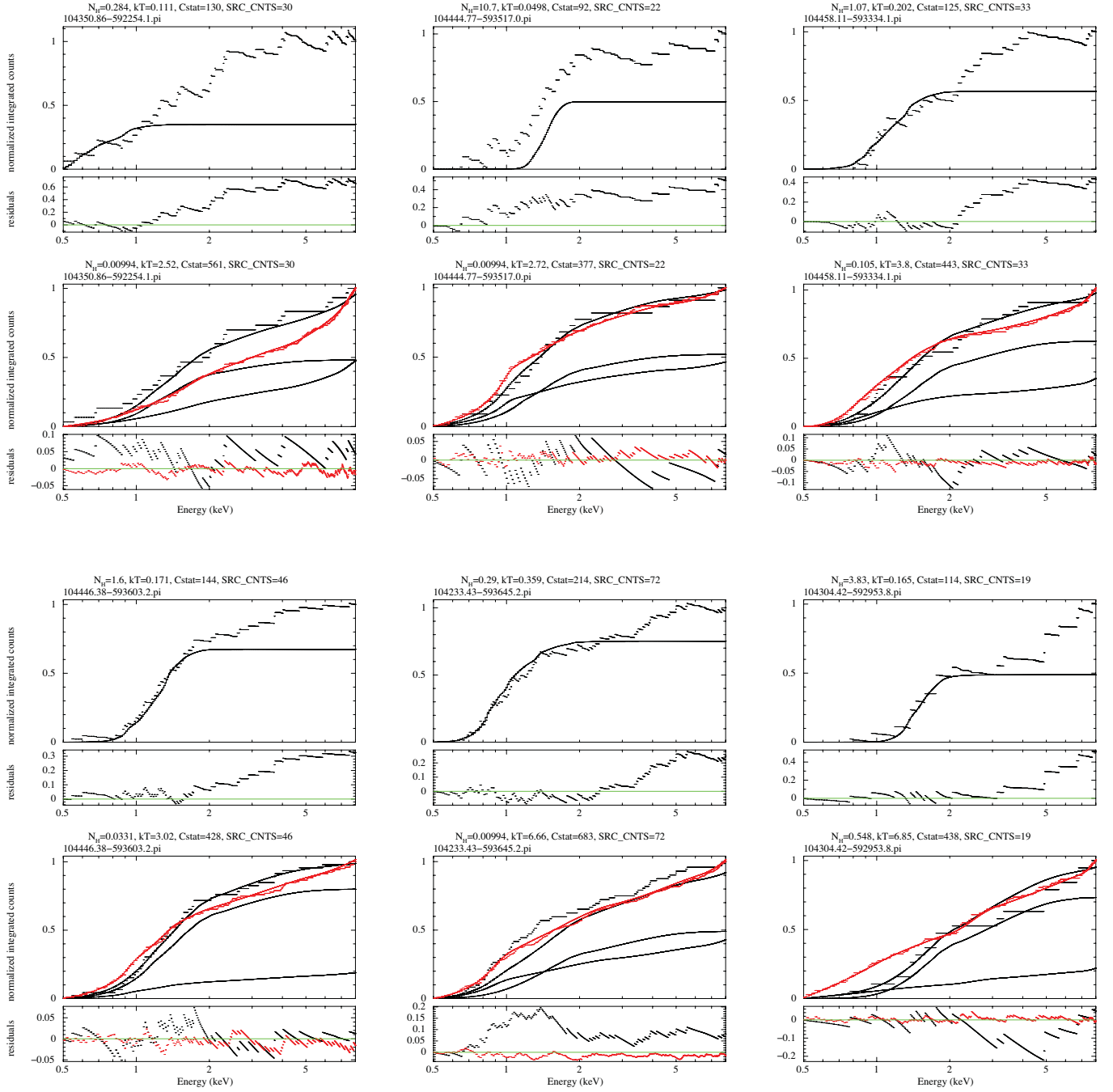


Figure 7: Spectral models for six sources derived using the Wachter (rows 1 and 3) and cplinear (rows 2 and 4) background models (see Figure 6). The Wachter plots (rows 1 and 3) show the observed cumulative *net* spectrum (stair-step curve) and the best-fit thermal plasma model (continuous curve) with residuals. Note that the fit is often poor. The cplinear plots are more complex. Two stair-step curves (black and red) show the cumulative spectra observed in the source aperture and background region. The continuous red curve shows the cplinear model of the background spectrum (often not visually distinguishable from the background data). Three black continuous curves depict the two components of the model for the events observed in the source aperture (lower curves), and their sum (upper curve). One component—a copy of the red *cplinear* background model, scaled down by the exposure ratio between the source and background regions—models the background in the aperture. The second component models a thermal plasma in the astrophysical source. Residuals are shown at the bottom.

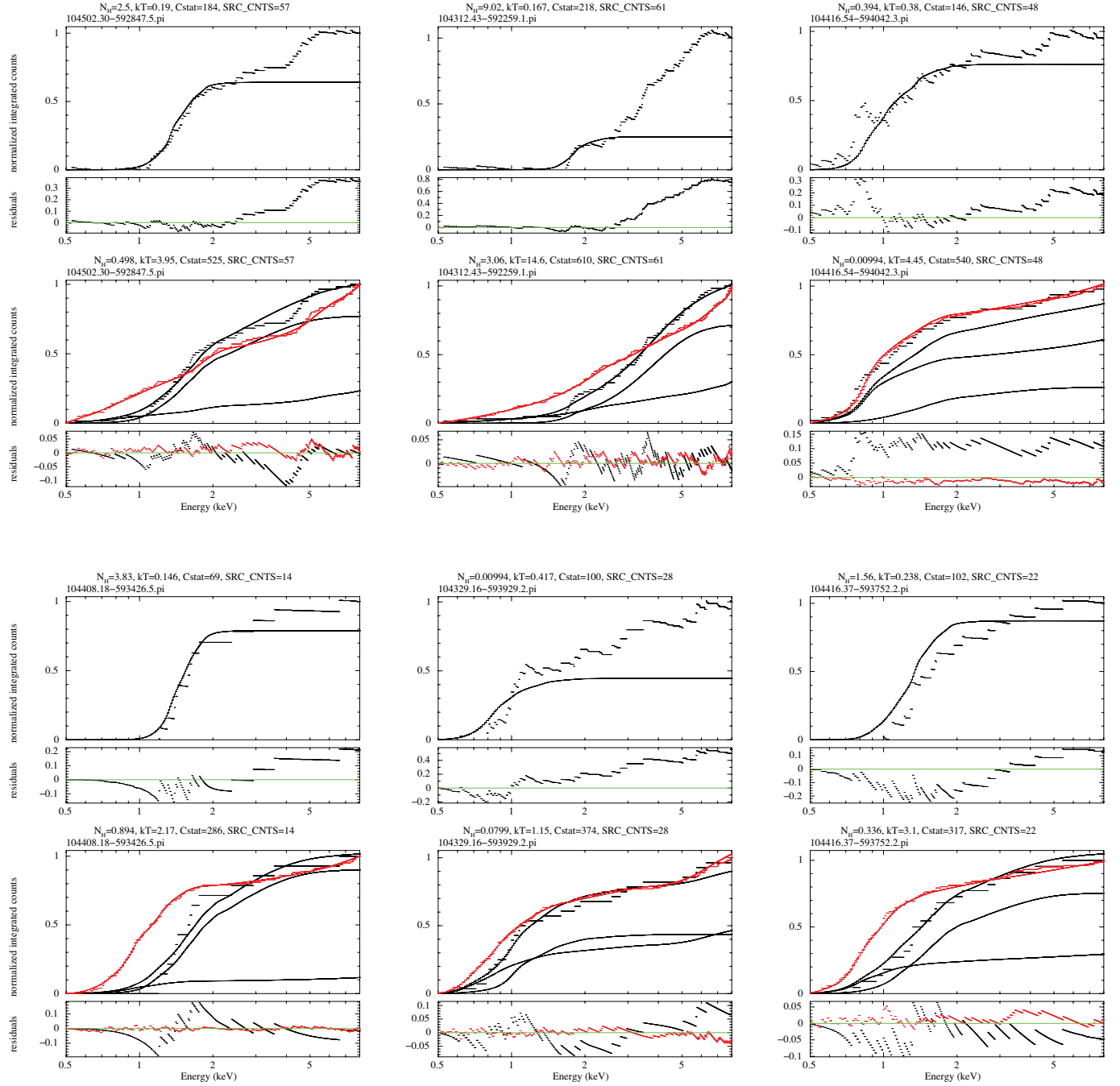


Figure 7: (continued) Spectral models for six sources derived using the Wachter (rows 1 and 3) and cplinear (rows 2 and 4) background models.

5.13 Discarding Observations

In a multi-observation analysis, single-observation extractions are combined to form multi-ObsId data products and source properties, as described in the previous sections. However, once the alternate strategy of merging extractions is adopted it would be foolish to blindly merge *all* the extractions available for every source.

Consider, for example, a source that has two extractions—one beautiful observation on-axis using a small aperture containing almost zero background, and a second observation far off-axis using a huge aperture containing many background counts. If the source is very bright and the off-axis observation is not crowded, then the source properties are most accurately estimated by merging the two extractions, since the extra signal gained from the off-axis observation is more important than the relatively insignificant additional background that is suffered. However, if the source is very weak, then retaining the horrible off-axis observation would utterly corrupt the quality of the source properties—the merged extractions may even produce negative net counts (by chance). Clearly, an observer tackling misaligned *Chandra* pointings must make some sort of decision about which source extractions should be discarded. AE implements several algorithms for discarding extractions, described in the following subsections.

The notion of discarding data may seem strange at first, but observers routinely discard ACIS data that was obtained during periods of very high instrumental background⁵⁸ (due to solar activity). For convenience, this is commonly done in the early stages of data analysis, guided by the damage the enhanced background will do to the most sensitive sources (e.g., diffuse sources). A somewhat better strategy would be to choose high-background periods to discard on a source-by-source basis, since very bright sources would benefit more from extra integration time than from a reduction to their already insignificant background. AE has not yet adopted this optimum strategy for time filtering because implementation is difficult.

5.13.1 Overlap Rejection

All merges will discard extractions that suffer extreme crowding because in those conditions we do not have confidence that we can effectively estimate backgrounds for each extraction. Crowding is measured by an OVERLAP metric assigned to each extraction by the `ae.make.catalog` tool (§7.5), quantifying how much the extraction region overlaps its nearest neighbor (closely-spaced sources can still overlap substantially even though AE automatically reduces the extraction region to minimize overlap; this is because we have imposed a floor in that reduction so that we never have an extraction region containing $\leq 40\%$ of the full PSF). The MERGE stage (§7.8) accepts an OVERLAP_LIMIT threshold parameter; extractions exceeding this threshold are considered to have excessive overlap and will be discarded.

5.13.2 Optimization for Source Validity (PROB_NO_SOURCE statistic)

When the observer is interested in the *validity* of proposed sources using AE’s PROB_NO_SOURCE statistic (the probability that the source is just a background fluctuation; §5.10.3 Weisskopf et al. (2007, Appendix A2), then we recommend an AE option that selects whatever subset of the available extractions optimizes (minimizes) PROB_NO_SOURCE for each source. A slogan that describes this strategy would be “I believe a source exists if it is significant in any observation, or in any combination of observations”. For highly variable astrophysical sources, such as young stars, this is a reasonable strategy even when the pointings are aligned (since the non-flaring observations can be ignored). An extraction much further off-axis than its peers will be included only when it contributes “more” signal than background.

AE’s algorithm is simply an extension of the common practice of searching for sources within each ObsID separately, and also within observer-defined combinations of ObsIDs. All such approaches increase a project’s sensitivity (especially to variable sources) at the expense of an increased false detection rate (from the additional number of random “trials” that can produce spurious detections). **If you have a large number of overlapping ObsIDs, this algorithm is able to try a very large number of combinations of those ObsIDs to find the strongest “detection”. The resulting “repeated trials” bias can be large.** In such cases, a better strategy may be to test source significance with a small number of pre-defined ObsID combinations, e.g. defined by ranges in off-axis angle (THETA_RANGE option, §7.8).

This algorithm is chosen by the `/MERGE_FOR_PB` option to the MERGE stage (§7.8).

⁵⁸<http://cxc.harvard.edu/ciao/threads/acisbackground/index.py.html>

5.13.3 Optimization for Source Position

When the observer is interested in the *position* of sources, then we recommend an AE option that first discards the ObsIDs that the MERGE_FOR_PB option would discard, and then selects whatever subset of the remaining extractions optimizes (minimizes) the expected position uncertainty (§5.3) for each source. A slogan that describes this strategy would be “I believe all observations of a source are well aligned, so let’s estimate the position using only the best data we have”. An extraction much further off-axis than its peers will be included only when the increase in the number of merged counts outweighs the increase in the spread of the merged counts.

This algorithm is chosen by the /MERGE_FOR_POSITIONS option to the MERGE stage (§7.8).

5.13.4 Optimization for Photometry and Spectral Fitting

When the observer is interested in *time-averaged photometric* properties (e.g., fluxes, spectra) then selecting the extractions to merge becomes more dangerous. Any time extractions are discarded in order to optimize a photometric quantity (e.g., signal-to-noise ratio) a bias can be introduced into all photometric properties because the discarded extraction may have, merely by chance, fewer observed counts from the source than the long-term average. Thus, the observer must balance two undesirable outcomes: sources whose photometry is spoiled by including very poor-quality extractions, and sources whose photometry suffers the suspicion of bias because some extractions were discarded after looking at their data.

AE offers an option that strikes this balance by discarding extractions only when retaining them would drive the signal-to-noise ratio of the merged data set significantly below the optimal signal-to-noise ratio. The observer specifies the minimum acceptable ratio between the signal-to-noise ratio achieved by the merge and the optimal signal-to-noise ratio achievable by discarding more extractions. A slogan that describes this strategy would be “I will tolerate some, but not too much, damage to SNR in order to avoid the specter of bias in the photometry”.

This algorithm is chosen by the /MERGE_FOR_PHOTOMETRY option to the MERGE stage (§7.8).

5.13.5 Optimized merges may hide source variability

Assessing source variability is sometimes an important science goal. Note that all these optimized merges tend to discard ObsIDs in which the source flux is low. That may lower the effectiveness of source variability metrics.

5.14 Diffuse Sources

Correct extraction of a diffuse source differs from extraction of a point source in a number of important ways.

5.14.1 The slippery concept of a region’s area on the sky

Observers commonly need to express the observed strength of a diffuse source in terms of surface brightness, for example normalizing a luminosity calculated via *XSPEC* by some measure of the size of the source on the sky. If the response of the observatory was constant within the selected extraction region, then the appropriate size normalization would simply be the geometric area of the region. However, in a typical ACIS observation the response varies strongly across the extraction region in several ways:

- The HRMA and ACIS CCDs have spatially varying responses.
- The extraction region will likely include regions of reduced or zero exposure time, including dithered bad columns, dithered chip gaps, dithered detector edges, and point source masks.

When a diffuse source is extracted in *CIAO*, this spatially-varying response is abstracted/averaged into a single set of response files (ARF and RMF). Obviously, the appropriate region size normalization depends on how this average response is calculated, since in the denominator of the final surface brightness expression, e.g. ($\text{arcsec}^2 \text{scm}^2$), the response of the observatory and the size normalization are degenerate (i.e. are multiplied together). Now, **if** one averaged the observatory response over the region in the SKY coordinate system, $\overline{ARF}(E) = \int_R ARF(E, x, y)$, (including the effects of bad columns, chip gaps, detector edges, and point source masks) then that multi-ObsId response would account for everything, and the appropriate size normalization would simply be the geometric area of the region. However, it is important to understand that this is **not** the algorithm employed by the tool *mkwarf*. Instead, *mkwarf* (through its WMAP input) forms a weighted average of the response of the observatory within a

set of cells on the detector. In this process there is no concept of reduced exposure time arising from dithering over unobserved parts of the focal plane, and there is no concept of point source masking.

The good news is that we have on hand a data product that does represent the “depth” of the observation (exposure time \times effective area) everywhere, namely the exposure map

$$emap(x, y) = ARF(E_0, x, y) \times EXPOSURE$$

computed for the energy E_0 . The unsubscripted function $ARF(E_0, x, y)$ is the response of the observatory at a specific position on the *sky* for energy E_0 . $ARF(E_0, x, y)$, and thus the exposure map, represents both variations in the observatory response across the focal plane, and the exposure time variations across the sky caused by dithering over bad pixels and detector edges. The AE workflow also applies the point source masks to the exposure map, producing a data product that fully maps the response of the observatory at the single energy E_0 . The integral of this masked exposure map over the extraction region, $I_{emap} = \int_R emap(x, y)$, represents precisely the denominator of the final surface brightness expression that we seek, in units of $arcsec^2 \times s \times cm^2 \times count/photon$, for the specific mono-energy E_0 .

Given that the ARF produced by *mkwarf* is the only convenient representation we have for the energy-dependence of the response, a reasonable approach would seem to be to choose any scaling for that ARF and/or for the EXPOSURE time and/or for the geometric region area such that in the end our extracted spectrum is normalized by I_{emap} at energy E_0 . AE chooses to scale *mkwarf*'s ARF to produce a “surface brightness ARF” (designated below by the subscript SB) as follows:

$$\begin{aligned} ARF_{SB}(E) &= ARF_{mkwarf}(E) \times \frac{\int_R emap(x, y)}{ARF_{mkwarf}(E_0) * EXPOSURE} \\ &= ARF_{mkwarf}(E) \times \frac{\int_R ARF(E_0, x, y) * EXPOSURE}{ARF_{mkwarf}(E_0) * EXPOSURE} \\ &= ARF_{mkwarf}(E) \times \frac{\int_R ARF(E_0, x, y)}{ARF_{mkwarf}(E_0)} \end{aligned}$$

At the energy $E = E_0$,

$$ARF_{SB}(E_0) = \int_R ARF(E_0, x, y)$$

, which is the observatory response averaged over the entire geometric area of the region (including unobserved portions of the sky).

The units of the ARF are thus changed from $cm^2 count/photon$ to $arcsec^2 cm^2 count/photon$. All “flux” quantities derived from *XSPEC* should then be understood to be surface brightness quantities with $arcsec^{-2}$ appended to the units. Actual integrated fluxes over the entire diffuse region are then estimated by multiplying inferred surface brightnesses by the geometric area of the region.

5.14.2 Multiple observations should be combined in surface brightness units

The conversion of each extraction’s calibration (ARF) to surface brightness units, described above, is very convenient when multiple observations are to be combined. Each extraction of a diffuse region will generally have different sub-regions that are unobserved in that ObsID, and thus different normalizations for *mkwarf*'s ARF. Once all the extractions are calibrated in surface brightness units, they can be straightforwardly merged in the same way that point sources are merged (§5.6). There is a clear analogy between this practice and the way AE handles PSF fractions when multiple point source extractions are merged; in that case since each extraction can have a different PSF fraction, AE chooses to scale each observation’s ARF by its PSF fraction prior to merging.

5.14.3 Analysis of background

There are several ways background matters can be approached in diffuse analysis. See §7.1.2.

6 Installation

Please see the *ACIS Extract* web page⁵⁹ for instructions on obtaining and installing the AE package and other *IDL* software required.

The following third-party software are also required:

- The **IDL Language Interpreter** version 8.5.1 or higher is required⁶⁰
- The 25-Jul-2016 or later version of Wayne Landsman’s *IDL* Astronomy Library⁶¹ (AstroLib) is required and must be in your *IDL* path.

AE versions prior to 2016sep22 MUST USE AstroLib versions prior to 22-Sep-2015! See What’s New (§2) for AE Version 2016sep22.

- The Coyote Graphics Library, either from its author⁶² or from http://idlastro.gsfc.nasa.gov/ftp/coyote_astron.tar.gz
- We strive to evolve AE to be compatible with the latest version of *CIAO* and the *Chandra* CALDB, so we generally recommend maintaining your *CIAO* installation at the current release level. When AE is approved for a new *CIAO* release, we will say so in the What’s New section of this manual. You must install the optional *CIAO* contributed scripts.
- The *MARX*⁶³ simulator, version 5.3 to 5.5.1, is required to model the *Chandra* PSF. Later versions are NOT acceptable. In the *MARX* installation instructions⁶⁴ watch for special steps required for the OS-X operating system.

MARX has tools in two different directories (`{marx_installation_directory}/bin/` and `{marx_installation_directory}/lib/marx/`), both of which must be placed into your Unix path, either by your login setup or by your *ciao* alias. *MARX* also requires that you set the environment variable `MARX_DATA_DIR` to `{marx_installation_directory}/share/marx/data/`.

- The *addarf* and *addrmf* tools in the **FTOOLS** package (*HEASOFT* version 5.2 or higher) from GSFC are also required.
- The **XSPEC** package from GSFC is required if automated spectral fitting is to be performed. The version of *XSPEC* required depends largely on the *XSPEC* scripts you use. The *XSPEC* scripts distributed with AE (§5.12) require *XSPEC* version 12.7.0 or higher. One of the optional capabilities of AE’s fitting stage (/INTERACTIVE) requires a minor configuration change to *XSPEC*, described in §7.11.
- \LaTeX is used in the fitting stage of AE to produce a summary of each spectral fit.

6.1 Unix Shells

AE executes *CIAO* and *HEASOFT* programs using the *IDL spawn* command to create a Unix shell which then parses and executes *CIAO/HEASOFT* command lines. **Five requirements govern your choice of Unix shell and the configuration of that shell:**

1. **Your shell must offer the *setenv* command for managing unix environment variables.** AE uses the *setenv* shell command to control `PATH` and other environment variables in shells it spawns.

If you use a shell (such as *bash*) that does not offer *setenv*, **there should be no need** to change the default shell of your account. Instead, you should be able to alter the `SHELL` environment variable in whatever window will be launching *IDL* so that the new shells that *IDL* will create will be *csh* (or *tcsh*). For example, if you use the *bash* shell then launch *IDL* like this:

⁵⁹http://personal.psu.edu/psb6/TARA/ae_users_guide.html

⁶⁰ **IDL version 6.0 definitely should not be used.**

⁶¹<http://idlastro.gsfc.nasa.gov/homepage.html>

⁶²http://www.idlcoyote.com/documents/programs.php#COYOTE_LIBRARY_DOWNLOAD

⁶³<https://space.mit.edu/ASC/MARX/inbrief/news.html>

⁶⁴<http://space.mit.edu/ASC/MARX/inbrief/install.html>

```
export SHELL=`which csh`  
idl
```

The *csh* shells spawned by IDL should inherit your unix path (\$PATH) from *bash*. However, you will have to establish other *csh* configurations (described below) via the file `~/.cshrc`, which *csh/tcsh* executes when it starts.

2. **Your Unix account must be configured such that within the spawned shells the alias *ciao* will configure *CIAO*, and the alias *heasoft* will configure *HEASOFT* and *XSPEC*.** The *CIAO* and *HEASOFT* documentation describe how to set up those packages. For example the configuration file (`~/.cshrc`) for the shell *tcsh* that I use contains the following lines:

```
alias ciao      'source /usr/astro/cxc/bin/ciao.csh -o'  
alias heasoft   'source $HEADAS/headas-init.csh'
```

If you use the “conda” package manager to install CIAO, then you would (I think) need a *ciao* alias of the form:

```
alias ciao      'conda activate ciao-4.12'
```

If the aliases *ciao* and *heasoft* are available in your normal interactive shells, but AE reports that those aliases are not defined, then you need to look for the reason that those alias statements in your shell configuration file would not be executed in the non-interactive shells that *IDL* spawns. First, confirm that *IDL* is spawning the shell that you expect:

```
spawn, 'echo $SHELL'
```

Second, note that shells often look for multiple configuration files at start-up, some of which are skipped for non-interactive shells. For example, a **non-interactive** *zsh* shell will execute the configuration file `.zshenv` but will skip `.zprofile`, `.zlogin`, and `.zshrc`. Finally, your shell configuration file could contain a branch (or bug) of some kind which skips the definition of the *CIAO* and *HEASOFT* aliases for non-interactive shells.

3. *MARX* requires that you set the environment variable `MARX_DATA_DIR` to `{marx_installation_directory}/share/marx/data/`.
4. The shells spawned by AE must not produce any output to stdout when they start up, e.g. your `~/.cshrc` must not print anything. AE often spawns *CIAO* commands that write to stdout (e.g. *pget* and *dmstat*) and then parses the lines returned by the spawn. If the shell start-up has produced output then the text AE is looking for will appear on the wrong line. You can use the environment variable `FAST_START` to skip over parts of your `~/.cshrc` file that produce output, as shown in the next section (§6.2).
5. On some operating systems the standard environment variable `TMPDIR` is set to a **unique** scratch directory in every shell. Unfortunately, this causes problems for the *xpa* package which AE uses to control the DS9 tool that it uses in some stages. You can determine if you suffer from this problem by examining the value of `TMPDIR` in two shells:

```
echo $TMPDIR
```

If the value is different in the two shells, then you must set `TMPDIR` yourself in your shell’s configuration file, e.g.:

```
setenv TMPDIR /tmp/
```

6.2 Tips on Minimizing Execution Times

6.2.1 Overhead When Launching Data Processing Tools

AE tools must spawn a large number of *HEASOFT* (*FTOOLS*), *CIAO*, and *MARX* commands. Those commands may launch faster if *CIAO*, *CALDB*, and *MARX* are installed on a local disk rather than on a remote server.

Each time a data processing command is run, a Unix shell must be created to manage that command. **On some computer systems, creating those Unix shells requires a significant amount of time (e.g. >1 second per shell created), significantly slowing down the data processing.** At start-up, *AE* will measure and report how long it takes to spawn a shell. If that delay is longer than 0.2 second, a warning message is printed.

Usually, shells are slow to launch because they are configured to set up software packages that have complex initialization scripts, such as *HEASOFT* and *CIAO*. *AE* provides a mechanism for your `~/.cshrc` file to skip slow package initializations in shells created by *AE* but retain those package initializations in normal interactive shells. In shells that *AE* spawns, the environment variable **FAST_START** is defined. Your `~/.cshrc` file can skip over slow package initializations by testing for that variable, e.g.

```
if ($?FAST_START == 0) source /usr/astro/cxc/bin/ciao.csh -o
if ($?FAST_START == 0) source $HEADAS/headas-init.csh
...
```

Experimentation may be required to determine which parts of your `~/.cshrc` file are slow to execute (and thus ripe for skipping in *AE*-spawned shells). During that experimentation, use the following IDL program to report the launch time of your shell.

```
idl
.run ae
.run
run_command
while 1 do begin
    t0=systime(1) & run_command, 'exit', /QUIET
    run_time = systime(1)-t0
    print, run_time, F='("Spawning a shell is taking %0.2f seconds.")'
    wait,5
endwhile
end
```

As of November 2007 the *scisoft* package seems to have a bug that requires the use of **FAST_START** to ensure that its configuration script is not run by every spawned shell.

6.2.2 Parallel Processing

When your project involves several ACIS observations a lot of time can be saved by running the single-ObsID parts of *AE* in separate IDL sessions, in parallel. See §A. For best performance, the CPU's of the machine hosting the filesystem storing your project should be fully loaded before moving any *AE* runs to remote machines.

6.2.3 Filesystem Performance

Keep in mind that most *CIAO* processing is “I/O limited”, rather than “CPU limited”. For best performance, the CPU's of the machine hosting the filesystem storing your project should be fully loaded before moving any *AE* runs to remote machines. The two computations that are CPU intensive are:

- Computing PSF images:
This is done by the first run of the `ae.make_catalog` tool for each ObsID, which is calling the `CONSTRUCT_REGIONS` stage of *AE*.
- Computing RMF files:
This is done by the first run of the `ae.standard.extraction` tool for each ObsID, which is calling the `EXTRACT_SPECTRA` stage of *AE*.

MERGE stages have tons of I/O and will definitely run faster on the machine hosting the filesystem.

The most effective way to increase the speed of any source extraction task is to work on “flash storage” rather than on a spinny disk drive.

Similarly, the “tmp” directories that AE builds should be on “flash storage” rather than on a spinny disk drive.

In anticipation that some observers will process data stored on an NFS file system remote from the machine running *IDL*, AE has been designed to make heavy use of a “scratch directory”, presumed to be on a fast storage device physically attached to the computer running *IDL*. This scratch directory is identified by calling the standard *IDL* function *filepath(/TMP)*. Normally, this returns the standard “temporary directory” defined by the operating system, which is often */tmp/*. *IDL* will however allow you to change the standard temporary directory via the standard shell environment variable *IDL_TMPDIR*, or via the *IDL* preference⁶⁵ *IDL_TMPDIR*.

6.2.4 NFS Performance Issues

CIAO commands used to suffer (and may still suffer) significantly degraded performance when writing to directories accessed via the Network File System (NFS) protocol. The slowdown arises from the interaction between *CIAO* and NFS, not from the speed of the network. The slowdown occurs even when the storage device is physically local, but is accessed via an NFS path. For example, if your target can be accessed by both an NFS path (e.g. */bulk/cochise.fast/targets/mytarget/*) and via a local path (e.g. */Volumes/cochise.fast/targets/mytarget/*) then **all AE/CIAO processing should be launched from shells that are “cd” to your target using the local path.**

6.2.5 Filesystem Congestion from Other Software

In the course of an AE extraction a large number of files (both permanent and temporary) are created. Anti-virus software, Apple’s Spotlight system, and many backup systems will try to immediately examine every new file AE creates. Such software can greatly increase the I/O load on the storage device, can consume significant CPU resources, and can significantly slow down data analysis.

You may be able to configure anti-virus software to ignore your target directory and AE’s scratch directory, or to ignore the set of filenames that AE produces. You can definitely configure Apple’s Spotlight system to ignore your target directory, either via the Privacy pane in Spotlight’s preferences or by adding the suffix “*.noindex*” to the name of the directory holding your target.

Backup systems are more complex to manage, because you must balance your desire for robust and prompt backup for your work against your desire for fast data processing. I can offer no general solution to backup that is both safe and fast.

6.2.6 Algorithmic Shortcuts

AE includes a number of undocumented capabilities that speed up processing for atypical projects. For example,

- For projects where a significant fraction of the sources proposed by your detection process will be repositioned or pruned from the catalog during the course of the AE processing, AE has optional mechanisms to postpone the expensive computation of RMFs and PSFs until the catalog is finalized.
- In wide-field survey projects most sources are not observed by any given ObsId. AE has optional mechanisms for efficiently skipping over those sources.

If execution time is a significant concern for your large project, contact Patrick Broos to discuss undocumented options that may be useful.

⁶⁵ The *IDL* manual describes several ways *IDL* preferences can be changed.

7 Using *ACIS Extract*

7.1 Recipes

AE has been used to reduce ACIS data since early 2002. Over that period it has grown to address a broad range of reduction tasks for both point and diffuse sources. These data processing capabilities have intentionally been divided into “stages” to offer the observer the flexibility to handle both simple and very complex reduction situations.

An overview of how these stages can be used for point source and diffuse source extraction is given in §7.1.1 and §7.1.2. Detailed descriptions of how each stage is called are given in §E.0.1–7.11. If care is taken, it is possible to run multiple instances of AE simultaneously; see Appendix A.

7.1.1 Point Sources

An overview of the standard workflow for extracting a set of point sources from multiple observations is shown in Figure 8.

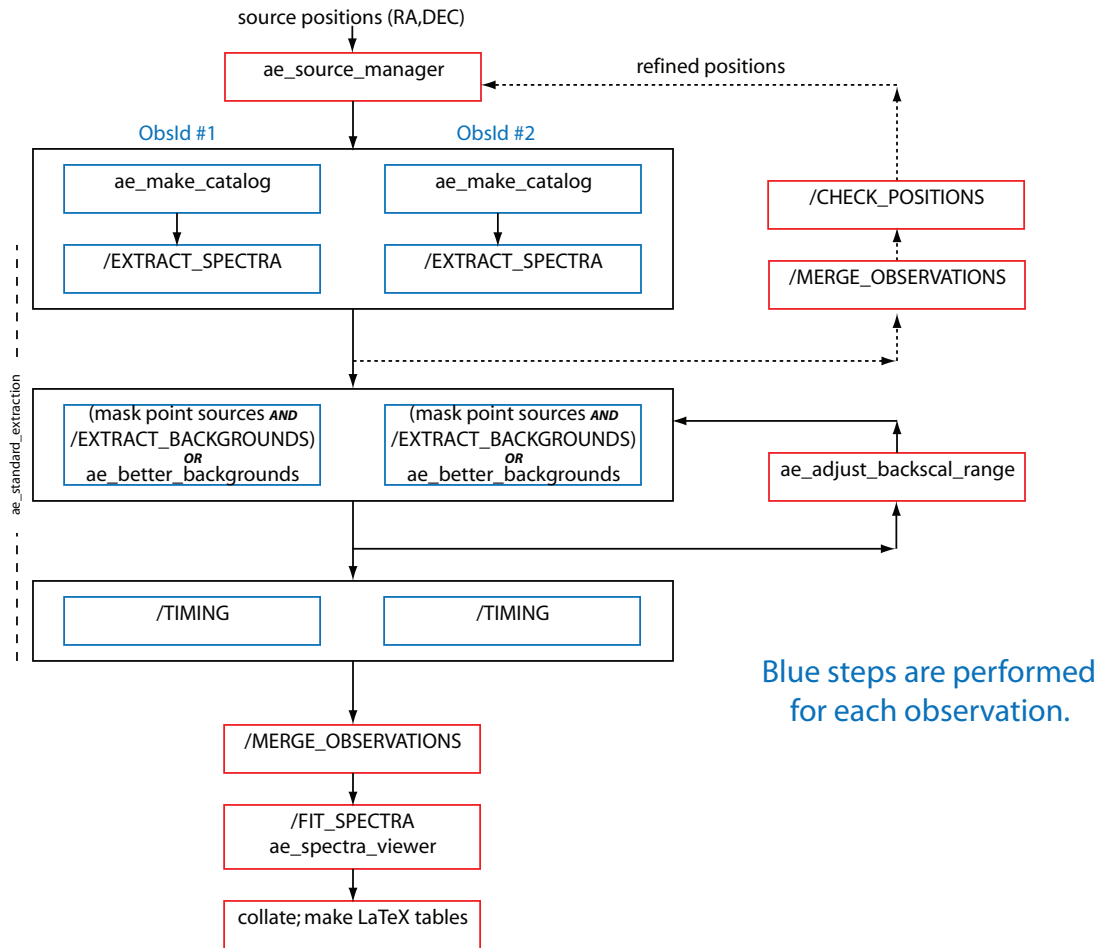


Figure 8: Overview of the Standard AE Workflow

1. Initial positions for the sources are supplied to the *ae_source_manager* tool, which creates an extraction directory for each source and maintains a list of the source names in the ASCII file *all.srclist*. This tool can be used later to add, re-position, or remove sources if necessary (§7.2).
2. For each ObsId, non-overlapping extraction regions are constructed by the *ae_make_catalog* tool (§7.5).

3. For each ObsId, source and background spectra are extracted, and light curves are constructed by the *ae_standard_extraction* tool (§7.6). Three algorithms for background extraction are provided.
4. The allowed range of background scaling (§5.6) for each source is adjusted, using the tool *ae_adjust_backscal_range* (§7.7), and the background spectra are re-extracted. This scaling adjustment and re-extraction cycle is repeated as needed. **Note that *ae_adjust_backscal_range* is run once per background extraction cycle, NOT run for each ObsId!**
5. The single-ObsId data products are combined (“merged”), and multi-ObsId photometry and spectra are produced using the MERGE_OBSERVATIONS stage (§7.8).
6. The multi-ObsId spectra are used to derive spectral models using the FIT_SPECTRA stage (§7.11).
7. All data products, including the spectral model chosen by the observer, are collated into a large FITS table using the COLLATION stage (§7.12).

