

APPROVAL SHEET

Title of Dissertation: Developing and Characterizing X-ray Concentrators for
 Space-Based Observations with the Neutron Star Interior
 Composition Explorer

Name of Candidate: Erin Rose Balsamo
 Doctor of Philosophy, 2017

Dissertation and Abstract Approved: _____
 Dr. Michael Hayden
 Professor
 Department of Physics

Date Approved: _____

ABSTRACT

Title of dissertation: DEVELOPING AND CHARACTERIZING X-RAY
CONCENTRATORS FOR SPACE-BASED
OBSERVATIONS WITH THE NEUTRON STAR
INTERIOR COMPOSITION EXPLORER

Erin Rose Balsamo, Doctor of Philosophy, 2017

Dissertation directed by: Dr. Michael Hayden, Ph.D
Department of Physics

A long standing need to resolve the equation of state (EOS) of neutron stars motivated the Neutron Star Interior Composition Explorer's (NICER) mission goals, including determining stellar radii to within $\pm 5\%$. This can be accomplished by observing the change in photon flux over time from pulsars (rotating neutron stars with a magnetic field) in the soft X-ray energy band (0.2-12.0 keV) using NICER's highly effective photon focusing system comprised of 56 X-ray concentrators (XRC). In this thesis, I prove the efficiency and functionality of the specialized fabrication process which allowed for the success of producing flight ready XRCs in a cost effective manner, which have been shown to exceed mission requirements through ground calibration. I have also conducted simulations of a challenging yet advantageous observation of the closest millisecond pulsar (MSP) which will provide astronomers with useful NICER data to further constrain the EOS.

X-rays are focused by grazing incident reflection with incident angles on the order of a degree. The NICER optics were designed as singly-reflecting concentrators

with a curved axial profile for improved photon concentration and a sturdy full shell structure for enhanced module stability. I assisted in developing a new substrate forming technique to accommodate these unique design elements. By analyzing hundreds of substrates' profiles post-forming, I found the profiles were copied, on average, to within $4.6\% \pm 3.7\%$, i.e. with $> 95\%$ accuracy. My ground calibration results and this analysis has shown that the heat shrink tape method is reliable, repeatable, and could be used in future missions to increase production rate and performance.

NICER's 6 arcminute field-of-view poses a challenge in resolving the energy spectra and light curves of the closest MSP, PSR J0437-4715, due to the bright nearby X-ray source, the Active Galactic Nucleus (AGN) RX J0437.4-4711, with an angular distance of 4.2 arcminutes from the pulsar. Since the optics function as concentrators, all image resolution is lost. However, due to the energy dependency of the XRC's point spread function (PSF), I have found that the best way to observe the MSP is to point the instrument 2.7 arcminutes off-axis from the pulsar, away from the AGN; the pulsar to AGN flux is maximized at this point. Within the simulations, I carefully consider the multi-dimensional instrument pointing statistics, calibrated XRC PSFs, and a current theory of neutron star emission processes.

DEVELOPING AND CHARACTERIZING X-RAY
CONCENTRATORS FOR SPACE-BASED OBSERVATIONS
WITH THE NEUTRON STAR INTERIOR COMPOSITION
EXPLORER

by

Erin Rose Balsamo

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, Baltimore County in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2017

Advisory Committee:

Dr. Michael Hayden, Chair/Advisor

Dr. Todd Pittman

Dr. Katja Pottschmidt

Dr. Markos Georganopoulos

Dr. Theodosia Gougousi

© Copyright by
Erin Rose Balsamo
2017

Dedication

Aunt Rose.

Acknowledgments

This has truly been an experience and I am greatly thankful to all who have been there for me along the way.

Without NICER, this dissertation would not have existed as we know it. Keith Gendreau, Zaven Arzoumanian, and the rest of the NICER team have worked many long hours to get this mission accepted and now she flies on the space station. To say I have touched something that is now in space is simply amazing. I would like to give a special thanks to the other scientists and engineers I worked with on a regular basis at NASA Goddard. Lalit Jalota, David Fickau, and Nick Spartana were there in the beginning providing so many different points of view and input; I really appreciate getting to work with this diverse group, even to the point of surviving an earthquake with them. Steven Kenyon and Devin Hahne, you guys are amazing engineers who cannot go without notice. And then there is the rest of the optics team, especially Takashi Okajima and Yang Soong.

My committee has been great, giving me so much input and advise and taking the time to help me with revisions while learning a little bit about the world of astronomical instrumentation. Michael Hayden, Todd Pittman, Markos Georganopoulos, Theo Gougousi, and especially Katja Pottschmidt. Katja has put in so much time and effort assisting me through my writing, answering questions, giving advise... there are not enough words to describe how much you have helped me, but there are approximately 220 pages to show for it!

Next up, I would like to acknowledge my professors and teachers throughout

the years and the schools' staff who allowed our programs to function at all. Ian George is responsible for getting me the research position at Goddard with Keith Gendreau and advising me for years. You are a great guy Ian, always helpful and understanding; I am extremely thankful to get the chance to work with you. My McDaniel professors, Jeff Marx, Apollo Mian, and Vasilis Pagonis. You made undergrad so much fun I just had to keep going. Thank you for your continued support. And lastly, Gary Mercer, my physics teacher and track coach from Western Tech. You have been so supportive over these years, I am glad to call myself a former student of yours.

Of course I have to give a special thanks to my officemates during my graduate school experience. Erika Nesvold and Diana Cheatham, you guys kept the fuel in the motor!

My mother, my father, and my brother, you have put up with me for all these years and have been supportive of me the entire time. Mom, Dad, Chris, you guys are the best! Marble was there helping me throughout my writing, sitting on my desk, even trying to type some pages for me. All she could spell was *meeoowwww*, but that's OK because it was the thought that counts. And all my friends who have come and gone in my life over the time. Every one of you believed in me and pushed me though. You all know who you are.

Table of Contents

List of Abbreviations	xix
1 Introduction	1
1.1 Scope of Thesis	2
1.1.1 Organization	4
1.2 X-ray Astronomy: History and Heritage	4
1.2.1 X-ray Mirrors	8
1.3 The International Space Station Mission: NICER	10
1.3.1 The XRCs	13
1.4 Fabrication Development	14
1.5 Hardware Performance: Calibration, Measurements, and Data Analysis	16
1.6 Simulating an Observation	18
2 The Neutron Star Interior Composition Explorer and the X-ray Concentrators	22
2.1 Design and Geometry of the X-ray Concentrators	25
2.2 Performance Parameters and Mission Requirements	29
2.3 Enhanced Performance through Curved Axial Profile Implementation	30
3 Development and Advancements of the XRC Fabrication	35
3.1 Fabrication Overview	35
3.2 Heat-Forming Aluminum Substrates	37
3.2.1 About the Heat Shrink Tape Forming Technique	38
3.2.2 Forming Mandrels Used as Shaping mold	42
3.2.3 Results Based on Foil Axial profile Measurements	44
4 XRC Ground Calibration, Performance, and Special Features of the Optics' Response	48
4.1 Calibration Facilities	48
4.1.1 X-ray Source	49
4.1.2 X-ray CCD	54
4.2 Data Analysis and Results	57
4.2.1 Interpretation of Raw CCD Output Data	57
4.2.2 Spectrum	63
4.2.3 Count Rate Stability	64

4.2.4	Calculating EEF and Effective Area	65
4.2.5	Example Analysis Report	68
4.3	XRC Performance Features	72
5	Planning the NICER Observation of Pulsar J0437-4715 from the Optics Point of View	76
5.1	Background	76
5.2	Simulation Summary	79
5.3	Objects in Field of View	81
5.3.1	PSR J0437-4715	81
5.3.2	AGN RX J0437.4-4711	85
5.4	Creating a Virtual Pulsar	86
5.4.1	Pulsar Flux: Deriving The Phase and Energy Dependent Flux $F(E, \phi)$	88
5.4.2	Generating the Emergent Intensity	92
5.4.3	Generating the Observed Spectra	93
5.5	Creating a Virtual NICER	93
5.5.1	Ray-tracing the XRCs using CONSIM with an Adjusted Scattering Function	95
5.5.2	Pointing Uncertainty	97
5.5.3	Constructing the Response Used in the Simulations	99
5.6	Tackling the Challenging Observation Simulation	104
5.6.1	Preliminary Study	107
5.7	Simulation and Data Analysis	111
5.7.1	Ideal Case	112
5.7.2	Estimating the Intrinsic Signal Using the Ideal Case Simulation	116
5.7.3	Counting with a Misalignment	116
5.8	Final Results and Conclusions	119
6	Conclusions	124
A	X-ray Physics and Interactions	128
B	XRC Fabrication	136
C	CCD Data Analysis, Troubleshooting, and Features	138
C.1	Unique Factors in X-ray Testing: Experiment Parameters	138
C.2	CCD Background Spectrum and Effect on Measured Effective Area .	139
C.2.1	Introduction	141
C.2.2	Modeling the CCD Background Spectra	142
C.2.3	Photon Counting Experiments	145
C.2.4	Noise Factor Calculation	145
C.2.5	Analysis	151
C.2.6	Conclusions	152
C.3	Dissecting CCD Pileup Caused by a High Flux X-ray Source	155

D	XRC Calibration Code	159
D.1	XrayAnalysisFunctions.py	159
D.1.1	Start-up variables and module imports	160
D.1.2	Load and Select Data	161
D.1.3	EEF Functions	165
D.1.4	Effective Area Functions	166
D.1.5	Other Functions	169
E	Simulated Observation Code	176
E.1	Sky X-ray Background	176
E.2	Pulsar Table Model	178
E.2.1	PulsarTableModelMaker.py	178
E.2.2	NeutronStar.py	181
E.2.3	Flux.py	183
E.3	McPHAC Input Parameters	184
E.4	Virtual NICER Codes	185
E.5	Simulation Analysis Codes	192
F	AGN RX J0437.4-4711 Variability in the Literature	207
F.1	Wang et al. (1998)	207
F.2	Halpern and Marshall (1996)	209
F.3	Grupe et al. (2001)	211
F.4	Summary of Variability	211

List of Tables

2.1	Key features and specifications about the NICER X-ray Timing Instrument and the X-ray Concentrators (Arzoumanian et al., 2014; Gendreau et al., 2012).	25
3.1	Specifications of Dunstone Company Inc.’s HI-SHRINK POLYESTER as seen on the company’s website (Dunstone Company Inc., 2013). . .	42
3.2	Results containing the root mean square values calculation from considering the difference between the forming mandrel profile and the average profile measured after heat-forming foil substrates. Column 3 is the RMS of the difference between the average foil profile and average mandrel profile for each size. Column 4 is the percent difference of the average formed foil profile from the mandrel profile. The value is calculated by dividing the RMS value in Column 3 by the measured mandrels’ bulge size. The final column reports the number of foils sampled per shell size. The key for the table column names is as follows: $\text{RMS}(\text{values})$ = root-mean-square of <i>values</i> ; M = average mandrel profile as calculated from measured scans; F = average foil profile as calculated from measured scans of individual foils.	47
4.1	Event File Columns	58
4.2	Pixel Values for Determining Grade	61
4.3	Measurement parameters of the example analysis report shown in this chapter.	69
5.1	Absorbed double blackbody model parameters for PSR J0437-4715 used in this work. These parameters are based on the fitted spectra in Halpern et al. (1996). In XSPEC, the normalization constant for a blackbody model is defined as “ L_{39}/D_{10}^2 , where L_{39} is the source luminosity in units of 10^{39} erg/s and D_{10} is the distance to the source in units of 10 kpc” (Arnaud et al., 2013).	82
5.2	Atmospheric emission spectral model parameters I used to simulate emission from PSR J0437-4715. The hot spot components were added to XSPEC as table models. The values are based on Bogdanov (2013). Refer to Table 5.1 for the definition of the blackbody normalization constant. The power law normalization is defined by XSPEC as “photons/keV/cm ² /s at 1 keV” (Arnaud et al., 2013).	85

5.3	Absorbed power law spectral model parameters for AGN RX J0437.4-4711 used in this work. The values are based on Halpern and Marshall (1996). Refer to Table 5.2 for the definition of the power law normalization constant.	86
5.4	Geometry values assumed for PSR J0437-4715. These are based off of Bogdanov (2013) (table 2, line 6).	87
5.5	Source-specific McPHAC input parameters for PSR J0437-4715 emergent spectra simulation. These values are for the emergent radiation from the hot spots only. See Table 5.2 for a complete list of the parameters.	93
5.6	Ray-tracing scattering function parameters fitted to actual XRC measurements.	96
5.7	This shows the required maximum allowed pointing uncertainty for the boresight (XTI pointing direction) and the XRCs based on a normal distribution. These values reflect those reported in NICER’s mission proposal (Gendreau, 2012).	98
5.8	Example calculations for estimating the pulsar signals. S is calculated using Equation 5.8. The time observed per phase bin in a 1 Ms observation, t_{exp} , is 50,000 s. These values are part of the 0.2–0.5 keV dataset shown in Figure 5.23.	119
C.1	Lorentzian fit parameters to model peak from CCD pile-up data . . .	157
F.1	Summary of AGN RX J0437-4711 Observations	208

List of Figures

1.1	Image of the NICER payload <i>Photo Credit: NASA/Keith Gendreau</i>	3
1.2	The ideal Wolter 1 (Wolter, 1956) geometry utilizes a parabolic surface as the primary mirror (or first reflector) and a hyperbolic surface as the secondary mirror (or second reflector). Two reflectors are used for imaging optics. If the second reflector is removed, the parabolic shape of the primary mirror will focus all the incident beams parallel to the optical axis, functioning as a concentrator.	10
1.3	Final calibration results of the flight ready modules (ones which passed the requirements) are shown here in the top plot as blue circles. The green dashed lines indicate the average 2 mm fraction ($91.3 \pm 1.8\%$) and effective area ($48.06 \pm 2.73 \text{ cm}^2$) of these modules. In comparison, the mission requirements, shown as black dashed lines, specified the 2 mm fraction to be at least 88% and the effective area to be at least 42 cm^2	21
2.1	Schematic of NICER deployed on the space station (Kenyon, 2013).	22
2.2	Every NICER (Gendreau et al., 2012) XRC module consists of 24 nested aluminum foil shells with epoxy replicated gold reflecting surfaces. A single XRC is approximately 10 cm in diameter and in height. The inner most reflector has a radius of 15.11 mm and the outer most has a radius of 51.20 mm.	24
2.3	Each reflector, or “foil,” in the XRC takes on a slightly different parabolic shape so that the set of nested reflectors will have a common focal point. (a) The gray dashed lines extrapolate from the foil locations to show the full parabolic profiles of all reflectors that were calculated to make the nested reflectors confocal. The blue solid lines show from where on the parabolas the XRC foils take their shape. In this figure, the common focal point is set at the origin and the vertices’s of each parabola are at small negative values on the y-axis. (Due to the figure’s scale, the focal point and vertices’s appear to overlap but they do not. Also, the gray lines appear to have a break in them, but that is due the location of the dashes which have no significance.) (b) A zoomed-in version of the cross-section shows labeled nested shell numbers as a reference.	28

2.4	Theoretical simulations and data measurements of a NICER XRC X-ray image are compared. A 2D histogram was created in SAOImage DS9 Version 7.2 (Joye and Mandel, 2003) using events (see colorbar) from an X-ray measurement of a flight quality NICER XRC module. Overlaid with these events are the contours (blue lines) of a ray-traced NICER module assuming a flat profile and considering scattering at the reflecting surfaces. The inner red dashed circle represents the 2 mm diameter aperture for which NICER requires the enclosure of 88% of total energy throughput. The solid circles, from smallest to largest, represent the X-ray data half power diameter (HPD) (smaller black circle), X-ray data 90% EEF (larger black circle), the 90% EEF of the ray-traced simulation without scattering (smaller green circle), and total intrinsic blur which would result from a theoretically perfect (without scattering) flat profile NICER XRC module (larger green circle). The outer most magenta dashed circle is the 90% EEF from the ray-traced simulation with scattering.	32
2.5	The EEFs of measured X-ray data of a NICER XRC module compared to the EEFs of ray traced data. The X-ray data are denoted by the solid line and are similar to that of the ray traced data using a simulated parabolic profile, denoted by the dash-dotted line. This simulation considered an estimated scattering function (see Section 5.5.1 for details on the scattering function). Without scattering, the theoretical EEF would go to 1 at $r=0$. The dashed line and the connected x's are the EEFs of ray traced data which used a simulated conical (flat) profile. The dashed line shows the data that did not consider scattering function where the connected x's show the data that did.	34
3.1	Epoxy Replication Process Schematic. (left) A glass Pyrex mandrel is placed in a chamber where gold is applied onto the mandrel's exterior surface through sputtering deposition. (middle) The gold covered mandrel is sprayed with an even coat of epoxy. (right) A formed foil is placed around the epoxy covered gold mandrel and subjected to a constant pressure. Once cured, the foil is removed from the glass mandrel and epoxy has transfered the gold to the foil's interior surface.	37
3.2	Heat shrink tape forming setup of a size 20 NICER foil. The precision cut foil is wrapped around the forming mandrel and compressed by heat shrink tape which is secured by Kapton Tape. The foil is flush with the bottom of the mandrel.	41
3.3	Design specifications for the bulge on the aluminum forming mandrels for NICER in comparison to the measured bulges on the manufactured forming mandrels. There is an unexplained systematic deviation between the design and measured bulge sizes that was not statistically significant enough for further investigation.	43

3.4	The complete results from the profile scans are shown here. The dashed line represents the design specifications for each curve. The solid line is the measured actual on the forming mandrels. The solid line with 'x's is the average profile of the formed substrates using the shrink tape heat-forming method. The RMS values of the difference between the mandrel and formed foil profiles are shown for each shell size.	45
3.5	The plot shows the overall accuracy of the shrink tape heat-forming technique by comparing data from measured formed foil profiles to a flat profile representing a completely unformed foil. The data in this graph are the RMS value of the difference between the forming mandrel and the foils' profile. The smaller the value, the closer the foil profile is to the actual mandrel profile.	46
4.1	GSFC's 600 meter beamline at GGAO (Goddard Geophysical and Astronomical Observatory, 2014). <i>Photo Credit: Erin Balsamo</i>	49
4.2	Diagram of X-ray source. A current applied across a filament allows electron production. A potential difference, provided by a high voltage power supply, accelerates those electrons to a heavy metal target. X-rays, at energies characteristic to the target material, are produced when the high-energy electrons collide with the metal target. During XRC calibration, I had control of the current (filament power supply), the high voltage (potential difference), and the material of the heavy metal target.	50
4.3	Setup of the Manson X-ray source in the "Source Hut" of the 600 m beamline.	52
4.4	(left) Manson source connected to the beamline. (right) A close-up view of the Manson source under the high voltage shield.	53
4.5	Attenuator. (left) The photon flux attenuator was positioned in front of the X-ray source generator to decrease the number of photons incident on the XRC. (right) A 3D printed mount was constructed allowing me to use different thicknesses of aluminum to control the attenuation.	53
4.6	Liquid nitrogen cooled CCD. (left) LN Dewar (black) is attached to the X-ray CCD. In this photograph, the CCD housing was placed while under vacuum and tested for air leaks, prior to being used at the beamline for X-ray testing. (right) The surface of the CCD is shown.	55
4.7	An XRC in the vacuum chamber at the detector end of the 600 m beamline. This is the point of view that the X-ray CCD had of the XRC.	55

4.8	A spectrum of the data set is made and the X-ray source lines are found. The pulse height for the lines is found and converted to energy in units of keV. In this case, the pulse height is 253 for a line at energy 1.49 keV so the conversion factor is 169.9 ph/keV as found by Equation 4.2. The standard deviation of ± 32.3 ph corresponds to an energy width of ± 0.2 keV.	59
4.9	These patterns show the “good” grades. For each pattern example, a 3 by 3 pixel region is outlined. The red circle represents where the charge cloud interacted with the CCD. The center pixel is main pixel of the interaction. The shaded regions are all the pixels affected by the charge cloud. The grade values are labeled under each pattern. Type 1 is shown by (a). Type 2 is shown by (b)-(e). Type 3 is shown by (f)-(i). Type 4 is shown by (j)-(m). A pattern outside of these would be considered a “bad” grade.	62
4.10	This is a histogram of events based on grades from an example sample data set. Labeled and colored in red are the events with “good” grade values.	63
4.11	Example light curve of an XRC calibration X-ray test. The incident beam measurements, blue and red, show a much lower count rate than the XRC Image, green, since the photon are not focused. The events counted in this light curve had an energy within a 3σ width of the pulse height spectrum peak. The flux of the total incident beam was calculated by combing the data of the first and second incident beam measurements.	65
4.12	A pictorial representation of the EEF error calculation from Equations 4.3 and 4.4.	67
4.13	Sample EEF found from an XRC X-ray calibration test. The events here had an energy within a 3σ width of the pulse height spectrum peak.	67
4.14	X-ray image of focused XRC data taken in a beamline measurement. (Left) 2D histogram. (Right) Scatter plot of events.	70
4.15	Energy spectrum, in units of the pulse height, of the XRC measurement of a single, flight ready, module. The data are shown in gray fit to a Gaussian shown by the black line. The fit parameters are printed on the plot.	71
4.16	Energy spectrum, in units of the pulse height, of an incident beam measurement collected prior to the XRC measurement. The data are shown in gray fit to a Gaussian shown by the black line. The fit parameters are printed on the plot. This spectrum should be similar to the incident beam measurement collected after the XRC was measured (Figure 4.17) in order for the measurement to be considered successful and useful for calibration.	72

4.17	Energy spectrum, in units of the pulse height, of an incident beam measurement collected after the XRC measurement. The data are shown in gray fit to a Gaussian shown by the black line. The fit parameters are printed on the plot. This spectrum should be similar to the incident beam measurement collected before the XRC was measured (Figure 4.16) in order for the measurement to be considered successful and useful for calibration.	73
4.18	Sample XRC calibration output. This shows the quantifying factors in determining the quality of a single XRC module.	74
4.19	Performance results of all 70 XRCs produced.	75
5.1	Current constraints for neutron star masses to radii are shown in the bottom plot along with current EOS models (black, blue, and, yellow lines and the blue and yellow shaded regions) and the constraint NICER will provide on the measurements (red region). The shaded gray region is the current mass and radius constraint for the labeled pulsar. The top plot shows the predicted constraints NICER will have on measuring the radius of PSR J0437-4715 for different lengths of measurements. This figure can be found in the About section of the NICER homepage (NASA, NICER, 2017). <i>Image credit: NASA</i>	78
5.2	Model energy spectra for AGN RX J0437-4711 and MSP PSR J0437-4715 created in XSPEC. The AGN is shown by the solid green line for an absorbed power law spectrum. Two different models of the pulsar are shown. The solid blue line is an absorbed double blackbody where the blue dashed and dotted lines are the blackbody components. BB(1) is the $kT = 8 \times 10^{-2}$ keV component and BB(2) is the $kT = 0.24$ keV component. The solid red line shows the absorbed atmospheric emission model from two hot spots, a cool blackbody, and a power law. The thick lines are for the hot spots where the dashed line is the primary spot at the phase of maximum flux and dashed-dotted line is the secondary spot at the phase of maximum flux. The thin dashed line is the cool blackbody and the dotted line is the power law. All model parameters are given in Tables 5.1, 5.2, and 5.3. The primary hot spot is typically defined as the emission spot seen more often by the observer than the secondary hot spot.	83
5.3	Pulsar Geometry. Shows variables used in text. Based on Figure 1 in Viironen and Poutanen (2004).	89
5.4	Comparison of the EEF from ray-tracing using scattering function fit parameters from Table 5.6 (blue line) to the average EEF from XRC calibration data (red line).	97
5.5	Randomly generated error on XRC alignment based on a pointing uncertainty requirement of $35''$ radius at 3σ	98
5.6	Randomly generated error on XTI boresight based on a pointing uncertainty requirement of $85''$ radius at 3σ	99

5.7	Example of pointing vectors for an instance of a Virtual NICER. The blue star is the AGN and the green star is the pulsar. The red x is the observation direction. Each black dot is the pointing direction of one XRC, considering boresight and XRC alignment errors. (top) XTI pointed 2' away from pulsar, towards AGN, (middle) XTI pointed at pulsar, (bottom) XTI pointed 2' away from pulsar, opposite of AGN.	100
5.8	Filter files used to generate NICER responses. Right: The effective area of a single XRC for different off-axis sources. This is the basis of the optical efficiency <i>efffil</i> files. Left: The NICER filter transmission (<i>filfil</i> , red) and detector efficiency (<i>detfil</i> , blue) are used along with an optical efficiency (<i>efffil</i>) to generate response functions using <i>genrsp</i> .	102
5.9	Responses as effective area curves for observing the pulsar and AGN. It is shown when NICER is pointing directly at the pulsar as well as 1', 2', and 3' off-axis, away from the AGN.	103
5.10	This flow chart shows the process in which to run the files in Appendix E.4 in order to produce virtual NICERs and generate the response functions needed in the simulations.	104
5.11	Examples of simulated observations showing pulsar spectra at 4 different rational phases and the AGN spectrum. NICER is pointed off-axis by 2.0' opposite the AGN direction.	106
5.12	This flowchart outlines the steps in the simulation for estimating the pulsar signal. (The preliminary study (Section 5.6.1) explores a subset of this parameter space, as described in the text.)	107
5.13	Pointing directions for the boresight and XRCs of 3 NICER instances.	108
5.14	<i>Preliminary study.</i> Pulsar and AGN count rates for NICER instance #03 at different energy bands and offset pointing angles ranging from 4.5' in the direction of the AGN to 3.5' opposite the AGN direction (negative values are towards the AGN). The solid lines represent the pulsar's count rates and the dashed lines represent the AGN. Black is 0.2–8.0 keV (the full range), blue is 0.2–0.5 keV, red is 0.5–3.0 keV, and magenta is 3.0–8.0 keV. An absorbed double blackbody plus power law model (Table 5.1) was assumed for the pulsar in this plot. The AGN model is described by an absorbed power law (Table 5.3).	109
5.15	<i>Preliminary study.</i> Pulsar to AGN count rate ratio for NICER instances #03, #13, and #17 at different pointing angles. Dashed black lines are drawn at the 5 and 10 times pulsar to AGN count rate levels. Spectral models described in Figure 5.14 and in the text.	110
5.16	<i>Preliminary study.</i> Count rate analysis using NICER instance #3. There is evidence that the count rate ratio between the sources suggest energy dependence, as seen by the different locations of the peaks on the plot to the right. (Left) Count rate plot of the AGN and pulsar for different energy bands. This is a zoomed in version of Figure 5.14. (Right) Ratio of pulsar to AGN count rates with maximums labeled.	111

5.17	Count rate comparison of the AGN and pulsar at different pointing angles in the 0.2–10.2 keV energy range when alignment is assumed to be ideal.	113
5.18	Ratio of pulsar to AGN counts observed for the ideal alignment case within the 0.2–10.2 keV energy band across several pointing directions. The peak CRR (count rate ratio) is marked at 3.0' off-axis along with the 2.5' direction. The optimal pointing direction was found to be between these when instrument misalignment is assumed in Section 5.7.3.	114
5.19	The XRC effective area response ratio between the two sources is compared at several energies and across a range of pointing directions. An observation at a lower energy should have the XTI pointed further off axis than needed for the higher energies due to energy dependent XRC response.	114
5.20	The light curves for 2 different simulated pointing angles are compared. This shows the observed light curve (solid line), pulsar contribution (dashed line) and AGN contribution (dotted line).	115
5.21	This flowchart outlines how I conducted the observation simulations described in Section 5.7.3. The code can be found in Appendix E.5.	117
5.22	Random pointing errors of the NICER realization sampled in this section.	118
5.23	The pulsar contribution to the overall (T_{obs} , observed) light curve is estimated ($S_{\text{estimated}}$, calculated signal) from the observed light curve for a random distribution of XTI and XRC alignments in the three energy bands. The overall lightcurve is the sum of the synthetic pulsar (S_{syn} , synthetic signal) and AGN (B_{syn} , not shown) signals. The alignments are shown in Figure 5.22. The estimate was calculated assuming the CRR (R_{ideal}) from Figure 5.18, using Equation 5.8. In this example, the offset pointing angle was set to 3.0'. The uncertainties were calculated by taking the standard deviation of the total number of counts and dividing it by the observation time of each phase bin. The resultant uncertainties are shown by error bars on the calculated signal and the difference between the synthetic and calculated. (see first part of <i>ReconstructPulsarSignal.py</i> on page 196)	121
5.24	Relative difference of the estimated pulsar contribution to the light curve compared to the actual pulsar lightcurve over pointing angle. This sort of data was analyzed for 10 energy ranges for all 20 instances of NICER alignment. The data were fit with a linear function to estimate the location of minimum difference.	122
5.25	The representative pointing angle suggested for every energy band is shown with the data points of optimal pointing from each of the 20 NICER alignment realizations tested.	123
A.1	The optical constants δ and β as a function of energy.	132

B.1	Poster of the XRC fabrication process. I made this for the NICER site visit to help the team become accepted as a NASA Mission of Opportunity	137
C.1	Changes in the pulse height spectra and counts per split level at different attenuator thickness placed after the X-ray source in the 600m beamline.	139
C.2	Changes in the pulse height spectra and counts per split level for different event threshold settings on the roper server (600m beamline).	139
C.3	Changes in the pulse height spectra and counts per split level for different split threshold settings on the roper server (600m beamline).	140
C.4	Changes in the pulse height spectra and counts per split level for different high voltage setting on the X-ray source (600m beamline).	140
C.5	Changes in the pulse height spectra and counts per split level when the position of a fan blowing on the CCD changed, outside of the chamber (600m beamline).	141
C.6	Pulse height spectra of four data sets collected to measure the CCD background. These data sets were collected by choosing different event and split thresholds (ET and ST respectively). When ET was set to 150, there were more lower energy events than when ET was set 180; the red and green lines have a higher count rate then the blue and cyan. The change in the ST setting shifted the spectrum peak of the lower energy events. The higher ST, 10, resulted in a peak shifted to a lower energy than the lower ST, 0; the blue and green peaks are to the left of the red and cyan.	143
C.7	CCD Background spectrum fit to the model described by Equation C.1.	144
C.8	Data Set 1: CCD background effect on data. An incident and XRC beam spectra are compared to CCD background spectrum. Show that background may have accounted for about 1% of the events in the 1.5 keV energy band. (a) Left- XRC (blue), incident (green), and CCD background (magenta) spectra with fits. Right top- Incident minus background spectra (blue) and XRC minus background spectra (green) Right bottom- background to incident ratio (green) and background to XRC ratio (blue). (b) Lightcurves in counts per mm ² over time (s). Top- incident beam data. Middle- XRC data. Bottom- CCD background data. Blue: 1.24-1.74 keV. Green: 0.98-1.99 keV. Red: 0.73-2.4 keV.	146
C.9	Data Set 2 (see caption to Fig. C.8)	147
C.10	Data Set 3 (see caption to Fig. C.8)	148
C.11	Noise factor calculations and possible changes in the effective area measurements. Due to significant change in the effective area values, background should be minimized during the experiment.	152

C.12	Acceptable CCD background levels by comparing ratios of number of events. The noise factor (nf) distinguishes between acceptable and unacceptable number of events collected in a measurement in order to properly calculate the effective area. n_b =(number of events in CCD background), n_{foc} =(number of events from XRC focused data), n_{inc} =(number of events from incident beam data)	154
C.13	The image here is an example of pile-up with a hole where the peak should be. This data set is used to estimate the peak and then the required attenuator thickness.	155
C.14	Histogram of the data points along the center of the image. A few points around the peak are selected to be fitted with a Lorentzian curve in order to estimate the size of the peak if there was no pile-up.	156
C.15	The selected points in the data set are fitted to a Lorentzian function to estimate what the peak would have been if pile-up was not a factor.	157
C.16	Here the fitted Lorentzian is plotted on top of the data points. We see that the data plateaued at 89 counts while it would have been at 361 counts. The peak is a factor of 4 larger than the plateau.	158
C.17	The calculation attenuation of 1.48670 keV photons through and Aluminum slab. To decrease the flux by a factor of 4.06, the attenuator must have a thickness of 12.9 microns.	159
E.1	A 1° cone is shown (green circle) around PSR J0437-4715 (green cross) from ROSAT PSPC data. This area was used to calculate an average X-ray background of 1.544 ± 0.061 ct/s for the 0.76–2.04 keV energy band. The image was generated by the X-ray Background Tool (Sabol, E. J., 2017).	177

List of Abbreviations

AGN	Active Galactic Nucleus
BB (or bbody)	Blackbody
CCD	Charge-Coupled Device
CONSIM	Concentrator Simulator
CRR	Count Rate Ratio
CTE	Coefficient of Thermal Expansion
EA (or A_{eff})	Effective Area
EEF	Encircled Energy Function
EOS	Equation of State
EPIC	European Photon Imaging Camera
EPIC MOS	EPIC Metal Oxide Semi-conductor
ESA	European Space Agency
FITS	Flexible Image Transport System
FOV	Field of View
GGAO	Goddard Geophysical and Astronomical Observatory
GSFC	Goddard Space Flight Center
HPD	Half Power Diameter
HS	Hot Spot
HWHM	Half Width at Half Max
ISS	International Space Station
JAXA	Japan Aerospace Exploration Agency
LOS	Line of Sight
LSM	Laser Scanner Micrometer
McPHAC	McGill Planar Hydrogen Atmosphere Code
MO	Mission of Opportunity
MSP	Millisecond Pulsar
NASA	National Aeronautics and Space Administration
NICER	Neutron Star Interior Composition Explorer
PH (or ph)	Pulse Height
PHA	Pulse Height Amplitude
phabs	Photon Absorption
PL	Powerlaw
PSF	Point Spread Function
PSPC	Position Sensitive Proportional Counters
RMS	Root-mean-square
ROI	Region of Interest
SDD	Silicon Drift Detector
XRC	X-ray Concentrator
XSPEC	X-ray Spectral Fitting Package
XTI	X-ray Timing Instrument

Mentioned Missions

Ariel 5	—
ASCA	Advanced Satellite for Cosmology and Astrophysics
ASTRO-H	<i>see Hitomi</i>
Chandra	—
Einstein Observatory	<i>(originally HEAO-2)</i>
EXOSAT	European X-ray Observatory Satellite <i>(originally HELOS)</i>
HEAO-1	High Energy Astrophysical Observatory
Hitomi	<i>(originally named ASTRO-H)</i>
NICER	Neutron Star Interior Composition Explorer
NuSTAR	Nuclear Spectroscopic Telescope Array
OSO-8	Orbiting Solar Observatory
ROSAT	Röntgen Satellite
RASS	ROSAT All Sky Survey
RXTE	Rossi X-ray Timing Explorer
SAS-3	Small Astronomy Satellite
Suzaku	<i>(originally named ASTRO-E2)</i>
Uhuru	—
XACT	X-ray Advanced Concepts Testbed
XMM-Newton	X-ray Multi-Mirror Mission - Newton

Chapter 1

Introduction

Organizing this thesis was quite a challenge. Sure, *my* story had a beginning, but *the* story is a continuous cycle. It is like the philosophical dilemma of the chicken and the egg; which came first, the hardware or the science? Do we pose scientific questions because we have the capability to answer them or do we pose a question and then invent the technology to find the answer? I believe it is a little bit of both.

After looking at the night sky and answering ancient philosophical questions using only our eyes as observation tools, we began to use our resources, such as mathematics and technology, to assist us in developing more in depth questions. Jumping forward to the 20th century, X-rays were observed as emission from celestial sources (Burnight, 1949; Giacconi et al., 1962). However, the questions that can be answered by observing the X-ray spectrum require collecting X-ray photons and be able to “see” what each photon is telling us. We do this with space-based X-ray telescopes consisting of carefully selected optics and detectors.

Some X-ray detectors are better at telling us about certain photon properties while others are better at distinguishing other properties; some types of detectors specialize in providing superior energy resolution while others may excel in conserving time or spatial resolution, or even polarization (Fraser, 1989). Different styles of lenses also may have special niches, like superior spatial resolution or high pho-

ton throughput (Spiller, 1994). Knowledge gained from an initial observation of some celestial source leads to additional follow-up questions. We can then determine what photon properties would be more useful in the next observation. Does the observed energy range need to be expanded? Would improved time resolution answer questions about a certain scientific theory? Is it more important to conserve spatial resolution or flux? Then we go back to the drawing board, optimize the instruments, and do it all over again. Developing technology to learn about our universe is a never ending cycle.

1.1 Scope of Thesis

This thesis revolves around work I have completed while working as part of the Neutron Star Interior Composition Explorer (NICER) team. NICER is a planned International Space Station (ISS) mission that will study the extreme physics associated with neutron stars (Gendreau et al., 2012). I have worked within the optics sub-group of the team on research, development, and calibration of NICER’s optics, which I overview below. From my work, this thesis contains components of hardware development and associated data analysis along with observational simulations directly related to the optics’ unique characteristics that separate those from ones used in other missions. An image of the completed NICER payload is shown in Figure 1.1.

Through this chapter, I will introduce the main aspects of this thesis along with some background on NICER, X-ray astronomy, and X-ray optics.

Note about Timeline of My Involvement

I started by working with the X-ray Advanced Concepts Testbed (XACT) (Gendreau et al., 2012) in 2010 until that mission was canceled around 2012. During this time, I contributed by designing a telescope structure and parabolic mirror mount to be used for optics alignment via optical collimation. This ultimately was used in lab experiments during NICER's build. I also used this time to begin the research I continued later with the fabrication development and performance analysis for the NICER XRCs. I carried my research from XACT over to the NICER concept study phase and continued with the group to the end of flight fabrication in 2015. This work was performed at NASA Goddard Space Flight Center (GSFC).



Figure 1.1: Image of the NICER payload *Photo Credit: NASA/Keith Gendreau*

1.1.1 Organization

Section 1.2 covers a brief background of the origin and evolution of X-ray astronomy. In Section 1.3, I discuss NICER, with more details to be found in Chapter 2. Fabrication challenges and developments of the NICER optics are discussed in Section 1.4, along with my accomplishments and contributions to the project, which is expanded upon in Chapter 3. Section 1.5 summarizes Chapter 4, which explains how I conducted performance testing on the X-ray Concentrator (XRC) modules and reports the final calibration results. The final section explains how I applied my knowledge and experience with the XRCs to simulate a future neutron star observation discussed in Chapter 5.

My refereed paper, titled “Shrink Tape Technique For Heat-Forming Aluminum Substrates For Thin Foil X-ray Mirrors And The NICER X-ray Concentrators” (Balsamo et al., 2016), is the main basis for Chapters 2 and 3. In Chapter 2, I included extra information valuable for other work in this thesis, especially in Section 2.2: “Performance Parameters and Mission Requirements.”

1.2 X-ray Astronomy: History and Heritage

X-ray observations (ranging from 0.1 keV to 10 keV for soft X-rays) are important in many areas of astrophysics and space exploration (Arnaud et al., 2011). Discovered in 1885 by Wilhelm Röntgen in his lab, X-rays have a relatively short mean-free path in the atmosphere, traveling only a few feet, preventing ground-based observations (Arnaud et al., 2011). After the development of high-altitude

rocketry during World War II, the first extraterrestrial X-rays were observed of the Sun by using a captured V-2 rocket (Friedman et al., 1951). In 1962, the first extra-solar source, the X-ray binary Scorpius X-1, and an isotropic X-ray background were discovered by Riccardo Giacconi’s team (Giacconi et al., 1962) in a sounding rocket¹ experiment. This discovery led to missions resolving individual cosmic X-ray sources, giving birth to X-ray astronomy (Giacconi et al., 1969).

The first X-ray astronomy missions used simple Geiger counters or proportional counters as detectors (Fraser, 1989). Combined with rocket pointing systems (that had been used for many previous years to study the sun) and a collimator, cosmic X-rays could be detected but not resolved spatially (Giacconi et al., 1969); an image was not created. A better understanding of X-ray sources came from resolving and detecting fainter objects through improving technology. By the end of the 1960s, about 60 sources had been discovered (Drake, 2014).

X-ray astronomy advanced in the 1970s with the launch of dedicated X-ray satellites including Uhuru, Ariel 5, SAS-3 (Small Astronomy Satellite), OSO-8 (Orbiting Solar Observatory), and HEAO-1 (High Energy Astrophysical Observatory) (Newman, 2014). These missions, along with the continuing sounding rockets and balloon flights, increased the number of known extra-solar X-ray sources to 680 by 1980 (Drake, 2014), finding even more classes of astronomical objects emitting X-rays including binary star systems, supernova remnants, radio galaxies and galaxy clusters (Giacconi et al., 1969). The potential of better spatial, spectral, and tem-

¹The name ‘sounding rocket’ comes from the nautical term “to sound” which historically means “to take measurements.” NASA continues to use sounding rockets today for scientific measurements that require a short collecting time (a few minutes), and a sub-orbital altitudes (less than a few hundred kilometers).

poral observations in the X-ray band to constrain physical models of the individual sources fueled the drive for improvement of instrumentation that could realistically be flown in space. The instrument sensitivity was the limiting factor mainly from the lack of focusing optics (Giacconi et al., 1979).

Reflecting X-ray focusing optics were introduced as flight ready instruments in the late 1970s and were in common use by the 1980s. Even though the commonly used geometry was patented by Hans Wolter in 1956 (Wolter, 1956), they were not planned for a flight until the mid-1970s due to difficulties in fabrication, launch, and efficiency (Gorenstein, 2012). The Einstein Observatory² was the first imaging X-ray satellite, the first true X-ray telescope (Giacconi et al., 1979; Gorenstein, 2012). This mission, and subsequent missions like EXOSAT, discovered more X-ray sources increasing the number by another factor of 10 to reach about 8,000 by 1990 (Drake, 2014).

ROSAT³ was introduced in the 1990s, as an all sky soft X-ray survey mission designed to catalog X-ray emitting sources, increasing the number to a quarter million (Voges et al., 1999) and by the early 2010s there are well over a million (Drake, 2014) known sources. Other than increasing the number of known sources, technology has also allowed us to better analyze the physical properties of these objects. The Rossi X-ray Timing Explorer (RXTE), launched in 1995, was designed for the study of time variability of X-ray emission from select sources (Swank, 1999).

Analyzing emission variations over time allows us to interpret how matter near com-

²The Einstein Observatory was previously named HEAO-2 before launch in part of the High Energy Astrophysical Observatory satellite series.

³ROSAT's energy band was 0.1-2.5 keV.

pact objects, like neutron stars and black holes, is affected by the local environment (Wang et al., 1998), helping us to understand the underlying physics. Due to the instruments timing capabilities, RXTE data were used in the discovery of evidence for the first known millisecond pulsar (MSP) that was powered by accretion (Zavlin and Pavlov, 1998).

To this day, the number of known sources, and quality of data retrieved, continues to increase from observations using the Chandra X-ray observatory, XMM-Newton (Jansen et al., 2001), ASCA (Serlemitsos et al., 1995), Suzaku (Mitsuda et al., 2007), NuSTAR (Koglin et al., 2009) and most recently, Hitomi⁴ (Soong et al., 2014). The advances in technology, specifically regarding focusing optics, have granted astronomers the ability to discover many faint sources by increasing the signal-to-noise ratio. With Chandra’s superb spatial resolution, Suzaku’s and XMM-Newton’s high effective area, and NuSTAR’s broad energy range, X-ray astrophysics had become a growingly extensive field providing insights ranging from planetary science to cosmology (Arnaud et al., 2011). NICER will contribute by offering the highest timing resolution flown on any astronomy instrument, 30 times more sensitive than RXTE, with about 8 times the energy resolution (Arzoumanian et al., 2014).

⁴Astro-H was renamed to Hitomi after launch and may be referred to as Astro-H in this thesis. However, the satellite lost communication shortly after launch and is no longer collecting data (JAXA, 2016).

1.2.1 X-ray Mirrors

Each of the missions mentioned above, aside from RXTE, used the Wolter I geometry for grazing incidence X-ray telescopes but had vastly different specifications due to the chosen material and fabrication methods (Petre, 2010). Mirror compositions were decided based on the mission’s scientific objectives.

Photons at X-ray energies interact very differently with matter than lower energy light waves, such as optical light. A glass lens would not be able to focus X-rays (see Appendix A). Due to this nature, focusing X-ray telescopes must be made using grazing incident reflections (Spiller, 1994). As an example, the critical reflecting angle of a 1 keV photon by a gold surface is 3.72° . Thus, to make an efficient reflector using a gold surface, the incoming photons must be incident at an angle less than this (relative to the surface). An even smaller grazing angle is required for higher energies. The reflector shape must also be chosen such as to focus the incoming rays onto the same plane. A popular geometry used in X-ray missions was developed, and patented, by Hans Wolter in the 1950s (Wolter, 1956). He suggested three geometries creatively named “Wolter I,” “Wolter II,” and “Wolter III.” Of these, the Wolter I has the shortest focal length and is preferred for space flight experiments (Serlemitsos, P. J., 1988). Thus the others will not be discussed here.

The Wolter I geometry involves two reflections off the concave side of curved surface as shown by the cross-section in Figure 1.2. (The surfaces are symmetrical around the axis, and hence circular when viewed along the optical axis.) The first

surface is a paraboloid. Light parallel to the axis will focus at $f = 1/4a$ where a is the second order coefficient of the parabola (Spiller, 1994). The second surface is a hyperboloid. True imaging optics requires two reflections (or some even number of reflections) according to the Abbe sine condition (Wolter, 1956). The primary surface serves as a "concentrator." Alone, such a surface works well only for on-axis point sources and thus does not resolve extended sources. Adding the second reflector minimizes the degradation of off-axis sources (Serlemitsos and Soong, 1996; Serlemitsos et al., 2010) but at the same time reduces the total flux output of the optical system.

Looking 'down' such a telescope along the optical axis, only photons coming in on a very thin ring will be brought to a focus. This means that a single shell is relatively inefficient with its lack of collecting surface area normal to the incident ray. The overall efficiency can be dramatically improved by nesting con-focal shells within one. This increases the effective collecting area while occupying the same amount of space, but also increases mass. The number of nested shells, and thus collecting area, depends upon the fabrication choices for the optical system.

The reflecting surface must be atomically smooth to reduce micro-roughness scattering and have a high reflectivity described by the Fresnel equations, Eq A.12 (see Appendix A for the physics of X-ray interactions) (Serlemitsos, P. J., 1988; Spiller, 1994). The choice of surface materials depends on the free electron density (N_e) and the reflectance at small grazing angles (Petre, 2010; Soong et al., 1995; Spiller, 1994). A higher N_e results in a wider range of energies because of low attenuation for higher energy photons. Lower N_e results in better reflectance at a

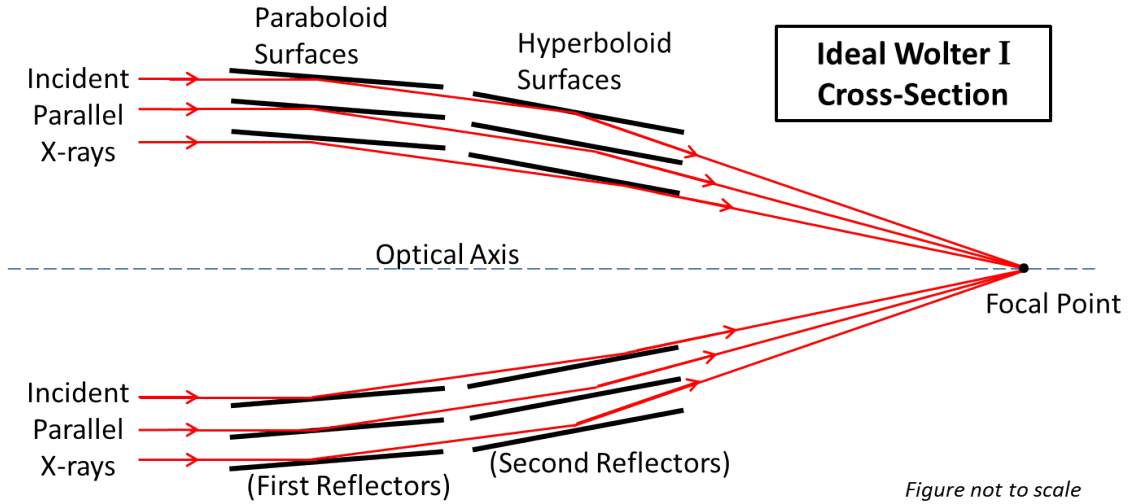


Figure 1.2: The ideal Wolter 1 (Wolter, 1956) geometry utilizes a parabolic surface as the primary mirror (or first reflector) and a hyperbolic surface as the secondary mirror (or second reflector). Two reflectors are used for imaging optics. If the second reflector is removed, the parabolic shape of the primary mirror will focus all the incident beams parallel to the optical axis, functioning as a concentrator.

lower energy but the larger mean free path allows for more attenuation with the higher energy photons (Henke et al., 1993). However, there are ease of fabrication, stability, and cost considerations as well. For these reasons, gold (atomic number $Z=79$) is a popular medium for surface replication (Soong et al., 1995).

Several X-ray missions have used the Wolter I design. Examples of these are Chandra, NuSTAR, XMM-Newton, Suzaku, and Hitomi (Petre, 2010). These missions had goals which called for different methods of fabricating the optics; there is not a one-size-fits-all recipe for X-ray optics.

1.3 The International Space Station Mission: NICER

NICER is a NASA Mission of Opportunity (MO) selected to study the variability of X-ray emission of neutron stars via an instrument payload on the ISS.

This mission was selected for concept studies in September 2011 (Brown, 2011) and later officially accepted for development in April 2013 (Harrington, 2013). NICER is scheduled for launch on May 14, 2017 (NASA, NICER, 2017).

Neutron stars are end products of the evolution of massive main-sequence stars formed from the gravitational collapse after a supernova. They are highly massive ($\sim 1.4 M_{\odot}$ (Chandrasekhar, 1931)) yet quite small (~ 10 km radius); imagine the sun crushed down to the size of New York City. With such a high density (higher than the nuclear density which is $\sim 2.3 \times 10^{17}$ km/m²) and extreme conditions (Lai et al., 2009), the structure and interior composition of these objects are not currently known or understood, only theorized through models (Arzoumanian et al., 2009). Pulsars are neutron stars with high magnetic fields ($\sim 10^{12}$ G) which facilitate the production of X-ray emission from the magnetic poles that can be observed and modeled. If a neutron star with a strong magnetic field rotates and its magnetic dipole axis is tilted with respect to its rotational axis, pulsations can be observed (“lighthouse effect”, “pulsar”).⁵ A pulsar may be either isolated or a component of a binary system. Isolated pulsars, also known as rotation-powered pulsars, are powered through the loss of rotational kinetic energy. In binaries, the neutron star accretes matter from the companion object and converts the potential energy to radiation producing X-rays.

There is a long standing need to further constrain the equation of state (EOS) of neutron stars. An EOS describes the relationship between two or more variables

⁵I may interchange the words ‘neutron star’ and ‘pulsar’ throughout this thesis since one is a category of the other and NICER is concerned with studying pulsars for timing research.

associated with the matter for a class of objects with a given set of physical conditions (i.e. only one EOS for all of neutron stars). This direction of research provides insight on the physics of matter in conditions too extreme to be recreated in a lab (Lai et al., 2009; Paerels et al., 2009). Studying the X-ray emission of neutron stars allows one to put constraints on their masses, i.e., through analyzing relativistic effects apparent in their lightcurves, as well as on their radii, i.e., by applying physical emission models to describe their pulse phase-resolved spectra (Arzoumanian et al., 2009; Lattimer and Prakash, 2001; Psaltis et al., 2014). The timing variability, relativistic effects of rapidly rotating pulsars, and their emission is further expanded upon in Chapter 5.

MSPs are ideal for studying the mass and radii of neutron stars. They are thought to be “recycled” pulsars of old binary systems, i.e. the neutron star has been spun up through accretion over a long time and the magnetic field has been reduced (“buried”) so the spin does not decay as quickly (Bhattacharya, 1995; Ghosh, Pranab, 2007). With pulse periods $\sim 1 - 10$ ms, the relativistic spin velocity introduces complexity into the pulse profiles, i.e. lightcurves (the fastest regular rotation powered pulsar is the Crab with a 34 ms pulse period and regular accretion powered pulsars range from 1 s to a few 100 s) (Psaltis et al., 2014). Energy-dependent lightcurves may allow observers to deduct the mass, radius, inclination of the spin axis relative to the observer, and location of the radiation emission at the surface (Psaltis et al., 2014).

NICER’s payload consists of an X-ray Timing Instrument (XTI) with the abilities to determine neutron star radii more accurately (to $< 5\%$ uncertainty) than

current systems (Gendreau et al., 2012). This is done by observing the lightcurves (Arzoumanian et al., 2009), i.e. the changes in flux over time of pulsars. The XTI optical subsystem consists of 56 highly effective grazing incidence XRCs. Each XRC is co-aligned with a small silicon drift detector (SDD) that has a 2 mm aperture placed on the detector surface (Arzoumanian et al., 2014; Gendreau et al., 2012).

The concentrators are non-imaging optics, meaning that only energy and timing data may be collected about the incident photons, and not spatial information. While high spatial resolution is desirable for resolving extended images, important information can be obtained from the non-spatial photon properties (Fraser, 1989). This allows elimination of the second reflection while still obtaining valuable information. This elimination decreases the number of absorbed and scattered photons and increases the photon efficiency.

Since concentrators [theoretically] focus the incident rays to a single point on the focal plane [when aligned on-axis], the detector’s exposed surface can be made smaller than one needed for resolving an extended image (such as the X-ray CCDs on Suzaku (Koyama et al., 2007)). A small detector area at the focal plane increases the signal to noise ratio by reducing the exposure to background radiation (Prigozhin et al., 2012).

1.3.1 The XRCs

My thesis studies the XRCs, which are an adapted form of the epoxy replicated thin foil optics (Petre, 2010). This form uses a thin aluminum substrate that

is shaped into a reflector’s form and then a reflecting gold surface is applied by transferring gold from a cylindrical glass mandrel to the aluminum with epoxy. The NICER team implemented a few innovations in to the optics design (Gendreau et al., 2012) to create a lighter weight mirror with a higher effective area to mass ratio and increase resolution for a short focal length (with respect to the focal length of many X-ray missions’ optics). To reach these goals, each shell in the mirror (1) has a full shell geometry, (2) has a curved mirror profile to further approximate the Wolter I geometry, and (3) is part of a single reflection concentrator (Arzoumanian et al., 2014; Gendreau et al., 2012). The full shell geometry increases the intrinsic stability of the mirrors requiring less support structure compared to the segmented shells used in other missions (such as ASCA, Suzaku, Hitomi, etc (Petre, 2010)). Using only one reflection further minimizes the total mass thus increasing the effective area to mass ratio. The implementation of a curve into the mirrors’ profile, verses a flat profile, allows incident light to focus at a smaller point. While these changes may seem straight forward as the next step in advancing epoxy replicated optics, significant research went into all details of the design, fabrication techniques, and performance testing methods that had to be invented and/or altered.

1.4 Fabrication Development

I contributed to developing the XRC fabrication method. Since the design was adapted from the thin foil epoxy replicated segmented optics used for mission such as Suzaku and Hitomi, the initial fabrication process was analogous to their

methods. Due to the differences in design, mentioned in Section 1.3.1, many aspects of the fabrication had to be changed. The significant differences affecting techniques in fabrication were: handling a full shell geometry for which the tools did not exist, while integrating a curvature into the mirror profile, and doing this auspiciously for over 1000 substrates. The NICER team needed a reliable and efficient process to be successful in producing 56 flight-ready concentrators by the mission’s deadline.

During the fabrication development phase, I conducted several analytical tests of different fabrication techniques (Centrella et al., 2012). This included testing new equipment, analyzing the feasibility of alternative methods and devising new, efficient ways to perform complicated procedures, developing and building measuring tools to evaluate the techniques, and other miscellaneous technical activities contributing to the larger project. The main contribution I will focus on in this thesis is how I confirmed the validity of the *heat shrink tape forming method* through designing and performing several scientific experiments. During the forming process, a sheet of aluminum is formed into the reflectors backbone, with the parabolic curvature. Heat is used to relieve stress of the aluminum alloy and a forming mandrel is used to guide the aluminum sheet into its new form (Serlemitsos et al., 2010; Soong et al., 1995). The heat shrink tape technique ultimately replaced the previous method of applying vacuum suction to the heated alloy.

The heat shrink tape method is quicker, easier, and cheaper, a major requirement for the amount of reflectors needed to be fabricated. Through my tests, I showed that using heat shrink tape for forming the substrates was successful in completely erasing the substrate memory structures before forming. For those that

passed the PSF/EEF tests (a test of concentration which is expanded upon in the next section), mandrel curvature was copied to the formed foils with an average of $> 95\%$ accuracy (Balsamo et al., 2016).

1.5 Hardware Performance: Calibration, Measurements, and Data Analysis

During XRC development, I continually tested the products for quality performance and have seen great improvement over time. This included measuring substrates on a microscopic level, testing how well optical and X-ray light focused, and determining reflectivity properties of the substrates. The main sign of improvement I saw over time were consistently stronger concentrations of events during X-ray testing. During the final production phase of NICER, in which flight instruments were produced, I helped to calibrate the performance properties of the XRCs by running tests in the lab and conducting the data analysis of these tests.

Due to the concentrators' unique design, the traditional way of calibrating X-ray mirrors at NASA GSFC had to be modified. Before the NICER team was established with seasoned scientists who had calibrated the mirrors of Suzaku and Hitomi, I had already ran X-ray tests⁶ on XACT and NICER concentrators. This experience provided me with knowledge of the XRC particularities and troubleshooting skills at the X-ray beamline testing facility. My advanced understanding of the

⁶I will refer to X-ray tests, or X-ray testing, through out this thesis. The terminology means that a concentrator was tested in an X-ray beamline (Section 4.1) which measured the X-ray photon concentration of the XRC by producing an image on a CCD.

unique features, presented in the results, was used to help the team determine a proper setup for flight performance calibration. I will discuss the setup and experiment parameters further in Chapter 4.

I produced my own code using Python 2.7 (see my codes in Appendix D) to analyze the calibration data. The main functionality of the final version of this code are presented in this thesis along with how it was used to obtain the results of the flight XRC calibration tests. A summary of these results is shown in Figure 1.3 below.

Two quantitative properties were measured for each XRC module produced to determine flight readiness. These are the effective area and encircled energy fraction (EEF). The effective area is a calculation of the optics surface area that reflects incident photons to the focal plane. The EEF is the fraction of the photons incident on the focal plane that are focused within a specific diameter about the center, i.e. ‘encircled.’ The spread can be described by a encircled energy function where the x-axis is diameter, or radius, on the focal plane and the y-axis is the fraction of encircled events at that width. This differs from a point spread function (PSF) in that the PSF is number of events at a specific radius. However, the EEF is proportional to the integral of the PSF. These quantities are defined again in Section 2.2).

Of the 70 modules produced, 64 met the mission requirements for effective area and EEF. The mean effective area of these modules is 48.06 cm^2 with a standard deviation of 2.73 cm^2 . The mean EEF at a 2 mm diameter on the focal plane is 91.3% with a standard deviation of 1.8%.

The encircled energy functions of these 64 modules were used to create a response function (Section 5.5) in the next part of my work, simulating an observation.

1.6 Simulating an Observation

Reducing data from an observation requires knowledge of the concentrators' special features and how to properly use the response function. A response function is a tool used by observational astronomers to convert instrument readout to an observed photon flux measurement (Arnaud, 1996; Arnaud et al., 2013). The result can be used to compare the information to that collected from other instruments and also to deduct physical properties about the target, such as luminosity and the mass accretion rate. NICER's response functions, and other auxiliary parameters, consist of calibration information from the optics, detectors, and pointing system.

An important source to study is the closest MSP known as PSR J0437-4715 (with a $0.2 M_{\odot}$ helium white dwarf companion: Bailyn (1993)). However, it is particularly challenging observation that is highly dependent on the XRC properties. MSP X-ray observations are useful in providing insight on neutron star properties (Freire et al., 2009). A very close source, like PSR J0437-4715 at a distance of about 156 pc, or 509 ly, can maximize flux, which is important in X-ray astronomy where X-ray flux is generally low, making analysis difficult. This pulsar's mass is well known, ($1.76 \pm 0.20 M_{\odot}$ derived by Verbiest et al. (2008)), from radio (long wavelength radiation) measurements and an observation with NICER would further constrain the radius measurement through the instrument's superb timing resolution (the SDD

offers a fast channel with 30 ns shaping time for individual X-ray photon timing) (Gendreau et al., 2012). The radius can be constrained via spectral data throughout different phases of the pulsar’s period (Bogdanov, 2013). The challenging part of the observation is that there is an active galactic nucleus (AGN), called AGN RX J0437-4711, approximately 10 times brighter than the pulsar in the X-ray band located 4.2 arcminutes away (Halpern and Marshall, 1996; Halpern et al., 1996).

During an on-axis observation of the pulsar, the AGN is in the field-of-view of NICER (NICER has a 6 arcminute diameter non-imaging FOV: Gendreau (2016)). Since photons collected in an observation cannot be spatially distinguished, there is no way to absolutely identify the source of each event. This has not been a problem with prior missions that have been able to differentiate the sources spatially.

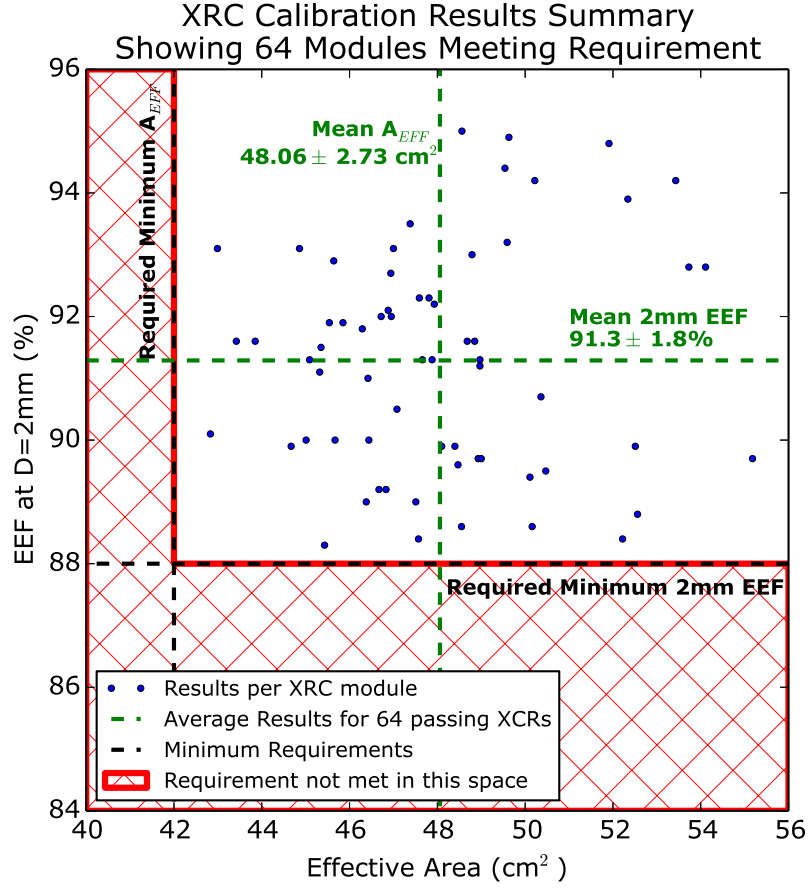
Therefore, my specific experiences with the XRCs were beneficial for designing an observation strategy of this pulsar. I used my knowledge of the properties and functionality of the XRCs and understanding how the optics affect observed flux to create a response function using data I analyzed based on the calibration tests.

An advanced aspect of this modeling is that I applied a state-of-the-art theory of pulsar X-ray emission. Instead of the common blackbody and power law emission, I applied the theory of atmospheric emission that has been discussed in Pavlov and Zavlin (1997), Zavlin and Pavlov (1998), Bogdanov et al. (2007), and references therein. The theory assumes a hydrogen atmosphere on the surface of neutron stars, hydrogen being the lightest element. Radiation that is emitted from the magnetic poles travels through different thicknesses of the atmosphere depending on the initial emission angle. The observed spectra change for each emission angle

since the radiation interacts with different amounts of matter. Therefore, the energy spectrum observed from a specific emission angle depends on the observer’s line of sight relative to the location of the magnetic poles. For pulsars, this means that the observed spectra are dependent upon the rotational phase of the neutron star. The lightcurves are then energy-dependent and the energy spectra are phase-dependent. This theory further described in Section 5.4.

Applying this model was a challenge because it did not exist in XSPEC⁷: An X-Ray Spectral Fitting Package (Arnaud et al., 2013) with a version suitable for pulse phase-resolved modeling. I used prior publications (Beloborodov, 2002; Bogdanov et al., 2007; Poutanen and Gierliński, 2003; Zavlin et al., 1996) to develop an equation for the emission that is dependent on energy and rotational phase; the equation was not clearly defined in any prior texts. Based on this equation (Equation 5.5), I ran simulations of the pulsar’s emission at several phases to develop and implement corresponding table models into XSPEC in order to simulate NICER’s observation of PSR J0437-4715.

⁷XSPEC is a computer program that fits astronomical data to spectral models in the X-ray band.



Histograms of Calibration Results

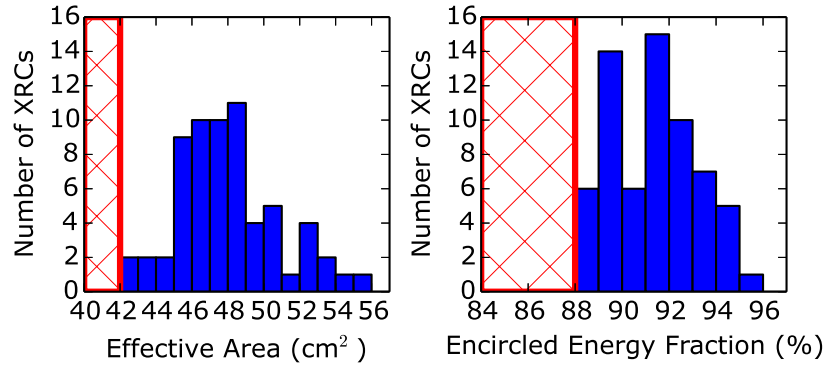


Figure 1.3: Final calibration results of the flight ready modules (ones which passed the requirements) are shown here in the top plot as blue circles. The green dashed lines indicate the average 2 mm fraction ($91.3 \pm 1.8\%$) and effective area ($48.06 \pm 2.73 \text{ cm}^2$) of these modules. In comparison, the mission requirements, shown as black dashed lines, specified the 2 mm fraction to be at least 88% and the effective area to be at least 42 cm^2 .

Chapter 2

The Neutron Star Interior Composition Explorer and the X-ray Concentrators

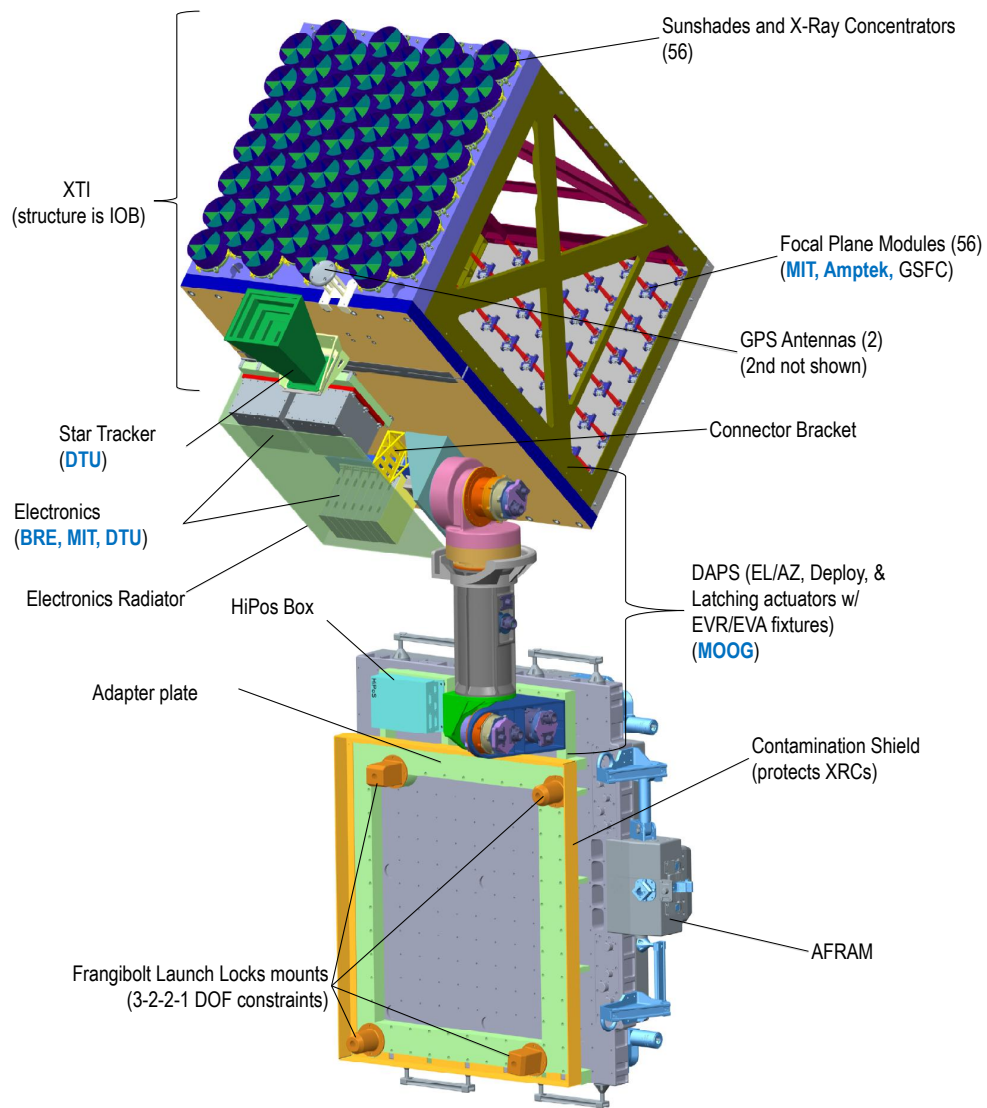


Figure 2.1: Schematic of NICER deployed on the space station (Kenyon, 2013).

NICER (Gendreau et al., 2012), an International Space Station mission, launched June 3, 2017 via SpaceX, will study the extreme physics associated with neutron stars. Its X-ray Timing Instrument (XTI), shown in Figure 2.1, will provide improved timing data to measure neutron star radii more accurately than most current instruments. With an absolute time-tagging resolution of 300 ns, it is $\sim 25\times$ better than RXTE and $\sim 100 - 1000\times$ better than XMM-Newton (Gendreau, 2016).⁸ The peak effective collecting area for target objects, specifically PSR J0437-4715, must be at least 1333 cm² in order to collect enough photons for sensitive light-curve modeling with NICER data (Arzoumanian et al., 2014). Pulsar PSR J0437-4715 is specifically important to observe because it is the closest MSP, and thus relatively bright. However, in order to achieve that sensitivity, it was necessary to require the optics to have a sharp point spread function of at least 88% of photons focusing within the 2 mm aperture on the focal plane (Gendreau et al., 2012). The combination of these requirements led to the unique design of the singly reflecting concentrators, rather than using doubly reflecting imaging optics. The XTI optical subsystem consists of 56 grazing incidence XRCs with high effective area to mass ratios (Gendreau et al., 2012). A single module is shown in Figure 2.2 with 24 nested reflectors ranging from approximately 3-10 cm in diameter. Each XRC is co-aligned with a small silicon drift detector (SDD) with a 2 mm diameter aperture located at the focal plane, 1085 mm from the XRC center. Thus, the high efficiency NICER’s XRCs, along with the small detector aperture, help to minimize

⁸Also from Gendreau (2016): NICER’s sensitivity is 3×10^{-14} ergs/s/cm⁻² (0.5-10 keV), which makes it better than RXTE’s by $\sim 30\times$ and XMM-Newton by $\sim 4\times$.

background radiation allowing for a relatively high signal to noise readout.



Figure 2.2: Every NICER (Gendreau et al., 2012) XRC module consists of 24 nested aluminum foil shells with epoxy replicated gold reflecting surfaces. A single XRC is approximately 10 cm in diameter and in height. The inner most reflector has a radius of 15.11 mm and the outer most has a radius of 51.20 mm.

The XRCs include many unique features that are further described in Section 2.1. Some of these features created engineering challenges during fabrication that ultimately led to the development of the shrink tape heat-forming method for the reflectors' aluminum substrates. The effects the features had on fabrication are expanded upon in Section 3.2.1.

2.1 Design and Geometry of the X-ray Concentrators

The unique characteristics of the XRCs set the concentrators apart from other thin-foil, epoxy replicated, X-ray mirror reflectors. The feasibility for fabricating XRCs with these features was first tested during XACT’s concept study phase when three concentrators were built, each of which containing 66 nested shells (Balsamo et al., 2012; Gendreau et al., 2012). The unique characteristics were adopted and the design optimized to meet the requirements of NICER. Table 2.1 summarizes some key features of the NICER XRCs and XTI.

Table 2.1: Key features and specifications about the NICER X-ray Timing Instrument and the X-ray Concentrators (Arzoumanian et al., 2014; Gendreau et al., 2012).

NICER XTI and XRC Properties

Property	Description
Number of Shells per XRC	24
Number of XRCs	56
Total Number of Reflectors	1344
Reflector Substrate Material	152 μm or 0.006 in thick aluminum
Reflecting Surface	gold
Shell Radius Range	15.11–51.20 mm
Focal Length	1085 mm
Minimum EEf at 2 mm	88%
Effective Area at 1.5 keV (estimated)	$>2000 \text{ cm}^2$
Effective Area at 6 keV (estimated)	600 cm^2
X-ray Energy Band	0.2-12 keV
Detector	SDD with 2 mm diameter aperture
Field-of-View	30 arcmin^2

Even though the NICER XRCs were produced as singly reflecting contractors, a second reflection could be incorporated with this technology if desired. Photons are lost through absorption and scattering with every reflection and interaction with

matter. Decreasing the number of reflections results in more efficient total photon throughput. Photon number conservation is crucial for typical NICER sources, such as those similar to PSR J0437-4715, which have a photon flux of ≥ 0.5 cts/ks/cm² (within 0.2–2.0 keV, this corresponds to an energy flux of $\sim 1 \times 10^{-13}$ ergs/s/cm²), in order to complete the observation in < 1.5 Ms (Arzoumanian et al., 2014).

Another unique feature is that each nested shell was designed as a single reflector made from a single sheet of aluminum, instead of the traditional multiple segments commonly constructed at NASA GSFC, used for many thin-foil epoxy replication X-ray mirrors (Petre, 2010, and references therein). (Each full, 360° shell, or single reflecting shell, is made from a practically rectangular sheet of aluminum, arcuated with a large radius of curvature, and is not to be confused with the electro-formed replicated optics discussed by Romaine et al. (2013).) Not only are there fewer reflectors to fabricate, but the need to properly align multiple segments is eliminated. Mechanical integrity is conserved with the 360° shells because the radius of each shell is maintained naturally, when assembled into the housing.

A parabolic curvature was added to the reflectors’ axial profile design as a way to decrease the image blurring intrinsic to traditional conical optics. A parabolic mirror focuses on-axis incident light to a single point, where as an approximation to the parabola would result in an extended spot on the focal plane. The conical geometry, introduced by Serlemitsos (1988), was originally used because it is easy to construct (Serlemitsos and Soong, 1996). Implementing a precise geometry, like a curvature, was predicted to pose as an engineering challenge. It was previously stated that a precise axial geometry would be difficult, or even impossible, to repro-

duce (Serlemitsos and Soong, 1996). Yet the NICER team⁹ was able to successfully fabricate concentrators with a PSF sharper than possible with a conical design (see Section 2.3).

To introduce this curvature into the reflectors, a parabolic profile was implemented into the forming mandrels so that the substrate of each reflector took on the appropriate shape. This required unique forming mandrels, each with a slightly different parabolic shape, described below, to be made for each reflector size. Assuming the substrates' shape is conserved during epoxy replication¹⁰, the reflectors share a common focal point when nested (Gendreau et al., 2012).

Figure 2.3a shows the extrapolated parabola used for the mathematical profile of each reflector as gray dashed lines. The solid blue lines represent the portion of the parabola that is used to model curvature of each reflector. The center of the concentrator is 1085 mm from the common focal point, making it the focal length. In the figure, the focal point is located at the plot's origin and the vertex of each parabola is at a small negative number (this is difficult to see due to the scale but mentioned here for clarity). Figure 2.3b offers a zoomed version of the XRCs' reflector cross-section.

⁹Those who originally developed the conical foils at GSFC were members, and a great asset, of the NICER team.

¹⁰Epoxy replication is the process that copies the smooth surface of glass onto the inside of an aluminum shell after the substrate is formed. Gold is sputter deposited onto a glass mandrel (the smooth surface) which is transferred to the aluminum shell via epoxy. This process is briefly explained in Section 3.1 and further in Balsamo et al. (2012).

NICER XRC's Nested Parabolic Reflector Cross-Section

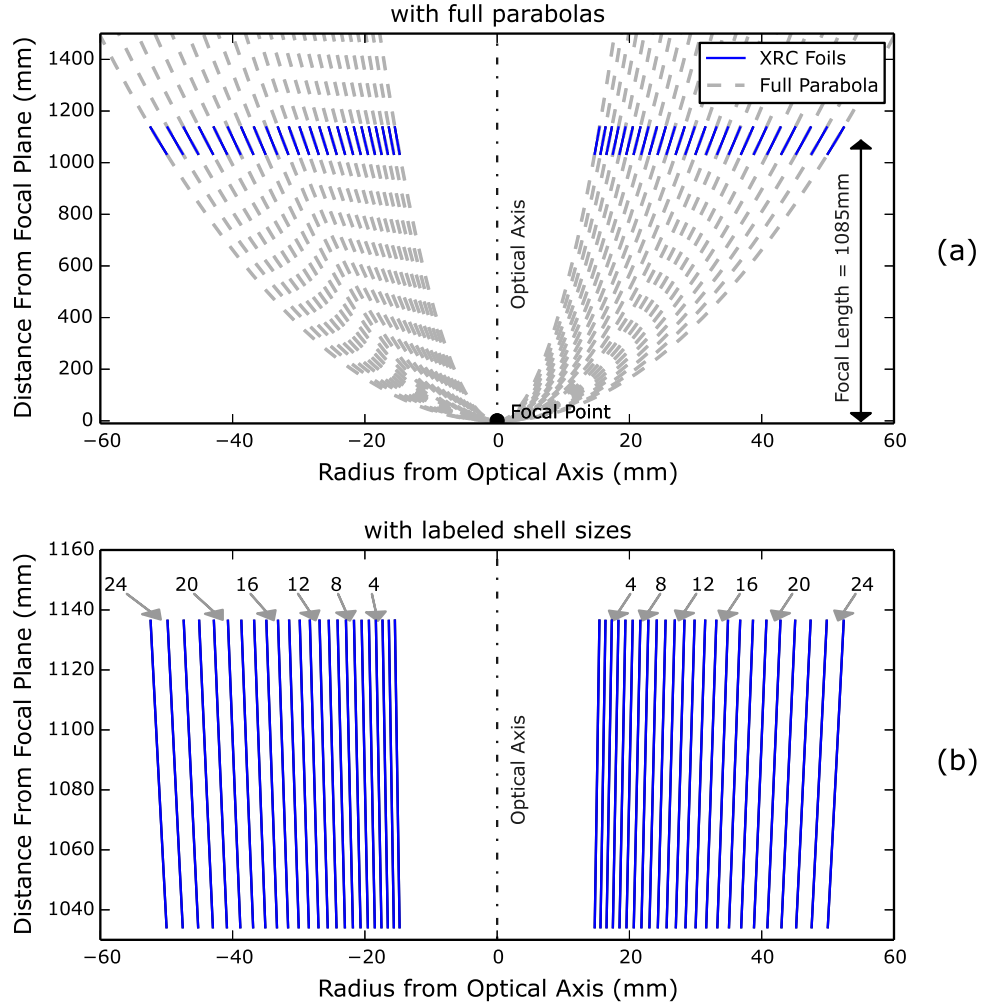


Figure 2.3: Each reflector, or “foil,” in the XRC takes on a slightly different parabolic shape so that the set of nested reflectors will have a common focal point. (a) The gray dashed lines extrapolate from the foil locations to show the full parabolic profiles of all reflectors that were calculated to make the nested reflectors confocal. The blue solid lines show from where on the parabolas the XRC foils take their shape. In this figure, the common focal point is set at the origin and the vertices’s of each parabola are at small negative values on the y-axis. (Due to the figure’s scale, the focal point and vertices’s appear to overlap but they do not. Also, the gray lines appear to have a break in them, but that is due the location of the dashes which have no significance.) (b) A zoomed-in version of the cross-section shows labeled nested shell numbers as a reference.

2.2 Performance Parameters and Mission Requirements

Single module XRC performance was determined based on two features of the analysis: encircled energy function (EEF) and effective area. These values were calculated using spatial data (x and y coordinates) collected on a CCD. Unlike the detector on the NICER instrument, a CCD can collect information about where each event occurred on the detector so that a spread of events, point-spread function (PSF), can be determined.

The EEF is a measure of the energy distribution on the focal plane. It is a fraction of flux within a circle around the center of an image as a function of radius. In practice, the number of photons within some radius is normalized by the total number of photons within some large radius. A general formula for EEF as a function of radius r is shown in Equation 2.1. $N(r_0)$ is the number of photons between radii r_0 and $r_0 + r_{\delta r}$ and the numerator is the sum of the number of photons from a radius of $r_0 = 0$ to the variable radius r . The denominator is the same sum ranging from $r_0 : 0 \rightarrow r_{max}$ where r_{max} is a large radius at which no photons reflected.

$$EEF(r) = \frac{\sum_{r_0=0}^r N(r_0)}{\sum_{r_0=0}^{r_{max}} N(r_0)} \quad (2.1)$$

The optics efficiency is a measure of how well it transmit photons. Efficiency is a ratio of the number of photons (or photon flux) that reached the detector (or focal plane) to the number of incident photons. That fraction is multiplied by the XRC aperture area to get an effective area. In Equation 2.2, the effective area A_{eff} is measured in practice by multiplying the XRC aperture area A_{ap} by the flux at

the focal plane $flux_{\text{focal}}$ divided by the incident flux on the XRC $flux_{\text{incident}}$.

$$A_{\text{eff}} = A_{\text{ap}} \frac{flux_{\text{focal}}}{flux_{\text{incident}}} \quad (2.2)$$

The requirement, determined by the NICER team, was to achieve a total effective area of 42 cm^2 at 1.5 keV for each XRC module with at least 88% encircled energy at the 2 mm diameter about the image center (also referred to as ‘aperture’ because it represents the detector aperture size used for NICER). This is equivalent to at least 37 cm^2 effective area at the 2 mm aperture.

2.3 Enhanced Performance through Curved Axial Profile Implementation

Implementing a curved axial profile into the aluminum substrate was the next logical step for improving the angular resolution performance of thin foil, epoxy-replicated, X-ray optics. The flat axial profile construction has been used for a couple of decades and the fabrication techniques have been approaching maximum efficiency. Other than the optics’ profile, many of the other features contributing to poor angular resolution have been greatly improved over the years through techniques for accurate substrate shaping, surface smoothing and reflector alignment (Petre, 2010). In fact, ASTRO-H has already shown improvements over Suzaku in angular resolution performance by a factor of two (Soong et al., 2014). In order to make optics with shorter focal length, hence take up less room on a payload and de-

crease costs, and improve spatial resolution, a curved profile was the quickest way to provide significant technological advancement. In the following, I demonstrate the significance a curved profile implementation has on NICER over a conical geometry.

The axial profile for a conical mirror follows the tangent to the curve described by the Wolter type I geometry, making a flat profile reflector. Previous work (Serlemitsos and Soong, 1996) reported that the intrinsic blur resulting from a flat profile reflector was not a significant factor in PSF degradation compared to other mirror features. However, those mirrors did not have a focal length as short as NICER's. If a flat profile were used with NICER's 1085 mm focal length, the resultant intrinsic blur would impact significantly the photon concentration.

I modeled the effects of a NICER XRC if it were had a flat profile, via ray-tracing (the ray-tracing program is explained further in Section 5.5.1), and compared the resultant image to that of an actual X-ray measurement with a flight quality XRC. For reference, a parabolic profile XRC module was also modeled using ray-tracing. The X-ray measurement was performed at NASA GSFC's 100 m beamline located at the Goddard Geophysical and Astronomical Observatory (Goddard Geophysical and Astronomical Observatory, 2014). Photons were generated at 1.5 keV and the detector was a liquid nitrogen cooled X-ray charge-coupled device (CCD). A ray-trace code, ConSim, was adapted, by another member of the NICER team¹¹, from the ASCA ray-trace code (Tsusaka et al., 1995) to model the NICER optics. My modifications to this code are expanded upon in Chapter 5 and Table 5.6. Some of the ConSim input options of importance for the work of this section, include se-

¹¹credited to Lalit Jalota

lecting photon energy, choosing a flat (conical) or parabolic profile for the reflectors, defining whether or not scattering was considered at the reflecting surface, and selecting photon flux. The X-ray measurement and ray-tracing both provided output data as event files, showing the location of each photon on the focal plane.

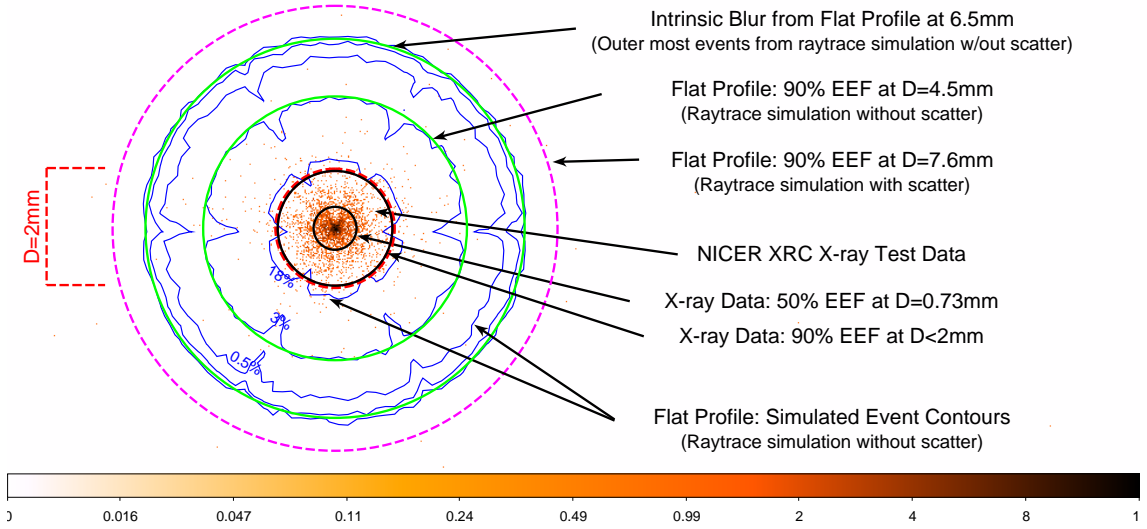


Figure 2.4: Theoretical simulations and data measurements of a NICER XRC X-ray image are compared. A 2D histogram was created in SAOImage DS9 Version 7.2 (Joye and Mandel, 2003) using events (see colorbar) from an X-ray measurement of a flight quality NICER XRC module. Overlaid with these events are the contours (blue lines) of a ray-traced NICER module assuming a flat profile and considering scattering at the reflecting surfaces. The inner red dashed circle represents the 2mm diameter aperture for which NICER requires the enclosure of 88% of total energy throughput. The solid circles, from smallest to largest, represent the X-ray data half power diameter (HPD) (smaller black circle), X-ray data 90% EE (larger black circle), the 90% EE of the ray-traced simulation without scattering (smaller green circle), and total intrinsic blur which would result from a theoretically perfect (without scattering) flat profile NICER XRC module (larger green circle). The outer most magenta dashed circle is the 90% EE from the ray-traced simulation with scattering.

Two ray-trace simulations were conducted with the flat profile model. One considered the effects of scattering at the reflectors' surfaces and the other did not.

Figure 2.4 was produced using the data visualization application SAOImage DS9 Version 7.2 (Joye and Mandel, 2003). Shown here are the events from a XRC X-

ray measurement (scaled with the colorbar) overlaid with contours (blue lines) from the ray-traced events of the flat profile without surface scattering considered. The contours were simulated because a physical flat profiled XRC was not fabricated for testing. Also shown are the EEf calculations for both the XRC X-ray data (black circles) and the flat profile ray-trace simulations (green circles). The inner most red dashed circle represents the 2 mm diameter aperture used for NICER's detector for which 88% of photon throughput is required to be encircled. The intrinsic blur for a NICER sized conical concentrator is illustrated by ray-tracing a flat profile without scattering (magenta dashed circle). For this simulation (no scattering considered), the events extended out to a 6.5 mm diameter (D) with 90% encircled energy at D=4.5 mm (larger green circle) and 48.2% encircled energy at D=2.0 mm (smaller green circle). When scattering was considered, the 90% EEf was at D=7.6 mm (intrinsic blur, magenta dashed circle) and 47.2% encircled energy at D=2.0 mm (red dashed circle). On the other hand, the X-ray data had 90.5% encircled events at D=2 mm (larger black) and a half power diameter (HPD)¹² at 0.73 mm (smaller black circle). This 90.5% EEf at 2 mm found during performance testing is more than 190% better than the 47.2% EEf calculated from the flat profile, with scattering, ray-trace data. In conclusion, a curved profile was necessary for the NICER optics in order to capture the maximum number of photons within a 2 mm diameter at the focal plane, i.e. the detector aperture.

Figure 2.5 shows the overlaid EEfs of all the simulations and X-ray measurements. The X-ray measurement matched well with the ray-traced parabolic profile,

¹²The HPD is the diameter at which 50% of the energy is encircled.

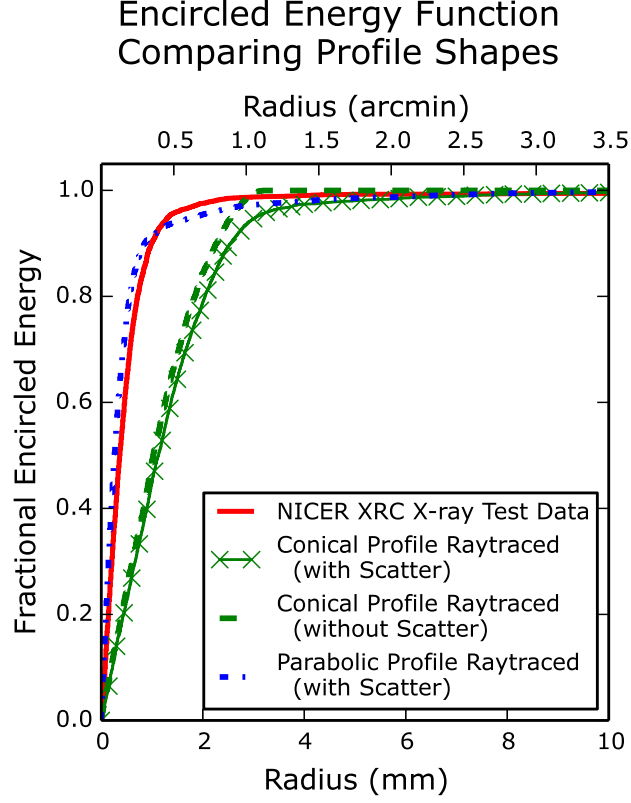


Figure 2.5: The EEFs of measured X-ray data of a NICER XRC module compared to the EEFs of ray traced data. The X-ray data are denoted by the solid line and are similar to that of the ray traced data using a simulated parabolic profile, denoted by the dash-dotted line. This simulation considered an estimated scattering function (see Section 5.5.1 for details on the scattering function). Without scattering, the theoretical EEF would go to 1 at $r=0$. The dashed line and the connected x's are the EEFs of ray traced data which used a simulated conical (flat) profile. The dashed line shows the data that did not consider scattering function where the connected x's show the data that did.

with scattering considered in the simulation, when the EEFs are compared. Both EEFs produced from the ray-tracing of flat profiles show a much worse resolution. Therefore, a much sharper image is produced when curved profiles are implemented into XRC reflector substrates.

Chapter 3

Development and Advancements of the XRC Fabrication

The majority of this chapter was published in the peer reviewed journal, SPIE's Journal of Astronomical Telescopes, Instruments, and Systems (JATIS). The citation can be found at Balsamo et al. (2016).

3.1 Fabrication Overview

Thin foil X-ray mirror fabrication, commonly used at NASA GSFC, has three major processes: foil substrate forming, reflecting surface replication (Soong et al., 1995), and assembly of the reflectors in a holding structure for nesting (see Figure 2.2 for a picture of a completed NICER concentrator module). The portion of the fabrication for which a technique was developed, as a joint effort between members of the NICER team (including myself), is part of the process for shaping the foil substrate backbone of each nested reflector. The substrate is made of an aluminum alloy foil ($152\text{ }\mu\text{m}$, or 0.006 in, thick for NICER) and was shaped into proper form through a process known as heat-forming.

The heat-forming method of shaping foils was introduced in the early 1990s as an improvement to only rolling the foils into shape (Serlemitsos et al., 2010). This allowed the substrates to have a more precise geometry and fit naturally in the holding structure. Aluminum forming mandrels of various sizes were used to shape

the reflectors of different radii. The mandrels’ aluminum alloy was selected such that the coefficient of thermal expansion (CTE) was similar to that of the foil substrate. A foil is secured around the forming mandrel and softens during the heating phase in order to take the shape of the mandrel. The original method of securing the foil used a vacuum between the foil and mandrel. This original method and resultant challenges faced are further discussed in Section 3.2.1.

The surface replication process involves replicating the smooth surface of a glass Pyrex mandrel to the foil substrate’s rough surface (Soong et al., 1995). A schematic of this process is shown below by Figure 3.1 and also in Appendix B, which is a poster I created for the NICER site visit¹³ outlining the XRC fabrication process (Figure B.1). The glass mandrel is sputter coated with gold (Figure 3.1 left) and then sprayed with an even coating of epoxy (Figure 3.1 middle). A formed foil is then placed around the epoxy covered gold mandrel (Figure 3.1 right) and subjected to a constant pressure. The epoxy aids in smoothing the substrate’s surface and transferring the gold to the aluminum to become the reflecting surface. Being a fluid, epoxy fills in the aluminum’s micro-rough surface, allowing the new gold surface to be smooth like the glass mandrel’s surface, hence “replication” of the glass surface. More details of the epoxy replication process, and the rest of the fabrication, can be found in Serlemitsos et al. (2010), Serlemitsos and Soong (1996), and Serlemitsos (1988) as well as my SPIE conference proceeding Balsamo et al. (2012).

¹³A site visit is an event when NASA representatives come to learn about a project for which a proposal was submitted in order to determine if the project should receive funding and be come accepted as a mission.

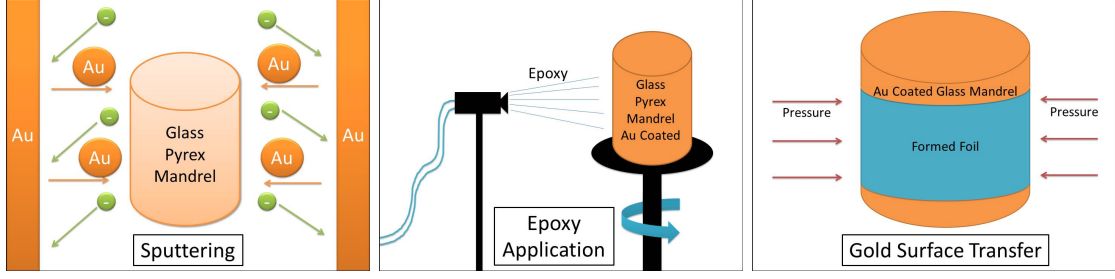


Figure 3.1: Epoxy Replication Process Schematic. (left) A glass Pyrex mandrel is placed in a chamber where gold is applied onto the mandrel’s exterior surface through sputtering deposition. (middle) The gold covered mandrel is sprayed with an even coat of epoxy. (right) A formed foil is placed around the epoxy covered gold mandrel and subjected to a constant pressure. Once cured, the foil is removed from the glass mandrel and epoxy has transfered the gold to the foil’s interior surface.

3.2 Heat-Forming Aluminum Substrates

The core development reported here is the innovative heat-forming method using heat shrink tape which compressed the foil substrate during the heating process to a precision cut forming mandrel, forcing the foil to take the shape of that mandrel. The idea for the shrink tape heat-forming technique was originally tested during the development of the XACT (Gendreau et al., 2012) sounding rocket XRCs as described in Balsamo et al. (2012). XACT predated NICER in XRC fabrication and served as a testbed for implementing the concentrators’ unique features and design elements described in Section 2.1. The original forming method, using vacuum, was unreliable in properly forming the substrates, which led to testing new techniques. Figure 5 from Balsamo et al. (2012) shows the results from both forming methods used during the XACT fabrication. Further described in Section 3.2.1 are the original heat-forming method and the development of the shrink tape technique.

Since then, this technique has been used to fabricate all of NICER’s (Gendreau

et al., 2012) XRCs.¹⁴ Using heat shrink tape to assist in the heat forming process of the aluminum substrate was shown to be accurate, reliable, and repeatable. This can be shown through measurements of the substrate shape implemented after the forming process of several NICER foil shells as demonstrated in Section 3.2.3. In order to determine the accuracy of the shrink tape heat-forming method, the axial profile of several formed foils was measured and compared to the shape of the forming mandrels. The foil and mandrel profiles were scanned with a Laser Scanner Micrometer (LSM) that measured surface fluctuations within a micron.

3.2.1 About the Heat Shrink Tape Forming Technique

For producing 56 XRCs, with a total of 1344 flight quality reflectors, the production rate was a top priority. A reliable system was needed to consistently produce foils with good image quality.

The original substrate heat forming method used in the beginning of XACT’s XRC development applied a vacuum force, less than 1 atm, between the forming mandrel and foil during the whole heating-cooling phase. Though this was a proven technique used for Suzaku and Astro-H mirror fabrication, it resulted in some technical challenges for the XRCs, such as achieving and maintaining an appropriate low pressure between the foil and mandrel for several hours. Achieving a complete seal around the foil while on the mandrel was necessary for successful production but

¹⁴Routine fabrication of flight hardware was performed by hired technicians. I fabricated some substrates during the development process to construct engineering test units (ETUs). I also assisted in the assembly of early flight modules. However, I played a larger roll in all fabrication steps of the previous generation XRCs that were planned to be used for XACT sounding rocket (Gendreau et al., 2012).

very difficult to achieve for a full shell reflector. With the vacuum forming method, XRC foil production rate was dependent upon the limited resources. Each mandrel required a separate vacuum connection while in an oven. Production was limited to the number of vacuum connections available and often the seal leaked, neither of which were concerns for the shrink tape method. Foils were often formed improperly during the XACT concept study phase and resulted in poor image quality (i.e. Figure 5 in Balsamo et al. (2012) as mentioned in Section 3.1). This method was unreliable for XRC production.

While developing a new forming method, the original concept was kept such that the foil was heated to a temperature at which the shape can be conformed by securing it tightly to a properly shaped mandrel. The change involved how the foil was to be pressed tightly to the forming mandrel. Wrapping the foil with heat shrink tape while on the mandrel provided a simple and viable solution.

The NICER XRC reflector substrates were made of aluminum alloy foil with a $152\,\mu\text{m}$, or 0.006 in, thickness. Aluminum sheets were prepared by a precision machine shop to the correct curvature and length, which saved time and eliminated error by manually cutting foils to size in the lab. These sheets were mechanically rolled into a radius approximating the general shape of the final substrate. They were pre-rolled to provide less tension on the heat shrink tape while forming. The pre-rolled foils were then placed around the forming mandrel in stacks of two and aligned to the bottom of the mandrel to ensure alignment of the optical axis. Stacking the foils by two increased the production rate and did not negatively affect the performance outcome. To prevent contamination from the heat shrink tape on the

foil substrates, a barrier foil was placed around the outer foil as protection.

There were 24 different forming mandrels, each with a unique parabolic shape to make reflectors of all the different radii whose profiles' were described in Section 2.1 and by Figure 2.3 . These mandrels were also produced by a precision machine shop with shapes that were described through specifying the mandrel radii along the axial profile, as described in Equation 3.1.

$$D = 2 \left(\sqrt{v_i^2 + 2v_i(h + f)} + t_{\text{epoxy}} \right) \quad (3.1)$$

The mandrel diameter D is dependent upon the axial height h . The coefficient v_i is different for each shell and is the product of the incident angle of an on-axis light ray, θ , and the radius at the bottom of the foil, r_b , (closest to the focal plane), i.e. $v_i = \theta r_b$. Variable f is the focal length and t_{epoxy} is the epoxy thickness applied during the replication process. This was the first time a separate forming mandrel was used for each reflector size in fabricating epoxy-replicated aluminum foil X-ray optics (Petre, 2010). I measured the mandrels in the lab and present the results in Section 3.2.2.

The foils' 10 cm length allowed the use of 15.24 cm (6 in) wide by 50.8 μm (0.002 in) thick heat shrink tape which was wrapped around the foil-mandrel setup 3 times and secured with Kapton®Tape¹⁵ (Dupont, 2017). The bottom of the heat shrink tape was cut to the right curvature in order to smoothly wrap around the mandrel so that even pressure would be rendered. Once prepared, several mandrels

¹⁵Kapton®Tape is a polyimide film silicone adhesive that holds up to high temperatures and is made by Dupont™.

would be placed in the same oven at 200°C for 9 hours; this temperature is required to relieve residual stress in the substrate, and even though the temperature is higher than recommended for the heat shrink tape (Table 3.1), the products are not harmed. The fact that many mandrels could be used to form foils at the same time was a significant achievement as a direct result of the shrink tape forming technique development. The production rate was no longer limited to the number of vacuum connections, the previous most resource limiting feature. At this temperature, the tape shrinks by 20% and relieves stress of the aluminum alloy to shape it into a new form. The setup is shown in Figure 3.2 for a size 20 NICER foil. The heat-shrink

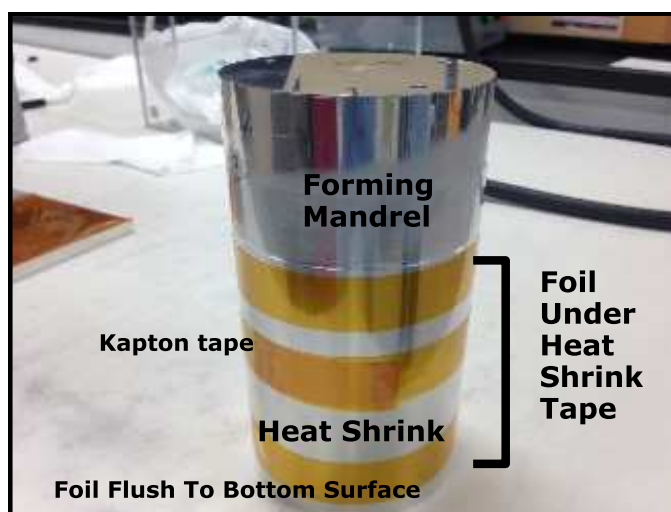


Figure 3.2: Heat shrink tape forming setup of a size 20 NICER foil. The precision cut foil is wrapped around the forming mandrel and compressed by heat shrink tape which is secured by Kapton Tape. The foil is flush with the bottom of the mandrel.

tape used was Dunstone Company Inc.’s HI-SHRINK POLYESTER tape (Dunstone Company Inc., 2013). The specifications, as listed on Dunstone Company Inc.’s website, are shown in Table 3.1.

In order to determine the accuracy of the shrink tape heat-forming method, I

Table 3.1: Specifications of Dunstone Company Inc.’s HI-SHRINK POLYESTER as seen on the company’s website (Dunstone Company Inc., 2013).

Shrink Tape Properties	
Property	Specification
Film	Polyester
Shrink Percentage	20%
Shrink Force (lbs)	7.0
Shrink Initiation	65°C
Temp @ Max Shrink Force	150°C
Max Recommended Usage Temperature	180°C
Melt Point	251°C

measured the axial profiles of several formed foils and compared them to the shape of the forming mandrels.¹⁶ The foil and mandrel profiles were scanned with a Laser Scanner Micrometer (LSM) that measured surface fluctuations within a micron.

3.2.2 Forming Mandrels Used as Shaping mold

To determine accuracy, I compared the mandrels’ shape to the design specifications, upon being manufactured by a precision machine shop. The curvature depth, referred to as bulge size, was determined from the mandrel profile scan data by finding the height of the peak of the curvature. Figure 3.3 shows the results of bulge size from the scan data in comparison to the original mandrel design.

Each of the 24 mandrels were scanned at four locations, 90° apart, about the optical axis, totaling 96 mandrel scans. The scans for each mandrel were averaged together and used to compare with the foil profile scan data in Section 3.2.3. The

¹⁶In addition to my measurements, I used the measured axial profile data collected by another NICER team member to increase statistical significance. I conducted measurements primarily during fabrication development and occasionally later on during flight production for supplemental research. The technician performed routine scans during flight fabrication. However, I conducted 100% of the profile measurements for the forming mandrels.

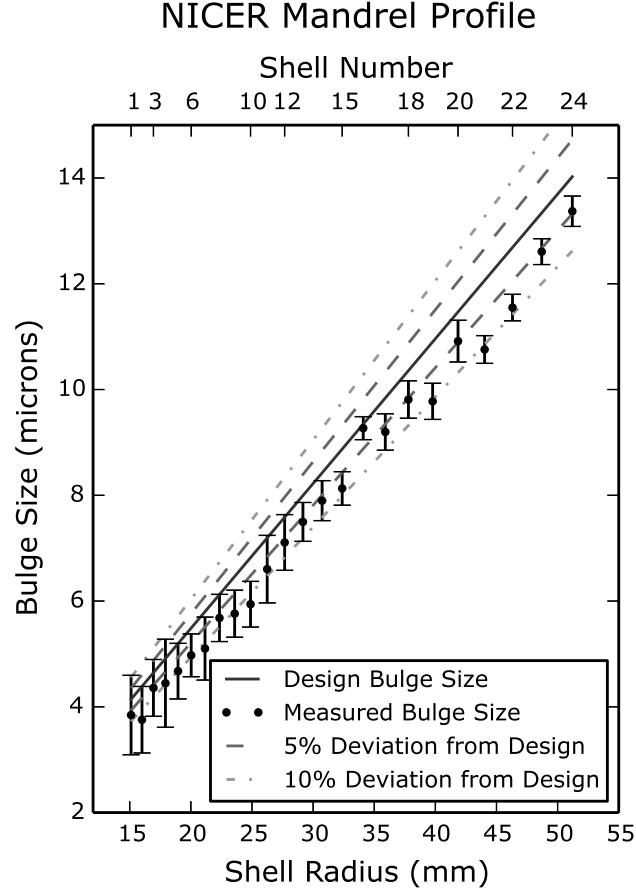


Figure 3.3: Design specifications for the bulge on the aluminum forming mandrels for NICER in comparison to the measured bulges on the manufactured forming mandrels. There is an unexplained systematic deviation between the design and measured bulge sizes that was not statistically significant enough for further investigation.

error bars on the bulge sizes plotted in Figure 3.3 were calculated using the mean of the standard deviation from the four scans per mandrel along with the system uncertainty due to the discrete values given by the scanner, about $0.18\,\mu\text{m}$. The mandrels were acceptably manufactured to within 10% of the design.

3.2.3 Results Based on Foil Axial profile Measurements

The axial profile scans of several NICER foils, a total of 519 scans, were measured after the heat forming process and compared to the previously measured mandrel profiles and the original design for forming accuracy. See Table 3.2 for the number of scans of each foil size. The raw scan data were smoothed using a Savitzky-Golay filter (Savitzky and Golay, 1964). This filter uses convolution and least squares fitting. The foil scans were averaged over all the samples, of same size foil, and the analysis of this resultant profile is described later, after the Figure 3.4 description. Figure 3.4 shows the profile scan results for every size foil in a NICER XRC. The design specification is plotted for comparison with the measured mandrel profile. The average foil profile calculated from the set of scans is overlaid with the mandrel profile plots.

The forming accuracy was quantified by using the root-mean-square (RMS) value of the residual profile between the averaged mandrel profile and the averaged foil profile of each size reflector. The RMS uncertainty was determined by taking the standard deviation of RMS values calculated by comparing each individual foil to it's corresponding mandrel profile. Using the RMS was chosen as an appropriate metric because the shapes of each individual foil were not necessarily smooth curves, i.e. some profiles had large scale structures such that using maximum deviation as a metric would not have been a good representation of the complete profile. The results are tabulated in Table 3.2. The residual RMS column shows the RMS resulting from the residual between the average foil profile and averaged mandrel

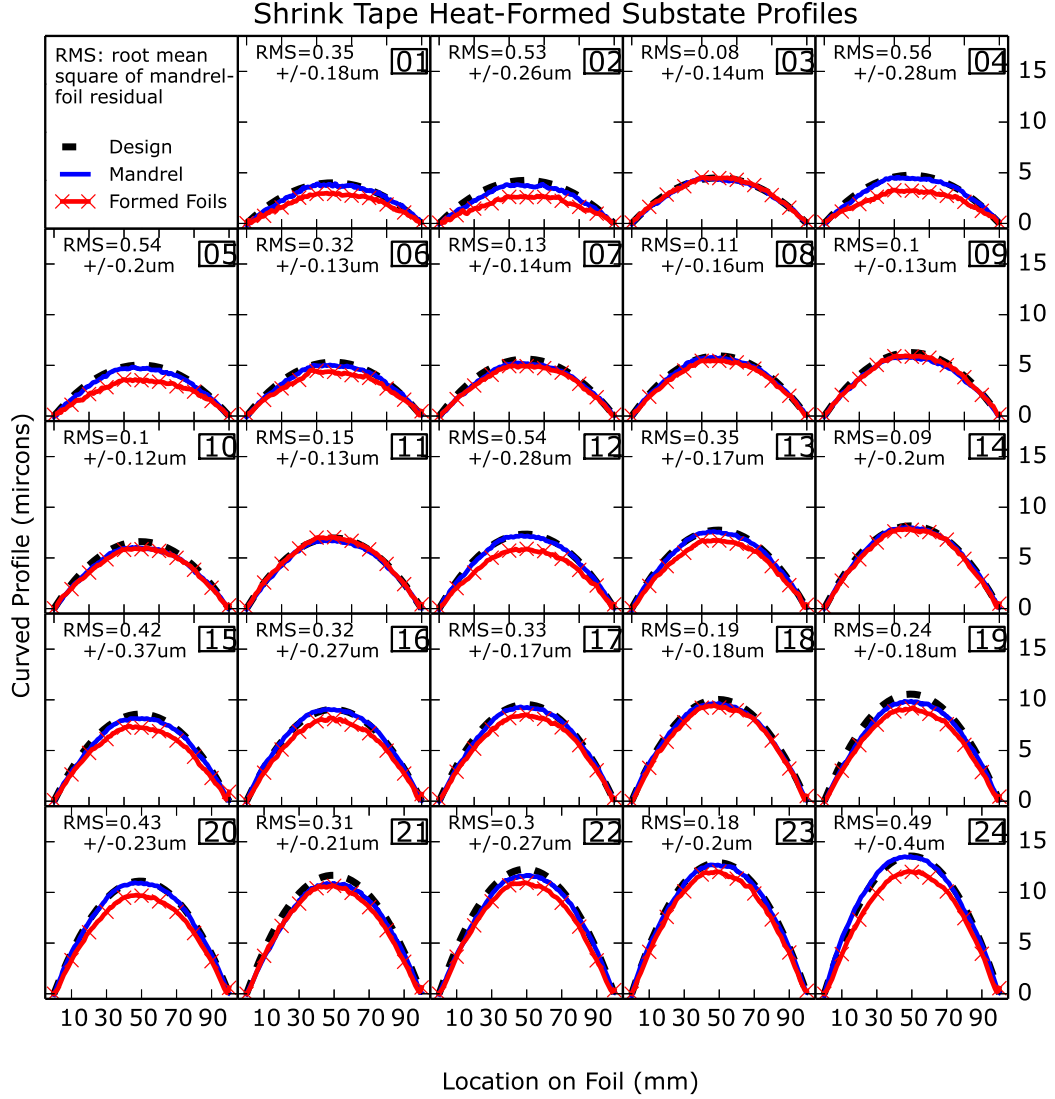


Figure 3.4: The complete results from the profile scans are shown here. The dashed line represents the design specifications for each curve. The solid line is the measured actual on the forming mandrels. The solid line with 'x's is the average profile of the formed substrates using the shrink tape heat-forming method. The RMS values of the difference between the mandrel and formed foil profiles are shown for each shell size.

scan. The maximum deviation from the mandrel's curvature (i.e. difference between the profiles) was less than 15%. On average, the deviation from the curvature was $4.6\% \pm 3.7\%$, or rather copied to within 95.4% of the mandrel curvature. This shows that the shrink tape forming technique does successfully shape the substrate to the

forming mandrel's profile.

If the forming technique were to completely fail, thus maintaining the initial flat profile of the foils, there would be a positive trend between the residual RMS values and foil size (Figure 3.5 blue circles). However, formed foils did not result in a positive trend. This shows that using shrink tape heat forming is completely successful in erasing the memories in the substrates before forming. Figure 3.5 shows the RMS value for the difference in the mandrel and foil profile over the 24 different foil sizes.

Overall Accuracy of Maintaining Profile Shape During Heat Forming Process

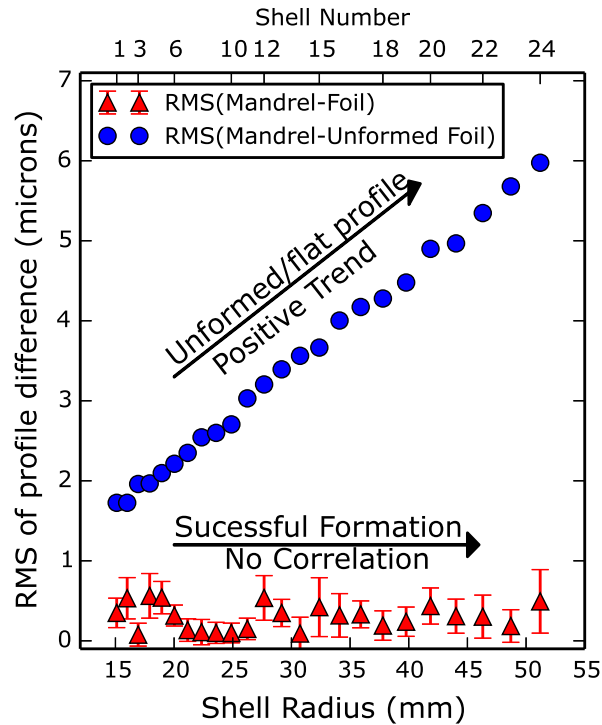


Figure 3.5: The plot shows the overall accuracy of the shrink tape heat-forming technique by comparing data from measured formed foil profiles to a flat profile representing a completely unformed foil. The data in this graph are the RMS value of the difference between the forming mandrel and the foils' profile. The smaller the value, the closer the foil profile is to the actual mandrel profile.

Table 3.2: Results containing the root mean square values calculation from considering the difference between the forming mandrel profile and the average profile measured after heat-forming foil substrates. Column 3 is the RMS of the difference between the average foil profile and average mandrel profile for each size. Column 4 is the percent difference of the average formed foil profile from the mandrel profile. The value is calculated by dividing the RMS value in Column 3 by the measured mandrels' bulge size. The final column reports the number of foils sampled per shell size. The key for the table column names is as follows: RMS(*values*) = root-mean-square of *values*; M = average mandrel profile as calculated from measured scans; F = average foil profile as calculated from measured scans of individual foils.

Nested Foil Position	Shell Radius (mm)	RMS(M-F) (μm)	Percent Deviation From Curvature	Number of Foils Sampled
1	15.11	0.35 ± 0.18	$9.07 \pm 4.78\%$	22
2	16.00	0.53 ± 0.26	$14.13 \pm 6.85\%$	14
3	16.94	0.08 ± 0.14	$1.76 \pm 3.27\%$	29
4	17.92	0.56 ± 0.28	$12.64 \pm 6.27\%$	16
5	18.95	0.54 ± 0.20	$11.57 \pm 4.33\%$	18
6	20.03	0.32 ± 0.13	$6.32 \pm 2.62\%$	18
7	21.16	0.13 ± 0.14	$2.60 \pm 2.74\%$	25
8	22.34	0.11 ± 0.16	$1.91 \pm 2.76\%$	27
9	23.58	0.10 ± 0.13	$1.71 \pm 2.28\%$	23
10	24.88	0.10 ± 0.12	$1.64 \pm 2.06\%$	22
11	26.24	0.15 ± 0.13	$2.22 \pm 2.00\%$	18
12	27.67	0.54 ± 0.28	$7.49 \pm 3.90\%$	29
13	29.17	0.35 ± 0.17	$4.60 \pm 2.25\%$	17
14	30.74	0.09 ± 0.20	$1.14 \pm 2.56\%$	18
15	32.38	0.42 ± 0.37	$5.16 \pm 4.51\%$	27
16	34.10	0.32 ± 0.27	$3.51 \pm 3.01\%$	20
17	35.90	0.33 ± 0.17	$3.57 \pm 1.81\%$	18
18	37.80	0.19 ± 0.18	$1.98 \pm 1.91\%$	24
19	39.78	0.24 ± 0.18	$2.43 \pm 1.86\%$	22
20	41.85	0.43 ± 0.23	$3.98 \pm 2.07\%$	24
21	44.03	0.31 ± 0.21	$2.84 \pm 1.96\%$	23
22	46.31	0.30 ± 0.27	$2.59 \pm 2.31\%$	21
23	48.70	0.18 ± 0.20	$1.46 \pm 1.61\%$	17
24	51.20	0.49 ± 0.40	$3.65 \pm 2.94\%$	27
Average		0.30 ± 0.15	$4.6\% \pm 3.7\%$	

Chapter 4

XRC Ground Calibration, Performance, and Special Features of the Optics' Response

This chapter focuses on the ground calibration of the flight ready XRCs including the X-ray beamline experiment set up, data output and parameters, and my data analysis which led to the selection of the 56 chosen modules used in the final XTI construction.

4.1 Calibration Facilities

The 100 meter beamline at GSFC was used to perform the ground calibration of the NICER XRCs. However, I used the facilities 600 meter beamline, Figure 4.1, during the concept study phases of NICER and XACT. The equipment in both beamlines is similar, i.e. the same type of X-ray CCDs and source generators. The whole beamline and its chambers are under vacuum (about 10^{-5} to 10^{-6} Torr) since X-rays do not transmit in air.

A beamline is a long, enclosed tube with an X-ray source at one end and a chamber for equipment and a detector at the other end. The long tube allows for the X-rays to be mostly parallel by the time they reach the optical chamber, simulating an infinitely far source. A source at a truly infinite distance would allow for the nominal focal length of the designed optic. According to the lens equation, the focal



Figure 4.1: GSFC’s 600 meter beamline at GGAO (Goddard Geophysical and Astronomical Observatory, 2014). *Photo Credit: Erin Balsamo*

length is changed by:

$$\frac{1}{F'} = \frac{1}{S} - \frac{1}{F} \quad (4.1)$$

where F' is the laboratory focal length, S is the distance between the source and the optic, and F is the nominal focal length (Cummings et al., 2004, p. 1036).

4.1.1 X-ray Source

In a typical X-ray generator, current is applied to a filament to produce an electron cloud via thermionic emission (Cummings et al., 2004, pg. 1103). At some distance away, on the order of centimeters, is a heavy metal target. A high voltage power supply provides a potential difference between the filament and the target, accelerating the electrons up to the high voltage setting. The high-energy electrons collide with the target causing characteristic X-rays emission. Figure 4.2 diagrams

this process.

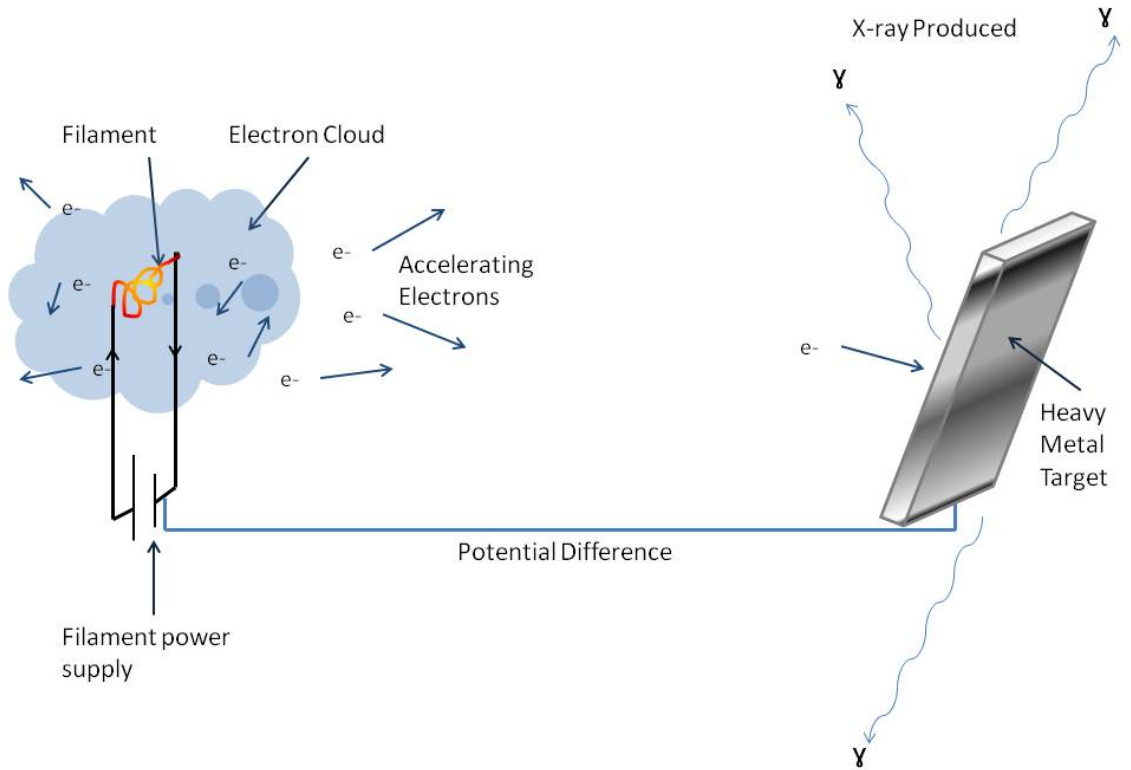


Figure 4.2: Diagram of X-ray source. A current applied across a filament allows electron production. A potential difference, provided by a high voltage power supply, accelerates those electrons to a heavy metal target. X-rays, at energies characteristic to the target material, are produced when the high-energy electrons collide with the metal target. During XRC calibration, I had control of the current (filament power supply), the high voltage (potential difference), and the material of the heavy metal target.

The NICER XRC calibration was performed using Manson® X-ray Source (Manson, 1985) as the X-ray generator. Since NICER’s mission requirements were based on the XRC response at 1.5 keV, we used an aluminum target for the calibration, which produced X-rays at 1.49 keV. During tests, I had control over the current and voltage applied to the generator. The current controlled the intensity of the X-ray beam while the voltage provided a high energy cut-off. In addition, I

fabricated¹⁷ an attenuator (for the 600 m beamline only - there was already one at the 100 m facility) to decrease the intensity further when I required it (i.e. XRC focusing $\sim 1 - 2$ photons per second was needed for quality measurements of the effective area). My attenuator was a 3D-printed apparatus allowing me to mount to it different thicknesses of aluminum foil. It was attached to the beamline through a vacuum sealed flange, just after the X-ray source (< 1 m from it). Example source settings are shown later in Section 4.2.5.

The following figures show the set up and controls of the Manson® X-ray Source at the X-ray generator end of the 600 m beamline (Figure 4.3), a close-up of the Manson® X-ray Source in that set up (Figure 4.4), and the attenuator I designed and installed (Figure 4.5).

¹⁷Another team member (Steven Kenyon) helped with making the 3D printed mount, which is the black plastic object shown to the right in Figure 4.5.

Manson X-ray Source Set Up

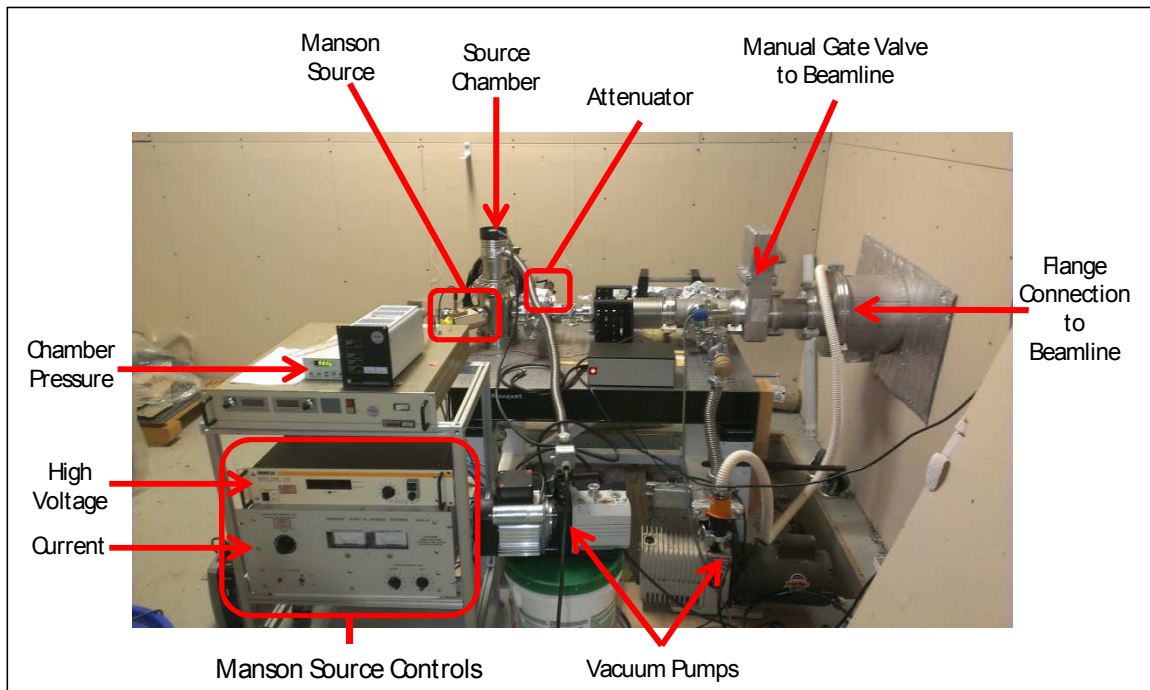


Figure 4.3: Setup of the Manson X-ray source in the "Source Hut" of the 600 m beamline.