Note that the tools *ae_make_catalog* and *ae_standard_extraction* require standardized file paths to various input data products, as shown in Appendix D.

As you would expect, the workflow must become iterative (i.e. include jumps back to previous steps), and thus more complex, if the observer chooses to **modify** the source catalog during the recipe. Here are some examples of catalog modifications that might be desired.

- The observer may wish to assign revised positions for sources, derived either by eye or via the CHECK_POSITIONS stage (the iterative workflow depicted with dotted lines in Figure 8).
- During visual review the observer may notice new sources that must be added using the *ae_source_manager* tool (§7.2).
- Invalid sources that must be moved or removed may be identified either by visual review or by examination of source significance statistics computed by AE.
- During visual review the observer may decide that the sizes of some extraction regions should be modified.

The SHOW stage (§7.10) provides easy visualization of the extractions for individual sources. Potentially helpful plots of various source properties across the whole catalog are available through the /PLOT options to the CONSTRUCT_REGIONS, EXTRACT_SPECTRA, EXTRACT_BACKGROUNDS, and MERGE_OBSERVATIONS stages. The COLLATE stage constructs a *DS9* region file showing a typical extraction region for each source.

My own personal recipe for analyzing ACIS point sources observed with multiple overlapping ACIS exposures is divided into two parts. A multi-pass “validation procedure”⁶⁶ starts with a set of source candidates that are assumed to contain some false detections and are assumed to have sub-optimal position estimates. This complex procedure repeatedly extracts the candidate catalog, prunes candidates found to be not significant, repositions the candidates using AE’s source position estimates, and helps the observer look and deal with various damaging instrumental effects. A more straightforward one-pass “photometry procedure”⁶⁷ performs the final extraction of a validated catalog. Our procedure for “automated fitting of point source spectra”⁶⁸ is also available; it may be overly-complex for most projects, but can serve as an example to inspire your own spectral fitting process. In these procedures, standard names are used for all files; thus you can execute most of the procedures by cutting and pasting.

Pile-up A draft recipe for extracting piled up sources⁶⁹ is also available. The strategy is to perform a series of annular extractions and spectral fits with increasingly large fractions of the PSF core excluded, and to then manually examine how fit parameters change as the size of the excluded core varies. One then tries to choose the extraction with the smallest excluded core (largest signal) where the fit parameters have “stopped changing”.

⁶⁶<http://personal.psu.edu/psb6/TARA/procedures/>

⁶⁷<http://personal.psu.edu/psb6/TARA/procedures/>

⁶⁸<http://personal.psu.edu/psb6/TARA/procedures/>

⁶⁹<http://personal.psu.edu/psb6/TARA/procedures/>

Stacking Given a set of sources on which an AE extraction has been performed, it’s easy to “stack” them together to produce a multi-ObsId spectrum, ARF, RMF, and photometric quantities. You simply have to “trick” AE into thinking that the N “observation” sub-directories you have for the several sources you wish to stack are really N observations of a single source, e.g. one named “stacked”. Then you can run the MERGE-OBSERVATIONS stage on the “stacked” source to get a combined spectrum, background, ARF, RMF, and photometric quantities.

Avoiding photometric bias is very important for stacking work, so we omit the /MERGE_FOR_PHOTOMETRY option (§5.13). That leaves the *observer* responsible for deciding whether any high-backgrounds extractions (e.g. off-axis) should be omitted!

An example of simple shell commands and the AE call to do this is below. Note that back-quote characters are used in several places in the shell commands below.

```
# Create a directory in which to perform your stacking analyses.
mkdir stacking
cd    stacking

# Create a symbolic link to the AE point source extraction directory.
ln -s /foo/bar/data/extract/point_sources point_sources

# Create a subdirectory for a group of sources you wish to stack, and move there.
mkdir weak_sources
cd    weak_sources

# By whatever means you choose, list the names of sources you wish to stack in a sourcelist file.
# For example:
idl
  bt = mrdfits('.././point_sources/tables/photometry.collated',1)
  include_in_stack = ( ... )
  forprint, TEXTOUT='stacked.srclist', bt.OBJECT, /NoCOMMENT, SUBSET=where(include_in_stack)

# Using symbolic links, make all the extraction directories to be stacked appear in the stacking directory.
foreach src (`egrep -v "^;" stacked.srclist`)
  echo $src
  foreach spectrum_file (`cd ../point_sources/$src; echo */source.pi`)
    set obs=`dirname $spectrum_file`
    ln -s ../point_sources/${src}/${obs} ${src}_${obs}
  end
end

# AE expects to find the file "stacked/source.stats" so just copy it from one of your sources.
cp ../point_sources/${src}/source.stats .

# Back in the stacking directory, run the MERGE stage and group the spectrum.
echo weak_sources > temp.srclist
cd ..
idl |& tee -a weak_sources/stacked.log
  .run
  acis_extract, 'temp.srclist', /MERGE_OBSERVATIONS, /SKIP_PSF, /SKIP_NEIGHBORHOOD, /SKIP_TIMING

  acis_extract, 'temp.srclist', COLLATED_FILENAME='weak_sources/weak_sources.collated'

  acis_extract, 'temp.srclist', /FIT_SPECTRA, /GROUP_WITHOUT_BACKGROUND,$
    CHANNEL_RANGE=[35,548], SNR_RANGE=[5,30]', $
    MODEL_FILENAME='point_sources/xspec_scripts/plot_gross_and_background_spectra.xcm'
```

```
exit
end
```

```
egrep -i "WARNING|ERROR|halted" weak_sources/stacked.log | \
egrep -v "DISPLAY variable|arithmetic error|no in-band|No HDUNAME|fitsio|FILTER"
```

Note that in the MERGE.OBSERVATIONS stage the source spectra are added; the ARFs are averaged (weighted by EXPOSURE); and the EXPOSURE keywords are added (§5.9). For example if you stack 8 sources with 10 counts each from the same 20ks observation then the summed spectrum will have 80 counts, the EXPOSURE keyword will be 160ks, and the multi-ObsId ARF will be the simple average of the 8 ARFs. The flux you derive from fitting with the merged files is the (average) flux **per source**.

7.1.2 Diffuse Sources

Our personal “recipe” for diffuse source extraction is (so far) straightforward since the issues that can make point source extraction complex—adjustments to the source list, positions, and PSF fractions—don’t exist. The process is outlined below, and described in more detail in our “diffuse procedure”⁷⁰.

1. By whatever clever means you can find, construct region files that define one or more diffuse “objects”. The regions must be in celestial coordinates if you have multiple observations, but can be in “physical” (*Chandra* SKY) coordinates if you have only one observation. When possible, construct an additional region that samples the astrophysical X-ray background, which we’ll call a “sky” region. Each region file can contain multiple or compound components if desired, e.g. a polygon minus a circle. All the region files must be in “ds9 format”, NOT “CIAO format” and must have names that end in “.reg”.
2. We recommend removing all identifiable point sources from the event data and exposure map that is used for diffuse analysis⁷¹. Such point source masking is easily done with tools designed for constructing background spectra for point sources (§7.6.2, 7.6.3).
3. Construct an ASCII file with two columns to serve as an AE “diffuse catalog” that lists ALL the regions you need to extract (both astrophysical objects and off-target “sky background” regions):
 - (a) The first column should be a name for the source (no spaces are allowed).
 - (b) The second column should be the path to the region file, in *DS9* (not *CIAO*) format, that defines the source.

For example, this simple C-shell script will build such a catalog for all the “.reg” files in the current directory, using the filename as the name of the source:

```
rm diffuse_all.srclist
foreach file (*_diffuse.reg)
    echo "`basename $file .reg` $file" >> diffuse_all.srclist
end
```

4. **For each observation**, run the AE stage CONSTRUCT_REGIONS (§E.1)—with the /DIFFUSE option—on the full source list, i.e. on all the “object” and “sky” regions. (Omit the point source related keywords EMAP_FILENAME, ASPHIST_DIR, MASK_FRACTION, MASK_MULTIPLIER, and /REGION_ONLY.) AE passes each region file through *DS9* to convert it to the sky coordinate system for that observation, and saves the region in the source directory tree in *CIAO* format.
5. **For each observation**, run the AE stage EXTRACT_SPECTRA (§E.3) on the full source list. Spectra are extracted and weighted ARFs and RMFs are constructed; an aspect file should be supplied to *mkacisrmf* via the ASPECT_FN parameter. Diffuse ARFs are renormalized by AE so that the calibration of the extraction is expressed in surface brightness (rather than flux) units (§5.14). **The event energy range used to construct the detector map passed to mkwarf is currently fixed at 500 to 2000 eV (suitable for soft diffuse emission studies) and the WMAP binning is fixed to 1x1 in the DET coordinate system.**

For regions that cross chip boundaries you may see *mkwarf* produce error messages of the form “Couldn’t determine chip position for pixel: ...” which is explained in a CIAO FAQ⁷². You may find RMF generation for diffuse sources takes a very long time if the source spans many WMAP pixels.

At this point you will have spectra and response files for all of the diffuse and sky regions (if available) for each observation. Diffuse sources are often heavily contaminated by background—both instrumental background and emission from foreground and background astrophysical sources that are not of interest. We are aware of three reasonable strategies to account for background in diffuse sources:

⁷⁰<http://personal.psu.edu/psb6/TARA/procedures/>

⁷¹ Theoretically, point sources could instead be eliminated by defining “compound” diffuse regions, say a polygon which *includes* a diffuse region and several small circles inside the polygon which *exclude* point sources.

⁷²<http://cxc.cfa.harvard.edu/ciao/faq/mkwarf-skippix.html>

Stowed Data with Sky Region One can directly subtract **instrumental** background from both the diffuse region and the sky region, for each ObsID, by running AE’s EXTRACT_BACKGROUND stage on suitably prepared “stowed event data” provided in CALDB (Hickox & Markevitch, 2006, §4.1.3). Our personal AE diffuse recipes (§7.1) uses this approach.

All the extractions of each region (diffuse object and sky region) are then merged by AE in the usual way to produce standard “source” and “background” spectra for the diffuse and sky regions. These two net spectra are then **simultaneously** fit using a shared model for the **X-ray** background in both regions plus a model for the diffuse emission of interest in one region.

Stowed Data without Sky Region If no appropriate sky region is available, then one can directly subtract **instrumental** background from the diffuse region, using the “stowed event data” provided in CALDB, as in #1 above. All the extractions of the diffuse object are then merged by AE in the usual way to produce standard “source” and “background” spectra for the diffuse object. Then, since no sky region is available, you must explicitly model each of the astrophysical components of the X-ray background, without the aid of a “sky” observation. External information such as space weather data or models of background components derived from other observations may be useful (Snowden et al., 2008).

Subtract Sky Region One can scale and **subtract** the observed sky spectrum from the observed diffuse object spectrum, as is commonly done with point sources, under the assumption that the sky spectrum adequately represents both the instrumental and astrophysical components of the background contaminating the aperture of the diffuse object. After the diffuse object and sky region have been extracted with AE (EXTRACT_SPECTRA stage), you can simply modify the files in the sky extractions so that they look like the files produced by AE’s EXTRACT_BACKGROUND stage. Within each extraction directory (ObsID) that was created for the **sky** region, the modifications required (at the Unix shell) are shown below.

```
mv source.pi background.pi
foreach kywd (ONTIME LIVETIME EXPOSURE)
  dmhedit background.pi filelist=none operation=add key=$kywd value=0
end
dmlist background.pi head,clean | egrep "ONTIME|LIVETIME|EXPOSURE" | grep -v HISTORY
```

Then, within each extraction directory (ObsID) that was created for the **diffuse** region, you create a symbolic link (at the Unix shell) that points to the corresponding sky background spectra created above. For example:

```
ln -s ../../sky_region/1875/background.pi diffuse_region/1875/
```

Then, all the extractions (ObsID’s) of the diffuse object (which now have background.pi symbolic links) can be merged by AE in the usual way and the resulting multi-ObsId source and background spectra can be fit in the usual way.

When the sky and diffuse sources regions are far from each other on the detector, the fundamental problem with this approach is that it is difficult to scale the sky spectrum such that it faithfully represents both the instrumental and astrophysical (derived from X-rays coming through the mirrors) components of the background found in the source region.

The scaling that AE applies to the sky spectrum will properly scale the astrophysical background at the energy used to build the exposure maps, since integrals of those maps define the scaling. However, to the degree that the effective area curves (ARFs) in the two regions have different **shapes**, the scaling of the astrophysical background at other energies will not be correct. With some effort, the observer could use the ratio of the two ARFs to derive an **energy-dependent** scaling for the (sky) background spectrum (represented as a BACKSCAL or AREASCAL column, rather than a BACKSCAL or AREASCAL keyword)⁷³. Such a scaling should be able to properly scale the portion of the background spectrum that is astrophysical (derived from X-rays coming through the mirrors).

⁷³ Prior to February, 2009 AE offered this sort of energy-dependent background scaling when a ARF file (background.arf) for the sky spectrum was supplied to the MERGE stage, but that capability disappeared when AE abandoned energy-dependent background scaling for point sources (§5.6.3).

However, whether the background scaling is a single number or an energy-dependent vector, the potential flaw in this approach is that instrumental components of the background do not arise from X-rays passing through the mirrors. The spatial variation of the instrumental background is not described by the ARF or by its cousin the exposure map; any scaling derived from them may not properly scale the instrumental background.

7.1.3 *XMM* Point Sources

We have completed initial development of a work flow for analyzing point sources observed with the EPIC instrument on *XMM*. *XMM*-versions of the single-ObsId stages of AE have been implemented in a tool called *epic_extract*. The extraction data products produced for the MOS1, MOS2, and PN detectors can then be merged and fit with the usual MERGE and FIT_SPECTRA stages in AE. This approach of merging MOS and PN spectra is probably not optimal for high-signal sources. An industrious *XMM* observer could use the MERGE stage (§7.8) to produce a MOS1+MOS2 spectrum, use the MERGE stage to produce a PN spectrum, and then write a script that simultaneously fits the two.

A recipe for a simple analysis of *XMM* point sources⁷⁴ is available. There is no position adjustment or fancy backgrounds, but it's probably good enough for many projects. This *XMM* work flow happily lets SAS do much of the heavy lifting that *ACIS Extract* had to take on for ACIS data (e.g. estimating source positions, constructing extraction and background regions, masking sources for background extractions, computing background scaling, computing ARFs for sources that span multiple CCDs, correcting the ARF for PSF fraction).

As of April 2008 the *XMM* work flow does not support multiple *XMM* pointings, however with simple modifications this can be accomplished.

⁷⁴<http://personal.psu.edu/psb6/TARA/procedures/>

7.2 Source Management with the *ae_source_manager* Tool

The overall process of identifying and extracting sources can become complex if your data are complex and you take a very careful approach. One fundamental bookkeeping problem is that the catalog may need to change over time. At multiple points in the data reduction process you may identify candidate sources that you wish to remove, you may identify new sources you wish to add, and you may choose to reposition existing sources, e.g. using position estimates computed by AE (§7.9). Usually, when a source’s position is changed you will want to rename it, a task that involves changing both the catalog and the name of the source’s extraction directory.

The tool *ae_source_manager* is used to create, remove, and re-position sources. It edits a master source list (stored in the file *all.srclist*) and manipulates AE source extraction directories. The ways it can be used are described below. All the inputs can be either scalars or vectors to manipulate one or multiple sources.

- To create a source, use `/ADD` and provide the coordinates.
`ae_source_manager, /ADD, RA=ra, DEC=dec, POSITION_TYPE='wavdetect'`
 - The required keyword `POSITION_TYPE` should be a string (or string vector) describing the method used to estimate the source position. This information becomes the `POSNTYPE` source property (in *source.stats*).
 - The optional keyword `PROVENANCE` can be used to provide a string (or string vector) describing how each source was detected, e.g. “wavdetect”, “reconstruction”, “by eye”, “counterpart”, etc. This information becomes the `PROVENAN` source property (in *source.stats*).
 - The optional keyword `LABEL` can be used to provide a string that will be used to label the source in *DS9* region files. If omitted the sequence number of the source is used.

An AE extraction directory is created for the source in order to store the information provided.

- To remove a source, use `/REMOVE` and identify the source either by its name or label.
`ae_source_manager, /REMOVE, NAME='104418.06-593059.4'`
`ae_source_manager, /REMOVE, LABEL='154'`
 - The optional keyword `TRASH_DIR` can be used to specify a directory to which the obsolete AE extraction directories will be archived. If omitted the directory `./trash` will be used.

Entries for obsolete sources are commented out in (not removed from) the master source list (*all.srclist*).

- To re-position a source, use `/MOVE`, identify the source either by its name or label, and provide either the new RA and DEC coordinates, or provide offsets (in sky-pixels) from the existing coordinates.
`ae_source_manager, /MOVE, POSITION_TYPE='by_eye', RA=ra, DEC=dec, NAME='104418.06-593059.4'`
`ae_source_manager, /MOVE, POSITION_TYPE='by_eye', RA=ra, DEC=dec, LABEL='154'`
 - The required keyword `POSITION_TYPE` should be a string (or string vector) describing the method used to estimate the revised source position; supplying the null string will cause any existing `POSITION_TYPE` to be retained.
 - Normally the source is renamed using the new coordinates, both in the master source list and in the extraction directory. If `/NO_RENAME` is specified the original name is retained.
- To re-position a source to either the “mean data” or “correlation” or “reconstruction” positions computed by AE (§7.9), use `/UPDATE_POSITIONS_DATA` or `/UPDATE_POSITIONS_CORR` or `/UPDATE_POSITIONS_RECON` and identify the source either by its name or label.
`ae_source_manager, /UPDATE_POSITIONS_DATA, NAME='104418.06-593059.4'`
`ae_source_manager, /UPDATE_POSITIONS_CORR, LABEL='154'`
 - The `POSNTYPE` source property is updated to reflect the type of position adopted for the source. If the position estimates you wish to apply were computed on a **named** merge (§7.8), then be sure to supply that merge name via the `MERGE_NAME` parameter.
- To sort the source list by RA and relabel by sequence number, use `/SORT_RA` and `/SET_LABEL_AS_SEQUENCE`.
`ae_source_manager, /SORT_RA`
`ae_source_manager, /SET_LABEL_AS_SEQUENCE`

- To sort the source list by distance from the middle of the field and relabel by sequence number, use `/SORT_BULLSEYE` and `/SET_LABEL_AS_SEQUENCE`. This labeling may be more convenient for locating specific sources in *DS9* during the data reduction process.

```
ae_source_manager, /SORT_BULLSEYE
ae_source_manager, /SET_LABEL_AS_SEQUENCE
```

When sources are added, removed, or re-positioned the most straightforward reprocessing strategy is to simply repeat your entire extraction recipe. Several important algorithms require knowledge of all sources in the catalog, including the de-crowding algorithm in the tool *ae_make_catalog* (§7.5) and the background algorithms (§7.6.1).

7.3 General Parameters

Most AE tools and stages described in the sections below explicitly or implicitly use the master source list (*all.srclist*) maintained by *ae_source_manager*. Several other parameters that are explicitly or implicitly used by these tools are described below.

obsname: a string that identifies an observation (usually an ObsId e.g. “1875”); it will name a sub-directory created for each source.

EXTRACTION_NAME an optional string or string array that identifies a set of extraction parameters. If supplied, the AE output files are structured with an extra directory level using the specified name(s). For example, if the observer wished to perform the extraction twice, with two different sets of PSF fractions, then EXTRACTION_NAME could be used to put the output from the two extractions in separate directories. Similarly, data products produced by merging multiple extractions can be named via the MERGE_NAME parameter to the MERGE_OBSERVATIONS Stage (§7.8).

obsdata_filename: the file name of the single-observation event list you wish to use for extraction. This manual will not attempt to discuss the various recipes observers use to prepare ACIS data for spectral extraction. **The only event list columns used by AE are TIME, CCD_ID, CHIPX, CHIPY, DETX, DETY, X, Y, PI, ENERGY, and STATUS; discarding the other columns will reduce AE run times.** We recommend that the event list should NOT be strongly filtered by energy; the data can be more correctly band-limited later during spectral analysis. The event data must be in the tangent plane coordinate system of the observation, not “reprojected” to some other observation (§7.13).

ACIS event lists contain a boolean flag column named STATUS that *CIAO* uses to record various data cleaning criteria that are commonly used to remove events likely to arise from the instrumental background. Two of those cleaning criteria—“afterglow” identification and “5x5 background” identification—produce false positives for sources with high count rates. Thus, low-rate sources are optimally extracted by applying these cleaning criteria and high-rate sources are optimally extracted by ignoring these cleaning criteria (Broos et al., 2010).

AE supports the optimal extraction of both groups of sources by applying a STATUS=0 filter to the source and background extractions of low-rate sources, and ignoring the STATUS column for high-rate sources. To take advantage of this capability, the observer should clean Level 2 event data using only the “safe” STATUS bits—those not associated with the “afterglow” and “5x5 background” criteria.

emap_filename: the file name of a single-observation exposure map that is aligned with the event data, that represents any masking done to the event data, and that has a field of view corresponding to the event data you’re extracting. The CXC provides threads and scripts for constructing exposure maps. We use the tool *ae_make_emap* (§7.14). It’s unclear whether the energy band used in computing the exposure map has any significant effect on the results. **AE requires that the exposure map has units of $s \times cm^2 \times count/photon$ (i.e. the `normalize=“no”` option was supplied to *mkexpmap*).** Note that the exposure maps supplied to the EXTRACT_SPECTRA and EXTRACT_BACKGROUND stages of AE are integrated over the source and background extraction regions in order to derive appropriate scaling for the background spectrum. The integrals alone determine the scaling; the EXPOSURE keywords in the two event lists are ignored.

There is no specific binning requirement for the exposure map, however keep in mind that both point source masks and background regions in AE are ultimately defined as regions corresponding to sets of exposure map

pixels rather than as geometric regions (e.g. *DS9* region files). We strongly recommend an exposure map pixel size of 1 sky pixel or smaller, particularly if the tools *ae_better_masking* (§7.6.3) or *ae_better_backgrounds* (§7.6.1) are to be used.

We recommend that you examine your exposure maps to ensure that they are reasonably cropped. Very large borders of zeros around your exposure maps will needlessly slow the execution of many data analysis steps.

7.4 Default Parameters

Several AE stages use optional parameters which have default values. Our choice of default values may not be appropriate for your study. The observer is especially encouraged to review the defaults used for these parameters:

- MIN_NUM_CTS (§E.4, 7.6.1, 7.6.2, 7.6.3, E.5, E.6, 7.7, 7.8).
- CHANNEL_RANGE, CSTAT, SNR_RANGE=snr_range, NUM_GROUPS_RANGE (§7.11).

7.5 The *ae_make_catalog* Tool

The *ae_make_catalog* tool is run for each observation, as shown in Figure 8. It uses the source information stored in AE extraction directories (by *ae_source_manager*) to iteratively define non-overlapping extraction apertures for a single ObsID.⁷⁵ This tool requires you to have set up a set of standardized paths to the input files needed by AE, as shown in the Getting Started section (3). The tool requires two parameters:

```
ae_make_catalog, '1875', EVTFILE\_BASENAME='spectral.evt'
```

- The tool uses the observation name (obsname) parameter to identify the directory `../obsXXXX/` where it expects to find various observation data products (with the standard names used by our AE recipe).
- The EVTFILE.BASENAME parameter specifies the filename for the event list (in `../obsXXXX/`) from which you wish to extract spectra.
- The optional keyword parameter SRCLIST.FILENAME can be used to specify a source list other than the default, *all.srclist*.
- The optional keyword parameter NOMINAL_PSF_FRACTION can be used to specify a nominal PSF fraction for isolated sources other than the default, 0.90.
- The optional keyword parameter MINIMUM_PSF_FRACTION can be used to specify a minimum PSF fraction other than the default, 0.40.
- The optional keyword parameter REGION_ONLY can be used to specify that PSFs should be constructed for only the nominal energy, saving computer time.
- If the optional keyword parameter SHOW is set to zero, then the *DS9* visualization normally spawned will be skipped.
- Any other keyword parameters supplied are passed on to the CONSTRUCT_REGIONS stage of AE.

Three files pertaining to the whole source list are written to the observation directory (e.g. `../obs1875/`):

obs.cat An AE catalog containing the sources observed in this ObsId, with the PSF fractions adjusted to avoid overlapping extraction regions.

extract.reg A *DS9* region file showing the positions and extraction regions for all the sources.

obs.collated A FITS table containing preliminary source properties (§7.12) for the sources observed in this ObsId.

Since extraction regions have a minimum size (controlled by parameter NOMINAL_PSF_FRAC) very crowded sources will have overlapping regions. A measure of each source region's overlap with its neighbor is saved in the source property OVERLAP in obs.stats.

⁷⁵ The AE stages CONSTRUCT_REGIONS (§E.1) and EXTRACT_EVENTS (§E.2) are executed multiple times as extraction aperture sizes are iteratively adjusted.

7.6 The *ae_standard_extraction* Tool

The *ae_standard_extraction* tool is run for each observation, as shown in Figure 8, to “extract” data corresponding to each of the apertures constructed by *ae_make_catalog*. A spectrum and light curve are constructed from the events within the aperture. A set of nearby events is carefully chosen to form an estimate of the background spectrum that contaminates the aperture; three strategies for choosing a sample of background events are described in the following sections.

7.6.1 Model-based Background Regions

When your sources are crowded, the background in an extraction aperture has multiple components. First, it has the usual *flat component*, i.e. instrumental and sky background that has a relatively flat spatial distribution. Any background region that is nearby should, after appropriate scaling, model this component. However the crowded source will, by definition, also have significant background components that arise from the PSF wings of one or multiple neighboring sources. Even a distant source can produce a background component if it has a strong ACIS readout streak.

We desire a background spectrum that models all these background components. The tool *ae_better_backgrounds* (§E.6) seeks to construct a single background region plus a background scaling that simultaneously models all the background components. The background region is constructed iteratively. At each iteration a number of possible expansions of the region are judged and the best one is adopted. This style of background extraction is selected by supplying the parameter BETTER_BACKGROUND=1 to *ae_standard_extraction*.

Judging the quality of a proposed background region consists of evaluating how well it is expected to model each of the background components corrupting our source. For the flat component, one preference we should have is that the region be reasonably compact, i.e. local to the source. In this tool we define a compactness metric which we seek to minimize. We can also judge a background region with respect to each background component that arises from the PSF wing of a neighboring source by considering its expected bias, i.e. the difference between the power we expect to find from that neighbor in our extraction and background regions (after scaling). An ideal background region would have zero bias for each neighbor, i.e. it would sample just enough power from each neighbor to account for the contamination from that neighbor. In this tool we compute these biases by integrating spatial models of each neighbor over the two regions. These models are simply PSF images, including any bright ACIS readout streaks, scaled using rough photometry provided by a previous AE extraction.

The goal of zero expected bias for **each** neighbor is useful because neighbors may have very different spectra. However, perhaps the most important goal is to have zero overall bias in the background subtraction, which one might refer to as avoiding “photometric bias”. We attempt to achieve this by careful selection of the scaling that is applied to the background region (see Appendix C).

The compactness metric and background component biases are combined to form a scalar metric which drives the iterative construction of a background region. The algorithm for choosing a background region consists of the following “search phases”.

1. The bkg region will grow until two conditions are met:
 - The region contains at least MIN_NUM_CTS in-band counts, AND
 - $BKSCL_{LO} = BACKSCAL$ ($BKSCL_{LO}$ is stored in source.stats)

The MIN_NUM_CTS requirement is to avoid background estimates that have huge statistical uncertainty, which inflates Pb and can cause a source to be pruned. The pathological case is a source with an uncrowded on-axis observation and a crowded far off-axis observation. The off-axis observation needs to keep BACKSCAL small in order to fairly sample his crowded neighbor, but that small BACKSCAL leaves the on-axis observation with very few (perhaps zero) counts. If the MERGE then chooses to ignore the off-axis observation, then we’re left with a composite extraction with a very uncertain background estimate, and an inflated Pb value. Aargh!

After phase #1, the algorithm will keep track of which bkg region, dubbed our “reserve region”, achieved the best (smallest) background imbalance metric.

2. After #1 is achieved, we add ONE more pixel to the bkg region before considering any stopping criteria. This tweak is designed to address a special case—where $BKSCL_{LO} = BKSCL_{GL}$, and the current extraction is responsible for that value (via a VOTE_LO from the previous run of this tool). In such a situation, we want to

proceed past the last acceptable region to determine if our previous VOTE_LO still applies. If we simply stop at BKSC_LO=BKSC_GL, and cast no VOTE_LO, then another ObsID can drag the scaling range upward; on the next pass this ObsID will again find that range to be unacceptable, and will recast a VOTE_LO. This leads to a never-ending cycle: - this ObsID casts a VOTE_LO - the range is adjusted to satisfy that vote - this ObsID abstains from voting - the range is adjusted upward, beyond what this ObsID can tolerate - this ObsID casts a VOTE_LO

3. After #2 is achieved, growth of the bkg region will continue until BKSC_GL j= BACKSCAL (the region reaches the goal specified for all ObsIDs, BKSC_GL, which is stored in source.stats).
4. After #3 is achieved, growth of the bkg region will stop when
 - If the background imbalance metric is less than BACKGROUND_IMBALANCE_THRESHOLD, then accept the bkg region, OR
 - When the region reaches the maximum allowed size (BKSC_HI j= BACKSCAL), move to phase #5.
5. Since condition 4B stopped the search (no acceptable region was found), we rebuild and then adopt the reserve region.

The best way to review the background regions constructed by the better backgrounds algorithm is the SHOW stage of AE (§7.10), where the background region (which is a set of pixels, not any sort of geometric region that *DS9* can understand) is depicted by blue plus signs. Also, the BETTER_BACKGROUND code prints a little table for each source showing how many counts from each neighbor are expected in the source aperture and in the scaled background region.

For some sources, the background region turns out to be very small, containing very few (even zero) counts. A typical source suffering this problem has a weak neighbor very nearby. The algorithm attempts to build a region that is expected to sample an appropriate amount of power from this neighbor, but quickly “runs out of signal” when the region has completely covered the neighbor. Adding further area to the region would simply dilute the power from this neighbor, taking the background spectrum even further from being a fair sample of all the background components. Essentially, the algorithm is faced with the impossible task of satisfying two competing goals:

1. Build a region that, after scaling, is a good model of the contamination in our aperture arising from that close neighbor’s wings.
2. Build a large region that has plenty of counts to get a precise estimate of the flat component of the background (i.e. instrumental background and the smooth X-ray sky).

We strongly recommend an exposure map pixel size of 1 sky pixel or smaller when using this algorithm.

The right way to deal with crowding is to simultaneously fit spatio-spectral models to groups of sources that interact, along with some kind of flat background spectrum thought to be free of point source power, calculating during the fitting process how much power from each source spills over into the other source apertures. *Sherpa* can probably do this, but that’s way beyond my abilities.

7.6.2 Circular Background Regions with Circular Masking

If your source list has no crowding—no source suffers significant background from the wings of a neighbor—then you may ask *ae_standard_extraction* to generate local background spectra extracted from simple circular regions around each extraction aperture (§E.5). The *ae_standard_extraction* tool “removes” all the point sources from the event list before performing this background extraction, using circular masks around each source that are sized to exclude virtually all the point source events. The *ae_standard_extraction* tool applies the same masks to the exposure map that is used in the background extraction, so that the “area” of the background region is accurately calculated. This style of background extraction is selected by supplying the parameter BETTER_BACKGROUND=0 to *ae_standard_extraction*.

The CONSTRUCT_REGIONS Stage (§E.1) creates a region file for each extraction that contains a generous circular mask region. The *ae_standard_extraction* tool gathers these circular regions to produce a mask region file for the entire catalog.

```
printf "# Region file format: \DSnine\ version 3.0 \nfield()\n" > ../obs1875/mask.reg
grep -h background */1875/extract.reg | awk '{print "-", $1}' >> ../obs1875/mask.reg
```

It then applies the mask to the exposure map.

```
dmcopy "../obs1875/obs.emap[sky=region(../obs1875/mask.reg)][opt full,update=no]" \
../obs1875/background.emap
```

Events that lie within exposure map pixels that are zero are then discarded.

```
dmimgpick "../obs1875/spectral.evt[cols time,ccd_id,chip,det,sky,pi,energy]" \
../obs1875/background.emap temp.evt method=closest
dmcopy "temp.evt[#8>1]" ../obs1875/background.evt
```

7.6.3 Circular Background Regions with Model-based Masking

The simple masking described in §7.6.2 is not optimal since masks for weak sources are excessively large, and the masks are circular rather than shaped like the PSF. An observer concerned about preserving as much diffuse emission as possible, or concerned about keeping source background spectra as “local” as possible can construct a far better masked background using the source models described in §7.6.1. The *ae_standard_extraction* tool can then extract that masked background data within circular background regions (§E.5).

The masking has to be done explicitly by the observer, using the *ae_better_masking* tool, which is described below and is shown in our AE diffuse recipe (§7.1). The *ae_standard_extraction* tool should then be called with the parameters REUSE_MASKING=1 and BETTER_BACKGROUNDS=0.

The goal of the *ae_better_masking* tool is to use large masks for bright sources and small masks for dim sources. The general method is to estimate, at each pixel location in the emap, the number of counts expected from the point sources in the catalog (saved as *star_counts.img*) and the number of background counts expected (saved as *bkg_counts.img*). Pixels are masked when $(star_counts/bkg_counts) > THRESHOLD$. There is no statistical theory behind this algorithm; it just seems like a reasonable approach to us.

If you have additional masking you want to do you can supply your own region file (in celestial coordinates) via the keyword EXTRA_MASKFILE. The remaining unmasked pixels are sorted by *star_counts*, and then processed in order of brightness. For each pixel, a local background is estimated (excluding of course any pixels already masked), the $(star_counts/bkg_counts)$ ratio is computed, and the pixel is masked if necessary.

Note that both *star_counts* and *bkg_counts* are computed only under the footprints of source PSFs. Outside of those, *star_counts* is assumed to be zero, and *bkg_counts* is irrelevant. If the resulting background event list is later used for extraction of diffuse emission, the masked *star_counts* image provides at least some estimate of the contamination of the diffuse spectra arising from the incomplete masking of known point sources.

We recommend an exposure map pixel size of 1 sky pixel or smaller if *ae_better_masking* is to be used. Note however that *ae_better_masking* may require considerable time to execute for exposure maps with many pixels.

Usage and Inputs This tool requires the standardized directory structure and file naming convention used in §3. This tool must estimate source photometry; thus **prior to running this tool you must extract source spectral and rough background spectra** (e.g. backgrounds derived from simple mask regions as described in the previous section).

```
ae_better_masking, obsname, EVTFILE_BASENAME=name, EMAP_BASENAME=name,
EXTRACTION_NAME=name, MIN_NUM_CTS=value, EXTRA_MASKFILE=filepath,
THRESHOLD=value, /REUSE_MODELS, /SKIP_EXTRACT_BACKGROUNDS
```

EXAMPLE:

```
idl |& tee ae_better_masking.1875.log
```

```
ae_better_masking, '1875', EVTFILE_BASENAME="validation.evt"
```

This tool builds models for all the sources in the catalog using single-observation photometry calculated by the MERGE_OBSERVATIONS stage of AE, then constructs a mask for each source.

- The tool uses the observation name (obsname) parameter to construct a directory path `../obsXXXX/` where it expects to find various observation data products (with the standard names used by our AE recipe).
- The required `EVTFILE_BASENAME` parameter specifies the filename for the event list (in `../obsXXXX/`) from which you wish to extract background spectra.
- By default the input exposure map is read from `../obsXXX/obs.emap`; you can specify a different file name in `../obsXXX/` via the optional `EMAP_BASENAME` parameter. .
- The optional `EXTRACTION_NAME` parameter is described in §7.3.
- The optional `MIN_NUM_CTS` parameter specifies the number of counts required under the adaptive smoothing kernel used in the algorithm; the default value is 100.
- The optional `EXTRA_MASKFILE` parameter specifies a region file designating additional masking regions that should be applied.
- The optional `THRESHOLD` parameter changes the threshold applied to the ratio (*star_counts/bkg_counts*) as described above; the default value is 0.5.
- The optional keywords `/REUSE_MODELS` and `/SKIP_EXTRACT_BACKGROUNDS` can be supplied to direct the tool to read in some data products that were computed and saved in an earlier run of the tool, as a way to save time when the tool has to be re-run for some reason. Obviously, this sort of re-use of data products is appropriate only if the catalog and source photometry is not expected to have changed since the earlier run..
- Normally, after constructing a masked exposure map and event list the tool will pass those to the `EXTRACT_BACKGROUNDS` stage of AE to compute local background spectra for each point source. The optional keyword `/SKIP_EXTRACT_BACKGROUNDS` can be supplied to skip this AE call.

Data Products The following files are produced:

- `../obs{obsname}/polymask.reg`: a region file containing all the extraction polygons.
- `../obs{obsname}/polymask.img`: a copy of the exposure map with the extraction polygons masked.
- `../obs{obsname}/star_flux.img`: a FITS image showing a model of the flux from all the point sources.
- `../obs{obsname}/star_counts.img`: a FITS image showing the estimated star counts remaining after masking.
- `../obs{obsname}/bkg_counts.img`: a FITS image showing the background estimated by the algorithm. Background values are computed only where the source models is non-zero.
- `../obs{obsname}/background.emap`: the final masked exposure map.
- `../obs{obsname}/background.evt`: the final masked event list.

We strongly recommend an exposure map pixel size of 1 sky pixel or smaller when using this tool.

7.7 Adjust BACKSCAL Stage

This stage is run **once** as shown in Figure 8, after extractions have been performed for each observation. Even if you have only one observation you will run this stage.

Prior to the July, 2009 release of AE, the sizes of background regions (and thus their scalings) were chosen independently for each ObsID that observed the source. However, as described in §5.6, we have come to understand that theoretical problems arise when trying to combine multiple extractions which use very different background scalings (BACKSCAL values). Starting with the July, 2009 release, a target background scaling range is chosen for each source, and the EXTRACT_BACKGROUND Stage (§E.5) and Better Backgrounds tool (§E.6) produce background regions whose scaling is limited to this target range. This target range is stored in the FITS keywords BKSCALLO, BKSCALGL, and BKSCALHI in the file source.stats.

This extraction architecture required the introduction to the AE workflow of a new tool, *ae_adjust_backscal_range*, that analyzes the source’s existing background extractions and then chooses a desirable target background scaling range individually for each source. These two inter-dependent operations—extraction of backgrounds that meet a scaling target and adjustment of that target—have to be repeated until the scaling target is reasonably stable. See the Getting Started section (§3) and our point source extraction recipe (§7.1).

Adjustment of the background scaling target is a compromise between several competing goals:

1. To encourage photometric accuracy, we would like the uncertainty in the estimate of net counts to be dominated by the uncertainty in the extracted source counts (SRC_CNTS). Specifically, we try to get enough background counts so that the ratio between the photometry error term associated with SRC_CNTS and the photometry error term associated with the scaled background subtraction (§5.10) is at least 4.0, which forces photometry errors to remain within 3% of the values they would have if there was zero uncertainty in the background. As SRC_CNTS decreases, its error term also decreases (in absolute terms); a corresponding reduction in the background subtraction error term can be achieved only by observing **more** background counts over a larger background region. Thus, weak sources will generally require larger (more accurate) backgrounds than strong ones. Intuitively, accurate background estimation is most important for weak sources.
2. The number of counts in the background region is used to compute the PROB_NO_SOURCE statistic (§5.10.3) Weisskopf et al. (2007, Appendix A2), which some observers use to prune their source catalog. Obviously, Poisson uncertainty in the background estimate induces a corresponding uncertainty in the PROB_NO_SOURCE statistic. Although a confidence interval for PROB_NO_SOURCE could be estimated, at present it’s not obvious how to choose a reasonable goal for that uncertainty. Instead, we simply adopt a reasonable goal for the minimum number of observed counts in the multi-ObsId (merged) background region; see MIN_NUM_CTS below.
3. The fidelity of single-ObsId background spectra constructed with the “Better Backgrounds” algorithm (§E.6) will typically suffer if the size of the background region (i.e. the background scaling) is forced to grow beyond a certain size. The algorithm records, separately for each extraction of a source, a “vote” for the desirable upper limit to the background scaling; the *ae_adjust_backscal_range* tool then tries to choose a target scaling that accommodates these votes.

7.7.1 Usage and Inputs

```
.run ae
ae_adjust_backscal_range, OVERLAP_LIMIT=value, MIN_NUM_CTS=value
```

EXAMPLE:

```
idl |& tee ae_adjust_backscal_range.log
.run ae
```

```
ae_adjust_backscal_range, MIN_NUM_CTS=100
```

- The optional parameter MIN_NUM_CTS (default 100) specifies the minimum number of counts you desire in the merged (multi-ObsId) background spectrum.

- The optional parameter `OVERLAP_LIMIT` has the same meaning as in the `MERGE_OBSERVATIONS` Stage (§7.8).

7.8 MERGE_OBSERVATIONS Stage

This stage is run once as shown in Figure 8, after extractions have been performed for each observation. Even if you have only one observation you will run MERGE_OBSERVATIONS.

7.8.1 Actions

- For each PSF energy, a multi-ObsId PSF is constructed as an average of the PSFs from all the observations being merged, weighted by the exposure map values in each observation. Each multi-ObsId PSFs is normalized so that it sums to 1.0.
- The “neighborhood” event lists (neighborhood.evt) for each observation are reprojected and merged to form a multi-ObsId neighborhood event list.
- The extracted source event lists (source.evt) for each observation are reprojected and merged to form a multi-ObsId source event list.
- The spectral extraction data products (source and background spectra, ARFs, RMFs) for all the observations of a source are combined into a single set of products ready for spectral fitting.
- Time variability of the source both within and among observations is quantified by comparing a uniform **flux** model to the distribution of time stamps for extracted events, and computing the 1-sample Kolmogorov–Smirnov statistic between the flux model and the observed distribution. The p-value of this multi-ObsId KS statistic is saved in the keyword MERGE_KS in source.stats; values near 0 suggest variability.

BEWARE: This K-S test currently makes no distinction between events produced by the source and background events. Variation in the aperture background rate among ObsIDs can lead to a spurious indication of source variability. Such background variation can occur because the aperture size can vary among ObsIDs (e.g. due to variations in off-axis angle), because background from neighboring sources can vary among ObsIDs, and because FI and BI devices have different levels of instrumental background.

This roughly-calibrated metric depends on the shape of the Effective Area curve, which varies between grating configurations, varies between ACIS detectors, and varies over time (§2.3).

- Time variability of the source among observations is quantified by calculating the p-value of the χ^2 statistic of the single-ObsID (background-subtracted) photon fluxes, under the null assumption that the flux is constant (at the weighted mean value). This p-value is saved in the keyword MERG_CHI in source.stats; values near 0 suggest variability.

BEWARE: This roughly-calibrated metric depends on the shape of the Effective Area curve, which varies between grating configurations, varies between ACIS detectors, and varies over time (§2.3).

- Two PostScript plots are produced showing light curves, produced by the TIMING stage (§E.4), from multiple observations—one uses a separate set of axes for each observation (Figure 9), the other shows all observations on a single “broken” time axis (Figure 10). Examination of the flux and median energy time-series can reveal source spectral changes during a flare.

BEWARE: these plots currently make no distinction between events produced by the source and background events. Variation in the aperture background rate among ObsIDs can lead to a spurious indication of source variability. Such background variation can occur because the aperture size can vary among ObsIDs (e.g. due to variations in off-axis angle), because background from neighboring sources can vary among ObsIDs, and because FI and BI devices have different levels of instrumental background.

- Wide-band photometry corrected for background is performed, as described in §5.10.

- A 2-sample Kolmogorov–Smirnov⁷⁶ statistic is computed to compare the source and background spectra, and the significance of that statistic is reported in the keyword KS_SPECT in source.stats. It is hoped that this statistic will characterize whether a candidate source is similar (significance ~ 1) or distinct (significance ~ 0) spectrally from its local background. Note that for some sources the source and background spectra may have very different numbers of points. For sources with enough counts, this may help to distinguish true point sources from clumpy diffuse emission.
- A number of observation-specific keywords from the obs.stats files are combined/propagated to the source.stats file:

NUM_OBS: number of observations extracted

OVRLP_LM: overlap limit used for discarding observations

OVRLP_LO: smallest overlap fraction from all observations

OVRLP_HI: largest overlap fraction from all observations

BESTOBS: ObsID corresponding to OVRLP_LO

WORSTOBS: ObsID corresponding to OVRLP_HI

MERGNUM: number of observations merged

MERGFRAC: fraction of extracted data merged

OBSNAME: list of the OBSNAME values for merged observations

PRIM_OBS: name of the deepest merged observation

THETA: a weighted average of the off-axis angles of the source in merged observations

THETA_LO: the smallest off-axis angle among merged observations

THETA_HI: the smallest off-axis angle among merged observations

SRC_AREA: a weighted average of the SRC_AREA keyword (source area) from merged observations

SRC_RAD: a weighted average of the SRC_RAD keyword (source radius) from merged observations

PSF_FRAC: a weighted average of the PSF_FRAC keyword (PSF fraction) from merged observations

MSK_RAD: largest value of the MSK_RAD keyword (mask radius) from merged observations

FRACSPEC: smallest target PSF fraction for merged observations

SRC_CNTS: total in-band counts extracted from merged observations

EMAP_TOT: sum of EMAP_AVG values for merged observations

FRACEXPO: fraction of time that the source position (not the whole PSF footprint) spent on live portions of the detector, derived from FRACEXPO in the *CIAO* ARF files

WARNFRAC: fraction of events in the “warning regions” of merged observations

EXPOSURE: total exposure in merged observations

EFFAREA: ARF value at nominal energy (1.49 keV)

SCAL_MAX: maximum value of BACKSCAL column in source spectrum

SCAL_MIN: minimum value of BACKSCAL column in source spectrum

BACKGRND: background intensity ($photons/pixel^2/cm^2/sec$), total-band, merged observations

PROB_KS: smallest single-ObsId KS variability statistic among merged observations

- A “mean data” source position estimate (§5.3) is computed as the mean position of the band-limited multi-ObsId event list, and is saved as keywords RA_DATA, DEC_DATA in source.stats. Position uncertainties for each axis are saved as keywords ERX_DATA and ERY_DATA.
- A “1 deviation root mean square” (dRMS or 1DRMS) error radius (§5.3) for the position estimate (RA_DATA, DEC_DATA) is computed and saved in the keyword ERR_DATA in units of arcseconds (NOT sky pixels).

⁷⁶Refer to “Numerical Recipes” by Press et al., 2nd edition (1992), Chapter 14 for discussion of the 2-sample KS statistic.

- A multi-ObsId *DS9* region file is constructed showing the extraction polygons for all observations.
- Plots summarizing the results of MERGE_OBSERVATIONS are presented. These plots are diagnostic tools that help you spot problems in your data, in your catalog, in your *CIAO* configuration, in the *CIAO* software, or in the AE software. Some plots may also convey scientifically interesting information about your sources.

7.8.2 Usage and Inputs

```
acis.extract, srclist_filename, /MERGE_OBSERVATIONS, MERGE_NAME=name, EBAND_LO=energy,
EBAND_HI=energy, OVERLAP_LIMIT=value, THETA_RANGE=[value,value],
/MERGE_FOR_PB, /MERGE_FOR_POSITION, /MERGE_FOR_PHOTOMETRY,
MIN_QUALITY=value
```

```
acis.extract, srclist_filename, /MERGE_OBSERVATIONS, /PLOT, CARTOON_TEMPLATE=cartoon_template
```

EXAMPLE:

```
idl |& tee MERGE_OBSERVATIONS.log
```

```
eband_lo = [0.5, 0.5, 1.0, 2.0, 4.0, 6.0]
```

```
eband_hi = [8.0, 1.0, 2.0, 4.0, 6.0, 8.0]
```

```
acis.extract, 'all.srclist', /MERGE_OBSERVATIONS, EBAND_LO=eband_lo, EBAND_HI=eband_hi
```

```
acis.extract, 'all.srclist', /MERGE_OBSERVATIONS, /PLOT, CARTOON_TEMPLATE='1875.emap'
```

- In situations where you wish to construct photometry, spectra, responses, etc. on a **subset** of the observations you've processed, you can supply an optional *obsname* parameter (§7.3) after the *srclist_filename* parameter. The parameter should be a vector of observation names, e.g. ['6420', '6421', '6403', '8460', '8461']; wildcards are permitted in these observation names. This unusual mode of execution is not shown in the example above to avoid confusion.
- The optional parameter *MERGE_NAME* allows the observer to name the merge session; the stage's output files will be structured with an extra directory level using the supplied name, separating those files from the results of other merge sessions. This parameter is analogous to the optional *EXTRACTION_NAME* parameter (§7.3) which can be used to retain multiple named extractions.
- *EBAND_LO*, *EBAND_HI* (optional) should be vectors defining the desired energy bands (in keV) over which photometry will be performed. If omitted these bands are used:
0.5 : 8.0
0.5 : 2.0 2.0 : 8.0
0.5 : 1.7 1.7 : 2.8 2.8 : 8.0
0.5 : 1.5 1.5 : 2.5 2.5 : 8.0
0.5 : 1.0 1.0 : 2.0 2.0 : 4.0 4.0 : 6.0 6.0 : 8.0
0.5 : 7.0 2.0 : 7.0
9.0 : 12.0
- The optional parameter *OVERLAP_LIMIT* (range [0,1], default 0.1) specifies the threshold applied against each extraction's *OVERLAP* property (assigned by the tool *ae_make_catalog*; §7.5) to decide if that extraction should be discarded, i.e. not included in the merge (§5.13). If **all** extractions have excessive *OVERLAP*, then the overlap limit for that source is relaxed enough to allow at least one extraction to be merged, and a warning is printed. The source properties *OVRLP_LO* and *OVRLP_HI* in *source.stats* record the *OVERLAP* range of the extractions merged.
- The optional parameter *THETA_RANGE* (a two-element vector) specifies that the ObsIDs merged must lie within a range of off-axis angle (in arcminutes).

- If `/MERGE_FOR_PB` is specified, then the set of extractions merged will be chosen to optimize source validity (§5.13, §5.10.3 Weisskopf et al. (2007, Appendix A2). The `MIN_NUM_CTS` parameter can be used to impose a minimum requirement on `SRC_CNTS` in the merge, for example to avoid asserting the validity of a source with only two counts (see our AE recipes (§7.1) for example usage).
- If `/MERGE_FOR_POSITION` is specified, then the set of extractions merged will be chosen to optimize the accuracy of the source position estimate (§5.13).
- If `/MERGE_FOR_PHOTOMETRY` is specified, then the set of extractions merged will be chosen to optimize source signal-to-noise ratio. The `MIN_QUALITY` parameter must be supplied to specify the minimum acceptable ratio between the signal-to-noise ratio achieved by the merge and the optimal signal-to-noise ratio achievable by discarding more extractions (§5.13). For example, specifying `MIN_QUALITY=0.50` means that you want AE to discard observations, risking photometric bias, only when necessary to prevent the signal-to-noise ratio of the merge from dropping below one-half of its optimal value.
- The optional keyword parameter `CARTOON_TEMPLATE` should be the name of a FITS image that will be used to define the “scene” (image dimensions, pixelization, and placement on the sky) used to build two cartoon images (`cartoon_acis.img`, `cartoon_sky.img`) of the point sources. The first cartoon uses the ACIS PSF scaled by the flux in the first photometry band; the second uses a Gaussian PSF of fixed size (currently hard-coded to `FWHM=3` image pixels).

7.8.3 Data Products

For each source the following files are produced:

{sourcename}/source.stats: a FITS file containing keyword information about the source

{sourcename}/source.psf: a FITS file containing a multi-ObsId PSF image

{sourcename}/neighborhood.evt: a FITS event list covering the immediate neighborhood of the source

{sourcename}/source.evt: a FITS table containing the events extracted for the source

{sourcename}/{sourcename}.pi: the multi-ObsId source spectrum (not grouped)

{sourcename}/{sourcename}_bkg.pi: the multi-ObsId background spectrum (not grouped)

{sourcename}/{sourcename}.arf: the multi-ObsId ARF

{sourcename}/{sourcename}.rmf: the multi-ObsId RMF

{sourcename}/{sourcename}.sequenced_lc.ps: a PostScript plot of the light curves and median energy time series for each observation shown on a single “broken” time axis. The pair of blue curves depict a 68% pointwise confidence band⁷⁹ for the continuous lightcurve, estimated from the individual X-ray events using an adaptively-sized sliding window. Note that the confidence band is inherently smooth (due to the sliding window); variations in the confidence band are significant only if they are large compared to the width of the band.

{sourcename}/{sourcename}.stacked_lc.ps: a PostScript plot of the light curves and median energy time series for each observation shown on separate time axes.

{sourcename}/source.photometry: a FITS file containing a table of the photometry values described in §5.10

{sourcename}/extract.reg A *DS9* region file showing the extraction polygons for all observations.

cartoon.img: a FITS image containing a cartoon of the field

Several interactive plots show characteristics of the merged extraction; see our AE recipes (§7.1) for descriptions of each.

⁷⁹http://en.wikipedia.org/wiki/Confidence_and_prediction_bands

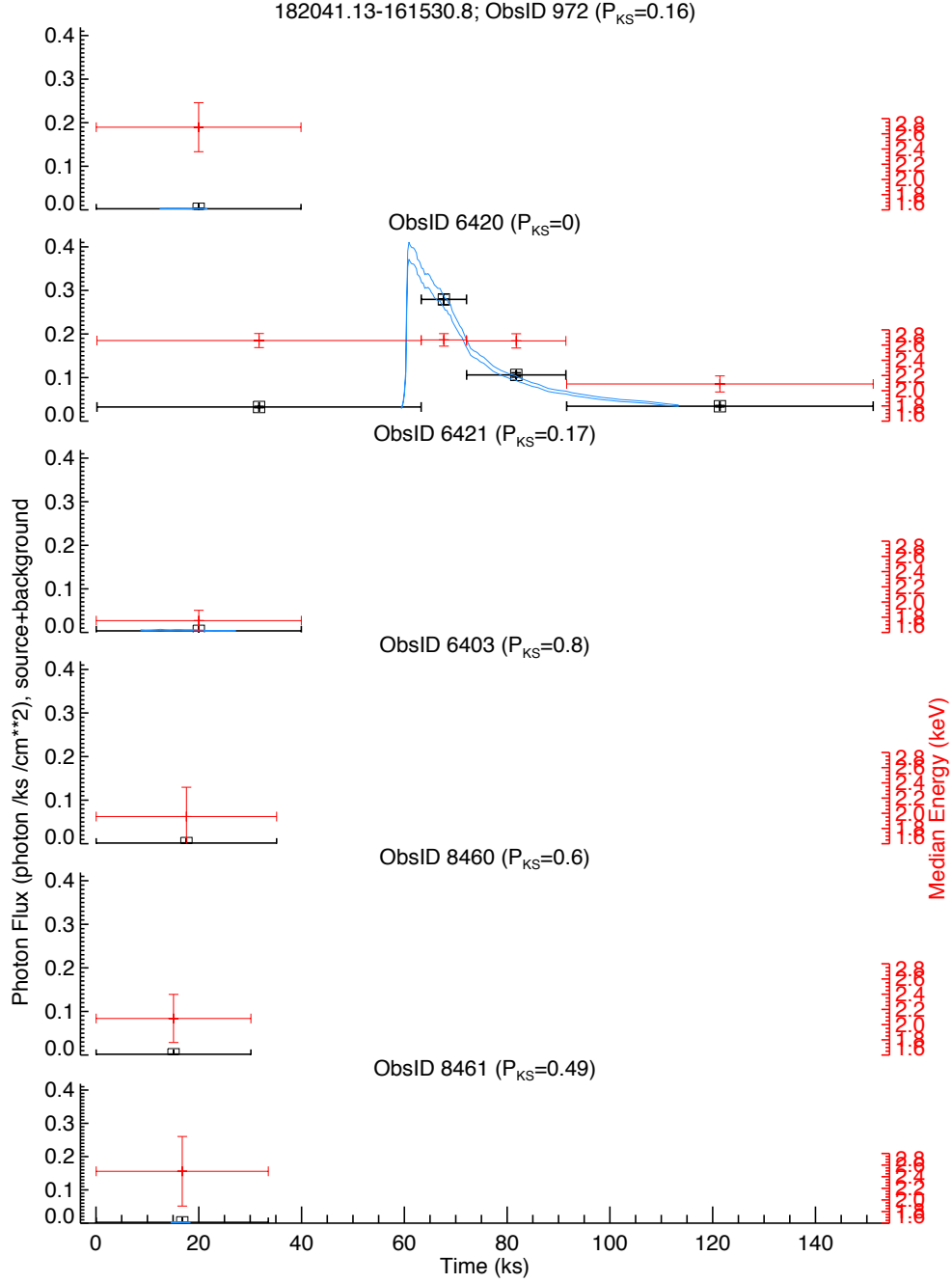


Figure 9: Flux and median energy time series for multiple observations of a source, shown on separate sets of axes. Three curves are shown. A binned light curve is shown in black, and the median energy of the events in the light curve bins is shown in red; the unequally sized time bins are represented by horizontal bars shaped like the letter “H”. Example Poisson errors within a single time bin are represented by black and red error bars along the Y-axis. The pair of blue curves depict a 68% pointwise confidence band⁷⁸ for the continuous lightcurve, estimated from the individual X-ray events using an adaptively-sized sliding window. Note that the confidence band is inherently smooth (due to the sliding window); variations in the confidence band are significant only if they are large compared to the width of the band.

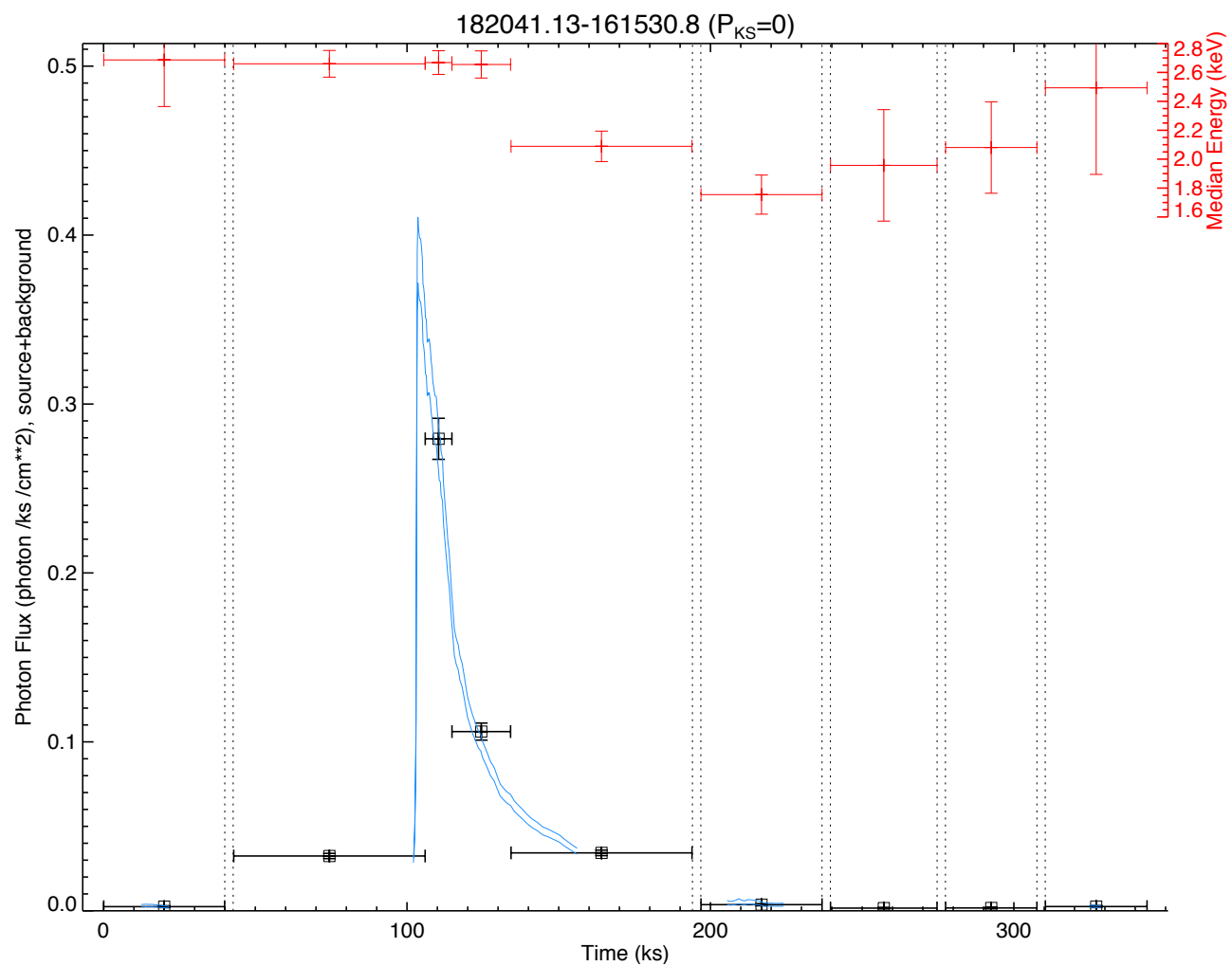


Figure 10: The same flux and median energy time series in Figure 9, shown on a single non-physical time axis. Time intervals between observations are indicated by black dotted vertical lines.

7.9 CHECK_POSITIONS Stage

This stage is run once, after trial extractions have been performed for each observation and the MERGE_OBSERVATIONS stage has been run, as shown in Figure 8. The source position estimates produced by this stage may be used to re-position the sources using the tool *ae_source_manager* (§7.2).

7.9.1 Actions

- A multi-ObsId data image of the source’s neighborhood is constructed by reprojecting the neighborhood events from each observation, filtering the events in the energy range specified, then binning them into an image using the same bin size as the multi-ObsId PSF.
- An estimate of the source position (§5.3) is obtained by correlating the multi-ObsId data image and the multi-ObsId PSF (§7.8), and is saved as keywords RA_CORR and DEC_CORR in source.stats.
- The multi-ObsId data image is reconstructed using a simple Maximum Likelihood algorithm (max_likelihood.pro in the *IDL* Astro Library).
- An estimate of the source position (§5.3) is obtained by centroiding a 3x3 pixel neighborhood around a nearby peak in the reconstructed image, and is saved as keywords RA_ML and DEC_ML in source.stats. Currently no uncertainty estimates are available for the correlation (RA_CORR, DEC_CORR) or Maximum Likelihood (RA_ML, DEC_ML) position estimates.
- An atlas showing the raw and reconstructed neighborhood images is produced.
- Plots summarizing the results of CHECK_POSITIONS are presented. These plots are diagnostic tools that help you spot problems in your data, in your catalog, in your *CIAO* configuration, in the *CIAO* software, or in the AE software. Some plots may also convey scientifically interesting information about your sources.

7.9.2 Usage and Inputs

```
acis_extract, srclist_filename, /CHECK_POSITIONS, ENERGY_RANGE=[energy,energy],  
MAXLIKELIHOOD_ITERATIONS=value, /SKIP_RECONSTRUCTION, /SKIP_CORRELATION,  
THETA_RANGE=[value,value],
```

```
acis_extract, srclist_filename, /CHECK_POSITIONS, /PLOT
```

EXAMPLE:

```
idl |& tee check_positions_1875.log
```

```
acis_extract, 'all.srclist', /CHECK_POSITIONS
```

```
acis_extract, 'all.srclist', /CHECK_POSITIONS, /PLOT
```

- The optional ENERGY_RANGE parameter defines the energy filter used in construction of the multi-ObsId data image; default is [0.5,8.0].
- The optional NOMINAL_PSF_ENERGY parameter selects which multi-ObsID PSF is used for correlation and image reconstruction; default is 1.5 keV.
- The optional THETA_RANGE parameter can be used to select a subset of sources for which this stage will be run, based on their off-axis angles.
- The optional keywords /SKIP_RECONSTRUCTION and /SKIP_CORRELATION can be supplied to speed up this stage by skipping the data image reconstructions and/or skipping the computation of a correlation position estimate.
- The optional keyword MAXLIKELIHOOD_ITERATIONS can be supplied to specify the number of reconstruction iterations; default is 400. If a vector of integers is supplied (e.g. [10,20,100,500]) then the reconstructed image is saved after each of the specified number of iterations.

7.9.3 Data Products

For each source the following files are produced:

{sourcename}/source.stats: a FITS file containing keyword information about the source

{sourcename}/neighborhood.img: the multi-ObsId (all observations) data image (primary HDU), and reconstructed image (first extension)

mugshots.ps: a PostScript atlas containing the raw and reconstructed neighborhood images printed 6 to a page. If you want to preview this atlas you may find that converting to PSF (distill) and viewing with Acroread gives a much better rendering than viewing the PostScript with gv or ghostview.

{sourcename}/evt.reg: a *DS9* region file containing a “point” region at the position of each event in neighborhood.img

{sourcename}/extract.reg: a *DS9* region file containing the source extraction polygons for all the observations, plus markers for the three estimates of the source position (catalog position, correlation peak, and mean of event data).

Several interactive plots show characteristics of the position estimates.

7.9.4 Choosing Among Position Estimates

The task of choosing, for each source, which of the available position estimates is best has no simple or obvious solution. The mean data (RA_DATA, DEC_DATA) and correlation (RA_CORR, DEC_CORR) position estimates can clearly be corrupted by a sloping background caused by the wings of a nearby source. We strongly suggest in our personal AE recipe (§7.1) that the observer visually review the position adjustments proposed by AE before adopting them.

Franz Bauer has investigated the various techniques for calculating source positions, using multiple data sets for the Chandra Deep Field North, which has good radio and optical source positions. Based on his work, the PSU ACIS group has decided to use centroid positions for sources < 5' off-axis and matched-filter (correlation) positions for sources > 5' off-axis.

For some crowded sources the observer may feel that the original position estimate is the most reliable one, and will want to avoid re-positioning those sources. The observer is encouraged to use the *PROVENANCE* source property (§7.2) to tag such sources so they can be excluded from re-positioning operations.

7.9.5 PSF and Neighborhood Image Binning

Note that the pixel size used in the multi-ObsId PSF image file and the neighborhood image file will often differ. The multi-ObsId PSF image is saved at high resolution (by MERGE_OBSERVATIONS), but is re-binned (locally in this stage) to a more appropriate resolution before constructing a matching neighborhood image and performing image reconstruction. If you wish to **manually** run a maximum likelihood reconstruction (outside of AE) then you may need to rebin the PSF to match an existing neighborhood image, as shown by the *IDL* code. You'll need a *TARA* release dated April 2007 or later to get the proper version (1.7 or higher) of the tool *maxlik.pro*.

```
; To test reconstruction set data_name=psf_name
psf_name  = 'source.psf'
data_name = 'neighborhood.img'

; Determine the ratio of PSF and data bin sizes.
psf =readfits(psf_name,  psf_hdr)
data=readfits(data_name, data_hdr)

rebin_ratio = sxpar(data_hdr,'CDEL11')/sxpar(psf_hdr,'CDEL11')
```

```
; If rebin ratio is not an integer then this approach won't work.
```



```

help, rebin_ratio

rebin_ratio = round(abs(rebin_ratio))

.run
if (rebin_ratio GT 1) then begin
    ; Rebin the PSF and save to a temporary file.
    print, 'rebinning the PSF'
    temp_name = 'temp.fits'
    cmd = string(psf_name, rebin_ratio, rebin_ratio, temp_name, F="('%dmcopy ""%s[bin #1:::%d,#2:::%d]"" %s
    print, cmd
    spawn, cmd
    psf_name = temp_name
endif
end

; Now we can do reconstructions:
Niter = 200
maxlik, data_name, psf_name, Niter, maxlik_img, maxlik_header, /plot

; The final iteration is returned in the \IDL\ array "maxlik_img".
writefits, 'maxlik200.fits', maxlik_img, maxlik_header

```

7.10 SHOW_REGIONS Stage

This stage is run as required to visually review one source at a time.

7.10.1 Actions

The tool has two modes of operation. In the **extraction review mode** (first example below, `DISPLAY_FILE` omitted) all AE extractions of a given source are shown simultaneously with each observation in a separate *DS9* frame. If a reconstructed neighborhood is available (from the `CHECK_POSITIONS` stage), it is displayed in an additional *DS9* frame. The source extraction and mask regions are overlaid, along with markers for every source in the catalog and optionally a set of user-supplied point regions.

In the **panning mode** (second example below, `DISPLAY_FILE` supplied) you supply one or more FITS files (images or event lists) and a corresponding set of region files that are passed to *DS9* for display. This tool then simply provides a convenient way method for panning to (moving to) each source in the catalog. For example, if you have multiple observations of your field this mode could be used to examine the sources in the merged event data.

In both modes you use simple commands displayed in the *IDL* window to navigate through the catalog. A specific source can be selected by entering any unique string found in its `LABEL`.

7.10.2 Usage and Inputs

```
acis_extract, COLLATED_FILENAME=filepath, /SHOW_REGIONS, SRCLIST_FILENAME=filepath,  
              REGION_FILE=filepath, ENERGY_RANGE=[energy,energy], /OMIT_BKG_REGIONS
```

```
acis_extract, COLLATED_FILENAME=filepath, /SHOW_REGIONS, SRCLIST_FILENAME=filepath,  
              REGION_FILE=filepath, DISPLAY_FILE=filepath
```

EXAMPLE:

```
idl
```

```
acis_extract, COLLATED_FILENAME='all.collated', /SHOW_REGIONS, DISPLAY_FILE='merged.evt',  
              REGION_FILE='all.reg'
```

```
acis_extract, COLLATED_FILENAME='all.collated', /SHOW_REGIONS
```

```
acis_extract, COLLATED_FILENAME='all.collated', /SHOW_REGIONS, SRCLIST_FILENAME='bright.srclist',  
              REGION_FILE='2mass.reg'
```

- The `COLLATED_FILENAME` parameter must be a FITS collated table produced by the `COLLATED_FILENAME` Stage (7.12).
- The optional `ENERGY_RANGE` parameter defines the energy range used to filter the event data displayed; default is [0.5,8.0].
- If you wish to display only a subset of the full catalog, or wish to display the catalog in a specific order, then list the source names to be displayed in an ASCII file and supply that file name via the optional `SRCLIST_FILENAME` parameter. The full source catalog should continue to be supplied via the `COLLATED_FILENAME` parameter so that all sources are marked in the *DS9* display.