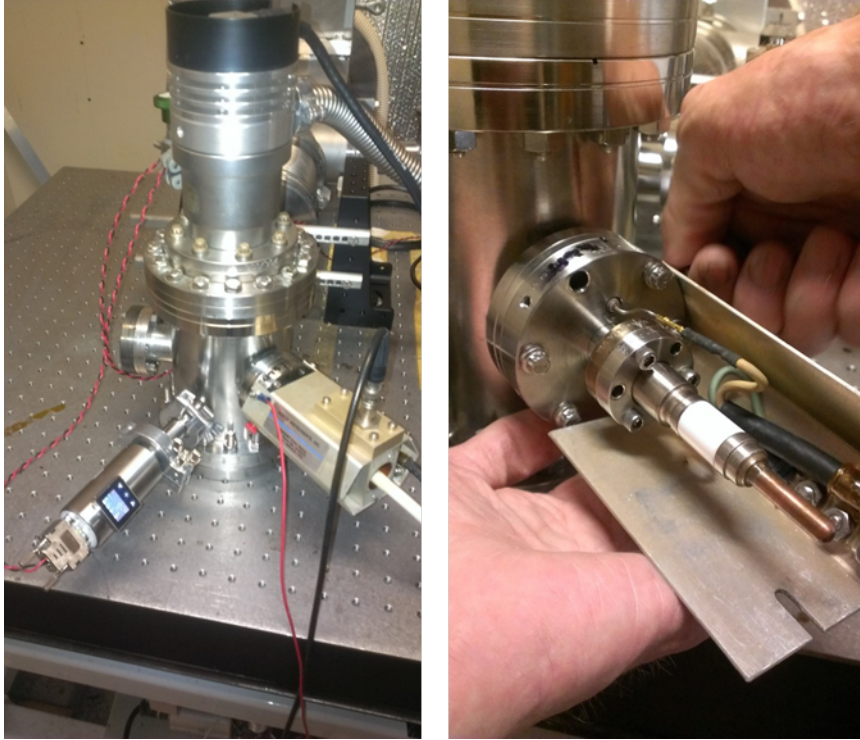


Figure 4.4: (left) Manson source connected to the beamline. (right) A close-up view of the Manson source under the high voltage shield.

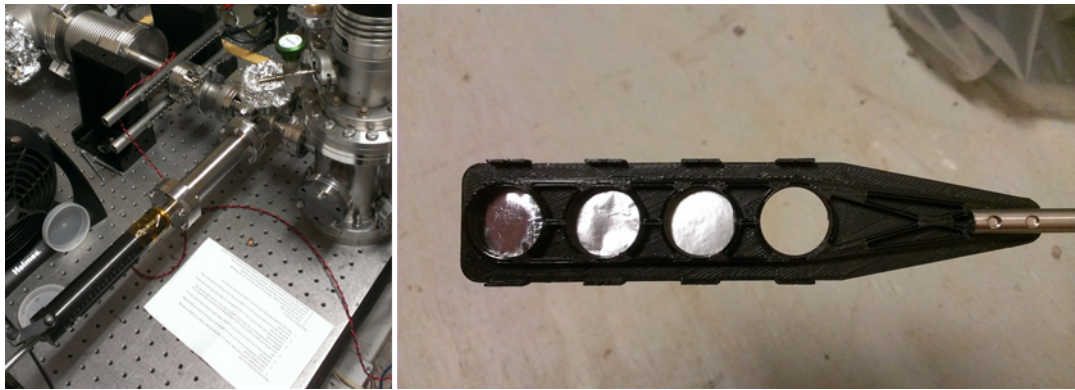


Figure 4.5: Attenuator. (left) The photon flux attenuator was positioned in front of the X-ray source generator to decrease the number of photons incident on the XRC. (right) A 3D printed mount was constructed allowing me to use different thicknesses of aluminum to control the attenuation.

4.1.2 X-ray CCD

A charged coupled device (CCD) is a type of imaging semiconductor detector consisting of a 2D array of pixelated capacitors. An X-ray photon interacts with the surface and the resulting electron cloud is stored as charge in one, or a few, pixels. The charge is transferred to the readout device through a potential change across the pixels. The data are recorded as an event file consisting of information about each X-ray. Understanding the details of these files allows for proper data reduction in order to eliminate noise or background events.

In my work, I used two X-ray CCDs that have very similar properties. The main differences are the cooling mechanics and the size and number of pixels. The next two paragraphs describe the CCDs I used during my work but after that, I will only be referring to the CCD cooled by liquid nitrogen.

The X-ray CCD used at the 100 m beamline, and occasionally at the 600 m beamline, was a 1300 series VersArray Imaging Camera (Roper Scientific, 2017). This one was cooled via an attached liquid nitrogen dewer. The detector array had 1340×1300 $20 \mu\text{m}$ pixels (spatial resolution) and we always cooled it to -120°C for testing. This camera was used as the detector in the flight calibration tests. The energy resolution was found to have a standard deviation of about $\pm 0.20 \text{ keV}$ (see an example spectrum in Figure 4.8). Figure 4.6 shows a picture of this detector and Figure 4.7 shows a picture of a NICER XRC in the 600 m beamline chamber from the point of view of the CCD.

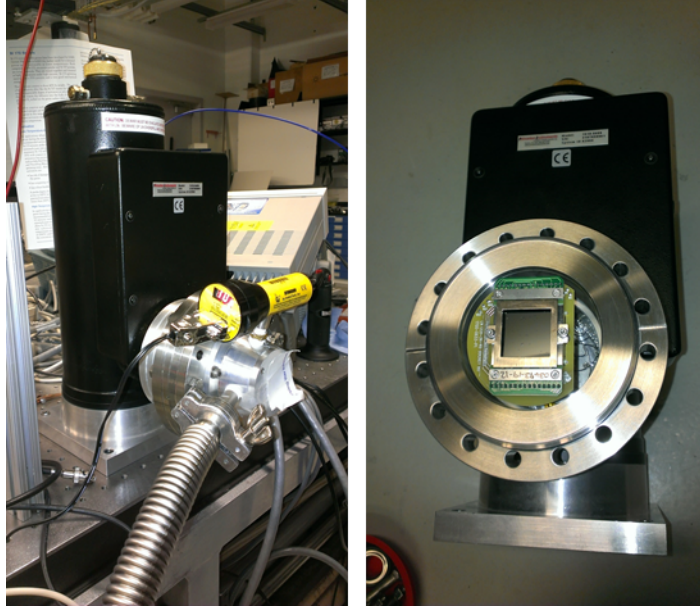


Figure 4.6: Liquid nitrogen cooled CCD. (left) LN Dewer (black) is attached to the X-ray CCD. In this photograph, the CCD housing was placed while under vacuum and tested for air leaks, prior to being used at the beamline for X-ray testing. (right) The surface of the CCD is shown.

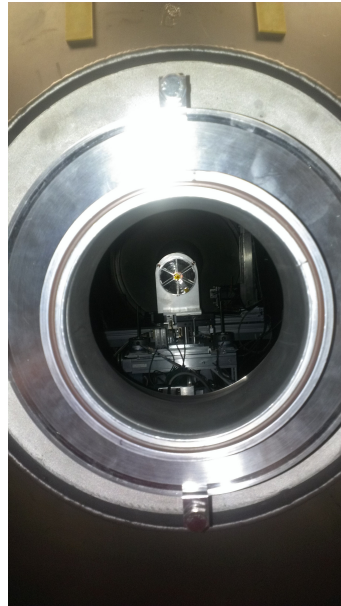


Figure 4.7: An XRC in the vacuum chamber at the detector end of the 600 m beamline. This is the point of view that the X-ray CCD had of the XRC.

The other X-ray CCD used at the 600 m beamline was a PIXIS-XO: 2048B

from Princeton Instruments with 2048 x 2048 pixels (Princeton Instruments, 2017). The temperature was controlled through a thermoelectric cooler (TEC) and was always set to -70°C during measurements. Warmer temperatures produce more noise on the detector and since the liquid nitrogen cooled CCD was capable of getting the temperature lower, it was used for the flight calibration.

The CCD was computer controlled with a Princeton Instruments software, WinView, a data acquisition program for imaging (Princeton Instruments, 2013). We (myself or a team member, whoever was performing the test) controlled the data readout settings such as the CCD region of interest (ROI), the frame time, event and split thresholds, and number of frames. The ROI settings were only constricted during partial module experiments during the XRC development phase. The frame time is an exposure time and it sets the length of time for which data is being collected by the CCD for a specific frame (a frame time of 1 s was used during calibration). If more than one photon interacts with a pixel during one frame, an effect known as pileup would happen. The event and split thresholds are adjusted based on the background noise by limiting the events recorded by WinView. The number of frames is proportional to the length of time an experiment would run. Example CCD settings are shown in Table 4.3.

A lot of my early work with the XRCs involved troubleshooting the measurements and changing the CCD settings to get data that were usable in determining the XRC performance parameters. For some of my work on the CCD trouble shooting and features, see Appendix C.

Keep in mind that NICER’s detectors are silicon drift detectors (SDD), which

are not imaging like these CCDs. The CCDs were needed during calibration to determine the PSF. However, SDDs are a type of semiconductor detector so they use the same basic physics and detect photons via an electron-hole pair.

4.2 Data Analysis and Results

I used data collected during X-ray measurements to calibrate and characterize the XRCs¹⁸. In order to reduce and analyze the data, I wrote code in Python to interpret the raw CCD output and quantize the performance parameters to determine if the modules meet or exceed the mission requirements. My code can be found in Appendix D. In this section I begin by describing the CCD output data in Section 4.2.1 and my interpretation of the raw data file. Following, I discuss how my code analyzes the energy spectrum (Section 4.2.2) and count rate stability (Section 4.2.3) in each set of data. The section ends with an outline of the EEf and effective area calculations (Section 4.2.4) and an example report of generated results (Section 4.2.5).

4.2.1 Interpretation of Raw CCD Output Data

The CCD output data was in the form of an ASCII file, where each event was described. The data files are event files ending in “*.evt*.” There is also a log file (ending in *.log*) produced to describe each frame. The event files are used to analyze each data set and the columns are listed in Table 4.1. The X and Y values described the pixel location of the event and the time-stamp was a function of the

¹⁸I did this during the development of XRCs and up to and during the flight module calibration.

time when the data were collected. The pulse height was a measure of the energy. The pulse height to energy conversion factor was calculated individually for each data set. The grade and split described how charge was spread on the CCD during photon interaction. They are further described below in Sections 4.2.1.2 and 4.2.1.3.

Table 4.1: Event File Columns

Column Name	Description	Range
X	x-coordinate on CCD	<i>CCD/ROI dependent</i> VersArray: 1-1340 PIXIS-XO: 1-2048
Y	y-coordinate on CCD	<i>CCD/ROI dependent</i> VersArray: 1-1300 PIXIS-XO: 1-2048
Pulse Height (ph)	measure of energy	> 0
Grade	describes pattern of charge spread on CCD	0-255
Split	number of additional pixels affected by the electron cloud	0-8
Time-stamp	describes when event was collected	> 0

When analyzing data, events were selected based on the criteria in this section. Parameters were set in the *setData()* function of Python module *XrayAnalysisFunctions.py*, included in Appendix D.1.

4.2.1.1 Pulse Height to Energy Conversion

The energy spectrum is a linear transformation of the pulse height spectrum. Every test has a slightly different pulse height to energy factor, thus it had to be calculated for every data set. For this, I plotted the spectrum, i.e., the number of counts measured for a range of pulse heights. I then found the pulse height at the peak, by fitting the spectrum to a Gaussian, and set that to the known line energy

the source produced, as shown in Figure 4.8, and found the conversion factor by Equation 4.2.

$$\text{line energy} \times \text{conversion factor} = \text{pulse height at peak} \quad (4.2)$$

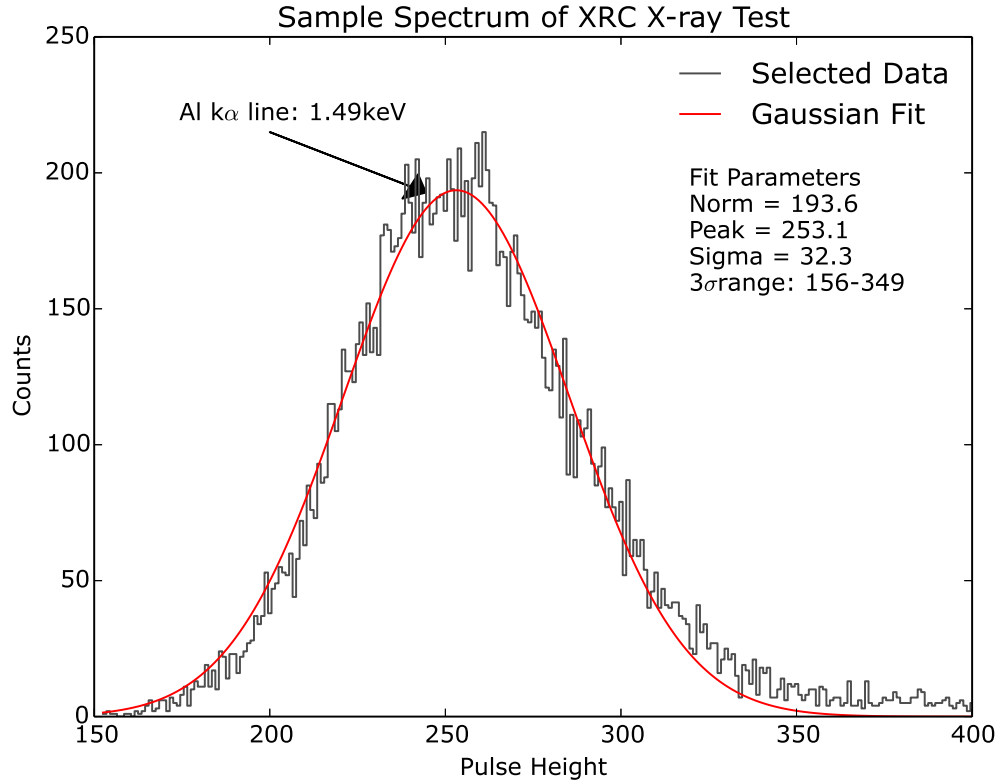


Figure 4.8: A spectrum of the data set is made and the X-ray source lines are found. The pulse height for the lines is found and converted to energy in units of keV. In this case, the pulse height is 253 for a line at energy 1.49 keV so the conversion factor is 169.9 ph/keV as found by Equation 4.2. The standard deviation of ± 32.3 ph corresponds to an energy width of ± 0.2 keV.

4.2.1.2 Split

The split describes charge spread. When a photon interacts with the CCD, it produces a photo electron which then produces a charge cloud. The split is the number of pixels the charge cloud interacts with in addition to the main pixel. Since the maximum number of neighboring pixels allowed in this interaction is 9, the maximum split number is 8. To calculate the energy, or pulse height, the charge from all the pixels involved in the interaction is added up. However, if a pixel did not receive enough charge, then the apparent energy of the photon would be lower. If it did not meet the required split threshold, it would not be counted. In addition, the energy may appear much higher if multiple photons interact with the same pixel and are read as one single event.

The split is a quantity that can be derived from the grade. I used split number during initial testing but the grade was ultimately used in event distinction.

4.2.1.3 Grade

Included in the CCD output event file is the grade of each event. The grade is similar to the split but is more specific in describing which neighboring pixels were affected by the electron cloud. Based on a binary system, the grade quantity ranges from 0 to 255 such that each value is a unique pattern of pixels in a 3 by 3 grid. The values uniquely describing each pixel are shown by Table 4.2 (Chandra X-ray Center, 2016, pg. 125).

I used the event grade to distinguish between “good” and “bad” events while

Table 4.2: Pixel Values for Determining Grade

32	64	128
8	0	16
1	2	4

analyzing the X-ray data. Good grades were patterns which a circle, representing the electron cloud, could realistically be placed over neighboring pixels (as in Figure 4.9). In addition to distinguishing grade significance through laboratory experiments and analysis, which I describe in the following paragraphs, CCD event grades were studied previously by Tsunemi et al. (1999) and Hiraga et al. (2001), and were also reported in Chandra X-ray Center (2016, Chapter 6).

I have encountered four types of patterns that are considered “good events.” I labeled them Type 1, 2, 3, and 4.

Type 1 is an interaction with only the central pixel. This type consists of one pattern what has a grade value of 0 (Figure 4.9a).

Type 2 is an interaction with the center pixel and an adjacent one. These patterns consist of the center pixel (value 0) with either pixel of value 2, 8, 16, or 64 (Figure 4.9b- 4.9e).

Type 3 is an “L” shaped pattern where center main pixel is the corner of the “L.” In addition to the 0 value pixel, each event adds either 64 and 8, 64 and 16, 8 and 2, or 16 and 2. The possible grade values (i.e. the sums of these pixel values) are 10, 18, 80, and 72 (Figure 4.9f- 4.9i).

Type 4 is a pattern such that the X-ray interacts with 4 pixels, one of which is the center. The possible grade values are 11, 22, 104, and 208 (Figure 4.9j- 4.9m).

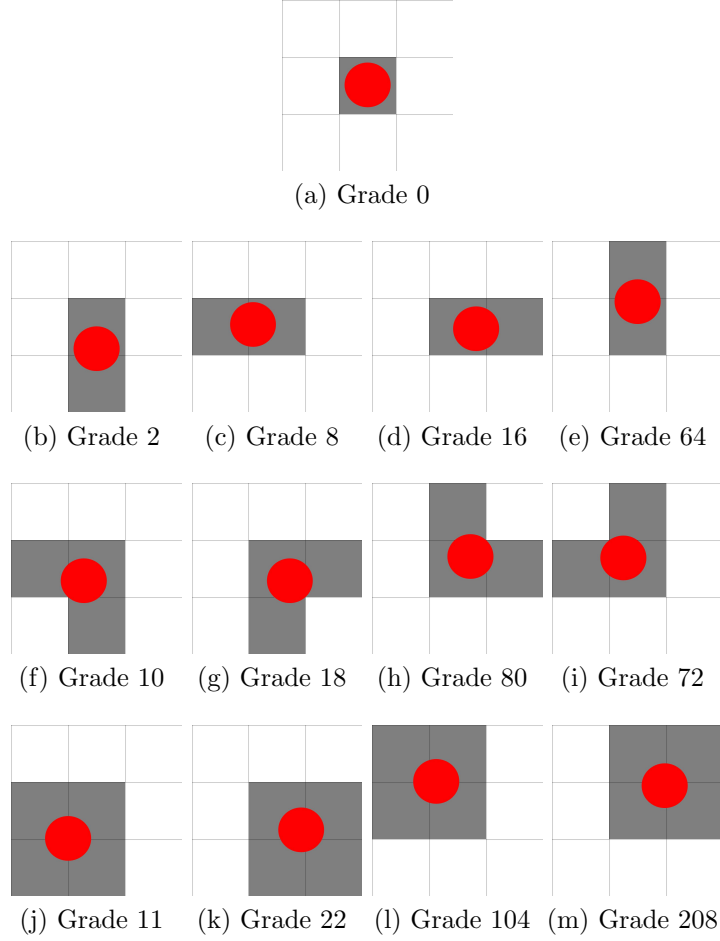


Figure 4.9: These patterns show the “good” grades. For each pattern example, a 3 by 3 pixel region is outlined. The red circle represents where the charge cloud interacted with the CCD. The center pixel is main pixel of the interaction. The shaded regions are all the pixels affected by the charge cloud. The grade values are labeled under each pattern. Type 1 is shown by (a). Type 2 is shown by (b)-(e). Type 3 is shown by (f)-(i). Type 4 is shown by (j)-(m). A pattern outside of these would be considered a “bad” grade.

Taking an example sample data set, I created a histogram of the events based on the grades (Figure 4.10). With the “good” events highlighted, I show that they are more abundant than the other events. There are a few grade values with comparable counts. Grade 255, which is when all 9 pixels are affected, has more counts than some of the “good” events. Other comparable grades are ones with 8

pixels affected. I isolated the events with good grade values and used only those in my analysis. I observed that when good grade events were isolated, noise from environmental factors was minimized.

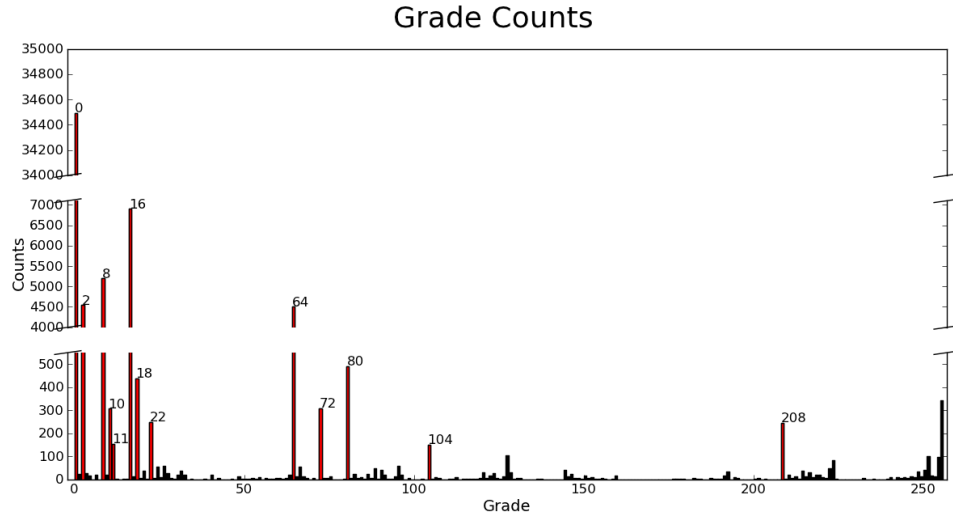


Figure 4.10: This is a histogram of events based on grades from an example sample data set. Labeled and colored in red are the events with “good” grade values.

4.2.2 Spectrum

The energy, or rather pulse height, spectrum of each data set was analyzed as part of the process in determining if a measurement could be used to properly characterize the measured XRC. A proper spectrum would show a single peak at a pulse height corresponding to the X-ray energy generated by the source. This peak should be stable over time; in some cases, the pulse height to energy conversion factor changed over time. I also required that excess background be minimized. A warming CCD or improper event threshold setting caused excess events all over the CCD within the energy band of the X-ray source. A spectrum that changed

over time or contained too many events outside of the characteristic line resulted in measurement rejection. Analyzing the spectrum was one of my most useful tools in troubleshooting the system and determine quality of the data. This is shown by reports in Appendix C.

During analysis, a pulse height range was determined by fitting a Gaussian to the data's energy spectrum. I used the fit parameters to calculate the pulse height to energy conversion factor (Equation 4.2) and the channel range. I used a 3σ width for discriminating events in every XRC calibration.

4.2.3 Count Rate Stability

The count rate of every X-ray measurement was calculated for events within a 3σ width of the pulse height spectrum peak and sometimes at other energies for diagnosis. Figure 4.11 shows a light curve plot for one of the XRC modules. The count rates during the XRC measurement was plotted with the count rates of the incident beam, which was measured before and after the XRC. The incident beam was measured by moving the XRC out of the pathway beamline, exposing the CCD to non-concentrated photons. The uncertainties are one standard deviation. Function *framecount()* in *XrayAnalysisFunctions.py* was the core of this calculation. The measured count rates were used in the effective area calculation (see Section 4.2.4).

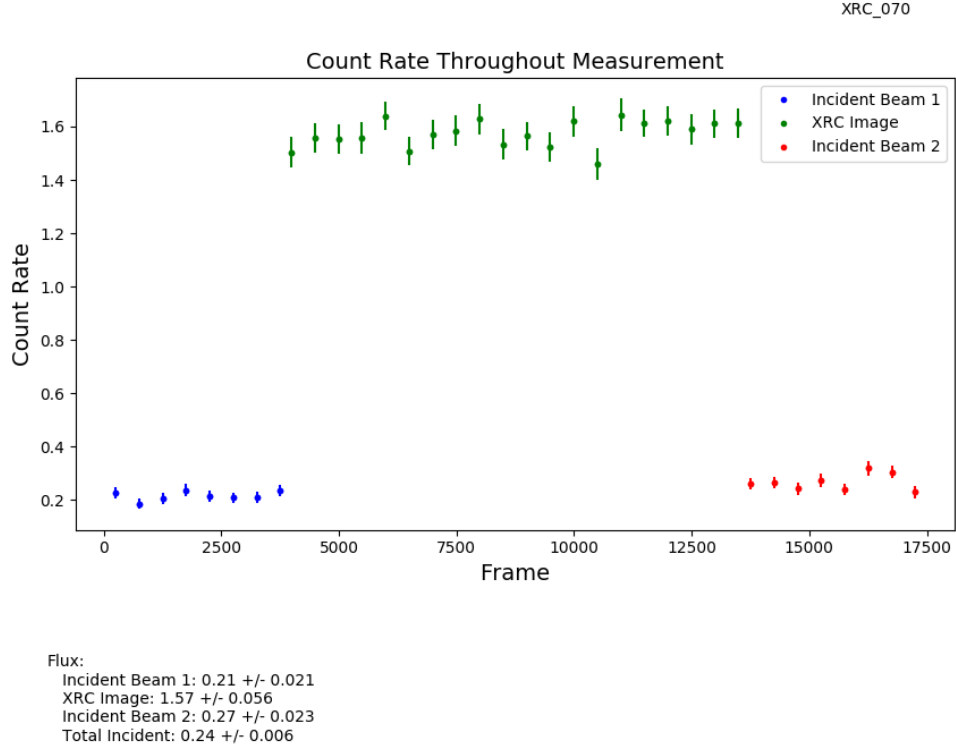


Figure 4.11: Example light curve of an XRC calibration X-ray test. The incident beam measurements, blue and red, show a much lower count rate than the XRC Image, green, since the photon are not focused. The events counted in this light curve had an energy within a 3σ width of the pulse height spectrum peak. The flux of the total incident beam was calculated by combing the data of the first and second incident beam measurements.

4.2.4 Calculating EEF and Effective Area

I introduced the calculation method for the effective area and EEF in Section 2.2. Here I discuss how I applied Equation 2.1 and 2.2 to the X-ray data for measuring these properties of the XRCs. The code for the EEF and effective area functions are in Appendix D.1.3 and D.1.4 respectively. Both the EEF and effective area are energy dependent, however, I report solely on the performance at 1.49 keV

due to technical requirements of the mission.

EEF: After the good data (selected as previously described) are isolated from the complete data set, I find the center of the image by locating the CCD coordinates with the highest number of events. The events are selected based on a pulse height range. A circle is defined for each radius in the calculation, up to the maximum radius. The maximum radius is set by the user and becomes the normalization radius. The number of events within each radius is summed up, creating the function in array form in Python. This array can then be used to determine the fraction of events in a certain radius or vice versa.

Let n be the number of events encircled at a given radius x and N be the total number of events. The EEF is then defined as

$$EEF(x) = \frac{n(x)}{N} \pm d\left(\frac{n(x)}{N}\right), \quad (4.3)$$

where the uncertainty is

$$d\left(\frac{n}{N}\right) = \sqrt{\left(\frac{\sqrt{n}}{n}\right)^2 + \left(\frac{\sqrt{N}}{N}\right)^2}. \quad (4.4)$$

Figure 4.12 illustrates Equations 4.3 and 4.4. Figure 4.13 is an example of an EEF plot created during the analysis.

A_{eff} : For the effective area, an additional measurement had to be taken. A direct beam (also referred to as the incident beam) was collected immediately before and/or after the XRC was measured in the X-ray beam. This had to be done without

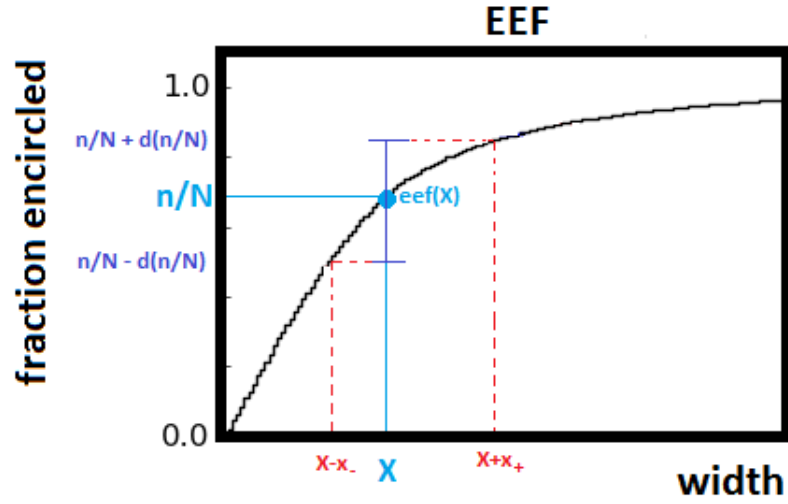


Figure 4.12: A pictorial representation of the EEF error calculation from Equations 4.3 and 4.4.

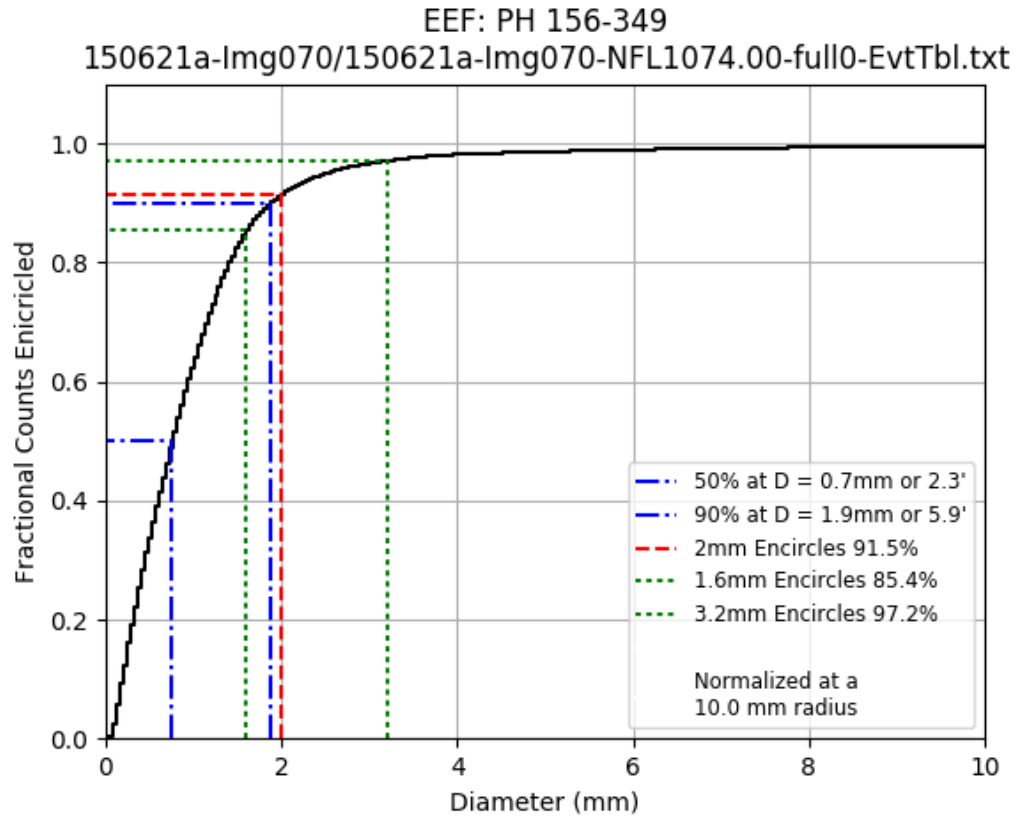


Figure 4.13: Sample EEF found from an XRC X-ray calibration test. The events here had an energy within a 3σ width of the pulse height spectrum peak.

the vacuum chamber being vented and without changing the parameter setting of the CCD or the X-ray source. As with the EEF calculation, the data are first screened based on the ideal parameters, including energy.

Equation 2.2 can be redefined as

$$A_{eff} = A_{ap} \times \frac{(total\ number\ of\ photons\ at\ focal\ point)}{(total\ number\ of\ incident\ photons)}$$

where the *total number of photons at focal point* is the number of events XRC data set. The events were constrained to those within a 2 mm diameter around the image center, the same size as the SDD aperture on NICER. The *total number of incident photons* was calculated from multiplying the in the XRC aperture area (A_{ap}) by the number of photons averaged per frame per mm² from the direct beam file (N_{db}) and by the number of frames used in the focused X-ray data set (N_{fr}):

$$total\ number\ of\ incident\ photons = A_{ap} \times N_{db} \times N_{fr}.$$

The value used for A_{ap} was the cross-sectional area on the incident side of the XRC module, but this variable is canceled out in the calculation.

4.2.5 Example Analysis Report

Along with Figures 4.11 and 4.13, Figures 4.14, 4.15, 4.16, 4.17 and 4.18, complete the set of an analysis report summary produced by my code for each XRC calibration test. The ID of the XRC module tested in this example is *XRC_070*

meaning that it was the seventieth flight-ready XRC module made. This module met the NICER requirements. The X-ray source and CCD settings of this example report are described in Table 4.3.

Table 4.3: Measurement parameters of the example analysis report shown in this chapter.

<i>Source Settings</i>	
Al Attenuator Thickness	0.5 mil (i.e., 50.8 μm)
Source High Voltage Setting	2 kV
Source Current Setting	0.065 mA
<i>CCD Settings</i>	
Frame Time	1 s
Event Threshold	100
Split Threshold	5
Number of Frames Collected for XRC	10000
Number of Frames Collected for Incident Beams	4000

Figure 4.14 shows the X-ray image of photons concentrated by the XRC and a scatter plot of events. The pixel coordinates are labeled, which shows that the image on the left is the zoomed in section of the scatter plot on the right. The events seen in this image were used throughout the analysis reported from this particular measurement.

Figures 4.15–4.17 are examples of three energy spectra (in units of the pulse height) taken from a full measurement. These plots show data fit to a Gaussian and display the fit parameters (normalization constant, peak location, and the sigma width), the 3 sigma range, and the number of channels within the 3 sigma range. Figure 4.15 is the energy spectrum from data collected when the XRC was in the pathway of the beam (i.e., “XRC image”). This corresponds to the green dots

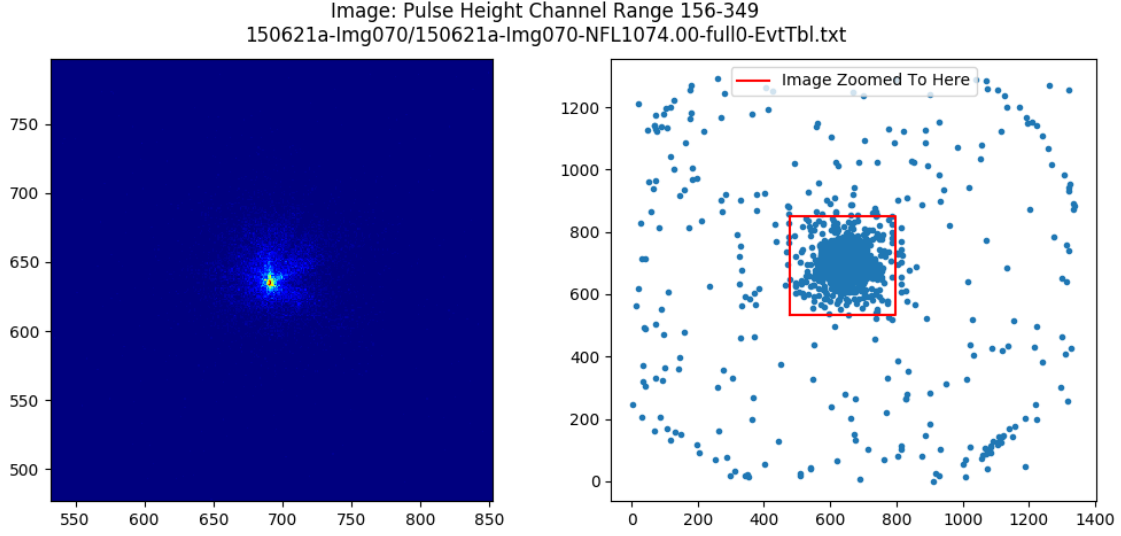


Figure 4.14: X-ray image of focused XRC data taken in a beamline measurement. (Left) 2D histogram. (Right) Scatter plot of events.

in Figure 4.11. Figures 4.16 and 4.17 are the energy spectra from data collected from a direct beam measurement, when the XRC was moved out of the pathway of the beamline (i.e., “incident beam” or “direct beam”). Incident beam 1 (Figure 4.16) was measured immediately before the XRC and the incident beam 2 data (Figure 4.17) were collected immediately after the XRC data collection. These correspond to the blue and red dots in Figure 4.11 respectively.

Figure 4.18 reports the information and measurements required for determining the quality of the measured XRC module. Included are the data files used for the analysis, EEF results, effective area results, pulse height channel selection, and photon count rate. The pulse height selection states the channels used in the rest of the results on the page. The EEF results reported the percent of photons encircled at three different diameters and stated the normalization radius used in the calculation. Under effective area results are the total and 2 mm effective area measured

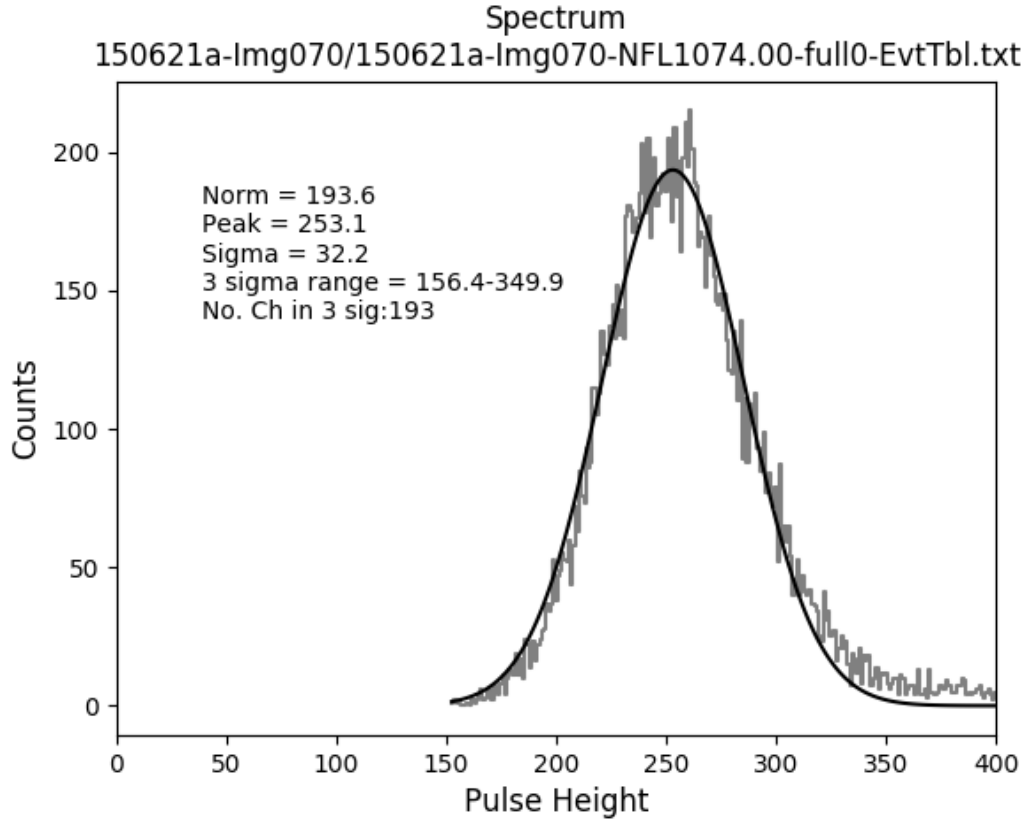


Figure 4.15: Energy spectrum, in units of the pulse height, of the XRC measurement of a single, flight ready, module. The data are shown in gray fit to a Gaussian shown by the black line. The fit parameters are printed on the plot.

directly from the data, including the 1σ uncertainty. The third line shows what the 2 mm effective area would be if the EEF was multiplied by the total effective area. The beam flux reported contains five values. The full image and 2 mm image are for the XRC data and represent the count rate over the whole CCD and the 2 mm aperture respectively. The final three lines are for the incident beams, calculated separate for the first two and averaged for the final. The effective area calculations used the count rate from the combined incident beams.

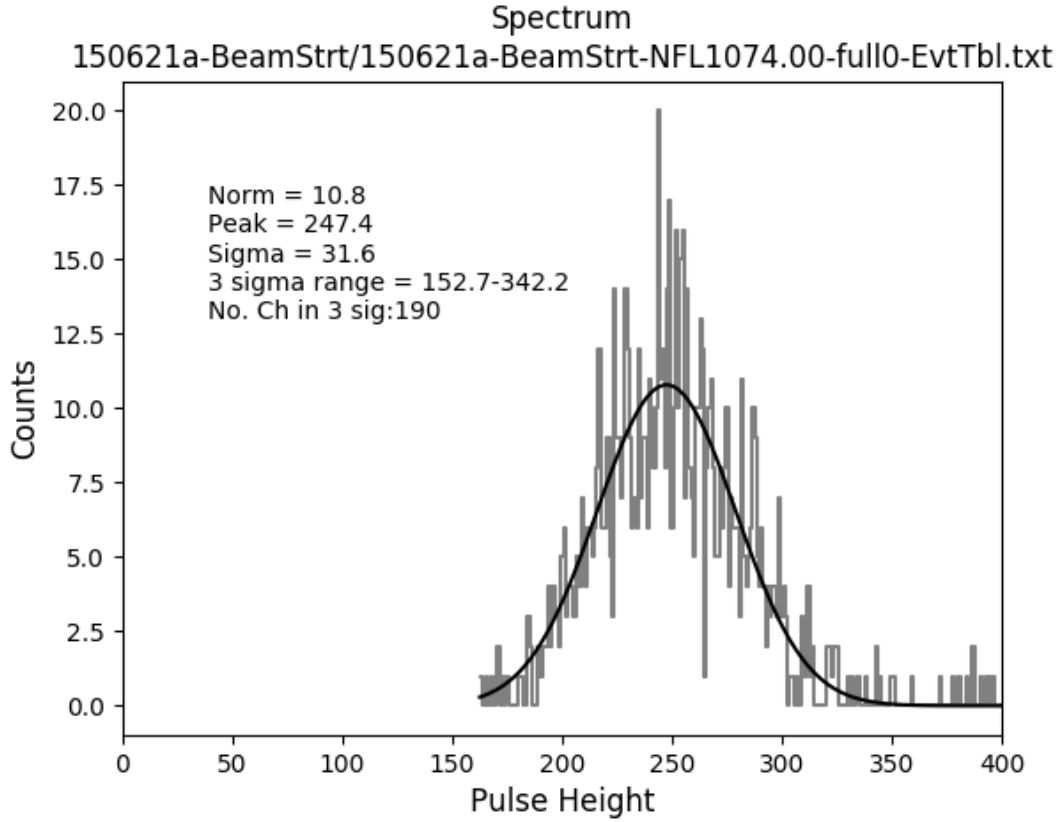


Figure 4.16: Energy spectrum, in units of the pulse height, of an incident beam measurement collected prior to the XRC measurement. The data are shown in gray fit to a Gaussian shown by the black line. The fit parameters are printed on the plot. This spectrum should be similar to the incident beam measurement collected after the XRC was measured (Figure 4.17) in order for the measurement to be considered successful and useful for calibration.

4.3 XRC Performance Features

In Figure 4.19 I present the quantifying factors I measured from the X-ray tests for each of the 70 XRCs produced during the flight fabrication stage of NICER. Of these 70 modules, 64 met missions requirements (Figure 1.3). This analysis shows an average effective area of $48.18 \pm 2.82 \text{ cm}^2$ and a mean EEf at 2 mm of $90.8\% \pm 2.39\%$.

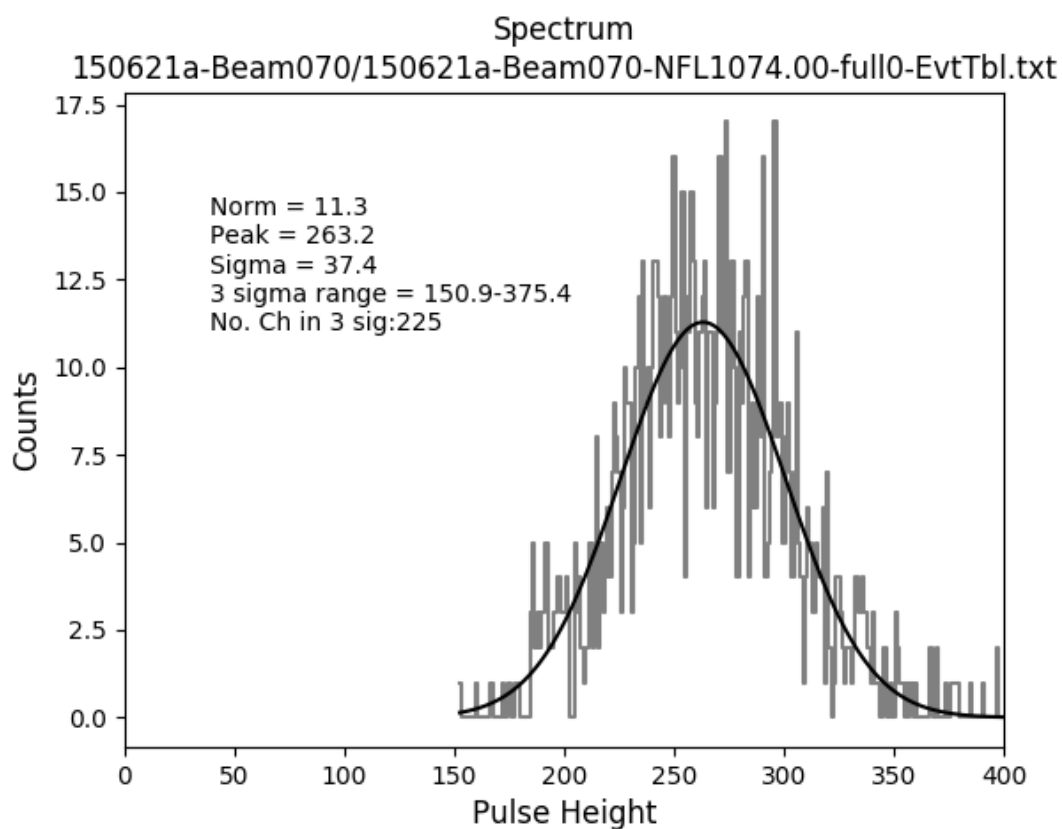


Figure 4.17: Energy spectrum, in units of the pulse height, of an incident beam measurement collected after the XRC measurement. The data are shown in gray fit to a Gaussian shown by the black line. The fit parameters are printed on the plot. This spectrum should be similar to the incident beam measurement collected before the XRC was measured (Figure 4.16) in order for the measurement to be considered successful and useful for calibration.

X-ray Data Results XRC SN#070 Data Set 150621a

Image File: 150621a-Img070/150621a-Img070-NFL1074.00-full0-EvtTbl.txt
Beam1 File: 150621a-BeamStrt/150621a-BeamStrt-NFL1074.00-full0-EvtTbl.txt
Beam2 File: 150621a-Beam070/150621a-Beam070-NFL1074.00-full0-EvtTbl.txt

EEF Results

D = 2mm: 91.5%
D = 1.6mm: 85.4%
D = 3.2mm: 97.2%
(normalized at R = 500.0 pix, 10.0 mm)

Effective Area Results

Total: 45.35 +/- 1.1 cm²
2mm: 41.51 +/- 1.01 cm²
Tot*2mmEEF: 41.5 +/- 1.01 cm²

Pulse Height Channel Selection

Image: 156 - 349 (193 total channels)
Beam1: 152 - 342 (190 total channels)
Beam2: 150 - 375 (225 total channels)

Beam Flux

Full Image: 1.56 +/- 1.25 ct/sec
2mm Image: 1.43 +/- 1.2 ct/sec
Inc Beam1: 0.21 +/- 0.47 ct/sec
Inc Beam2: 0.27 +/- 0.52 ct/sec
Tot Inc Beam: 0.24 +/- 0.49 ct/sec

Figure 4.18: Sample XRC calibration output. This shows the quantifying factors in determining the quality of a single XRC module.

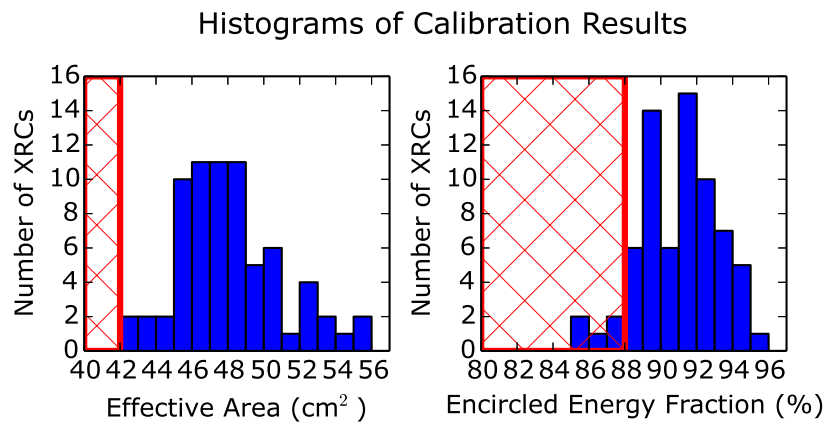
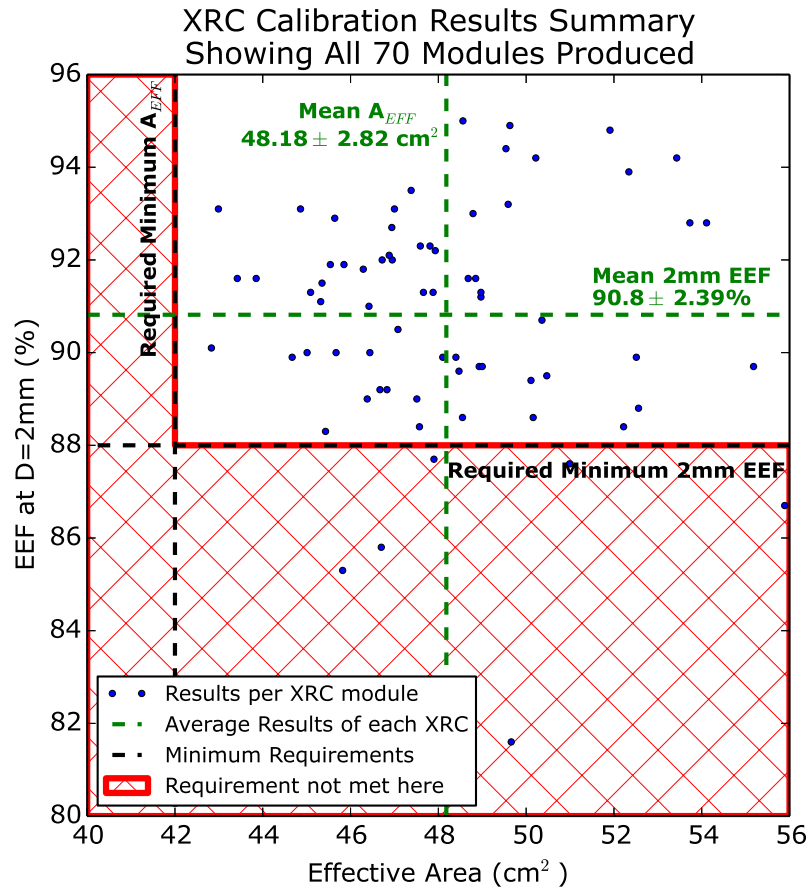


Figure 4.19: Performance results of all 70 XRCs produced.

Chapter 5

Planning the NICER Observation of Pulsar J0437-4715 from the Optics Point of View

In this chapter, I describe a method to effectively observe PSR J0437-4715 by pointing off-axis from the source. I conducted a simulated observation with a focus on the response of non-imaging optics using data analysis from Chapter 4. Recall from Chapter 1 that an AGN is located $4.2'$ from the pulsar. With a brightness $\sim 10\times$ brighter than PSR J0437-4715, the flux from AGN RX J034-4711 could negatively affect the observation of the pulsar. My results suggest to point NICER $2.7'$ away from PSR J0437-4715, the opposite direction of AGN RX J034-4711, in order to maximize the pulsar to AGN flux ratio.

5.1 Background

Being one of the stable configurations at the end of a star's life cycle, neutron stars' properties and their distribution across the galaxy are significant interest in astronomy (Baykal, 2005). An accurately measured Equation of State (EOS) could allow us to understand the microscopic behavior of nuclear matter through inferring the interior composition of neutron stars (Freire et al., 2009). This is done through observing the general relativistic effects from the strong gravitational fields to calculate the mass to radius relation of a neutron star (van der Klis, 2005).

Figure 5.1 shows current EOS models along with NICER’s expected contribution for constraining the mass to radius relationship.

Lightcurve analysis, via phase-dependent spectra, can be used to measure the radii of neutron stars, which, when combined with mass measurements, constrains the EOS for neutron stars (Gendreau et al., 2012). As an example of work done with data from a previous mission, Bogdanov (2013) observed PSR J0437-4715 with XMM-Newton and produced energy-resolved lightcurves and phase-resolved spectra. He used four energy ranges between 0.5 and 1.8 keV, which were chosen solely for optimal photon statistics. Energies above 3 keV were excluded due to the high background and photons below 0.5 keV were contaminated by electronic noise. Also due to photon statistics, Bogdanov (2013) chose to extract the spectra from only four phase intervals. In order to measure the radius, he fitted neutron star emission models¹⁹ to phase-resolved spectra simultaneously with fit parameters for emission and pulsar geometry tied across phases. He concluded that this method is a “potentially powerful approach” to constraining neutron star parameters (Bogdanov, 2013, Section 7). While the neutron star radius was constrained to >11.1 km at 3σ confidence (assuming $M = 1.76 M_{\odot}$), greater photon statistics would allow for even more stringent constraints, especially at energies >3 keV (Bogdanov, 2013).

The photon statistics will be greatly improved by the XRCs on NICER’s X-ray Timing Instrument (XTI). In comparison, the effective area, which is a significant factor for increasing photon statistics, for XMM-Newton’s X-ray telescope

¹⁹Emission models tested in Bogdanov (2013) for phase-resolved spectroscopy were a double blackbody model and a hydrogen atmosphere model, each with an additional non-thermal component (Bogdanov, 2013, Section 5 and Table 2).

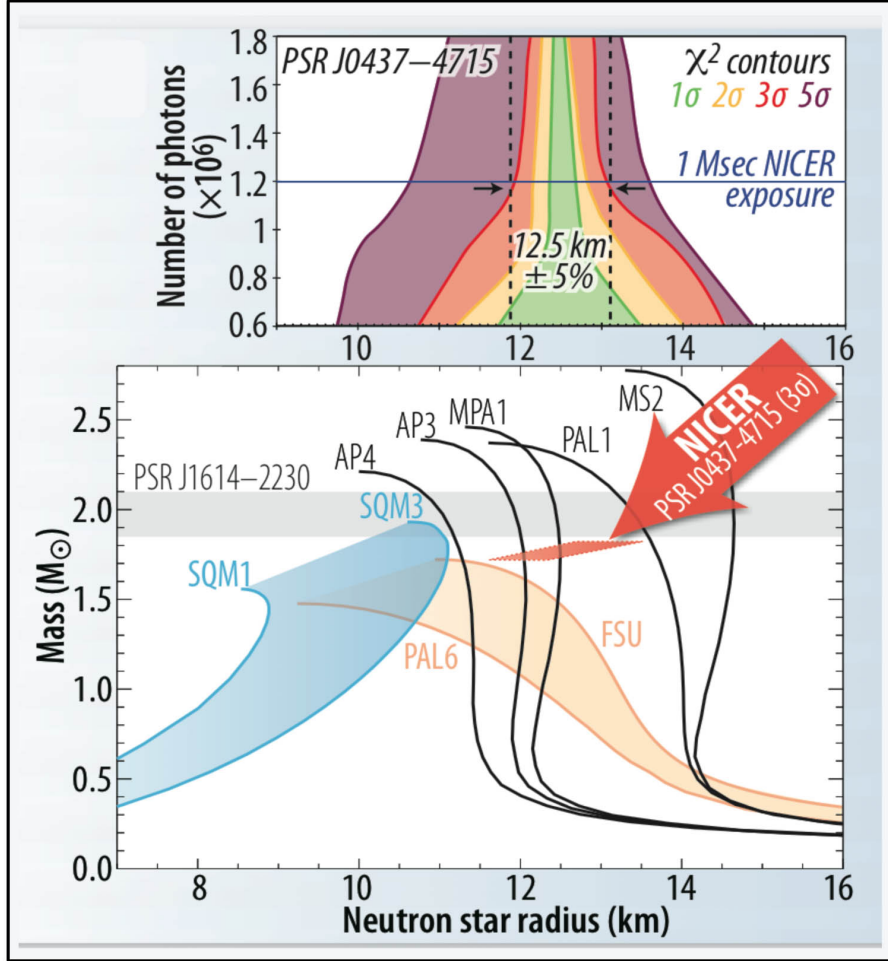


Figure 5.1: Current constraints for neutron star masses to radii are shown in the bottom plot along with current EOS models (black, blue, and, yellow lines and the blue and yellow shaded regions) and the constraint NICER will provide on the measurements (red region). The shaded gray region is the current mass and radius constraint for the labeled pulsar. The top plot shows the predicted constraints NICER will have on measuring the radius of PSR J0437-4715 for different lengths of measurements. This figure can be found in the About section of the NICER homepage (NASA, NICER, 2017). *Image credit: NASA*

is 1500 cm^2 at 1 keV, for ROSAT (PSPC instrument) is 240 cm^2 at 1 keV, and for NICER's XTI is 2500 cm^2 at 1.5 keV (NASA, 2013). The technology on NICER allows for more incoming source photons to be processed by the detector while concentrating the light well enough to limit the instrument induced background noise. By increasing the number of photons collected by the detector and decreasing the contribution of background events we can produce spectra based on small phase intervals and lightcurves in more narrow energy bands than previously possible with XMM-Newton and ROSAT. With high photon statistics a good spectral model fit will improve constraints on physical properties of the neutron star, e.g. the shape and size of the star and location of the polar caps.

5.2 Simulation Summary

NICER's observation plan includes observing PSR J0437-4715 for 1–1.5 Msec, which would allow for energy-dependent lightcurve analysis to infer the neutron star's radius with $\pm 5\%$ accuracy (Arzoumanian et al., 2014). This target is of interest because it is the nearest MSP and therefore bright with respect to other pulsars. The AGN RX J0437.4-4711 is close enough to NICER's $6'$ diameter FOV²⁰ of the pulsar, located only $4.2'$ away. The AGN photon contribution could be detrimental to a successful timing observation since they cannot definitively be distinguished from the pulsar's photons. I ran a series of simulations to better understand the AGN's photon contribution in order to propose an observation method for the MSP.

Pointing the XTI slightly off-axis from the pulsar, away from the direction to-

²⁰A $6'$ diameter FOV corresponds to ≈ 30 arcminutes squared FOV area.

wards the AGN, increases the count rate ratio of pulsar to AGN photons NICER can observe. Knowing that the optics effective area curve is dependent on the photons' incident angle and energy, the count rate ratio between the pulsar and AGN contribution must be calculated for different energy bands and pointing directions. To do this, I modeled the energy spectra of the AGN and the pulsar. In my preliminary studies, I used a simple model for the pulsar and then advanced to a more realistic, atmospheric emission, model as described by Bogdanov (2013). I conducted a preliminary study (Section 5.6.1) to develop my procedure for the more complicated simulation. In my preliminary study, I used a blackbody model without energy-dependent phase analysis and in the main study I applied an atmospheric emission model with energy-dependent lightcurves.

In Section 5.3, I describe the simulation of both sources and the parameters I used to create models of the emergent spectra. Then in Section 5.4 I discuss the specifics about how I made the phase-phase dependent pulsar model. I derived an equation for the flux of the pulsar as a function of energy and rotational phase. A component of this equation is the emergent intensity as a function of energy and angle of emission. I computed this factor using The McGill Planar Hydrogen Atmosphere Code (McPHAC), developed by Haakonsen et al. (2012). By combining the McPHAC output with pulsar flux equation in Python, I simulated energy-dependent lightcurves and phase-dependent energy spectra. I then generated NICER response files to simulate an observation of theses spectra (Section 5.5). My codes are found in Appendix E.

5.3 Objects in Field of View

PSR J0437-4715 is located at RA $04^{\text{h}}37^{\text{m}}15.81^{\text{s}}$ and Dec $-47^{\circ}15'8.6''$. AGN RX J0437.4-4711 is a Seyfert 1 galaxy located at RA $04^{\text{h}}37^{\text{m}}28.14^{\text{s}}$ and Dec $-47^{\circ}11'29.86''$. Both sources' locations are based on the J2000 equinox (RX in the AGN's name represents the mission by which it was discovered, ROSAT). The angular distance between the sources is reported as $4.3'$ by Halpern et al. (1996) and $4.2'$ by Wang et al. (1998).

5.3.1 PSR J0437-4715

PSR J0437-4715 is the closest millisecond pulsar and has a cool white dwarf companion, making it a binary. It is located at a distance of 156.3 pc , or 509.8 ly , and rotates with a period of 5.76 ms (Bogdanov, 2013).

There are several different models in the literature describing its energy-spectrum. For example, Becker and Trümper (1993) considered a power-law model and a power-law with a thermal component, ultimately showing that a power-law model alone was not sufficient to describe the PSR J0437-4715 energy spectrum and suggesting that there must be a thermal component present. Halpern et al. (1996) reports on a power-law model, a power-law plus blackbody model, and a double blackbody model. My work is based on the models used by Bogdanov (2013) that were mentioned in Section 5.1. I used a double blackbody plus power-law model to simulate the pulsar energy spectrum for my preliminary studies and a double hydrogen atmospheric emission model with a power law for the more in depth simulation.

Halpern et al. (1996) used a ROSAT observation of the pulsar, collected over a 20 day period, to fit the data with an absorbed double blackbody spectrum within the 0.11–2.00 keV energy band. A double blackbody represents two components of thermal emission from the surface of the neutron star. I used parameters from this fit, which consist of temperatures $T = (1.0\text{--}3.3) \times 10^6$ K and $T = (4\text{--}12) \times 10^6$ K for the blackbodies and a column density of $N_{\text{H}} = (5\text{--}8) \times 10^{19} \text{ cm}^{-2}$ to recreate this spectrum as a model in XSPEC 12.8.2 (Arnaud et al., 2013). The parameters I used are given in Table 5.1. The normalization values were found by matching the curve to Figure 8 in Halpern et al. (1996) and are shown by the blue curves in Figure 5.2. Even though the model does not represent > 2 keV well, I used the results from Halpern et al. (1996), to simulate the pulsar’s spectrum as an absorbed double blackbody, because the spectral parameters I chose to use for the AGN were found with data from the same observation (see Section 5.3.2).