For example, one could review the catalog sorted by decreasing brightness using *IDL* code like this:

```
; Collate the photometry previously computed.  
acis_extract, 'all.srclist', COLLATED_FILENAME='all.collated'  
  
; Read collated table and extract the NET_CNTS values for the first energy band.  
bt = mrdfits('all.collated', 1)  
  
band_full = 0
```

```

NET_CNTS = bt.NET_CNTS[band_full]
print, 'Using the energy band ', bt[0].ENERG_LO[band_full], bt[0].ENERG_HI[band_full]

; Write an ASCII source list that is sorted by NET_CNTS.
forprint, TEXTOUT='bright.srclist', bt.CATALOG_NAME, SUBSET=reverse(sort(NET_CNTS)), /NoCOMMENT

acis_extract, COLLATED_FILENAME='all.collated', /SHOW_REGIONS, SRCLIST_FILENAME='bright.srclist'

```

- The optional `REGION_FILE` parameter specifies the name of an observer-supplied region file (e.g. from a catalog in another wave band) that should also be displayed. Only “point” regions in that file are passed on to *DS9*, and celestial coordinates in decimal degrees must be used.
- If `/OMIT_BKG_REGIONS` is supplied then graphics depicting the background region built by the “Better Backgrounds” algorithm is not shown.

7.10.3 A Complementary Visualization Technique

The `SHOW_REGIONS` stage is designed to display extraction regions from a single-source, multiple-observation perspective. It’s also very helpful to see extraction regions for the whole catalog displayed on top of the data. Such a catalog-wide *DS9* region file can be easily constructed by the `COLLATED_FILENAME` Stage (§7.12). Visual examination of the catalog in this way will often reveal problems in the source list that must be corrected (§7.2), e.g. sources that were missed or poorly positioned by your source-finding algorithm.

7.11 FIT_SPECTRA Stage

This stage is run at least once after MERGE_OBSERVATIONS has been completed to perform automated spectra fitting using the *XSPEC*⁸⁰ spectral fitting package by Keith Arnaud. A suitable *XSPEC* script, meeting specific AE interface requirements described below, must be provided. Several such scripts, implementing simple *XSPEC* models, are distributed with AE (§5.12).

An excellent introduction to the concept of fitting X-ray spectra can be found in one of Keith Arnaud's PowerPoint presentations⁸¹.

7.11.1 Actions

- The multi-ObsId source spectrum is grouped as described in §5.7.
- For each source AE prepends to the specified *XSPEC* script several TCL assignment statements that pass source information to the script:

- `set spectrum_filename "104357.47-593251.3.pi"`
The spectrum to be fit (ungrouped for the C-statistic; grouped for χ^2).
- `set extra_spectrum_filename "104357.47-593251.3_grp5.0.pi"`
A second spectrum that should be plotted against the model. AE and AE-supplied fitting scripts use this mechanism to produce two plots for every source, a grouped spectrum (ldata.ps) and a cumulative ungrouped spectrum (icounts.ps), regardless of whether the fit statistic is χ^2 or C-statistic.
- `set ignore_spec "1-34,549-**"`
The channels that should be ignored in *spectrum_filename*, as specified by CHANNEL_RANGE.
- `set extra_ignore_spec "1,98"`
The channels that should be ignored in *extra_spectrum_filename*, as specified by CHANNEL_RANGE.
- `set model_name "nogrp_T:Solar,Solar"`
A name for the HDU that will contain the fit results in the file *source.spectra*.
- `set c_stat_flag 1`
A flag to specify whether the fitting statistic should be χ^2 or the C-statistic.
- `set spectrum_description "NET_CNTS=150-10.7"`
A descriptive string for use in plot titles.
- `set bkg_ignore_spec "1-7,686-**"`
The channels that should be ignored in the background spectrum when fit with a model (CSTAT option).
- `set cplinear_energies {0.5 1.0 1.5 2.0 3.0 4.0 5.0 6.0 7.0 8.0}`
The energies at which the vertices of the *cplinear* background model (C-statistic only) are placed (§5.12).
- `set model_directory "/users/research/patb/TARA/code/ae/xspec_scripts"`
The path to the fitting components of the local AE package.
- `set interactive_flag 0`
A flag that is set when /INTERACTIVE is specified.

- *XSPEC* is then started in the source directory and is passed the modified script. The screen output of *XSPEC* is directed to the file *xspec.run.log*. Since *XSPEC* sometimes hangs during a computation, AE will kill the *XSPEC* process if it consumes 600 seconds of CPU time, print a warning message, and move on to the next source. Examination of the *xspec.run.log* for such sources often reveals that the hang occurred during the *XSPEC error* command. The AE-supplied fitting scripts allow one to skip the parameter error computation by setting the TCL variable *skip_errors* to one.

When the χ^2 statistic is used, AE will skip any source which has < 4 groups (< 2 “noticed” groups) in its spectrum. Most fitting scripts will of course require more groups than that to function; it's the responsibility of the script to check that the number of noticed groups is adequate before attempting the fit.

⁸⁰<http://heasarc.gsfc.nasa.gov/docs/xanadu/xspec/>

⁸¹<http://lheawww.gsfc.nasa.gov/kaa/india2003/lowresspec.ppt>

Fitting scripts should assume that the multi-ObsId background spectrum, ARF, and RMF are listed in the standard FITS keywords BACKFILE, ANCRFILE and RESPFILE in the source spectrum.

- When *XSPEC* finishes, AE expects to find files named *ldata.ps* and *xspec.log* in the source directory. AE then runs a small L^AT_EX program which formats *ldata.ps* and *xspec.log* together in a PostScript summary of the fit, named *summary.ps*.
- AE archives the fitting script and the files produced during the fit in a sub-directory *spectral_models* in the source directory.
- The fitting script is expected to have written the fit parameters it determined as FITS keywords in an HDU named using the script variable *model_name* (see above) in the file *source.spectra*. If the observer runs the FIT_SPECTRA stage multiple times, with different models and/or differently grouped spectra, then the file *source.spectra* should contain multiple HDUs each named for the spectrum+model used. If the *source.spectra* files are so created, then the COLLATED_FILENAME stage may be run to collate across the catalog the fit parameters for a given spectrum+model, producing a FITS binary table with columns holding the fit parameters, plus the sourcename, RA, and DEC. Any “-” characters appearing in fit parameter names are changed to “_” in the column names to keep *IDL* happy.

7.11.2 Usage and Inputs

```
acis_extract, srclist_filename, /FIT_SPECTRA, CHANNEL_RANGE=[value,value], MODEL_FILENAME=filepath,
MODEL_CHANGES_FILENAME=filepath, /CSTAT, SNR_RANGE=[value,value],
NUM_GROUPS_RANGE=[value,value], /GROUP_WITHOUT_BACKGROUND, /INTERACTIVE,
FIT_TIMEOUT=value
```

EXAMPLE:

```
idl |& tee fit_spectra.log
```

```
acis_extract, 'all.srclist', /FIT_SPECTRA, CHANNEL_RANGE=[35,548],
MODEL_FILENAME='~/my_models/T:Solar,Solar.xcm'
```

EXAMPLE:

```
idl |& tee fit_spectra.log
```

```
acis_extract, 'all.srclist', /FIT_SPECTRA, CHANNEL_RANGE=[35,548],
MODEL_FILENAME='~/my_models/T:Solar,Solar.xcm',
MODEL_CHANGES_FILENAME='~/my_models/NhFrozen.xcm', /CSTAT
```

EXAMPLE:

```
idl |& tee fit_spectra.log
```

```
acis_extract, 'all.srclist', /FIT_SPECTRA, CHANNEL_RANGE=[35,548],
MODEL_FILENAME='~/my_models/T:Solar,Solar.xcm', /CSTAT, SNR_RANGE=[1,5],
NUM_GROUPS_RANGE=[6,250]
```

- The two-element integer vector CHANNEL_RANGE controls the range of PI channels (i.e. the spectral band) used for fitting. The values [35,548] correspond to the approximate energy range of 0.5—8 keV. When the spectrum is grouped (§5.7) these two values directly control the placement of the first and last grouping bin boundaries.
- The optional keywords SNR_RANGE and NUM_GROUPS_RANGE change the spectral grouping parameters. See §5.7 for the meaning of these parameters, each of which is a 2-element vector.
- If set the optional keyword GROUP_WITHOUT_BACKGROUND excludes the background from the computation of group significance: $SNR = \frac{SRC_CNTS}{1 + \sqrt{SRC_CNTS + 0.75}}$ This is provided for observers who prefer the traditional approach to grouping in which a threshold number of source counts is required for each group.

- MODEL_FILENAME should be the path to an appropriate *XSPEC* script (.xcm file).
- The optional MODEL_CHANGES_FILENAME parameter provides a “hook” for fitting scripts to provide a capability for the observer to modify the behavior of the script. The convention is that the fitting script should contain this line

```
# AE CUSTOMIZATIONS
```

at an appropriate place in the script where the model can be conveniently changed by inserting new *XSPEC* commands. (The scripts distributed with AE support this.) AE inserts after this line the contents of the file MODEL_CHANGES_FILENAME which should contain an *XSPEC* script fragment that changes the default behavior of the fitting script as desired.

WARNING! It is very easy to introduce typographical mistakes into the model-changes script that silently defeat the intended changes to fitting script variables that you intend, especially since variable names in the TCL language are case-sensitive. We strongly recommend that model-changes scripts should verify that a variable name exists before trying to change its value. The example model-changes script below first makes a table of (name, value) pairs for the script variables you wish to update, then runs a little block of TCL code that verifies that the specified variable names already exist in the fitting script:

```
set custom_value(kT0_init)      2.0
set custom_value(NH0_init)      1.0

foreach variable_name [array names custom_value] {
    if [info exists $variable_name] {
        set $variable_name $custom_value($variable_name)
    } else {
        puts "ERROR: '$variable_name' is not the name of a variable in the fitting script!"
        tclexit 98
    }
}
```

To freeze a parameter, set its initial value and parameter range to the same value, for example:

```
set custom_value(kT0_min)  2.0
set custom_value(kT0_init) 2.0
set custom_value(kT0_max)  2.0

foreach variable_name [array names custom_value] {
    if [info exists $variable_name] {
        set $variable_name $custom_value($variable_name)
    } else {
        puts "ERROR: '$variable_name' is not the name of a variable in the fitting script!"
        tclexit 98
    }
}
```

AE looks in two places for the script(s) specified in MODEL_CHANGES_FILENAME. For example suppose MODEL_FILENAME='~/my_models/T:Solar,Solar.xcm' and MODEL_CHANGES_FILENAME='~/my_models/NhFrozen.xcm'

AE first looks for the file <sourcename>/spectral_models/NhFrozen.xcm. If that is not found then AE looks for a file at the exact path supplied, '~/my_models/NhFrozen.xcm'. In this way you can use a single *XSPEC* script fragment to modify all source fits in the same way, and/or you can modify individual source fits independently, e.g. specifying a different frozen N_H value for each source.

If an *XSPEC* script fragment is found its basename is appended to the `model_name` used in the various fitting files and in the HDU that stores the fit results, e.g. *grp5.0-T:Solar,Solar_NhFrozen*.

If you wish to apply multiple customization scripts then `MODEL_CHANGES_FILENAME` should be an array of file names.

- The optional keyword `/CSTAT` can be supplied to specify (via the *c_stat_flag* script variable) that *XSPEC* use the C-statistic. The fitting scripts supplied with AE use the *cplinear* background model (§5.12) when `/CSTAT` is specified.
- When the optional keyword `/INTERACTIVE` is supplied the fitting script will pause after the model is set up but before the source is fit. The observer can then interact with *XSPEC* to perform the fit (e.g. to adjusting abundances in a vpec model). When satisfied with the model, the observer should enter the command “c” (for continue); the fitting script will resume, calculating fluxes and producing the normal output data products.

To use this feature, the observer must configure their *XSPEC* environment to access a small TCL program *interact.tcl* distributed with AE:

```
% cd ~/.xspec
% cp <AE package>/xspec_scripts/interact.tcl .
% xspec
XSPEC12> auto_mkindex .
XSPEC12> exit
```

- The number of CPU seconds that *XSPEC* is allowed to consume can be changed via the `FIT_TIMEOUT` option.

AE’s fitting scripts include a placeholder for loading a frozen observer-supplied model for background components that are not represented by the background observation that has been extracted. For example, when the ACIS “stowed” observations are extracted as the background spectrum for a diffuse source, then the observer will want to supply a model for the celestial background components (which are not present in the stowed data). You should ponder whether it is more appropriate for this celestial background model to be frozen or free:

- The placeholder in the fitting scripts shows how a background model, presumed to be **frozen**, is loaded and “connected to” the source spectrum you are fitting. This celestial background model would perhaps be derived by previously fitting a “blank sky” spectrum, extracted from a nearby patch of sky thought to lack emission from the diffuse object under study.
- If the “blank sky” spectrum you have available is very poor, then you might conclude that it is more appropriate to **simultaneously** fit the “blank sky” spectrum (connected to your celestial background model) and the source spectrum (connected to both your celestial background model and some source model). Modifications to AE’s fitting scripts would be required for this approach, e.g. code to load the “blank sky” spectrum, code to connect it to the celestial background model, code to save the parameters of the celestial background model, etc.

7.11.3 Troubleshooting

If an *XSPEC* session encounters a problem it will often be able to notify AE that the fit was not completed successfully. AE will print an error message, and suggest that you look in the log file produced by the *XSPEC* run. That log file can be confusing to read, because the commands that are being run by the script are not (by default) echoed to the log file. Your best shot is to skip to the end of the log file and look for a message that gives a clue about the problem.

If the log file is no help, then your next debugging step should be to repeat the *XSPEC* run manually in a shell in order to reproduce the failure outside the context of AE. Although the Unix command line that AE spawns to run *XSPEC* (shown in the *IDL* window) is complex at first glance, the heart of it is simply the normal syntax required to have *XSPEC* run a TCL script, for example:

```
xspec - spectral_models/grp3.0_tbabs_pow.xcm
```

Although you can manually run the fitting script in that manner (supplying it as a parameter to *XSPEC* on the command line), you will often find that *XSPEC* will exit immediately after the problem is encountered, preventing you from exploring the state of the model, looking at the fit, etc. Thus, it is often better to start *XSPEC* and then load the script, for example:

```
xspec
@spectral_models/grp2.0_tbabs_pow.xcm
```

If the fitting script is running normally, but the spectral model parameters are running “off into the weeds”, then you should first suspect that the problem is arising during execution of the “error” command. When a model parameter is poorly constrained, the “error” command can explore astrophysically extreme realms of parameter space while searching for a rise in the fit statistic commensurate with the specified uncertainty. Sometimes, a second and slightly better local maximum will be found during that search and will be adopted as the new best fit, even if its model parameters are astrophysically extreme (e.g. very high NH and very low kT in a thermal plasma). The best-fit *XSPEC* model that existed **before** estimation of parameter errors is saved in the file `model_before_errors.xcm` (see below). Estimation of parameter errors can be disabled by setting the `skip_errors` variable in the AE fitting script, a task most easily accomplished via the `MODEL_CHANGES_FILENAME` mechanism.

7.11.4 Data Products

Four data products will remain in the source directory:

{sourcename}_grp{significance}.pi: the multi-ObsId source spectrum grouped with the indicated significance.

spectral_models/grp{significance}_{model_name}.xcm: the script passed to *XSPEC*

spectral_models/grp{significance}_{model_name}: a directory holding the following files which may have been written during the fitting session:

ldata.ps A plot of the grouped spectrum with model.

icounts.ps A plot of the cumulative ungrouped spectrum with model.

xspec.log A text file showing the fit information.

summary.ps Summary showing ldata.ps and xspec.log.

xspec_run.log STDOUT from the *XSPEC* process.

model.xcm The final best-fit *XSPEC* model that was written to the file `source.spectra` and was used to make plots.

model_before_errors.xcm The best-fit *XSPEC* model that existed **before** the fitting script attempted estimation of parameter errors. This model will differ from that in `model_before_errors.xcm` in cases where the search for parameter errors revealed a better fit.

source.spectra: a FITS file containing the fit results in an HDU named `model_name`.

7.12 COLLATED_FILENAME Stage

This stage is run to collate all source parameters into a large FITS table (one source per row).

7.12.1 Usage and Inputs

`acis.extract`, `srclist.filename`, `obsname`, `/SINGLE_ObsId`, `COLLATED_FILENAME=filepath`, `HDUNAME=name`,
`REGION_FILE=filepath`, `REGION_TAG=value`, `LABEL_FILENAME=filepath`,
`MATCH_EXISTING=filepath`

EXAMPLE:

`acis.extract`, `'all.srclist'`, `COLLATED_FILENAME='all.collated'`, `HDUNAME='BEST_MDL'`, `REGION_FILE='all.reg'`

- The multi-ObsId source parameters in `source.stats` (e.g. position estimates) are reported.
- If the optional parameter `obsname` is supplied then the observation-specific contents of `*/<obsname>/obs.stats` are reported. If `obsname` is omitted then each source reports `obs.stats` values from whichever observation was the deepest (largest exposure map value) for that source. The name of the deepest observation is reported in the table column `PRIM_OBS`.
- The source parameters in `source.photometry` are reported.
- The fitting results from one spectral model are reported. The optional keyword `HDUNAME` (FITS Header Data Unit Name) can be used to specify the preferred spectral model (name of the preferred HDU in the FITS file `source.spectra`) when multiple models have been run. `HDUNAME` can be a scalar string if the same preference applies to all sources or can be a string vector. Four types of values for `HDUNAME` are supported:
 1. If `HDUNAME` is the reserved string “BEST_MDL” (best model) then AE looks for the FITS keyword `BEST_MDL` in the Primary HDU of the file `source.spectra` to obtain the name of the preferred model. An interactive tool for observers to review spectral models and record preferences in `BEST_MDL` is provided in §7.14.
 2. If `HDUNAME` is a simple string (e.g. “nogrps:T:Solar,Solar”) then AE looks for a spectral model (HDU) with that name.
 3. If `HDUNAME` is a pattern containing wildcards recognized by the *IDL* routine `strmatch` (e.g. “*T:Solar,Solar”) then the most recent spectral model with a matching name is used. This construct is handy when grouped spectra are fit since the group size prefix in the model name can vary between sources.
 4. If `HDUNAME` is omitted or is the empty string then the most recent spectral model (last HDU in `source.spectra`) is used.
- The optional `REGION_FILE` parameter specifies the name to use for a region file AE will construct containing extraction regions from the deepest of the observations merged, position markers, and labels for all the sources in the catalog. The optional `REGION_TAG` parameter (scalar or vector string) specifies an additional “tag” that will be assigned to the *DS9* regions.
- If the optional parameter `/SINGLE_ObsId` is specified then
 - observation-specific source properties (from `obs.stats`) take precedence over multi-ObsId properties of the same name (from `source.stats`).
 - Photometry and spectral fitting information (both multi-ObsId entities) are not reported.
- The optional `LABEL_FILENAME` parameter specifies the name of an ASCII table that will be written listing each source’s name and `LABEL`.
- The optional `MATCH_EXISTING` parameter can be used to explicitly define the columns desired in the collated table, saving about half of the run time of this stage. If the specified collated filename already exists and you simply wish to update its contents then specify `/MATCH_EXISTING`. If you wish to use the same table columns found in another collated file, then pass its name (e.g. `MATCH_EXISTING=template.collated`). If you have a template structure already defined in *IDL*, then pass that structure (e.g. `MATCH_EXISTING=collated_table`).

Some accessory tools are available for working with spectral fit results. See §7.14. The FITS binary table produced by this stage can of course be examined and manipulated by standard FITS tools, e.g. *fv* in *HEASOFT* or *chips* in *CIAO*. The Event Browser application in the *TARA* package⁸² can also be pressed into service for browsing any FITS binary table, e.g. the table of source parameters produced by the *COLLATED.FILENAME* stage. Start Event Browser with the */GENERIC* keyword. The table columns can be examined in various ways, e.g.:

- plot one column as a function of another (*function_1d*)
- compute univariate distribution of a column (*dataset_1d*)
- compute bivariate distribution of two columns (*dataset_2d*)
- compute trivariate distribution of three columns (*dataset_3d*)
- compute a statistic map using three columns (*dataset_3d*)
- examine how the distribution of one column depends on another column (*trend_1d*)

7.13 Caveats

- The design of many *CIAO* tools rests on the assumption that the event data are expressed in the original SKY coordinate system of the observation, NOT IN A COORDINATE SYSTEM REPROJECTED TO MATCH ANOTHER OBSERVATION. The *CIAO* thread Merging Data from Multiple Imaging Observations⁸³ states:

The merged event list should not be used for spectral analysis, since it does not contain sufficient information to generate correct response files. The recommended technique for the spectral analysis case is to generate separate PHA, RMF, and ARF files for each observation and to analyze them simultaneously.

This requirement is simple to understand—all the tools that produce calibration data products (e.g. *mkarf*, *mkacisrmf*, PSF generating tools like *MARX*) need to know where **on the detector** the source lies. To the extent that the ObsIDs that you reproject and merge are dissimilar in their pointing position and roll angle, the notion of a single position on the detector becomes undefined. Thus, judgment on your part is required. Some projects that have the luxury of a single pointing with fairly consistent roll angles among all ObsIDs have used a strategy of combining the data to make it look like one ObsID (with long “bad time intervals”).

Small shifts and roll angle variations among merged ObsIDs are expected to induce only small distortions to PSF models and RMFs, because these characteristics vary smoothly with position on the detector. The ARF also depends in part on smoothly varying quantities (vignetting and detector QE). However, the ARF also depends on a quantity with high spatial frequency—the dither pattern of the CCD edges and CCD bad columns, projected onto the sky—as can be seen in the exposure map. Thus, a source extracted from reprojected and merged data will suffer significant calibration errors if in reality it falls on very different features in the single-ObsID exposure maps.

Observers sometimes wish to tweak the astrometry of the ACIS data to better match an astrometric reference. We recommend that you choose a method of astrometry adjustment that will not change (within *CIAO*) the inferred location of events on the detector. The only astrometry tweaking method that we feel comfortable with is changing the RA and DEC columns in the aspect time series (i.e. “fixing” the aspect solution itself), and then regenerating the event list with *acis_process_events* or with *reproject_events*.

- When the path to the directory in which you run AE gets too long the path AE must put in the PFILES environment variable gets too long for the *addrmf* tool to handle and *addrmf* ungracefully dies with a “core dump” message. Similarly, the fitting scripts can fail on a call to *fstruct* when the path in PFILES gets too long.
- One of the simplifying assumptions AE makes is that local background regions can be constructed without worrying about CCD boundaries, i.e. that the instrumental background and the response to astrophysical X-rays is very similar on all CCDs. This is a reasonable assumption on the ACIS-I array, where all CCDs are front-illuminated

⁸²<http://personal.psu.edu/psb6/TARA/>

⁸³<http://cxc.harvard.edu/ciao/threads/combine/>

(FI). However, the back-illuminated (BI) CCD S3 has very different characteristics. Incorrect background spectra will be produced if you allow AE to mix source extractions from one type of CCD with background extractions that cover the other type of CCD. The safest strategy is to divide each actual observation into two ObsIDs (event file and exposure map) that are run through AE—one that has only FI CCDs and one that has only S3.

7.14 Accessory Tools

7.14.1 Verify the astrometric alignment of individual ObsIDs

The *ae_interObsID_astrometry* tool helps the observer verify the astrometric alignment of individual ObsIDs after an AE extraction has been performed. The astrometric offset (with uncertainty) between each pair of ObsIDs is estimated using the set of reliable (`PROB_NO_SOURCE < 0.003`) sources that are extracted on-axis (`THETA < 5'`) and are uncrowded (`PSF_FRAC > 0.85`) in both ObsIDs.

If a reference catalog (such as 2MASS) has been converted into the format used by our *match_xy* tool, then the offset between it and each ObsID is also estimated.

Usage and Inputs

`ae_interObsID_astrometry, obsname_list, ASTROMETRY=fits_astrometry, REF_CATALOG=ref_cat`

EXAMPLE (assuming the standard directory structure (§3):

```
setenv OBS_LIST ""
foreach dir (../obs*)
  if (! -d $dir) continue
  set obs=`basename $dir | sed -e "s/obs/"`
  setenv OBS_LIST "$OBS_LIST $obs"
end
echo $OBS_LIST

idl
.run ae
; Without a reference catalog
ae_interObsID_astrometry, strtrim(strsplit(getenv('OBS_LIST'), /EXTRACT), 2), recommendation

; With a reference catalog
.run match_xy
event2wcs_ast = get_astrometry_from_eventlist('../../tangentplane_reference.evt')

twomass_cat    = build_FITS_cat('2mass_highqual.fits', event2wcs_ast, RA_EXPRESSION='tb.RAJ2000', DEC_

ae_interObsID_astrometry, strtrim(strsplit(getenv('OBS_LIST'), /EXTRACT), 2), ASTROMETRY=event2wcs_ast
```

7.14.2 Convert source labels to source names

The *ae_make_catalog* tool (§7.5) assigns a LABEL property to each source for use in several contexts as a shorter source identifier than the source name. Given a source name, one can easily find its LABEL by examining its source.stats FITS file. Given a source LABEL, to find the source name one must consult a table that lists LABELs and names for the entire catalog.

The collated FITS table itself can be searched for a list of LABELs using the tool *ae_label2name*.

Usage and Inputs

`ae_label2name, labels, collated_filename, catalog_names, indexes`

EXAMPLE:

```
.run ae
```

`ae_label2name, ['101','202','303'], 'all.collated'` `ae_label2name, 3, 'all.collated'` The parameter *labels* can either be a string array, or an integer. If an integer is supplied, then the tool prompts for you to enter the specified number labels.

Alternatively, such a table can be constructed in ASCII form using the LABEL_FILENAME option to the Collate Stage (§7.12), and then searched for a specific LABEL either manually or with *grep*. For example, if the Collate

Stage has produced the ASCII table *label.txt* then this shell alias would change directories to the source with the specified label:

```
alias asl 'setenv LINE `grep -i "\\!:" label.txt`; echo $LINE; if ( "$LINE" != "" ) pushd `echo $LINE`
```

7.14.3 Constructing Exposure Maps

An exposure map can be conveniently constructed using the tool *ae_make_emap*.

Usage and Inputs

```
ae_make_emap, obsdata_fn, CCD_LIST=value, SCALING_PARAMS=structure, MONOENERGY=value,  
ARDLIB_FILENAME=filepath, ASPECT_FN=filepath, MASKFILE=filepath,  
SPECTRUM_FN=filepath, MATCHFILE=filepath, /ONLYTIME, /REUSE_ASPHIST,  
/REUSE_INSTMAP
```

EXAMPLE:

Create an emap for the I-array CCDs at 1 keV, named “obs.iarray.emap”.

```
.run ae  
params = {map_type           : 'exposure map'           , $  
          map_filename       : 'obs.iarray.emap'        , $  
          units               : 's cm^2 ct /photon'     , $  
          normalize_before_scaling: 0B                  , $  
          ccd_scaling         : replicate(1.0, 10) $    , $  
          }  
  
ccd_list = '0123'
```

```
ae_make_emap, 'acis.evt', SCALING_PARAMS=params, CCD_LIST=ccd_list, MONOENERGY=1.0, $  
ARDLIB_FILENAME='ardlib.par', ASPECT_FN='acis.asol1', MASKFILE='acis.msk1'
```

- Any event list containing exposure-related FITS keywords and the DETNAM keyword can be supplied as *obsdata_fn*.
- The filenames of the exposure maps desired are supplied via the structure *SCALING_PARAMS*, as demonstrated above. For a normal exposure map, the parameters *normalize_before_scaling* and *ccd_scaling* must be as shown above.
- The list of CCDs to be used in each emap are supplied via the string scalar *CCD_LIST*. If omitted all CCDs are used.
- For a mono-energy emap supply *MONOENERGY* (in keV) or let it default to 1.0. Alternatively you can pass a spectrum to *mkinstmap* via the keyword *SPECTRUM_FN*.
- A pixel size of 1 sky pixel is normally used in the exposure maps. We recommend using the *CIAO* tool *reproject_image* to resample this high-resolution emap when you need an emap to match a specific field of view and binning scheme used in an image. However if you wish to directly compute an emap on a specific pixel grid, then supply a template image via the *MATCHFILE* parameter.
- The *ardlib.par* file (configured for your observation), aspect file, and mask file required by the *CIAO* tools *mkinstmap* and *mkexpmap* are supplied via the keywords *ARDLIB_FILENAME*, *ASPECT_FN*, and *MASKFILE*.
- If */ONLYTIME* is specified then *mkexpmap* is told to model the HRMA and ACIS with a perfect QE, resulting in an exposure map the represents only the geometric effects of dithered chip edges and bad pixels.
- A sub-directory named *asphist* is created to hold aspect histogram and instrument map files.
- If */REUSE_ASPHIST* is specified then any aspect histograms found from a previous run of the tool are re-used.
- If */REUSE_INSTMAP* is specified then any instrument maps found from a previous run of the tool are re-used.

7.14.4 Grouping

AE’s spectral grouping algorithm (§5.7) is available as a tool (*ae_group_spectrum*) that can be used to manually group a spectrum. The parameters of this routine are shown below:

Usage and Inputs

```
ae_group_spectrum, src_spectrum_fn, bkg_spectrum_fn, grouped_spectrum_fn, CHANNEL_RANGE=[value,value],  
                  SNR_RANGE=[value,value], NUM_GROUPS_RANGE=[value,value]
```

EXAMPLE:

```
.run ae
```

```
ae_group_spectrum, '182029.89-161044.4.pi', '182029.89-161044.4.bkg.pi', "", SNR_RANGE=[5,10]
```

- The input (ungrouped) spectrum file name is specified as *src_spectrum_fn*.
- If you want group significance calculations to use the background, supply the background file name via *bkg_spectrum_fn*; otherwise supply the empty string.
- The output (grouped) spectrum file name is specified as *grouped_spectrum_fn*; set to the empty string to use an auto-generated name.
- The optional *CHANNEL_RANGE*, *SNR_RANGE*, and *NUM_GROUPS_RANGE* parameters are described in §7.11.

7.14.5 Time-resolved extractions

Time-resolved extractions can be performed with the tool *ae_timerange_extract*. The parameters of this tool are shown below:

Usage and Inputs

```
ae_timerange_extract, sourcename, obsname, time_filter, extraction_name, EVTFILE_BASENAME='spectral.evt'
```

EXAMPLE:

```
.run ae
```

```
ae_timerange_extract, '182029.80-161045.5', '6420', '270786808.58:270854668.99', '6420.1', EVTFILE='spectral.evt'  
ae_timerange_extract, '182029.80-161045.5', '6420', '270854678.71:270897521.73', '6420.2', EVTFILE='spectral.evt'  
ae_timerange_extract, '182029.80-161045.5', '6420', '270897531.45:270938175.77', '6420.3', EVTFILE='spectral.evt'
```

- The *time_filter* input should be a single string or string array of *CIAO* time specifications of the form “tstart:tstop”.
- The *extraction_name* input should be a corresponding single string or string array of names for the extractions. These names are used to create subdirectories for the extractions (via the parameter *EXTRACTION_NAME* passed to *ae_standard_extraction* and the parameter *MERGE_NAME* passed the the /MERGE stage of AE).
- The *EVTFILE_BASENAME* parameter is passed to *ae_standard_extraction* to specify the filename for the event list (in *../obsXXXX/*) from which you wish to extract spectra.

7.14.6 Comparing event data with a PSF model

Many observers want to know if their Chandra source is “extended”. This task is inherently difficult for a variety of reasons.

- If the PSF model and the event data are not well aligned (i.e. the estimated source position is incorrect) then data that are truly consistent with a point source may be found to be inconsistent with the PSF model.
- Bright sources are particularly problematic. In principle they give you nice signal to detect deviations from the PSF. However, even moderate pile-up will invalidate the PSF model. Also, the requirements for accuracy in the PSF model rise as the number of observed counts rises.

- Since the Chandra PSF is strongly energy-dependent, a PSF model constructed assuming a spectrum different from the spectrum of the event data may produce a spurious finding of inconsistency. Note that AE produces only mono-energy PSF models.
- There are many other ways in which a PSF model can be flawed, e.g.
 - The model of the HRMA may not match the actual HRMA. See, for example, the scary “hook” (containing 5% of the PSF power) recently discovered in the HRMA PSF⁸⁴.
 - The *MARX* parameters (“AspectBlur”, “pixadj”) controlling the blurring applied to the HRMA rays to model ACIS pixelization, aspect reconstruction, and sub-pixel positioning effects may not be accurate.
 - Most PSF models (including AE’s) do not account for background events. Thus, sources with a large fraction of background events are likely to produce a spurious finding of inconsistency with the PSF. Background events that are concentrated at the edge of the extraction aperture (e.g. the wings of a nearby source) are particularly inconsistent with the PSF.

CIAO provides a tool named *srcextent* that may be helpful to you. AE provides a very simple tool named *ae_radial_profile* that can quickly scan every AE source for signs that a source *might be* inconsistent with the PSF. The tool compares the merged event data with the merged PSF. The basic call is shown below. The 2011mar16 or later version of our TARA package is required.

```
.run ae
ae_radial_profile, report, SRCLIST='all.srclist'
```

If you ran AE’s MERGE stage with a MERGE_NAME option, then supply that same option above.

The output data structure (“report” above) is an array of structures. Each source reports K-S probabilities quantifying the similarity between the event data and PSF model within a circular aperture defined by SRC_RADIUS, in three ways:

1. radial distance from the source position declared in source.stats
2. X-offset from the source position
3. Y-offset from the source position

K-S probability is the fraction of true point sources that would produce observations that would show a deviation from the PSF model larger than what you actually observed, assuming of course that the model is correct. Thus, a very small K-S probability is evidence that your source is NOT consistent with the PSF. The inconsistency could be in either sense—the data are too tightly bunched (e.g. from afterglow events) or are too loosely bunched (e.g. from an extended object).

From this report, you can make a list of sources that claim inconsistency with the PSF (low K-S values) using forprint, e.g.

```
forprint, report.SOURCENAME, SUBSET=where(report.PROB_KS_DISTANCE LT 0.01), $
/NOCOM, TEXTOUT='/tmp/low_ks.txt'
```

Then, you can run *ae_radial_profile* a second time on this shorter source list, using the PLOT option to produce three interactive plots for the source showing cumulative distributions of radial distance, X-offset, and Y-offset; press the “next source” to move to the next source.

```
ae_radial_profile, report, SRCLIST='/tmp/low_ks.txt' , /PLOT
```

The tool takes an optional ENERGY_RANGE parameter that specifies the event energy band on which you want to compare to the PSF. The default is [0.5,8] keV, which may be too broad for your taste or not appropriate for the PSF energy you have used.

⁸⁴http://cxc.harvard.edu/ciao/caveats/psf_artifact.html

Although the plots look reasonable, I certainly cannot guarantee that there are no mistakes in this tool. In such code it's easy to be off-by-one-pixel in a calculation, or to confuse PSF pixels and sky pixels, or to make any number of other mistakes. For any source that this tool claims is “extended”, you should try to verify that conclusion by other means.

The `ae_radial_profile` code is fairly well commented. You should be able to read it without too much difficulty if you want to understand what it is doing.

7.14.7 PSF “Hook” Feature

In October 2010 the *Chandra* X-ray Center announced the discovery of a hook-shaped feature in the *Chandra* PSF⁸⁵ that extends $\sim 0.8''$ from the main peak and contains $\sim 5\%$ of the flux. Its location depends upon the roll angle of the observation; given its limited spatial extent, the hook is not discernable for sources observed substantially off-axis. At this time, its energy dependence (either spatially or as a fraction of the total power) is not well-characterized. No revised PSF models incorporating the hook-shaped feature are currently available.

Since this feature is not represented in the available PSF models, image reconstructions may generate spurious candidate point sources near bright sources (e.g. those with > 100 counts obtained with a single roll angle and located $\sim < 4'$ off-axis), where enough counts might be distributed along the hook that it could be reconstructed into what appears to be one or more independent, faint point sources.

Until a model for this feature is available, observers are encouraged to review the regions around all bright sources expected to be significantly contaminated by the hook feature to identify detections that are spatially coincident with the hook feature. Such sources should be considered suspicious, unless they contain substantially more than 5% of the flux of their neighboring bright source.

The tool `ae_make_psf_hook_regions` will construct a region file marking the hook region (for each ObsID) around sources that have more than a specified number of counts expected in the hook and lie within a specified off-axis angle. For example:

```
.run ae
ae_make_psf_hook_regions, COLLATED_FILENAME='all.collated', $
    HOOK_CNTS_THRESHOLD=4, THETA_THRESHOLD=5
```

Once a list of suspicious detections is identified, their individual extractions can be examined with the hook regions displayed via the SHOW stage, e.g.

```
acis_extract, COLLATED_FILENAME='all.collated', SRCLIST_FILENAME='near_psf_hook.srclist', /SHOW_REGIONS,
REGION_FILENAME='psf_hook.reg'
```

7.14.8 Printing spectra

The AE package includes a little Perl program `plot_spectra.pl` (originally developed by Franz Bauer) to print PostScript spectra produced by the AE fitting scripts, laid out 12 to a page.

EXAMPLE:

```
plot_spectra.pl 18*/nogrp_T:Solar,Solar/ldata.ps
```

⁸⁵http://cxc.harvard.edu/ciao/caveats/psf_artifact.html

7.14.9 Reviewing spectral fits

The *IDL* program *ae_spectra_viewer* provides a graphical user interface to help the observer conduct a source-by-source review of multiple spectral models produced by multiple sessions of the FIT_SPECTRA stage. The observer is shown a table of fit results (parameter values, luminosities, and *XSPEC* plots) for all the models available, and the observer is able to designate the preferred model (by clicking on a row of that table). That model preference is stored in a FITS keyword (*BEST_MDL*) in the file holding the spectral models (*source.spectra*) and a symbolic link pointing to the preferred model is created; the preferred models can then be collated by the COLLATED_FILENAME stage (§7.12).

This tool is a re-implementation of the Perl tool *spectra_viewer.pl* developed by Konstantin Getman. The X Windows program *gv* is used to display the Postscript spectra produced by *XSPEC*. You must have the GNU implementation of *gv* (version 3.6.1 or higher) in your Unix path.

Usage and Inputs

```
ae_spectra_viewer, collation_filename, SRCLIST_FILENAME=filepath, DISTANCE=value,  
    DISPLAY_FILE=filepath, CATEGORY_LIST=value, KEYLIST=value, /READ_ONLY,  
    HDUNAME=name, MERGE_NAME=name
```

EXAMPLE:

```
; Build a collation of the sources that have been fit.  
.run ae  
acis_extract, 'xspec.srclist', COLLATED_FILENAME='xspec.collated'  
  
; Review all collated sources; calculate luminosities for a distance of 2300 parsecs.  
ae_spectra_viewer, COLLATED_FILENAME='xspec.collated', DISTANCE=2300  
  
; Show target.evt in a ds9 session, panning as each source is selected.  
ae_spectra_viewer, COLLATED_FILENAME='xspec.collated', DISTANCE=2300, DISPLAY_FILE='target.evt'  
  
; Change the set of “categories” that the reviewer may assign to sources.  
list = ['good fit', 'bad fit', 'poor grouping', 'fit by hand']  
ae_spectra_viewer, COLLATED_FILENAME='xspec.collated', CATEGORY_LIST=list  
  
; Change the set of columns in the collation that are displayed in the table of fits.  
list = ['NH1', 'NH1_ERRU', 'NH1_ERRL', 'NH1_ERRS', 'KT1', 'CHI_SQR']  
ae_spectra_viewer, COLLATED_FILENAME='xspec.collated', KEYLIST=list  
  
; Browse the sources in read-only mode (so that user actions are recorded).  
ae_spectra_viewer, COLLATED_FILENAME='xspec.collated', /READ_ONLY  
  
; Review a subset of sources in the collation, listed in review.srclist.  
ae_spectra_viewer, COLLATED_FILENAME='xspec.collated', SRCLIST_FILENAME='review.srclist'  
  
; Collate the decisions recorded during the review, and the parameters of the selected “best model”.  
acis_extract, 'xspec.srclist', COLLATED_FILENAME='xspec.collated', HDUNAME='BEST_MDL'
```

The source list passed via the optional SRCLIST_FILENAME input is a mere convenience—it allows you to step through a subset of the sources in the collation, with a different sorting. (When this option is specified, a second set of navigation controls is shown in the tool.) The same functionality could be achieved by building another collation that contains the desired sources in the desired order.

An event list or FITS image passed via the `DISPLAY_FILE` input will be displayed in a ds9 session. Each time you navigate to a new source in *ae_spectra_viewer*, that ds9 session will be panned to the position of that source.

If you wish to assign a default “best model” to a set of sources, before they are reviewed in *ae_spectra_viewer*, then use the tool *ae_default_model_preference*, e.g.

EXAMPLE:

```
; Set the default “best model” to “T:Solar,Solar”.
; Use /FORCE keyword to override any existing BEST_MDL values.
ae_default_model_preference, 'one-temperature.srclist', '*_T:Solar,Solar'
```

7.14.10 Catalog matching

The *TARA* package includes a tool (*match_xy.pro*) to match catalogs using positional uncertainties for individual sources and a match significance threshold, rather than using a fixed matching radius as is commonly employed. This tool can be used to match catalogs from different observatories, or can be used to find the intersection of *Chandra* source lists obtained from multiple source detection runs. (See §10.1 of the *TARA* User’s Guide⁸⁶)

7.14.11 Adding source properties

The little program *ae_poke_source_property* (in `acis_extract_tools.pro`) will “poke” a FITS keyword into the `source.stats` or `obs.stats` file for each source in the `srclist`. (Obviously this could be done with *CIAO* or *HEASOFT* tools as well.) Some observers might use such keywords to carry ancillary information about sources.

7.14.12 Performing a spectral fit using an existing script

The code the AE fitting stage uses to spawn *XSPEC* and manage the data products produced has been pulled out into a separate routine *ae_perform_fit* which can be used to execute an existing *XSPEC* script created by the fitting stage. This might be useful if you have hand-edited the script for some reason.

EXAMPLE:

```
.run ae
ae_perform_fit, '181948.82-160624.2', 'nogrp-T:Solar,Solar'
```

- The first input is the name of an AE source.
- The second input is the name of the *XSPEC* script (without the `.xcm` extension), assumed to be in `<source>/spectral_models/`

7.14.13 Scanning for Residual Afterglow Events

An example of using the tool *ae_afterglow_report* is shown in our AE recipes (§7.1).

7.14.14 Table generation

The COLLATE stage of AE produces a FITS table with vector columns, in order to record source properties over many energy bands. The tool *ae_flatten_collation* transforms the AE collation into a more convenient and scientifically useful FITS table.

- Results are retained for only the three energy bands we typically care about. Those quantities are stored in independent columns, eliminating all vector columns.
- The unfriendly column names used by AE are replaced with more meaningful names.
- The *CIAO* tool *aprates* is used to estimate confidence intervals on `NET_CNTS`, replacing the less accurate AE photometry confidence intervals.

⁸⁶<http://personal.psu.edu/psb6/TARA/>

- The observed energy flux is very simply estimated by multiplying AE’s photon flux by the median observed event energy.
- The confidence intervals on spectral model parameters produced by *XSPEC* are formatted into \LaTeX strings, ready for use in a \LaTeX table of fitting results (either the ones we provide with AE, or custom tables produced by the observer).
- *XSPEC* confidence intervals on model normalizations are converted to emission measures and formatted into \LaTeX strings. *XSPEC* fluxes are converted to luminosities.

EXAMPLE:

```
.run ae
ae_flatten_collation, COLLATED_FILENAME='photometry.collation', FLAT_TABLE_FILENAME='xray_properties.fits',
SORT=0
```

The tool *hmsfr_tables* (in *acis_extract_tools.pro*) converts the FITS table produced by *ae_flatten_collation* into several \LaTeX tables of AE results that we find useful in our star formation research. Stubs of these are shown in Figures 1, 2, and 3. Several *IDL* “.sav” files holding various source parameters are also produced. This program has not yet been adequately documented, although the code comments give some information and it is used in our example recipe (§7.1).

A Running Multiple Instances of *ACIS Extract*

When your project involves several ACIS observations and you have a computer with several CPUs, a lot of time can be saved by running multiple instances of AE in parallel, each in its own IDL session. The speedup you will achieve depends on many factors, and on the stage of AE (since some stages are CPU-limited and some stages are filesystem-limited).

AE takes the following precautions to help support multiple instances running simultaneously:

CIAO and HEASOFT The *CIAO* and *HEASOFT* packages do NOT nominally support running multiple instances of the same tool. The problem is that tools write to the user's parameter files, pointed to by the environment variable \$PFILES. To avoid parameter file conflicts among multiple instances of any *CIAO* or *HEASOFT* tool, AE creates a scratch directory for parameter files and configures the environment variable PFILES to look for parameter files in the following paths:

- the scratch directory
- \$ASCDS_INSTALL/contrib/param/
- \$ASCDS_INSTALL/param/
- \$HEASOFT/syspfiles

The value of PFILES inherited from the observer is ignored. Thus the user is freed from confusing and error-prone manipulations of the PFILES environment variable. The one parameter file which must contain observation-specific information, *ardlib.par*, is explicitly provided to AE by the observer in the EXTRACT_SPECTRA stage (§E.3).

AE Scratch Files The scratch files used by AE are placed in a uniquely named scratch directory (/tmp/AExxxx.noindex/) to avoid conflict with other running AE sessions.

Lock Files In the FIT_SPECTRA stage lock files are used to prevent multiple instances of *XSPEC* running on the same source.

The one known opportunity for conflict between multiple instances of AE is the */<ObsId>/obs.stats FITS files where various source properties are stored by several AE stages. The EXTRACT_EVENTS, EXTRACT_SPECTRA, TIMING, and EXTRACT_BACKGROUND stages all write to these files. Thus it is not completely safe to run these stages simultaneously for the **same** observation. AE sessions processing *different* observations should run correctly.

As shown in Appendix D, the AE recipe tools write to the obsXXXX directories in the standard AE directory tree. Thus obsXXXX directories must not be shared among multiple projects.

B Comparison of Wachter and *cplinear* Background Models

§5.12 describes the Wachter method for modeling the background when the C-statistic is used, and describes the alternative *cplinear* background model that we use in AE.

We have not invested resources in attempting to fit simulated spectra with the cplinear method to characterize the performance of the method. We have however compared the Wachter and cplinear methods for 900 ACIS sources in the Tr14 star cluster, many of which are weak. For each source, the cplinear model was fit first and those *tbabs*vapec* model parameters were used as the initial conditions of a second fit using the Wachter model. Figures 11 to 17 compare the *tbabs*vapec* model parameters and observed fluxes in three bands obtained by the two methods. The general trend is that for many sources the derived plasma temperatures and observed fluxes are lower in the Wachter fit, consistent with the examples shown in Figure 7. The hard band shows the largest flux deficit, consistent with the fit residuals in Figure 7 which tend to be largest in the hard band. The column density results have more scatter, but the Wachter fit tends to give higher values. These experiments do not directly address the biases of each method with regard to the true spectral parameters, but examination of many plots like those in Figure 7 has convinced us that the cplinear fits are a better representation of our ACIS spectra.

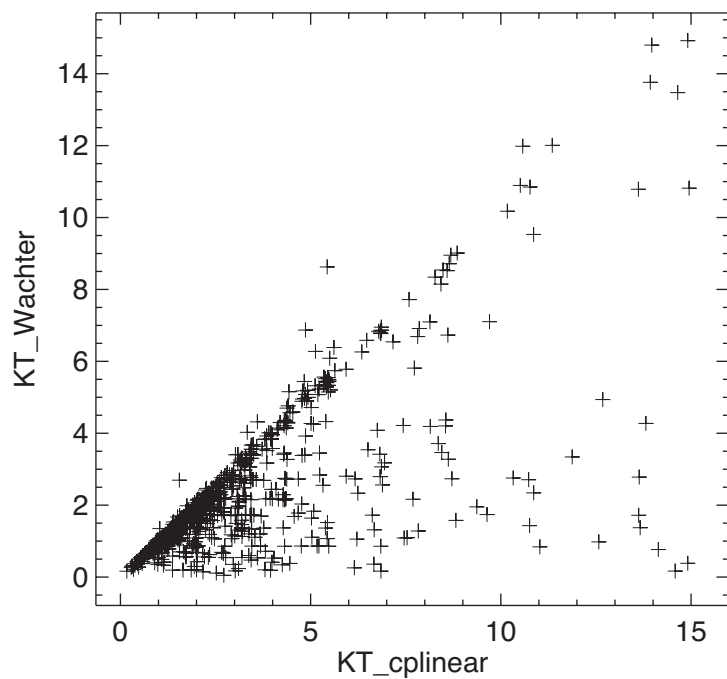


Figure 11: Plasma temperatures derived using the Wachter (Y-axis) and cplinear (X-axis) background models for 900 ACIS sources.

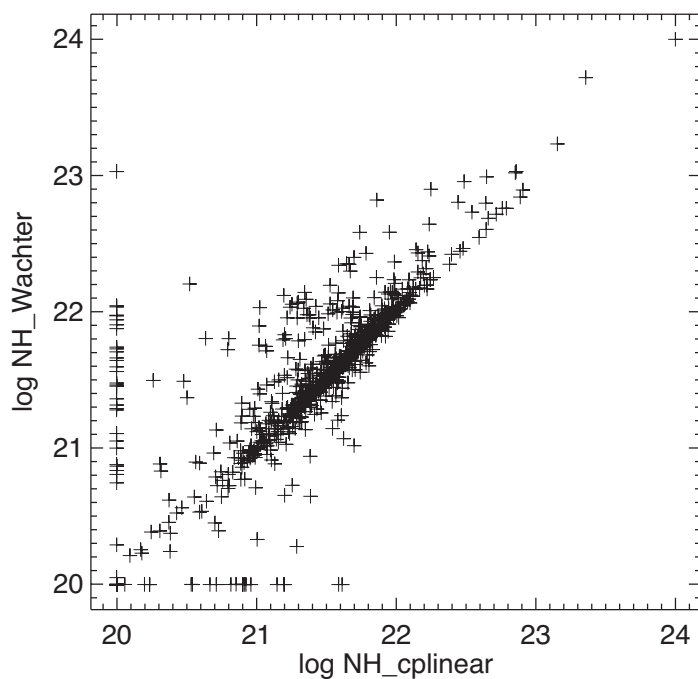


Figure 12: Column densities derived using the Wachter (Y-axis) and cplinear (X-axis) background models for 900 ACIS sources. The symbol at $\sim (20, 20)$ depicts 98 sources where NH was clipped at its lower bound.

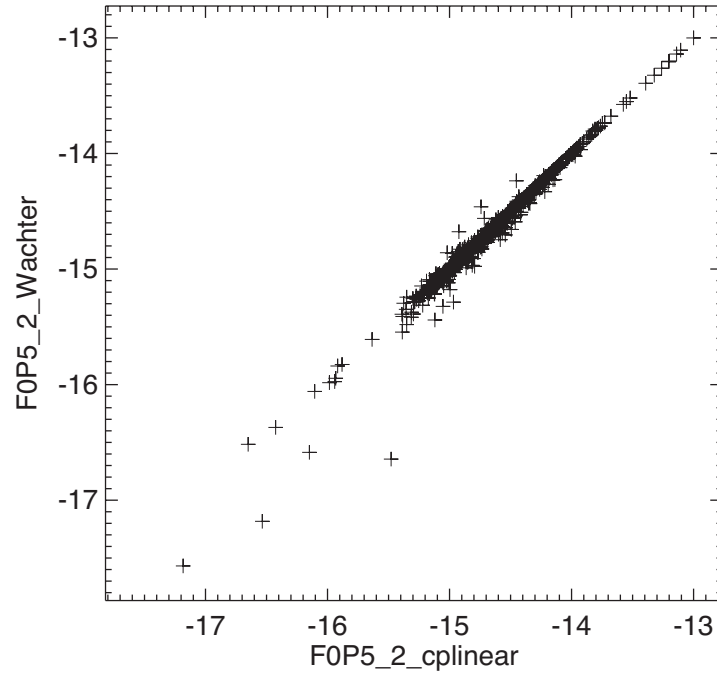


Figure 13: Fluxes (0.5-2 keV, NOT absorption-corrected) derived using the Wachter (Y-axis) and cplinear (X-axis) background models for 900 ACIS sources.

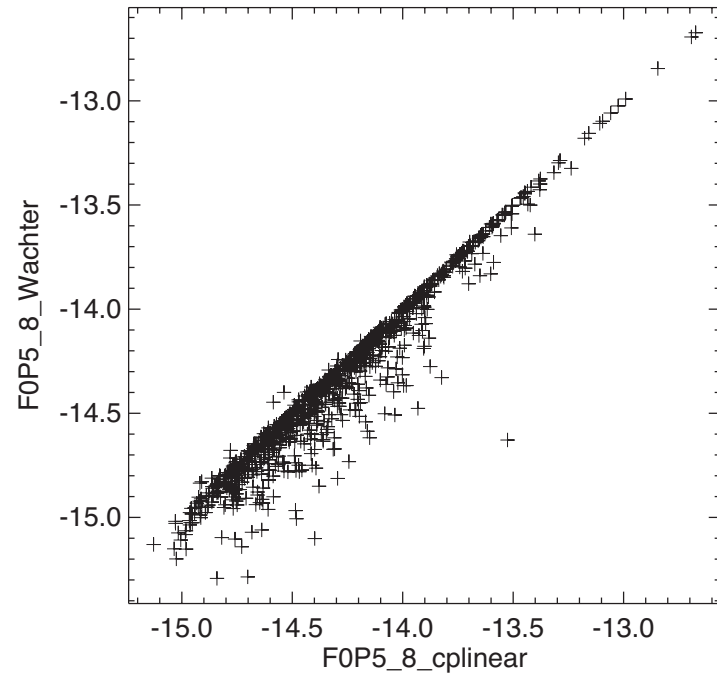


Figure 14: Fluxes (0.5-8 keV, NOT absorption-corrected) derived using the Wachter (Y-axis) and cplinear (X-axis) background models for 900 ACIS sources.

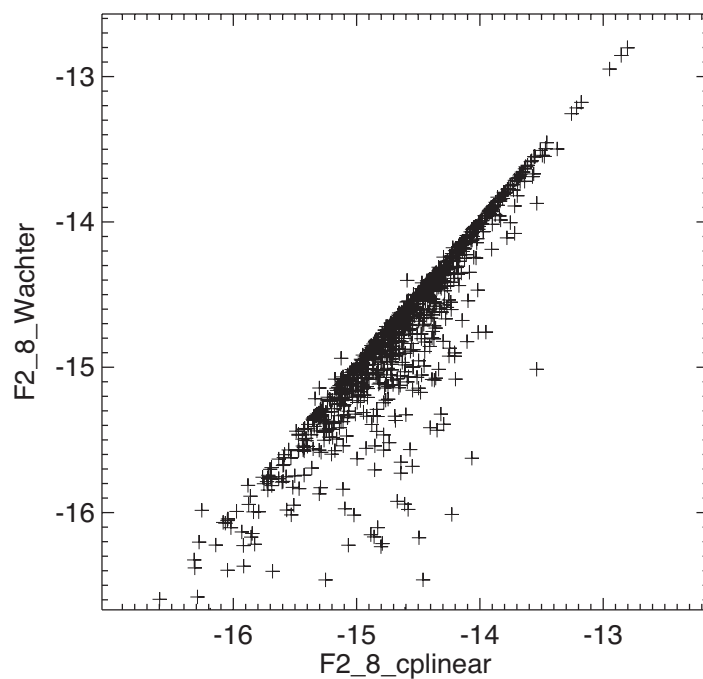


Figure 15: Fluxes (2-8 keV, NOT absorption-corrected) derived using the Wachter (Y-axis) and cplinear (X-axis) background models for 900 ACIS sources.

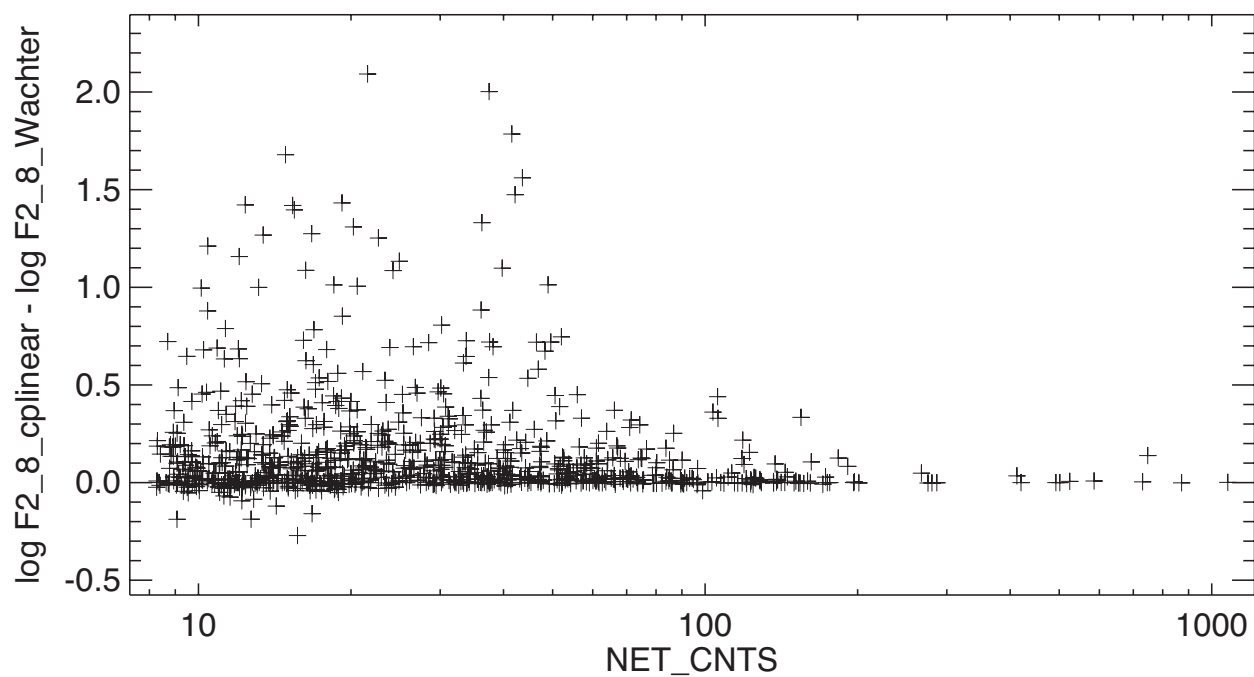


Figure 16: Hard-band (2-8 keV, NOT absorption-corrected) flux differences between the Wachter and cplinear background models as a function of net counts extracted.

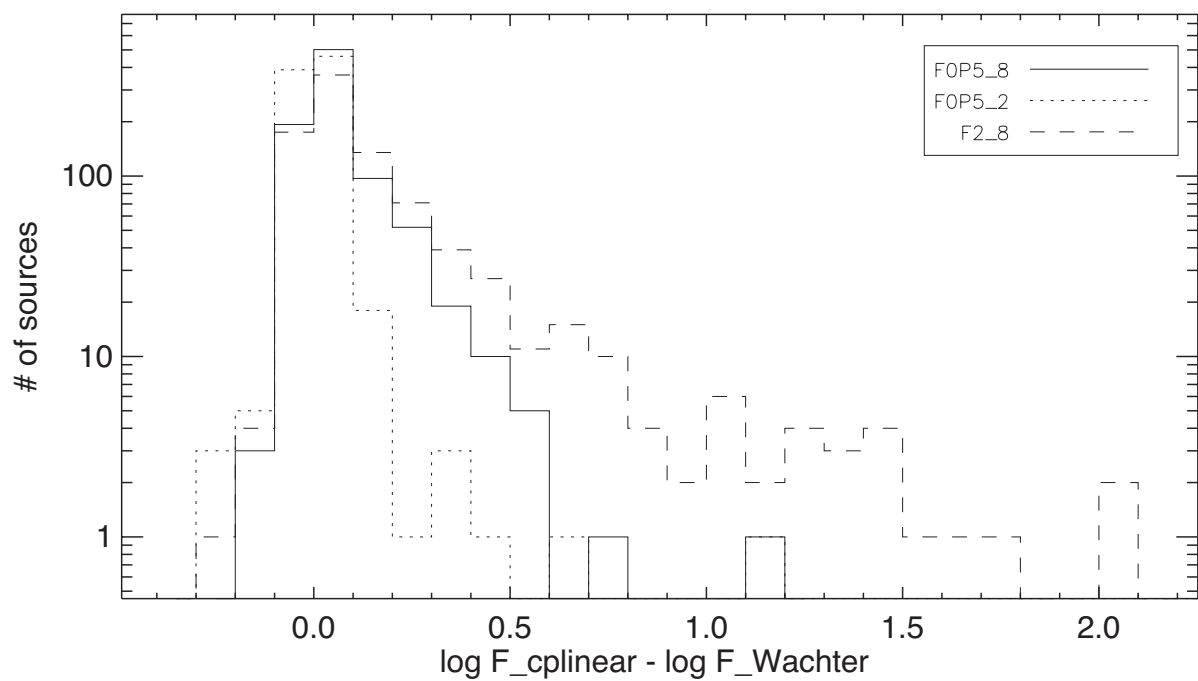


Figure 17: Distributions of flux differences (NOT absorption-corrected) in three bands between the Wachter and cplinear background models.

C Correction for Photometric Bias

Although the *ae_better_backgrounds* algorithm (§E.6) seeks to define a background region that is unbiased, i.e. one where the flux from each neighbor that we expect in the background exactly matches the contamination we expect from that neighbor, the algorithm will sometimes be forced to construct a background that has significant photometric bias over the nominal energy band. This is very undesirable because inaccuracy in the broad band background level translates to inaccuracy in the source significance statistics (§5.10) which are commonly used for pruning the source catalog, as well as inaccuracy in photometry. Thus, the algorithm includes a final step which seeks to adjust the scaling of the background spectrum so as to correct for the photometric bias. That corrected scaling is derived below.

First, note that the algorithm described above produces several important quantities for each source:

- N : the integer number of in-band counts observed in the background region.
- $BACKSCAL_{src}, BACKSCAL_{bkg}$: the measures (integrals of the exposure map) of the source and background regions. For convenience we will denote the nominal background scaling as $S = \frac{BACKSCAL_{src}}{BACKSCAL_{bkg}}$.
- B_p : the total number (real-valued) of counts from neighboring point sources expected to contaminate the source region, obtained by integrating models of the neighboring sources over the source region.
- N_p : the total number (real-valued) of counts from neighboring point sources expected to be observed in the background region, obtained by integrating models of the neighboring sources over the background region..
- $\Delta_p = B_p - N_p * S$: the estimated “photometric bias” of the background region (with regard to contamination from neighboring point sources), when the nominal scaling is applied. As said earlier, the algorithm seeks to drive Δ_p to zero (among other goals).

Second, we infer the number (real-valued) of background counts observed in the background region but **not** attributed to detected point sources as

$$N_f = \max(N - N_p, 0)$$

Since we have no model for the spatial distribution of this background component, we assume it is flat (denoted by the f subscript in N_f). Thus the number of counts from this background component expected in the source region is

$$B_f = N_f * S$$

Third, we wish to choose a correction to the background scaling, c , such that the final scaled background is unbiased with respect to our models, i.e.

$$\begin{aligned} (N_p + N_f) * c * S &= B_p + B_f \\ &= (N_p * S + \Delta_p) + N_f * S \\ &= (N_p + N_f) * S + \Delta_p \end{aligned}$$

Solving for c we have

$$c = 1 + \frac{\Delta_p}{(N_p + N_f) * S}$$

Note that when there is no photometric bias, $\Delta_p = 0$, $c = 1$, and the nominal background scaling (ratio of the measures of the source and background regions) is used. When Δ_p is positive (we estimate that our background region does not contain enough power from neighboring sources), then $c > 1$ and we choose to subtract more background than the nominal background scaling (S) would call for. When Δ_p is negative then $c < 1$ and we choose to subtract less background than the nominal background scaling would call for. In the case where the estimated flat background component is zero, i.e. the model predicts more counts from neighboring point sources in the background region than were actually observed ($N_p > N$), then we must assume that N_f and B_f are zero, and we have

$$c = \frac{B_p}{N_p * S}$$

D Standardized Directory Trees and Data Products

D.1 Input to AE

Many ACIS Extract (AE) tools (§7), such as *ae_make_catalog* and *ae_standard_extraction*, expect to find various input data products at standardized paths relative to the directory in which AE is executed. Below is a depiction of the data products needed for extracting point sources (§7.1.1) from ObsIDs named “1875” and “1876.”

Names of directories and files are in boldface; indentation and a trailing slash (/) represent directories. Notes on categories of directories/files are given in italics.

extract/

Data products related to each of the target’s Chandra observations (ObsIDs)

obs1875/ : ObsID 1875 data products

Primary inputs for point source extraction

spectral.evt : lightly-cleaned L2 event list (single ObsID)

obs.emap : exposure map (single ObsID)

Other standard data products from the CIAO archive⁸⁷

ardlib.par : “Analysis Reference Data Library” parameter file

obs.asol : “aspect solution” file

asphist/ : directory containing aspect histograms

obs.mskfile : “detector mask” file

obs.pbkfile : “parameter block” file

obs.eph : “ephemeris” file

Primary inputs for diffuse source extraction

diffuse.evt : heavily-cleaned L2 event list with point sources masked

diffuse.emap : exposure map with point sources masked

stowed.evt : “stowed background” event list

stowed.emap : exposure map for “stowed background” data

obs1876/ : ObsID 1876 data products

...

Multi-ObsID data products

target.evt : event list⁸⁸

target.emap : exposure map

⁸⁷http://cxc.harvard.edu/ciao/data_products_guide/

⁸⁸ Target-level events lists, built by merging event lists from multiple ObsIDs, are convenient for visualization but are not suitable for source extraction.

D.2 Point Source Output from AE

Our personal “recipes” for using AE (§7.1) to extract point sources generate data products with standardized names and directory structure. Below is a depiction of some of those data products, for point sources (§7.1.1) extracted from ObsIDs named “1875” and “1876.” Filenames ending in “.reg” are ds9 region files.

extract/

Summaries of the point source extractions from each ObsID

obs1875/ : ObsID 1875 data products

polygons.reg : extraction apertures for all sources observed by this ObsID

possibly_piled.srclist : list of sources at risk of photon pile-up

obs1876/ : ObsID 1876 data products

...

Point source extractions

point_sources/

Catalog summaries

all.srclist : ASCII table listing IAU names of point sources

label.txt : ASCII table pairing IAU names and colloquial source labels

xray_positions.reg : positions of point sources

polygons.reg : a representative extraction aperture for each point source

Pbslice.reg : point source extraction apertures, color coded by detection significance (§ 5.10.3)

psf_hook.reg : for bright sources, the expected location of the *Chandra* PSF hook (§ 7.14.7) in reconstructed images

tables/ : collations of source properties (§7.12)

catalog_and_photometry.collation : FITS table of point source properties, vector columns

xray_properties.fits : FITS table of point source properties, scalar columns; see Table 3 in Townsley et al. (2018)⁸⁹

xray_properties.pdf : PDF table of selected point source properties

observing_log.tex, **target_observing_log.pdf** : Observing Log (list of ObsIDs)

⁸⁹<https://doi.org/10.5281/zenodo.1067749>

Point source extractions (continued)

point_sources/

An extraction directory for each point source

053958.73-690610.3/ : extraction of point source “053958.73-690610.3”

ObsID_photometry.txt : table of ObsID-averaged photon flux and median energy

lightcurve.ps : PostScript plot of ObsID-averaged photon flux vs date

median_energy.ps : PostScript plot of ObsID-averaged median energy vs date

extractions.ps : PostScript image of events in source neighborhood in selected ObsIDs.

Directories produced by the MERGE_OBSERVATIONS stage

all.inclusive/ : merge of all ObsIDs

ObsIDs_merged.fits : list of the ObsIDs merged

neighborhood.evt : events in source neighborhood

Light curves (without background subtraction) and median energy time series

053958.73-690610.3.sequenced_lc.ps : each observation shown on a single “broken” time axis

053958.73-690610.3.stacked_lc.ps : each observation shown on separate time axes

source.lc : FITS lightcurve table

photometry/ : multi-ObsID merge, optimized for photometry and spectral fitting (§5.13).

ObsIDs_merged.fits : list of the ObsIDs merged

source.photometry : photometry for the merged ObsIDs

neighborhood.evt : events in source neighborhood

Spectra and calibration products, ready for XSPEC

053958.73-690610.3_bkg.pi : background spectrum, ungrouped

053958.73-690610.3.pi : extracted spectrum, ungrouped

053958.73-690610.3_grp4.0.pi : extracted spectrum, grouped (§5.7) to a signal-to-noise goal (here 4.0), suitable for fitting with χ^2 .

053958.73-690610.3.arf : ARF response file

053958.73-690610.3.rmf : RMF response file

XSPEC fitting (Spectra and models are NOT ACCURATE for piled sources.)

source.spectra : FITS tables collating all XSPEC models and fluxes

spectral_models/ : XSPEC fitting scripts, saved models, plots, etc.

best_model : symlink to preferred spectral fit

***.xcm** : fitting scripts

grp*/ : data products produced by fitting with the χ^2 statistic

ldata.ps : PostScript plot of grouped spectrum and model

icounts.ps : PostScript plot of cumulative spectrum and model

xspecl.log : Summary of model and fluxes.

summary.ps : **ldata.ps** +

xspecl.log

model.xcm : XSPEC save-file

model.txt : List of model parameters and fluxes (ASCII); also found in file **source.spectra**

xspecl_run.log : log of XSPEC session (difficult to read)

nogrp*/ : data products produced by fitting with the C-statistic

Directories produced by the MERGE_OBSERVATIONS stage (continued)

EPOCH_1875/ : single-ObsID “merge,” with possible pile-up correction⁹⁰.

Spectra and calibration products (as in *photometry*)

053958.73-690610.3_grp3.6.pi : Grouped spectrum, corrupted by pile-up.

source.photometry : Single-ObsID photometry, corrupted by pile-up.

recon_053958.73-690610.3_grp3.6.pi : Grouped spectrum, corrected for pile-up.

model.photometry : Single-ObsID photometry, corrected for pile-up.

recon_053958.73-690610.3_grp3.6.ps : PostScript plot showing observed and corrected spectra.

recon_053958.73-690610.3_grp3.6/ : Workspace for pile-up correction.