Pulsar: Absorbed Double Blackbody Spectral Model XSPEC: phabs(bbody+bbody)			
Spectral Parameter		Value	
Photon Absorption	N_{H}	5.0×10^{-3}	$(\times 10^{22}) \text{ cm}^2$
Blackbody 1 Temperature	kT	8.0×10^{-2}	keV
Blackbody 1 Normalization	norm	1.32×10^{-2}	
Blackbody 2 Temperature	kT	0.24	keV
Blackbody 2 Normalization	norm	8.05×10^{-6}	

Table 5.1: Absorbed double blackbody model parameters for PSR J0437-4715 used in this work. These parameters are based on the fitted spectra in Halpern et al. (1996). In XSPEC, the normalization constant for a blackbody model is defined as “ L_{39}/D_{10}^2 , where L_{39} is the source luminosity in units of 10^{39} erg/s and D_{10} is the distance to the source in units of 10 kpc” (Arnaud et al., 2013).

The second energy spectral model I used for my simulations of PSR J0437-4715 was based on the theory of a hydrogen atmosphere being present on the pulsar’s

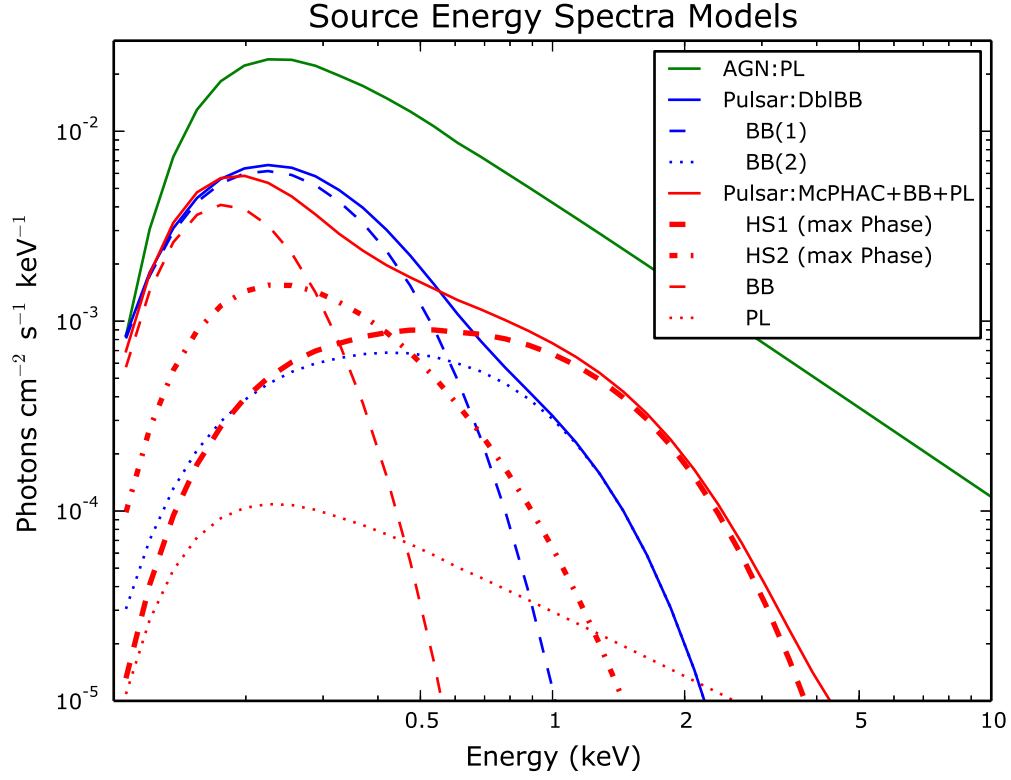


Figure 5.2: Model energy spectra for AGN RX J0437-4711 and MSP PSR J0437-4715 created in XSPEC. The AGN is shown by the solid green line for an absorbed power law spectrum. Two different models of the pulsar are shown. The solid blue line is an absorbed double blackbody where the blue dashed and dotted lines are the blackbody components. BB(1) is the $kT = 8 \times 10^{-2}$ keV component and BB(2) is the $kT = 0.24$ keV component. The solid red line shows the absorbed atmospheric emission model from two hot spots, a cool blackbody, and a power law. The thick lines are for the hot spots where the dashed line is the primary spot at the phase of maximum flux and dashed-dotted line is the secondary spot at the phase of maximum flux. The thin dashed line is the cool blackbody and the dotted line is the power law. All model parameters are given in Tables 5.1, 5.2, and 5.3. The primary hot spot is typically defined as the emission spot seen more often by the observer than the secondary hot spot.

surface. The work in Bogdanov (2013) presents several spectral models, including some with phase-dependency, of PSR J0437-4715 using data collected by XMM-Newton in the 0.1–10 keV band (instrument: EPIC MOS1/2 imaging CCDs) and the 0.3–2 keV (instrument: EPIC pn in one-dimensional imaging mode.). To recreate

the model in XSPEC, I made table models of the two emission spots on the pulsar at several rotational phases. Parameters for the star’s geometry and hot spot emission came from the work in Bogdanov (2013). Section 5.4 explains the physics that went into creating the table models, including specific parameters. In addition to the table model for each of the two emission spots, I added a power-law and a cool blackbody and applied photon absorption. The cool blackbody represents an emission region around the high-temperature hot spots at the polar caps. Bogdanov (2013) used this component as it was implied to be present by Zavlin et al. (2002) and Zavlin (2006). The power-law represents a non-thermal component, generally attributed to synchrotron radiation in the magnetosphere. However, the source of the non-thermal radiation in PSR J0437-4715 is not completely understood (Guillot et al., 2016). Bogdanov et al. (2006) suggests that the Comptonization of thermal radiation from the polar caps could result the non-thermal emission.

The parameters I used for the cool blackbody, photon absorption, and power law came from Bogdanov (2013), Table 1, line 12 and the hot spots’ emission parameters from Bogdanov (2013), Table 2, line 6. I chose the normalization values by matching the spectrum to Bogdanov (2013), Figure 3 and selecting the phase of maximum flux for each hot spot to reproduce an averaged spectrum. The parameters are given in Table 5.2 and the spectral components are plotted in Figure 5.2, represented by the red lines.

Pulsar: Atmospheric Emission Spectral Model XSPEC: phabs(table[HS1]+table[HS2]+bbody+powerlaw)			
Spectral Parameter		Value	
Photon Absorption	N_{H}	7.0×10^{19}	cm^2
Hot Spot 1 Temperature	$T_{\text{eff},1}$	2.21×10^6	K
Hot Spot 1 Radius	$R_{\text{eff},1}$	0.16	km
Hot Spot 2 Temperature	$T_{\text{eff},2}$	0.57×10^6	K
Hot Spot 2 Radius	$R_{\text{eff},2}$	5.5	km
Blackbody Temperature	kT	4.41×10^{-2}	keV
Blackbody Normalization	norm	0.40×10^{-6}	
Power Law	Γ	1.15	
Power Law	norm	0.15×10^{-5}	

Table 5.2: Atmospheric emission spectral model parameters I used to simulate emission from PSR J0437-4715. The hot spot components were added to XSPEC as table models. The values are based on Bogdanov (2013). Refer to Table 5.1 for the definition of the blackbody normalization constant. The power law normalization is defined by XSPEC as “photons/keV/cm²/s at 1 keV” (Arnaud et al., 2013).

5.3.2 AGN RX J0437.4-4711

AGN RX J0437.4-4711 is a Seyfert 1 galaxy that was discovered by the ROSAT All Sky Survey (RASS) (Wang et al., 1998). Data collected over a 20 day period from ROSAT PSPC were analyzed and fit in Halpern and Marshall (1996) to an absorbed power law spectrum in the energy range 0.11–2.47 keV. I reproduced this spectrum as a model in XSPEC using parameters in Table 5.3. The spectrum is shown by the green line in Figure 5.2. The normalization was found by setting the flux in the 0.11–2.47 keV band to $1.47 \times 10^{-11} \text{ ergs cm}^{-2} \text{ s}^{-1}$, based on the results in Halpern and Marshall (1996), for which they used data taken over a 6142 s observation over a two day period in September 1992 with a measured luminosity of $1.7 \times 10^{44} \text{ ergs/s}$. This time period was too short to observe any significant flux variability over time.

Flux variability has been observed by Halpern and Marshall (1996) and Wang

AGN: Absorbed Power Law XSPEC: phabs(powerlaw)		
Spectral Parameter		Value
Photon Absorption	N_{H}	$1.0 \times 10^{-2} \quad (\times 10^{22}) \text{ cm}^2$
Power Law	Γ	1.56
Power Law	norm	4.3×10^{-3}

Table 5.3: Absorbed power law spectral model parameters for AGN RX J0437.4-4711 used in this work. The values are based on Halpern and Marshall (1996). Refer to Table 5.2 for the definition of the power law normalization constant.

et al. (1998) with large changes occurring over several days, but spectral variability was not observed. There have been few observations of this object and not much is known about the long term variability in the soft X-ray range. Ideally, the best observation of PSR J0437-4715 would be conducted when the AGN flux is low. As an example, analysis by Halpern and Marshall (1996) of the 20 day observation conducted in 1994 showed a maximum luminosity within 0.1–2.4 keV band to be 5×10^{44} ergs/s, averaging 2×10^{44} ergs/s over the time period. Further information on the AGN observations from the literature is provided in Appendix F.

5.4 Creating a Virtual Pulsar

In order to properly simulate a NICER observation, I have required the virtual pulsar to contain information of observed flux in the form of energy-dependent light curves and phase-dependent spectra. The radiation from a millisecond pulsar is dominated by thermal emission at the magnetic poles while rotating about an independent spin axis (Zavlin, 2009). The emission properties can change depending on the stellar surface composition and presence of an atmosphere. To simulate emission from the neutron star’s magnetic poles, i.e. the hot spots, I have chosen

to use the atmospheric emission model, McPHAC, described in Haakonsen et al. (2012) that solves the radiative transfer equation to produce a simulated anisotropic emergent spectrum. Thermal radiation at the surface interacts with a hydrogen atmosphere causing anisotropic emission because photons emitted at different angles relative to the surface penetrate different thicknesses of the atmosphere (Özel, 2013). I have chosen the orientation of the poles relative to the spin axis and the spin axis to the observer’s line of sight (LOS). During rotation, the angles between the poles and LOS change, which directly affects the observed flux over phase. Stellar mass and radius, distance from observer, spin period, and apparent emission spot sizes are further input parameters of the model. With complete control of the stellar geometry and emission, I have a useful tool to complete my NICER simulations. The geometric input parameters are listed in Appendix E.2 and in Table 5.4.

Assumed Pulsar Geometry Values for PSR J0437-4715		
Variable	Value	Description
R_{NS}	13.5 km	Neutron Star Radius
M_{NS}	1.76 M_{\odot}	Neutron Star Mass
D	156.3 pc	Distance
P	5.76 ms	Rotational Period
i	42°	Spin Axis Inclination
θ	36°	Primary Hot Spot Colatitude
$\Delta\theta$	-25°	Secondary Hot Spot Inclination Offset (from 180° of Primary)
$\Delta\phi$	-20°	Secondary Hot Spot Phase Offset (from 180° of Primary)
$R_{\text{eff,HS1}}$	0.16 km	Primary Hot Spot Effective Radius
$R_{\text{eff,HS2}}$	5.5 km	Secondary Hot Spot Effective Radius

Table 5.4: Geometry values assumed for PSR J0437-4715. These are based off of Bogdanov (2013) (table 2, line 6).

5.4.1 Pulsar Flux: Deriving The Phase and Energy Dependent Flux

$$F(E, \phi)$$

The observed flux is commonly defined as the intensity emitted from an object multiplied by the amount of area in the observer's sky this object occupies, the solid angle. In this section, I begin with the definition of bolometric flux, to derive an equation for calculating the observed monochromatic flux as a function of rotational phase. The final equation (Equation 5.5) is in terms of the physical parameters in the co-rotating frame at the stellar surface, i.e., the non-redshifted parameters. This allows me to calculate the observed flux based on the emergent radiation at the surface (Section 5.4.2) and the real geometry of the pulsar.

Unprimed variables represent the observers reference frame (e.g., A)²¹ and primed variables represent the co-rotating reference frame at the stellar surface (e.g., A'). Since the simulations in this chapter use discrete energy and phase bins, I define some variables as ΔA rather than the differential dA . The geometric variables in this derivation are shown in Figure 5.3.

²¹Let A be some variable.

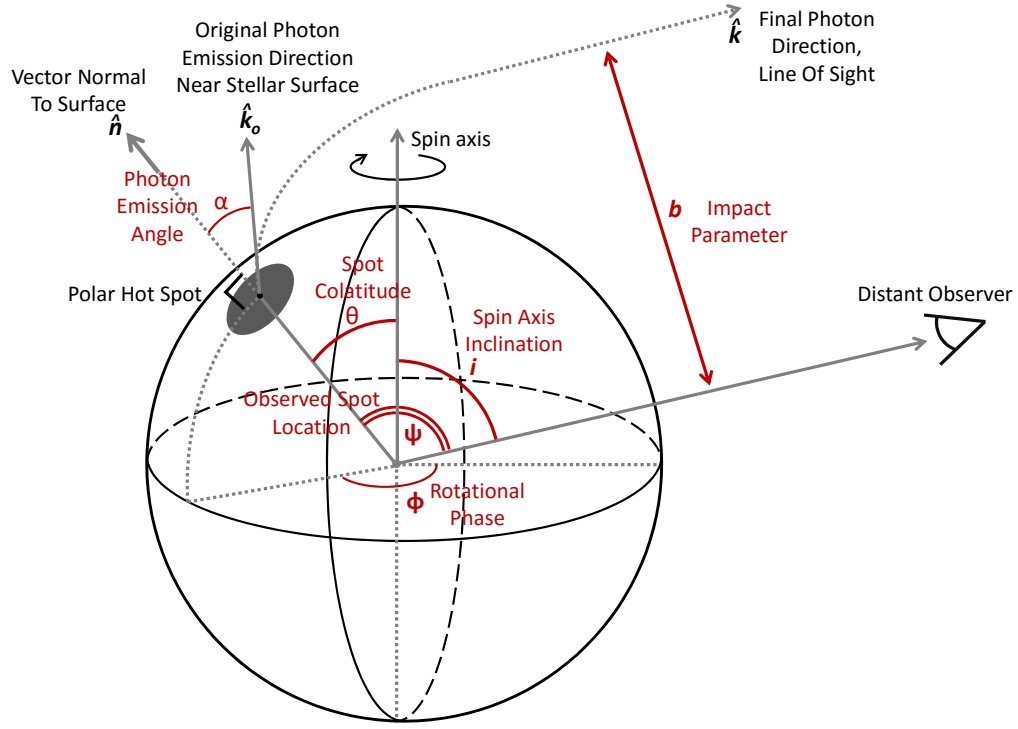


Figure 5.3: Pulsar Geometry. Shows variables used in text. Based on Figure 1 in Viironen and Poutanen (2004).

Equation 5.1 defines flux, F , as the product of intensity, I , and solid angle, Ω . The observed flux from the solid angle $\Delta\Omega$ subtended by a hot spot element is described as a function dependent upon a specific energy, E , and the pulsar's rotational phase, ϕ .

$$\Delta F(E, \phi) = I(E, \phi) \Delta\Omega \quad (5.1)$$

Intensity is brightness of the object, i.e., the power emitted per unit area for each solid angle in the observer's sky. At a specific energy, it is transformed from the observer's frame to the co-rotating frame on the stellar surface through the

conservation of phase space density stated in Liouville's Theorem ($I/\nu^3 = I'/\nu'^3$) (Beloborodov, 2002). Since emission is not assumed to be isotropic, the observed intensity can be defined in terms of the energy and emergent intensity from the stellar surface, $I'(E', \phi')$ as

$$I(E, \phi) = \left(\frac{E}{E'} \right)^3 I'(E', \phi'). \quad (5.2)$$

Here, ϕ' represents the emission angle on the stellar surface, α , for photons observed corresponding to the rotational phase. The flux equation can now be written as

$$\Delta F(E, \phi) = \left(\frac{E}{E'} \right)^3 I'(E', \phi') \Delta \Omega$$

I use the Schwarzschild metric to describe the curved space-time caused by the neutron star's strong gravitational force. The Schwarzschild metric describes the gravitational field outside of a spherical mass, is independent of time, and is spherically symmetric (Hartle, 2003). Energy conversion is defined by

$$E' = \frac{E}{\delta \sqrt{1 - \frac{R_S}{R}}} \quad (5.3)$$

where R_S is the Schwarzschild radius ($R_S = 2GM/c^2$) of the pulsar, R is the real radius, and δ is the rotational Doppler factor (Equation 5.3 is from Section 3 of Poutanen and Gierliński, 2003). The term R/R_S is the emission radius in Schwarzschild

units (Beloborodov, 2002). At this point, the flux equation is

$$\Delta F(E, \phi) = \delta(\phi)^3 \left(1 - \frac{R_S}{R}\right)^{3/2} I'(E', \phi') \Delta\Omega$$

Variable ϕ' can also be written as $\alpha(\phi)$. The light bending formula for the transformation was given by Equation 1 in Beloborodov (2002) and Equation 1 in Bogdanov et al. (2007)²² under the assumption that $R \geq 2R_S$. Variable α is defined in Figure 5.3 as the angle between direction photon emission and the surface normal.

The solid angle of the emission spot as seen by the observer is defined by Figure 5.3 and

$$\Delta\Omega = \frac{\Delta S' \cos \alpha(\phi)}{D^2} \quad (5.4)$$

where S' is the emission spot size and D is the distance of from the observer (Equation 5.4 is Equation 8 in Poutanen and Gierliński, 2003).

The final observed flux equation is

$$\Delta F(E, \phi) = \delta(\phi)^3 \left(1 - \frac{R_S}{R}\right)^{\frac{3}{2}} I'(E', \phi') \frac{\Delta S' \cos \alpha(\phi)}{D^2}, \quad (5.5)$$

which I use in the following sections to calculate energy-dependent light curves and phase-dependent energy spectra. Equation 5.5 is related to Equation 19 from Viironen and Poutanen (2004), Equation 18 from Zavlin et al. (1996), Equation A3 from Beloborodov (2002), Equations 13 and 16 from Poutanen and Gierliński (2003), and Equation 10 from Bogdanov et al. (2007).

²²Bogdanov et al. (2007)'s Equation 1 defines Ψ in Figure 5.3 as the angle between the observer and the polar hot spot through $\cos \Psi = \sin i \sin \theta \cos \phi + \cos i \cos \theta$.

5.4.2 Generating the Emergent Intensity

I used McPHAC (Haakonsen et al., 2012) to create the emergent intensity from the pulsar’s emission spots ($I'(E', \phi')$, or rather $I'(E', \alpha(\phi))$). The code requires input parameters to set up the initial structure. Based on the literature, I selected an appropriate effective temperature, T_{eff} , and surface gravity, g_s , to simulate PSR J0437-4715’s emission. The effective temperature was taken from Bogdanov (2013)’s observations of this pulsar using data from XMM-Newton. I chose to use the effective temperatures he found by modeling the energy spectra with two atmospheric models (each hotspot), one blackbody (the rest of the pulsar), and a power law (see Table 5.2 for the full model). His results state the temperatures of one hotspot as $T_{\text{eff,spot1}} = 2.21^{+0.08}_{-0.10} \times 10^6$ K and the other as $T_{\text{eff,spot2}} = 0.57^{+0.05}_{-0.08} \times 10^6$ K (Bogdanov, 2013). The surface gravity was calculated by

$$g_s = \frac{GM}{R^2 g_r} = \frac{1.33 \times 10^2 M}{M_{\odot} R_{\text{km}}^2 g_r} \text{ cm s}^{-1} \quad (5.6)$$

and

$$g_r = \sqrt{1 - \frac{2GM}{c^2 R}} = \sqrt{1 - \frac{2.952M}{M_{\odot} R_{\text{km}}}} \quad (5.7)$$

using the assumed neutron star mass and radius from the same paper (Bogdanov, 2013), $M = 1.76M_{\odot}$, $R=13.5$ km. In Equations 5.6 and 5.7 (Zavlin et al., 1996), g_s is surface gravity, G is the gravitational constant, M is the neutron star mass, M_{\odot} is solar mass, R is the neutron star radius, g_r is the gravitational parameter based on the Schwarzschild metric for the general theory of relativity, and c is the speed

of light. The source-specific input parameters I used with McPHAC are displayed in Table 5.5. The complete list of variables can be found in Appendix E.3 in the bash script I used for running the code.

McPHAC Variable	Input Value	Description
<i>Physical Parameters</i>		
LOGTEFF (hot spot 1)	6.34	log of effective temperature $2.21 \times 10^6 \text{ K} = 10^{6.34} \text{ K}$
LOGTEFF (hot spot 2)	5.76	log of effective temperature $0.57 \times 10^6 \text{ K} = 10^{5.76} \text{ K}$
GSURFACE	1.85×10^{14}	Surface gravity cm s^{-2}

Table 5.5: Source-specific McPHAC input parameters for PSR J0437-4715 emergent spectra simulation. These values are for the emergent radiation from the hot spots only. See Table 5.2 for a complete list of the parameters.

5.4.3 Generating the Observed Spectra

Using Equation 5.5 from the derivation in Section 5.4.1 and the emergent intensity calculated by methods described in Section 5.4.2, I created spectra for 20 phase bins with a code I developed using Python. The code is in Appendix E.2.

5.5 Creating a Virtual NICER

A virtual NICER is needed to observe the virtual pulsar. Finding the optimal pointing direction for observing PSR J0437-4715, while considering a bright source (AGN RX J0437.4-2711) within the field of view, is solved by finding which off-axis angle the XTI should point to in order to maximize the pulsar to AGN count rate ratio. At each pointing angle, the instrument will respond differently. Count rate per energy band within the SDD aperture will differ dramatically, changing the ob-

served spectra. For each pointing direction, the sources will be observed at different offset angles, $4.2'$ apart. In addition, the whole XTI and each of the 56 XRCs have uncertainties in their pointing directions, described by the pointing uncertainty of allowed error (Section 5.5.2). Each pointing angle consists of 56 simulated XRC pointing directions. Since two sources are responsible for the total photon flux²³, I considered two response functions for each simulation. These response functions were generated separately by summing the 56 XRCs' effective areas based on the XRCs' individual pointing directions (see Section 5.5.3). I used the ray-trace simulator, CONSIM (Section 5.5.1), to calculate the effective areas and the FTOOLS (Blackburn, 1995; Irby B. K., 2017) command, *genrsp* (Section 5.5.3, step 4), to make the response functions. In the simulation, I investigated 22 offset pointing angles ranging from $-1.0'$ to $3.5'$ for 20 simulated instances of the NICER XTI and XRC pointing directions.²⁴

In the following sections, I explain the NICER XRC ray-trace code, the pointing uncertainty of the XTI and XRCs, and my method for producing XRC response functions. The goal here is to simulate NICER's performance with the possibility of the instruments being misaligned in accordance with the pointing uncertainties from the mission proposal.

²³The X-ray sky background is neglected. See Appendix E.1.

²⁴The offset pointing angle of NICER is described by the following coordinate system, where A = offset pointing angle: $A = 0'$ is on-axis with the pulsar, $A < 0'$ is the direction towards the AGN, and $A > 0'$ is the direction opposite the AGN.

5.5.1 Ray-tracing the XRCs using CONSIM with an Adjusted Scattering Function

CONSIM (short for concentrator simulator) is the FORTRAN-based ray-tracing program for the NICER XRCs. It was adapted by the NICER team from the ASCA ray-tracing code (Tsusaka et al., 1995). The core of this program uses a Monte-Carlo simulation of incident events onto the concentrators' incident surface plane and traces each photon through its journey to the detector plane. Along the way, the photon may interact with the reflector's surface where it is then absorbed or reflected. If reflected, the photon's exit angle may be affected by scattering. The scattering function simulates small impurities on the reflecting surface. Without this effect, all photons would focus exactly on the focal point for an on-axis measurement. It is easy to see this is not true in reality from the X-ray image in Figure 4.14. The user may specify the number of photons to be traced to the focal plane and whether or not these events must be within an aperture. Other inputs the user may choose are the incident angle of the photons (i.e., the off-axis angle of the source) and the energy range. Examples can be seen in Appendix E.4 in *raytraceEA.bash*. All these properties affect the total run time of the CONSIM, which can range from a few seconds to over ~ 5 –10 min. A FITS (Flexible Image Transport System) (Blackburn, 1995) file is produced with information on every photon that was traced, including if it was focused onto the detector plane or if it experienced some obstruction along the way.

I used CONSIM to produce effective area curves at different off-axis pointing

angles for both sources in order to construct the instrument response (Section 5.5.3). I adapted CONSIM to my analysis of the flight NICER XRCs by adjusting the scattering function parameters as follows below.

The scattering function contains three components, a Gaussian, a Lorentzian, and an exponential curve (Tsusaka et al., 1995). I systematically adjusted the parameters in the *consim.par* file (program’s input parameters) over several runs of CONSIM to produce an EEF with minimal deviation from the XRC calibrations test results. I compared every ray-trace output with the average measured EEF of all the XRCs that met requirements (see Chapter 4). The calibration based EEF is compared to the ray-traced one with optimal parameters in Figure 5.4. The scattering parameters I found, and used in this work, are reported in Table 5.6. While it is possible to fit other sets of parameters, I chose to use parameters similar to the original, which were provided in the ASCA ray-tracing code (Tsusaka et al., 1995).

Parameter	Fit to Measurements
Gaussian Normalization	3.12
Gaussian Sigma	1.55
Lorentzian Normalization	0.001
Lorentzian Sigma	10.0
Exponential Normalization	0.4
Exponential Sigma	0.4

Table 5.6: Ray-tracing scattering function parameters fitted to actual XRC measurements.

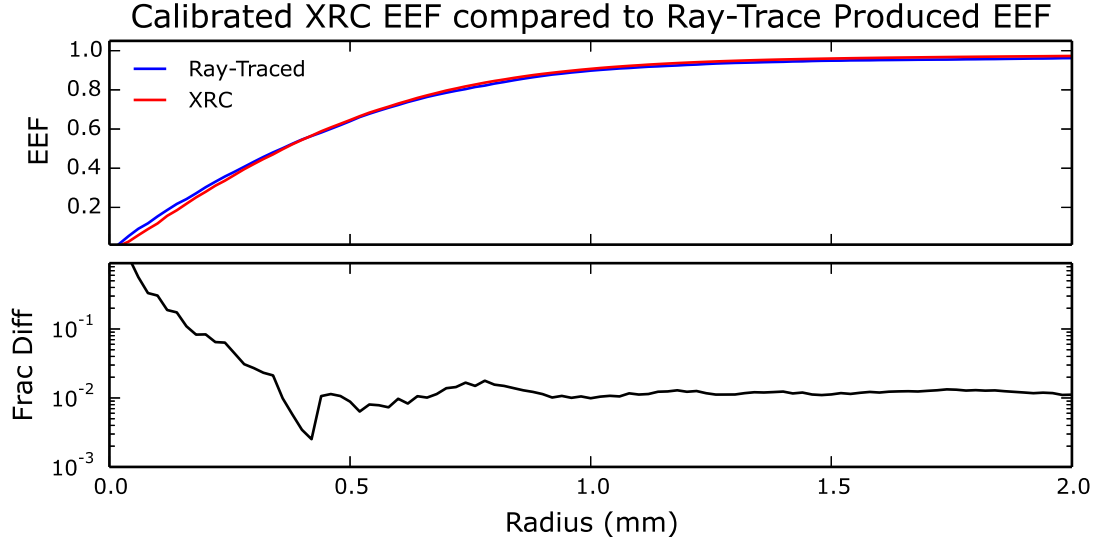


Figure 5.4: Comparison of the EEF from ray-tracing using scattering function fit parameters from Table 5.6 (blue line) to the average EEF from XRC calibration data (red line).

5.5.2 Pointing Uncertainty

As with all instruments, there is a possibility for small alignment errors, which may be caused by mechanical pointing errors or alignment changes during launch and deployment. This section explains the reasonable ranges of uncertainty anticipated in NICER’s optical system.

The true direction of each XRC and the XTI as a whole may deviate from the nominal pointing direction according to a normal distribution as described in Table 5.7 (Gendreau, 2012; Arzoumanian, 2014, priv. comm.).²⁵ I produced simulated pointing angles for each instance of NICER using numpy’s *numpy.random.normal()* function in Python with my code *makevectors.py* of Appendix E.4. This script was called within a bash file as part of the steps in constructing the responses. These

²⁵More about the NICER pointing system can be found in Jason Budinoff et al. (2016) and Gendreau, K.C. and Arzoumanian, Z. and Adkins, P.W. and others (2016).

Pointing Uncertainty Requirement

	3σ
Boresight (XTI)	85''
XRC	35''

Table 5.7: This shows the required maximum allowed pointing uncertainty for the boresight (XTI pointing direction) and the XRCs based on a normal distribution. These values reflect those reported in NICER’s mission proposal (Gendreau, 2012).

steps are described next in Section 5.5.3. Figure 5.5 shows an example of the XRC pointing directions generated for a single instance of NICER. Figure 5.6 shows the randomly generated XTI boresight pointing directions for 20 different possible instances of NICER. Figure 5.7 shows an example of pointing vectors calculated for

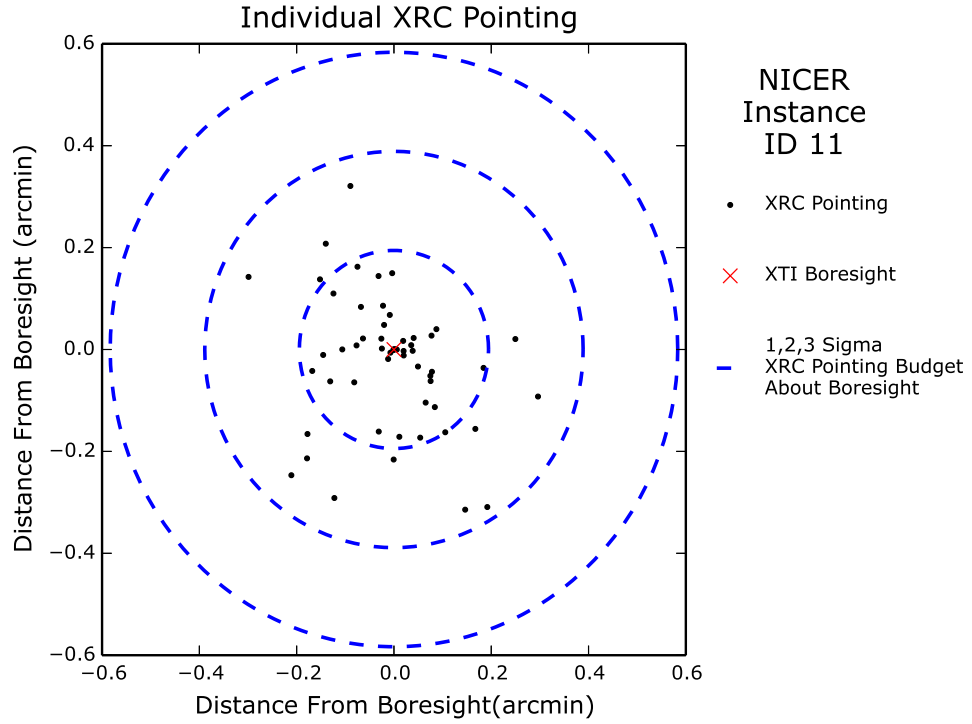


Figure 5.5: Randomly generated error on XRC alignment based on a pointing uncertainty requirement of 35'' radius at 3σ .

an instance of a virtual NICER where the XTI was pointed at three different locations, one is on-axis towards the pulsar and two are pointed 2' away from pulsar,

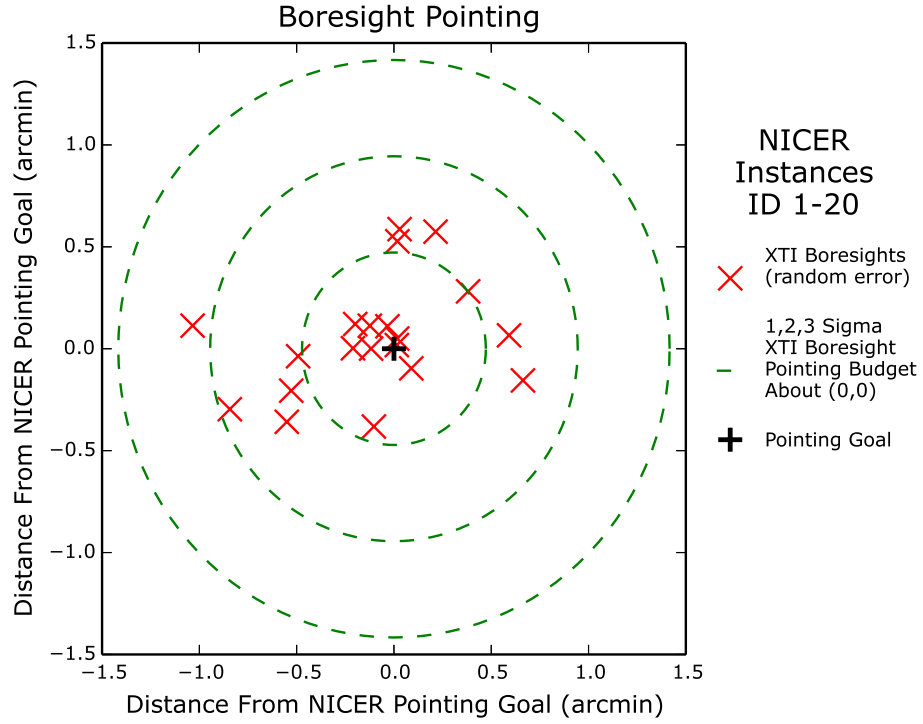


Figure 5.6: Randomly generated error on XTI boresight based on a pointing uncertainty requirement of $85''$ radius at 3σ .

one in and one against the direction to the AGN.

5.5.3 Constructing the Response Used in the Simulations

A response file is a calibration file that describes how observed data are affected by the instrument. The response files I used are composed of three components: the XRC effective area (*efffil*), the detector efficiency (*detfil*), and the filter transmission (*filfil*). These components are described in step 4 below.

Step 1 The first thing I did was create effective area curves in CONSIM. I ran CONSIM to ray-trace the XRCs at several different off-axis angles using the scattering function parameters described in Section 5.5.1. I required a specified

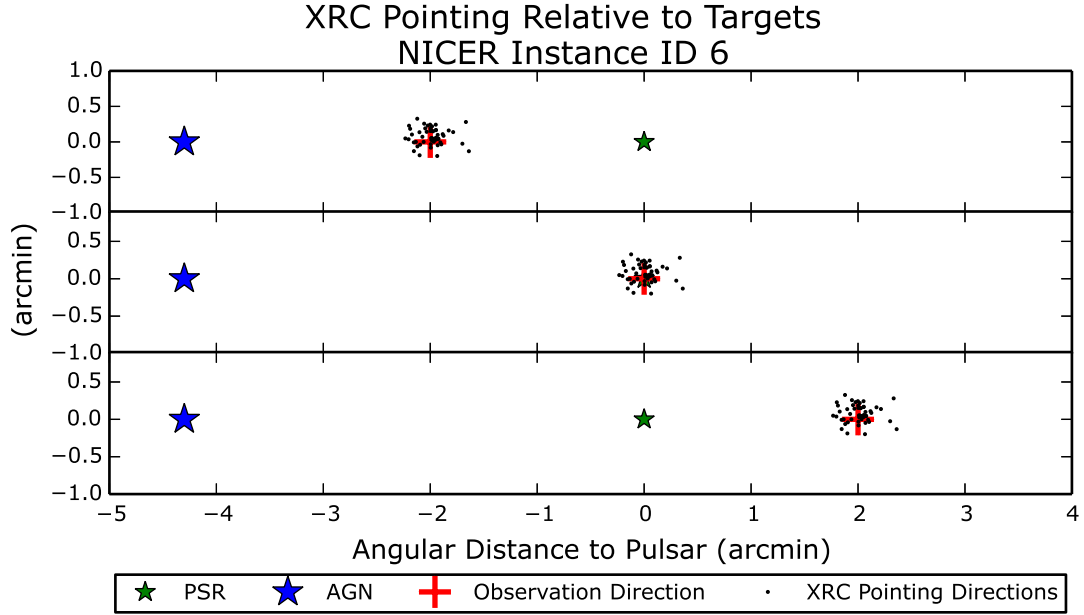


Figure 5.7: Example of pointing vectors for an instance of a Virtual NICER. The blue star is the AGN and the green star is the pulsar. The red x is the observation direction. Each black dot is the pointing direction of one XRC, considering boresight and XRC alignment errors. (top) XTI pointed $2'$ away from pulsar, towards AGN, (middle) XTI pointed at pulsar, (bottom) XTI pointed $2'$ away from pulsar, opposite of AGN.

number of photons to reach the focal plane within a 2 mm aperture. The smaller the effective area, the longer it will take CONSIM to run in order to get the same number of photons collected on the focal plane. Higher energies and larger off-axis angles took significantly longer to calculate within CONSIM. See the plot on the right in Figure 5.8. I ran the program with several sets of parameters (different energy ranges and number of events) and joined the resultant effective area files afterwards. This saved computation time for angles that took a lot of time to calculate with CONSIM. The sets of parameters are described within the code *raytraceEA.bash* on pages 185–187. I calculated the effective area for a total of 91 off-axis angles ranging from $0.0'$ to $9.0'$ in increments of $0.1'$ in preparation of generating the responses.

Offset pointing angles $\leq 0'$, later in the simulation, use the response corresponding to their offset angle magnitude.

Step 2 Next I generated efficiency files for the effective area curves that were needed to generate response functions with command *genrsp* (see step 4 for an explanation of *genrsp*). In script *GenerateEAEfficiencyFiles.bash* on page 187, I used *fdump* to get the effective area arrays from CONSIM's output FITS files. Since CONSIM was run in different sets (energy ranges and photons numbers mentioned in step 1 above), I had to combine the effective area files using *attachEAarr.py* (page 188), as mentioned in step 1.

Step 3 Next I made unique NICERs each with a unique set of pointing vectors. File *CreateUniqueNICER.bash* (page 188) provided a wrapper for *makevectors.py* (page 189). First a boresight error is chosen based on the pointing budget (Table 5.7). This is the distance from the nominal pointing direction to the XTI's actual pointing direction. Then a random angle from 0° – 360° is selected as the direction, in polar coordinates, for the boresight error. This is done an additional 56 times for the XRCs using their pointing budget numbers. The vectors are added and saved to a text file in (x,y) coordinates.

Step 4 Finally, I made the response files using *genrsp*, a generic spectral response generator²⁶. A Gaussian energy redistribution matrix is assumed for which I set FWHM to 0.01 keV. This is an acceptable first order assumption for the NICER detectors. For each response, I chose a linear energy grid with 10,000 energy bins.

²⁶The help file for the *genrsp* command can be found at <https://heasarc.gsfc.nasa.gov/ftools/caldb/help/genrsp.txt>

The *genrsp* command requires an optical efficiency file (*efffil*), a filter transmission file (*filfil*), and a detector efficiency file (*detfil*). The latter two I got from the NICER team and they are plotted in Figure 5.8 along with the effective area at different off-axis angles, used as the basis of the optical efficiency file for one XRC. See page 190 in Appendix E.4, file *PointNICER.bash*, for the call of the *genrsp* command.

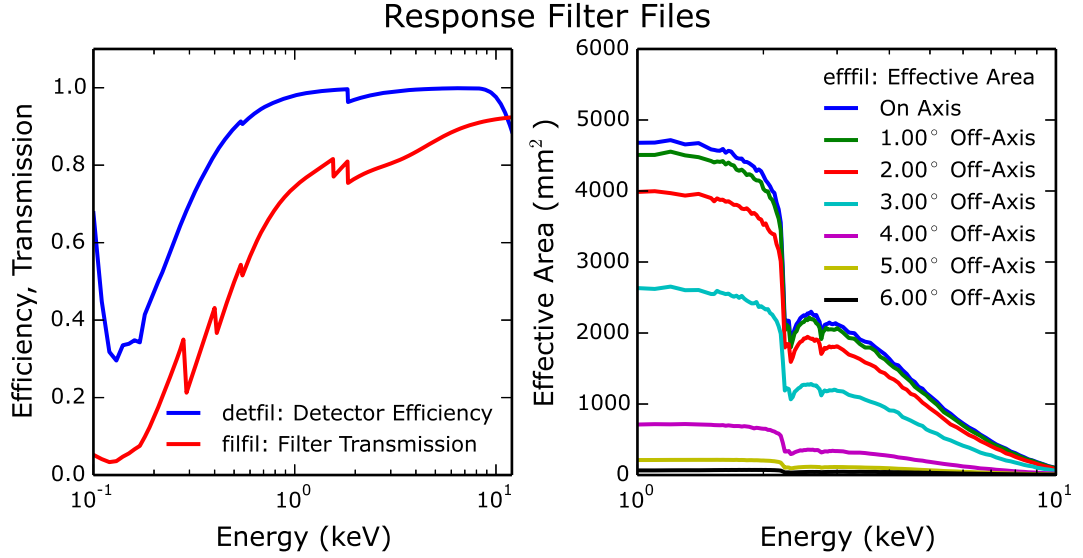


Figure 5.8: Filter files used to generate NICER responses. Right: The effective area of a single XRC for different off-axis sources. This is the basis of the optical efficiency *efffil* files. Left: The NICER filter transmission (*filfil*, red) and detector efficiency (*detfil*, blue) are used along with an optical efficiency (*efffil*) to generate response functions using *genrsp*.

The optical efficiency files are the ones I generated based on data as described in the previous steps. Running *PointNICER.bash* generates two response functions for a single pointing angle in the sky, one for the pulsar and one for the AGN. The angular distance to each source from each XRC is found with *sourcedist.py* and saved to a text file. The saved file is loaded into *addefffiles.py* where the efficiency files for all XRCs are summed together to create a single efficiency file for the whole XTI system with the 56 XRCs pointing in different directions. This results in one

efffil file for each source. Those are then read into *genrsp* to produce the responses. *PointNICER.bash* is placed in a wrapper script to make the simulated observed spectra in Section 5.6. Figure 5.9 presents the response functions produced by this method for example pointing directions for sources at the pulsar and AGN locations.

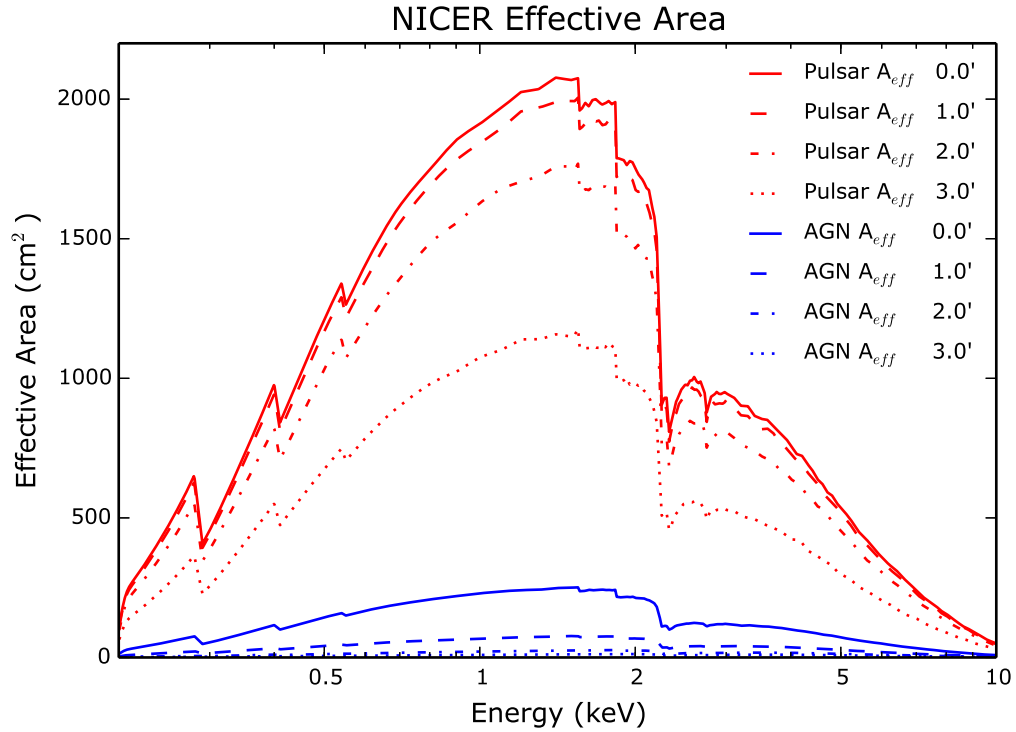


Figure 5.9: Responses as effective area curves for observing the pulsar and AGN. It is shown when NICER is pointing directly at the pulsar as well as 1', 2', and 3' off-axis, away from the AGN.

Figure 5.10 provides a flow chart that outlines how to use the codes and scripts provided in Appendix E.4 that allows on to create virtual NICERs and generate response functions needed for this simulations in the rest of this chapter.

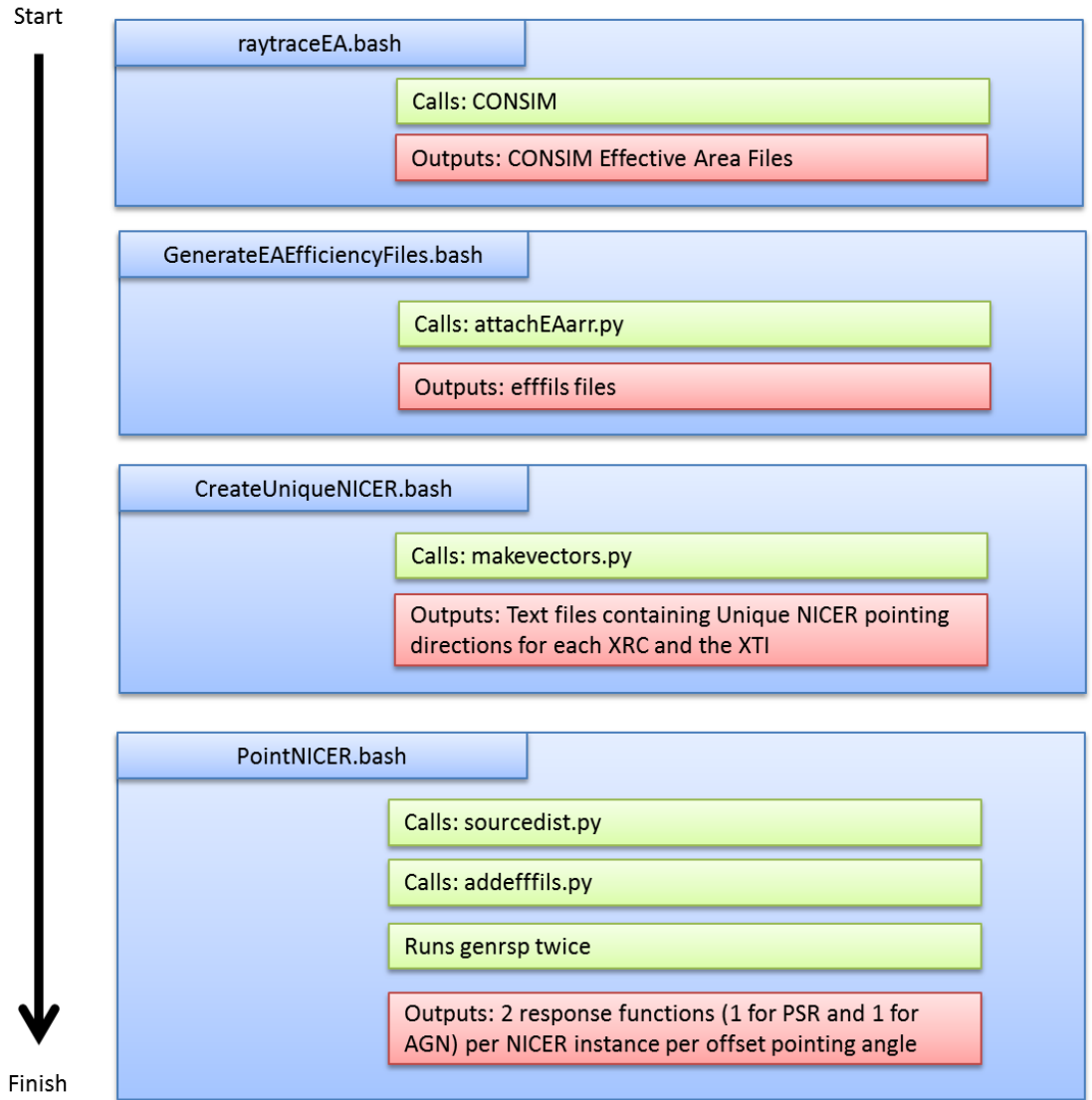


Figure 5.10: This flow chart shows the process in which to run the files in Appendix E.4 in order to produce virtual NICERs and generate the response functions needed in the simulations.

5.6 Tackling the Challenging Observation Simulation

In Section 5.4, I created a phase dependent emergent spectral model of PSR J0437-4715 based on the knowledge from previous works described in Section 5.3. I also have a model for the AGN. Then in Section 5.5, I optimized and ran a ray-tracing program simulating the NICER XRCs and applied a method to combine

the optics responses with the rest of the instrument components' specifications to generate response functions for observations at several pointing angles relative to the sources.

With XSPEC the response functions and spectral models can be used to simulate an observed spectrum with the *fakeit* command. Given some length of exposure, this command uses a model defined by the user and folds it with the instrument response to create a simulated spectrum as expected to be observed by the detector. The spectrum can be saved as a pulse height amplitude (PHA) file to use future analysis of a potential observation²⁷.

For each NICER pointing angle I wished to analyze, I generated a separate response function for the pulsar and AGN. To consider possible misalignments of each XRC, there were 56 angle-dependent effective area files summed together in order to create each response file (described in Section 5.5.3). These were used in XSPEC to make a PHA files for both the pulsar and AGN.

For every instance of NICER (20 total), I simulated 20 rotational phases. Twenty-one new PHAs (1 for the AGN for a 10^6 s exposure and 20 for the pulsar for a $\frac{1}{20} \times 10^6$ s exposure each) were generated for each offset pointing angle (22 total, ranging from $-1.0'$ to $3.5'$) tested.

I used XSPEC to calculate the number of counts and count rate from each spectrum for several different energy ranges. To do this, I wrote a script to loop through loading each PHA into XSPEC and output a file from which the counts

²⁷Here, a PHA file is a simulated data file in FITS format. The pulse height file introduced in Section 4.2.2, which also described an energy spectrum, was in ASCII format.

were extracted. Figure 5.11 is an example of pulsar spectra at 4 different rational phases and the AGN.

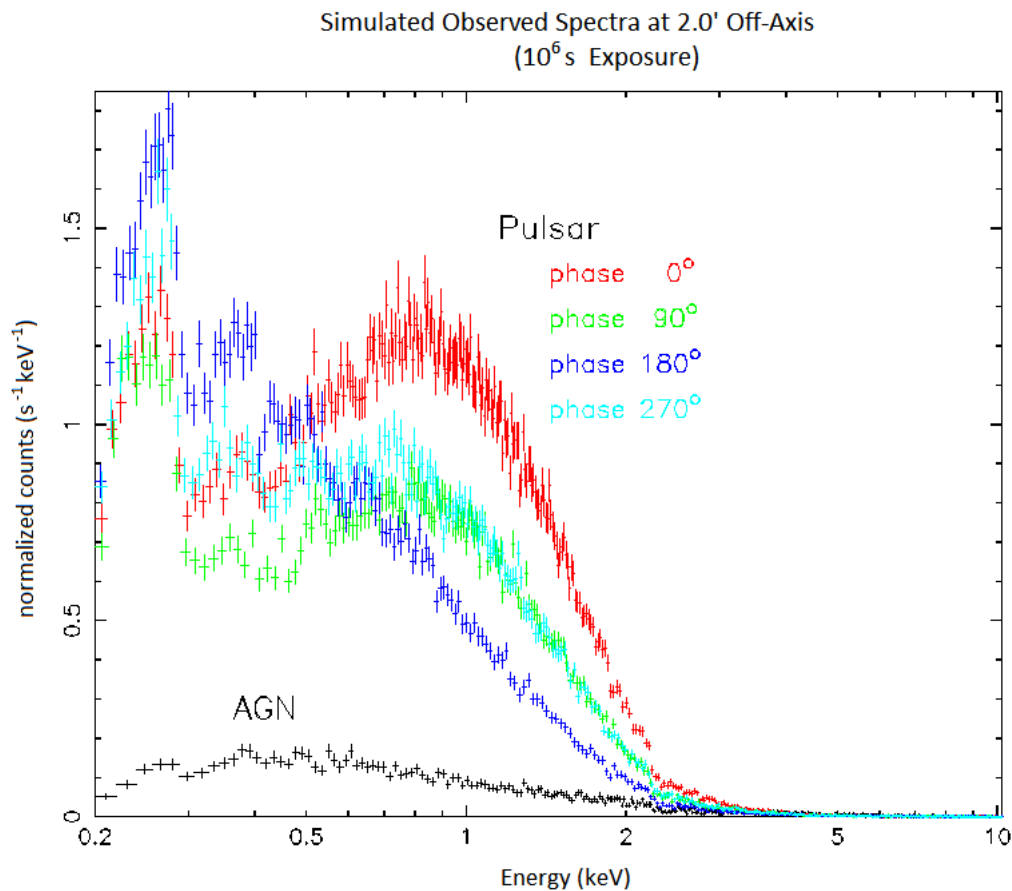


Figure 5.11: Examples of simulated observations showing pulsar spectra at 4 different rational phases and the AGN spectrum. NICER is pointed off-axis by $2.0'$ opposite the AGN direction.

These data were used to determine the ratio of pulsar to AGN counts detected at different pointing angles and to generate light curves (pulse profiles) and ultimately estimate the pulsar signal based on a simulated NICER observation. This is described in more detail in Section 5.7. Figure 5.12 outlines the steps of my simulations and subsequent conclusions.

Simulation Flowchart: Estimate Pulsar Signal Calculation

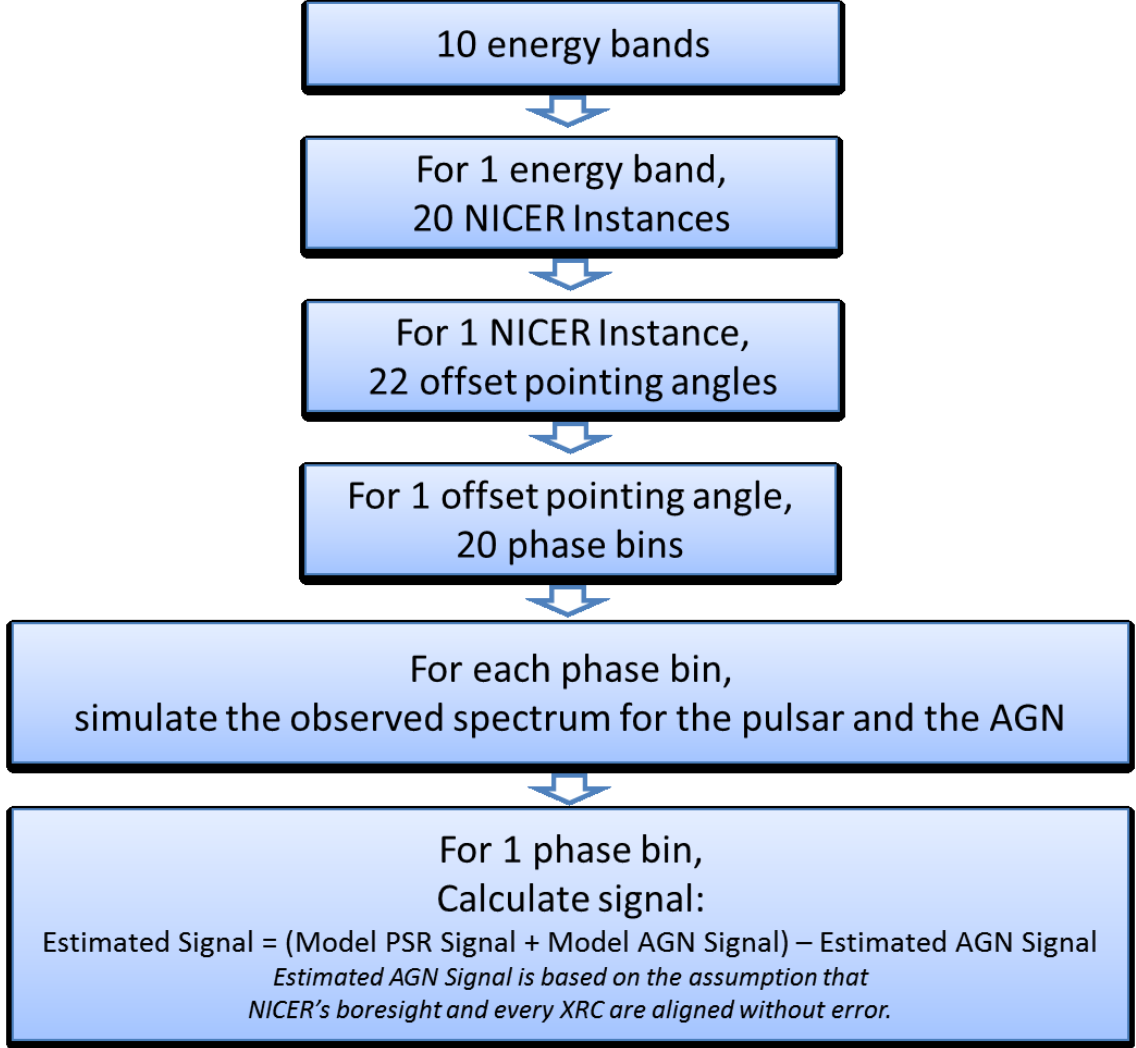


Figure 5.12: This flowchart outlines the steps in the simulation for estimating the pulsar signal. (The preliminary study (Section 5.6.1) explores a subset of this parameter space, as described in the text.)

5.6.1 Preliminary Study

In this preliminary study I assumed a pulsar with the double blackbody energy spectrum with parameters described in Table 5.1 of Section 5.3 and an AGN with

the absorbed power law spectrum of Table 5.3 from the same section. I used three of the 20 NICER instances I made to conduct this study, instance #03, #13, and #17. Instance #03 had a boresight closest to the actual pointing direction with an offset of only 0.023' off. Instances #13 and #17 had the worst boresight pointing errors of 1.003' and -1.041', respectively. Figure 5.13 shows the XRC pointing directions of each instance in this study relative to the goal pointing direction of NICER.

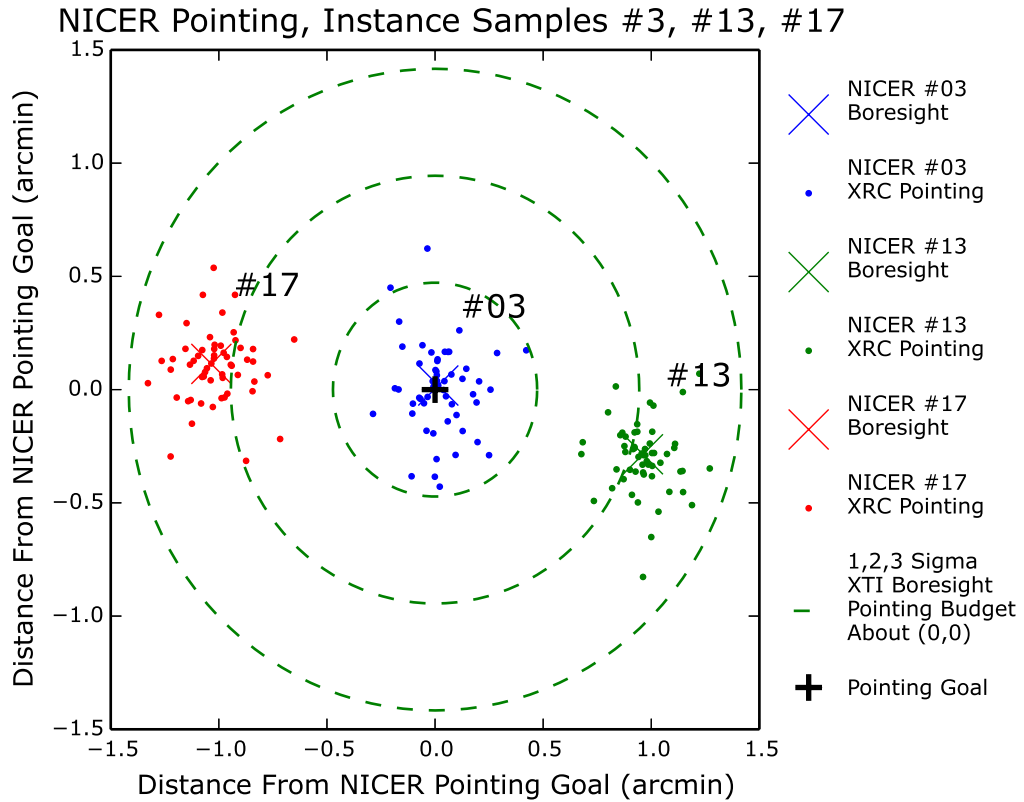


Figure 5.13: Pointing directions for the boresight and XRCs of 3 NICER instances.