EPOCH_1876/ : single-ObsID “merge,” with possible pile-up correction.

...

position/ : multi-ObsID merge, optimized for position estimation (§5.13)

An extraction directory for each ObsID

1875/

source.evt.ps : PostScript scatter plot showing Time and Energy for each event

source.psf : local PSF

extract.reg : extraction aperture (contour of **source.psf**)

neighborhood.evt : events in source neighborhood

source.evt : events in extraction aperture

source.pi : spectrum of **source.evt**

source.arf, **source.rmf** : calibration files

background_pixels.reg, **background.evt**, **background.pi** : background region, event list, and spectrum

1876/

...

⁹⁰ Our tools for correcting pile-up in observed spectra (Broos et al., 2011) are not currently released. Contact Patrick Broos (patrick.broos@icloud.com) or Leisa Townsley (townsley@astro.psu.edu) to discuss pile-up correction.

D.3 Diffuse Source Output from AE

Our personal “recipes” for using AE (§7.1) to visualize and extract diffuse emission generate data products with standardized names and directory structure. Below is a depiction of some of those data products. Filenames ending in “.reg” are ds9 region files.

extract/ :

Single-ObsID masked event lists and exposure maps

obs1875/ : ObsID 1875 data products

diffuse.evt : observation event list with point sources masked

diffuse.emap : observation exposure map with point sources masked

obs1876/ : ObsID 1876 data products

...

adaptive_smoothing/ : flux images with point sources excised, adaptively smoothed

(Broos et al., 2010)⁹¹ (Townesley et al., 2011b)⁹²

full_band/ (0.5–7 keV)

Outputs from adaptive smoothing

sig015/tophat/ : “tophat” kernel; SNR=15

fullfield.diffuse.filled.flux : smoothed photon flux image

fullfield.diffuse.filled.signif : map of smoothing significance (SNR of flux values)

fullfield.diffuse.filled.radius : map of smoothing kernel radii

fullfield.diffuse.filled.reg : depiction of the median kernel size

Inputs to adaptive smoothing

fullfield.diffuse.img : merged masked observations

fullfield.bkg.img : instrumental background map (“stowed background data,” scaled)

fullfield.diffuse.emap : merged masked exposure maps

Other products

fullfield.diffuse.evt : merged masked observations

fullfield.img : merged unmasked observations

fullfield.emap : merged unmasked exposure maps

hard_band/ (2–7 keV)

...

soft_band/ (0.5–2 keV)

...

diffuse_sources/ : diffuse source extractions (§7.1.2)

extra_maskfile.reg : regions where extra masking was applied (e.g. to remove dust scattering halos, PWNe, SNRs), beyond the regular point source masking, prior to adaptive-kernel smoothing to generate diffuse images

diffuse_wvt_sig040/ : tessellation extractions (e.g. Townesley et al., 2011a)⁹³

⁹¹<http://adsabs.harvard.edu/abs/2010ApJ...714.1582B>

⁹²<http://adsabs.harvard.edu/abs/2011ApJS..194...16T>

⁹³<http://adsabs.harvard.edu/abs/2011ApJS..194...15T>

E Low-level AE Stages

AE has been evolving and improving for over a decade. As more and more automation has been added, some AE components are no longer commonly called directly, but are instead called by higher-level tools. For reference, this appendix retains the original description of several AE “stages” that are now rarely called directly.

E.0.1 Source Lists and Catalogs

The first input parameter to all stages of AE is the name of an ASCII file containing a list of sources that should be processed. Usually this list will include all the sources you’re working with, but in some stages it is useful to define subsets of sources that should be processed in different ways, e.g. sets of sources to be fit with different spectral models.

For the first stage (CONSTRUCT_REGIONS) of AE, this file must be a table consisting of 5 whitespace-separated columns specifying the fundamental parameters of each source:

1. source name (e.g. “182001.09-160717.1”); used to name the Unix directory that will hold the source’s data products. A typical source name would be constructed from the coordinates with this *IDL* expression:

```
strcompress(/REMOVE_ALL, adstring( ra, dec, 1, /TRUNCATE))
```

Note that when the “precision” parameter to `adstring()` is zero, standard CXC names⁹⁴ of the form “182001.9-160717” are built. In crowded fields where those names may not be unique observers may wish to use a precision value of 1 to build names of the form “181913.56-160129.9”.

2. RA (in degrees); 6 digits to the right of the decimal are recommended.
3. DEC (in degrees); 6 digits to the right of the decimal are recommended.
4. desired PSF fraction (< 1.0); controls the relative size of the extraction region.
5. “primary” PSF energy, i.e. energy (in keV) at which that PSF fraction is to be computed. A value of 1.5 keV is recommended for most sources since the observed spectrum generally peaks near there, however you are free to choose another energy.

We refer to this 5-column table as a “**catalog**”. All other stages of AE require only a list of source names. These stages will accept either a 5-column catalog, or a 1-column list of source names, referred to as a “**source list**”. In both catalogs and source lists, lines starting with a semicolon are interpreted as comments. Here’s an example of the format of a catalog:

```
; Comments start with semicolon.
181913.56-160129.9 274.806530 -16.024992 0.90 1.4967
181913.69-160110.5 274.807070 -16.019584 0.90 1.4967
; Comments can be anywhere.
181914.42-161234.6 274.810120 -16.209638 0.90 1.4967
181922.01-160314.9 274.841730 -16.054165 0.90 1.4967
```

E.1 CONSTRUCT_REGIONS Stage

This stage is run at least once for each observation. If a source’s position is revised this stage must be run again.

E.1.1 Actions

- Source names are constructed from the celestial coordinates provided⁹⁵, and source extraction directories are made.
- PSF images are constructed for each source at each of 5 energies.

⁹⁴<http://asc.harvard.edu/cdo/naming.html>

⁹⁵If the coordinates RA=0, DEC=0 are provided then AE looks for coordinates in an existing source.stats file.

- Source extraction regions that meet the requested PSF fraction at the specified “primary” PSF energy are proposed. Circular mask regions that completely cover the PSF (at the “primary” energy) are computed.
- Plots summarizing the results of CONSTRUCT_REGIONS are presented. These plots are diagnostic tools that help you spot problems in your data, in your catalog, in your *CIAO* configuration, in the *CIAO* software, or in the AE software. Some plots may also convey scientifically interesting information about your sources.

E.1.2 Usage and Inputs

```
acis_extract, catalog_filename, obsname, obsdata_filename, /CONSTRUCT_REGIONS,
    EMAP_FILENAME=filepath, ASPECT_FN=filepath, MASK_FRACTION=value,
    MASK_MULTIPLIER=value, /REGION_ONLY, /DIFFUSE
```

```
acis_extract, srclist_filename, obsname, /CONSTRUCT_REGIONS, /PLOT
```

EXAMPLE:

```
idl |& tee construct_regions.1875.log
```

```
acis_extract, 'all.cat', '1875', '../obs1875/spectral.evt', /CONSTRUCT_REGIONS,
    EMAP_FILENAME='../obs1875/obs.emap', ASPECT_FN='../obs1875/obs.asol'
```

```
acis_extract, 'all.srclist', '1875', /CONSTRUCT_REGIONS, /PLOT
```

In the above example, *IDL* is started with a tee to a log file, for documentation purposes.

- The keyword ASPECT_FN must specify a *CIAO* aspect file⁹⁶ covering the time range of your observation. The aspect file is provided to *MARX* during PSF construction so that the simulated source dithers across edges of the detector in precisely the same way as your real source. If *MARX*'s internal dither model was used the observed and simulated PSFs would show inconsistent distortions due to detector edges. *MARX* accepts only a single aspect file, not a “stack” of files. If your observation has multiple aspect files, you must concatenate them using *dmmmerge*, as discussed in the *CIAO* Data Products Guide⁹⁷. The aspect files must be combined in time order; the string of numbers in the file name (e.g. “pcadf111767021N001_asol1.fits”) refers to the start time of the period for which the aspect solution is valid.
- The optional keywords MASK_FRACTION (default 0.99) and MASK_MULTIPLIER (default 1.1) indirectly control the size of a circular mask region constructed around each source (used later for construction of a background event list). Choice of these parameters is largely a matter of scientific judgment based on the analysis goals. For example the generous mask defaults may be good for studies of faint diffuse emission where you want to be certain to remove point source wings. Reduction in the mask region sizes may be necessary in very crowded fields where very little background area survives the masking process.

Starting with a radius that encloses the extraction region, a circular region is enlarged until it encloses a PSF fraction of at least MASK_FRACTION. The mask region radius is then multiplied by MASK_MULTIPLIER. Due to the choice of starting radius one cannot produce a mask fraction less than the extraction fraction (assuming MASK_MULTIPLIER ≥ 1). The design assumption was that the mask region should be well outside the extraction region.

- The optional keyword /REGION_ONLY should be supplied if running this stage only for the purpose of constructing extraction regions (e.g. to display on the data). This option speeds up this stage by constructing only a single PSF image (at the fiducial energy). You must eventually run this stage without /REGION_ONLY to enable computation of PSF fractions.
- The optional keyword /DIFFUSE should be supplied when diffuse sources are being extracted (§7.1.2), i.e. when catalog_filename contains a list of diffuse source region files (without the “reg” part). The point source related keywords EMAP_FILENAME, MASK_FRACTION, MASK_MULTIPLIER, and /REGION_ONLY, are ignored.

⁹⁶<http://asc.harvard.edu/ciao/why/asol.html>

⁹⁷http://asc.harvard.edu/ciao/data_products_guide/asol_descrip.html

E.1.3 Data Products

For each source the following files are produced:

{sourcename}/source.stats: a FITS file containing keyword information about the source; view with **dmclist source.stats header**. If this file already exists it is modified, rather than destroyed and re-created (as of AE 3.106).

{sourcename}/{obsname}/extract.reg: a *DS9* region file containing source extraction and mask regions. The mask region is identified using the *DS9* region property “background”.

{sourcename}/{obsname}/source.psf: a FITS file containing PSF images at 277, 1497, 4510, 6400, and 8600 eV, placed at the source position

{sourcename}/{obsname}/obs.stats: a FITS file containing keyword information about this observation of the source; view with **bf dmclist obs.stats header**

Several interactive plots show characteristics of the PSF images and extraction apertures; see our AE recipes (§7.1) for descriptions of each. These plots are largely educational. However, experienced AE users are encouraged to continue scanning these plots because an atypical instance of plots like these may be our first indication of something wrong with a data set, with CIAO, or with the AE software.

E.2 EXTRACT_EVENTS Stage

This stage is a subset of the EXTRACT_SPECTRA Stage (§E.3). To save computer time it would be run (instead of running EXTRACT_SPECTRA) in preparation for the optional stage SHOW_REGIONS, as shown in Figure 8.

E.2.1 Actions

- Data from the neighborhood of the source are selected to help visualize the source in the SHOW_REGIONS stage.
- The PSF fraction associated with the source extraction region is computed for all the energies at which a PSF is available. A warning is printed if the source extraction region falls outside the field of view of the PSF images so the observer will know that the PSF fraction is underestimated.
- The observation data is filtered by the source extraction region to produce a source event list.
- The source events are energy filtered and various statistics are computed on the in-band events and saved as FITS keywords in obs.stats including:
X_DATA, Y_DATA: the mean sky coordinates of the in-band extracted events.

EX_DATA, EY_DATA: 1- σ error estimates for X_DATA, Y_DATA.⁹⁸

- Plots summarizing the results of EXTRACT_EVENTS are presented. These plots are diagnostic tools that help you spot problems in your data, in your catalog, in your CIAO configuration, in the CIAO software, or in the AE software. Some plots may also convey scientifically interesting information about your sources.

The source extraction region (`{sourcename}/{obsname}/extract.reg`) is of course normally a polygon constructed by the CONSTRUCT_REGIONS stage. Any CIAO-compatible region file may be used however, including ones containing multiple components using the “+” and “-” CIAO syntax. For example, to extract a source suffering pile-up, an observer may replace AE’s region with an annular region. Whatever region is found in `extract.reg` is used both to filter the event data and to filter the PSF images to compute a PSF fraction curve.

E.2.2 Usage and Inputs

acis.extract, srclist.filename, obsname, obsdata.filename, /EXTRACT_EVENTS,
ENERGY_RANGE=[*energy,energy*], EMAP.FILENAME=*filepath*,

⁹⁸ These “standard errors” of the means X_DATA, Y_DATA are estimated as

$$EX_DATA = \frac{\sigma_{X,parent}}{\sqrt{N}}$$

$$EY_DATA = \frac{\sigma_{Y,parent}}{\sqrt{N}}$$

where $\sigma_{X|Y,parent}$ is the standard deviation of the parent distribution for the in-band extracted events. That parent distribution is (ignoring background) assumed to be the source’s PSF truncated by the extraction region and thus $\sigma_{X|Y,parent}$ can be calculated “perfectly”.

This simple σ/\sqrt{N} calculation for the position errors is correct if the parent distributions are Normal, and is asymptotically correct for any parent distribution as N gets large. For non-normal PSFs with few counts I do not know a better estimate for the position errors.

Note that an alternate approach (used in earlier AE versions) would be to estimate the variances of the parent distribution from the extracted count data themselves, then compute a confidence interval on our sample mean with the aid of the Student’s T distribution (which accounts for the imperfect variance estimates for the parent distribution). This is a poor approach however because one often has very few extracted counts which can produce wildly inaccurate estimates for the variances of the parent distribution. We prefer to make use of the information on the parent distribution carried in the truncated PSF.

Note that the error estimation above is concerned only with uncertainty related to counting statistics. Even if you had a very large number of counts there are at least two sources of systematic position error:

- The parent distribution of the data in our extraction region is NOT simply our model PSF image truncated by the extraction region. Our model PSF image is by construction aligned to the extraction region, whereas the actual star’s PSF is assumed to be somewhat **misaligned** to the extraction region—that’s why we’re trying to estimate a better position in AE. Thus iterative adjustment of the source position and re-construction of the extraction region is probably a good idea.
- Off-axis PSFs are asymmetric and the position of the celestial source does not correspond to the centroid of the PSF. This is why correlation of the data with the source’s PSF (§7.9) is a useful technique for estimating source positions.

WARNING_REGION_FILENAME=*filepath*, NEIGHBORHOOD_SIZE=*value*,
/REUSE_NEIGHBORHOOD

acis_extract, srclist_filename, obsname, /EXTRACT_EVENTS, /PLOT

EXAMPLE:

idl |& tee extract_events_1875.log

acis_extract, 'all.srclist', '1875', '../obs1875/spectral.evt', /EXTRACT_EVENTS, ENERGY_RANGE=[0.5,8],
EMAP_FILENAME='../obs1875/obs.emap', WARNING_REGION_FILENAME='warning.reg'

acis_extract, 'all.srclist', '1875', /EXTRACT_EVENTS, /PLOT

- The optional ENERGY_RANGE parameter defines the band over which statistics are computed; default is [0.5,8.0].
- The optional WARNING_REGION_FILENAME parameter names a *CIAO* region file that defines regions of the field on which sources will require special attention. AE will determine what fraction of the source events fall in the “warning regions”, and report the fraction in the keyword WARNFRAC in obs.stats.
- The optional NEIGHBORHOOD_SIZE parameter specifies the minimum width/height (in arcseconds) of the source neighborhood that is extracted; default is 50.
- The optional keyword /REUSE_NEIGHBORHOOD directs AE to use any existing neighborhood.evt file it finds from a previous run, saving significant time for an observation with many events.
- If the optional parameter /ONLY_EDITED is set the only sources processed are those for which the extraction region file (extract.reg) appears to have been edited by *DS9* (e.g. during the SHOW_REGIONS stage). This is merely a convenience for the situation where reprocessing a complete catalog would take a long time and making a list of the extraction regions you’ve edited would be a hassle.

E.2.3 Data Products

For each source the following files are produced:

{sourcename}/{obsname}/neighborhood.evt: a FITS event list covering the immediate neighborhood of the source

{sourcename}/{obsname}/obs.psffrac: a FITS file containing a table of PSF fractions at all the energies for which a PSF was computed.

{sourcename}/{obsname}/source.evt: the events falling in the source extraction region

{sourcename}/{obsname}/obs.stats: a FITS file containing keyword information about this observation of the source

Several interactive plots show characteristics of the source extraction; see our AE recipes (§7.1) for descriptions of each.

E.3 EXTRACT_SPECTRA Stage

This stage is run for each observation to perform the final spectral extraction in preparation for spectral fitting, as shown in Figure 8.

E.3.1 Actions

- All actions from the EXTRACT_EVENTS stage (§E.2) are performed.
- Source spectra are extracted.
- Statistics that characterize the exposure map pixels that fall in the extraction region are saved in the keywords EMAP_NUM, EMAP_AVG, EMAP_MED, EMAP_MIN, EMAP_MAX.
- ARF and RMF files are constructed. **Note that AE will re-use any source RMFs existing from an earlier extraction in order to avoid long-running calls to the tool *mkacisrmf*.**
- Plots summarizing the results of EXTRACT_SPECTRA are presented. These plots are diagnostic tools that help you spot problems in your data, in your catalog, in your *CIAO* configuration, in the *CIAO* software, or in the AE software. Some plots may also convey scientifically interesting information about your sources. For historical reasons some of the plots in this stage use information from the TIMING stage, so run EXTRACT_SPECTRA then TIMING then EXTRACT_SPECTRA with the /PLOT option.

E.3.2 Usage and Inputs

The *CIAO* tools used in this stage require that you configure *ardlib.par*⁹⁹ for the observation you’re processing:

- The *ardlib.par* file must point to an appropriate bad pixel file¹⁰⁰, which is used by *mkarf*. Generally each observation has its own bad pixel file, which may have been augmented by the observer during data reduction. The *CIAO* tool *acis_set_ardlib* will help you configure an *ardlib.par* file for your observation.
- The AXAF_ACIS?.QEU_FILE parameters in *ardlib.par* must specify an appropriate set of QEU (quantum efficiency uniformity) files, which are used by *mkarf*. If the data have been through the Townsley et al. CTI corrector¹⁰¹ then the QEU files supplied with that corrector must be specified (e.g. with *pset*), rather than the standard value of “CALDB”. At PSU, these files are */usr/common/Townsley_CTI_corrector/public/120C/ccd?_120.qeu[1]* or */usr/common/Townsley_CTI_corrector/public/110C/ccd?_110.qeu[1]*.
- The AXAF_RMF_FILE parameter in *ardlib.par* must specify an appropriate source for constructing RMF files. Normally, this is set to its default value, ‘CALDB’. However, if the data have been through the Townsley et al. CTI corrector¹⁰² then AXAF_RMF_FILE should be set (e.g. with *pset*) to the path to a directory that contains the appropriate set of RMF files supplied with the CTI corrector. (At Penn State that path is */usr/common/Townsley_CTI_corrector/public/120C/* or */usr/common/Townsley_CTI_corrector/public/120C/*)

AE will print the relevant parts of *ardlib.par* and pause briefly **for you to review these, to make sure that the correct *ardlib.par* file is being referenced and that it contains the correct bad pixel, QEU, and RMF entries.**

```
acis_extract, srclist_filename, obsname, obsdata_filename, /EXTRACT_SPECTRA,  
    ENERGY_RANGE=[energy,energy], EMAP_FILENAME=filepath, ASPECT_FN=filepath,  
    ASPHIST_DIR=directory, ARDLIB_FILENAME=filepath, PBKFILE=filepath, MSKFILE=filepath,  
    WARNING_REGION_FILENAME=filepath, NEIGHBORHOOD_SIZE=value,  
    /REUSE_NEIGHBORHOOD, /USE_MKRMF, EMAP_ENERGY=energy,  
    WMAP_ENERGY_RANGE=energy
```

⁹⁹<http://cxc.harvard.edu/ciao/ahelp/ardlib.html>

¹⁰⁰<http://cxc.harvard.edu/ciao/threads/badpix/>

¹⁰¹<http://www.astro.psu.edu/users/townsley/cti/>

¹⁰²<http://www.astro.psu.edu/users/townsley/cti/>

acis_extract, srclist_filename, obsname, /EXTRACT_SPECTRA, /PLOT

EXAMPLE:

```
idl |& tee extract_spectra_1875.log
```

```
acis_extract, 'all.srclist', '1875', './obs1875/spectral.evt', /EXTRACT_SPECTRA, ENERGY_RANGE=[0.5,8],  
    EMAP_FILENAME='./obs1875/obs.emap', ASPECT_FN='./obs1875/obs.asol',  
    ASPHIST_DIR='./obs1875/asphist', ARDLIB_FILENAME='./obs1875/ardlib.par',  
    PBKFILE='./obs1875/obs.pbkfile', MSKFILE='./obs1875/obs.mskfile',  
    WARNING_REGION_FILENAME='warning.reg'
```

```
acis_extract, 'all.srclist', '1875', /EXTRACT_SPECTRA, /PLOT
```

- The optional ENERGY_RANGE parameter defines the band over which statistics are computed; default is [0.5,8.0]. An energy filter is NOT applied to the spectra.
- The optional parameter ASPECT_FN is recommended so that various *CIAO* tools (e.g. dmcoords, mkacisrmf, mkwarf) can be supplied with an aspect file.
- The ASPHIST_DIR should be the path to a directory that contains appropriate aspect histogram files for each active CCD with names ccd0.asphist, ccd1.asphist,... Each CCD must have its own aspect histogram file because the tool *asphist* considers a CCD's GTI table when computing the aspect histogram. If omitted the directory *./asphist/* is assumed. Methods of constructing aspect histograms are discussed in the *CIAO* documentation. If you use the *CIAO* script *merge_all* to construct **SINGLE ObsId** exposure maps you can rename the aspect histograms files it produces, e.g. merged_asp_0.fits → ccd0.asphist. If you use the AE tool *ae_make_emap* (§7.14) to construct exposure maps then an aspect histogram directory named *asphist* with the correct file names is produced.

AE confirms that the exposure times represented in the aspect histograms supplied are similar to the EXPOSUR* keywords in the event data as a potential aide in catching mistakes in aspect histogram construction.

- The required ARDLIB_FILENAME should be the path to an *ardlib.par* parameter file that has been correctly configured for this observation. **To avoid mistakes we STRONGLY recommend that you store a correctly configured *ardlib.par* in the area where you reduced the observation, rather than in the default parameter directory referenced in your shell's PFILES environment variable.**
- The optional PBKFILE parameter enables the “Dead Area calibration” available in the *mkarf/mkwarf* tools. See the *ahelp* documentation in *CIAO* or the appropriate *CIAO* thread¹⁰³.
- The optional MSKFILE parameter should be the path to the observation's “mask file”¹⁰⁴, which is used by *mkarf* and *mkwarf*.
- See §E.2 for a description of the WARNING_REGION_FILENAME, NEIGHBORHOOD_SIZE, and REUSE_NEIGHBORHOOD parameters.
- Supply /USE_MKRMF to build RMFs using the *CIAO* tool *mkrmf* instead of the tool *mkacisrmf*. The CXC recommends¹⁰⁵ the *mkrmf* tool only for data taken at -110 C on certain CCDs.
- The keyword parameter EMAP_ENERGY **must be supplied for diffuse sources** (§7.1.2); it should specify the mono-energy (in keV) that was used to build the exposure map (§7.14.3). You can look up this energy in the headers of the “instrument maps” produced by *CIAO*, for example

```
dmlist asphist/ccd0.instmap head | egrep "ENERG_LO|ENERG_HI"
```

- The optional keyword parameter WMAP_ENERGY_RANGE **can be supplied for diffuse sources** (§7.1.2) to specify the energy range (in keV) over which a “weight map” (needed by *mkwarf*) is constructed ; the default range is 0.5–2 keV.

¹⁰³<http://cxc.harvard.edu/ciao/why/acisdeadarea.html>

¹⁰⁴http://cxc.harvard.edu/ciao/threads/intro_data/index.html

¹⁰⁵<http://asc.harvard.edu/ciao/why/mkacisrmf.html>

E.3.3 Data Products

For each source the following files are produced:

All data products from the `EXTRACT_EVENTS` stage (§E.2) are also produced by `EXTRACT_SPECTRA`.

{sourcename}/{obsname}/source.pi: the spectrum of the source

{sourcename}/{obsname}/source.rmfi: the RMF at the source position for the CCD containing the majority of the band-limited events in `source.evt`

{sourcename}/{obsname}/sourcei.arfi: the ARF at the source position for CCD *i*

{sourcename}/{obsname}/source.arf: the multi-CCD, PSF fraction reduced ARF at the source position

{sourcename}/{obsname}/obs.stats: a FITS file containing keyword information about this observation of the source

Several interactive plots show characteristics of the source extraction; see our AE recipes (§7.1) for descriptions of each.

E.4 TIMING Stage

This stage is run for each observation to perform timing analyses, as shown in Figure 8.

BEWARE: the timing analysis in this stage makes no distinction between events produced by the source and background events. Variation in the aperture background rate within an ObsID can lead to a spurious indication of source variability.

E.4.1 Actions

- A “grouped” light curve (i.e. a light curve with unequal bin sizes) is constructed. The median energy of events in each time bin is computed. Exposure in each time bin is computed using the appropriate GTI table.
- Errors on the light curve and median energy values are estimated.
- An adaptively smoothed light curve is computed.
- Time variability of the source within the specified observation is quantified by comparing a uniform *count rate* model to the distribution of source event time stamps and computing the 1-sample Kolmogorov–Smirnov statistic. The p-value of this single-ObsID KS statistic is saved in the keyword PROB_KS in obs.stats; values near 0 suggest variability. The code accounts for gaps in the observation using the GTI tables.

E.4.2 Usage and Inputs

```
acis_extract, srclist_filename, obsname, /TIMING, ENERGY_RANGE=[energy,energy],  
SNR_RANGE=[value,value], NUM_GROUPS_RANGE=[value,value]
```

EXAMPLE:

```
idl |& tee timing_1875.log
```

```
acis_extract, 'all.srclist', '1875', /TIMING, ENERGY_RANGE=[0.5,8], SNR_RANGE=[2,20], NUM_GROUPS_RANGE=[2,100]
```

- The optional ENERGY_RANGE parameter defines the energy band over which the light curve is computed; default is [0.5,8.0].
- The optional keywords SNR_RANGE and NUM_GROUPS_RANGE change the time binning parameters. See §5.7 for the meaning of these parameters, each of which is a 2-element vector.

E.4.3 Data Products

For each source the following files are produced:

{sourcename}/{obsname}/source.lc: a FITS table containing the grouped light curve, grouped median energy time-series, exposure time series, and adaptively smoothed light curve. The exposure values in this table attempt to account for both the integration time within each time bin (obtained from the GTI tables) and the ARF value (effective area and PSF fraction) associated with this ObsId. This facilitates testing for variability across ObsIds in the MERGE stage.

{sourcename}/{obsname}/source.evt.ps: a PostScript file containing a scatter plot of event arrival times and energies, and a plot of the observed and uniform cumulative distribution functions used to compute KS probability (Figure 18).

To accommodate multiple observations, the binned and smoothed light curves are plotted in the MERGE.OBSERVATIONS stage.

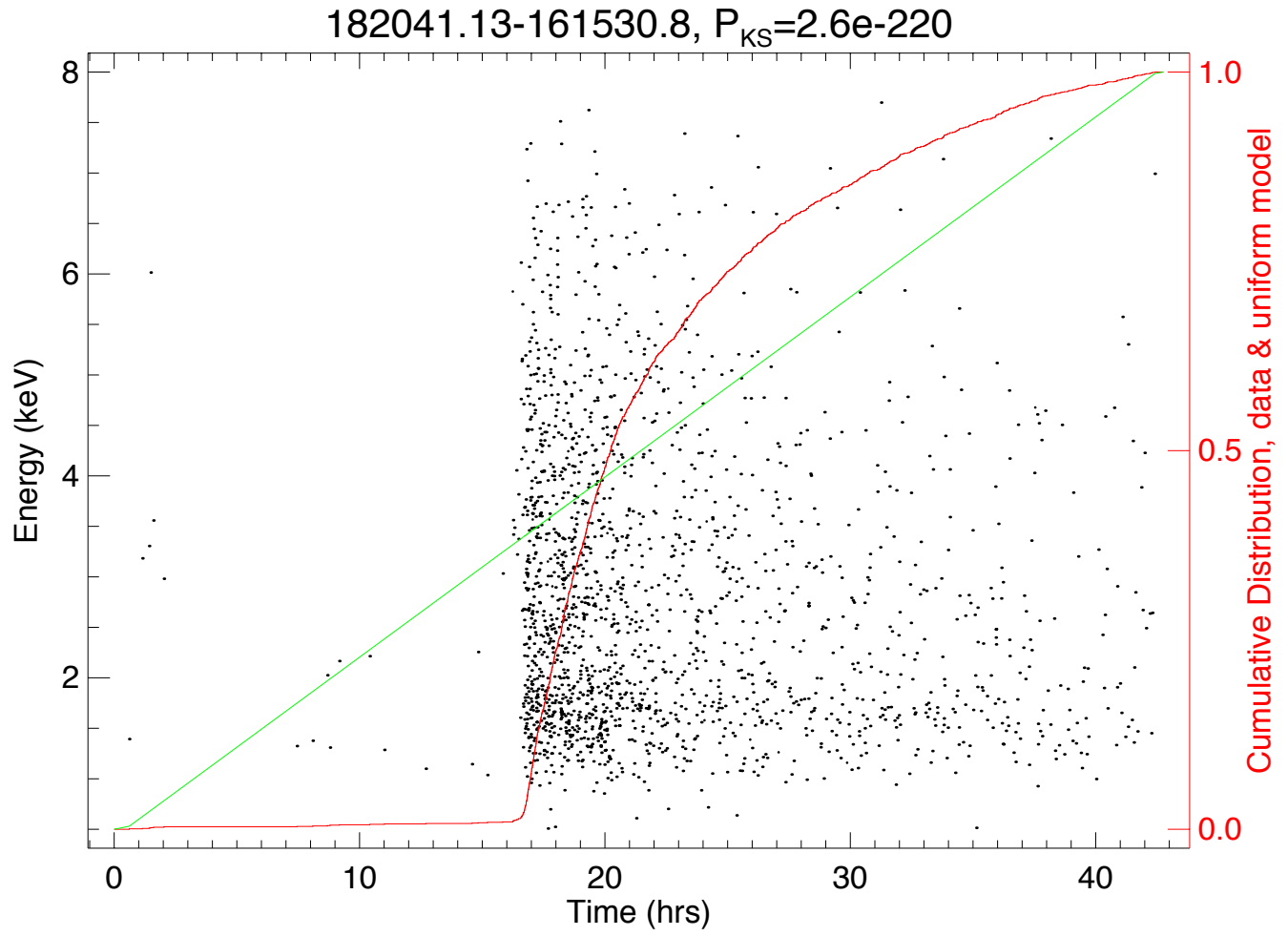


Figure 18: Scatter plot of event arrival times and energies, and a plot of the observed and uniform cumulative distribution functions used to compute KS probability.

E.5 EXTRACT BACKGROUNDS Stage

This stage is run for each observation to perform the final background spectral extraction in preparation for spectral fitting, as shown in Figure 8.

E.5.1 Actions

- For each source, a search is performed to find a local circular background region using the following criteria:
 1. The background scaling (BACKSCAL) associated with the region **is constrained** to fall within the target range defined for the source (§5.6 and 7.7).
 2. The algorithm **tries** to achieve a background scaling close to the goal defined for the source (§5.6 and 7.7).
 3. The algorithm **tries** to achieve a background region that contains at least the specified minimum number of total-band counts (default is 5 counts, modified via MIN_NUM_CTS option).

The observer may supply an arbitrary¹⁰⁶ background region for an individual source by placing the region specification in a file named *background.reg* in the source's extraction directory for this observation (e.g. src-name/obs/background.reg). When a background region file is supplied then the two criteria above are ignored. Note that for a source observed multiple times, there are theoretical reasons to design the background regions so that the scaling values they require with respect to their source apertures are similar (§5.6.3).

- Background event lists are extracted.

Note that the background region (whether an auto-generated circle or an observer-supplied file) is NOT directly used to filter the event list. In order to accurately measure the exposure-area of the background spectrum, the background region specification is applied to the masked exposure map supplied, and the emap pixels that are found to be “inside” the *DS9/CIAO* region define the actual set of events that form the background spectrum. Thus, the effective background region is a pixelized (as defined by the exposure map) version of the *DS9/CIAO* region.

- Background spectra are extracted, and scaled using the integrated exposure in the region.
- Plots summarizing the results of EXTRACT BACKGROUNDS are presented. These plots are diagnostic tools that help you spot problems in your data, in your catalog, in your *CIAO* configuration, in the *CIAO* software, or in the AE software. Some plots may also convey scientifically interesting information about your sources.

E.5.2 Usage and Inputs

```
acis_extract, srclist_filename, obsname, background_obsdata_filename, /EXTRACT BACKGROUNDS,  
EMAP_FILENAME=filepath, MIN_NUM_CTS=value, ENERGY_RANGE=[0.5,8]
```

```
acis_extract, srclist_filename, obsname, /EXTRACT BACKGROUNDS, /PLOT
```

EXAMPLE:

```
idl |& tee extract_backgrounds_1875.log
```

```
acis_extract, 'all.srclist', '1875', '../obs1875/background.evt', /EXTRACT BACKGROUNDS,  
EMAP_FILENAME='../obs1875/background.emap', MIN_NUM_CTS=20
```

```
acis_extract, 'all.srclist', '1875', /EXTRACT BACKGROUNDS, /PLOT, REGION_FILE='background.reg'
```

- The *obsdata_filename* and *emap_filename* should be paths to an event list and exposure map in which **all point sources have been removed**; see §7.6.2, 7.6.3.

¹⁰⁶Note that *CIAO* 2.3 cannot parse region files with more than 1024 characters. To work around this limitation, the supplied background region file may be reduced in size in AE by reducing the number of vertexes in polygon components.

- The optional keyword MIN_NUM_CTS (default 5) should be the minimum number of in-band counts you *prefer* to have in the background spectrum for this single observation. The number of counts you wish to have in the final multi-ObsID background spectrum must be controlled via the MIN_NUM_CTS parameter to the *ae_adjust_backscal_range* tool (§5.6 and 7.7).
- The optional ENERGY_RANGE parameter defines the band over which statistics are computed; default is [0.5,8.0]. An energy filter is NOT applied to the spectra.

E.5.3 Data Products

For each source the following files are produced:

{sourcename}/{obsname}/background.emap: the background exposure map (which defines the background region)

{sourcename}/{obsname}/background.evt: the events falling in the background region

{sourcename}/{obsname}/background.pi: the spectrum corresponding to background.evt

{sourcename}/{obsname}/obs.stats: a FITS file containing keyword information about this observation of the source

Several interactive plots show characteristics of the source extraction; see our AE recipes (§7.1) for descriptions of each.

The optional REGION_FILE parameter to the /PLOT stage specifies the name to use for a region file AE will construct containing the background regions for all the sources in the catalog.

E.6 *ae_better_backgrounds* Tool

E.6.1 Usage and Inputs

This tool requires the standardized directory structure and file naming convention used in §3 and in our AE recipes (§7.1). This tool must estimate source photometry; thus **prior to running this tool you must extract source spectra**. As described above, this tool computes backgrounds that are tailored to the source extraction regions; thus **if you modify extraction regions you should rerun *ae_better_backgrounds***.