I used XSPEC's *fakeit* function to make a fake observed spectrum for each of these instances at offset pointing angles ranging from $-4.5'$ to $3.5'^{28}$. Negative

²⁸While the preliminary study had lower limit of $-4.5'$, the main study only went down to $-1.0'$. The simulation in this preliminary study required less computation power because a simpler model was assumed for the pulsar, allowing more offset pointing angles to be tested.

pointing angles used responses from their positive counter part (Section 5.5.3). I then conducted a count rate analysis of these spectra. I studied 10^6 s exposures in the energy band 0.2–8.0 keV and subsets thereof. Figure 5.14 shows a plot of the net count rate for the PSR J0437-4715 and AGN RX J0437.4-4711 as a function of the offset pointing angle. Figure 5.15 shows the ratio count rate plot. The location of

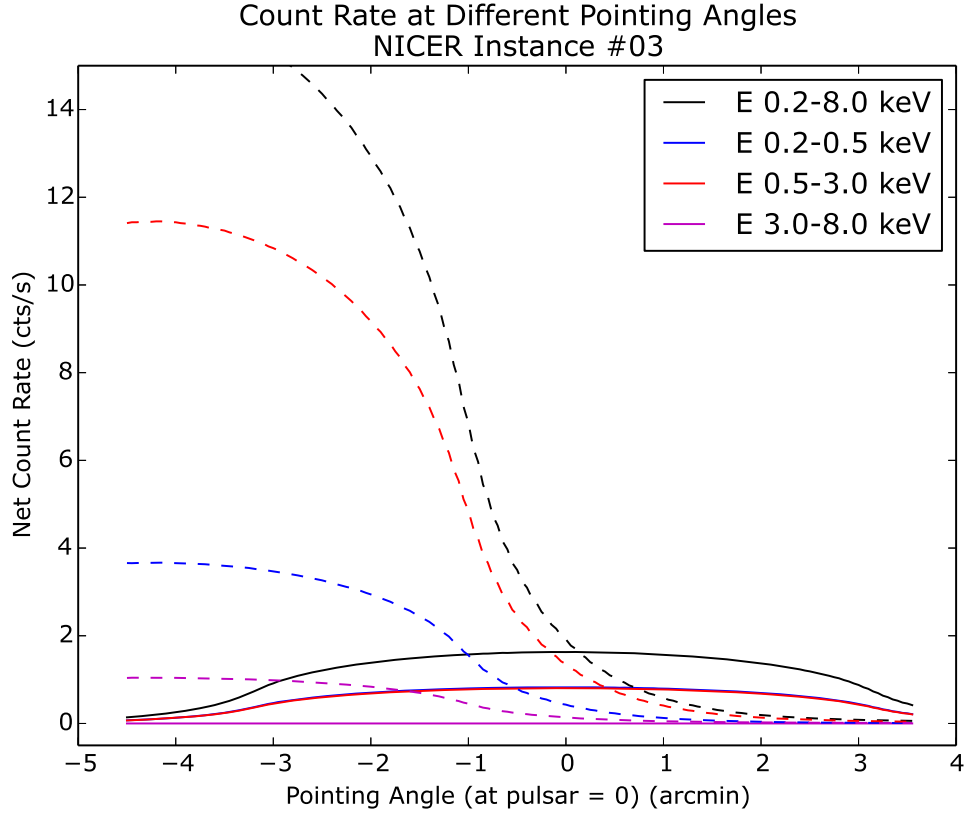


Figure 5.14: *Preliminary study.* Pulsar and AGN count rates for NICER instance #03 at different energy bands and offset pointing angles ranging from $4.5'$ in the direction of the AGN to $3.5'$ opposite the AGN direction (negative values are towards the AGN). The solid lines represent the pulsar's count rates and the dashed lines represent the AGN. Black is 0.2–8.0 keV (the full range), blue is 0.2–0.5 keV, red is 0.5–3.0 keV, and magenta is 3.0–8.0 keV. An absorbed double blackbody plus power law model (Table 5.1) was assumed for the pulsar in this plot. The AGN model is described by an absorbed power law (Table 5.3).

the count rate ratio peaks suggest an optimal pointing angle between $2.5'$ and $3.0'$, if

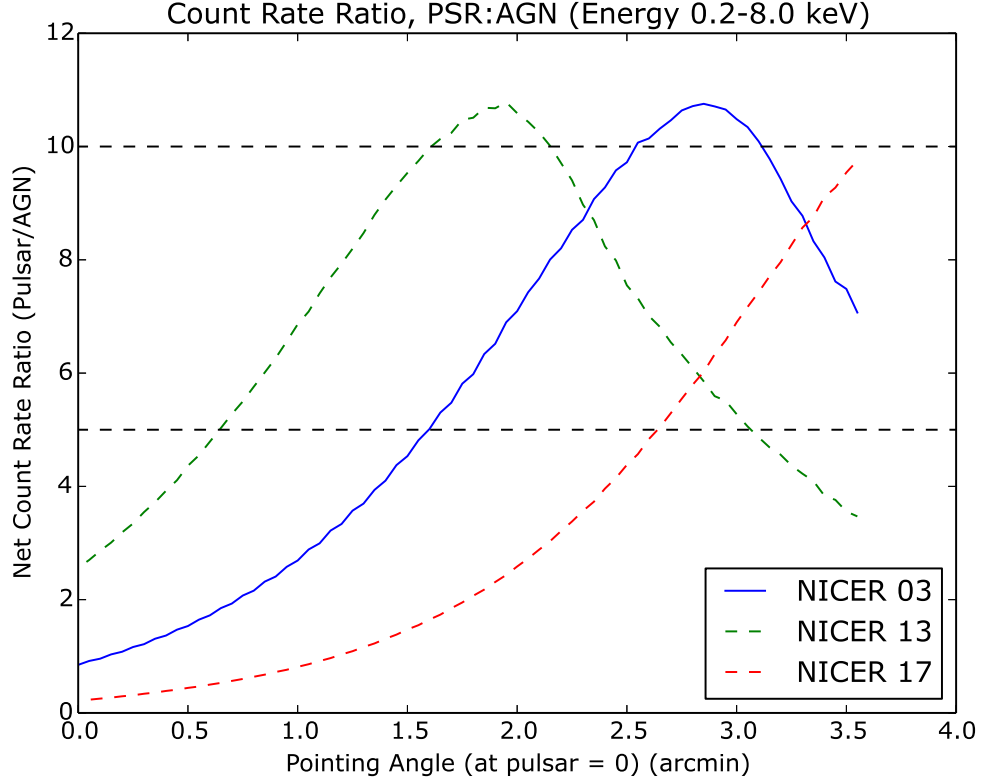


Figure 5.15: *Preliminary study*. Pulsar to AGN count rate ratio for NICER instances #03, #13, and #17 at different pointing angles. Dashed black lines are drawn at the 5 and 10 times pulsar to AGN count rate levels. Spectral models described in Figure 5.14 and in the text.

the boresight is near optimal. NICER instance #03, being the closest to ideal XTI pointing direction, was examined more closely by calculating the ratios at different energy levels as shown in Figure 5.16. This first order examination found that the peak count rate ratio may suggest an energy dependence since a small deviation of the peak location in different energy bands is observed.

Conclusions of Preliminary Study When choosing the optimal pointing angle, one must chose an angle such that the net count rate ratio of pulsar to AGN accommodates large errors in boresight alignment. It is also worth to look at the

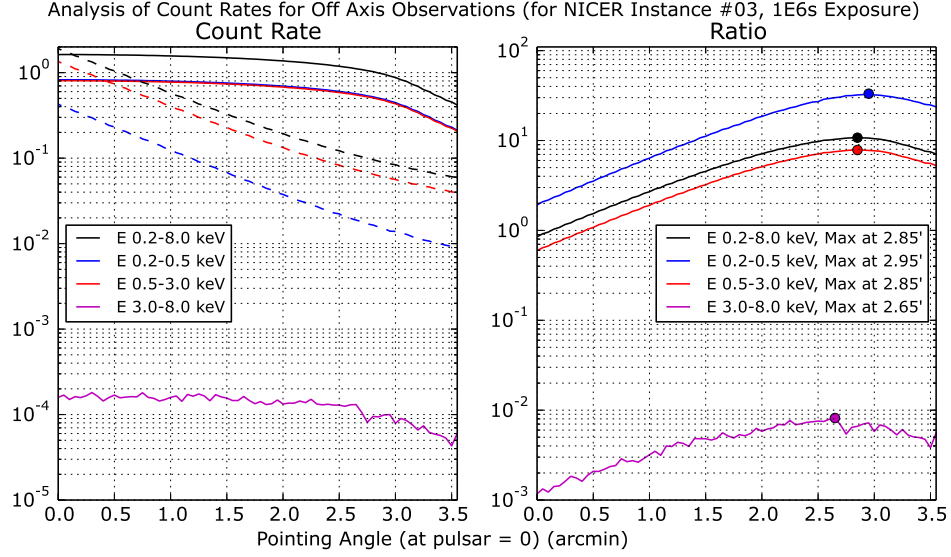


Figure 5.16: *Preliminary study.* Count rate analysis using NICER instance #3. There is evidence that the count rate ratio between the sources suggest energy dependence, as seen by the different locations of the peaks on the plot to the right. (Left) Count rate plot of the AGN and pulsar for different energy bands. This is a zoomed in version of Figure 5.14. (Right) Ratio of pulsar to AGN count rates with maximums labeled.

energy dependence of the ratio's peak as seen minimally in Figure 5.16. When preparing an observation, one should consider the energy spectra hardnesses of all objects in the field of view along with the XRC effective area curves.

5.7 Simulation and Data Analysis

With two sources in the FOV, the best way to observe one target, is to maximize the count rate ratio of target to non-target. In this section, I begin by examining the ideal case in which the XTI and each XRC are perfectly aligned (Section 5.7.1). Then I provide a method to calculate the estimated pulsar contribution from an observation which includes both the pulsar and AGN flux (Section 5.7.2). The method is using the ideal case simulation for extracting the pulsar signal. This

section culminates in simulating observations for the 20 different NICER instances (from Section 5.5) where each has a unique set of misalignment errors (Section 5.7.3). The codes referenced throughout this section are provided in Appendix E.5.

In this simulation, I assume that the X-ray sky background is negligible because the pulsar is $\sim 100\times$ brighter (see Appendix E.1).

5.7.1 Ideal Case

The ideal case was analyzed using the Python code *AnalyzingIdealCase.py*, which can be found on page 192 in the Appendix.

Assuming that the XTI boresight and XRCs' alignments are ideal, an observation of PSR J0437-4715, with the instrument pointed directly at the target results in an estimated total of 3.46×10^6 photons (0.2–10.2 keV) collected over a 1×10^6 s observation. Of these, about 1.50×10^6 would have come from the pulsar. But if NICER were pointed $1'$ off of the pulsar, away from the AGN, then only 2.04×10^6 photons would be collected, but a higher percentage would have emitted from the pulsar (71.6% compared to 43.4%). These values were calculated based on the pulsar and AGN spectral parameters from Tables 5.2 and 5.3, respectively. The spectra were produced in XSPEC using the table model made with *PulsarTableModelMaker.py* (page 178) to simulate the pulsar and using an absorbed power law for the AGN.

Figure 5.17 shows the average count rate per second observed from each source at several pointing directions over a 1×10^6 s observation. At around $0.25'$ off-axis,

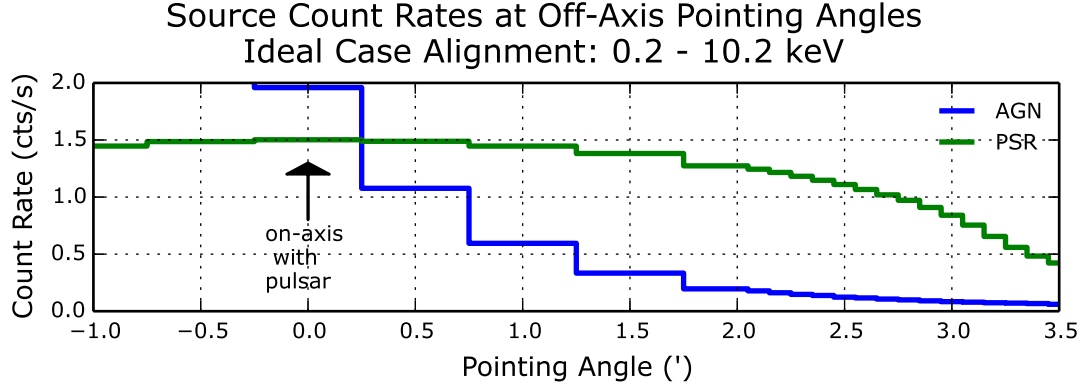


Figure 5.17: Count rate comparison of the AGN and pulsar at different pointing angles in the 0.2–10.2 keV energy range when alignment is assumed to be ideal.

the counts from each source are equal and the pulsar counts dominate at larger off-axis pointing directions.

The count rate ratio (abbreviated as CRR from here on) values across the selected pointing range are displayed in Figure 5.18. The ratio of the pulsar and AGN is signals peak at the 3.0' off-axis pointing direction with the pulsar being 10 times brighter than the AGN. The CRR values are as follows: $CRR(2.5') = 9.02 \pm 0.03$ and $CRR(3.0') = 10.04 \pm 0.04$. The 2.5'–3.0' offset pointing angles range is the range of interest in this investigation. This is related to the ratio of the NICER responses for each source; the maximum ratios are located at approximately the same pointing angles (between 2.7' and 3.0'), see Figure 5.19 comparing the ratio of the instrument efficiency at select energies for observing the pulsar and AGN for the same off axis pointing angles as in Figure 5.18. A few examples of the efficiency at different pointing angles were plotted above in Figure 5.9. Note that the CRR based on number of counts in the simulated data is for the full, 0.2–10.2 keV energy range while the effective area ratios are plotted for specific energies. The latter show

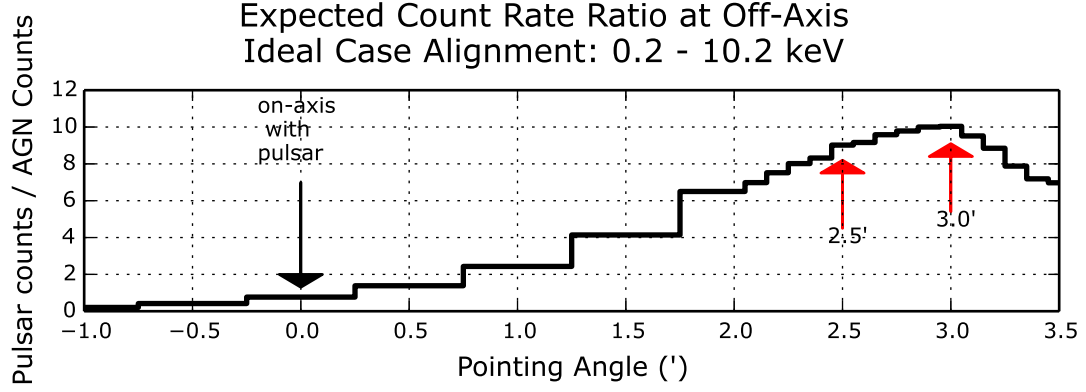


Figure 5.18: Ratio of pulsar to AGN counts observed for the ideal alignment case within the 0.2–10.2 keV energy band across several pointing directions. The peak CRR (count rate ratio) is marked at 3.0' off-axis along with the 2.5' direction. The optimal pointing direction was found to be between these when instrument misalignment is assumed in Section 5.7.3.

a peak shift; see Figure 5.19. Based on the effective area ratios, I suggest that an observation requiring lower energy data should have the XTI pointed further off-axis than an observation needing higher energies.

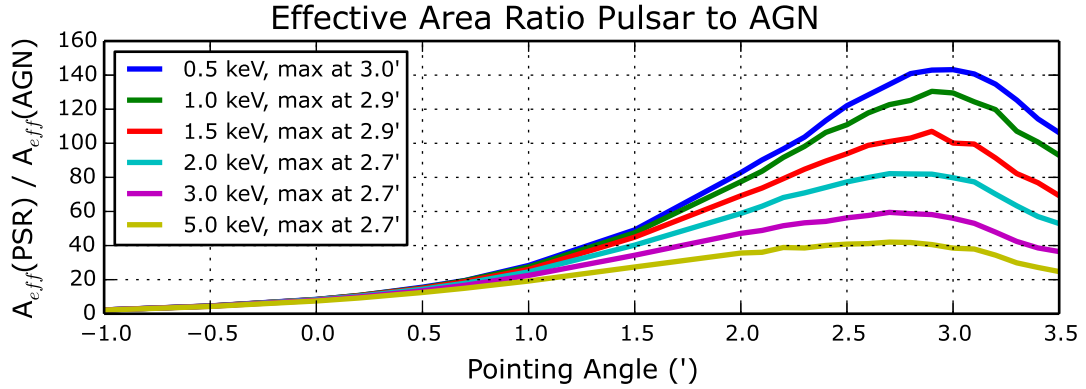


Figure 5.19: The XRC effective area response ratio between the two sources is compared at several energies and across a range of pointing directions. An observation at a lower energy should have the XTI pointed further off axis than needed for the higher energies due to energy dependent XRC response.

Next, I looked at the light curve (pulse profile, Figure 5.20) from data collected at the pointing angle which maximizes the CRR, 3.0', to determine the pulsed

fraction and to compare the flux to the one at pointing angle $2.5'$. To produce this figure, data were simulated at each phase over 1 Ms, for each line in the plot. The total count values were divided by the amount of time for each phase bin (50,000 s) in order to get a count rate per phase bin. The pulsed fraction, f_p , is defined as ratio of the number of counts above the minimum to the total number of counts (Zavlin et al., 2002). It can be useful in determining geometrical structure of pulsars (Harding and Muslimov, 1998). In this example, the pulsed fraction for the $3.0'$ observation, $43.7\% \pm 0.4\%$, was comparable to the one for $2.5'$, $43.5\% \pm 0.4\%$. However, the total counts from the $2.5'$ observation was $34.1\% \pm 0.2\%$ larger than from the $3.0'$ observation.

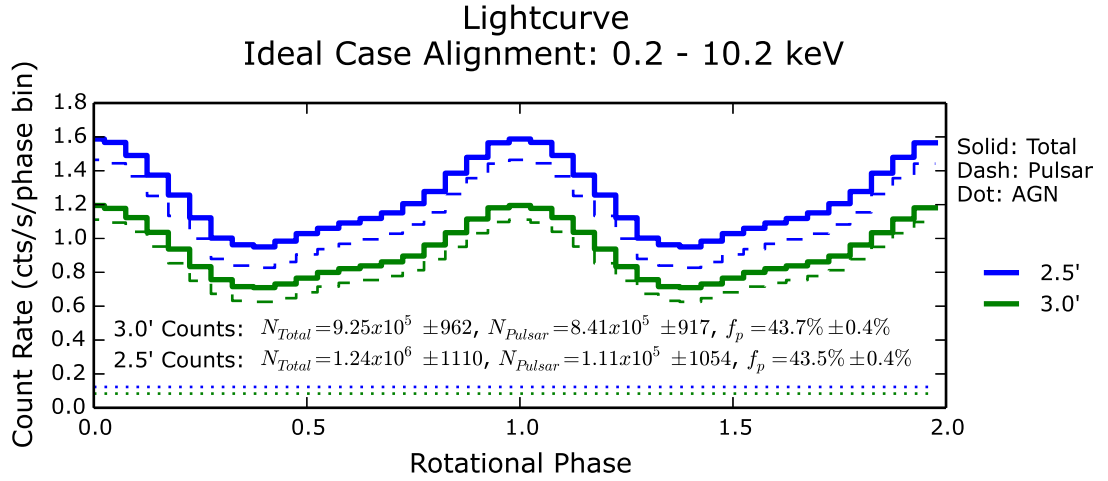


Figure 5.20: The light curves for 2 different simulated pointing angles are compared. This shows the observed light curve (solid line), pulsar contribution (dashed line) and AGN contribution (dotted line).

5.7.2 Estimating the Intrinsic Signal Using the Ideal Case Simulation

The total number of counts, T , is a sum of the pulsar signal, S , and AGN signal, B . The count rate ratio (CRR), R , is also defined by S and B .

$$T = S + B, \quad \text{CRR} = R = \frac{S}{B}$$

The pulsar signal can be solved for in terms of the total counts, T , and the count rate ratio, R .

$$S = \frac{TR}{R + 1}$$

In the “observed” case, T_{obs} is the sum of intrinsic, or synthetic, signals S_{syn} and B_{syn} : $T_{\text{obs}} = S_{\text{syn}} + B_{\text{syn}}$. To the observer, S_{syn} and B_{syn} are unknown. To estimate the intrinsic pulsar signal from an observation, I assumed the CRR from the ideal case, $R_{\text{ideal}} = S_{\text{ideal}}/B_{\text{ideal}}$, where S_{ideal} are the pulsar counts and B_{ideal} are the AGN counts of an ideal alignment scenario. Therefore the synthetic signal can be estimated using the total number of observed counts, T_{obs} , assuming the ideal case count rate ratio R_{ideal} , at any particular phase and/or energy range, by

$$S_{\text{estimated}} = \frac{T_{\text{obs}} R_{\text{ideal}}}{R_{\text{ideal}} + 1} \quad (5.8)$$

5.7.3 Counting with a Misalignment

The codes and scripts referenced in this section are provided in Appendix E.5.

Figure 5.21 outlines the use of the codes.

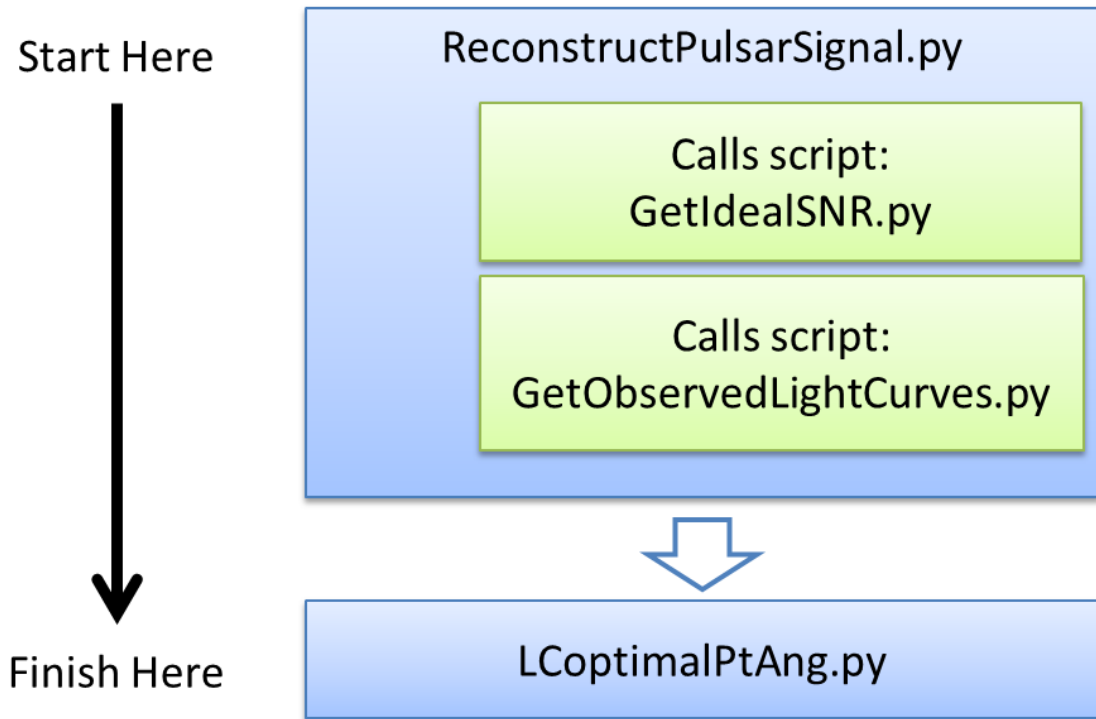


Figure 5.21: This flowchart outlines how I conducted the observation simulations described in Section 5.7.3. The code can be found in Appendix E.5.

By looking at random instances of instrument misalignment configurations, one can determine limits on the accuracy of reconstructing the signal. I will demonstrate this using NICER ID #11. The pointing directions for the boresight and XRCs are shown in Figure 5.22.

For a given instance of NICER, with some pointing misalignment, the pulsar signal was reconstructed from the simulated “observed” data using Equation 5.8 and compared to the known “synthetic” signal generated of the pulsar. I performed this for three energy bands at the 3.0’ offset pointing angle found from the counts ratio analysis of the ideal alignment case above. Figure 5.23 shows this for NICER instance #11 whose alignment configuration is described by Figure 5.22.

I’ve selected a few example phases and one energy band to demonstrate how

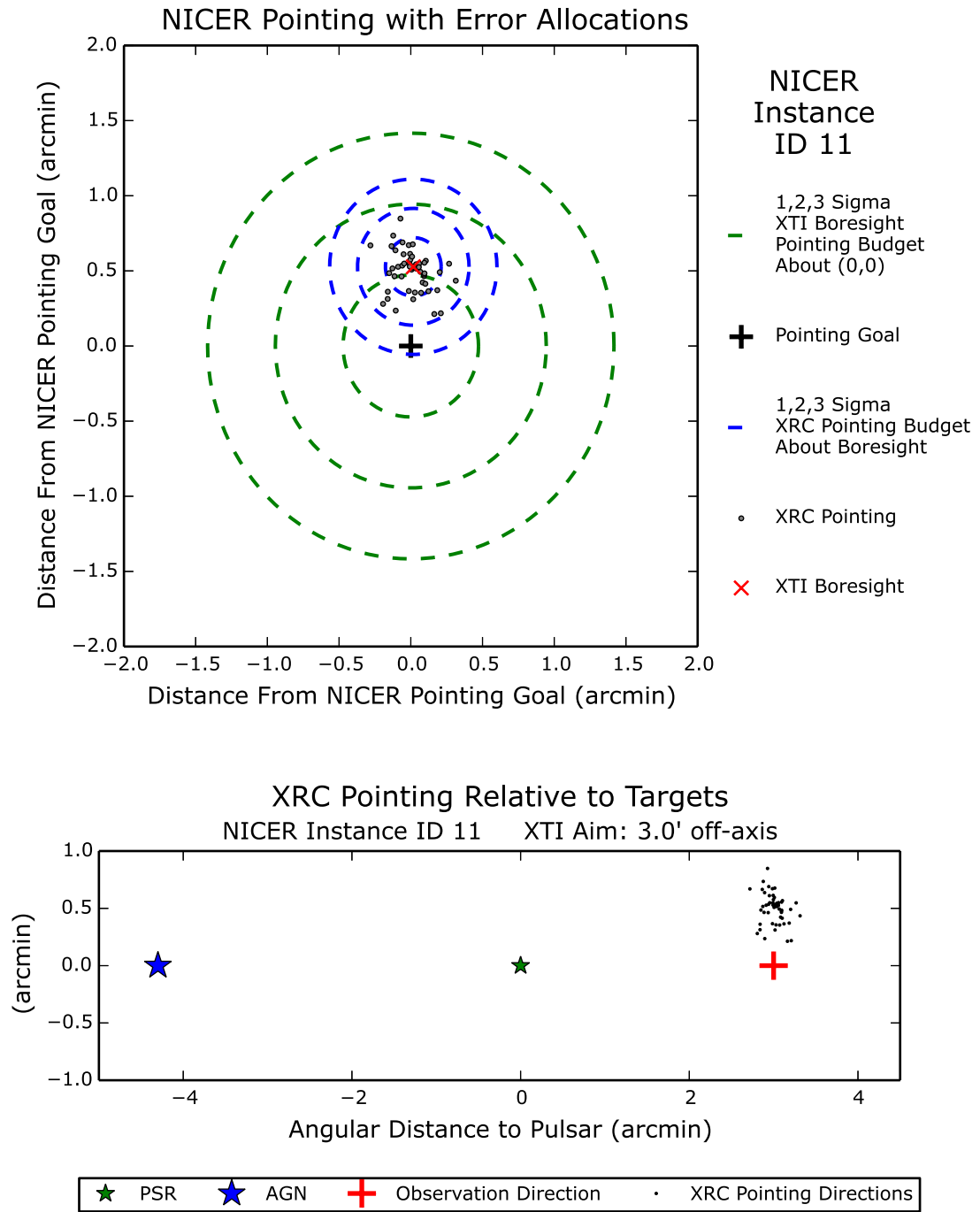


Figure 5.22: Random pointing errors of the NICER realization sampled in this section.

the values shown in Figure 5.23 were evaluated based on Equation 5.8, see Table 5.8.

Phase	Total Counts (T) (cts)	CRR (R)	Est. Signal (S) (cts)	S/t_{exp} (cts/s)
0.0	$(9.33 \pm 0.097) \times 10^3$	13.1 ± 0.5	$(8.67 \pm 0.09) \times 10^3$	$(1.73 \pm 0.02) \times 10^{-1}$
0.25	$(7.90 \pm 0.089) \times 10^3$	10.8 ± 0.4	$(7.23 \pm 0.08) \times 10^3$	$(1.45 \pm 0.02) \times 10^{-1}$
0.65	$(1.18 \pm 0.011) \times 10^4$	16.7 ± 0.6	$(1.11 \pm 0.01) \times 10^4$	$(2.23 \pm 0.02) \times 10^{-1}$

Table 5.8: Example calculations for estimating the pulsar signals. S is calculated using Equation 5.8. The time observed per phase bin in a 1 Ms observation, t_{exp} , is 50,000 s. These values are part of the 0.2–0.5 keV dataset shown in Figure 5.23.

An important aspect of these light curves is the difference between the synthetic and calculated signal. I determined the minimum of the relative difference between these light curves for 20 instances of NICER and 10 energy bands to estimate the optimal pointing angle as follows: Figure 5.24 is an example of the relative differences (see equation given in the figure) between lightcurves calculated for pointing angles $-1'$ to $3.5'$ in the 0.2–10.2 keV energy band, for one instance of NICER. I fit a linear function to the data points and determined where the line intersected the x-axis, the point at which the difference in the lightcurves is zero. The errors were calculated using the covariance of the parameters and represent the 1σ confidence interval.

5.8 Final Results and Conclusions

The results for the average optimal pointing angle, based on all 20 instances, are presented in Figure 5.25 for several energy bands with resultant representative offset angles ranges between $2.0'$ and $3.0'$. The plot shows that no significant energy dependence can be concluded from this study, as predicted in Section 5.6.1. This study suggests to initially point the XTI at the representative offset angle of $2.7'$

from the pulsar, in the opposite direction of the AGN. However, this angle will be further constrained through calibration observations when NICER is operational. The results shown are specific for the spectral parameters chosen to model the pulsar and AGN. For PSR J0437-4715, an atmospheric emission model was assumed with model parameters described in Table 5.2. AGN RX J0437.4-4711 was assumed to be constant over time and followed a the spectral model described by Table 5.3. If the AGN's flux were to change, then it is expected the optimal pointing direction would change as well. The technique I used to calculate the observed signal considered an ideal case XTI and XRC alignment. The count rate ratio of PSR to AGN flux modeled in the ideal case was used in Equation 5.8 along with the simulated observed data to give an estimated signal.

It would be beneficial to co-observe AGN RX J0437.4-4711 with another instrument while NICER is observing PSR J0437-4715 in order to estimate the AGN contribution more accurately. If this is not possible, the AGN should at least be observed by NICER prior to and after the pulsar's observation. Since the pulsar contribution to an AGN observation is negligible (see the $-4.2'$ pointing direction in Figure 5.14), the AGN can be observed on-axis. Since the AGN's flux appears to be mostly stochastic, predicting the variability is nearly impossible.

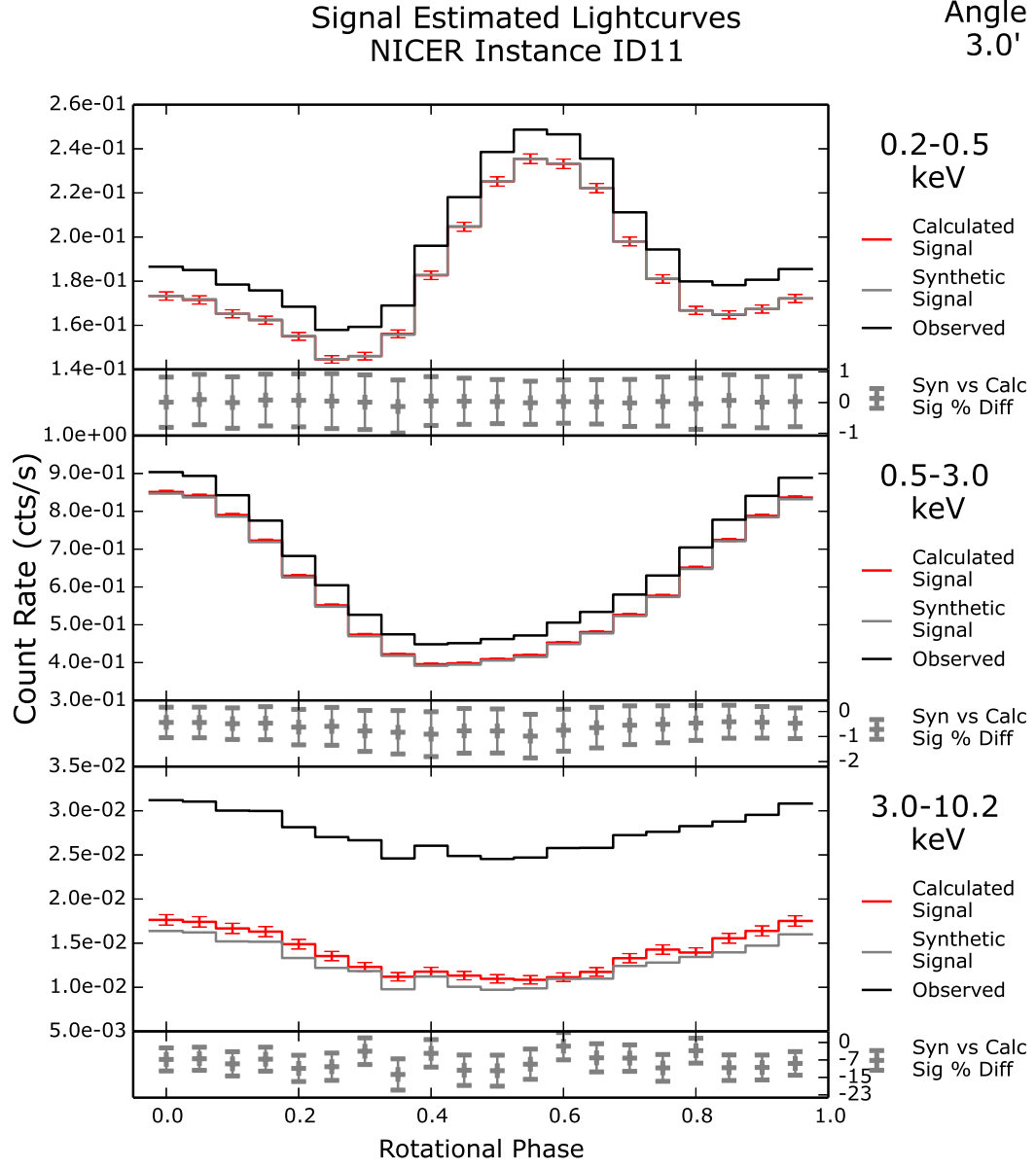


Figure 5.23: The pulsar contribution to the overall (T_{obs} , observed) light curve is estimated ($S_{\text{estimated}}$, calculated signal) from the observed light curve for a random distribution of XTI and XRC alignments in the three energy bands. The overall lightcurve is the sum of the synthetic pulsar (S_{syn} , synthetic signal) and AGN (B_{syn} , not shown) signals. The alignments are shown in Figure 5.22. The estimate was calculated assuming the CRR (R_{ideal}) from Figure 5.18, using Equation 5.8. In this example, the offset pointing angle was set to 3.0'. The uncertainties were calculated by taking the standard deviation of the total number of counts and dividing it by the observation time of each phase bin. The resultant uncertainties are shown by error bars on the calculated signal and the difference between the synthetic and calculated. (see first part of *ReconstructPulsarSignal.py* on page 196)

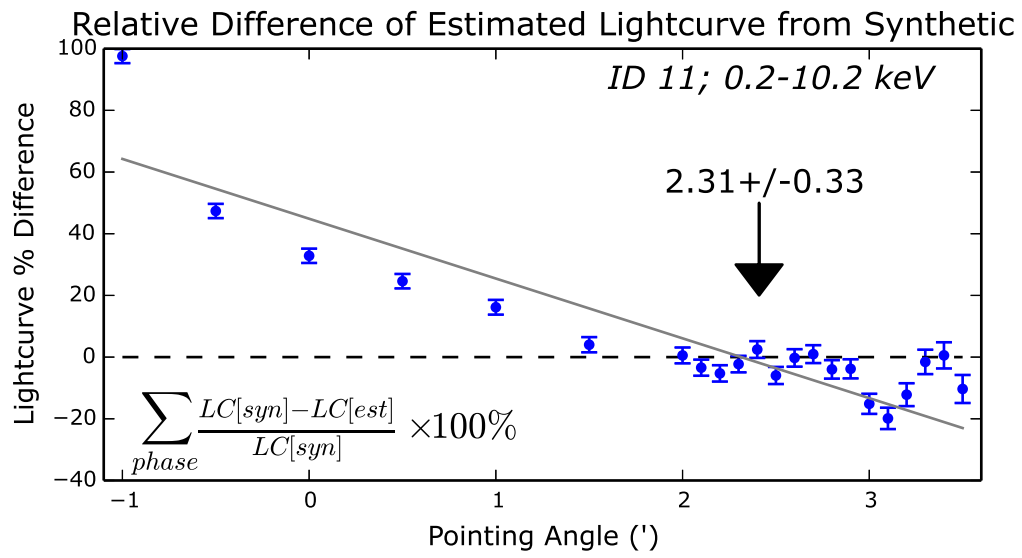


Figure 5.24: Relative difference of the estimated pulsar contribution to the light curve compared to the actual pulsar lightcurve over pointing angle. This sort of data was analyzed for 10 energy ranges for all 20 instances of NICER alignment. The data were fit with a linear function to estimate the location of minimum difference.

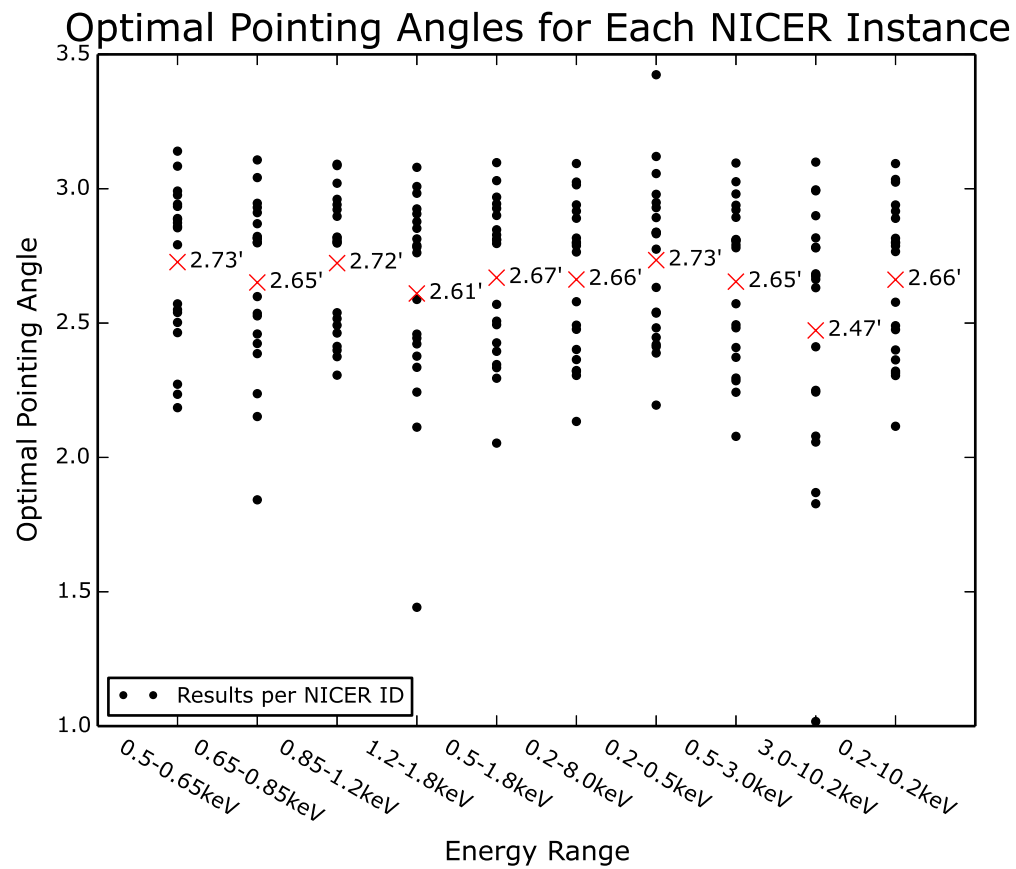


Figure 5.25: The representative pointing angle suggested for every energy band is shown with the data points of optimal pointing from each of the 20 NICER alignment realizations tested.

Chapter 6

Conclusions

This thesis included the study of several aspects of a new technology, the NICER XRCs, in order to provide efficient fabrication techniques and to develop an observation method, considering special features, useful for future observations with NICER. The efforts within the fabrication development and performance testing have already benefited the current project and has shown that a more precise design in thin foil epoxy replicated segmented X-ray optics is not only possible but also achievable within a limited budget. I answered the question of whether or not the alternative heat forming technique, the shrink tape forming method, was reliable and efficient. In the next part of the thesis, by simulating an observation with these concentrators, I contributed insight that may have been overlooked if one did not understand the hardware on the level of the optics response. I calculated the characteristic pointing direction for observing PSR J0437-4715 with NICER while considering a modern theory of pulsar emission, atmospheric emission.

The shrink tape heat forming method replaced vacuum forming of aluminum foil substrates, ultimately saving time and preserving resources. While the vacuum forming could have worked with equipment modification, testing the shrink tape method required much less modification and thus was more cost effective. The analysis of the substrate profiles successfully, in addition to the X-ray calibration

tests, proved that this technique is successful and viable. This analysis, in addition to the overall performance testing of early engineering test units and the flight ready XRCs, has changed the outlook on future design and fabrication of thin foil epoxy replicated X-ray optics at GSFC. Shrink tape heat forming will be used in future substrate fabrication.

Curved profiles and single substrate reflector shells can now be implemented into future designs due to work presented in this thesis and the successful efforts demonstrated by the NICER team. I recommend that accurately curved profiles of the Wolter I geometry be integrated into the design of imaging optics next. This would most likely be done on a small scale mission, such as a sounding rocket or balloon flight, and thus the heat shrink tape technique could be used since these sort of missions typically have stricter budget constraints. It may be possible to use spare NICER concentrators if a second set of reflectors could be designed with a hyperbolic profile to make imaging optics out of the pair. And to take this a step further, it would be nice to implement polarimeter for the detector and carry out research started with the XACT and GEMS missions.

The NICER observation of millisecond pulsar PSR J0437-4715 would benefit from pointing the XTI at $2.7'$ away from the pulsar, opposite the direction of the AGN, for a 1 Ms observation. The results show that $2.7' \pm 0.3'$ is the optimal pointing directions considering instrument alignment requirements stated in Table 5.7. A smaller uncertainty on the boresight alignment would allow this simulation to suggest an optimal pointing direction with more accuracy. Statistical uncertainty could also be decreased by conducting simulations assuming more virtual NICER

instances and longer observation times.

The simulation presented would have benefited from a better understanding of the AGN RX J0437-4711 emission spectra in the soft X-ray energy range. An AGN flux different than presented here would affect the optimal pointing angle. I recommend that this source be observed before and after observing the target MSP. I used this method during performance testing the XRCs in the X-ray beamline to get the most accurate measurement of effective area; I measured the incident beam before and after the test. An additional method would be to observe the AGN with another instrument at the same time as NICER is observing the pulsar.

Additional studies could be done to compliment the simulation performed in this thesis, which focused on the XRC response. Such studies would include other aspects of the NICER response function. The detector efficiency could be optimized with data from calibration of the flight SDDs, including a more specific redistribution matrix than the one assumed here. Another study could include the quality testing analysis of the thermal shield that may change the transmission filter file used in generating the response file. The most significant change in my work would come from the analysis of actual pointing ability rather than using only the allocation specifications from the proposal. A study of this sort could include data from ground testing²⁹ of alignment features and ISS-based observations of well known sources. By pointing at a well known source, maybe the Crab pulsar, one could collect data from off-axis pointing and perform a space-based calibration of the vignetting function, allowing one to infer the average error of the XTI and XRC

²⁹Ground calibrations were not available in time for the simulations in this thesis.

alignment.

From here on forward, thin foil epoxy replicated X-ray optics will continue to improve for large and small scale missions, thus benefiting research in X-ray astronomy. Observations with NICER will soon contribute data to the database with high timing resolution within the soft energy range up to 12 keV. This will include phase resolved energy spectra of the closest millisecond pulsar, which will benefit from the results in this thesis. By understanding the emission of PSR J0437-4715 in the NICER energy band and generating energy-dependent light curves with NICER's superb timing resolution, the neutron star radius can be calculated within 5% accuracy and further constrain the equation of state.

A NICER understanding of nuclear physics is in our future!

Appendix A

X-ray Physics and Interactions

When electromagnetic radiation interacts with matter, processes such as absorption and scattering (including reflection) can occur. The probability of each type of interaction occurring depends strongly on the energy of the radiation and the physical conditions (composition, density, etc) of the interacting material. Reflection plays a crucial role in the production of focusing optics. Absorption and scattering processes are required for the detection of radiation and are important during the passage of radiation from its source to the detector. We briefly review the terminology and relationships in common use for each of these processes paying particular attention to the soft & medium X-ray regimes. In the X-ray regime, radiation is usually characterized by the photon energy measured in kiloelectronvolts (keV), where 1 keV corresponds to 1.606×10^{-16} J, a frequency of 2.418×10^{17} Hz, and a wavelength of 1.240×10^{-9} m.

“Soft-energy” (0.1-2 keV) and “Medium-energy” (2-10 keV) X-ray photons have energies comparable to the K-shell energy levels in the cosmically abundant elements (from Carbon [Z=6] through Nickel [Z=28]). As a result, when an X-ray photon interacts with a molecule or an atom, it can be photo-electrically absorbed which would either raise an electron to a higher energy level (exciting the atom) or completely expel an electron from the atom (ionizing the atom). These processes

form the physical basis of X-ray detectors. They are also responsible for the removal of photons from the line of sight to cosmic X-ray sources.

The attenuation of radiation of incident intensity I_0 as it travels through a medium can be expressed as

$$I = I_0 e^{-\tau} \quad (\text{A.1})$$

where τ is the “optical depth” of the material. If $\tau \gg 1$ then the material is considered to be “optically thick,” while if $\tau \ll 1$ then the material is “optically thin.” The Earth’s atmosphere, for example, is optically thick at X-ray energies thus requiring space based observations (Spiller, 1994). (On the other hand, the atmosphere is optically thin to optical light as well as some regions of the radio band.) The dimensionless quantity τ is composed of values depends upon material properties and path length. There are a few different ways to parameterize τ with slightly different terminology depending on the application (and somewhat on different traditions of different fields).

The optical depth is composed of two parts: the “stuff factor,” more commonly referred to as the absorption coefficient, α , and the “distance factor,” or the length of travel in the material, ds .

$$d\tau = \alpha ds \quad (\text{A.2})$$

The absorption coefficient can be described in a couple different ways; however each way is broken up into two variables: one describing the medium as a whole and one

describing the intrinsic properties of the material.

$$\alpha = n\sigma \tag{A.3a}$$

$$\alpha = \rho\kappa \tag{A.3b}$$

Equation A.3a writes α as the property of the number of particles per unit volume (number density), n , and the cross section of the particles, σ . This definition is often used in situations where n is relatively low, like when considering the attenuation of X-rays through interstellar space. Equation A.3b uses the mass density of the material, ρ , along with the mass absorption coefficient, or opacity, κ , of the material. An equation like this may be more useful for a solid surface, such as a reflecting gold surface of an X-ray mirror.

The Optical Constants In X-ray optics it is more common to relate above α to the optical constants. In that case

$$\alpha = \frac{4\pi\beta}{\lambda} \tag{A.4}$$

where β is a property of the material and a function of the atomic scattering factors. β is also found in the imaginary part of the refractive index and sometimes referred to as the extinction coefficient (Spiller, 1994). The atomic scattering factors,

$$f = f_1 + if_2 \tag{A.5}$$

are based on measurements of photo-absorption of elements(Henke et al., 1993). The values are tabulated by Henke et al. (1993) for the range of 50 eV to 30 keV. Physically, the atomic scattering factor is the effective number of electrons per atom when a photon of a particular energy interacts with that atom(Spiller, 1994). The reflection properties of material can be describe by the “refractive index”

$$n = 1 - \delta + i\beta \tag{A.6}$$

which is defined by the optical constants δ and β . Both constants are defined by the tabulated atomic scattering factors and are each dependent on the incident energy, or wavelength. The constants relate to the properties of light and material by approximately

$$\delta \propto \lambda^2 f_1 \tag{A.7a}$$

$$\beta \propto \lambda^2 f_2 \tag{A.7b}$$

so each constant decreases as the photon energy increases (see Figure A.1).

At small grazing incident angles, X-rays undergo total external reflection. The incident angle is equal to the angle of reflection, as for all optical reflections. Only if the angle is small enough is there total external reflection, otherwise some refraction may occur. Refraction of radiation at a vacuum/medium boundary is explained by Snell’s law is where θ_r is the angle of refraction, θ_i is the grazing incident angle, and

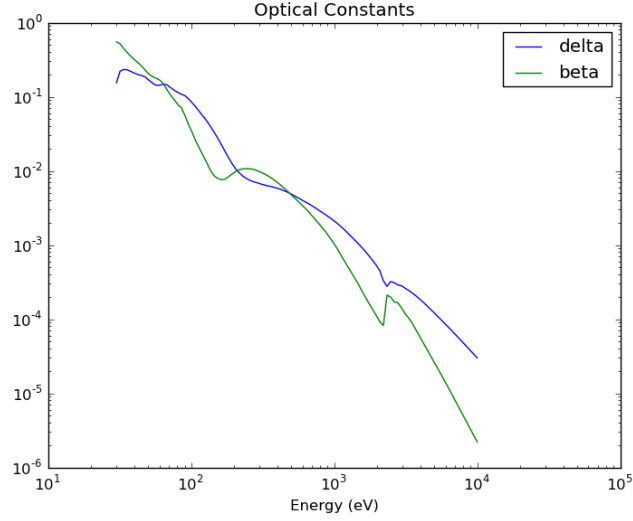


Figure A.1: The optical constants δ and β as a function of energy.

δ is one of the optical constants defining the index of refraction:

$$\cos\theta_r = \frac{\cos\theta_i}{1 - \delta} \quad (\text{A.8})$$

The index of refraction contains the optical constants δ and β from Eq A.6 which are medium dependent. The imaginary term of the refractive index is irrelevant in Eq A.8 since θ_r is physically meaningful only if real. For the derivation of the critical angle of reflection, we use the δ for energies well away from any absorption edges (Arnaud et al., 2011)

$$\delta = \frac{r_e \lambda^2 N_e}{2\pi} \quad (\text{A.9})$$

where r_e is the electron radius, λ is the wavelength of light and related to energy by $\lambda[\text{\AA}] = 12.39/E[\text{keV}]$, and N_e is the free electron density. β is non-zero for a conductive medium but not applied in this derivation. In the X-ray regime, both constants are much smaller than unity which allows for small angle approximations

in solving for the critical angle.

The critical angle, θ_c , is the minimum incident angle needed for total external reflections, which is also the maximum angle at which refraction is possible. Finding this first depends on the solution of the refracted angle, θ_r . When θ_r does not have a real solution, the light will not be refracted but only reflected. Therefore, we set θ_r to zero at the critical angle. At X-ray wavelengths the optical constant δ is always positive and therefore $1 - \delta$ may not exceed unity. So now lets replace the incident angle with the critical grazing angle and solve for the critical angle in terms of the optical constant δ for X-rays. Since $\theta_r = 0$, $\cos(\theta_r) = 1 = \cos(\theta_c)/(1 - \delta)$. This leaves $(1-\delta)=\cos(\theta_c)$ which can be approximated by the binomial expansion of cosine for small angles so that $(1 - \delta) \approx 1 - \theta_c^2/2$. Therefore the critical angle can in turn be described by δ as(Arnaud et al., 2011)

$$\theta_c = \sqrt{2\delta} \tag{A.10}$$

A proportionality argument can now be given to relate the critical grazing angle of light to its energy and the surface material. By combining Eqs A.9 and A.10, it is shown that the critical angle is inversely proportional to the energy and proportional the free electron density of the reflecting material:

$$\theta_c \propto \frac{\sqrt{N_e}}{E} \tag{A.11}$$

This shows that it is more beneficial to use high free electron dense metals for

X-ray reflections in order to widen the range of incident angles for high energies. (There are of course many other constraints on the chosen material based on the fabrication method and robustness for space flight). It should be noted that Eq A.11 does not account for energies near the absorption edges of the atoms. There is minimal reflection at these edges and therefore these edges should be considered in selecting the optimum material for the project's scientific interest. The reflection across energies at a specific incident angle is calculated using the atomic scattering factor, or the index of refraction.

Fresnel Equations Across a surface, the X-ray reflectivity depends upon the optical constants, found in the index of refraction, and the incident angle. By applying the plane wave equation to the electric field and allowing the tangential components of the electric field and normal components of the magnetic field to be continuous across the reflective boundary, the amplitude of reflective light can be deduced (Jackson, 1998). Comparing the amplitude of the electric fields and introducing Snell's law, Fresnel's equations calculate the reflectivity of incident photons at a specific angle onto a medium with index of refraction n . There exists a parallel (r_p) and a perpendicular (r_s) component of the reflection. Fresnel's equations are:

$$r_p = \frac{n^2 \sin \theta - \sqrt{n^2 - \cos^2 \theta}}{n^2 \sin \theta + \sqrt{n^2 - \cos^2 \theta}} \quad (\text{A.12a})$$

$$r_s = \frac{\sin \theta - \sqrt{n^2 - \cos^2 \theta}}{\sin \theta + \sqrt{n^2 - \cos^2 \theta}} \quad (\text{A.12b})$$

For unpolarized light the reflectivity is the averaged amplitude of r_p and r_s :

$$R = \frac{|r_p|^2 + |r_s|^2}{2} \quad (\text{A.13})$$

These equations assume an infinitely smooth and optically thick (and dense) surface in a vacuum, a perfect world.

Appendix B

XRC Fabrication

The X-ray Concentrator's fabrication method has been discussed in my published conference proceeding from the 2012 SPIE Astronomical Telescopes and Instrumentation. This paper is Balsamo et al. (2012).

I contributed a poster for the NICER program site visit. The site visit is what got the mission accepted as a NASA Mission of Opportunity. I have included that poster in this appendix as Fig B.1.

NICER Concentrator Fabrication Process





Inlet and outlet spokes are connected by a solid dumbbell.



Grooved spokes allow the shells to sit in their proper locations.



Each shell, beginning with the smallest, is carefully lowered over the dumbbell and into grooves onto one set of spokes.



The second set of spokes are put on and the shells are epoxied into the grooves to complete a NICER concentrator.

PROPRIETARY INFORMATION: Use or disclosure of data contained on this sheet is subject to the restriction on the title page of this proposal.

Figure B.1: Poster of the XRC fabrication process. I made this for the NICER site visit to help the team become accepted as a NASA Mission of Opportunity

Appendix C

CCD Data Analysis, Troubleshooting, and Features

C.1 Unique Factors in X-ray Testing: Experiment Parameters

Figures C.1 - C.5 show results of setting changes on the CCD and/or X-ray source at the 600 m beamline. Specifically the results show differences in pulsar height spectrum and split distribution.

Understanding the split distribution allowed me to understand if the events should be discriminated by split in the analysis. I determined that grade is a better discriminator than split and chose to use all split but only select grades in my final calibration analysis.

By studying the pulse height spectra, I learned how to reduce noise by any parameters I was able to control allow most of the events to be at the chosen source energy and eliminate the possibility of pile-up.

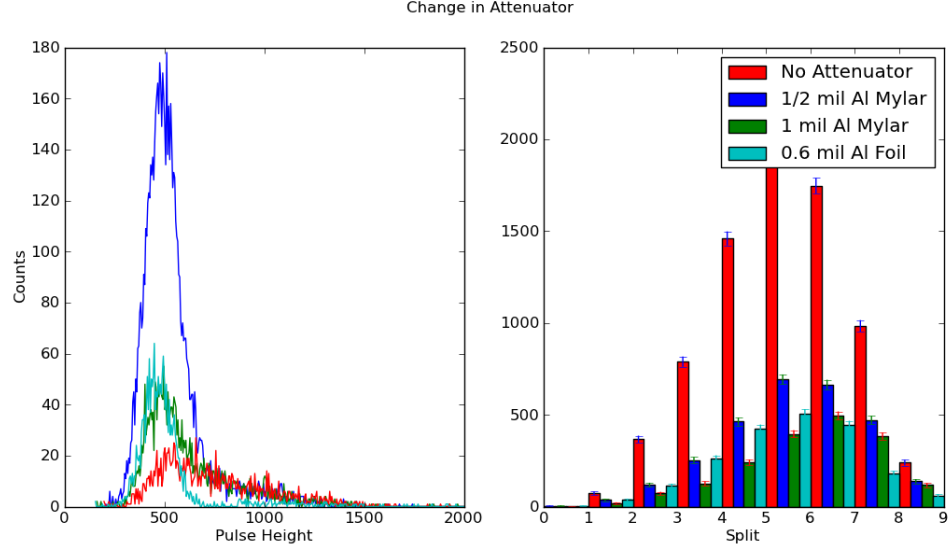


Figure C.1: Changes in the pulse height spectra and counts per split level at different attenuator thickness placed after the X-ray source in the 600m beamline.

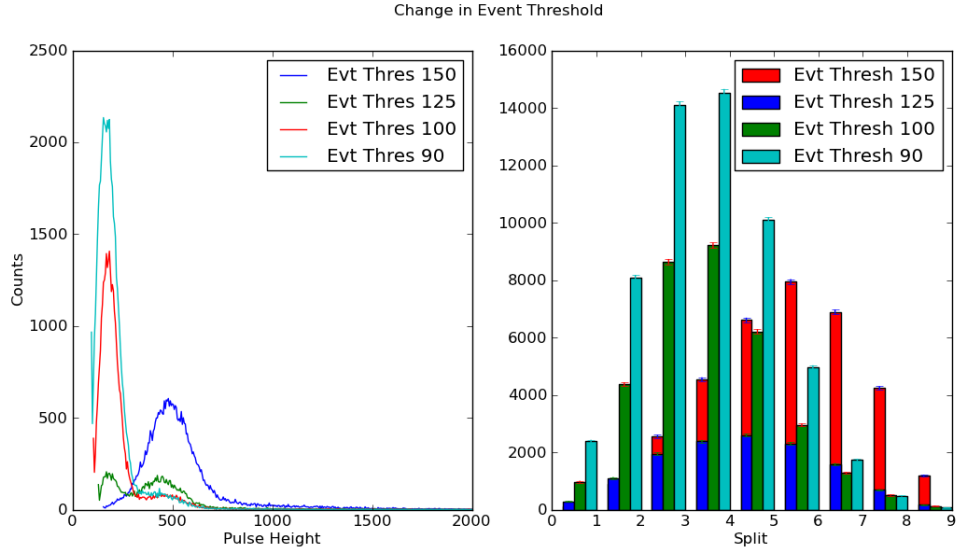


Figure C.2: Changes in the pulse height spectra and counts per split level for different event threshold settings on the roper server (600m beamline).

C.2 CCD Background Spectrum and Effect on Measured Effective Area

In this section of the appendix, I present my analysis and observations of the CCD background (of the TEC Cooling CCD) from the 600 m beamline. If instru-

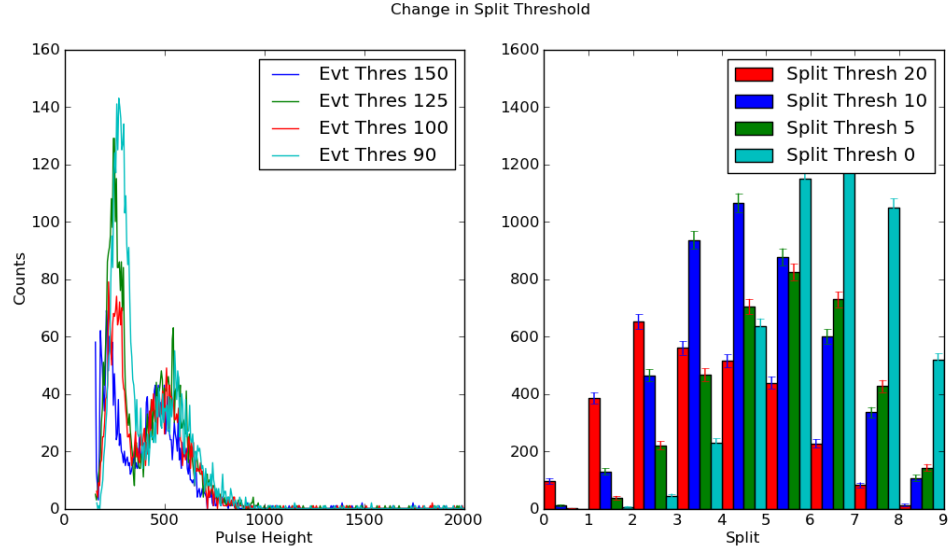


Figure C.3: Changes in the pulse height spectra and counts per split level for different split threshold settings on the roper server (600m beamline).

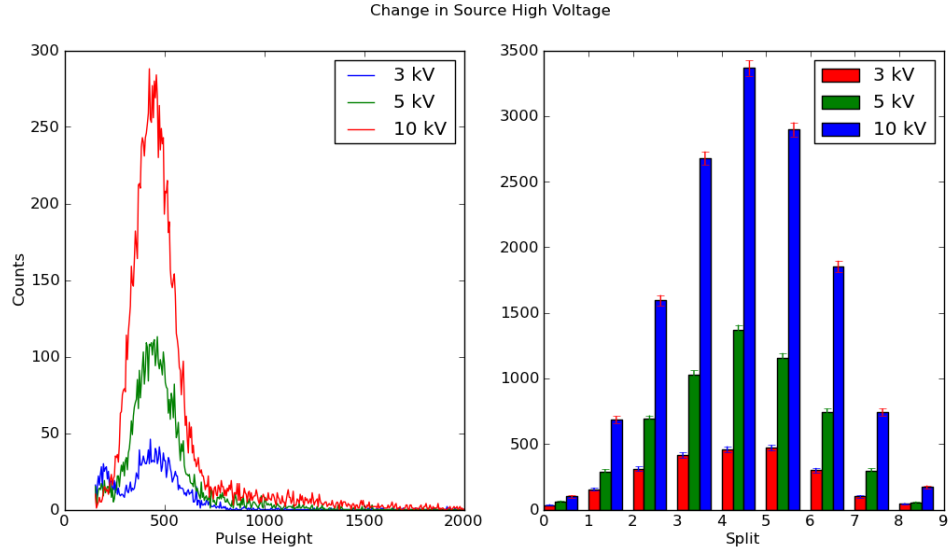


Figure C.4: Changes in the pulse height spectra and counts per split level for different high voltage settings on the X-ray source (600m beamline).

ment background is significant, it can affect the XRC effective area measurements.

Results from this work placed a limit on allowable CCD background (see Fig. C.12)

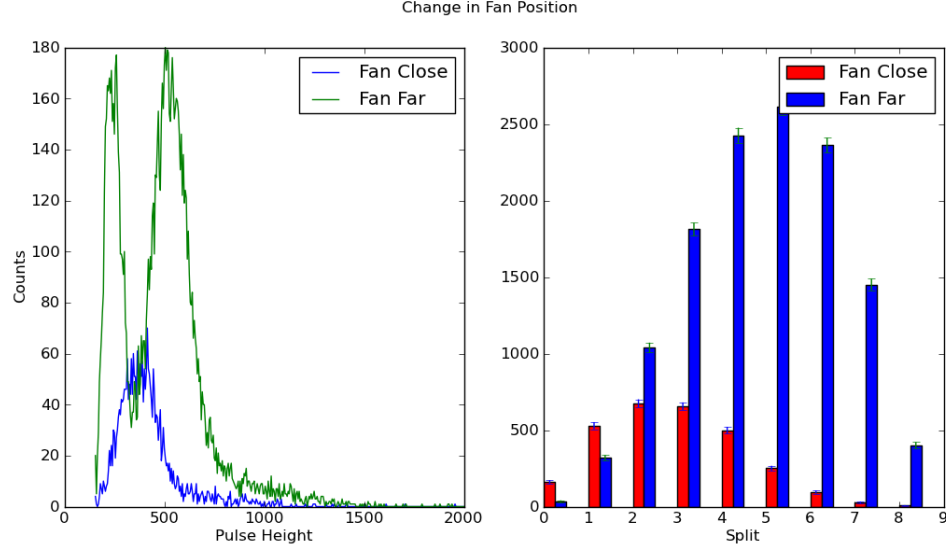


Figure C.5: Changes in the pulse height spectra and counts per split level when the position of a fan blowing on the CCD changed, outside of the chamber (600m beamline).

C.2.1 Introduction

While performing X-ray tests in XRC developmental units, I noticed that the photon counting experiments produced inconsistent results, such as significant count rate changes between frames, even with no X-rays. Therefore, I tested the CCD background to understand its behavior. I then analyzed the pulse height spectra of the CCD background and compared the count rates to a few example measurements from the photon counting experiments to determine the possible effect CCD background may have had on the effective area calculations for when it is not accounted. The effective area was calculated by measuring photon flux of with and without the XRC in the pathway of the X-rays. The flux density on the CCD surface is very different for both of those types of measurements (i.e. the XRC may measure roughly 40 events per cm^2 per unit time while the incident beam may show

0.7 events per cm² per unit time). Therefore, an unaccounted for CCD background could artificially inflate the measured incident beam counts making the resultant effective area appear much lower than it actually is.