We strongly recommend an exposure map pixel size of 1 sky pixel or smaller when using this tool.

```
ae_better_backgrounds, obsname, EVTFILE_BASENAME=name, BACKGROUND_MODEL_FILENAME=filepath,  
    EXTRACTION_NAME=name, SAVEFILE_BASENAME=filepath, /REUSE_MODELS,  
    SRCLIST_FILENAME=filepath, EMAP_BASENAME=name, MIN_NUM_CTS=value,  
    BACKGROUND_IMBALANCE_THRESHOLD=value, COMPACTNESS_GAIN=value,  
    /PAUSE_FOR_REVIEW, VERBOSE=value, THETA_RANGE=[value,value]
```

EXAMPLE:

```
idl |& tee ae_better_backgrounds_1875.log
```

```
.run ae ae_better_backgrounds, '1875', EVTFILE_BASENAME='spectral.evt'
```

This tool builds and stores models for all the sources in the catalog using single-observation photometry calculated by the MERGE-OBSERVATIONS stage of AE, then uses those source models to construct background regions and spectra.

Only two parameters are required, as shown above.

- The tool uses the observation name (obsname) parameter to construct a directory path `../obsXXXX/` where it expects to find various observation data products (with the standard names used by our AE recipe).
- The EVTFILE_BASENAME parameter specifies the filename for the event list (in `../obsXXXX/`) from which you wish to extract background spectra. This event list should be the same one from which you extracted source spectra, and should **not** have the point sources masked.

The optional parameters that may be commonly useful are:

- The BACKGROUND_MODEL_FILENAME parameter can be used to supply a model (FITS image) of any background component not represented by the point sources in the catalog. The FITS image supplied should be in units of observed counts (i.e. the effects of exposure variation should be represented in the model). As of January 2009 this option has not been tested very well.
- The EXTRACTION_NAME parameter is described in §7.3.

For unusual circumstances, these optional parameters are provided:

- The initialization of the algorithm's data structures (source models) takes considerable time. These data structures are saved in an IDL "save file" named "ae_better_backgrounds.sav" by default; a different name can be supplied via SAVEFILE_BASENAME. If /REUSE_MODELS is supplied then these data structures are restored from the file rather than constructed. This mechanism allows you to run *ae_better_backgrounds* on a sub-catalog containing just a few sources using a complete set of source models that were saved from a prior run of *ae_better_backgrounds* on the full catalog.
- By default the source list is read from `../obsXXX/all.cat`; you can specify a different file path via the optional SRCLIST_FILENAME parameter.
- By default the input exposure map is read from `../obsXXX/obs.emap`; you can specify a different file name in `../obsXXX/` via the optional EMAP_BASENAME parameter.
- The keyword MIN_NUM_CTS (default 5) specifies the minimum number of counts you desire in the background spectrum for this single observation.

- The `BACKGROUND_IMBALANCE_THRESHOLD` parameter is a unit-less threshold on the background imbalance metric described earlier.
- The `COMPACTNESS_GAIN` parameter is a unit-less parameter that adjusts the relative importance of the compactness term in the background region metric we are minimizing.
- The `VERBOSE` and `/PAUSE_FOR_REVIEW` parameters can be used to see more details of the computations.
- The `THETA_RANGE` parameter can be used to select a subset of sources for which background regions will be defined, based on their off-axis angles. One might take the trouble to reduce processing time by using this capability to process on-axis sources with an exposure map binned finely and to process off-axis sources with an exposure map binned coarsely.

Data Products For each source the following files are produced:

{sourcename}/{obsname}/background_pixels.reg: a *DS9* region file marking the exposure map pixels which define the background region. This region file is displayed by the `SHOW_REGIONS` stage (§7.10).

{sourcename}/{obsname}/background.emap: the background exposure map (which defines the background region)

{sourcename}/{obsname}/background.evt: the events falling in the background region

{sourcename}/{obsname}/background.pi: the spectrum corresponding to background.evt

{sourcename}/{obsname}/obs.stats: a FITS file containing keyword information about this observation of the source

F Additional Release History

F.1 AE Version 2016feb1

- As described in the 2015jul7 release notes, *AE* versions prior to 2015jul7 are not compatible with *CIAO* versions 4.7 and later.
- I have reviewed the release notes for *CIAO* 4.8 and have run *AE* with it. I believe *CIAO* 4.8 is compatible with *AE* version 2015jul7 and later.
- If you have projects that involve multiple ObsIDs, then you should be careful when mixing ObsIDs that were processed with different versions of *CIAO*. *CIAO* 4.8 changed the default ordering of the columns in event lists produced by the tool *acis_process_events*. The *dmmerge* tool will not accept inputs that have different column ordering. Thus, the MERGE stage of *AE* cannot handle ObsIDs that have different column ordering.
If you have an *AE* extraction of ObsIDs that used the original column ordering, then you should be sure to retain that ordering if you have to reprocess any of those ObsIDs, or if you add a new ObsID to the project. To retain the original column ordering, simply set the *acis_process_events* parameter *eventdef* to
“d:time,s:ccd_id,s:node_id,l:expno,s:chip,s:tdet,f:det,f:sky,s:phas,l:pha,l:pha_ro,f:energy,l:pi,s:fltgrade,s:grade,x:status”.
- I have reviewed the release notes for *MARX* 5.1 and for *MARX* 5.2 and have run *AE* with *MARX* 5.2. I believe both those *MARX* versions are compatible with *AE* version 2014feb5 and later.
- **Note that installing *MARX* on OS-X (the Mac operating system) is currently a complex task that deserves your careful attention.** On many Macs, *MARX* will install without incident but will not operate correctly with certain inputs. The problem seems to be bugs in Apple’s latest C compiler (called *clang*). See <http://space.mit.edu/cxc/marx/inbrief/install.html#mac-os-x> for details and instructions or contact marx-help@space.mit.edu. **Note that this issue arises whether *MARX* is installed by hand or with the *CIAO* tool *install_marx*.**
- *AE* has begun using some capabilities introduced in *IDL* version 8.0 (released in 2010). In this *AE* release, only a few tools are affected, but over time *AE* is expected to become more and more incompatible with *IDL* versions prior to 8.0.
- This release includes the most recent versions of the scripts and recipe documents that we use locally for our own *AE* processing (§7.1). These are continuously improved as we use them in our own ACIS projects.
Recall that the workflow implemented by these scripts and recipe documents is very complex and conservative; many *Chandra* observers will be happy with a more straight-forward single-pass extraction of a fixed catalog, e.g. as shown in the Getting Started section (3).

F.2 AE Version 2015jul7

- **AE may not be compatible with MARX 5.1!** Since I have not yet installed or tested *MARX* 5.1, you are strongly advised not to use that version of *MARX* with *AE*.
- This version of *AE* is compatible with *CIAO* 4.7. **However, using *CIAO* 4.7 with all previous versions of *AE* can, under certain circumstances, significantly damage your data analysis.**

The problem is that *CIAO* 4.7 altered the values of certain astrometry-related keywords in the aspect files produced by the *wcs_update* tool. (Note that *wcs_update* is called by the *AE* script *adjust_astrometry.csh* and by the *CIAO* tool *reproject_aspect*.) Those header changes lead to an astrometric offset in the PSF images that *AE* builds using *MARX*. That offset damages your data analysis in a variety of ways.

If you have ever modified your aspect files using the *CIAO* 4.7 version of the tool *wcs_update*, then your *AE* extractions are at risk. You should contact me, and we will discuss how to judge whether your data are affected and how to patch your project. This issue was first announced in an email to the *AE* user’s group on April 30.

This and future versions of *AE* address this *CIAO*/*AE* incompatibility in two ways. First, the *AE* code that builds PSFs will now produce an error message and halt if it finds that the aspect file’s header does not conform to

AE's expectations. Second, the AE script *adjust_astrometry.csh*, after running *wcs_update*, now restores certain keywords in the aspect file header to the values AE expects.

- AE Version 2014sep10 expanded the capabilities of the tool *ae.interObsID_astrometry*, which is used in one of our local recipes (*validation_procedure.txt*) for estimating the astrometric offsets between each ObsID and a reference catalog. An implementation mistake causes that version of *ae.interObsID_astrometry* to recommend shift estimates that are not the most accurate available.

A patch to this tool was released on 2014 October 15, and is included in this release.

- This release includes the most recent versions of the scripts and recipe documents that we use locally for our own AE processing (§7.1). These are continuously improved as we use them in our own ACIS projects.

Recall that the workflow implemented by these scripts and recipe documents is very complex and conservative; many *Chandra* observers will be happy with a more straight-forward single-pass extraction of a fixed catalog, e.g. as shown in the Getting Started section (3).

- Supplying a *CIAO* orbit ephemeris file to the TIMING stage, via the optional input ORBITFILE, now applies a barycentric correction to the timestamps of the extracted events.
- The (undocumented) options PAGE_LONG_DIM and PAGE_SHORT_DIM have been added to the MERGE stage to control the size of lightcurve plots.

F.3 AE Version 2014sep10

- **AE has been revised to run under CIAO 4.6.** Note that several tools in *CIAO* 4.6 no longer use a parameter-block filename parameter (*pbkfile*). To support observers using *CIAO* 4.5, AE continues to supply this parameter to those tools. Note that *CIAO* 4.6 may require some older event files to be reprocessed or patched (with the *r4_header_update* script); see the *CIAO* 4.6 release notes¹⁰⁷.
- The scripts and recipe documents that we use locally for our own AE processing (§7.1) are now less vulnerable to failures caused by race conditions between communicating IDL processes, and failures caused by buffer-overflow problems when IDL commands are pasted into terminal windows.

Recall that the workflow implemented by these scripts and recipe documents is very complex and conservative; many *Chandra* observers will be content with a more straight-forward single-pass extraction of a fixed catalog, e.g. as shown in the Getting Started section (3).

- AE now requires the “2014sep10” or later version of the *TARA* package¹⁰⁸.

F.4 AE Version 2014feb17

- The scripts and recipe documents that we use locally for our own AE processing (§7.1) have been improved and repaired since the previous release. We are using AE continuously right now, and these recipes may continue to evolve for a while longer. Recall that the workflow implemented by these scripts and recipe documents is very complex and conservative; many *Chandra* observers will be content with a more straight-forward single-pass extraction of a fixed catalog, e.g. as shown in the Getting Started section (3).
- The run time of the *ae.better_backgrounds* tool has been significantly reduced.
- **We still have *not* tested AE under CIAO 4.6.**
- AE now requires the “2014feb16” or later version of the *TARA* package¹⁰⁹.

¹⁰⁷<http://cxc.harvard.edu/ciao/releasenotes/ciao.4.6-release.html>

¹⁰⁸<http://personal.psu.edu/psb6/TARA/code//>

¹⁰⁹<http://personal.psu.edu/psb6/TARA/code//>

F.5 AE Version 2014feb5

- **AE now requires version 5.0** of the *MARX*¹¹⁰ simulator. This version of *MARX* explicitly models the PSF blurring effects that occur when the *CIAO* tool *acis_process_events* estimates the position of an event in CHIP coordinates (§5.1.2). In previous AE versions, *MARX* was used to model only the HRMA; other blurring effects were modeled by AE.
- **We recommend *CIAO* 4.5 for this release of AE. It has NOT been tested under *CIAO* 4.6.**
- The PIPELINE_RANDOMIZATION input to AE is no longer needed and has been removed.
- Our own recipe documents for using AE (§7.1) have been significantly revised to reduce the observer’s cut-and-paste workload, particularly for projects with large numbers of ObsIDs. We believe this revised recipe also has a less confusing workflow and is generally easier to understand. These recipes probably require the “2014jan27” or later version of the *TARA* package¹¹¹.
- The tool *ae_radial_profile* (§7.14.6) has been improved in various ways.
- The tool *ae_make_emap* can now build exposure maps for 0th-order grating data.
- As of Oct. 2013, the *ae_chart_interface* tool has been removed from AE because *ChART* cannot simulate PSFs for data processed with the EDSER sub-pixel positioning algorithm, which is now the default in the *Chandra* pipeline.

F.6 AE Version 2013mar6

- The XSPEC script we distribute for fitting two-temperature thermal plasma models has been producing incorrect 2-component flux values, reported in the collation table columns FCH8, FCH2, FC28, since Jan 27 2011 (version 3852 of *tbabs.2vapec.xcm*). The single-component fluxes are each ok, and so the total flux could be recomputed post-facto by summing the two components. This script has been repaired.
- **This release of AE is NOT yet compatible with MARX 5.0. I will certainly let you know when I have updated AE, but I do not have a good estimate of when that will occur.**

F.7 AE Version 2012nov1

- AE now requires the “2012nov1” or later version of the *TARA* package¹¹².
- **This release of AE is NOT yet compatible with MARX 5.0. I will certainly let you know when I have updated AE, but I do not have a good estimate of when that will occur.**
- We have created a tool (*ae_interObsID_astrometry*) for checking inter-ObsID astrometric alignment in projects with multiple overlapping observations and for checking the alignment between the ACIS data and an astrometric standard catalog, such as 2MASS. This tool is intended to be used in the context of our “validation procedure” (§7.1).
- A significant bug in the selection of bin boundaries in AE lightcurves has been identified and fixed.
- Our own recipe for using AE (formerly named *recipe.txt*) has been split into multiple documents (§7.1).
- Numerous small improvements have been made throughout the AE package.

¹¹⁰<http://space.mit.edu/ASC/MARX/>

¹¹¹<http://personal.psu.edu/psb6/TARA/code//>

¹¹²<http://personal.psu.edu/psb6/TARA/code//>

F.8 AE Version 2012apr05

- We have exercised this version of AE under *CIAO* 4.4.
- **This release of AE is NOT yet compatible with MARX 5.0. I will certainly let you know when I have updated AE, but I do not have a good estimate of when that will occur.**
- We recommend the “2012apr05” or later version of the *TARA* package¹¹³.

F.9 AE Version 2012-03-19

- ACIS event lists contain a boolean flag column named STATUS that *CIAO* uses to record various data cleaning criteria that are commonly used to remove events likely to arise from the instrumental background. Historically, two of those cleaning criteria—“afterglow” identification and “5x5 background” identification—produce false positives for sources with high count rates. Thus, low-rate sources are optimally extracted by applying these cleaning criteria and high-rate sources are optimally extracted by ignoring these cleaning criteria (Broos et al., 2010).

AE supports the optimal extraction of both groups of sources by applying a STATUS=0 filter to the source and background extractions of low-rate sources, and ignoring the STATUS column for high-rate sources. To take advantage of this capability, the observer should clean Level 2 event data (§7.3) using only the “safe” STATUS bits—those not associated with the “afterglow” and “5x5 background” criteria.

Event lists supplied to AE must now include the STATUS column, whether you care about AE’s conditional STATUS filtering or not.

- In the existing model-based algorithm for constructing background spectra (§7.6.1) and the existing model-based algorithm for masking point sources (§7.6.3), each point source model now includes the ACIS readout streak when it is bright.
- The set of plots produced by AE has been pruned to eliminate the less useful ones. Virtually all plots are now automatically saved as PostScript files.
- The new tool *ae_interObsID_astrometry* (§7.14.1) helps the observer verify the astrometric alignment of individual ObsIDs after an AE extraction has been performed.
- AE has been added to the Astrophysics Source Code Library¹¹⁴. The citations we request (§4) for work that relies on AE has been updated.
- **This release of AE is NOT yet compatible with MARX 5.0. I will certainly let you know when I have updated AE, but I do not have a good estimate of when that will occur.**
- **This release of AE has NOT been tested under CIAO 4.4. I will certainly let you know when I have exercised AE with this new version of CIAO.**
- AE now requires the “2012mar19” or later version of the *TARA* package¹¹⁵.
- AE now requires the AstroLib version dated 22-Sep-2011 or later.
- AE now requires XSPEC version 12.7.0 or higher, because the AE fitting scripts now expect that the *cplinear* model (§7.11) is built into XSPEC, rather than compiled locally by the observer.
- Our own recipe for using AE¹¹⁶ (§7.1) has been significantly revised.
- Because the scope of AE and its level of automation have evolved over time, the AE manual had become poorly organized—giving too much emphasis to low-level AE components that are no longer commonly called directly by the observer. I have attempted to re-organize the *Using ACIS Extract* section to better describe the interfaces to AE that the modern observer is likely to use. I’m sure the manual could be better still, in countless ways. By the end of the *Chandra* mission I may perhaps have finally explained clearly how to use AE!

¹¹³<http://personal.psu.edu/psb6/TARA/code/>

¹¹⁴<http://ascl.net/1203.001>

¹¹⁵<http://personal.psu.edu/psb6/TARA/code/>

¹¹⁶http://personal.psu.edu/psb6/TARA/ae_users_guide/recipe.txt

- The input parameter `S_AIMPOINT` has been eliminated; AE now infers the aimpoint using the SIM position.

F.10 AE Version 2011-04-06

- AE now requires the “2011apr8” or later version of the *TARA* package¹¹⁷.
- Released a new tool, `ae_make_psf_hook_regions` (§7.14.7), to help observers identify detections that are spatially coincident with the recently discovered hook-shaped feature in the *Chandra* PSF¹¹⁸.

F.11 AE Version 2011-03-15

- Small changes were made to accomodate *CIAO* 4.3.
- Due to a conflict between IDL version 8 and the historical AstroLib, if you are using IDL version 8 you need to update your AstroLib to the Sept. 2010 or later version (<http://idlastro.gsfc.nasa.gov/ftp/>), and update your copy of our TARA package (if you use that) to the 2011mar16 version (<http://personal.psu.edu/psb6/TARA/code//tara2011mar16.tar.gz>).

- Any time you need to run a tool in the files `acis_extract_tools.pro` or `ae_recipe.pro` you should now load *all* the AE code with the IDL command

```
.run ae
```

rather than with any of these now obsolete methods:

```
.run acis_extract_tools
```

```
.run ae_recipe
```

```
.run aet
```

```
.run aer
```

- A discussion of ObsIDs that include both FI and BI detectors has been added to §5.5.
- The process of generating FITS and L^AT_EX tables that collate AE results has been improved. The COLLATE stage of AE still produces a FITS table with vector columns, in order to record source properties over many energy bands. We have introduced the new tool `ae_flatten_collation` (§7.14.14) that transforms the AE collation into a more convenient and scientifically useful FITS table.
 - Results are retained for only the three energy bands we typically care about. Those quantities are stored in independent columns, eliminating all vector columns.
 - The unfriendly column names used by AE are replaced with more meaningful names.
 - The *CIAO* tool `aprates` is used to estimate confidence intervals on `NET_CNTRS`, replacing the less accurate AE photometry confidence intervals.
 - The observed energy flux is very simply estimated by multiplying AE’s photon flux by the median observed event energy.
 - The confidence intervals on spectral model parameters produced by *XSPEC* are formatted into L^AT_EX strings, ready for use in a L^AT_EX table of fitting results (either the ones we provide with AE, or custom tables produced by the observer).
 - *XSPEC* confidence intervals on model normalizations are converted to emission measures and formatted into L^AT_EX strings. *XSPEC* fluxes are converted to luminosities.
- Our own recipe for using AE¹¹⁹ (§7.1) has been revised.
- The XSPEC fitting scripts supplied with AE have been revised.
 - XSPEC version 12.6 or higher is now required.

¹¹⁷<http://personal.psu.edu/psb6/TARA/code//>

¹¹⁸http://cxc.harvard.edu/ciao/caveats/psf_artifact.html

¹¹⁹http://personal.psu.edu/psb6/TARA/ae_users_guide/recipe.txt

- The scripts have always saved information about the fitted model using FITS keywords. The 8-character limit imposed on FITS keywords forces us to use some very unreadable keyword names. Unfortunately, we historically have not chosen these names very carefully. Thus, over time, the names for some keywords have evolved.

For example, keywords holding XSPEC fluxes have to encode the energy range over which the flux was computed. We originally used a naming convention like F0P5_8 to mean flux in the 0.5 to 8 keV range; now we use the letter “H” as an abbreviation for 0.5 and we omit the underscore, resulting in the replacement name FH8.

For another example, we originally represented parameters from the *first* component of thermal models with keywords that omitted the number “1”, e.g. “NH”, “KT”, “NORM”; now we include the “1” in the names.

- The design of the scripts has been modified to reduce the labor required to implement a new spectral model. These new scripts have NOT yet been well tested.
- The tool *ae_radial_profile* (§7.14.6) has been simplified.
- Units are more consistently recorded for source properties and shown on plots. The format follows the standard defined by the document “Specification of Physical Units within OGIP FITS files”.
- The execution speed of image reconstructions (CHECK_POSITIONS stage) has been improved by reducing neighborhood sizes and cropping PSF images.
- The AstroLib routine convolve.pro, used for image reconstruction, has evolved over time. With some versions of the AstroLib, the source position estimates AE derives via image reconstruction may have systematic errors. In this and subsequent versions of AE, the CHECK_POSITIONS stage begins by reconstructing a fake data image with a fake PSF, and then verifying that the astrometry of the result is correct. This defensive step should detect user configurations where AE and the AstroLib are making *different assumptions* regarding the astrometry of PSF images.
- An AE user (G. Sivakoff) has generously contributed IDL code that will query the CXC’s archive to determine the aimpoint of an ObsID; see §E.1.
- An optional parameter WMAP_ENERGY_RANGE has been added to the EXTRACT_SPECTRA Stage (§E.3).

F.12 AE Version 2010-02-26

- Our own recipe for using AE¹²⁰ (§7.1) has been significantly streamlined during the course of training several colleagues to use it.
- The syntax for calling the various /PLOT stages of AE has been revised.
- The DTCOR value in an observation’s FITS header is now applied to lightcurves to account for exposure time lost during CCD frame transfers.
- Fixed a bug (introduced at the last release) that caused failures in the CHECK_POSITIONS stage.
- Changed the type of file supplied as the *obsfile* parameter to *mkarf*. Originally, the CXC recommended supplying the aspect histogram file, but since the release of CIAO 3.4 in 2007 the CXC has recommended supplying an event file as the *obsfile* parameter. I only recently noticed this change, and have revised AE accordingly. I do not expect this change to the *obsfile* parameter to produce any significant change in the resulting ARF.
- The parameters accepted by the utility *ae_make_psf* have been revised.

F.13 AE Version 2009-12-15

- I have realized that distributing compiled versions of our *cplinear* background model is not possible because the resulting library contains an absolute path reference to the directory at Penn State where the model was compiled. The manual has been revised to state that observers must compile *cplinear* locally (§6).

¹²⁰http://personal.psu.edu/psb6/TARA/ae_users_guide/recipe.txt

F.14 AE Version 2009-12-01

- Observers working with multiple overlapping pointings will be pleased to know that AE now has an additional option for discarding unhelpful extractions (§5.13) during the MERGE stage (see MERGE_FOR_PHOTOMETRY in §7.8). This new option seeks to optimize signal-to-noise ratio while controlling photometric bias.
- *CIAO* 4.1 is naughty—it sets the environment variable *HEADAS* and breaks *XSPEC*. This should not affect AE itself, since AE takes great pains to maintain separate environments for spawning *CIAO* and *HEASOFT* commands. However, if you wish to configure both *CIAO* and *HEASOFT* in normal shells then I suggest the following lines in *.cshrc*:

```
if ($?FAST_START == 0) then
    source $HEADAS/headas-init.csh

    set headas=$HEADAS
    source $ciao_dir/bin/ciao.csh -o -q
    setenv HEADAS $headas
endif
```

We configure *CIAO* last, so that 'pget' comes from *CIAO* not *HEASOFT*, and then repair the environment variable *HEADAS*.

- A caveat (§7.13) has been added to warn observers that mixing front-illuminated and back-illuminated CCDs in the same AE run is dangerous.
- The naming convention that AE uses to organize the compiled versions of the *cpllinear* model has been changed in an attempt to be compatible with more computing environments. To avoid confusion, install AE in a **clean directory**, not on top of an existing AE installation. The instructions for compiling *cpllinear* (§7.11) for platforms not already supported have been slightly modified.
- The description of the AE release on 2009-06-22 describe a significant change in strategy for choosing the sizes of background regions. Background scaling is now represented as a (traditional) scalar quantity in the multi-ObsId background spectrum, rather than as an energy-dependent column. A side effect of this change was that AE no longer supports the notion of separate ARF files for background spectra (background.arf)—a construct relevant to one of the possible approaches to dealing with background in diffuse sources (“Subtract Sky Region” in §7.1.2). Thus, the MERGE stage no longer accepts an EMAP_ENERGY option, formerly needed when background.arf files were supplied..

F.15 AE Version 2009-08-09

- Figure 8 and accompanying text have been revised to better describe the standard AE workflow.
- A small section on troubleshooting spectral fitting runs has been added to §7.11.
- The *What’s New* entry for 2009-06-22 (below) has been revised to show the patches one must make to older AE sources in order to execute the new tool *ae_adjust_backscal_range*, which was introduced into the AE workflow at the 2009-06-22 release.
- The /INTERACTIVE option to the FIT_SPECTRA Stage has been improved so that the *XSPEC* prompt presented to the user implements command recall via the arrow keys. To use this feature, you must re-install the revised TCL script *interact.tcl* distributed with AE (§7.11).
- AE’s *XSPEC* fitting scripts now include a placeholder for loading an observer-supplied model for background components that are not represented by the background observation that has been extracted (§7.11).
- For diffuse sources, AE catalogs now consist of two columns (§7.1.2).

- The WMAP used for computing responses for diffuse sources is now built using events in the energy range 500 to 2000 eV.
- AE spectra and responses now use 1024 (0:15 keV) channels by default; if the old channel range (685) is needed for backward compatibility, then specify the option DETCHANS=685.
- The file name AE’s fitting scripts use to store the best-fit *XSPEC* model has been changed from model.xspecsav to model.xcm. The scripts now save a second model—the best fit found prior to the parameter error estimation process—in the file model.before.errors.xcm (§7.11).
- A significant error in the Appendix (preparing the “ACIS stowed data” in CALDB for use in diffuse analysis) of our personal recipe for using AE has been found and fixed.
- The obsolete single-ObsId source property MEDIAN_E (“median energy”, found in obs.stats files) is no longer calculated by AE because observers can better characterize the median event energy for a source using the multi-ObsId source property “ENERG_PCT50_OBSERVED” (“50th percentile energy”, found in source.photometry files). The ENERG_PCT50_OBSERVED statistic (§5.10) has two advantages: (1) it is corrected for background, (2) it is available for several energy bands.
- The example latex table generator supplied with AE is currently not functioning; it will be revised soon.
- A new option, /USE_MKRMF, has been added to the EXTRACT_SPECTRA stage to force AE to use the *mkrmf* tool (e.g. for certain data taken at -110 C).

F.16 AE Version 2009-06-22

- AE now **requires** *CIAO* version 4.1 or later. Due to structural changes in the CALDB, this and subsequent versions of AE are not compatible with earlier versions of *CIAO*. As recommended by the CXC, the observation aspect file is now used for most calls to the tool *dmcoords*, and thus the aspect file is now required by more AE stages/tools than before.
- When fitting with the C-statistic, AE now requires *XSPEC* version 12.5.0 or later.
- AE now requires the “2009jun22” or later version of the *TARA* package¹²¹.
- Theoretical problems regarding AE’s historical approach to merging background spectra extracted from multiple observations have been identified after an AE user noticed some unusual results. In short, we used to naively assume that the sizes of the background regions in each extraction could be chosen independently and then combined in a way that produced an energy-dependent background scaling (AREASCAL column). However, we now believe that this approach causes problems when the background scaling (BACKSCAL value) of the individual extractions are dissimilar. The algorithm for merging background spectra has been revised—see §5.6—and AREASCAL is now a scalar quantity.

Thus, a new constraint has unfortunately been added to the recommended AE workflow—namely that every extraction of a specific source should use a similar scaling (BACKSCAL) for its background. The AE workflow now includes a new tool, *ae_adjust_backscal_range*, that adjusts the allowed BACKSCAL range for each source. See §3.4, 7.1, and 7.7.

In order to run *ae_adjust_backscal_range* on existing AE sources you must add three keywords to all the source.stats files, like this:

```
; Collate the existing extractions to find a typical BACKSCAL for each source.
acis_extract          , 'all.srclist', /SINGLE_ObsId, COLLATED_FILENAME='temp.collated'

bt = mrdfits('temp.collated', 1)

; Use that typical value to create new keywords BKSCL_LO, BKSCL_GL, BKSCL_HI in source.stats.
```

¹²¹<http://personal.psu.edu/psb6/TARA/code/>

```

.run
BKSCL_GL = bt.BACKSCAL
BKSCL_LO = (2./3.) * BKSCL_GL
BKSCL_HI = 2 * BKSCL_LO
ind = where(~finite(BKSCL_GL), count)
if (count GT 0) then begin
    BKSCL_LO[ind] = 20.0
    BKSCL_GL[ind] = 30.0
    BKSCL_HI[ind] = 40.0
endif
end

.run ae
ae_poke_source_property, SRCLIST_FILENAME='all.srclist', KEYWORD='BKSCL_LO', VALUE=BKSCL_LO, $
COMMENT='smallest BACKSCAL allowed'

ae_poke_source_property, SRCLIST_FILENAME='all.srclist', KEYWORD='BKSCL_GL', VALUE=BKSCL_GL, $
COMMENT='target BACKSCAL'

ae_poke_source_property, SRCLIST_FILENAME='all.srclist', KEYWORD='BKSCL_HI', VALUE=BKSCL_HI, $
COMMENT='largest BACKSCAL allowed'

```

- The *cpllinear* background model that we use for CSTAT fitting (§5.12) now uses a variable number of vertices, and the vertices are placed more carefully to better avoid numerical problems during the fitting process.
- In the source.stats file AE’s multi-ObsId variability statistic has been renamed to MERGE_KS; the PROB_KS quantity there now holds the single-ObsId KS variability statistic from the observation exhibiting the strongest variability.
- Our own recipe for using AE¹²² (§7.1) has been significantly revised.
- Our pile-up recipe has been updated to reflect this version of AE.
- The observation “mask file” which is used by *mkarf* and *mkwarf*, is now a standard input required by AE (§3).
- We have tried to fix a bug in the code that builds PSF images for observations that use the HETG or LETG. This case has not been tested however, so please use with care and report any problems you suspect.

F.17 AE Version 2009-01-27

- Our own recipe for using AE¹²³ (§7.1) has been significantly revised. Our recipe for using AE on *XMM* data¹²⁴ has been simplified to refer to the ACIS recipe when possible.
- The *ae_better_backgrounds* (§E.6) tool is generally no longer explicitly called by the observer; instead, the tool *ae_standard_extraction* (§7.6.1) now has an option /BETTER.BACKGROUNDS which directs it to call *ae_better_backgrounds* instead of performing the masked background computation.
- AE now supports *MARX* version 4.3.0. We recommend that you keep your *MARX* installation up to date in order to have the best possible PSF calibration. Note that the latest *MARX* has tools in two different directories ({marx_installation_directory}/bin/ and {marx_installation_directory}/lib/marx/), both of which must be in your Unix path.
- AE now **requires** *CIAO* version 4.0 or later. AE does **not** yet work under *CIAO* version 4.1 (due at least to changes in the structure of CALDB).

¹²²http://personal.psu.edu/psb6/TARA/ae-users_guide/recipe.txt

¹²³http://personal.psu.edu/psb6/TARA/ae-users_guide/recipe.txt

¹²⁴<http://personal.psu.edu/psb6/TARA/procedures/xmm-recipe.txt>

- For projects involving multiple observations, we have added several options to the MERGE stage (§7.8) to address the difficult issue of deciding, for each source, when **discarding** an observation is expected to improve the quality of the merged extraction (§5.13).
 - A limit on the acceptable level of overlap between neighboring extraction regions (§7.5) can now be specified via the OVERLAP_LIMIT option .
 - The MERGE_FOR_PB option can optimize the merge for source validity.
 - The MERGE_FOR_POSITION option can optimize the merge for source position.
- AE's grouping algorithm now includes a stronger mechanism to combat the bias that can arise from the asymmetry inherent in the standard grouping algorithm (see 3rd bullet of §5.7).
- The algorithm used by the tool *ae_make_catalog* to choose PSF fractions that prevent overlapping extraction regions has been significantly improved. *CIAO*'s data model is now used to measure the amount of overlap between neighboring regions; previously we used circular parameterizations of the regions to assess overlap.
- The source position that AE estimates from an image reconstruction of the source neighborhood (RA_ML,DEC_ML) is now a centroid of a 3x3 pixel region in the reconstructed image, rather than the center of the brightest pixel (§7.9).
- When AE calls *dmcoords* to convert a position to CHIP coordinates, the aspect file is now supplied to *dmcoords*.
- The EXTRACT_SPECTRA Stage (§E.3) now accepts an observation mask file¹²⁵ (via the parameter MSKFILE) for use by *mkarf* and *mkwarf*.
- For diffuse sources (§7.1.2), the EXTRACT_SPECTRA Stage (§E.3) now accepts an aspect file (via the parameter ASPECT_FN) for use by *mkacisrmf*.
- The SHOW_REGIONS stage (§7.10) now requires a FITS collated table rather than a source list.
- The *ae.better.backgrounds* (§E.6) and *ae.better.masking* (§7.6.3) tools can now be run on multiple ObsIds at the same time.
- §6.2 describes a number of undocumented AE capabilities that speed up processing for atypical projects.
- The installation section (§6.1) now states a requirement that your shell must not produce any output to stdout when it starts up.
- PSF construction via *mkpsf* is no longer supported.
- The *ae_make_catalog* and *ae_standard_extraction* tools now require the EVTFILE_BASENAME parameter (e.g. EVTFILE_BASENAME='spectral.evt' as shown in §3).

F.18 AE Version 2008-06-09

- After discovering that the standard procedures for calling *MARX* (e.g. those listed on the *ChaRT* web page in May 2008) do not account for off-nominal SIM positions which occur in some observations, we have attempted to improve the algorithm AE uses to call *MARX*. We have also attempted to implement the published *MARX* workaround for simulating S-array sources with an I-array aim point, and vice versa.

These changes to the *MARX* calling procedure were deduced by a couple of AE users, with very limited assistance from the CXC and *MARX* teams. I am **not** confident that the method accounts for evolution of the observatory geometry and default operating parameters. As has been the case throughout the mission, I encourage you to always suspect the accuracy (size, position, and normalization) of the PSFs that AE generates using the mission tools (originally *mkpsf*, now *MARX*). If you find flaws in the PSFs that concern you, I encourage you to reproduce the problems outside the context of AE and to report your concerns to the CXC.

¹²⁵<http://asc.harvard.edu/ciao/dictionary/maskfile.html>

- **Note that application of the CXC thread “Improving the Astrometry of your Data: Correct for a Known Processing Offset” will induce an offset between your data and the *MARX* PSF. We do not understand the header magic involved here, but fortunately this thread is not applicable to data re-processed after 2004.**
 - As of June 2008 the CXC’s afterglow cleaning tools have significant problems. See the *CIAO* web page on afterglows¹²⁶ for a discussion of the afterglow issue. The first afterglow tool, *acis_detect_afterglow*, finds many afterglows, but eats out the cores of bright sources. The second tool, *acis_run_hotpix*, is gentle on sources, but misses many afterglows, leading to lots of spurious source detections for many projects. We recommend an analysis work flow which is bifurcated—a source detection branch uses both CXC afterglow tools for aggressive afterglow cleaning, and a source extraction branch omits the *acis_detect_afterglow* in order to preserve the cores of bright sources.
- For projects in which you have already extracted a catalog that may be contaminated with lots of spurious “afterglow sources”, we have written a simple tool, *ae_afterglow_report*, that looks for suspected afterglow events in the extracted data for each AE source. Calls to this tool are shown in our personal recipe (§7.1), but it is not documented further in this manual. Obviously, this tool is a desperate workaround to the unpleasant situation where you have extracted data that are contaminated with afterglows; no afterglow algorithm run this late in the analysis can be as effective as an algorithm run on the L1 data.
- The AE package now includes a convenient tool for constructing exposure maps (§7.14).

F.19 AE Version 2008-04-29

- We have completed initial development of a work flow for analyzing point sources observed with the EPIC instrument on *XMM* (§7.1.3). Scientific validation of this work flow has been very limited; observers are encouraged to use this work flow but to closely and skeptically examine the results, particularly background spectra and background scaling. Note that I have almost zero experience with *XMM* data.
- Our theoretical understanding of how multiple observations of diffuse emission should be combined has improved (§5.14); AE has been revised accordingly.
- Our theoretical understanding of how surface brightness quantities should be calculated from a diffuse extraction has improved (§5.14) and has been implemented in AE.
- Our theoretical understanding of the various ways that background can be handled in diffuse analysis has improved (§7.1.2).
- Our personal recipe for diffuse analysis using an off-target “sky” region (§7.1.2) now uses the “stowed event data” provided in CALDB to subtract instrumental background, and then simultaneously fits the diffuse and sky net spectra using a shared model for the X-ray background in both regions plus a model for the diffuse emission in one region.
- The AE fitting scripts based on the *wabs* absorption model have been retired, replaced by scripts based on the *tbabs* model; the *vapec* model has replaced the *apec* model (§5.12).
- Simultaneous instances of the AE FIT_SPECTRA stage can now be run on the same catalog, saving time for observers who wish to fit multiple models and have multiple processors available.

F.20 AE Version 2008-03-03

- **Removed** requirement that the *ciao* and *heasoft* aliases must not print anything (formerly in §6.1).
- Added requirement that \$TMPDIR must be consistent across shells (§6.1).
- Added workaround for some *CIAO* 4 problems.

¹²⁶<http://cxc.harvard.edu/ciao/why/afterglow.html>

F.21 February, 2008 (AE Version 3.175)

- Starting in version 3.172 AE will re-use any source RMFs existing from an earlier extraction in order to avoid long-running calls to the tool *mkacisrmf*. This release fixes a bug which caused that re-use mechanism to generate an error.

F.22 January, 2008 (AE Version 3.173b)

- For a long time now, a bug in *CIAO*¹²⁷ has complicated the construction of masked exposure maps for use in the EXTRACT_BACKGROUND stage, leading to cropping of the background data. Brief discussion of this bug was added to this manual in several places in October, 2006. However, readers may have easily missed those references to the problem since some were in footnotes. Since the bug remains in the recently-released *CIAO* 4.0 and will probably be with us for a while longer, I’ve modified the *ae_standard_extraction* tool work around this bug (by adding small mask regions at the corners of the exposure map). If you choose to prepare masked exposure maps manually, §7.6.2 now describes in more detail the options you have for dealing with this bug.

F.23 January, 2008 (AE Version 3.172)

- By default *MARX*, rather than the *mkpsf* tool, is now used to model the HRMA PSF. You can specify that AE revert to using *mkpsf* via the new option (/PSF_FROM_LIBRARY, §E.1).

Note that there appears to be a systematic astrometry difference between *MARX* and *mkpsf*—the centroid of the two PSFs (before blurring) are generally offset from each other. I find this offset is near zero on-axis and increases linearly with off-axis angle at a rate of ~ 0.12 sky pixels per arc minute. I do not, of course, know which PSF has the more correct astrometry. One implication of this offset is that one should avoid the situation where a source’s position was determined using a PSF from one system (e.g. via AE’s “correlation position” estimate or via image reconstruction), and the source’s extraction region was constructed using a PSF from the other system. In a multi-observation project, one should not extract some observations with one flavor of PSF and some with the other unless you have considered very carefully how that would affect your analysis.

- Post-HRMA blurring effects are now modeled individually rather than via a single Gaussian smoothing as was done in previous versions (§5.1.2). I made this change because I have two concerns about the Gaussian method. First, I have theoretical concerns about the use of Gaussian models for two of the blur components. Second, even if the Gaussian model is adequate, as of January 2008 the recommended parameter value to use in that model (e.g. the *DitherBlur* parameter in *MARX*) has no solid calibration provenance.
- Spectral fitting with the C-statistic now employs a completely different model family (*cpllinear*) for the background, significantly improving the fit for weak sources (§5.12). To support this innovation, AE multi-ObsId spectra now represent background scaling using the *AREASCAL* FITS keyword/column rather than the more common *BACKSCAL* keyword/column (§5.6). Thus, if you wish this AE to fit sources extracted by an earlier version of AE, then you must first re-run the MERGE stage to rebuild the spectra.
- The tool *ae_better_backgrounds* (§E.6) for constructing background spectra in crowded fields has been improved and further tested. I use it routinely myself, and now recommend it whenever a point source’s extraction region may contain significant contamination from another point source.
- Our own recipe for using AE¹²⁸ (§7.1) has been significantly revised.
- A tool that assists with time-resolved spectroscopy (*ae_timerange_extract*) is now available (§7.14).
- A tool to manually group spectra using the AE grouping algorithm (*ae_group_spectrum*) is now available (§7.14).
- A tool to plot radial profiles of a source’s data and PSF (*ae_radial_profile*) is now available (§7.14).
- A tool that partially automates the complex process of building AE-compatible PSFs from *ChaRT* (*ae_chart_interface*) is now available (§7.14).

¹²⁷<http://asc.harvard.edu/ciao/bugs/dm.html>

¹²⁸<http://personal.psu.edu/psb6/TARA/procedures/recipe.txt>

- Fitting scripts can now pause after the source is fit, but before fluxes and output files are produced, so the observer can interact with *XSPEC* to refine the fit (see the `/INTERACTIVE` option in §7.11).
- Please note that fitting scripts and data products are now stored in a sub-directory named *spectral_models* (§7.11).
- Please note that the parameters expected by the tool *ae_make_catalog* (§3, §7.5) have changed to be more consistent with the tools *ae_source_manager* and *ae_standard_extraction*—the source list file name parameter is now optional, defaulting to “all.srclist”.
- Please note that the use of *MARX* to generate PSFs requires that an additional *CIAO* data product, the aspect file, be provided to AE (§3 and §E.1, keyword `ASPECT.FN`).
- Please note that for some of the multi-ObsId source properties stored in the *obs.stats* file (§7.8) the method for combining the corresponding single-ObsId properties has changed. For example, `PSF_FRAC` was defined as the smallest value found among the observations; now it is a weighted average.
- The `CHECK_POSITIONS` stage (§7.9) has a new option, `THETA_RANGE`, that can be used to limit which sources are processed.
- The `SHOW_REGIONS` stage (§7.10) has a new *panning mode*.
- This manual now includes a suggestion for how users of the *bash* shell can run AE (§6.1).
- I believe the earliest version of *IDL* that will run AE is 6.1; I am currently using 6.4.1.
- The *XSPEC* scripts distributed with AE now require *XSPEC* version 12.3.1 or higher.
- The *MARX* simulator is now listed as one of the packages AE requires (§6).
- AE has **not** yet had much run-time under *CIAO* 4.0; please report any problems you encounter.
- Although I try to avoid them, there may very well be backward compatibility problems, beyond those mentioned above, between this version of AE and data products produced by earlier versions. Please contact me if you have specific concerns.

F.24 April, 2007 (AE Version 3.135)

- Added a third stopping criterion for defining the background region in the `EXTRACT_BACKGROUND`s stage (§E.5).
- To support analysis of point sources in zeroth-order HETG grating data, AE now uses the `GRATING` keyword in the event file to properly set the *grating* parameter for *mkarf*. I have not worked with grating data, and I am not certain that this is the only accommodation required in AE to support such analysis.

F.25 March, 2007 (AE Version 3.128)

- A new experimental tool for constructing background spectra for crowded sources has been briefly described in §E.6. **As of April 2007 development of this tool is on-going. Revisions are not announced or routinely released.**
- Additional discussion of AE’s source significance statistics has been added to §5.10.3; some misinformation previously stated there has been removed. The calculation of the `PROB_NO_SOURCE` statistic has been revised to follow Weisskopf et al. (2007, § A2).
- AE now requires the AstroLib version dated 12 February, 2007 or later in order to get the correct version of *forprint.pro*. You should remove the *forprint.pro* file previously distributed with AE.

F.26 February, 2007 (AE Version 3.125)

- A description of how anti-virus applications can slow down AE extractions has been added to this manual (§6.2).
- Time variability is now quantified in two ways: within each observation in the TIMING stage (§E.4), and across multiple observations in the MERGE_OBSERVATIONS stage (§7.8). Two plots are now produced showing light curves from multiple observations—one uses a separate set of axes for each observation (Figure 9), the other shows all observations on a single “broken” time axis (Figure 10).
- The TIMING stage (§E.4) now produces a “grouped” light curve (i.e. a light curve with unequal bin sizes) and computes the median energy of events in each time bin. Errors on those median energy statistics are now computed and plotted.
- This version of AE seems to be compatible with CIAO 3.4.
- A new optional input parameter, PBKFILE, has been added to support the “Dead Area calibration”¹²⁹ capability recently added to CIAO (§E.3).
- TARA version 2007jan18 or later is now required!
- The AE distribution includes a version of an AstroLib routine (forprint.pro) which has a bug fix that has not yet been released in the AstroLib. Your IDL path should be set up so that the AE directory appears ahead of the AstroLib directory.
- The FINAL_PRODUCTS stage has been renamed to MERGE_OBSERVATIONS to better reflect its purpose.
- The MERGE_OBSERVATIONS stage must now be run prior to the CHECK_POSITIONS stage (Figure 8).
- The MERGE_OBSERVATIONS stage now will accept an optional observation name in situations where you wish to compute photometry, responses, etc. on a single observation but you have extracted multiple observations.
- We now recommend that exposure maps should be constructed with a pixel size of 1 sky pixel or smaller if the tools *ae_better_masking* or *ae_better_backgrounds* are to be used (§7.6.3, E.6).
- We’ve added to this manual example code for performing image reconstruction outside of AE (§7.9.5).

F.27 October, 2006 (AE Version 3.107)

In the course of extracting a new ACIS observation recently I became annoyed with the complexity of executing my personal AE recipe (*recipe.txt*, distributed in §7.1). By far the hardest part of the extraction work I’m involved with is fiddling with the catalog prior to the final pass through AE. This is the period where we’re noticing missed sources that must be added, identifying invalid sources that we want to remove, and re-positioning sources. Two aspects of this work were particularly awkward: managing the source list (adding, remove, re-positioning sources) and source directories, and generating the various *DS9* region files needed at various points in the battle. In this release I’ve made modest attempts at easing these burdens.

Firstly, I’ve written a tool to manage the source list and source directories: *ae_source_manager* (§7.2). Under this paradigm the observer no longer manages lists of source coordinates, manages the AE catalog, or worries about discarding source directories which have become obsolete. Sources are added, removed, and re-positioned via *ae_source_manager*. The repository of all source information is the AE extraction directories themselves. Observers can attach a *PROVENANCE* string to each source to help keep track of the method used to detect each source.

Secondly, I’ve tried to bring some sanity to the generation of *DS9* region files that depict the whole catalog by adding a region file generation option to the COLLATE stage (§7.12). Celestial coordinates are used now rather than the silly sky coordinates used previously. Sources now have a *LABEL* property (managed by *ae_source_manager*) that is used for labeling *DS9* regions. This allows consistent labeling of sources even when region files are generated for sub-sets of the catalog. By default these labels are the sequence number of the source in the catalog, but the observer can supply different labels if desired.

Other changes found in this release are listed below:

¹²⁹<http://cxc.harvard.edu/ciao3.4/why/acisdeadarea.html>

- AE now requires *TARA* version 2006sept4 or later.
- The algorithm used by the tool *ae-make-catalog* to choose PSF fractions that prevent overlapping extraction regions has been significantly improved. When two sources are in conflict their PSF fractions are adjusted to simultaneously prevent overlap and to exclude approximately the same number of counts from each source. The net effect is that bright sources get larger PSF fractions than weak sources (when necessary). Also, as a side effect of introducing *ae-source-manager*, the parameters to *ae-make-catalog* have changed (§7.5).
- The tool *ae-better-masking* (formerly *ae-optimal-masking*) has been briefly described in §7.6.3.
- These manual sections have been significantly revised: §3, 7.1, 7.2, 7.12.
- My personal AE usage recipe (*recipe.txt*, distributed in §7.1) has been significantly revised.
- A list of default parameter values that observers should review has been added to the manual (§7.4).
- The /UPDATE_POSITIONS_DATA and /UPDATE_POSITIONS_CORR stages have been removed from AE.
- The OBS_REGION_FILENAME option has been removed from the EXTRACT_SPECTRA stage.

Note that there is a known bug in *CIAO* 3.3¹³⁰ that can affect the source masking used to obtain a background event list. Until *CIAO* is revised you'll probably find that your background data set is cropped somewhat. If this is a concern you can remove the “update=no” option to *dmcoppy* in whatever code you're using to mask the sources, e.g. the tools *ae-standard-extraction* or *ae-better-masking*, or a manual *dmcoppy* command (§E.5).

F.28 September, 2006 (AE Version 3.101)

- The Getting Started section of the manual, formerly very lame, has been improved. (Many thanks to Ann Hornschemeier for suggestions.) In that process the syntax for calling the scripts *ae-make-catalog* and *ae-standard-extraction* in *ae-recipe.pro* have been changed slightly.
- Version 12.2.1ap (at least) of *XSPEC* is known to have a bug that corrupts the PostScript plots produced by the AE fitting scripts; version 12.3.0 resolves that bug.

F.29 August, 2006 (AE Version 3.100)

- Facilities have been developed to help the observer conduct a source-by-source review of multiple spectral models produced by multiple sessions of the FIT_SPECTRA stage. The observer is shown the fits results (parameter values, luminosities, and spectral plots) for all the models available for a given source, and the observer is able to designate the preferred model. See §7.14 and §7.12.
- The *TARA* package now includes a tool (*match_xy.pro*) to match catalogs using positional uncertainties for individual sources and a match significance threshold, rather than using a fixed matching radius as is commonly employed. This tool can be used to match catalogs from different observatories, or can be used to find the intersection of *Chandra* source lists obtained from multiple source detection runs. (See §10 of the *TARA* User's Guide¹³¹)
- Our two-temperature thermal plasma fitting script, *wabs_2apec.xcm*, has been updated to work with the current version of AE.
- AE's behavior when the MODEL_CHANGES_FILENAME parameter is provided to the FIT_SPECTRA stage (§7.11) has changed slightly.
- Keyword OBS_REGION_FILENAME added to EXTRACT_BACKGROUND stage (§E.5).

¹³⁰<http://asc.harvard.edu/ciao/bugs/dm.html>

¹³¹<http://personal.psu.edu/psb6/TARA/>

F.30 January, 2006 (AE Version 3.96)

- The *XSPEC* scripts distributed with AE now require *XSPEC* version 12.2.1e or higher.
- AE versions 3.86 through 3.94 contained a bug which caused the *IDL* command *retall* to be executed at the end of AE rather than the command *return*. This was irrelevant for AE runs executed from the command lines, but this would break attempts to call AE from other *IDL* programs.

F.31 December, 2005 (AE Version 3.94)

- AE now supports and requires *CIAO* version 3.3 or higher.

F.32 November, 2005 (AE Version 3.91)

- To more conveniently handle infinite loops during spectral fits, the *XSPEC* process is now killed if it consumes 600 seconds of CPU time.

F.33 October, 2005 (AE Version 3.89)

- AE's spectral grouping algorithm now defines groups based on SNR in the background-subtracted (net) spectrum. Traditional grouping algorithms work with the source spectrum alone, sometimes leading to significant under-grouping where the background is high. New AE keyword parameters control the algorithm—see §5.7 and §7.11.
- The *XSPEC* scripts officially supplied by AE are now distributed in the AE package itself rather than via a separate web page. See §5.12.
 - The scripts/models currently available are *wabs_apec.xcm* and *wabs_pow.xcm*. Each supports both the χ^2 and C statistics.
 - **These scripts require *XSPEC* version 12.2.0bz or higher.** Note that *XSPEC* 12 currently has some crashing problems under the OS-X Tiger operating system, as described at <http://heasarc.gsfc.nasa.gov/docs/software/lhe>.
 - **The default confidence intervals for model parameters and fluxes are now 90% rather than 1- σ .**
 - Two plots showing the data and model are now produced: the traditional grouped spectrum plot (*ldata.ps*) and an ungrouped cumulative spectrum plot (*icounts.ps*). The latter is perhaps useful for visualizing fits to very weak sources. Note however that when the C-statistic (recommended for weak sources) is used one often sees what appear to be sub-optimal model normalizations in these cumulative plots, even though the minimum in the C-statistic seems to have been found.
 - A Tcl variable *skip_errors* can now be used to force the scripts to skip over the computation of model parameter errors when that computation causes problems, such as hanging *XSPEC*.
- For better file organization, the files produced by each fitting session are now stored in a sub-directory of the source directory and some file names have been changed (§7.11).
- The algorithm for estimating position errors has been improved (see footnote in §E.2).
- AE's timing analysis (e.g. light curves, variability) has been moved from the EXTRACT_SPECTRA stage to a new TIMING stage (§E.4).

F.34 July, 2005 (AE Version 3.79)

- Added support and instructions for “stacking” sources (§7.1.1).
- Added support for new *CIAO* tool *mkacisrmf* (§E.3). *CIAO* version 3.2.2 and the corresponding CALDB update (June 2005) are required if you want to use *mkacisrmf*. Note that the energy range of the RMF is chosen to be 0.3 to 9.886 to deal with a quirk in the *mkwarf* program (CXC Helpdesk Ticket #7870).
- Added explanation of how ARFs are combined in MERGE.OBSERVATIONS stage (§5.9).
- Changed convention for contents of optional input INDEX_FILE in /SHOW_REGIONS from zero-based indexing to one-based indexing to be more compatible with line numbers when viewing a catalog in a text editor.

F.35 January, 2005 (AE Version 3.65)

- AE now requires *CIAO* 3.2. The format of *ardlib.par* has changed in *CIAO* 3.2.
- AE now requires the “05jan25” or later version of the *TARA* package.
- AE now requires a version of the *IDL* Astronomy User’s Library¹³² that includes the change made to *create_struct.pro* on 08-Sep-2004¹³³.
- AE now creates a scratch parameter file directory for *CIAO* and *HEASOFT* tools, and forces the observer to explicitly specify the *ardlib.par* file for the observation (§E.3). This helps remind the observer that *ardlib* must be properly configured, and simplifies running multiple instances of AE.
- Added new source validity statistic PROB.NO.SOURCE to source.photometry table (§5.10).
- Added the AE “driver” programs *ae_make_catalog* and *ae_standard_extraction* to simplify the AE recipe. Revised the AE usage discussion in §3 and §7.1.
- Significantly revised the example recipe¹³⁴.
- Added the AE “driver” program *ae_optimal_masking* to improve source masking (§E.5).
- Added option MODEL.CHANGES.FILENAME to FIT_SPECTRA stage (§7.11) to allow fitting scripts to be customized on a source by source basis (e.g. to freeze a fit parameter to a different value for each source).
- Added option /SKIP_CORRELATION to CHECK_POSITIONS stage (§7.9).
- Optional keyword OBS_REGION_FILENAME moved from EXTRACT_EVENTS/EXTRACT_SPECTRA stages to CONSTRUCT_REGIONS Stage (§E.1).

F.36 October, 2004 (AE Version 3.56)

- Added work around for a **bug in the *CIAO* tool *dmimgpick*** which caused **inaccuracies in background scaling**. Helpdesk ticket #7381 describes an apparent bug in *dmimgpick*. The effect of the bug is that in the /EXTRACT_BACKGROUNDS stage events falling near the edges of the background.emap image are excluded from the background spectrum, even if they fall on non-zero pixels in background.emap. In other words in AE versions prior to 3.56 the background.emap and background.evt files may not be consistent with each other, leading to an error in background scaling and subtraction.

The bug is that *dmimgpick* returns NaN values for positions on the image supplied but near the edge. The work around in AE is to use [opt full] when creating background.emap to keep the non-zero pixels far from the edge of the image.

- Added more source property keywords to source.stats file (see §7.8 and §7.9).

F.37 September, 2004 (AE Version 3.51)

- Changed default energy bands in photometry table (EBAND_LO and EBAND_HI keywords in §7.8).
- The COLLATED.FILENAME option to /FIT_SPECTRA has been generalized to its own stage (§7.12) which collates all sources parameters into a large FITS table.
- Added computation of upper and lower errors on NET.CNTS (§5.10).
- Added computation of radial error on the (RA_DATA,DEC_DATA) source position, saved as FITS keyword ERR_DATA during /CHECK_POSITIONS (§7.9).

¹³²<http://idlastro.gsfc.nasa.gov/>

¹³³<http://idlastro.gsfc.nasa.gov/news.html>

¹³⁴http://personal.psu.edu/psb6/TARA/ae_users_guide/recipe.txt

- Now using NaN instead of -1 as the value put in FITS keyword PROB_KS when KS test cannot be run on lightcurve.
- Renamed keyword in source.stats holding the KS significance between the source and background spectra from PROB_KS to KS_SPECT to avoid confusion with the PROB_KS keyword used for light curves.

F.38 August, 2004 (AE Version 3.42)

- Added support for diffuse sources based on AE branch created by Mike Muno. See §7.1.2.
- Replaced RMF_DIR parameter in AE with parameter AXAF_RMF_FILE in *ardlib.par*. **If you use the Townsley et al. CTI correction you MUST now set the AXAF_RMF_FILE parameter in *ardlib.par* appropriately—see §E.3.**
- Added new keyword MAXLIKELIHOOD_ITERATIONS to /CHECK_POSITIONS (§7.9).
- Added note about relationship between source significance and extraction region size (§5.10).

F.39 June, 2004 (AE Version 3.34)

- **AE now requires the “24 May, 2004” or later version of the *IDL* Astronomy User’s Library (<http://idlastro.gsfc.nasa.gov/>).**
- Fixed a bug that led to systematic overestimation of flux for some sources in multi-observation extractions. Prior to this release if a source’s extraction region contained no in-band counts (not even background counts) then no extraction was performed for that observation. Since the ignored observation’s exposure time was not included in the flux calculations (either in AE or in *XSPEC*) the source’s flux was overestimated.

If you have an existing AE extraction you can identify sources suffering from this problem in several ways:

- The observation directories suffering this problem will contain files named *neighborhood.evt* and *source.evt* but will not contain *source.pi*.
- If a log of the messages AE produced when /EXTRACT_SPECTRA was run is available this message indicates that an extraction was aborted: “EXTRACTION SKIPPED: no in-band data found in source region”.
- The number of observations AE extracted for each source is plotted in the /MERGE_OBSERVATIONS, /PLOT stage. *If* you have some way of knowing how many observations there *should* be for each source then you can determine which sources suffer from aborted extractions.
- Fixed a bug that sometimes caused an incorrect PSF to be used to build reconstructed images in the CHECK_POSITIONS Stage.

If you have an existing AE extraction you can identify sources suffering from this problem in two ways:

- If a log of the messages AE version 3.19 produced when /CHECK_POSITIONS was run is available this message will appear for sources suffering from this problem: “Rebinned multi-ObsId data image to ...”
- If you rerun /CHECK_POSITIONS using AE version 3.34 then any source producing the message “Rebinning multi-ObsId data and multi-ObsId PSF images by 2 ...” probably suffered from this problem in the prior run.
- Released draft recipe for extracting piled up sources—see §7.
- Keyword REGION_FILE added to SHOW_REGIONS stage and controls implemented to navigate through catalog (§7.10).
- Keyword OBS_REGION_FILENAME added to EXTRACT_EVENTS/EXTRACT_SPECTRA stages (§E.2).
- A richer set of photometry statistics has been implemented (§5.10).
- Fixed mistake in computation of MEAN_ARF keyword in obs.stats. (The MEAN_ARF column in source.photometry is not affected.)
- Improved support for C-statistic in *XSPEC*. See new keyword /CSTAT in §7.11.

F.40 February, 2004 (AE Version 3.19)

- AE now requires the “February, 2004” or later version of the *IDL Astronomy User’s Library* (<http://idlastro.gsfc.nasa.gov/>).
- PSF images (used to construct extraction polygons, compute PSF fractions, refine source positions, and reconstruct images) have been significantly improved by using smaller pixel sizes on axis, and by using the most appropriate CALDB PSF library (f1, f2, or f3) for each source.
- Significant improvements in the accuracy of the PSF fractions (which affect the shape of the ARFs) have been accomplished by better estimating the total power in each PSF image (the integral of the PSF out to infinity)¹³⁵.
 - AE is now careful to request from mkpsf an image size large enough to avoid cropping the library images.
 - AE now estimates the fraction of the power that is missing from PSF images derived from the f1 and f3 libraries (which have small fields of view) by using the f2 library (which has a larger field of view).

The figure below shows the difference in PSF fractions computed by AE versions 3.4 and 3.19 for 1600 I-array sources.

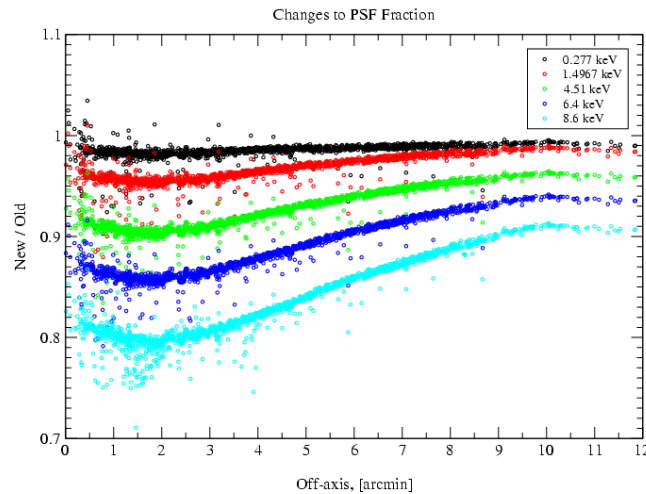


Figure 19: Ratios of PSF fractions computed by AE versions 3.4 and 3.19 for 1600 I-array sources.

- AE will now print a warning when the source extraction region falls outside the field of view of the PSF images so the observer will know that the PSF fraction is underestimated.
- Allowed values for keywords EXTRACTION_NAME (§7.3) and HDUNAME (§7.11) have been expanded.
- New keywords have been added: /REGION_ONLY (§E.1), /PIPELINE_RANDOMIZATION (§E.1), INDEX_FILE (§7.10), /SKIP_RECONSTRUCTION (§7.9).
- Revised blurring level applied to PSF images for the case of no pipeline randomization from 0.21 to 0.27 arcsec (based on CXC Helpdesk ticket #6819)—see §5.2.

¹³⁵The mkpsf tool is currently unable to provide normalization information for the PSF images it generates

F.41 November, 2003 (AE Version 3.5)

- AE now requires the “7nov2003” or later version of the *TARA* package.
- A mistake¹³⁶ in the computation of the KS Probability for light curves has been corrected. As shown in the left figure below the effect of the mistake is to make sources look less variable than they should, i.e. the KS probabilities previously calculated are too large. As shown in the right figure below the error is largest for sources with a small number of counts.

A “patch program” that recalculates only the KS probabilities is available if you have an existing AE extraction on which you do not wish to re-run the full EXTRACT_SPECTRA stage.

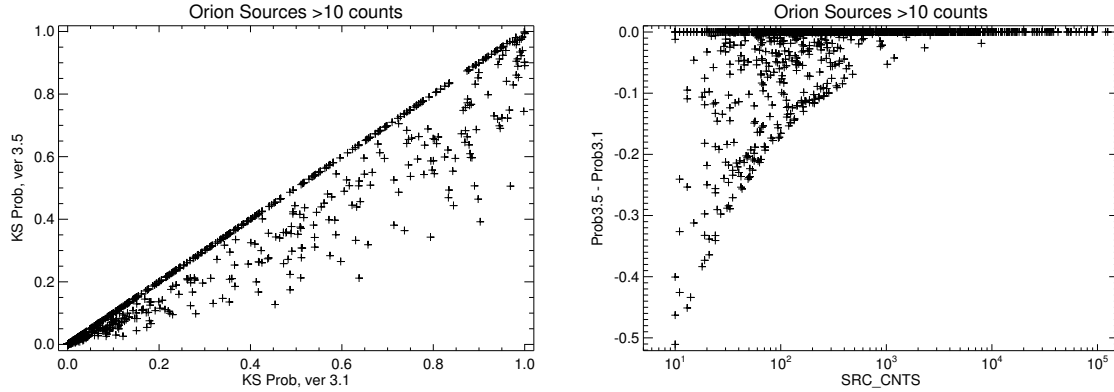


Figure 20: Error in KS Probability for Orion light curves

- The EXTRACT_SPECTRA TIMING stage now produces a PostScript plot of event arrival times and energies—see §E.4.
- The way background regions are used to select the events that form the background spectrum has been revised to improve the accuracy of the background scaling, particularly for regions that are small compared to the pixel size of the exposure map. See note in §E.5.
- The keyword MIN_EXPOSURE_RATIO has been added to the EXTRACT_BACKGROUNDS stage to improve observer’s control over the size of background regions.

¹³⁶The mistake was that, prior to version 3.2, AE was computing distances between the stair-step cumulative distribution of the data and the uniform distribution only at ONE end of each stair-step. The KS routine in the *IDL* Astro Lib, derived from Numerical Recipes, indicates that BOTH ends of each stair-step must be compared to the uniform distribution.

F.42 September, 2003 (AE Version 3.0)

- AE revised to run under *CIAO* 3.0. Earlier versions of *CIAO* are NOT SUPPORTED.
- AE now requires the “30sep2003” or later version of the *TARA* package.
- Format of “catalog” files revised—see §E.0.1. Construction of source names moved outside of CONSTRUCT_REGIONS stage to decouple source name from current estimate of source position.
- Revised PSF construction
 - Revising binning and extent of PSF images (generated in CONSTRUCT_REGIONS stage) to better match the PSF sizes.
 - Revised blurring of PSF images from 0.35 to 0.21 arcsec to better match performance of *Chandra* aspect system—see §5.2.
 - PSF images from a previous AE run are now re-used when possible, greatly speeding up CONSTRUCT_REGIONS stage when only PSF fraction is being changed.
- Revised CHECK_POSITIONS stage—see §7.9:
 - Added Maximum Likelihood reconstruction of each source neighborhood. Peak of reconstruction image saved for possible use as source position estimate.
 - Source neighborhood image is now energy filtered to reduce background.
 - Source neighborhood image filename changed from *source.img* to *neighborhood.img*.
- Revised EXTRACT_EVENTS/EXTRACT_SPECTRA stages:
 - Added computation and plotting of adaptively smoothed light curve and median energy time series—see §E.3.
 - Added note about using observer-supplied extraction regions, e.g. to extract sources suffering from **pileup**—see §E.2.
 - Added keywords EMAP_NUM, EMAP_AVG, EMAP_MED, EMAP_MIN, EMAP_MAX to characterize the exposure map values found inside the source extraction region—see §E.3.
 - Added optional input WARNING_REGION_FILENAME—see §E.2.
 - Added statistical errors on position (X_DATA,Y_DATA) estimated from data—see §E.2.
 - Fixed potential inaccuracy in EXPOSURE keyword assigned by dmextract—see §5.4.
- Improved method for applying mask regions to create “Swiss-cheese” background event list—see §E.5.
- Added optional automatic selection of minimum spectral group size—see §7.11.
- Released example recipe for a complex, multi-observation AE reduction—see §7.
- Added note about discarding unused event list columns to reduce execution time—see §7.3.

Some of these changes are not backwards compatible with extraction file trees produced by earlier AE versions. If you are working with a dataset previously extracted by AE keep in mind:

- If EXTRACT_SPECTRA is re-run you must also re-run the EXTRACT_BACKGROUNDS stage, and vice versa.
- Before re-running CHECK_POSITIONS you must re-run the EXTRACT_EVENTS stage.
- The following AE output file names are obsolete: *source.img*, *source.arf*.

F.43 May, 2003 (AE version 2.22)

- Added support for optional observer-supplied background regions—see §E.5.
- Added optional keyword EXTRACTION_NAME—see §7.3.
- Revised to use Gehrels’ approximation to the $1\text{-}\sigma$ confidence intervals on SRC_CNTS and BKG_CNTS when computing SRC_SIGNIF—see §5.10.

F.44 10 February, 2003

- Added /EXTRACT_EVENTS, /ONLY_EDITED stage.
- Revised §7 and §7.10.

F.45 9 January, 2003

- Moved spectral grouping from the MERGE_OBSERVATIONS stage to the FIT_SPECTRA stage, and revised the manual to match. This simplifies the instructions for calling AE, and makes it more convenient for observers to fit the spectra multiple times using different grouping levels.
- Revised acis_fef_lookup call to work with *CIAO* 2.3.
- Added code to detect common environment setup mistakes.

F.46 Initial Release, October 2002

References

- Anders, E., & Grevesse, N. 1989, *Geochim. Cosmochim. Acta*, 53, 197
- Arnaud, K. A. 1996, *Astronomical Data Analysis Software and Systems V*, 101, 17
- Broos, P. S., Townsley, L. K., Feigelson, E. D., Getman, K. V., Bauer, F. E., & Garmire, G. P. 2010, *ApJ*, 714, 1582
- Broos, P. S., Townsley, L. K., Feigelson, E. D., et al. 2011, *ApJS*, 194, 2
- Broos, P. S., Townsley, L. K., Getman, K. V., Bauer, F. E. 2012, *Astrophysics Source Code Library*, 1203.001
- Broos, P. S., & Townsley, L. K. 2016, ACIS Extract software package, Zenodo, <https://doi.org/10.5281/zenodo.781433>
- Cash, W. 1979, *ApJ*, 228, 939
- Davis, J. E., Bautz, M. W., Dewey, D., et al. 2012, *Proc. SPIE*, 8443, 84431A
- Fruscione, A., et al. 2006, *Proc. SPIE*, 6270
- Gehrels, N. 1986, *ApJ*, 303, 336
- Güdel, M., et al. 2007, *A&A*, 468, 353
- Hickox, R. C., & Markevitch, M. 2006, *ApJ*, 645, 95
- Hong, J., Schlegel, E. M., & Grindlay, J. E. 2004, *ApJ*, 614, 508
- Proposers' Observatory Guide, v.10
- Snowden, S. L., Mushotzky, R. F., Kuntz, K. D., & Davis, D. S. 2008, *A&A*, 478, 615
- Townsley, L. K., Broos, P. S., Chu, Y.-H., et al. 2011, *ApJS*, 194, 15
- Townsley, L. K., Broos, P. S., Chu, Y.-H., et al. 2011, *ApJS*, 194, 16
- Townsley, L. K., Broos, P. S., Garmire, G. P., et al. 2018, *ApJS*, accepted
- Wachter, K., Leach, R., & Kellogg, E. 1979, *ApJ*, 230, 274
- Weisskopf, M. C., Wu, K., Trimble, V., O'Dell, S. L., Elsner, R. F., Zavlin, V. E., & Kouveliotou, C. 2007, *ApJ*, 657, 1026
- Wilms, J., Allen, A., & McCray, R. 2000, *ApJ*, 542, 914