C.2.2 Modeling the CCD Background Spectra

Figure C.6 is the overlaid spectra of each CCD background dataset collected in terms of pulse height and counts per second. I collected four background datasets with different event and split thresholds (abbreviated as ET and ST in this appendix section).

I fit the spectra based on the two main components: the low energy noise (mostly pulse height $< 6 \times 10^{-3}$) and high energy noise (mostly pulse height $> 6 \times 10^{-3}$). I modeled the low energy noise as an exponential and the higher energy as a blackbody. Each term was multiplied by its own photo-electric absorption component (fit variables $b1$ and $b2$ respectively). This model is described in Equation C.1 (where x is for pulse height).

$$(A1 * e^{-a1x}) * e^{-b1/x^3} + \left(A2 * \left(\frac{x^2}{e^{x/T}} \right) + y0 \right) * e^{-b2/x^3} \quad (\text{C.1})$$

The low energy noise was predicted to be a result of dark current since it was observed to decrease in intensity as the CCD cooled to the operation temperature. (Howell, 2006) I chose to model this feature with a function that rapidly decreases as pulse height increases because count rate was lower when ET was set higher (recall that ET is a threshold related to the minimum pulse height read by

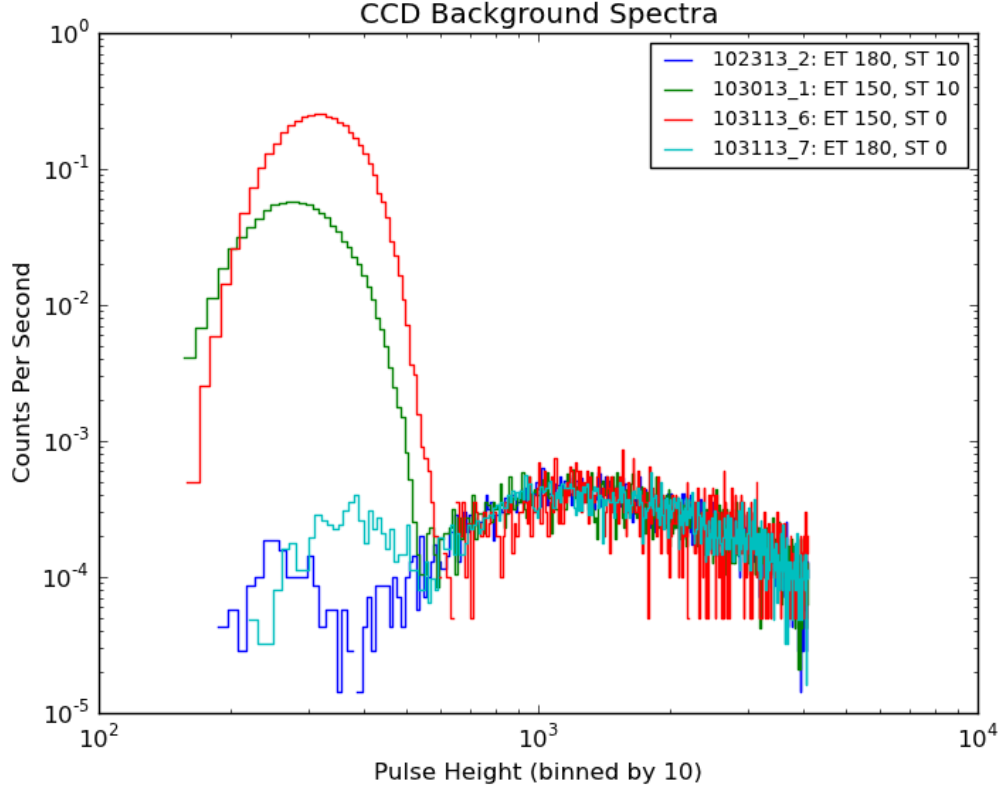


Figure C.6: Pulse height spectra of four data sets collected to measure the CCD background. These data sets were collected by choosing different event and split thresholds (ET and ST respectively). When ET was set to 150, there were more lower energy events than when ET was set 180; the red and green lines have a higher count rate then the blue and cyan. The change in the ST setting shifted the spectrum peak of the lower energy events. The higher ST, 10, resulted in a peak shifted to a lower energy than the lower ST, 0; the blue and green peaks are to the left of the red and cyan.

the CCD). The exponential (term coefficient $A1$ and exponent coefficient $-a1$) was the best fit I found. I further tested this by fitting the model to the difference in spectra of the ET=180 and ET=150 data sets (i.e. I subtracted the spectrum of the 150 ET 10 ST data set from the 180 ET 10 ST data set, and same for 0 ST).

The higher energy noise was constant among all four data sets and independent of the threshold settings. I expected it was a result of thermal noise and therefore

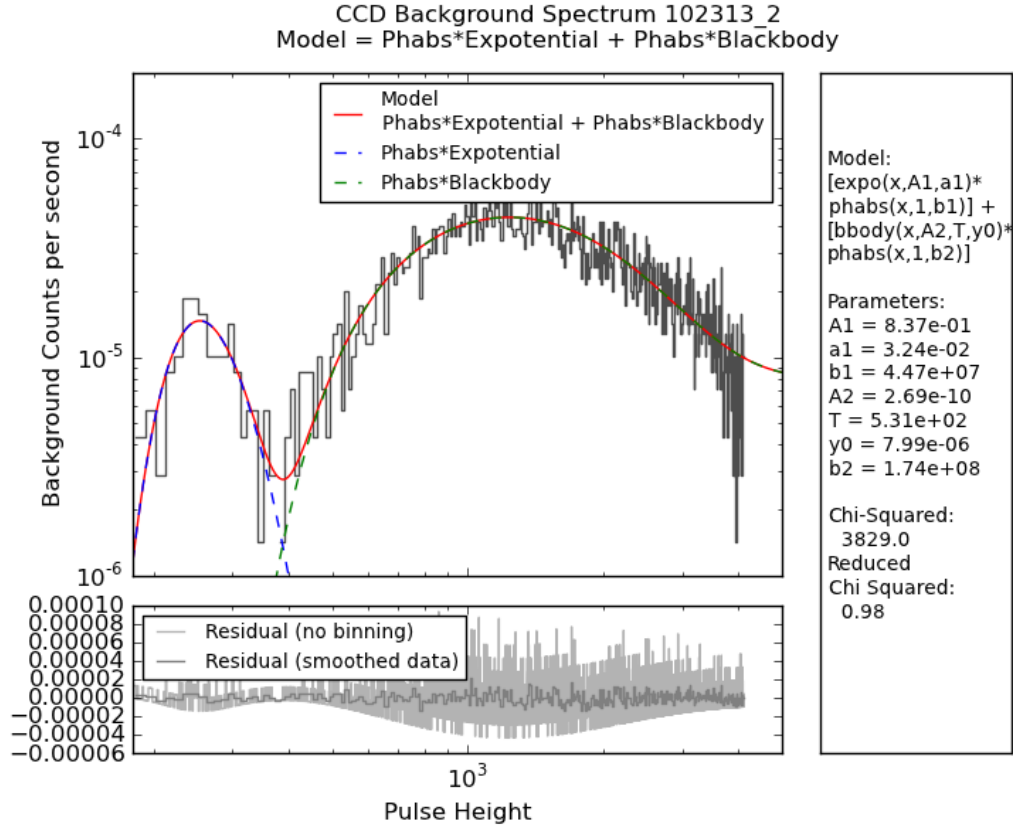


Figure C.7: CCD Background spectrum fit to the model described by Equation C.1.

chose to model this feature with a blackbody (coefficient $A2$ and temperature $\propto T$). A constant component, $y0$, was added to this term to account for any possible cosmic rays.

Figure C.7 shows the model described by Equation C.1 fitted to the spectrum of the 180 ET 10 ST data. These settings are the same roper server settings as the measurements from the photon counting experiments (previous measurements for effective area calculations), so this particular data set was used as the example CCD background spectra in the following analysis for the CCD contributions effect on effective area calculations.

C.2.3 Photon Counting Experiments

In order to continue my analysis on this issue, I chose three sets of incident X-ray data and the corresponding XRC data for which to compare to the CCD background spectrum. The ROIs and duration are accounted for in the spectra.

The data are shown in Figures C.8-C.10. The spectra were normalized to counts per second per 512 pix x 512 pix ROI on the (a) left of each figure along with overlaid fits. The (a) top right of each figure shows the incident beam and XRC spectra with the background spectra subtracted. The (a) bottom right shows the ratio of X-ray to background counts. The shaded region in each plot is the 1, 2, and 3 sigma widths around the Al K-alpha peak of the XRC measurement of each set. The ratio plots showed me that background may have accounted for about 1% of the events in the 1.5 keV energy band. Since I observed that the incident beam was affected by the background much more than the XRC events, this lead me to suggest that the effective area calculation must be affected.

C.2.4 Noise Factor Calculation

(Note: variables defined in list at end of this section, C.2.4)

I calculated a factor due to the CCD background based on the maximum percentage of background to data in each of the 3σ energy widths. I call this the *noise factor*. For an energy dependent model, I fit all the spectra involved and used the energy (or rather pulse height) dependent ratios to find the *noise factor*.

The effective area calculation is the reflectivity, R , of the XRC multiplied by

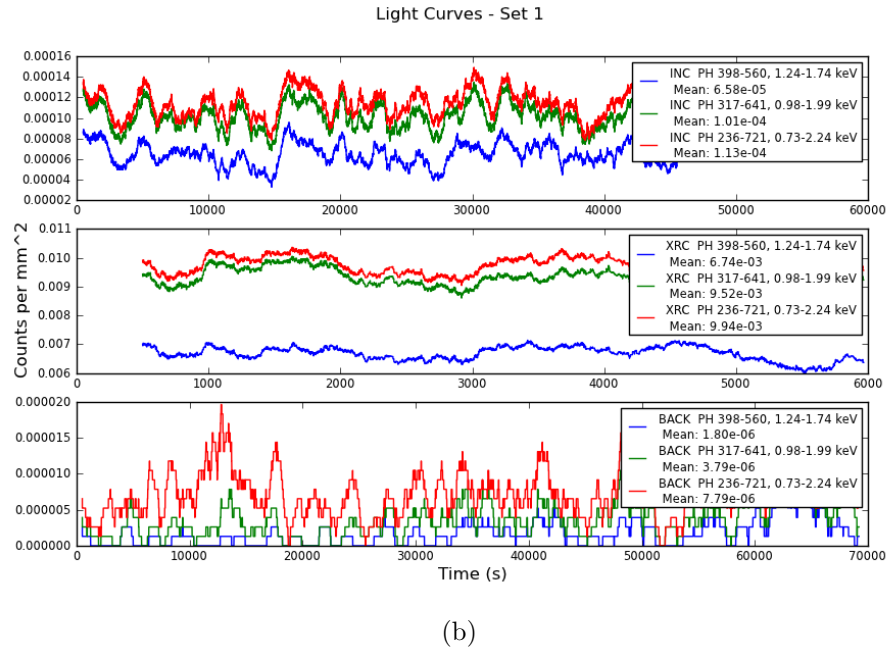
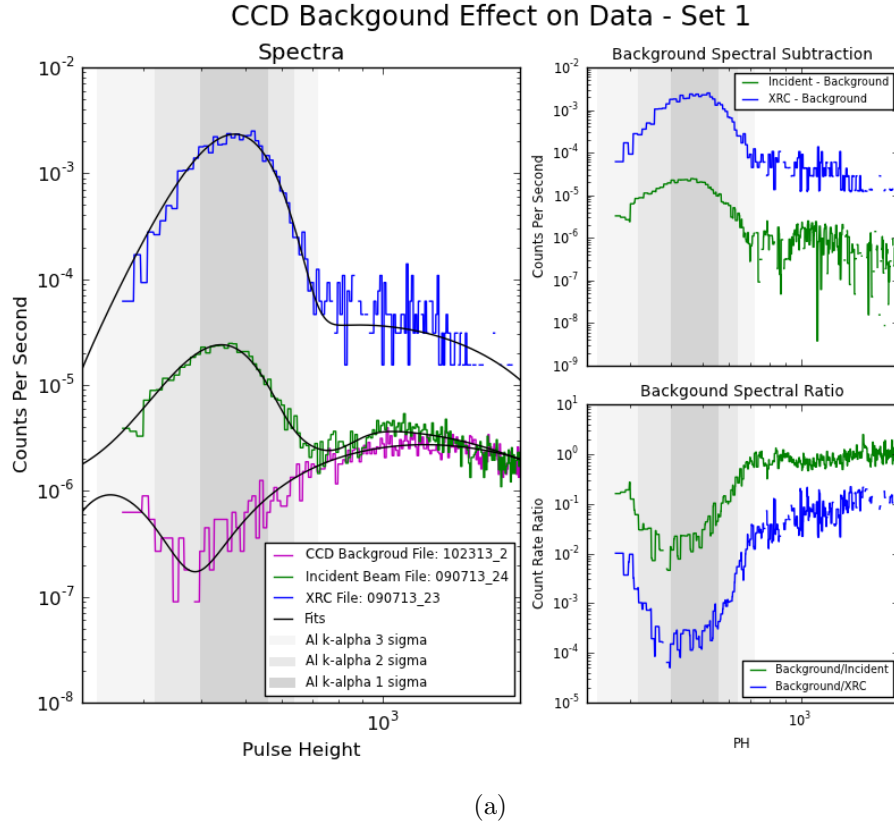
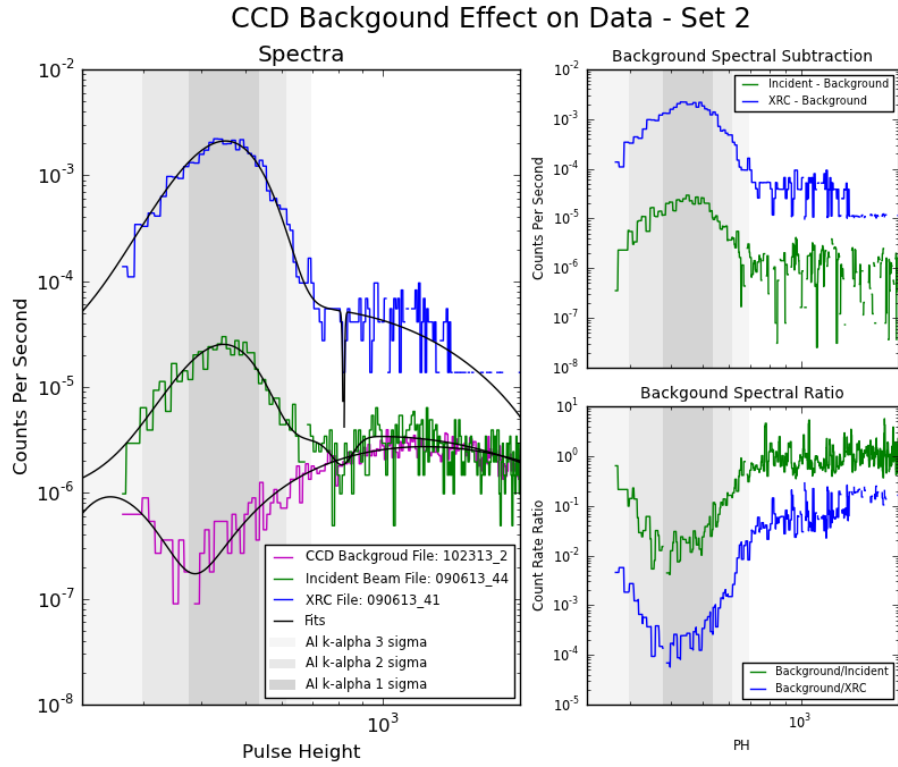
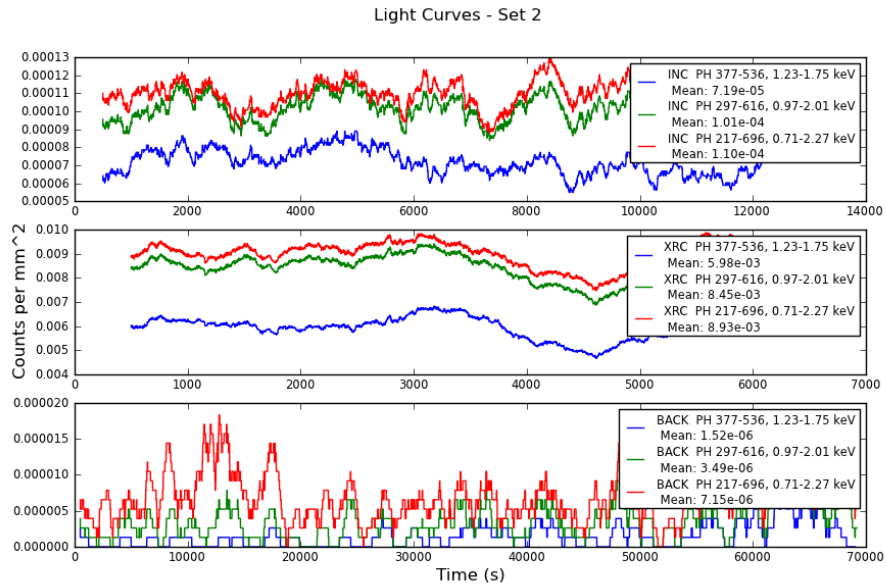


Figure C.8: Data Set 1: CCD background effect on data. An incident and XRC beam spectra are compared to CCD background spectrum. Show that background may have accounted for about 1% of the events in the 1.5 keV energy band. (a) Left- XRC (blue), incident (green), and CCD background (magenta) spectra with fits. Right top- Incident minus background spectra (blue) and XRC minus background spectra (green) Right bottom- background to incident ratio (green) and background to XRC ratio (blue). (b) Lightcurves in counts per mm^2 over time (s). Top- incident beam data. Middle- XRC data. Bottom- CCD background data. Blue: 1.24-1.74 keV. Green: 0.98-1.99 keV. Red: 0.73-2.4 keV.



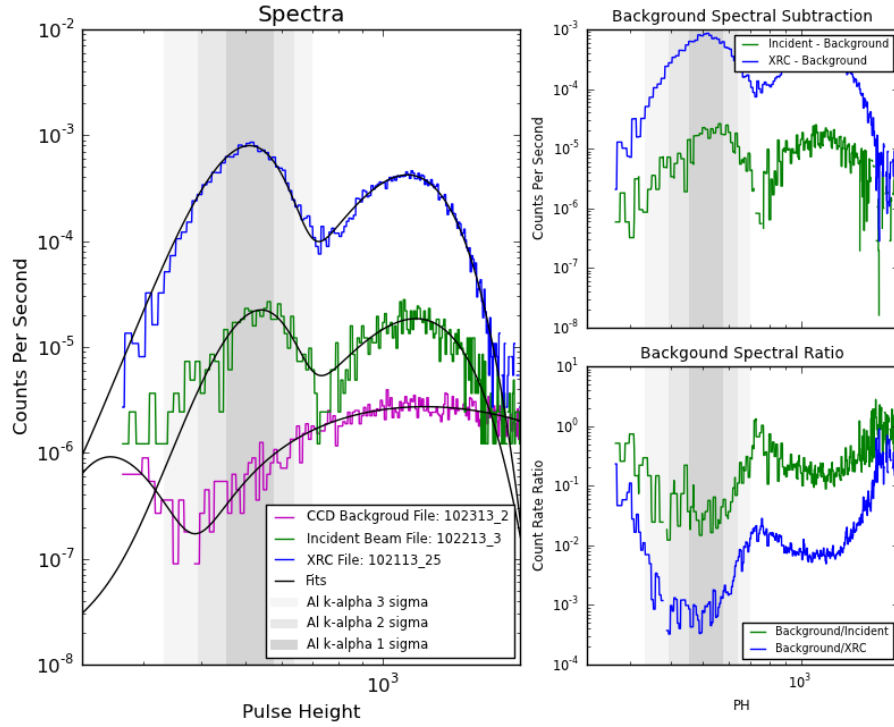
(a)



(b)

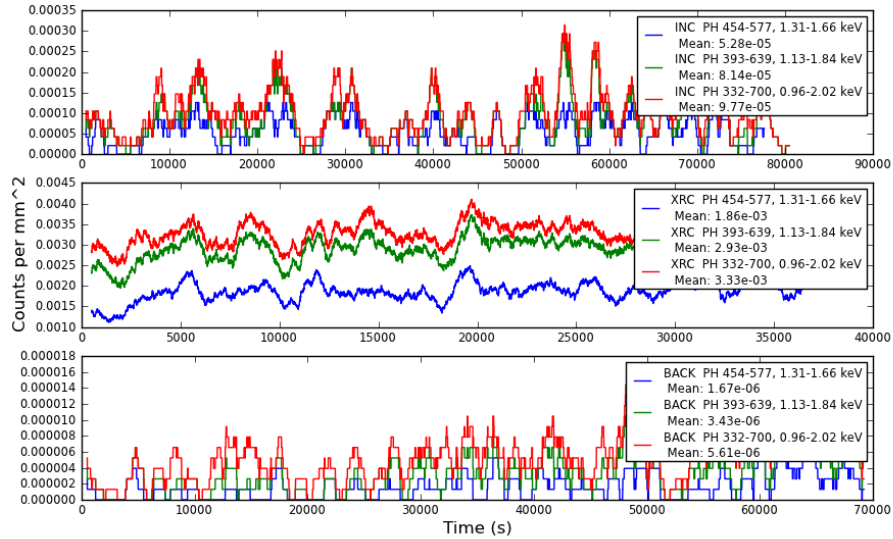
Figure C.9: Data Set 2 (see caption to Fig. C.8)

CCD Background Effect on Data - Set 3



(a)

Light Curves - Set 3



(b)

Figure C.10: Data Set 3 (see caption to Fig. C.8)

the geometric area, A_G , of the XRC where the reflectivity is the fraction of the number of photons observed on the focal plane, N_{foc} , compared to the number of photon incident onto XRC, N_{inc} (equation below). This equation is equivalent to Eq. 2.2.

$$A_{eff} = R \times A_G = \frac{N_{foc}}{N_{inc}} \times A_G$$

In practice, the number of photons at the focal plane is the number of events counted on the CCD with the XRC module in place. This can be broken down into the flux at the focal, n_{foc} , the length of the measurement, t_x , and the area at the focal or the area of the CCD, A_{CCD} . The number of incident photons must be estimated using the flux at the focal due to the incident beam of X-rays, n_{inc} , and multiplying it by the length of the XRC measurement and geometric area of the XRC module.

$$R_{measured} = \frac{N_{foc}}{N_{inc}} = \frac{n_{foc} \times t_x \times A_{CCD}}{n_{inc} \times t_x \times A_G}$$

$$R_{measured} = \frac{n_{foc}}{n_{inc}} \times \frac{A_{CCD}}{A_G}$$

Here, $R_{measured}$ is the measured reflectivity using data from two measurements, the XRC and the incident beam measurements. However, there is an extra term in the fluxes, the CCD background flux.

$$n_{foc} = n_x + n_b \quad \& \quad n_{inc} = n_d + n_b$$

With no CCD background, the background flux would be set to 0 and the actual

reflectivity is

$$R_{actual} = \frac{n_x}{n_d} \times \frac{A_{CCD}}{A_G}$$

With the contribution of the CCD background, I expect the measured reflectivity to be less than the actual reflectivity by some *noise factor*. I define the relationship between the actual and measured reflectivities as

$$R_{actual} \equiv noise\ factor \times R_{measured}$$

I can now substitute in the calculations above into this definition and determine the *noise factor*.

$$\frac{n_x}{n_d} \times \frac{A_{CCD}}{A_G} = noise\ factor \times \frac{n_{foc}}{n_{inc}} \times \frac{A_{CCD}}{A_G} \rightarrow \frac{n_x}{n_d} = noise\ factor \times \frac{n_x + n_b}{n_d + n_b}$$

$$\frac{n_x}{n_d} = noise\ factor \times \left[\frac{n_x}{n_d + n_b} + \frac{n_b}{n_d + n_b} \right]$$

\vdots

$$= noise\ factor \times \left[\frac{1 + n_b/n_d}{1 + n_b/n_x} \right] \times \left(\frac{n_x}{n_d} \right)$$

Now I am left with the *noise factor* as Eq C.2.

$$noise\ factor = \left[\frac{1 + n_b/n_d}{1 + n_b/n_x} \right] = \left[\frac{1 + n_b/(n_{inc} - n_b)}{1 + n_b/(n_{foc} - n_b)} \right] \quad (C.2)$$

Now I have a relationship between the previously measured values and the actual

values in terms of measured fluxes, Eq C.3.

$$R_{actual} \equiv \left[\frac{1 + n_b/(n_{inc} - n_b)}{1 + n_b/(n_{foc} - n_b)} \right] \times R_{measured} \quad (C.3)$$

Variables:

A_{eff} : Effective Area

A_G : Geometric Area of XRC

A_{CCD} : Area of CCD

R_{actual} : Reflectivity with CCD background effect

$R_{measured}$: Reflectivity previously measured without background consideration

t_x : Time duration of XRC measurement

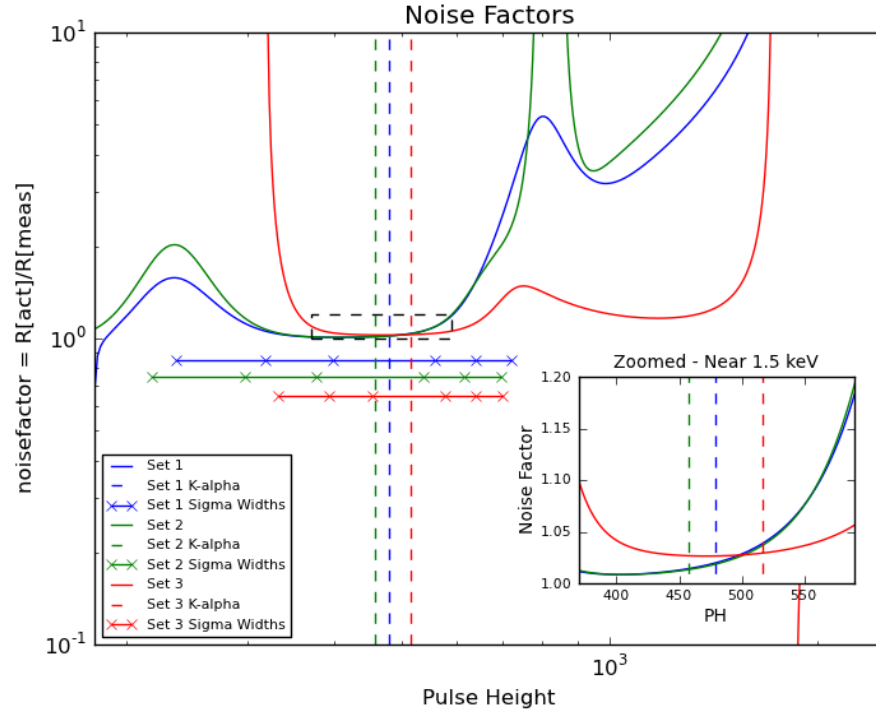
N_{foc} : Number of photon at focal point

N_{inc} : Number of photons incident on XRC

n_{foc} : measured counts per time per area of XRC reflected photons

C.2.5 Analysis

In Fig C.11, I show the noise factors from Eq C.2 for each of the three set examples. The noise factor is close to 1 around the aluminum peak, 1.5 keV. To calculate the noise factor for a particular sigma width about the peak energy, I take the mean noise factor in that energy range. The figure also shows the possible changes in the effective area calculation when the noise factor is taken into consideration, using Eq C.3. When CCD background is taken into consideration, and the noise factor is applied as a correction to the effective area calculation, the effective area is increased by $> 2\text{cm}^2$ for the 2-sigma width and $> 10\text{cm}^2$ for the 3-sigma width. This is a significant difference that, as explained below, should not be neglected.



Example Noise Factors			
	Sigma 1 Noise Factor	Sigma 2 Noise Factor	Sigma 3 Noise Factor
Set 1	1.029	1.083	1.296
Set 2	1.019	1.063	1.261
Set 3	1.032	1.043	1.109

Possible Changes in Effective Area due to CCD Background Contribution						
Set	sigma	noisefactor	Old Ae _{eff} (cm ²) R	Old % of Theory w/ 5A roughness	New Ae _{eff} (cm ²) R*noisefactor	New % of Theory w/ 5A roughness
Set 1	2	1.083	39.1	74	42.3	80
Set 1	3	1.296	38.3	73	49.6	94
Set 2	2	1.063	43.2	82	45.9	87
Set 2	3	1.261	42.2	80	53.2	101

Figure C.11: Noise factor calculations and possible changes in the effective area measurements. Due to significant change in the effective area values, background should be minimized during the experiment.

C.2.6 Conclusions

This analysis shows that the background events produced by the CCD should not be neglected in photon counting experiments with a very low count rate. While

the increase of the event threshold does decrease the background count rate, one must still be aware of this feature. The results presented here still show an inconsistency in the effective area calculation suggesting that a single measurement of the background does not provide all the answers. It is possible that the background changes over time and earlier measurements (such as Sets 1 and 2) cannot be compared to this background set. The very low effective area in Set 3 may be explained by the small ROI used in the incident beam measurement. This module was also a post-vibration test module so the results could have also been degraded due to that test. Another source of error may exist in applying the noise factor calculation to data where the fit to the data could have been better. Another difference that was not tested is the effect the chamber pressure may have on the observed events. During X-ray testing, the pressure is slightly higher in the chamber since the pipe gate is open to the whole beamline.

The results from this analysis can be used during data collection by determining if the background to X-ray ratios lies in the acceptable or unacceptable regions. Figure C.12 shows the ratios related to a noise factor of 1.05 is plotted as the black line and the acceptable ratios are in the blue region. One would have to test the background, incident flux, and focused flux in order to calculate the ratios and determine if an experiment may be conducted.

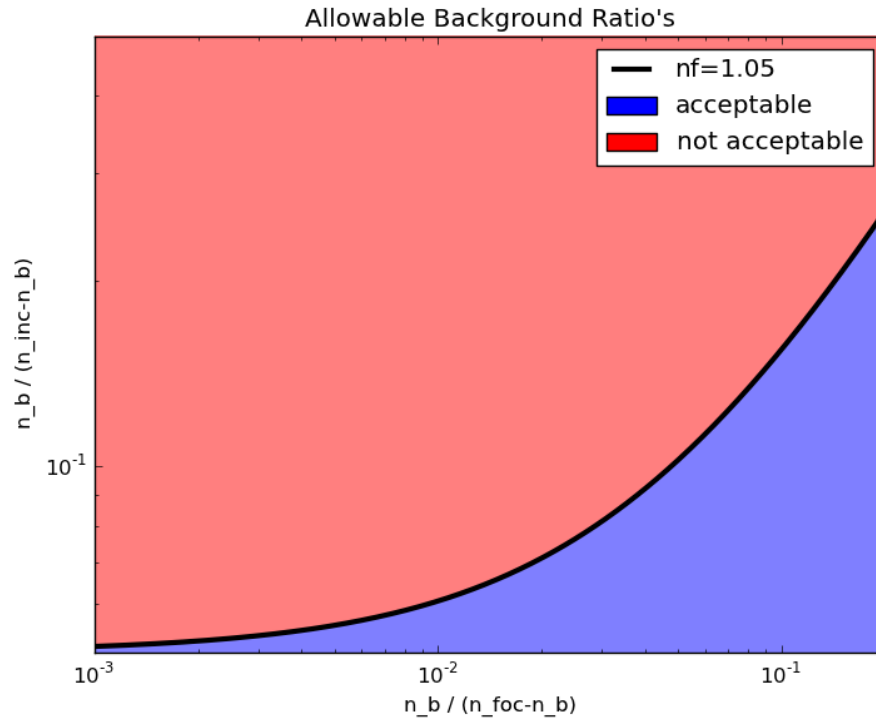


Figure C.12: Acceptable CCD background levels by comparing ratios of number of events. The noise factor (nf) distinguishes between acceptable and unacceptable number of events collected in a measurement in order to properly calculate the effective area. n_b =(number of events in CCD background), n_{foc} =(number of events from XRC focused data), n_{inc} =(number of events from incident beam data)

C.3 Dissecting CCD Pileup Caused by a High Flux X-ray Source

The first concentrated images we collected at the 1.48670 keV energy, using the X-ray source with an Aluminum target, appears to have pile-up (Figure C.13 is an image of the collected data). The intensity at the center of the image plateaus instead of peaks. The flux was set as low as possible. In order to get it lower we need to apply an attenuator which would absorb some of the photons incident on the concentrator. From the collected data we must estimate where the peak would have been in order to calculate the thickness of the attenuator needed.

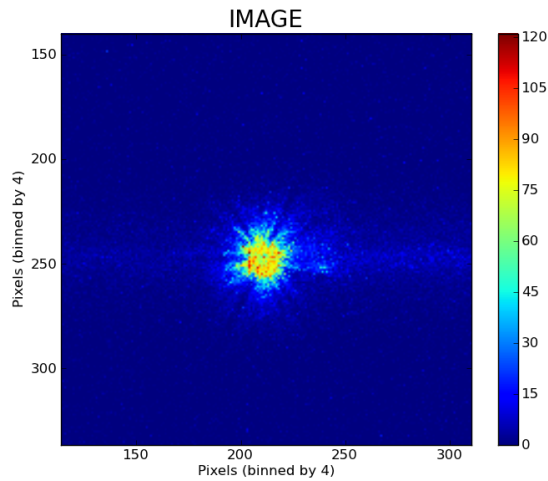


Figure C.13: The image here is an example of pile-up with a hole where the peak should be. This data set is used to estimate the peak and then the required attenuator thickness.

I estimated the peak that we would expect if the collected data did not have pile-up. To do this I first take the data set make a histogram of the event along the center line. I choose to select a line along the readout streak which is 20 pixels thick. I bin the events and create a histogram which allows me to see the shape of the peak with pile-up. I then select a few points around the peak. These points

should not include pile-up and do not extend far from the center (see Figure C.14).

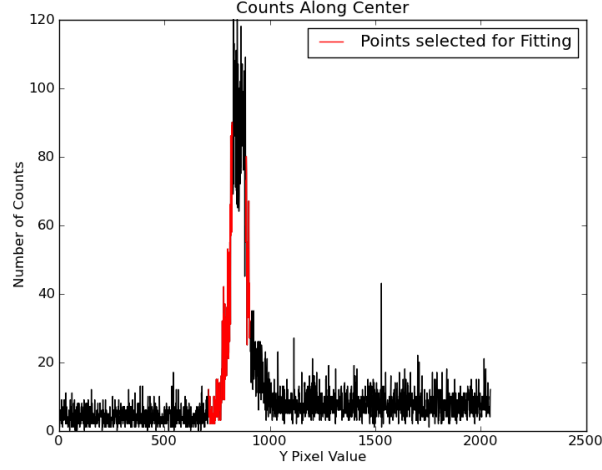


Figure C.14: Histogram of the data points along the center of the image. A few points around the peak are selected to be fitted with a Lorentzian curve in order to estimate the size of the peak if there was no pile-up.

Using the selected points, I apply a Lorentzian fit which is used to model the peaks of concentrated point sources. The Lorentz function I use is displayed in Equation C.4.

$$y(x) = I \frac{\gamma^2}{(x - x_o)^2 + \gamma^2} \quad (\text{C.4})$$

In Equation C.4, I is the height (or peak), γ is the half width at half maximum (HWHM), and x_o is the x location of the peak. By applying SciPy’s “curve_fit”(Jones et al., 2001) function in Python the parameters are found and described in Table C.1. This fit is plotted in Figure C.15 along with the selected data points.

Finally I can find the attenuation factor needed to decrease the flux. The counts at the plateau in the data is average at 89 count per bin. The Lorentzian fit estimates the peak to have 361 counts at the center bin. This is 4.06 times the

Table C.1: Lorentzian fit parameters to model peak from CCD pile-up data

Parameter	Value
I	360.6
γ	16.4
x_o	852.3

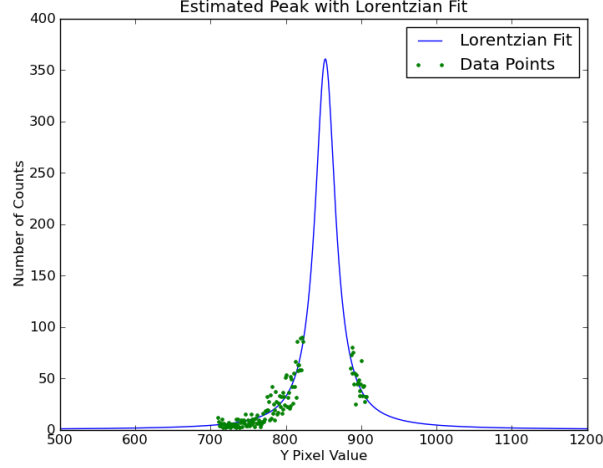


Figure C.15: The selected points in the data set are fitted to a Lorentzian function to estimate what the peak would have been if pile-up was not a factor.

measured amount (see Figure C.16). The flux must be decreased by a factor of 4.06 at the Aluminum edge (1.56 keV).

The transmission of X-ray through a slab of thickness d is describe by Equation C.5.

$$T = e^{-n\mu_a d} \quad (\text{C.5})$$

Here, n is the atomic density (number of atoms per unit volume) of the material and μ_a is the photoabsorption cross section. For a photon of energy 1486.7 eV transmitting through aluminum, these values are 2.70 g/cm^3 and $402.9 \text{ cm}^2/\text{g}$ respectively. Figure C.17 plots the transmission calculated by these values and

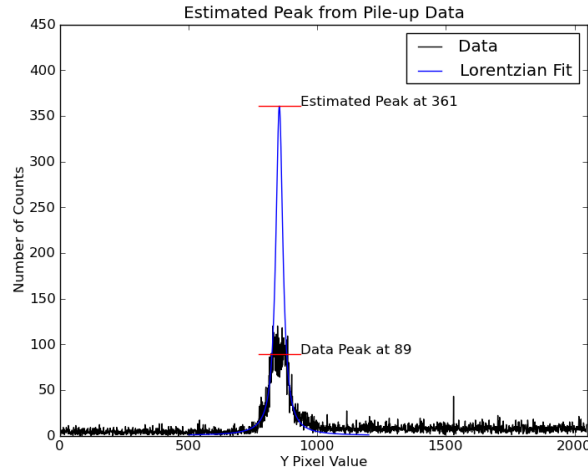


Figure C.16: Here the fitted Lorentzian is plotted on top of the data points. We see that the data plateaued at 89 counts while it would have been at 361 counts. The peak is a factor of 4 larger than the plateau.

Equation C.5 and yields a thickness of 12.9 microns to decrease the flux by a factor of 4.06.

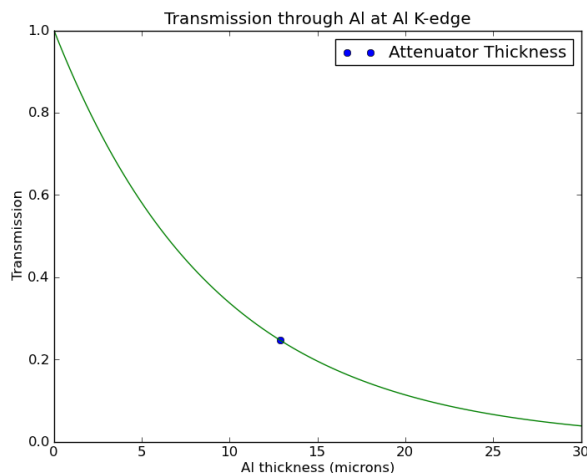


Figure C.17: The calculation attenuation of 1.48670 keV photons through and Aluminum slab. To decrease the flux by a factor of 4.06, the attenuator must have a thickness of 12.9 microns.

Appendix D

XRC Calibration Code

Code I used to analyze the X-ray beamline data.

D.1 XrayAnalysisFunctions.py

The file XrayAnalysisFunctions.py was the core of my code that I used to read and analyze X-ray data collected from the beamline. It contained a start-up script that assigned variables and several functions used in the analysis. Some functions are grouped in separate subsections of this appendix for easier referencing.

D.1.1 Start-up variables and module imports

```
#XrayAnalysisFunctions.py
## also written as XAF for short in this document
##
## Written by Erin Balsamo
## Condensed from original (April 25, 2016)
import os
from Tkinter import *
from tkFileDialog import askopenfilename
import math
from math import *
import matplotlib
from matplotlib.pyplot import fill,between
import matplotlib.gridspec as gridspec
from matplotlib.backends.backend_pdf import PdfPages
import Tkinter, Tkconstants, tkFileDialog, tkMessageBox, tkSimpleDialog
import numpy as np
from scipy.optimize import curve_fit
from scipy import *
from scipy.interpolate import interp1d
from pylab import *
from mpl_toolkits.mplot3d import Axes3D
import cmath
import pickle
from mirror import MirrorImage
from time import time
from datetime import datetime
import sys
from numpy import pi,sin,cos

# Basic Input
## This input defines the features of the CCD, the X-ray source,
## and the interpretation of raw data (i.e. split and grades)

unitsize=14
labsize=20 #x,y label size (16)
titsize=24 #title size (22)
mission = 'NICER'
if mission == 'NICER': fl = 1.085 #meter
if mission == 'XACT': fl = 2.8 #meter
ccdbinning = 1.
DetectorType = 'LN' # 'TEC' for electric cooled CCD
                    # 'LN' for liquid nitrogen cooled CCD
if DetectorType == 'TEC':
    pixszmicron = 13.5
    numberOfPixelsX = 512./ccdbinning
    numberOfPixelsY = 512./ccdbinning
    ccdsz = 27.6 #mm
    exposureTime = 5. #sec
    readoutTime = 2.26549 #sec ROI=512x512: 1.44039s (5 sec exp),
                    # ROI=2048x2048: 2.26549s (5 sec exp)
if DetectorType == 'LN':
    pixszmicron = 20.
    numberOfPixelsX = 1340/ccdbinning
    numberOfPixelsY = 1300/ccdbinning
    ccdsz = 26. #mm -> 26x26.8 LN
pixelsize = math.atan((pixszmicron*ccdbinning/1000)/(fl*1000))*(180/math.pi)*60
pixelsizemm = (pixszmicron/1000.)*ccdbinning #0.0135 for TEC, 0.020 for LN
ccdszX = numberOfPixelsX*pixelsizemm #mm
ccdszY = numberOfPixelsY*pixelsizemm #mm
CrKa = 5.41 #chromium k-alpha line keV
CrKb = 5.95 #chromium k-beta line keV
AlKa = 1.4867 # Aluminum k-alpha emission line in KeV
AlKb = 1.48627 # Aluminum k-beta emission line in KeV
goodgrades = [0,2,8,10,11,16,18,22,64,72,80,104,208]
```

```

badgrades = []
for i in range(256):
    if i not in goodgrades: badgrades.append(i)
totalspokethickness = ((5.*2)+(1.*3.9))
maskedpercentage = 0.
onemillimeter = 1./pixelsizemm
energyConversion = AlKa/455
AuDen=18.53

XAF_Settings = str('XAF Settings\n'+'\n    mission = '+str(mission)+'\n    fl = '+
    str(fl)+' meter'+'\n    ccdbinning = '+str(ccdbinning)+'\n    DetectorType = '+
    DetectorType+'\n    pixszmicron = '+str(pixszmicron)+' microns'+'\n    pixelsize = '+
    str(pixelsize)+' arcmin'+'\n    pixelsizemm = '+str(pixelsizemm)+' mm'+
    '\n    ccdsz = '+str(ccdsz)+' mm'+'\n    ccdszX = '+str(ccdszX)+' mm'+
    '\n    ccdszY = '+str(ccdszY)+' mm'+'\n    numberOfPixelsX = '+str(numberOfPixelsX)+
    '\n    numberOfPixelsY = '+str(numberOfPixelsY)+'\n    goodgrades = '+str(goodgrades)+
    '\n    energyConversion = '+str(energyConversion)+' eV per pulse height'+
    '\n    totalspokethickness = '+str(totalspokethickness)+' mm'+'\n    maskedpercentage = '+
    str(maskedpercentage*100)+'%'+'\n    onemillimeter = '+str(onemillimeter)+' pixels per mm\n\n')

print '          XAF Settings'
print '    mission = ',mission
print '    fl = ',fl,'meter'
print '    ccdbinning = ',ccdbinning
print '    DetectorType = ',DetectorType
print '    pixszmicron = ',pixszmicron,'microns'
print '    pixelsize = ',pixelsize,'arcmin'
print '    pixelsizemm = ',pixelsizemm,'mm'
print '    ccdsz = ',ccdsz,'mm'
print '    ccdszX = ',ccdszX,'mm'
print '    ccdszY = ',ccdszY,'mm'
print '    numberOfPixelsX = ',numberOfPixelsX
print '    numberOfPixelsY = ',numberOfPixelsY
try:
    print '    exposureTime = ',exposureTime,'s'
    print '    readoutTime = ',readoutTime,'s'
except: pass
print '    goodgrades = ',goodgrades
print '    energyConversion = ',energyConversion,'eV per pulse height'
print '    totalspokethickness = ',totalspokethickness,'mm'
print '    maskedpercentage = ',maskedpercentage*100,'% '
print '    onemillimeter = ',onemillimeter,'pixels per mm\n'

```

D.1.2 Load and Select Data

```

def getData(myfile,dontprint=False,beamline='100'):
    """
    Reads in raw data file from X-ray test and formats data to
    be used in analysis.
    Returns:  [frnum,x,y,ph,energy,grade,split,t]

    Variables are defined as follows:
        frnum = Frame number
        x = x location on axis
        y = y location on axis
        ph = pulse height (energy (keV) = ph*3/1000)
        energy = energy in keV
        grade = grade of event (0 is best, max value is 255)
        split = number of extra pixels counted for single event
        t = time stamp (0 if tp (type) 1)
    """
    if beamline == '600':
        if dontprint: print 'energy conversion used: ',energyConversion

```

```

file=myfile
if dontprint: print 'Sorting Data.'
f_out=open('tmpfile.evt','w') # 'w' overwrites file if file exists
f_in=open(file,'r')
framenum=0
tp=0
while tp==0:
    foo = f_in.readline()
    foo_tmp = foo.split()
    if foo_tmp[0] == '#Frame': tp=1
    if foo_tmp[0] == '#':
        if foo_tmp[1] == 'Frame': tp = 3
        else: tp=2
f_in.close()
f_in=open(file,'r')
# tp=1, the data set includes the frame number
if tp==1:
    for i in range(0,file_len(file)-1):
        foo = f_in.readline()
        foo_tmp = foo.split()
        if foo_tmp[0] == '#Frame':
            framenum = int(foo_tmp[1])
            f_out.write(foo)
        else:
            mystr = str(framenum)+' '+str(0)+' '+str(foo)
            f_out.write(mystr)
    f_out.close()
    f_in.close()
    frmnum,t,x,y,ph,grade,split = np.loadtxt('tmpfile.evt', unpack=True)
    energy = np.array(ph)*energyConversion
# tp=2, the data set needs a frame number
if tp==2:
    framenum=0
    for i in range(0,file_len(file)-1):
        foo = f_in.readline()
        foo_tmp = foo.split()
        if foo_tmp[0] == '#':
            framenum = framenum+1
            f_out.write(foo)
        else:
            mystr = str(framenum)+' '+str(foo)
            f_out.write(mystr)
    f_out.close()
    f_in.close()
    frmnum,x,y,ph,grade,split,t = np.loadtxt('tmpfile.evt', unpack=True)
    energy = np.array(ph)*energyConversion
if tp==3:
    try:
        frmnum,x,y,ph,energy,grade,split,t = np.loadtxt(file, unpack=True)
        f_in.close()
        frmTmp, xTmp, yTmp, phTmp, gTmp, sTmp, tTmp = [],[],[],[],[],[],[]
        for i in range(len(frmnum)):
            frmTmp.append(int(frmnum[i]-frmnum[0]+1))
            xTmp.append(int(x[i]))
            yTmp.append(int(y[i]))
            phTmp.append(int(ph[i]))
            gTmp.append(int(grade[i]))
            sTmp.append(int(split[i]))
            tTmp.append(int(t[i]))
        frmnum,x,y,ph,grade,split,t = frmTmp, xTmp, yTmp, phTmp, gTmp, sTmp, tTmp
    except ValueError:
        print 'here'
        framenum=0
        for i in range(0,file_len(file)-1):
            foo = f_in.readline()
            foo_tmp = foo.split()
            if foo_tmp[0] == '#':
                framenum = framenum+1

```



```

        f_out.write(foo)
    else:
        mystr = str(framenum)+' '+str(foo)
        f_out.write(mystr)
f_out.close()
f_in.close()
frmnum,x,y,ph,grade,split,t = np.loadtxt('tmpfile.evt', unpack=True)
frmTmp, xTmp, yTmp, phTmp, gTmp, sTmp, tTmp = [],[],[],[],[],[],[]
for i in range(len(frmnum)):
    frmTmp.append(int(frmnum[i]-frmnum[0]+1))
    xTmp.append(int(x[i]))
    yTmp.append(int(y[i]))
    phTmp.append(int(ph[i]))
    gTmp.append(int(grade[i]))
    sTmp.append(int(split[i]))
    tTmp.append(int(t[i]))
    frmnum,x,y,ph,grade,split,t = frmTmp, xTmp, yTmp, phTmp, gTmp, sTmp, tTmp
    energy = np.array(ph)*energyConversion
if dontprint: print 'Data sorting Complete.'
dataset = [frmnum,x,y,ph,energy,grade,split,t]
for i in dataset: i=np.array(i)
if beamline == '100':
# arrays 'ph', 'split', and 'time' and place holders for the
# 100m beamline file type at this time
    framenum = 0
    f = open(myfile,'r')
    frame, x, y, ph, energy, grade, split, time = [],[],[],[],[],[],[],[]
    for line in list(f):
        myline = line.split()
        if myline == [] or myline[0] == 'Thermal': pass
        elif myline[0] == '#' and myline[1] != 'Frame': pass
        elif myline[0] == '#' and myline[1] == 'Frame': framenum = framenum+1
        else:
            frame.append(framenum)
            x.append(int(myline[0]))
            y.append(int(myline[1]))
            ph.append(int(myline[2]))
            energy.append(int(myline[3]))
            grade.append(int(myline[4]))
            split.append(99)
            time.append(99)
    dataset = [frame,x,y,ph,energy,grade,split,time]
return dataset

def setData(dataset,
    lowframe=None,highframe=None,
    lowx=None,highx=None,
    lowy=None,highy=None,
    lowph=None,highph=None,
    lowenergy=None,highenergy=None,
    grades=range(256),
    lowsplit=None,highsplit=None,
    lowtime=None,hightime=None,
    energyconversion = None,
    selectCircle = False,
    r = 0, xcen=0, ycen=0,
    selectPieSlice = False,
    pieangle = 0, piewidth = 0,
    xcenPie = 0, ycenPie = 0):
''' Data specifications

Modify initial raw data set by frame, x coordinates,
y coordinates, pulse height, energy, grade, split,
time, points within a selected circle, and/or points within
a selecte pie slice of a circle.

input data set: [frmnum,x,y,ph,energy,grade,split,t]
output: list of modified data set

```

```

Other local XAF functions:
    selector() '''
myModifiedData=dataset
## Frame index 0
if lowframe!=None or highframe!=None:
    if lowframe==None: lowframe=min(myModifiedData[0])
    if highframe==None: highframe=max(myModifiedData[0])
    myModifiedData=selector(myModifiedData,lowframe,highframe,0)
## X index 1
if lowx!=None or highx!=None:
    if lowx==None: lowx=min(myModifiedData[1])
    if highx==None: highx=max(myModifiedData[1])
    myModifiedData=selector(myModifiedData,lowx,highx,1)
## Y index 2
if lowy!=None or highy!=None:
    if lowy==None: lowy=min(myModifiedData[2])
    if highy==None: highy=max(myModifiedData[2])
    myModifiedData=selector(myModifiedData,lowy,highy,2)
## Pulse Height index 3
if lowph!=None or highph!=None:
    if lowph==None: lowph=min(myModifiedData[3])
    if highph==None: highph=max(myModifiedData[3])
    myModifiedData=selector(myModifiedData,lowph,highph,3)
## Energy index 4
if lowenergy!=None or highenergy!=None:
    if lowenergy==None: lowenergy=min(myModifiedData[4])
    if highenergy==None: highenergy=max(myModifiedData[4])
    myModifiedData=selector(myModifiedData,lowenergy,highenergy,4)
## Grades index 5
if grades!=range(256):
    Data3=[[],[],[],[],[],[],[],[]]
    for i in grades:
        Data2=selector(myModifiedData,i,i,5)
        for j in range(len(myModifiedData)):
            Data3[j]=np.array(list(Data3[j])+list(Data2[j]))
    myModifiedData=Data3
## SPLIT index 6
if lowsplitlet!=None or highsplitlet!=None:
    if lowsplitlet==None: lowsplitlet=min(myModifiedData[6])
    if highsplitlet==None: highsplitlet=max(myModifiedData[6])
    myModifiedData=selector(myModifiedData,lowsplitlet,highsplitlet,6)
## Timestamp index 7
if lowtime!=None or hightime!=None:
    if lowtime==None: lowtime=min(myModifiedData[7])
    if hightime==None: hightime=max(myModifiedData[7])
    myModifiedData=selector(myModifiedData,lowtime,hightime,7)
## Circle selection
if selectCircle:
    tmpdat = []
    for i in range(len(myModifiedData)):
        tmpdat.append([])
    for i in range(len(myModifiedData[0])):
        val = (float(myModifiedData[1][i])-float(xcen))*2 + (float(myModifiedData[2][i])-float(ycen))*2
        if val <= float(r)**2:
            for j in range(len(myModifiedData)):
                tmpdat[j].append(myModifiedData[j][i])
    myModifiedData = tmpdat
## Energy Conversion
if energyconversion != None:
    myModifiedData[4] = myModifiedData[3]*energyconversion
## Pie Slice Selection
if selectPieSlice:
    def magnitude(x,y): return np.sqrt(x**2+y**2)
    def CoordAngle(x,y):
        a = np.arccos(x/magnitude(x,y))
        if x==0 and y==0: a = 0 # origin
        if y>=0: a = a # quadrant 1 + 2

```

```

        if y<0: a = 2*np.pi - a # quadrant 3 + 4
        return a
    leadingAngle = np.radians(pieangle) + np.radians(piewidth)/2.
    trailingAngle = np.radians(pieangle) - np.radians(piewidth)/2.
    tmpdat = []
    for i in range(len(myModifiedData)):
        tmpdat.append([])
    for i in range(len(myModifiedData[0])):
        x1 = float(myModifiedData[1][i])-float(xcenPie)
        y1 = float(myModifiedData[2][i])-float(ycenPie)
        if trailingAngle < CoordAngle(x1,y1) <= leadingAngle:
            for j in range(len(myModifiedData)):
                tmpdat[j].append(myModifiedData[j][i])
        elif trailingAngle+2.*np.pi < CoordAngle(x1,y1) <= leadingAngle+2.*np.pi:
            for j in range(len(myModifiedData)):
                tmpdat[j].append(myModifiedData[j][i])
        else: pass
    myModifiedData = tmpdat
    return myModifiedData

def selector(dataArray,myMin,myMax,indexOfDataArray):
    '''Select which data to look at based on a min and a max
    whether it be location, pulse height, grade, split, or time
    and outputs the new sedef selt of data

    dataArray: form of a list of arrays. [x,y,ph,grade,split,t]
    myMin: lower limit of the set of data you are selecting (ie- max ph)
    myMax: upper limit of the set of data you are selecting (ie- min ph)
    indiceOfDataArray: index of the array you want to select from in
    dataArray (ie. [2] for ph) '''
    DA=np.array(dataArray)
    dat=indexOfDataArray
    if myMin!=myMax:
        ind1 = np.where(DA[dat]>=myMin)
        tmpData=[]
        for i in range(len(DA)):
            tmpData.append(DA[i][ind1])
        ind2 = np.where(tmpData[dat]<=myMax)
        newData=[]
        for i in range(len(tmpData)):
            newData.append(tmpData[i][ind2])
    if myMin==myMax:
        ind1 = np.where(DA[dat]==myMin)
        newData=[]
        for i in range(len(DA)):
            newData.append(DA[i][ind1])
    return newData

```

D.1.3 EEf Functions

```

def eef(x,y,cx,cy,res,shape='circle',maxradius=0,printcount=True,printinfo=True):
    ''' Find the encircled energy at each radius given the x an y values,
    center, and number of points between each radii (res)

    Can choose circle or semi-circle for area to use in calcuating the EEf.

    Other local XAF functions:
        enclosedEnergy() '''
    if shape == 'circle':
        if printinfo:print abs(max(x)-cx),abs(min(x)-cx),abs(max(y)-cy),abs(min(y)-cy)
        c = min(abs(max(x)-cx),abs(min(x)-cx),abs(max(y)-cy),abs(min(y)-cy))# max radius
        # chooses smallest distance because it would be edge of CCD collecting area
    if shape == 'semi':

```

```

        c = min(abs(max(x)-cx),abs(max(y)-cy),abs(min(y)-cy))# max radius
if maxradius>0: c=maxradius
if printinfo:print 'max radius = ',c,'pixels'
p = int(c/res) #number of radii
r = []
cts = []
inc = 0
for i in range(p):
    whichradius=res*(i+1)
    cts.append(enclosedEnergy(cx,cy,x,y,whichradius,shape))
    if printcount:
        if whichradius%50==0:
            print 'at radius ',whichradius,'pixels. ',round(whichradius/float(c)*100),'% done.'
        r.append(whichradius)
return r,cts

def enclosedEnergy(centerX,centerY,x,y,r,shape='circle'):
    ''' Calculates points inside a specific radius.
    User can choose circle, semi-circle, or square as shape of
    enclosed area. '''
    counts=0
    if shape == 'circle':
        for i in range(len(x)):
            if math.sqrt((x[i]-centerX)**2+(y[i]-centerY)**2)<r:
                counts = counts+1
    if shape == 'semi':
        for i in range(len(x)):
            if x[i] > centerX:
                if math.sqrt((x[i]-centerX)**2+(y[i]-centerY)**2)<r:
                    counts = counts+1
        counts=counts*2
    if shape == 'square':
        width=r*2. # width = width of box
        x0=centerX-r # x0 = x corner of box
        y0=centerY-r # y0 = y corner of box
        # x = x data
        # y = y data
        counts=0
        for i in range(len(x)):
            if x[i]>x0:
                if x[i]<x0+width:
                    if y[i]>y0:
                        if y[i]<y0+width:
                            counts = counts+1
    return counts

```

D.1.4 Effective Area Functions

```

def XRC_Aeff(shellNum=[], roughnessRMS=0,
    densityFile = 'IndexOfRefraction_30-10000eV_D18,53.txt'):
    ''' Calculates the effective Area of 1 X-ray concentrator module

```

Input needed:

- (1) Mirror Geometry
 - Includes:
 - (a) Surface Profile/Curvature
 - (b) Top and Bottom Radii
 - (c) Mission Name
 - (d) Shells in module
 - (e) Roughness of Surface
 - (f) Spoke and Mask
- (2) Energy Dependant Reflectivity (gold)
 - Includes:

```

        (a) Fresnel Equation
        (b) Index of Refraction (alpha nad beta)
        (c) Energies      '''
#Functions
def surface(mirrornumber,steps):
    ''' returns inner and outer radii and ANGLE for each 'step'
    of mirror size 'mirrornumber' '''
    co = [0.1051786,0.1179908,0.1322073,0.1479731,0.1654474,0.1848059,
          .2062414,0.2299663,0.2562142,0.2852418,0.3173317,0.3527946,
          0.3919719,0.4352392,0.4830093,0.5357359,0.5939178,0.6581032,
          0.7288944,0.8069533,0.8930071,0.9878548,1.0923738,1.2075282]
    a = co[int(mirrornumber)-1] #parabolic coeff of mirror
    foilheight = 101.6
    stepsize = foilheight/steps
    angle, ro, ri = [], [], []
    epoxythickness, foclen = 0.0127, 1085.
    FtoOutlet = (foclen - ( foilheight /2.))
    def radiusToSurface(h): return sqrt(a**2 + 2.*h*a) + epoxythickness
    for i in range(steps):
        yo, yi = i*stepsize, (i+1)*stepsize #Distance to outlet plane
        yc = (yo+yi)/2.
        hi, ho, hc = yi + FtoOutlet, yo + FtoOutlet, yc + FtoOutlet
        #Distance to focal plane
        ri.append( radiusToSurface(hi) ) #radius inlet side of segment
        ro.append( radiusToSurface(ho) ) #radius outlet side of segment
        angle.append( degrees(atan( a / radiusToSurface(hc) ) ) )
        #angle at center of segment
    return angle,ro,ri

def Reflectance(angle,sigma=0, denFile = densityFile):
    '''Reflectance vs. Energy for a single angle with RMS = sigma (angstrom)

    Index of Refraction data:
        'IndexOfRefraction_100-15000eV.txt'
        optical constants delta and beta
        for 500 energies 100 to 15000 eV
        with GOLD surface
        NO POLARIZATION

    Fresnel Equations:
        sqrt[ Rp^2 + Rs^2 ]
        R = ----- For unpolarized light
            2

        n^2 sin(th) - sqrt[ n^2 - cos(th)^2 ]
        Rp = ----- P-polarized
        n^2 sin(th) + sqrt[ n^2 - cos(th)^2 ]

        sin(th) - sqrt[ n^2 - cos(th)^2 ]
        Rs = ----- S-polarized
        sin(th) + sqrt[ n^2 - cos(th)^2 ]

    Reflectivity with Roughness"
    R = Ro * e^( - (4*pi*sigma*sin(th)/lambda)^2 ) '''
    if AuDen == 19.3:
        energy, delta, beta = np.loadtxt('/IndexOfRefraction_30-10000eV.txt',

    if AuDen == 18.53:
        energy, delta, beta = np.loadtxt(denFile, unpack=True)
    energy = energy/1000. #keV
    th = math.radians(angle)
    n = (1-delta) + (1j * beta)
    rptop=n*n*sin(th)-np.sqrt(n*n-cos(th)*cos(th))
    rpbot = n*n*sin(th)+np.sqrt(n*n-cos(th)*cos(th))
    rp=rptop/rpbot
    rstop=sin(th)-np.sqrt(n*n-cos(th)*cos(th))
    rsbot=sin(th)+np.sqrt(n*n-cos(th)*cos(th))
    rs=rstop/rsbot

```

```

    reflectivity = (rp*rp.conjugate()+rs*rs.conjugate()).real/2
    #roughness
    hc = (4.1357*(10.**-18)) * (3.*(10.**18)) #keV, Angstrom/s
    expTop = (4.*pi*sigma*sin(radians(angle)))
    expBot = (hc/energy)
    reflectivity = reflectivity *np.exp( -1* ( expTop / expBot )**2 )
    return energy,reflectivity
# NICER
Aeff_XRC = 0
A=0 #after spoke and mask subtraction
totarea = 0 #total geometric area
if shellNum == []: shellNum = [1,2,3,4,5,6,7,8,9,10,
                               11,12,13,14,15,16,17,
                               18,19,20,21,22,23,24]
for i in range(len(shellNum)): #find effective area of each shell
    angle, ro, ri = surface(shellNum[i],100)
    # The parabolic surface is composed of 100 flat surfaces
    Aeff_shell = 0
    for j in range(len(angle)):
        energy, reflectivity = Reflectance(angle[j], sigma = roughnessRMS)
        #reflectance of small, flat, segment of one shell
        area = (pi*ri[j]**2) - (pi*ro[j]**2)
        #ring area of the shell segment, ri>ro b/c inlet radius > outlet radius
        totarea = totarea+area
        area = area - (totalspokethickness*(ri[j]-ro[j]))
        area = area * (1.-maskedpercentage)
        A=area+A
        Aeff_seg = area*reflectivity #effective area shell segment
        Aeff_shell = Aeff_shell + Aeff_seg # sum the segments Aeff
    Aeff_XRC = Aeff_XRC + Aeff_shell # sum the shells Aeff
Area_XRC = A
return energy, Aeff_XRC, Area_XRC

def measuredAeff(numEventsFP,time,DBperMM2perTIME,atEnergy=AlKa,roughness=0,
                 mission='NICER',
                 foils=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24],
                 AuDensityFile = 'IndexOfRefraction_30-10000eV_D18,53.txt'):
    ''' Effective Area = geometric area * total photons at FP / total incident photons

    return Aeff, percentOfTheory, theoryAtEnergy, theory, energies, Area_xrc

    aef = measuredAeff( <parameters> )
    Output:
    aef[0] = Effective Area mm^2
    aef[1] = Fraction of Theory
    aef[2] = Analytical Effective Area at energy mm^2
    aef[3] = Analytical Effective Area Array
    aef[4] = Energy Array
    aef[5] = Incident Area of XRC (obstruction subtracted) '''
    energies, theory, Area_XRC = XRC.Aeff(shellNum=foils,
                                           roughnessRMS = roughness,
                                           densityFile = AuDensityFile)

    Aeff = numEventsFP/(DBperMM2perTIME*time)
    theoryAtEnergy = findY(energies,theory,atEnergy)
    percentOfTheory = Aeff/theoryAtEnergy
    return Aeff, percentOfTheory, theoryAtEnergy, theory, energies, Area_XRC

def AeffError(Nxrc, Ndb, Txrc, Axrc, Tdb, Adet):
    ''' input:
    Nxrc = number of photons from concentrator
    Ndb = number of photons for direct beam
    Txrc = Length in time in concentrator data
    Axrc = Area of XRC
    Tdb = Length in time in XRC data
    Adet = area of detector for incident beam data '''
    Nxrc_err = sqrt(Nxrc)
    fluxdb_err = sqrt(Ndb)/Tdb/Adet
    n = Ndb/Tdb/Adet

```

```

Ni = n*Tsrc*Axrc
Ni_err = fluxdb_err*Tsrc*Axrc
return Axrc * sqrt( ( (Nxrc_err/Nxrc)**2 + (Ni_err/Ni)**2 ) ) * (Nxrc/Ni)

```

D.1.5 Other Functions

```

def ETUeef(data,shape='circle',getpsf=True,
           eeftitle='EEF',psftitle='PSF',
           expand=[],furtherextension=False,
           binfactor=1,res=1,save=False,savename='',
           cx=None,cy=None,checkcenter='no',
           limit=None,side=None,colorscheme='jet',
           maskreadout=False,beforeaftermask=False,
           maskpoints=[0,0,0,0],maskededge=[0,2048],
           MaxR=0,showframecount=False,makeplot=True,
           trackcalculation=True,printinfo=True,showSDD=True,
           show80=False,show90=True,showHPR=True,
           makefigs=True, arcminonly=True, getNumCts=False):
''' Find the EEf and PSF of a dataset using the user input dataset
with x values as index 1 and y values as index 2.

```

User can choose the resolution (number of radii per data point), maximum radius, and center of image, resulting in different results.

Specialized user options include masking CCD read out, extending the CCD region, making plots/images of the results, tracking the EEf calculation in the command line as standard output, and shape around center of image to use for EEf calculation.

Parameters:

```

data
shape='circle'
getpsf=True
eeftitle='EEF'
psftitle='PSF'
expand=[]
furtherextension=False
binfactor=1
res=1
save=False
savename=''
cx=None
cy=None
checkcenter='no'
limit=None
side=None
colorscheme='jet'
maskreadout=False
beforeaftermask=True
maskpoints=[0,0,0,0]
maskededge=[0,2048]
MaxR=0
showframecount=False
makeplot=True
trackcalculation=True
printinfo=True
showSDD=True
show80=False
show90=True
showHPR=True
makefigs=True
arcminonly=True

```

```

    getNumCts=False

Function returns:
    r, ct, psf if arcminonly = True
    OR
    r, ct, psf, rpix, rmm if arcminonly = False

Other local XAF functions:
    findCenter()
    checkCenter()
    extendData()
    HtoXY()
    maskReadout()
    eef()
    normalizeCounts()
    convertList() '''
x=data[1]
y=data[2]
#Center
if cx==None or cy==None:
    if checkcenter=='no':
        cx,cy=findCenter(x,y,doprint=printinfo)
    if checkcenter=='yes':
        cx,cy=checkCenter(x,y)
#Copy Data - expand region
if expand==[]:H=np.histogram2d(x,y,[max(x),max(y)])[0]
else:
    H=extendData(expand[0],expand[1],binfactor=binfactor,extendmore=furtherextension)
    x,y=HtoXY(H)
#Masking Readout
if maskreadout==True: x,y=maskReadout(x,y,maskpoints,maskedge,beforeaftermask=beforeaftermask,save=save)
# find eef
a,b=x,y
if cx==None or cy==None:ca,cb=findCenter(a,b)
else: ca,cb=cx,cy
r,ct = eef(a,b,ca,cb,res,shape=shape,maxradius=MaxR,printcount=trackcalculation,printinfo=printinfo)
if printinfo: print 'Total Counts in Integration =',ct[-1]
if ct[-1] == 0:
    print 'Cannot Compute EEF'
    return [[0],[0],[0],[0],[0]]
else:
    numcts = ct
    ct=normalizeCounts(ct)
    # convert radii to arcmin and millimeter
    rpix = r
    r = convertList(rpix,pixelsize)
    rmm = convertList(rpix,pixelsizemm)
    # Plot EEF figure
    if makefigs:
        figure()
        plt.axis([min(r),max(r),min(ct),max(ct)+.2])
        plt.plot(r,ct,'k')
        at10, at20, at30, at40, at50, at60, at70, at80, at90, atSDD = findX(r,ct,.1), findX(r,ct,.2),
            findX(r,ct,.3), findX(r,ct,.4), findX(r,ct,.5), findX(r,ct,.6), findX(r,ct,.7),
            findX(r,ct,.8), findX(r,ct,.9), findY(r,ct,arcmin(1.,1085.))
        val=round(atSDD*100.,0)
        if showSDD: plt.plot([0,arcmin(1.,1085.),arcmin(1.,1085.)],[atSDD,atSDD,0],'b',
            linestyle='solid',label="Detector: Diameter 2mm\n{:0.0f}% Power Encircled".format(val))
        if showHPR: plt.plot([0,at50,at50],[.5,.5,0],'r',linestyle='dashed',
            label="50% radius at {:0.2f}'".format(at50))
        if show80: plt.plot([0,at80,at80],[.8,.8,0],'r',linestyle='dotted',
            label="80% radius at {:0.2f}'".format(at80))
        if show90: plt.plot([0,at90,at90],[.9,.9,0],'r',linestyle='dotted',
            label="90% radius at {:0.2f}'".format(at90))
        plt.title(eeftitle,size=titsize)
        plt.xlabel('Radius (arcmin)',size=labsize)
        plt.ylabel('Encircled Energy Function',size=labsize)
        plt.legend(loc=4)

```



```

        if save==False:plt.show()
        if save==True:
            filename=file[17:-4]
            plt.savefig('MyFigs/'+filename+'_EETUeef(EF'+savename)
            clf()
    if printinfo: print at10, '10%\n',at20, '20%\n',at30, '30%\n',at40, '40%\n',at50,
        '50%\n',at60, '60%\n',at70, '70%\n',at80, '80%\n',at90, '90%\n',atSDD,
        '% within SDD\n','HPD = ',at50*2

# PSF
if getpsf:
    # r is in arcmin
    psf, ctAtR, areaAtR = [], [], []
    for i in range(len(numcts)):
        if len(psf)==0:
            ctAtR.append( numcts[i] )
            areaAtR.append( math.pi*r[i]**2 )
        if len(psf)>0:
            ctAtR.append( numcts[i]-numcts[i-1] )
            areaAtR.append( (math.pi*(r[i])**2) - (math.pi*(r[i-1])**2) )
        psf.append(ctAtR[-1]/areaAtR[-1])
    psf = normalizeCounts(psf)
    # Plot PSF figure
    if makefigs:
        figure(figsize=(10,6))
        subplot(131)
        plt.plot(r,psf,'k')
        if showSDD:
            plt.plot([arcmin(1.,1085.),arcmin(1.,1085.)],[0,max(psf)],
                'r',label='Silicon Drift Detector')
            plt.legend()
        plt.title(psftitle,size=titsize)
        plt.xlabel('Radius (arcmin)',size=labsize)
        plt.ylabel('Counts per Area',size=labsize)
        subplot(132)
        plot(r, ctAtR)
        title('Counts',size=titsize)
        xlabel('Radius (arcmin)', size=labsize)
        ylabel('Counts at R', size=labsize)
        subplot(133)
        plot(r, areaAtR)
        title('Area',size=titsize)
        xlabel('Radius (arcmin)', size=labsize)
        ylabel('Area at R', size=labsize)
        if save==False:plt.show()
        if save==True:
            filename=file[17:-4]
            plt.savefig('MyFigs/'+filename+'_PSF'+savename)
            clf()

#Return data from EEF and PSF
if arcminonly: return r,ct,psf
else:
    if getNumCts: return r,ct,psf,rpix,rm, numcts
    else: return r,ct,psf,rpix,rm

close()

def arcmin(width,distance):
    ''' Converts a width to arcmin given a distance '''
    width, distance = float(width), float(distance)
    return math.atan(width/distance)*(180/math.pi)*60

def checkCenter(x,y,centerX=None,centerY=None):
    ''' Allows user to vizually check the center of a histogram when the
    center is found using findCenter()

    Other local XAF functions:
        findCenter()
    cx,cy=centerX,centerY
    if cx==None or cy==None:

```

```

    cx,cy=findCenter(x,y)
    print "cx =",cx,"\ncy =",cy
    agree=raw_input('Do you agree with center? ')
    if agree == 'no':
        seeing = raw_input('Would you like to see an image? ')
        if seeing=='yes':
            H,xedges,yedges=np.histogram2d(x,y,[(max(x)-min(x))-1,(max(y)-min(y))-1])
            figure()
            imshow(H)
            show()
            cx = int(raw_input('Enter X Center '))
            cy = int(raw_input('Enter Y Center '))
    else:
        print "cx =",cx,"\ncy =",cy
    return cx,cy

def HtoXY(H):
    ''' Takes a histogram and turns it into a list of every x and y. '''
    x,y = [],[]
    for i in range(len(H[0])):
        for j in range(len(H)):
            amount = int(H[j,i])
            for n in range(amount):
                x.append(j)
                y.append(i)
    return x,y

def extendData(limit,side,binfactor=1,extendmore=False):
    ''' Extend the CCD area by copying over data points if the center of
    the image was too close to the CCD. (This feature should not be used
    in acutal calibration.) '''
    def copyData(H,limit,side=0):
        if side == 0:
            HH=H
        if side == 1:
            HH=H[:,::-1].transpose()
        if side == 2:
            HH=H[:,::-1]
        if side == 3:
            HH = H.transpose()
        a=HH[:,0:limit] #slects the 1st limit rows in HH that will be copied over
        a=a[:,::-1] #reverse the rows so they can append to the end of image HH
        Hnew=np.zeros((len(HH),len(HH[0])+len(a[0]))) #creates array of zeros with
                                                    # correct size for extended array
        for i in range(len(HH)):
            Hnew[i]=np.append(HH[i],a[i])
            #changes the Hnew array from zeros to values
            #each row in a is appended onto the rows in HH
        return Hnew
    if limit==None:
        limit = int(raw_input('What is the limit (enter int)?\n(Choose cx*2,cy*2,(max(x)-cx)*2, or (max(y)-cy)*2)\n'))
    if side==None:
        side = int(raw_input('Which side is the center on?\nright choose "0"\ntop choose "1"
        \nleft choose "2"\nbottom choose "3"\n'))
    h=np.histogram2d(x,y,[max(x)/binfactor,max(y)/binfactor])[0]
    H=copyData(h,limit,side=side)
    if extendmore==True:
        imshow(H)
        title('note new limit')
        show()
        limit = int(raw_input('What is the NEW limit (enter int)?'))
        side = int(raw_input('What is the NEW side (enter int)?'))
        H=copyData(H,limit,side=side)
    return H

def maskReadout(x,y,maskpoints,maskedge,beforeaftermask=True,save=False):
    ''' Masks data points either by a rectangle around the center (maskpoints)
    or at the edge of the CCD (maskedge). Created image in requested.

```

```

maskpoints = [xmin,xmax,ymin,ymax] (point around the center)
maskededge = [xmin,xmax] (boundries on data outside of region) '''
a,b=x,y
mp=maskpoints
me=maskededge
a2=[]
b2=[]
for i in range(len(a)):
    if a[i]<=mp[0]:
        if a[i]>me[0]:
            a2.append(a[i])
            b2.append(b[i])
    if a[i]>=mp[1]:
        if a[i]<me[1]:
            a2.append(a[i])
            b2.append(b[i])
    if a[i]>mp[0]:
        if a[i]<mp[1]:
            if b[i]>mp[2]:
                if b[i]<mp[3]:
                    a2.append(a[i])
                    b2.append(b[i])
if beforeaftermask==True:
    plt.subplot(121)
    plt.plot(a,b,'k.')
    plt.title('Before Mask')
    plt.axis([min(a),max(a),min(b),max(b)])
    plt.subplot(122)
    plt.plot(a2,b2,'k.')
    plt.title('After Mask')
    plt.axis([min(a),max(a),min(b),max(b)])
    if save==False: plt.show()
    if save==True:
        filename=file[17:-4]
        plt.savefig('MyFigs/'+filename+'__b4afMask_'+str(mp))
        clf()

a=a2
b=b2
return a,b

def normalizeCounts(counts):
    ''' Normalized list to max value of list given '''
    cnew=[]
    for i in range(len(counts)):
        cnew.append(float(counts[i])/float(max(counts)))
    return cnew

def convertList(x,conversion):
    ''' Multiplies every value in the list by conversion and return xnew '''
    conversion=float(conversion)
    xnew=[]
    for i in range(len(x)):
        xnew.append(x[i]*conversion)
    return xnew

def findX(x,y,yval,findAll=False):
    ''' Find X when Y is about a select value.
    Y does not have to contain that value. '''
    curve = interp1d(y,x)
    xval = curve(yval)
    return xval

def findY(x,y,xval,findAll=False):
    ''' Find Y when X is about a select value.
    X does not have to contain that value. '''
    curve = interp1d(x,y)
    yval = curve(xval)

```

```

    return yval

def framecount(data,doMeans=False,binSize=1,doprint=False):
    ''' Get number of points per frame in a data set. Can choose bin size
    of frames. Can print and/or output mean and standard deviation if
    requested.

    Other local XAF functions:
        meanstdv() '''
    f=data[0]
    histmin=min(f)
    histmax=max(f)+1
    numbins=((histmax-histmin)/binSize)+1
    count,frame=np.histogram(f,linspace(histmin,histmax,numbins))
    #np.histogram(...) returns hist,bin-edges
    if doMeans:
        mean,std=meanstdv(count)
        if doprint:
            print 'Average Counts Per ',binSize,' Frame(s) =',mean
            print 'Standard Deviation of Frame bin-counts =',std
            print 'Total Counts =',len(f)
        return frame[:-1],count, mean, std
    return frame[:-1],count

def meanstdv(x):
    ''' Find mean and standard deviation of input array '''
    if len(x)==1:mean,std=-1,-1
    else:
        from math import sqrt
        n, mean, std = len(x), 0, 0
        for a in x:
            mean = mean + a
        mean = mean / float(n)
        for a in x:
            std = std + (a - mean)**2
        std = sqrt(std / float(n-1))
    return mean, std

def HistPlot(myArray,binSize,Nmin=None,Nmax=None,centerize=True):
    ''' returns x and y values for a histogram with x as bins and
    y and number of counts '''
    if Nmin==None: histmin=min(myArray)
    else: histmin=Nmin
    if Nmax==None: histmax=max(myArray)+1
    else: histmax=Nmax+1
    numbins=((histmax-histmin)/binSize)+1
    y,binedges=np.histogram(myArray,linspace(histmin,histmax,numbins))
    if centerize: centers = 0.5*(binedges[1:]+binedges[:-1])
    if not centerize: centers = (binedges[1:]+binedges[:-1])
    return centers,y

def boxArray(x,y,center=False):
    ''' boxArray(x,y,center=False)
    return xx, yy
    takes array x and places points in the middle of
    each point so that the plot looks more like a
    histogram (i.e. 'box') than a scatter plot with line. '''
    xx, yy, dif = [], [], np.ediff1d(x)
    dif = append(dif,dif[-1])
    xprim = np.array(x)+np.array(dif)
    for i in range(len(x)):
        if center:
            half = (xprim[i]-x[i])/2.
            xx.append(x[i]-half)
            xx.append(xprim[i]-half)
        else:
            xx.append(x[i])
            xx.append(xprim[i])

```

```

        yy.append(y[i])
        yy.append(y[i])
    xx, yy = np.array(xx), np.array(yy)
    return xx, yy

def findCenter(x,y, doprint=True):
    '''X and Y array and finds the mean based on number of points.
    converts arrays using HistPlot in order to do so.

    Other local XAF functions:
        HistPlot() '''
    a,b=HistPlot(x,1)
    a=a-.5
    centx = a[int(mean(list(np.where(b==max(b))[0])))]
    a,b=HistPlot(y,1)
    a=a-.5
    centy = a[int(mean(list(np.where(b==max(b))[0])))]
    del a,b
    if doprint: print 'center at',centx,centy
    return centx,centy

def gauss(x, A, mu, sigma):
    '''A1, mu1, sigma1 = p'''
    return A*np.exp(-(x-mu)**2/(2.*sigma**2))

```

Appendix E

Simulated Observation Code

This Appendix includes how I determined X-ray background negligibility (Section E.1) followed by the code and scripts (Sections E.2, E.3, and E.4) I wrote to conduct the simulations in Chapter 5.

E.1 Sky X-ray Background

The background X-rays are not considered in the Chapter 5 simulations because I considered them negligible since the pulsar is $\sim 100\times$ brighter than the background. I used the X-ray Background Tool (Sabol, E. J., 2017; Snowden et al., 1997), WebPIMMS (Mukai, 1993; Mukai, K., 2017), and the HEASARC Archive (NASA HEASARC Database Group, 2016) provided by NASA for this calculation. The X-ray background tool uses ROSAT PSPC data to generate an estimated background calculation. WebPIMMS is mission count rate simulator (PIMMS v4.8d) made for use in proposal preparation. The HEASARC Archive is a database of astronomy mission data.

Using the X-ray Background Tool, I found the average count rate for a 1° radius cone around PSR J0437-4715. The background has an average count rate of 1.544 ± 0.061 ct/s in the R67 energy band for ROSAT PSPC (0.76–2.04 keV). The area used is shown by Figure E.1.

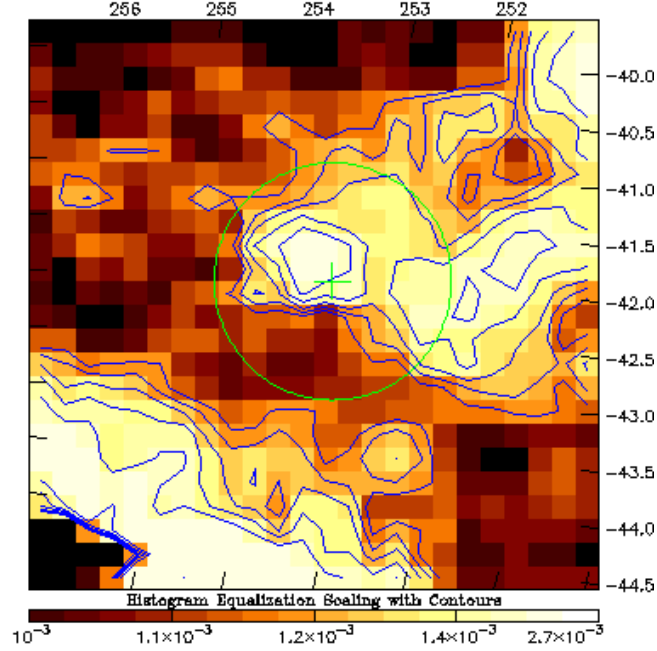


Figure E.1: A 1° cone is shown (green circle) around PSR J0437-4715 (green cross) from ROSAT PSPC data. This area was used to calculate an average X-ray background of 1.544 ± 0.061 ct/s for the 0.76–2.04 keV energy band. The image was generated by the X-ray Background Tool (Sabol, E. J., 2017).

Using WebPIMMS, this count rate correlates to a flux of 3.99×10^{-12} ergs/s/cm² (assuming $N_H = 1.03 \times 10^{20}$ cm⁻², reported by the X-ray background tool based on the Leiden/Argentine/Bonn N_H map). When the angular radius is divided out of this value ($\pi \times 60'^2 = 11376'^2$), the brightness is then 3.51×10^{-16} ergs/s/cm²/arcmin². Multiply this value by 30 arcmin² (the NICER FOV according to the NICER homepage (NASA, NICER, 2017)) and the total flux observed by the background is estimated to be 1.05×10^{-14} ergs/s/cm².

The ROSAT PSPC data reports a count rate of 2.02×10^{-1} ct/s for PSR J0437-4715 in the 0.1–2.0 keV energy band³⁰. Using WebPIMMS, this correlates to 1.53×10^{-12} ergs/s/cm² for 0.76–0.2.04 keV. Therefore, the pulsar is $\sim 100\times$ brighter

³⁰Observation name: 2RXP J043715.8-471509 from the ROSPSPC catalog

than the background, allowing the background to be negligible in this simulation.

E.2 Pulsar Table Model

E.2.1 PulsarTableModelMaker.py

Source File: PulsarTableModelMaker.py

```
# PulsarTableModelMaker.py
# Written by Erin Balsamo, 2016
#
# This script is used to create table data text files that are then converted
# into XSPEC table models via command flx2tab. This script also contains the
# code to create figures found in my dissertation.
# The pulsar modeled in this code is PSR J0437-4715. Emergent spectra data
# was created separately using McPHAC.
# For information on flx2tab, please go to the following NASA website:
# https://heasarc.gsfc.nasa.gov/ftools/caldb/help/flx2tab.html
# For information on McPHAC, visit the following url for a link to the article:
# http://adsabs.harvard.edu/abs/2012ApJ...749...52H

# Import modules
# NeutronStar.py and Flux.py were written by Erin Balsamo
import numpy as np
import math
from scipy import interpolate
from scipy import integrate
import NeutronStar as NS
import Flux
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec

# PSR J0437-4715 input values
# Values based on Bogdanov's assumptions and results found in article:
# http://adsabs.harvard.edu/abs/2013ApJ...762...96B
# Bogdanov, Slavko
# "The Nearest Millisecond Pulsar Revisited with XMM-Newton:
# Improved Mass-radius Constraints for PSR J0437-4715"
# The Astrophysical Journal, Volume 762, Issue 2, article id. 96, 9 pp. (2013).
# Spectrum values (including hot spot size) based on Table 2, line 6 on
# the aforementioned work.
nstiltangle = 42.
nsradius = 13.5
nsmass = 1.76
nsdist = 156.3
nsspinpd = 5.76
hs1inc = 36.
hs1size = 0.16 #tab1ln12 r=.17 tab2ln6 r=.16
hs2incOffset = -25.
hs2inc = 180.-hs1inc+hs2incOffset
hs2size = 5.5 #tab1ln12 r=1.1 tab2ln6 r=5.5
hs2dPhi = -20
psr = NS.Pulsar(nstiltangle, nsradius, nsmass, nsdist, nsspinpd)

def SingleHotSpotFlux(hsinc, spotarea, emergentspecfile,
                      phi_low, phi_high, name='HS1'):
    # Unique Hot Spot Values
    theta = hsinc
    area = spotarea
    datafile = emergentspecfile
```



```

# Phase values
phase_low, phase_high = phi_low, phi_high
phase = (phase_high + phase_low) / 2.
# make hot spot object, rename variables.
hs = NS.HotSpot(theta, area, psr, datafile)
Rs, R = hs.psr.schradius, hs.psr.radius
NSTilt, HStilt = hs.psr.NStilt, hs.HStilt
# create two arrays of low and high frequencies and energies
freq_obs = Flux.ConvertEsurf2Eobs(hs.listfreq, Rs, R)
energy_obs = Flux.ConvertEsurf2Eobs(hs.listenergy, Rs, R)
j = np.array(range(len(freq_obs)-2))
freq_low = (freq_obs[j+1]+freq_obs[j])/2
freq_high = (freq_obs[j+2]+freq_obs[j+1])/2
energy_low = (energy_obs[j+1]+energy_obs[j])/2
energy_high = (energy_obs[j+2]+energy_obs[j+1])/2
# Find Mu value for this phase (use average phase value)
Mu = Flux.ConvertPhase2Mu(phase, Rs, R, NSTilt, HStilt, doppler=1)
# Find emergent intensity for each frequency/energy at this Mu(phase)
EI = hs.emisHz(Mu, freq_obs)[0]
# Find Flux. Following lines provide flux in ergs and photons
fluxergs = Flux.flux(Rs, R, Mu, area, EI, psr.dist) # ergs/s/cm^2/Hz
fluxergs_tab = fluxergs[j+1]
fluxPho = Flux.FluxInPhotons(fluxergs, energy_obs) # Photons/s/cm^2/keV
fluxPho_tab = fluxPho[j+1]
energy_tab = energy_obs[j+1]
fluxPho_tab2 = fluxPho_tab*energy_tab # Photons/s/cm^2
# Check visibility
visible = Flux.visibility(phase, Rs, R, NSTilt, HStilt, doppler=1)
if not visible:
    len1, len2 = len(fluxergs), len(fluxergs_tab)
    fluxergs, fluxPho = np.zeros(len1), np.zeros(len1)
    fluxergs_tab, fluxPho_tab = np.zeros(len2), np.zeros(len2)
    fluxPho_tab2 = np.zeros(len2)
# Merge energy_low and energy_high arrays for plotting
# Do same for the flux
x_energy = list(energy_low)
y_fluxPho = list(fluxPho_tab)
for i in range(1, len(x_energy)+1):
    x_energy.insert(i*2-1, energy_high[i-1])
    y_fluxPho.insert(i*2-1, fluxPho_tab[i-1])
# next few lines for plotting purposes
cosphiTermOne = np.sin(HStilt)*np.sin(NStilt)*np.cos(phase)
cosphiTermTwo = np.cos(HStilt)*np.cos(NStilt)
cosphi = cosphiTermOne + cosphiTermTwo
viscond = -(Rs/R)/(1-(Rs/R))
# Sum the flux. provide a couple different values for flux output
ind1 = range(157, 260) # .1-.5 keV
totf11 = np.sum((energy_high[ind1]-energy_low[ind1])*fluxPho_tab[ind1])
ind2 = range(260, 305) # .5-1 keV
totf12 = np.sum((energy_high[ind2]-energy_low[ind2])*fluxPho_tab[ind2])
ind3 = range(305, 349) # 1-2 keV
totf13 = np.sum((energy_high[ind3]-energy_low[ind3])*fluxPho_tab[ind3])
pltdat = [x_energy, y_fluxPho, Mu, totf11, totf12, totf13, cosphi, viscond]
ergdat = [freq_low, freq_high, fluxergs_tab]
return energy_low, energy_high, fluxPho_tab2, phase, pltdat, ergdat, fluxPho_tab

fname1 = 'PriHotSpotEmergSpec/EmergentSpectrum.200.6.dat'
fname2 = 'SecHotSpotEmergSpec/EmergentSpectrum.200.6.dat'
bn = math.pi/300. # phase intervals
phaserange = np.arange(-bn, 2*math.pi+bn, bn)
p1x, p1y, p1m, p1f, p1f2, p1f3, p1c = [], [], [], [], [], [], []
p2x, p2y, p2m, p2f, p2f2, p2f3, p2c = [], [], [], [], [], [], []
plow = []
flux1, flux2, ferg1, ferg2, falt1, falt2 = [], [], [], [], [], []
para = []
for p in phaserange:
    # Hot Spot 1: Primary Hot Spot
    v1, v2, v3, v4, v5, v6 = hslinc, hslsize, fname1, p, p+bn, 'HS1-' + str(p) #input

```

```

elo1,ehi1,fPh1,phas,pd1,fe1,fa1 = SingleHotSpotFlux(v1,v2,v3,v4,v5,name=v6)
flux1.append(list(fPh1)) # fluxPho_tab2 units Photons/s/cm^2
ferg1.append(list(fe1[2])) # fluxergs_tab units ergs/s/cm^2/Hz
falt1.append(list(fa1)) # fluxPho_tab units Photons/s/cm^2/keV
para.append((phas/(math.pi))*180.) #(phase value for parameter file)
# Hot Spot 2: Secondary Hot spot
philow,phihi = np.radians(180+hs2dPhi)+p, np.radians(180+hs2dPhi)+p+bn
v1,v2,v3,v4,v5,v6 = hs2inc,hs2size,fname2,philow,phihi,'HS2-'+str(p)#input
elo2,ehi2,fPh2,dum,pd2,fe2,fa2 = SingleHotSpotFlux(v1,v2,v3,v4,v5,name=v6)
flux2.append(list(fPh2)) # fluxPho_tab2 Photons/s/cm^2
ferg2.append(list(fe2[2])) # fluxergs_tab units ergs/s/cm^2/Hz
falt2.append(list(fa2)) # fluxPho_tab units Photons/s/cm^2/keV
# Following line provided to make light curve plots with
# plot data (pd1 and pd2)
p1x.append(pd1[0])
p1y.append(pd1[1])
p1m.append(pd1[2])
p1f.append(pd1[3])
p1f2.append(pd1[4])
p1f3.append(pd1[5])
p1c.append(pd1[6])
p2x.append(pd2[0])
p2y.append(pd2[1])
p2m.append(pd2[2])
p2f.append(pd2[3])
p2f2.append(pd2[4])
p2f3.append(pd2[5])
p2c.append(pd2[6])
vis = pd1[7]
# energy and frequency for each hotspot
hs1en = [elo1, ehi1]
hs2en = [elo2, ehi2]
hs1hz = [fe1[0], fe1[1]]
hs2hz = [fe2[0], fe2[1]]
hs1ena = [elo1, ehi1]
hs2ena = [elo2, ehi2]
# put in flx2tab table data format
for i in range(len(flux1)):
    hs1en.append(flux1[i])
    hs2en.append(flux2[i])
    hs1hz.append(ferg1[i])
    hs2hz.append(ferg2[i])
    hs1ena.append(falt1[i])
    hs2ena.append(falt2[i])
np.savetxt('tabledata/HS1.txt',np.array(hs1en).T)
np.savetxt('tabledata/HS2.txt',np.array(hs2en).T)
np.savetxt('tabledata/HS1hz.txt',np.array(hs1hz).T)
np.savetxt('tabledata/HS2hz.txt',np.array(hs2hz).T)
np.savetxt('tabledata/HS1a.txt',np.array(hs1ena).T)
np.savetxt('tabledata/HS2a.txt',np.array(hs2ena).T)

# Create parameter file
# line 1: phasedeg
# line 2: 0
# line 3: 0 1 0 0 359 359
# line 4: para
l1 = 'phasedeg\n'
l2 = '0\n'
l3 = '0.0 1.0 0.0 0.0 359.0 359.0\n'
l4 = ''
for i in range(len(para)):
    l4 = l4+str(para[i])+ ' '
l4 = l4+'\n'
f = open('tabledata/parafile.txt','w')
f.write(l1)
f.write(l2)
f.write(l3)
f.write(l4)

```

```
f.close()
```

E.2.2 NeutronStar.py

Source File: NeutronStar.py

```
# -*- coding: utf-8 -*-
# NeutronStar.py

'''
This module contains two classes which define features of the
neutron star. One class creates a pulsar object and another class
creates a hot spot object on a particular pulsar.
'''

from matplotlib.pyplot import *
import numpy as np
import math
from scipy import interpolate

class Pulsar(object):
    '''
    Defines the geometric features of a pulsar. A Pulsar() object
    is required to create one or more HotSpot() objects as emission
    from the pulsar's magnetic poles.

    See __init__ method for input and output variables.
    '''

    #####
    ##      Pulsar Object Initialization      ##
    #####
    ##Input:
    ##  NStilt      - Neutron star tilt angle. Angle in degrees
    ##                between the observer's line of sight and the
    ##                pulsar's spin axis
    ##  radius      - Neutron star radius in kilometers
    ##  mass         - Neutron star mass in solar mass units
    ##  distance     - Star's distance from observer in parces
    ##  spinperiod   - Spin period of pulsar in milliseconds
    ##Output:
    ##  Pulsar.NStilt - NStilt input converted to radians as float
    ##  Pulsar.radius - radius input as float in km
    ##  Pulsar.mass   - mass input as float in solar mass units
    ##  Pulsar.schradius - Schwarzschild radius = Pulsar.mass*2.95
    ##  Pulsar.dist    - distance input in km
    ##  Pulsar.period  - spinperiod input as float
    def __init__(self, NStilt, radius, mass,
                  distance, spinperiod):
        self.NStilt = math.radians(float(NStilt))
        self.radius = float(radius)
        self.mass = float(mass)
        self.schradius = self.mass*2.95 # km
        self.dist = distance*(3.08567758*10**16)
        self.period = float(spinperiod)

class HotSpot(object):
```

```

'''
Defines the emission from a single magnetic pole on a Pulsar() object.
The object requires an emission file with columns of frequency (nu),
cosine of the emission angle (mu), and intensity (I). McPHAC emergent
spectrum output can be used (i.e. a file with name such as
"EmergentSpectrum.200.6.dat" is appropriate).
'''

## Planck's Constant in keV*s
h = 4.136*(10**-18)

#####
## HotSpot Object Initialization ##
#####
##Input:
## HStilt - Emission spot inclination. Angle in degrees
## between spin axis and normal to surface at
## emission spot
## spotarea - Area of emission spot in km^2
## pulsarClass - Pulsar() class object
## EmissionFile - Path to file of emergent spectrum
##Output:
## self.HStilt - HStilt input in radians as float
## self.spotsize - spotarea input as float
## self.psr - pulsarClass input
## self.NU - Array of 'nu' from EmissionFile
## self.MU - Array of 'mu' from EmissionFile
## self.INTEN - Array of 'I' from EmissionFile
## self.nfreq - Number of unique frequencies
## self.nmu - Number of unique emission angles
## self.E - List of energies created by multiplying self.NU
## by planck's constant. Units of keV
## self.freqarr - Reshaped frequency array to contain nfreq array
## elements, each of which has nmu elements
## (np.shape = nfreq,nmu)
## self.muarr - Reshaped mu array to contain nfreq array
## elements, each of which has nmu elements
## (np.shape = nfreq,nmu)
## self.intenarr - Reshaped intensity array to contain nfreq array
## elements, each of which has nmu elements
## (np.shape = nfreq,nmu)
## self.listfreq - list of frequencies based off of the first
## element of each array in self.freqarray
## self.listenergy - list of energies in keV caculated by multiplying
## all the elements in self.freqarray by planck's
## constant
## self.listmu - list of mu values which is first array in
## self.muarr
## self.emis - Interpolation function of intensity based on
## a grid of mu and energies.
## Uses scipy.interpolate.RectBivariateSpline
## self.emisHz - Interpolation function of intensity based on
## a grid of mu and frequencies.
## Uses scipy.interpolate.RectBivariateSpline
def __init__(self, HStilt, spotarea, pulsarClass, EmissionFile):
    self.HStilt = math.radians(float(HStilt))
    self.spotsize = float(spotarea)
    self.psr = pulsarClass

    self.NU, self.MU, self.INTEN = np.loadtxt(EmissionFile,unpack=True)
    self.nfreq = len(np.unique(self.NU))
    self.nmu = len(np.unique(self.MU))
    self.E = self.h*self.NU

    ##each element in array has nmu elements
    self.freqarr = np.reshape(self.NU,(self.nfreq,self.nmu))
    self.muarr = np.reshape(self.MU,(self.nfreq,self.nmu))

```

```

self.intenarr = np.reshape(self.INTEN,(self.nfreq,self.nmu))

# list of frequencies
self.listfreq = self.freqarr.T[0]
self.listenergy = self.listfreq*self.h
# list of mus
self.listmu = self.muarr[0]

# Interpolated emission function (input = mu and energy)
self.emis = interpolate.RectBivariateSpline(self.listmu, self.listenergy, self.intenarr.T)
# Interpolated emission function (input = mu and frequency)
self.emisHz = interpolate.RectBivariateSpline(self.listmu, self.listfreq, self.intenarr.T)

```

E.2.3 Flux.py

Source File: Flux.py

```

import numpy as np
import math
from scipy import integrate

'''
This module contains functions needed to calculate the pulsar
flux. The user should create a NeutronStar.Pulsar() and
one or more NeutronStar.HotSpot() objects.
'''

def ConvertPhase2Mu(Phase, Rs, R, NSTilt, HStilt, doppler=1):
    '''
    Converts the phase as observed by observer to the emergent
    emission angle on the surface in the non-rotating frame.
    cosphi is the cosine of the angle between the observer's line
    of sight and normal to surface at emission spot.
    When doppler=1, the doppler shift from the pulsar's rotation
    is not considered.
    '''
    cosphiTermOne = np.sin(HStilt)*np.sin(NStilt)*np.cos(Phase)
    cosphiTermTwo = np.cos(HStilt)*np.cos(NStilt)
    cosphi = cosphiTermOne + cosphiTermTwo
    mu = doppler*(cosphi*(1-(Rs/R))+(Rs/R))
    return mu

def ConvertEsurf2Eobs(Esurf, Rs, R):
    return Esurf*np.sqrt(1-(Rs/R))

def flux(Rs, R, spotarea, Mu, emerginten, psrdist):
    return ((1-Rs/R)**(3./2.))*emerginten*(spotarea*Mu)/(psrdist**2)

# # Flux conversion ergs/cm^2/s/Hz to photons/cm^2/s/keV
# x [photons/cm^2/s/keV] = 1.5092x10^26 F [ergs/cm^2/s/Hz] / E [keV]
def FluxInPhotons(FluxInErgs,Energy):
    return FluxInErgs * (1.5092*10**26) / Energy

def visibility(Phase, Rs, R, NSTilt, HStilt, doppler=1):
    cosphiTermOne = np.sin(HStilt)*np.sin(NStilt)*np.cos(Phase)
    cosphiTermTwo = np.cos(HStilt)*np.cos(NStilt)
    cosphi = cosphiTermOne + cosphiTermTwo
    vis = -(Rs/R)/(1-(Rs/R))
    if cosphi > vis: return True

```

```
else: return False
```

E.3 McPHAC Input Parameters

Source File: McPHAC.bash

```
#!/bin/bash

# Physical Parameters
LOGTEFF=6.34439227368511 #Log Target effective temperature, K
GSURFACE=1.85e14 #Surface gravitational acceleration, cm s-2
# PSR J0437-4715
# HS1 logteff = 6.34439227368511 (2.21x106K)
# HS2 logteff = 5.755874855672497 (0.57x106K)
# gsurface = 1.85e14

# Computational parameters
MINCOL=-9.0 # Min. log(y) considered for Teff<106.5K
MAXCOL=2.0 # Initial max. log(y), increased if necessary to meet MAXTAUTAU criterion
# (set MAXCOLTAU=0.01 to keep unchanged)

MAXCOLTAU=80.0 # Consider log(y) large enough to have at least
# MAXCOLTAU optical depths at the largest freq.
#MAXCOLTAU=0.01 # Set this value of MAXCOLTAU to keep the
# largest y considered unchanged

TGUESSBDYCOND=0.264837817 # Ratio of T to Teff at the surface to
# use for initial temperature profile

NDEPTHS=200 # Initial number of depths points in the temperature correction
MAXFACTOR=1 # Maximum factor to multiply NDEPTHS by
# (NDEPTHS doubled until and including this factor)
NDEPTHSNU=200 # Initial number of depths points in the radiative transfer
MAXFACTORNU=1 # Maximum factor to multiply NDEPTHSNU by
# (NDEPTHSNU doubled until and including this factor)
NMU=200 # Number of mu points over range [0,1]
NFREQ=500 # Number of photon frequency bins

MAXFRACTEMPCHANGE=0.0001 # Continue iteration until max.
# fractional temp. change < MAXFRACTEMPCHANGE
MAXITER=20 # Maximum number of iterations allowed

ANIST=0 # Whether or not to treat Thomson scattering anisotropically
# (should only be set if FEAUTRIER is)

# Run McPHAC
cmdnd="./McPHAC $LOGTEFF $GSURFACE $MINCOL $MAXCOL $NDEPTHS $MAXFACTOR
$NDEPTHSNU $MAXFACTOR $NMU $NFREQ $MAXFRACTEMPCHANGE $MAXITER $ANIST
$MAXCOLTAU $TGUESSBDYCOND"
echo $cmdnd
$cmdnd
```

E.4 Virtual NICER Codes

Source File: raytraceEA.bash

```
#!/bin/bash

# Do effective area
doea=2
# scattering parameters
gauss_norm=3.12
gauss_sigma=1.55
lorentz_norm=0.001
lorentz_sigma=10.0
exp_norm=0.1
exp_sigma=0.4

#####
### SET 3.1
### 5,000 photons
### angles 7.1-9.0
### energies 0.2-1.5 keV (100 eV bins)
#####
# energy range
minE=0.2
maxE=1.5
# energy bin size (eV)
energybin=100
setnum=3.1
for ang in [ 7.10 7.20 7.30 7.40 7.50 7.60 7.70 7.80 7.90 8.00
8.10 8.20 8.30 8.40 8.50 8.60 8.70 8.80 8.90 9.00 ]; do
filename="/home/erin/Desktop/VirtualNICER/consimEAfiles/EA$ang-$setnum.fits"
cd /$HOME/CONSIM
./consim << EOF
$filename
/home/erin/CONSIM/mirrorFITS/NICER_PARABOLIC_XRT.fits
p
$doea
$minE
$maxE
$energybin
yes
$gauss_norm
$gauss_sigma
$lorentz_norm
$lorentz_sigma
$exp_norm
$exp_sigma
no
$ang
1.0
no
EOF
done

#####
#####
#####

### NOTE ###
# For computation time, I ran several sets of different angles
# where I changed the input parameters. Here are the
```

```

# descriptions of each set.  The effective area files need
# to be pieced together based on this information.
#####
### SET 1.1
### 20,000 photons
### angles 0.0-5.0
### energies 0.2-1.5 keV (100 eV bins)
#####
# energy range
#minE=0.2
#maxE=1.5
# energy bin size (eV)
#energybin=100
#setnum=1.1
#[ 0.00 0.10 0.20 0.30 0.40 0.50 0.60 0.70 0.80 0.90 1.00 1.10 1.20 1.30 1.40
#1.50 1.60 1.70 1.80 1.90 2.00 2.10 2.20 2.30 2.40 2.50 2.60 2.70 2.80 2.90
#3.00 3.10 3.20 3.30 3.40 3.50 3.60 3.70 3.80 3.90 4.00 4.10 4.20 4.30 4.40
#4.50 4.60 4.70 4.80 4.90 5.00 ]
#####
### SET 1.2
### 20,000 photons
### angles 0.0-7.0
### energies 1.525-3.0 keV (25 eV bins)
#####
# energy range
#minE=1.525
#maxE=3.0
# energy bin size (eV)
#energybin=25
#setnum=1.2
#[ 0.00 0.10 0.20 0.30 0.40 0.50 0.60 0.70 0.80 0.90 1.00 1.10 1.20 1.30 1.40
#1.50 1.60 1.70 1.80 1.90 2.00 2.10 2.20 2.30 2.40 2.50 2.60 2.70 2.80 2.90
#3.00 3.10 3.20 3.30 3.40 3.50 3.60 3.70 3.80 3.90 4.00 4.10 4.20 4.30 4.40
#4.50 4.60 4.70 4.80 4.90 5.00 5.10 5.20 5.30 5.40 5.50 5.60 5.70 5.80 5.90
#6.00 6.10 6.20 6.30 6.40 6.50 6.60 6.70 6.80 6.90 7.00 ]

#####
### SET 2.1
### 10,000 photons
### angles 5.1-7.0
### energies 0.2-1.5 keV (100 eV bins)
#####
# energy range
#minE=0.2
#maxE=1.5
# energy bin size (eV)
#energybin=100
#setnum=2.1
#[ 5.10 5.20 5.30 5.40 5.50 5.60 5.70 5.80 5.90 6.00 6.10 6.20 6.30 6.40 6.50
#6.60 6.70 6.80 6.90 7.00 ]
#####
### SET 2.2
### 10,000 photons
### angles 7.1-9.0
### energies 1.525-3.0 keV (25 eV bins)
#####
# energy range
#minE=1.525
#maxE=3.0
# energy bin size (eV)
#energybin=25
#setnum=2.2
#[ 7.10 7.20 7.30 7.40 7.50 7.60 7.70 7.80 7.90 8.00 8.10 8.20 8.30 8.40 8.50
#8.60 8.70 8.80 8.90 9.00 ]
#####
### SET 2.3
### 10,000 photons
### angles 0.0-7.0

```



```

### energies 3.1-5.0 keV (100 eV bins)
#####
# energy range
#minE=3.1
#maxE=5.0
# energy bin size (eV)
#energybin=100
#setnum=2.3
#[ 0.00 0.10 0.20 0.30 0.40 0.50 0.60 0.70 0.80 0.90 1.00 1.10 1.20 1.30 1.40
#1.50 1.60 1.70 1.80 1.90 2.00 2.10 2.20 2.30 2.40 2.50 2.60 2.70 2.80 2.90
#3.00 3.10 3.20 3.30 3.40 3.50 3.60 3.70 3.80 3.90 4.00 4.10 4.20 4.30 4.40
#4.50 4.60 4.70 4.80 4.90 5.00 5.10 5.20 5.30 5.40 5.50 5.60 5.70 5.80 5.90
#6.00 6.10 6.20 6.30 6.40 6.50 6.60 6.70 6.80 6.90 7.00 ]

#####
### SET 3.1
### 5,000 photons
### angles 7.1-9.0
### energies 0.2-1.5 keV (100 eV bins)
#####
# energy range
#minE=0.2
#maxE=1.5
# energy bin size (eV)
#energybin=100
#setnum=3.1
#[ 7.10 7.20 7.30 7.40 7.50 7.60 7.70 7.80 7.90 8.00 8.10 8.20 8.30 8.40 8.50
#8.60 8.70 8.80 8.90 9.00 ]
#####
### SET 3.2
### 5,000 photons
### angles 7.1-9.0
### energies 3.1-5.0 keV (100 eV bins)
#####
# energy range
#minE=3.1
#maxE=5.0
# energy bin size (eV)
#energybin=100
#setnum=3.2
#[ 7.10 7.20 7.30 7.40 7.50 7.60 7.70 7.80 7.90 8.00 8.10 8.20 8.30 8.40 8.50
#8.60 8.70 8.80 8.90 9.00 ]
#####
### SET 3.3
### 5,000 photons
### angles 0.0-9.0
### energies 5.2-10.2 keV (200 eV bins)
#####
# energy range
#minE=5.2
#maxE=10.2
# energy bin size (eV)
#energybin=200
#setnum=3.3
#[ 0.00 0.10 0.20 0.30 0.40 0.50 0.60 0.70 0.80 0.90 1.00 1.10 1.20 1.30 1.40 1.50
#1.60 1.70 1.80 1.90 2.00 2.10 2.20 2.30 2.40 2.50 2.60 2.70 2.80 2.90 3.00 3.10
#3.20 3.30 3.40 3.50 3.60 3.70 3.80 3.90 4.00 4.10 4.20 4.30 4.40 4.50 4.60 4.70
#4.80 4.90 5.00 5.10 5.20 5.30 5.40 5.50 5.60 5.70 5.80 5.90 6.00 6.10 6.20 6.30
#6.40 6.50 6.60 6.70 6.80 6.90 7.00 7.10 7.20 7.30 7.40 7.50 7.60 7.70 7.80 7.90
#8.00 8.10 8.20 8.30 8.40 8.50 8.60 8.70 8.80 8.90 9.00 ]

```

Source File: GenerateEAEfficiencyFiles.bash

```

#!/bin/bash
# Generate efffil (effective area) file for each off-axis
# raytraced file. Output is in mm2

```

```

# 1) need to use fdump to get EA files for each fits file
# 2) then combine the EA files for the same angle
# 3) use the complete EA files to create the efffil files

# angles 0.0-5.0, sets 1.1, 1.2, 2.3, 3.3
for angle in 0.00 0.10 0.20 0.30 0.40 0.50 0.60 0.70 0.80 0.90 1.00 1.10 1.20 1.30
    1.40 1.50 1.60 1.70 1.80 1.90 2.00 2.10 2.20 2.30 2.40 2.50 2.60 2.70
    2.80 2.90 3.00 3.10 3.20 3.30 3.40 3.50 3.60 3.70 3.80 3.90 4.00 4.10
    4.20 4.30 4.40 4.50 4.60 4.70 4.80 4.90 5.00; do
for set in 1.1 1.2 2.3 3.3; do
consimfile="consimEAfiles/EA$angle-$set.fits"
outeff="EfficiencyFiles/fdumpOut/EA$angle-$set.txt"
fdump $consimfile[1] $outeff columns="energy,area0" rows=- prhead=no showcol=no
    showrow=no showunit=no clobber=yes
done
python attachEAarr.py $angle 1.1 1.2 2.3 3.3
done

# angles 5.1-7.0, sets 2.1, 1.2, 2.3, 3.3
for angle in 5.10 5.20 5.30 5.40 5.50 5.60 5.70 5.80 5.90 6.00 6.10 6.20 6.30 6.40
    6.50 6.60 6.70 6.80 6.90 7.00; do
for set in 2.1 1.2 2.3 3.3; do
consimfile="consimEAfiles/EA$angle-$set.fits"
outeff="EfficiencyFiles/fdumpOut/EA$angle-$set.txt"
fdump $consimfile[1] $outeff columns="energy,area0" rows=- prhead=no showcol=no
    showrow=no showunit=no clobber=yes
done
python attachEAarr.py $angle 2.1 1.2 2.3 3.3
done

# angles 7.1-9.0, sets 3.1, 2.2, 3.2, 3.3
for angle in 7.10 7.20 7.30 7.40 7.50 7.60 7.70 7.80 7.90 8.00 8.10 8.20 8.30 8.40
    8.50 8.60 8.70 8.80 8.90 9.00; do
for set in 3.1 2.2 3.2 3.3; do
consimfile="consimEAfiles/EA$angle-$set.fits"
outeff="EfficiencyFiles/fdumpOut/EA$angle-$set.txt"
fdump $consimfile[1] $outeff columns="energy,area0" rows=- prhead=no showcol=no
    showrow=no showunit=no clobber=yes
done
python attachEAarr.py $angle 3.1 2.2 3.2 3.3
done

```

Source File: attachEAarr.py

```

import numpy as np
import sys
angle=sys.argv[1]
sets=sys.argv[2:]
totx, toty = np.array([]), np.array([])
for s in sets:
    myfile="EfficiencyFiles/fdumpOut/EA"+angle+"-"+s+".txt"
    x, y = np.loadtxt(myfile, skiprows=2, unpack=True)
    totx = np.append(totx, x)
    toty = np.append(toty, y)
np.savetxt("EfficiencyFiles/efffil/EA"+angle+".txt",
    (totx,toty),header="energy(keV) EA(mm2)")

```

Source File: CreateUniqueNICER.bash

```

#!/bin/bash
# STEP - Create a NICER
#         unique set of pointing vectors
## Pointing budget:
##   Pointing (boresight): 85" at 3 sig
##   Alignment (XRC): 35" at 3 sig

```

```

# Ask user for an ID
echo "NICER version ID: "
read ID
if [ -d "NICERVersions/ID$ID" ]; then
# If ID was already use, tell user and exit program
echo "ID number $ID already used. Try again."
else
# Otherwise...
# Set NICER version ID and create directory
mkdir "NICERVersions/ID$ID"
# Create Vector Arrays
python makevectors.py "NICERVersions/ID$ID/NICER$ID-PtVect.txt"
fi

```

Source File: makevectors.py

```

# makevectors.py
## Creates array files of pointing Errors and vectors
from scipy.stats import norm
import numpy as np
import sys
outfile = sys.argv[1]
# BORESIGHT ERROR budget: 85" at 3 sigma
sig, mu = (85./60.)/3., 0.0 # convert to 1-sig in arcminute
bore = list(np.zeros(56) + np.random.normal(mu, sig))
# BORESIGHT POINTING DIRECTION (degree)
boreDir = np.zeros(56) + np.random.rand(1)*360.
# XRC ERRORS budget: 35" at 3 sigma
sig, mu = (35./60.)/3., 0.0 # convert to 1-sig in arcminute
xrc = []
for i in range(56): xrc.append(np.random.normal(mu, sig))
# XRC POINTING DIRECTIONS (degree)
xrcDir = np.random.rand(56)*360.
# SUM X AND Y VECTORS OF BORESIGHT AND XRCs
x = bore*np.cos(np.radians(boreDir)) + xrc*np.cos(np.radians(xrcDir))
y = bore*np.sin(np.radians(boreDir)) + xrc*np.sin(np.radians(xrcDir))
# RESULTANT POLAR COORDINATE VECTORS
r = np.sqrt(x**2+y**2)
theta = np.tan(y/x)
# save to file
np.savetxt(outfile, (x,y,r,theta,bore,boreDir,xrc,xrcDir),
            header='Realization of pointing errors\n'+
            'Xvec Yvec Rvec THETAvec boreErr boreDir xrcErr xrcDir')
## Xvec Yvec Rvec THETAvec boreErr boreDir xrcErr xrcDir
## Xvec, Yvec: x and y final pointing vectors for each XRC, the
##             sum of boresight and alignment error
## Rvec, THETAvec: polar coordinated of final pointing direction
## boreErr, xrcErr: boresight and XRCs radial error
## boreDir, xrcDir: boresight angular direction of error

```

Source File: PointNICER.bash

```

#!/bin/bash
# STEP Point NICER to Make an observation
# 1) Find angle between each source in FOV and each XRC
# 2) Add 56 EA files together, for each source, corresponding
#    to the pointing vectors
# 3) Create a RSP for each source using the total effil and
#    the detfil and filfil.

# Ask user which NICER to use
echo "Which Unique NICER ID are you using? "
read ID
# Ask user for pointing angle
# set Pulsar at origin and AGN located at x=-4.2
# --> pointing at pulsar is:

```

```

#         PtAngX = 0.0
#         PtAngY = 0.0
#         --> pointing at AGN is:
#         PtAngX = -4.2
#         PtAngY = 0.0
echo "Where are your pointing? (PSR: x=0,y=0 AGN:x=-4.2,y=0)"
echo "  enter X direction (PtAngX): "
read PtAngX
echo "  enter Y direction (PtAngY): "
read PtAngY

# Make directory for pointing angle.
# Remove it if it already exists.
obsdir="NICERVersions/ID$ID/Observation_x=$PtAngX-y=$PtAngY"
### comment out below if you do not want to remove obsdir
if [ -d $obsdir ]; then
    rm -rf $obsdir
fi
mkdir $obsdir

# Make directory for pointing angle.
# Offer to remove if it already exists.
# These lines only uncommented if this script is not
# part of a loop in an external program/script.
##if [ -d $obsdir ]; then
##    echo "Directory $obsdir already exists."
##    echo "Would you like to remove it? (yes/no)"
##    read ans
##    if [ $ans != 'yes' ]; then
##        echo 'EXITING PROGRAM'
##        exit 0
##    else
##        echo "Are you sure you want to remove it?"
##        read answ
##        if [ $answ == 'yes' ]; then
##            rm -rf $obsdir
##            echo "REMOVED $obsdir"
##        else
##            echo 'EXITING PROGRAM'
##            exit 0
##        fi
##    fi
##fi
##mkdir $obsdir
##echo "Made Directory $obsdir"

# Load values into python and find distance between each
vecfile="NICERVersions/ID$ID/NICER$ID-PtVect.txt"
python sourcedist.py $vecfile $PtAngX $PtAngY $ID $obsdir
echo 'Distance to sources found'

# Sum the effective area arrays from the raytraced files
# to be used in creating the unique RSP files for this
# observation with the current version of NICER.
DistanceFile="$obsdir/SourceDistances.txt"
python addeffils.py $DistanceFile $obsdir

# make RSP files. One for the pulsar and one for the AGN
energymin=0.2
energymax=10.2
detectorEfficiency="EfficiencyFiles/detfil_NICER.txt"
filterEfficiency="EfficiencyFiles/filfil_NICER.txt"
EAprs="$obsdir/efffilPSR.txt"
EAagn="$obsdir/efffilAGN.txt"
RMFpsr="$obsdir/rmfpsr.rsp"
RMFagn="$obsdir/rmfagn.rsp"
# PULSAR
genrsp inrfil=none rmffil=$RMFpsr resol_reln=constant fwhm=0.01

```

```

disperse=no tlescpe=NICER instrm=XRC resp_reln=linear resp_low=$energymin
resp_high=$energymax resp_number=10000 chan_reln=linear
chan_low=$energymin chan_high=$energymax chan_number=10000
efffil=$EApsr detfil=$detectorEfficiency filfil=$filterEfficiency
max.elements=1000000 clobber=yes
# AGN
genrsp inrfil=None rmfil=$RMFagn resol_reln=constant fwhm=0.01
disperse=no tlescpe=NICER instrm=XRC resp_reln=linear resp_low=$energymin
resp_high=$energymax resp_number=10000 chan_reln=linear
chan_low=$energymin chan_high=$energymax chan_number=10000
efffil=$EAagn detfil=$detectorEfficiency filfil=$filterEfficiency
max.elements=1000000 clobber=yes

```

Source File: sourcedist.py

```

import numpy as np
import sys
import matplotlib.pyplot as plt
# Get vector file and pointing direction
#vectorfile, ptangX, ptangY = "NICERVersions/ID01/NICER01-PtVect.txt", "0", "0"
#ID, obsdir = "01", "NICERVersions/ID01/Observation_x=0-y=0"
vectorfile, ptangX, ptangY = sys.argv[1], sys.argv[2], sys.argv[3]
ID, obsdir = sys.argv[4], sys.argv[5]
# pulsar location is at origin
psrX, psrY = 0.0, 0.0
# AGN location is at x=-4.2
agnX, agnY = -4.2, 0.0
xrcX, xrcY = np.loadtxt(vectorfile)[0:2]
# location of each xrc pointing direction
nicerX = xrcX + float(ptangX)
nicerY = xrcY + float(ptangY)
# find distance to each source for each xrc
def dist(x1,x2,y1,y2):
    return np.sqrt((x1-x2)**2+(y1-y2)**2)
psrdist = dist(nicerX,psrX,nicerY,psrY)
agndist = dist(nicerX,agnX,nicerY,agnY)

# save distance angles as txt file in observation folder
np.savetxt(obsdir+'SourceDistances.txt',(psrdist,agndist),
           header='Distance to sources (arcmin)\nPulsar    AGN')

# Make and save figure of observation pointing diagram
plt.figure(figsize=(12,3))
plt.plot(psrX, psrY, 'g*', ms=10, label = 'PSR')
plt.plot(agnX, agnY, 'b*', ms=20, label = 'AGN')
plt.plot(nicerX, nicerY, 'k.', ms=2, label = 'XRC Pointing Directions')
plt.plot(ptangX,ptangY, 'rx', label = 'Observation Direction', ms=15)
plt.xlim(-6,6)
plt.ylim(-1.5,1.5)
plt.legend(prop={'size':'small'})
plt.title('NICER ID '+ID+' Pointing x='+ptangX+' y='+ptangY)
plt.savefig(obsdir+'pointingdiagram.eps')

```

Source File: addeffils.py

```

import numpy as np
import sys
import matplotlib.pyplot as plt
# Get input and load data
distfile = sys.argv[1]
obsdir = sys.argv[2]
psrdist, agndist = np.loadtxt(distfile)
psr = np.round(psrdist,1)
agn = np.round(agndist,1)

```

```

# make lists of string values of the angles
psrStr, agnStr = [], []
for i in range(56):
    psrStr.append(str(psr[i])+'.0')
    agnStr.append(str(agn[i])+'.0')

# Load each efffil file corresponding to the angles in psr and agn.
# When loading, divide by 100 because values are in mm2 and we want cm2
energy = np.loadtxt('EfficiencyFiles/efffil/EA0.00.txt')[0]
psreatot = np.zeros(len(energy))
agneatot = np.zeros(len(energy))
for n in psrStr:
    ea = np.loadtxt('EfficiencyFiles/efffil/EA'+n+'.txt')[1]/100.
    psreatot = psreatot+ea
##    plt.plot(energy,ea,'k')
for n in agnStr:
    ea = np.loadtxt('EfficiencyFiles/efffil/EA'+n+'.txt')[1]/100.
    agneatot = agneatot+ea
##    plt.plot(energy,ea,'k')
plt.plot(energy,psreatot,label='PSR EA')
plt.plot(energy,agneatot,label='AGN EA')
plt.title("Effective Areas Used for Each Source's Off Axis Observation")
plt.xlabel('Energy (keV)')
plt.ylabel('Effective Area of all 56 XRCs (cm2)')
plt.legend()
plt.savefig(obsdir+'/EAplot.eps')
np.savetxt(obsdir+'/efffilPSR.txt',np.transpose([energy,psreatot]))
np.savetxt(obsdir+'/efffilAGN.txt',np.transpose([energy,agneatot]))

```

E.5 Simulation Analysis Codes

Source File: AnalyzingIdealCase.py

```

import numpy as np
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
import matplotlib.cm as cm
from uncertainties import ufloat
from uncertainties.umath import *
from uncertainties import unumpy
import matplotlib
matplotlib.rc('font',**{'size':12})
matplotlib.rc('axes',**{'labelsize':10})
matplotlib.rc('xtick',**{'labelsize':8})
matplotlib.rc('ytick',**{'labelsize':8})
matplotlib.rc('legend',**{'fontsize':8})
matplotlib.rc('legend',frameon=False)
matplotlib.rc('lines',lw=2)
matplotlib.rc('figure',**{'dpi':200})

def loadcountfile(filename):
    dtype={'names':('angle','counts'),'formats':(float,float)}
    data = np.loadtxt(filename,dtype=dtype)
    sorteddata = np.sort(data,order='angle')
    A = sorteddata['angle']
    index = np.array([y for y in [np.where(A==x)[0].flatten('K')
                                for x in PTangle] if len(y)>0]).flatten()

##    print sorteddata[index]
    return sorteddata[index]

def IDEALPSR(phase, Elow, Ehigh, ID='0', ET='1E6'):
    return 'CountRate/ID'+ID+'/COUNT-McPHAC'+phase+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'

```

```

def IDEAL_OBS(phase, Elow, Ehigh, ID='0', ET='1E6'):
    return 'CountRate/ID'+ID+'/COUNT-OBS'+phase+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'
def flatplot(x,y):
    a = np.array(x)
    da = a[1:]-a[:-1]
    b = a[:-1]+da/2.
    acopy=list(a.copy())
    acopy[1:]=a[1:]-da/2.
    [acopy.insert(ind*2+1,b[ind]) for ind in range(len(b))]
    acopy[0]=acopy[0]-(da[0]/2.)
    acopy.append(acopy[-1]+da[-1])
    X = np.array(acopy).copy()
    del a,b,acopy
    a = np.array(y)
    b = y[:-1]
    acopy=list(a.copy())
    [acopy.insert(ind*2+1,b[ind]) for ind in range(len(b))]
    acopy.append(acopy[-1])
    Y = np.array(acopy).copy()
    return X,Y
PTangle=[-1., -0.5, 0., 0.2, 0.5, 0.7, 1.,
          1.5, 2., 2.1, 2.2,
          2.3, 2.4, 2.5, 2.6, 2.7,
          2.8, 2.9, 3., 3.1,
          3.2, 3.3, 3.4, 3.5 ]

#####
## ANALYZING THE IDEAL CASE ##
#####

#####
# 1 #
#####
# Poinging angle vs counts for AGN and pulsar
# All energies (0.2-10.2), sum phases
# Figure name(s) produced: "Source Counts"
ID,ET,Elow,Ehigh='0','1E6','0.2','10.2'
phasearray=[0,18,36,54, 72, 90, 108, 126, 144 ,162, 180,
             198, 216, 234, 252, 270 ,288 ,306, 324, 342]
IDEALAGN='CountRate/ID'+ID+'/COUNT-AGN-'+ET+'-'+Elow+'-'+Ehigh+'.txt'
Barr=loadcountfile(IDEALAGN)
A = Barr['angle']
B = Barr['counts']*20
var1 = np.zeros(len(B))
Serror = []
for phase in phasearray:
    var2 = loadcountfile(IDEAL_PSR(str(phase), Elow, Ehigh))
    var1 = var2['counts'] + var1
    Serror.append(np.sqrt(var2['counts']))
S = var1
fig = plt.figure("Source_Counts",figsize=(5.9,2.2),tight_layout=True)
fig.subplots_adjust(left=None, bottom=None, right=None, top=.85,
                    wspace=None, hspace=None)
gs = gridspec.GridSpec(6,5)
plt.subplot(gs[1,:])
a,b=flatplot(A,B)
plt.plot(a,b/1000000.,label='AGN')
a,s=flatplot(A,S)
plt.plot(a,s/1000000.,label='PSR')
##plt.plot(A,(S+B)/1000000.)
##errs=[]
##for i in range(len(A)):
##    v=[(Serror[j][i]/S[i])**2 for j in range(len(phasearray))]
##    errs.append(np.sqrt(sum(v)))
##plt.errorbar(A,S/1000000.,yerr=errs)

#####

```

```

# 2 #
#####
# Poinging angle vs SNR
# All energies (0.2-10.2), sum phases
# Figure name(s) produced: "CRR", #"Percent"
fig = plt.figure("CRR", figsize=(5.9,2.2), tight_layout=True)
fig.subplots_adjust(left=None, bottom=None, right=None, top=.85,
                    wspace=None, hspace=None)

gs = gridspec.GridSpec(6,5)
plt.subplot(gs[1:,:])
ratio=S/B
a,r=flatplot(A,ratio)
L1 = plt.plot(a,r, 'k')
ratioU=unumpy.uarray(S,np.sqrt(S))/unumpy.uarray(B,np.sqrt(B))

##plt.figure("Percent")
### percent Signal
###def sig(r): return r/(r+1)
###def bac(r): return 1/(r+1)
###ps=sig(ratio)
###pn=bac(ratio)
###a,pss=flatplot(A,ps)
###a,pnn=flatplot(A,pn)
###plt.plot(a,pss)
###plt.plot(a,pnn)

#####
# 3 #
#####
# Phase vs counts (lightcurve) at promising pointing angles.
# All energies (0.2-10.2), sum phases
# Figure name(s) produced: "Lightcurve", "dS"
ANGLE=[2.5,3.0]
IDEALAGN='CountRate/ID'+ID+' /COUNT-AGN-'+ET+'-'+Elow+'-'+Ehigh+'.txt'
Barr=loadcountfile(IDEALAGN)
A = Barr['angle']
fig = plt.figure("Lightcurve", figsize=(5.9,2.7), tight_layout=True)
gs = gridspec.GridSpec(6,8)
plt.subplot(gs[1:,-1])
##plt.gcf().set_size_inches(10,5)
fig.subplots_adjust(left=None, bottom=None, right=None, top=.85,
                    wspace=None, hspace=None)

for i in range(len(ANGLE)):
    angind = np.where(A==ANGLE[i])
    B = Barr['counts'][angind]
    LC, LCerr = [], []
    totcts = 0
    for phase in phasearray:
        var2 = loadcountfile(IDEALPSR(str(phase), Elow, Ehigh))
        totcts = totcts + var2['counts'][angind[0][0]]
        LC.append(var2['counts'][angind[0][0]]/50000.)
        LCerr.append(np.sqrt(var2['counts'][angind[0][0]]/50000.))
    maxind=np.where(LC==max(LC))[0][0]
    minind=np.where(LC==min(LC))[0][0]
    LCmax=ufloat(LC[maxind], LCerr[maxind])
    LCmin=ufloat(LC[minind], LCerr[minind])
    fp = (LCmax-LCmin)/LCmax
    ph,lc=flatplot(phasearray,LC)
    ph,lc=np.array(list(ph)+list(ph+360.)), list(lc)+list(lc)
    ax, = plt.plot(ph/360.,lc, ls='-', lw=1)##, label=str(ANGLE[i])+"")
    cc=plt.getp(ax, 'color')
    plt.plot([-9/360.,(351.+360.)/360.], [B/50000.,B/50000.],
            color=cc, ls=':', lw=1)
    LC=[]
    LCerr=[]
    totcts2=0
    for phase in phasearray:

```



```

        var2 = loadcountfile(IDEAL_OBS(str(phase), Elow, Ehigh))
        totcts2 = totcts2 + var2['counts'][angind[0][0]]
        LC.append(var2['counts'][angind[0][0]]/50000.)
        LCerr.append(np.sqrt(var2['counts'][angind[0][0]]/50000.)
    ph,lc=flatplot(phasearray,LC)
    ph,lc=np.array(list(ph)+list(ph+360.)),list(lc)+list(lc)
    plt.plot(ph/360.,lc,ls='-',color=cc,label=str(ANGLE[i])+""")
    print totcts
    print totcts2
    print fp
##for phase in phasearray:
##    plt.plot([(phase+171)/360.,(phase+171)/360.],[0,1.8],lw=.7,ls='-',color='
    grey')
##plt.plot([(phase+9)/360.,(phase+9)/360.],[0,1.8],lw=.7,ls='-',color='grey')

# CONTOUR of dS
CTS, RAT = np.logspace(5.5,6.5,100), np.linspace(.1,10,100)
def dS(cts, rat):
    T,R = cts, rat
    S = (T*R)/(R+1)
    B = T/(R+1)
    dT = np.sqrt(S)+np.sqrt(B)
    dR = R * np.sqrt((np.sqrt(S)/S)**2 + (np.sqrt(B)/B)**2 )
    ScalC = S
    dScalC = ScalC * np.sqrt((dT/T)**2 + (dR/(R*(R+1)))**2 )
    return dScalC/ScalC
ccc, rrr=np.meshgrid(CTS,RAT)

plt.figure("dS",dpi=100)
plt.contourf(ccc, rrr, dS(ccc, rrr))#, levels=np.logspace(-6,-2.5,100))
##plt.contour(ccc, rrr, (ccc*rrr)/(rrr+1), cmap=cm.gray)
plt.colorbar()
plt.xscale('log')
plt.yscale('log')

#####
## Figure Features
#####
plt.figure("Source_Counts")
plt.grid(axis='both', which='both')
plt.ylim(0,2)
plt.ylabel('Count_Rate_(cts/s)')
plt.xlabel("Pointing_Angle_(')")
plt.legend(loc=0)
plt.arrow(0, .8, 0, .4, head_width=0.2, head_length=0.1, fc='k', ec='k')
plt.annotate('on-axis\nwith\npulsar', xy=(-.2,.2),
            size=matplotlib.rcParams['legend.fontsize'])
plt.suptitle('Source_Count_Rates_at_Off-Axis_Pointing_Angles\n'+
            'Ideal_Case_Alignment: 0.2-10.2 keV')#,
plt.xlim(-1,3.5)
plt.savefig('Figures/IDEALCASE-countrateplot.eps')

plt.figure("CRR")
plt.arrow(0, 7, 0, -5, head_width=0.2, head_length=0.7, fc='k', ec='k')
plt.annotate('on-axis\nwith\npulsar', xy=(-.2,8.2),
            size=matplotlib.rcParams['legend.fontsize'])
plt.suptitle('Expected_Signal_to_Noise_Ratio_at_Off-Axis\n'+
            'Ideal_Case_Alignment: 0.2-10.2 keV')
plt.ylabel('Pulsar_counts_/AGN_Counts')
plt.xlabel("Pointing_Angle_(')")
plt.grid(axis='both', which='both')
plt.xlim(-1,3.5)
plt.arrow(2.5, 4.5, 0, 3, head_width=0.2, head_length=0.7, fc='r', ec='r')
plt.annotate("2.5""", xy=(2.43,3.7), size=matplotlib.rcParams['legend.fontsize'])

```

```

plt.arrow(3.0, 5.4, 0, 3, head_width=0.2, head_length=0.7, fc='r', ec='r')
plt.annotate("3.0", xy=(2.93, 4.6), size=matplotlib.rcParams['legend.fontsize'])
##plt.arrow(2.9, 6, 0, 2, head_width=0.07, head_length=0.3, fc='k', ec='k')
##plt.annotate("2.6", xy=(2.5, 5.3), size=14)
##plt.arrow(3.2, 5, 0, 2, head_width=0.07, head_length=0.3, fc='k', ec='k')
##plt.annotate("3.2", xy=(3.10, 4.3), size=14)
plt.savefig('Figures/IDEALCASE-countratioplot.eps')

plt.figure("Lightcurve")
##plt.grid(axis='both', which='both')
plt.ylabel('Counts_per_phase_bin')
plt.xlabel('Rotational_Phase')
plt.legend(bbox_to_anchor=(1.03, .5), loc=2, borderaxespad=0.)
plt.suptitle('Lightcurve\n'+ 'Ideal_Case_Alignment: 0.2-10.2 keV')
plt.figtext(.865, .60, 'Solid: Total\nDash: Pulsar\nDot: AGN',
            size=matplotlib.rcParams['legend.fontsize'])
plt.xlim(0, 2)
plt.ylim(0, 1.8)
h, v = .12, .34
plt.figtext(h, v, "3.0 'Counts:",
            size=matplotlib.rcParams['legend.fontsize'])
plt.figtext(h+.13, v, r"$N_{Total}=9.25\times 10^5\pm 962$, "+
            r"$N_{Pulsar}=8.41\times 10^5\pm 917$, "+
            r"$f_p=43.7\%\pm 0.4\%$",
            size=matplotlib.rcParams['legend.fontsize'])
h, v = .12, .28
plt.figtext(h, v, "2.5 'Counts:",
            size=matplotlib.rcParams['legend.fontsize'])
plt.figtext(h+.13, v, r"$N_{Total}=1.24\times 10^6\pm 1110$, "+
            r"$N_{Pulsar}=1.11\times 10^5\pm 1054$, "+
            r"$f_p=43.5\%\pm 0.4\%$",
            size=matplotlib.rcParams['legend.fontsize'])
plt.savefig('Figures/IDEALCASE-lightcurveplot.eps')

plt.show()

```

Source File: ReconstructPulsarSignal.py

```

import numpy as np
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
import matplotlib.cm as cm
from uncertainties import ufloat
from uncertainties.umath import *
from uncertainties import unumpy
from matplotlib.ticker import MultipleLocator, FormatStrFormatter
from scipy import interpolate
from scipy.optimize import curve_fit
from scipy.optimize import brentq

import GetIdealSNR2 as GISNR
import GetObservedLightCurves2 as GOLC

import matplotlib
matplotlib.rcParams['font', **{'size': 12}]
matplotlib.rcParams['axes', **{'titlesize': 12}]
##matplotlib.rcParams['figure', **{'size': 12}]
matplotlib.rcParams['axes', **{'labelsize': 10}]
matplotlib.rcParams['xtick', **{'labelsize': 8}]
matplotlib.rcParams['ytick', **{'labelsize': 8}]
matplotlib.rcParams['legend', **{'fontsize': 8}]
matplotlib.rcParams['legend', frameon=False]
matplotlib.rcParams['lines', lw=1]
matplotlib.rcParams['figure', **{'dpi': 100}]

'''
- collect for list of energy ranges:

```

```

        [['0.5','0.65'], ['0.65','0.85'], ['0.85','1.2'], ['1.2','1.8'],
         ['0.5','1.8'], ['0.2','8.0']]
- collect at every phase
        ['0','18','36','54','72','90','108','126','144','162',
         '180','198','216','234','252','270','288','306','324','342']
- collect at several pointing angles

Input file:
    LightCurveFiles/SNR/IdealSNR-PointingAngle<ANGLE>-E<el>-<eh>.txt
- Columns: Phase    SNR    dSNR
Input file:
    LightCurveFiles/Observed/ID<id>/Observed-PointingAngle<ANGLE>-E<el>-<eh>.txt
- Columns: Phase    S+B    d(S+B)
,,,

def pulsefrac(array):
    return (max(array)-min(array))/max(array)

def signal(observedLC, snrLC):
    # S = T*R/(R+1), T=S+B, R=S/B
    T,R=observedLC, snrLC
    return (T*R)/(R+1)

def background(observedLC, snrLC):
    # B = T/(R+1), T=S+B, R=S/B
    T,R=observedLC, snrLC
    return T/(R+1)

def flatplot(x,y):
    a = np.array(x)
    da = a[1:]-a[:-1]
    b = a[:-1]+da/2.
    acopy=list(a.copy())
    acopy[1:]=a[1:]-da/2.
    [acopy.insert(ind*2+1,b[ind]) for ind in range(len(b))]
    acopy[0]=acopy[0]-(da[0]/2.)
    acopy.append(acopy[-1]+da[-1])
    X = np.array(acopy).copy()
    del a,b,acopy
    a = np.array(y)
    b = y[:-1]
    acopy=list(a.copy())
    [acopy.insert(ind*2+1,b[ind]) for ind in range(len(b))]
    acopy.append(acopy[-1])
    Y = np.array(acopy).copy()
    return X,Y

,,,

#####
## Run this part of the code to calculate the pulsar
## light curve and compare to synthertic LC in a
## figure. Run this one offset pointing angle
## at a time.
#####
#####
## 1 ##
#####
useangle='2.7'
##idlist=['11']
idlist=['1','2','3','4','5','6','7','8','9','10',
        '11','12','13','14','15','16','17','18','19','20']

diff=[]
actu=[]
fpdiff=[]
myerrors=[]

```

```

Energies=[[ '0.2', '0.5'], [ '0.5', '3.0'], [ '3.0', '10.2']]
for idd in idlist:
    fig = plt.figure(" LightCurve", figsize=(5.9,7))
    gs = gridspec.GridSpec(15,7, wspace=0, hspace=0)
    SPs1=[gs[:4,:-1], gs[4,:-1], gs[5:9,:-1], gs[9,:-1],
          gs[10:14,:-1], gs[14,:-1]]
    diff.append([])
    actu.append([])
    fpdiff.append([])
    myerrors.append([])
    i=0
    for ei in Energies:
        plt.subplot(SPs1[i])
        i=i+1
        ID, Elow, Ehigh, Angle= idd, ei[0], ei[1], useangle
        AssumedSNR=GISNR.calcSNRarray('0', Elow, Ehigh, Angle)
        ObservedLC = GOLC.OBSarray(ID, Elow, Ehigh, Angle)
        phases = ObservedLC[0]
        # Calculated vs. Actual Signal
        CalculatedSignal = signal(ObservedLC[1], AssumedSNR[1])
        ActualSignal = GOLC.realPSRarray(ID, Elow, Ehigh, Angle)[1]
        # plot calculated signal lightcurve
        phases = np.array(phases)/360.
        CSx, CSy = phases, unumpy.nominal_values(CalculatedSignal.T[0])/50000.
        CSx_flt, CSy_flt = flatplot(CSx, CSy)
        CSy_err = unumpy.std_devs(CalculatedSignal.T[0])/50000.
        ax, = plt.plot(CSx_flt, CSy_flt, color='r',
                      label='Calculated\nSignal')
        cc=plt.getp(ax, 'color')
        plt.errorbar(CSx, CSy, yerr=CSy_err,
                    color=cc, fmt=' ')
        # plot acutal signal light curve
        ASx, ASy = phases, unumpy.nominal_values(ActualSignal.T[0])/50000.
        ASx_flt, ASy_flt = flatplot(ASx, ASy)
        ASy_err = unumpy.std_devs(ActualSignal.T[0])/50000.
        plt.plot(ASx_flt, ASy_flt, color='grey', label='Synthetic\nSignal')
        # plot observed Light Curve
        Ox, Oy = phases, unumpy.nominal_values(ObservedLC[1].T[0])/50000.
        Ox_flt, Oy_flt = flatplot(Ox, Oy)
        plt.plot(Ox_flt, Oy_flt, color='k',
                label='Observed')
        plt.title(ei[0]+'-'+ei[1]+' \n keV', y=.64, x=1.15)
        # plot residual (% diff)
        ax=plt.subplot(SPs1[i])
        i=i+1
        resid = unumpy.nominal_values(100*((ActualSignal-CalculatedSignal)/
                                           ActualSignal).T[0])
        resid_err = unumpy.std_devs(100*((ActualSignal-CalculatedSignal)/
                                           ActualSignal).T[0])
        plt.errorbar(phases, resid, yerr=resid_err, color='grey', fmt='+', mew=2,
                    label='Syn vs Calc\nSig % Diff')
        ax.yaxis.tick_right()
        majorLocator = MultipleLocator((max(resid_err)*1.9)/1.7)
        ax.yaxis.set_major_locator(majorLocator)
        ax.yaxis.set_major_formatter(FormatStrFormatter('%d'))
    for i in [0,1,2,3,4,5]:
        ax=plt.subplot(SPs1[i])
        plt.xlim(-.05,1)
        if i in [1,3,5]:
            plt.legend(bbox_to_anchor=(1.03,1), loc=2,
                      borderaxespad=0., numpoints=1)
        if i in [0,2,4]:
            plt.legend(bbox_to_anchor=(1.03,.6), loc=2, borderaxespad=0.)
            ax.yaxis.set_major_formatter(FormatStrFormatter('%1.1e'))
        if i<5: ax.set_xticklabels([])
        if i==5: ax.set_xlabel('Rotational Phase')
    plt.suptitle('Signal Estimated Lightcurves\nNICER Instance ID'+idd)
    plt.figtext(.01,.6, 'Count Rate (cts/s)', rotation=90)

```

```

plt.figtext(.9,.94,'Angle\n '+useangle+'")
plt.savefig('Figures2/EstimateSignal-lightcurveID'+idd+'Ang'+
            useangle+'.eps',pad_inches=0)
plt.close('all')
##plt.show()
,,,

#####
## Run this part of the code to generate a text file
## with optimal pointing directions for each ID.
## This file is used in LCoOptimatPtAng.py
#####
#####
# 2 #
#####

angleslist=[-1.0, -0.5, 0.0, 0.5, 1.0,
            1.5, 2.0, 2.1, 2.2, 2.3,
            2.4, 2.5, 2.6, 2.7, 2.8,
            2.9, 3.0, 3.1, 3.2, 3.3,
            3.4, 3.5]
##angleslist=[3.0]
##idlist=['1','2','3','4','5','6','7','8','9','10',
##        '11','12','13','14','15','16','17','18','19','20']
##idlist=['11']#,'12','13']
##Energies=[[ '0.2','10.2'],[ '0.2','0.5'],[ '0.5','3.0'],[ '3.0','10.2'],
##           [ '0.5','0.65'],[ '0.65','0.85'],[ '0.85','1.2'],[ '1.2','1.8'],
##           [ '0.5','1.8'],[ '0.2','8.0']]
##ei = Energies[0]
Energies=[[ '0.2','10.2']]

result=[]
for ei in Energies:
    ## Angle of Convergence
    fig = plt.figure('PerDiff',figsize=(5.9,3))
    fig.subplots_adjust(left=None, bottom=.15, right=None, top=.9,
                        wspace=None, hspace=None)

    Zero=[]
    ANGfile=open('Figures2/AanglefileID'+ei[0]+'-'+ei[1]+' .txt','w')
    for idd in idlist:
        diff = np.array([])
        for ANGLE in angleslist:
            # This loop gets data for ANGfile
            ID, Elow, Ehigh, Angle= idd, ei[0], ei[1], str(ANGLE)
            AssumedSNR=GISNR.calcSNRarray('0', Elow, Ehigh, Angle)
            ObservedLC = GOLC.OBSarray(ID, Elow, Ehigh, Angle)
            phases = ObservedLC[0]
            # Calculated vs. Synthetic Signal
            CalculatedSignal = signal(ObservedLC[1], AssumedSNR[1])
            ActualSignal = GOLC.realPSRarray(ID, Elow, Ehigh, Angle)[1]
            # (Act-Calc)/Act % Diff
            AS, CS = ActualSignal, CalculatedSignal
            PerDiff = 100.*((AS-CS)/AS)
            diff = np.append(diff, np.sum(PerDiff))

        # Plot % Diff
        diffY, diff_err = unumpy.nominal_values(diff), unumpy.std_devs(diff)
        x = np.arange(-1,3.6,.1)
        plt.plot([-1,3.5],[0,0], 'k—')
        plt.errorbar(angleslist,diffY, yerr=diff_err,color='b',
                    fmt='.', mew=1)

        # fit to curve
        x = np.arange(-1,3.6,.1)
        # fit 1
        def curve(x,b,c):
            return b*x + c
        fit = curve_fit(curve,np.array(angleslist),
                        diffY,sigma=diff_err,maxfev=10000)#,sigma=diff_err)

```

```

plt.plot(x, curve(x,* fit [0]), color='grey')
b = unumpy. uarray ( fit [0][0], np. sqrt ( np. diag ( fit [1]) ) [0])
c = unumpy. uarray ( fit [0][1], np. sqrt ( np. diag ( fit [1]) ) [1])
Zero.append(-c/b)
##      print idd,unumpy. nominal_values ( Zero [-1]),unumpy. std_devs ( Zero [-1])
ANGfile. write (idd+'_###'+str (unumpy. nominal_values ( Zero [-1]))+'_##'+
                str (unumpy. std_devs ( Zero [-1]))+'\n')
ANGfile. close ()
##      #      fit 2 # Alternate fit I tried
##      def curve(x,a,b,c,d):
##          return a*np.exp(-(x-b)/c)+d
##      fit = curve_fit (curve, np. array ( angleslist ), diffY, maxfev=10000)
##      plt. plot (x, curve (x,* fit [0]))
##      a,b,c,d=fit [0][0], fit [0][1], fit [0][2], fit [0][3]
##      print 'Zero:', b - c * np. log (-d/c)
##      plt. close ('all')
result. append ([ ei [0], ei [1], np. mean (Zero)])

#####
## Uncomment below to make figure "PerDiff" above
## for a single NICER ID and single Energy Band
## -change arrays before loop above to represent
## only 1 ID and 1 energy band.
#####

plt. figure ('PerDiff')
plt. title ('Relative_Difference_of_Estimated_Lightcurve_from_Synthetic')
plt. xlabel ("Pointing_Angle_(')")
plt. ylabel ('Lightcurve_%_Difference')
plt. figtext (.15,.22,
             r '$\sum_{\{phase\}} \frac{LC[syn]-LC[est]}{LC[syn]} \times 100 \%$',
             size=14)
plt. arrow (2.41, 50, 0, -20, head_width=0.25, head_length=10, fc='k', ec='k')
plt. figtext (.62,.65, '{:.2u}'.format (np. mean (Zero)))
plt. figtext (.57,.83, 'ID_11; ~'+ei [0]+'-'+ei [1]+'_keV', fontstyle='italic')
plt. xlim (-1.1, 3.6)
plt. savefig ('Figures2/LCdiffID11. eps', pad_inches=0)
plt. show ()

```

Source File: GetObservedLightCurves.py

```

import numpy as np
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
import matplotlib.cm as cm
from uncertainties import ufloat
from uncertainties.umath import *
from uncertainties import unumpy

'''
This script calculated the Observed light curve.

- collect for list of energy ranges:
  [[ '0.5', '0.65'], [ '0.65', '0.85'], [ '0.85', '1.2'], [ '1.2', '1.8'],
    [ '0.5', '1.8'], [ '0.2', '8.0']]
- collect at every phase
  [ '0', '18', '36', '54', '72', '90', '108', '126', '144', '162',
    '180', '198', '216', '234', '252', '270', '288', '306', '324', '342']
- collect at several pointing angles

Output files: 'LightCurveFiles/Observed/ID'+IDs[i]+'/Observed-PointingAngle'+
              str (ANGLEarray [angleindex])+'-E'+Ens+'.txt'
- Columns: Phase      S+B      d(S+B)
'''

```

```

def loadcountfile(filename):
    dtype={'names':('angle','counts'),'formats':(float,float)}
    data = np.loadtxt(filename,dtype=dtype)
    return np.sort(data,order='angle')

def uncertainty(counts):
    return np.sqrt(counts)

def PSR(phase, Elow, Ehigh, ID, ET='1E6'):
    return 'CountRate/ID'+ID+'/COUNT-McPHAC'+phase+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'

def AGN(Elow, Ehigh, ID, ET='1E6'):
    return 'CountRate/ID'+ID+'/COUNT-AGN'+phase+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'

def OBSERVED(phase, Elow, Ehigh, ID, ET='1E6'):
    return 'CountRate/ID'+ID+'/COUNT-OBS'+phase+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'

def OBSarray(ID, Elow, Ehigh, ANGLE, ET='1E6', out=True):
    phasearray=[0,18,36,54, 72, 90, 108, 126, 144 ,162, 180, 198, 216, 234,
                252, 270 ,288 ,306, 324, 342]
    t=[]
    for p in phasearray:
        T_array=loadcountfile(OBSERVED(str(p), Elow, Ehigh, ID, ET))
        angleindex=np.where(T_array['angle']==float(ANGLE))
        t.append(T_array['counts'][angleindex])
    t=np.array(t)
    T=unumpy.uarray(t,np.sqrt(t))
    if out:
        outfile='Figures2/OBS-ID'+ID+'-Angle'+ANGLE+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'
        COL1,COL2,COL3=phasearray,unumpy.nominal_values(T),unumpy.std_devs(T)
        np.savetxt(outfile, np.transpose([COL1,COL2,COL3]))
    return phasearray, T

def realPSRarray(ID, Elow, Ehigh, ANGLE, ET='1E6', out=True):
    phasearray=[0,18,36,54, 72, 90, 108, 126, 144 ,162, 180, 198, 216, 234,
                252, 270 ,288 ,306, 324, 342]
    s=[]
    for p in phasearray:
        S_array=loadcountfile(PSR(str(p), Elow, Ehigh, ID, ET))
        angleindex=np.where(S_array['angle']==float(ANGLE))
        s.append(S_array['counts'][angleindex])
    s=np.array(s)
    S=unumpy.uarray(s,np.sqrt(s))
    if out:
        outfile='Figures2/ACTUALPSR-ID'+ID+'-Angle'+ANGLE+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'
        COL1,COL2,COL3=phasearray,unumpy.nominal_values(S),unumpy.std_devs(S)
        np.savetxt(outfile, np.transpose([COL1,COL2,COL3]))
    return phasearray, S

```

Source File: GetIdealSNR.py

```

import numpy as np
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
import matplotlib.cm as cm
from uncertainties import ufloat
from uncertainties.umath import *
from uncertainties import unumpy

```

'''

This script calculated the idea SNR for

the pulsar signal to the AGN background counts.

```

- Use ID 30.
- collect for list of energy ranges:
    [[ '0.5', '0.65'], [ '0.65', '0.85'], [ '0.85', '1.2'], [ '1.2', '1.8'],
      [ '0.5', '1.8'], [ '0.2', '8.0']]
- collect at every phase
    [ '0', '18', '36', '54', '72', '90', '108', '126', '144', '162',
      '180', '198', '216', '234', '252', '270', '288', '306', '324', '342']
- collect at several pointing angles

Output files: IdealSNR-PointingAngle<ANGLE>-E<el>-<eh>.txt
- Columns: Phase SNR dSNR
,,,
```

```

def loadcountfile(filename):
    dtype={ 'names':( 'angle', 'counts'), 'formats':( float, float)}
    data = np.loadtxt(filename, dtype=dtype)
    return np.sort(data, order='angle')

def ratiouncertainty(numerator, denominator):
    x,y=numerator, denominator
    r=x/y
    a=( np.sqrt(x)/x )**2 + ( np.sqrt(y)/y )**2
    return r*np.sqrt(a)

## LIGHT CURVE = PHASE vs COUNTS
## options: Energy Range; Pointing Angle; NICER ID Number, Exposure
ID, ET, Elow, Ehigh, ANGLE = '0', '1E6', '0.5', '1.8', '3.0'
IDEALAGN='CountRate/ID'+ID+'/COUNT-AGN-'+ET+'-'+Elow+'-'+Ehigh+'.txt'

def IDEALPSR(phase, Elow, Ehigh, ID='0', ET='1E6'):
    return 'CountRate/ID'+ID+'/COUNT-McPHAC'+phase+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'

def calcSNRarray(ID, Elow, Ehigh, ANGLE, ET='1E6', out=True):
    B_array=loadcountfile('CountRate/ID'+ID+'/COUNT-AGN-'+ET+'-'+Elow+'-'+Ehigh+'.txt')
    angleindex=np.where(B_array['angle']==float(ANGLE))
    phasearray=[0,18,36,54, 72, 90, 108, 126, 144, 162, 180, 198, 216, 234, 252,
                 270, 288, 306, 324, 342]
    s,b=[],[]
    for p in phasearray:
        S_array=loadcountfile(IDEALPSR(str(p), Elow, Ehigh))
        s.append(S_array['counts'][angleindex])
        b.append(B_array['counts'][angleindex])
    s,b=np.array(s), np.array(b)
    S=unumpy.uarray(s, np.sqrt(s))
    B=unumpy.uarray(b, np.sqrt(b))
    snr=S/B
    if out:
        COL1,COL2,COL3=np.array(phasearray), np.array(unumpy.nominal_values(snr)), np
            .array(unumpy.std_devs(snr))
        outfile='Figures2/SNR-ID'+ID+'-Angle'+ANGLE+'-'+ET+'-'+Elow+'-'+Ehigh+'.txt'
        np.savetxt(outfile, np.transpose([COL1,COL2,COL3]))
    return phasearray, snr
```

Source File: LCoOptimalPtAng.py

```

import numpy as np
import matplotlib.lines as mlines
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
import matplotlib.cm as cm
from uncertainties import ufloat
from uncertainties.umath import *
```



```

from uncertainties import unumpy
import matplotlib

matplotlib.rc('font',**{'size':12})
matplotlib.rc('axes',**{'labelsize':10})
matplotlib.rc('xtick',**{'labelsize':8})
matplotlib.rc('ytick',**{'labelsize':8})
matplotlib.rc('legend',**{'fontsize':8})
matplotlib.rc('lines', lw=1)
matplotlib.rc('figure', **{'dpi':100})

En=[[ '0.5', '0.65'], [ '0.65', '0.85'], [ '0.85', '1.2'], [ '1.2', '1.8'],
     [ '0.5', '1.8'], [ '0.2', '8.0'], [ '0.2', '0.5'], [ '0.5', '3.0'],
     [ '3.0', '10.2'], [ '0.2', '10.2']]
idlist=[ '1', '2', '3', '4', '5', '6', '7', '8', '9', '10',
          '11', '12', '13', '14', '15', '16', '17', '18', '19', '20']
fig=plt.figure(figsize=(5.9,5))
fig.subplots_adjust(left=None, bottom=.2, right=None, top=.9,
                    wspace=None, hspace=None)
ax=plt.subplot(111)
for i in range(len(En)):
    E1,E2=En[i][0],En[i][1]
    datfile='anglefileID'+E1+'-'+E2+'.txt'
    # angle file headers: ID OptimalPtAngle OptPtAngError
    idn, a, er = np.loadtxt(datfile,unpack=True)
    plt.plot(np.zeros(20)+(i),a,'k.')
    # plots average optimal pt ang for each e-band
    plt.plot(i,np.mean(a),'rx')
    plt.annotate(str(np.round(np.mean(a),2))+",",
                 xy=(i+.15,np.mean(a)-.02),size=8)
lab=[]
for i in range(len(En)):
    E1,E2=En[i][0],En[i][1]
    lab.append(E1+'-'+E2+'keV')

dots = plt.plot([], [], marker='.',color='black',lw=0,
                 markersize=5, label='Results_per_NICER_ID')
plt.legend(loc=3)
plt.xlabel('Energy_Range')
plt.ylabel('Optimal_Pointing_Angle')
plt.title('Optimal_Pointing_Angles_for_Each_NICER_Instance')
plt.xticks(range(len(En)), lab, rotation=-30)
plt.xlim(-1,len(En))
plt.savefig('optimalptang2.eps')
plt.show()

vals=[]
for i in range(len(En)):
    E1,E2=En[i][0],En[i][1]
    datfile='anglefileID'+E1+'-'+E2+'.txt'
    idn, a, er = np.loadtxt(datfile,unpack=True)
    vals.append(unumpy.unicode_array(a,er))
    thismean=np.mean(vals[-1])
    thisval=unumpy.nominal_values(thismean)
    thisstd=np.std(unumpy.nominal_values(vals[-1]))
    thissdom=unumpy.std_devs(thismean)
    print E1,'-',E2,'_&_&_',thisval,'$\\pm$',thisstd,'_&_&_',thissdom

#####
## Loop through each ID and plot
## Optimal Pointing angle w/ Error
## Bars to check Energy Dependence
#####

for thisID in idlist:
    fig=plt.figure(figsize=(5.9,5))
    fig.subplots_adjust(left=None, bottom=.2, right=None,
                        top=.9,wspace=None, hspace=None)

```

```

ax=plt.subplot(111)
for i in range(len(En)):
    E1,E2=En[i][0],En[i][1]
    datfile='anglefileID'+E1+'-'+E2+'.txt'
    # angle file headers: ID OptimalPtAngle OptPtAngError
    idn, a, er = np.loadtxt(datfile,unpack=True)
    # isolate current ID row from datfile
    idindex = np.where(int(thisID)==idn)[0][0]
    (-, caps, -) = plt.errorbar(i,a[idindex],yerr=er[idindex],
                                fmt='+',elinewidth=2,capsize=4,
                                color='r')

    for cap in caps:
        cap.set_markeredgewidth(2)
    # plots optimal pt ang for each e-band
    plt.plot(np.zeros(1)+(i),a[idindex], 'k. ')
    plt.plot(i,np.mean(a), 'bx')

lab=[]
for i in range(len(En)):
    E1,E2=En[i][0],En[i][1]
    lab.append(E1+'-'+E2+'keV')

dots = plt.plot([], [], marker='.',color='black',lw=0,
                 markersize=5, label='Optimal_Offset_Pointing_Angle')
bars = plt.errorbar([], [], yerr=[], fmt='+', elinewidth=2,
                    capsizesize=4,color='r', label='Averaged')
avgss = plt.plot([], [], marker='x',color='blue',lw=0,
                  markersize=5,label='Average_over_all_20_NICER_IDs')
plt.legend(loc=3#[dots],[dots])
plt.xlabel('Energy_Range')
plt.ylabel('Optimal_Pointing_Angle')
plt.title('Optimal_Pointing_Angles_for_NICER_ID_'+thisID)
plt.xticks(range(len(En)), lab, rotation=-30)
plt.xlim(-1,len(En))
plt.ylim(0,5)
plt.savefig('optimalptang-ID'+thisID+'ALLebands.eps')
plt.close()

#####
## Loop through each ID and plot
## Optimal Pointing angle w/ Error
## Bars to check Energy Dependence
## LOW ENERGY BANDS (.05-1.8)
#####
En=[[ '0.5 ', '0.65 '], [ '0.65 ', '0.85 '], [ '0.85 ', '1.2 '], [ '1.2 ', '1.8 ']]
for thisID in idlist:
    fig=plt.figure(figsize=(5.9,5))
    fig.subplots_adjust(left=None, bottom=.2, right=None,
                        top=.9, wspace=None, hspace=None)
    ax=plt.subplot(111)
    for i in range(len(En)):
        E1,E2=En[i][0],En[i][1]
        datfile='anglefileID'+E1+'-'+E2+'.txt'
        # angle file headers: ID OptimalPtAngle OptPtAngError
        idn, a, er = np.loadtxt(datfile,unpack=True)
        # isolate current ID row from datfile
        idindex = np.where(int(thisID)==idn)[0][0]
        (-, caps, -) = plt.errorbar(i,a[idindex],yerr=er[idindex],
                                    fmt='+',elinewidth=2,capsize=4,
                                    color='r')

        for cap in caps:
            cap.set_markeredgewidth(2)
        # plots optimal pt ang for each e-band
        plt.plot(np.zeros(1)+(i),a[idindex], 'k. ')
        plt.plot(i,np.mean(a), 'bx')

```

```

lab=[]
for i in range(len(En)):
    E1,E2=En[i][0],En[i][1]
    lab.append(E1+'-'+E2+'keV')

dots = plt.plot([], [], marker='.',color='black',lw=0,
                 markersize=5, label='Optimal_Offset_Pointing_Angle')
bars = plt.errorbar([], [], yerr=[],fmt='+',elinewidth=2,
                   capsize=4,color='r',label='Averaged')
avgss = plt.plot([], [], marker='x',color='blue',lw=0,
                 markersize=5,label='Average_over_all_20_NICER_IDs')
plt.legend(loc=3#[dots],[dots])
plt.xlabel('Energy_Range')
plt.ylabel('Optimal_Pointing_Angle')
plt.title('Optimal_Pointing_Angles_for_NICER_ID_'+thisID)
plt.xticks(range(len(En)), lab, rotation=-30)
plt.xlim(-1,len(En))
plt.ylim(0,5)
plt.savefig('optimalptang-ID'+thisID+'LOWebands.eps')
plt.close()

#####
## Loop through each ID and plot
## Optimal Pointing angle w/ Error
## Bars to check Energy Dependence
## WIDE RANGE ENERGY BANDS (.2-10.2)
#####
En=[['0.2','0.5'], ['0.5','3.0'], ['3.0','10.2']]
for thisID in idlist:
    fig=plt.figure(figsize=(5.9,5))
    fig.subplots_adjust(left=None, bottom=.2, right=None,
                       top=.9, wspace=None, hspace=None)
    ax=plt.subplot(111)
    for i in range(len(En)):
        E1,E2=En[i][0],En[i][1]
        datfile='anglefileID'+E1+'-'+E2+'.txt'
        # angle file headers: ID OptimalPtAngle OptPtAngError
        idn, a, er = np.loadtxt(datfile,unpack=True)
        # isolate current ID row from datfile
        idindex = np.where(int(thisID)==idn)[0][0]
        (_, caps, _) = plt.errorbar(i,a[idindex],yerr=er[idindex],
                                   fmt='+',elinewidth=2,capsize=4,
                                   color='r')

        for cap in caps:
            cap.set_markeredgewidth(2)
        # plots optimal pt ang for each e-band
        plt.plot(np.zeros(1)+(i),a[idindex], 'k.')
        plt.plot(i,np.mean(a), 'bx')
    lab=[]
    for i in range(len(En)):
        E1,E2=En[i][0],En[i][1]
        lab.append(E1+'-'+E2+'keV')

    dots = plt.plot([], [], marker='.',color='black',lw=0,markersize=5,
                    label='Optimal_Offset_Pointing_Angle')
    bars = plt.errorbar([], [], yerr=[],fmt='+',elinewidth=2,capsize=4,
                      color='r',label='Averaged')
    avgss = plt.plot([], [], marker='x',color='blue',lw=0,markersize=5,
                    label='Average_over_all_20_NICER_IDs')
    plt.legend(loc=3#[dots],[dots])
    plt.xlabel('Energy_Range')
    plt.ylabel('Optimal_Pointing_Angle')
    plt.title('Optimal_Pointing_Angles_for_NICER_ID_'+thisID)
    plt.xticks(range(len(En)), lab, rotation=-30)
    plt.xlim(-1,len(En))
    plt.ylim(0,5)

```

```
plt.savefig('optimalptang-ID'+thisID+'WIDEbands.eps')  
plt.close()
```

Appendix F

AGN RX J0437.4-4711 Variability in the Literature

The literature shows that this AGN is not completely stable over time (Halpern and Marshall, 1996). Since the flux of this AGN determines the confidence levels for the pulsar observation, it would be convenient to observe the pulsar when AGN flux is low. However, there are very few observations available for the source in the soft X-ray band and the long term timing analysis is limited to a single 20-day observation conducted by EUVE's (Extreme Ultraviolet Explorer) Deep Survey (DS) instrument in October to November 1994, sensitive in the range 65-200Å, or 60-190 eV. Even though the range is lower, the spectra corresponds nicely to the ROSAT data from 1992 (Halpern and Marshall, 1996). This means that it is likely the same variability exists at the NICER energy range of 0.2-10.2 keV.

Flux measurements over several years from different observations may show to be useful in providing limits to the X-ray variability. Table F.1 summarizes the literature and observations that provide useful information on several flux measurements.

F.1 Wang et al. (1998)

Wang et al. (1998) reported on spectral analysis of the AGN using observations from ASCA and ROSAT. The ASCA observations are from 1993 (observation

AGN RX J0437-4711 Observations			
Instrument	Observation ID	Date	Exposure (s)
ROSAT RASS II	?	Jul 30, 1990	1037
<i>Grupe et al. (2001)</i>			
ROSAT PSPC	RP701184N00	Sept 20, 1992	6142
<i>Wang et al. (1998); Halpern and Marshall (1996); Grupe et al. (2001)</i>			
ASCA SIS&GIS	41013000	Nov 20, 1993	22,800\22,900 (SIS) 24,10 (GIS)
<i>Wang et al. (1998)</i>			
ASCA SIS&GIS	41013010	Feb 25, 1994	13,000\13,200 (SIS) 18,300 (GIS)
<i>Wang et al. (1998)</i>			
EUVE DS	PSR J0437-4715	Oct 20, 1994	495,897
<i>Halpern and Marshall (1996)</i>			

Table F.1: Summary of AGN RX J0437-4711 Observations

41013000) and 1994 (41013010) with instruments SIS0, SIS1, GIS2 and GIS3 (Solid-state Imaging Spectrometer and Gas Imaging Spectrometer). The pointed ROSAT observation is from 1992 (observation RP701184N00) with the PSPC (Position Sensitive Proportional Counters) instrument.

The ROSAT spectrum was fitted best to an absorbed powerlaw plus blackbody model using energies 0.1-2.0 keV only. The ASCA data however was fit only to an absorbed powerlaw using energies 1.0-5.5 keV. The blackbody is present for the soft X-ray excess resolved in the ROSAT data but the excess is too soft for the ASCA observations and left out of the models. The differences in photon index was believed to be due to the sensitivity of the ROSAT spectrum to a thermal component and poor spectral resolution. Wang choose to constrain the column density to that found by Heiles and Cleary (1979).

They then fit the ROSAT data the 1993 ASCA data simultaneously to an absorbed powerlaw plus blackbody to account for the weaknesses in both observations. The powerlaw normalization variables were left to vary separately and showed only a 10% difference in observations. The photon index was the same (within error) to that of the ASCA data fitted alone suggesting consistency of the AGN's spectral hardness. However flux was the largest noticeable change between the observations. There was a 47% increase in flux between the ASCA 1993 and 1994 observations and a 35% increase between a previous ROSAT observation in 1990 and the reports 1992 observation.

Wang et. al. also discusses an explanation for the soft excess shown in the spectra pointing out that this excess is dominated by the blackbody component. The blackbody peak is about 0.2 keV, right around the peak for the pulsar according Bogdanov's atmospheric model (Bogdanov, 2013) yet at the tail end of NICER's sensitivity.

F.2 Halpern and Marshall (1996)

Halpern and Marshall (1996) studied and reported on the long, 20-day, EUVE observation of RX J0437-4711. Included in this paper are the modeled spectra for the ROSAT 1992 observation, the spectral and timing analysis of the EUVE observation and a brief spectral analysis from an International Ultraviolet Explorer (IUE) 1992 observation.

The ROSAT data is fit to an absorbed powerlaw with energies 0.11-2.47 keV

included in the fit. The fitted photon index was lower than the one found by Wang et al. (1998) but so was the column density which was not constrained to the 1979 survey. The spectra fit for the EUVE data agrees with the fit found.

The timing analysis from the 20-day observation showed variability in the count rate by a factor of 4 over the whole exposure. Rapid changes were characterized by a minimum doubling time of 5 hours. The modeled ROSAT spectrum folded with the EVUE DS response showed the 1992 ROSAT data count rate to be in agreement with the EUVE data. No significant variability was found in the short 1992 ROSAT observation. The power spectrum fits a powerlaw with index of -1.25 ± 0.25 .

The rapid changes are more frequent in the second half of the observation and are considered to be a flaring state. The period of these changes was found to be about 0.906 days. However, a later paper by Halpern et al. (2003) shows a power spectrum of only the second half of the observation. The same signal is enhanced but at 0.89 days. Theory suggests that the variability in the EUV is actually the variability of the primary energy source which we are able to see if there is an absence of a substantial electron-scattering corona (Green et al., 1993). However, there is no reason to believe that the signal here is periodic or persistent, there is not enough evidence. While theory may suggest transient periods in Seyfert galaxies that would cause a flaring state such as the one observed here, there is not enough observational proof Halpern and Marshall (1996).

F.3 Grupe et al. (2001)

Grupe et. al. studied a sample of 113 AGN (a complete set (at the time) for all AGN meeting criteria: $\text{rate} \geq 0.5 \text{ cts/s}$, $\text{HR1} < 0.00$, and $|b| > 20^\circ$) and compared the ROSAT RASS data from 1990 to the ROSAT pointed data from 1992 in order to find new X-ray transient AGN. X-ray transience, the change in X-ray emission, is believed to be caused by changes in the accretion rate and/or changes in the column density. In AGN the flux may change on the order of years, hence the study compares data 2 years apart.

RX J0437-4711 was not found to be transient in this report and not one of the most variable. A trend was found that the low luminosity sources are more variable. AGN RX J0437-4711 is considered to be very soft and thus of low luminosity.

F.4 Summary of Variability

Short term variability follows a powerlaw with index of -1.25 ± 0.25 . While periods outside of this power spectrum may be characteristic, there is not enough data to tell if it is persistent feature. During the flare state in the 20-day EUVE observation, the period was 0.89 days but on average the short term variability was 0.906 days. Analysis of the EUVE observation reported the maximum luminosity for 0.1–2.4 keV was $5 \times 10^{44} \text{ ergs/s}$ and the average was $2 \times 10^{44} \text{ ergs/s}$ over the 20 days.

Bibliography

- K. Arnaud, R. Smith, and A. Siemiginowska. *Handbook of X-ray Astronomy*. Cambridge Observing Handbooks for Research Astronomers. Cambridge University Press, 2011. ISBN 9780521883733.
- K. A. Arnaud. XSPEC: The First Ten Years. In G. H. Jacoby and J. Barnes, editors, *Astronomical Data Analysis Software and Systems V*, volume 101 of *Astronomical Society of the Pacific Conference Series*, page 17, 1996.
- K. A. Arnaud, B. Dorman, and C. Gordon. *Xspec: An X-ray Spectral Fitting Package User's Guide for version 12.8.1*. HEASARC Astrophysics Science Division NASA GSFC, August 2013.
- Z. Arzoumanian, S. Bogdanov, J. Cordes, et al. X-ray Timing of Neutron Stars, Astrophysical Probes of Extreme Physics. In *astro2010: The Astronomy and Astrophysics Decadal Survey*, volume 2010 of *Astronomy*, 2009. URL <http://adsabs.harvard.edu/abs/2009astro2010S...6A>.
- Z. Arzoumanian, K. C. Gendreau, C. L. Baker, et al. The neutron star interior composition explorer (NICER): mission definition. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 9144, pages 914420–914420–9, 2014. doi: 10.1117/12.2056811. URL <http://dx.doi.org/10.1117/12.2056811>.
- C. D. Bailyn. The optical counterpart of the bright nearby millisecond pulsar PSR J0437-4715. *ApJ*, 411:L83–L85, July 1993.
- Erin Balsamo, Keith C. Gendreau, Zaven Arzoumanian, et al. Development of full shell foil x-ray mirrors. In *Proc. SPIE*, volume 8450, pages 845052–845052–19, 2012.
- Erin Balsamo, Keith Gendreau, Takashi Okajima, et al. Shrink tape technique for heat-forming aluminum substrates for thin foil x-ray mirrors and the Neutron Star Interior Composition Explorer x-ray concentrators. *Journal of Astronomical Telescopes, Instruments, and Systems*, 2(1):015001, jan 2016.
- A. Baykal. Preface. In A. Baykal, S.K. Yerli, S.C. Inam, and S. Grebenev, editors, *The Electromagnetic Spectrum of Neutron Stars*, volume 210 of *NATO Science Series II: Mathematics, Physics and Chemistry*. Springer, 2005.
- W. Becker and J. Trümper. Detection of pulsed X-rays from the binary millisecond pulsar J0437 - 4715. *Nature*, 365:528–530, October 1993.
- A. M. Beloborodov. Gravitational Bending of Light Near Compact Objects. *ApJ*, 566:L85–L88, February 2002.
- D. Bhattacharya. The evolution of the magnetic fields of neutron stars. *Journal of Astrophysics and Astronomy*, 16(2):217–232, 1995.

- J. K. Blackburn. FTOOLS: A FITS Data Processing and Analysis Software Package. In R. A. Shaw, H. E. Payne, and J. J. E. Hayes, editors, *Astronomical Data Analysis Software and Systems IV*, volume 77 of *Astronomical Society of the Pacific Conference Series*, page 367, 1995.
- S. Bogdanov. The Nearest Millisecond Pulsar Revisited with XMM-Newton: Improved Mass-radius Constraints for PSR J0437-4715. *ApJ*, 762:96, January 2013.
- S. Bogdanov, G. B. Rybicki, and J. E. Grindlay. Constraints on Neutron Star Properties from X-Ray Observations of Millisecond Pulsars. *ApJ*, 670:668–676, November 2007.
- Slavko Bogdanov, Jonathan E. Grindlay, and George B. Rybicki. X-Rays from Radio Millisecond Pulsars: Comptonized Thermal Radiation. *The Astrophysical Journal Letters*, 648(1):L55, 2006.
- D. Brown. NASA Selects Science Investigations For Concept Studies. Online, September 2011. URL https://www.nasa.gov/home/hqnews/2011/sep/HQ_11-328_Science_Proposals.html.
- Thomas Robert Burnight. Soft x-radiation in the upper atmosphere. *Phys. Rev*, 76 (165):19–9, 1949.
- Joan Centrella, Francis Reddy, and Pat Tyler. Goddard’s astrophysics science division annual report 2011. Technical report, NASA, 2012. URL http://science.gsfc.nasa.gov/astrophysics/content/uploadFiles/highlight_files/ASDAnnualReport2010.pdf.
- Chandra X-ray Center. The Chandra Proposers’ Observatory Guide, Version 19.0. Online, December 15 2016. URL <http://asc.harvard.edu/proposer/POG/pdf/MPOG.pdf>. Accessed: 2017-04-13.
- S. Chandrasekhar. The Maximum Mass of Ideal White Dwarfs. *ApJ*, 74:81, July 1931.
- K. Cummings, P. Laws, E. Redish, and P. Cooney. *Understanding Physics*. John Wiley & Son, Inc., Hoboken, NJ, 2004. ISBN 0471370991.
- Stephen A. Drake. How Many Known X-Ray Sources Are There? Online, June 2014. URL http://heasarc.gsfc.nasa.gov/docs/heasarc/headates/how_many_xray.html. Accessed: 2017-03-28.
- Dunstone Company Inc. Product shrink tapes, 2013. URL <http://www.shrinktape.com/products/hi-shrink-tape/hi-shrink-tape.aspx>. Accessed: 2017-03-28.
- Dupont. Kapton®polyimide film. Online, 2017. URL <http://www.dupont.com/products-and-services/membranes-films/polyimide-films/brands/kapton-polyimide-film.html/>. Accessed: 2017-03-29.

- G. W. Fraser. *X-ray detectors in astronomy*. Cambridge University Press, The Edinburgh Building, Cambridge CB 8RU, UK, 1989. ISBN 052132663.
- P. C. Freire, D. Nice, and J. and others Lattimer. Constraining the Bulk Properties of Dense Matter by Measuring Millisecond Pulsar Masses - A White Paper for the Astronomy and Astrophysics Decadal Survey, CFP Panel. *ArXiv e-prints*, February 2009. URL <http://adsabs.harvard.edu/abs/2009arXiv0902.2891F>.
- H. Friedman, S. W. Lichtman, and E. T. Byram. Photon counter measurements of solar x-rays and extreme ultraviolet light. *Phys. Rev.*, 83:1025–1030, Sep 1951. doi: 10.1103/PhysRev.83.1025. URL <http://link.aps.org/doi/10.1103/PhysRev.83.1025>.
- K. Gendreau. NICER Science Overview. Online, Feb 2016. URL https://heasarc.gsfc.nasa.gov/docs/nicer/slides/NICER_Science_Overview.pdf. Accessed: 2017-03-28.
- K. C. Gendreau. NICER: An Explorer Mission of Opportunity Concept Study Report. Technical report, NASA Goddard Space Flight Center, September 2012. Proposal submitted to NASA Program Announcement Number: NNH08ZDA009O.
- K. C. Gendreau, Z. Arzoumanian, and T. Okajima. The Neutron star Interior Composition ExploreR (NICER): an Explorer mission of opportunity for soft x-ray timing spectroscopy. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 8443, pages 844313–844313–8, 2012. doi: 10.1117/12.926396. URL <http://dx.doi.org/10.1117/12.926396>.
- Keith Gendreau, Zaven Arzoumanian, Fumi Asami, et al. The x-ray advanced concepts testbed (XACT) sounding rocket payload. *Proc. SPIE*, 8443:84434V, September 2012.
- Gendreau, K.C. and Arzoumanian, Z. and Adkins, P.W. and others. The Neutron star Interior Composition Explorer (NICER): design and development, 2016.
- Ghosh, Pranab. *Rotation and Accretion Powered Pulsars*. World Scientific Publishing Co. Pte. Ltd., 5 Toh Tuck Link, Singapore 596224, 2007. ISBN 9810247443.
- R. Giacconi, W. P. Reidy, G. S. Vaiana, et al. Grazing-incidence telescopes for X-ray astronomy. *Space Science Reviews*, 9:3–57, 1969. ISSN 0038-6308.
- R. Giacconi, G. Branduardi, U. Briel, et al. The Einstein /HEAO 2/ X-ray Observatory. *ApJ*, 230:540–550, June 1979.
- Riccardo Giacconi, Herbert Gursky, Frank R Paolini, and Bruno B Rossi. Evidence for x rays from sources outside the solar system. *Physical Review Letters*, 9(11): 439, 1962.

- Goddard Geophysical and Astronomical Observatory. GGAO home page. Online, jul 2014. URL <https://cddis.nasa.gov/ggao/>. Accessed: 2017-03-22.
- P. Gorenstein. Grazing incidence telescopes for x-ray astronomy. *Optical Engineering*, 51(1):011010, January 2012.
- A. R. Green, I. M. McHardy, and H. J. Lehto. On the nature of rapid X-ray variability in active galactic nuclei. *MNRAS*, 265:664, December 1993.
- D. Grupe, H.-C. Thomas, and K. Beuermann. X-ray variability in a complete sample of Soft X-ray selected AGN. *A&A*, 367:470–486, February 2001.
- S. Guillot, V. M. Kaspi, R. F. Archibald, et al. The NuSTAR view of the non-thermal emission from PSR J0437-4715. *MNRAS*, 463:2612–2622, December 2016.
- Christian Bernt Haakonsen, Monica L. Turner, Nick A. Tacik, and Robert E. Rutledge. The mcgill planar hydrogen atmosphere code (mcphac). *ApJ*, 749(1):52, 2012.
- J. P. Halpern and H. L. Marshall. A Long EUVE Observation of the Seyfert Galaxy RX J0437.4-4711. *ApJ*, 464:760, June 1996.
- J. P. Halpern, C. Martin, and H. L. Marshall. Soft X-Ray Properties of the Binary Millisecond Pulsar J0437-4715. *ApJ*, 462:908, May 1996.
- J. P. Halpern, K. M. Leighly, and H. L. Marshall. An Extreme Ultraviolet Explorer Atlas of Seyfert Galaxy Light Curves: Search for Periodicity. *ApJ*, 585:665–676, March 2003.
- A. K. Harding and A. G. Muslimov. Pulsar x-ray and gamma-ray pulse profiles: Constraint on obliquity and observer angles. *The Astrophysical Journal*, 500(2): 862, 1998.
- J. D. Harrington. NASA Selects Explorer Investigations for Formulation. Online, April 2013. URL https://www.nasa.gov/home/hqnews/2013/apr/HQ_13-088_Astro_Explorer_Mission_.html.
- James B. Hartle. *Gravity: an introduction to Einstein's general relativity*. Pearson Education Inc., Addison Wesley, 1301 Sansome St., San Francisco, CA 94111, 2003. ISBN 0805386629.
- C. Heiles and M. N. Cleary. The HI column density as a function of position in the southern sky. *Australian Journal of Physics Astrophysical Supplement*, 47:1–58, September 1979.
- B. L. Henke, E. M. Gullikson, and J. C. Davis. X-Ray Interactions: Photoabsorption, Scattering, Transmission, and Reflection at $E = 50\text{--}30,000$ eV, $Z = 1\text{--}92$. *Atomic Data and Nuclear Data Tables*, 54:181–342, July 1993.

- Junko Hiraga, Hiroshi Tsunemi, and Emi Miyata. Subpixel spatial resolution of the x-ray charge-coupled device based on the charge cloud shape. *Japanese Journal of Applied Physics*, 40(3R):1493, 2001.
- S.B. Howell. *Handbook of CCD Astronomy*. Cambridge Observing Handbooks for Research Astronomers. Cambridge University Press, 2006. ISBN 9781139450768.
- Irby B. K. FTOOLS: A General Package of Software to Manipulate FITS Files. Online, 2017. URL <http://heasarc.gsfc.nasa.gov/ftools>. Accessed: 2017-04-26.
- J.D. Jackson. *Classical Electrodynamics*. John Wiley & Sons, 1998. ISBN 9780471309321.
- F. Jansen, D. Lumb, B. Altieri, et al. XMM-Newton observatory. I. The spacecraft and operations. *A&A*, 365:L1–L6, January 2001.
- J. Jason Budinoff, K. Gendreau, Z. Arzoumanian, et al. Design Development of a Combined Deployment and Pointing System for the International Space Station Neutron Star Interior Composition Explorer Telescope. In *Proceedings of the 43rd Aerospace Mechanisms Symposium*, 2016.
- JAXA. Communication anomaly of X-ray Astronomy Satellite “Hitomi” (ASTRO-H). Online, 2016. URL http://global.jaxa.jp/press/2016/03/20160327_hitomi.html. Accessed: 2017-03-28.
- Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL <http://www.scipy.org/>. [Online; accessed 2017-03-08].
- W. Joye and E. Mandel. New Features of SAOImage DS9. In *Astronomical Data Analysis Software and Systems XII Astronomical Society of the Pacific Conference Series*, volume 294, page 489, 2003.
- Steven Kenyon. personal communication, December 2013.
- Jason E. Koglin, HongJun An, Kenneth L. Blaedel, et al. NuSTAR hard x-ray optics design and performance. In *Proc. SPIE*, volume 7437, pages 74370C–1, 2009.
- Katsuji Koyama, Hiroshi Tsunemi, Tadayasu Dotani, et al. X-ray imaging spectrometer (xis) on board suzaku. *Publications of the Astronomical Society of Japan*, 59(sp1):S23–S33, 2007.
- D. Lai, M. van Kerkwijk, J. Arons, et al. Extreme Astrophysics with Neutron Stars. In *astro2010: The Astronomy and Astrophysics Decadal Survey*, volume 2010 of *Astronomy*, 2009. URL <http://adsabs.harvard.edu/abs/2009astro2010S.170L>.

- J. M. Lattimer and M. Prakash. Neutron star structure and the equation of state. *The Astrophysical Journal*, 550(1):426, 2001. URL <http://stacks.iop.org/0004-637X/550/i=1/a=426>.
- J.E. Manson. X-Ray calibration system, March 19 1985. US Patent 4,506,375.
- K. Mitsuda et al. The X-Ray Observatory Suzaku. *PASJ*, 59:1–7, January 2007.
- K. Mukai. PIMMS and Viewing: proposal preparation tools. *Legacy*, 3:21–31, May 1993.
- Mukai, K. WebPIMMS: A Mission Count Rate Simulator Powered by PIMMS v4.8d. Online, January 16 2017. URL <https://heasarc.gsfc.nasa.gov/cgi-bin/Tools/w3pimms/w3pimms.pl>. Accessed: 2017-04-26.
- NASA. NASA’s HEASARC: Observatories. Online, September 25 2013. URL <https://heasarc.gsfc.nasa.gov/docs/observatories.html>. Accessed: 2017-04-23.
- NASA HEASARC Database Group. Access to the catalogs and astronomical archives of the HEASARC. Online, December 19 2016. URL <https://heasarc.gsfc.nasa.gov/docs/archive.html>. Accessed: 2017-04-26.
- NASA, NICER. The Neutron star Interior Composition ExploreR Mission. Online, March 21 2017. URL <https://heasarc.gsfc.nasa.gov/docs/nicer/>. Accessed: 2017-04-11.
- Phil Newman. X-ray Astronomy. Online, November 2014. URL http://imagine.gsfc.nasa.gov/docs/science/known_12/history_xray.html. Accessed: 2017-03-28.
- F. Özel. Surface emission from neutron stars and implications for the physics of their interiors. *Reports on Progress in Physics*, 76(1), January 2013.
- F. Paerels, M. Méndez, M. Agueros, et al. The Behavior Of Matter Under Extreme Conditions. In *astro2010: The Astronomy and Astrophysics Decadal Survey*, volume 2010 of *Astronomy*, 2009. URL <http://adsabs.harvard.edu/abs/2009astro2010S.230P>.
- G. G. Pavlov and V. E. Zavlin. Mass-to-Radius Ratio for the Millisecond Pulsar J0437-4715. *ApJ*, 490:L91, November 1997.
- R. Petre. Thin Shell, Segmented X-Ray Mirrors. *X-Ray Optics and Instrumentation, 2010. Special Issue on X-Ray Focusing: Techniques and Applications*, id.412323, 2010:412323, 2010.
- J. Poutanen and M. Gierliński. On the nature of the X-ray emission from the accreting millisecond pulsar SAX J1808.4-3658. *MNRAS*, 343:1301–1311, August 2003.

- G. Prigozhin, K. Gendreau, R. Foster, et al. Characterization of the silicon drift detector for NICER instrument. In *High Energy, Optical, and Infrared Detectors for Astronomy V*, volume 8453 of Proc. SPIE, July 2012.
- Princeton Instruments. WinView 32 Software User’s Manual. Online, 2013. URL <ftp://ftp.piacton.com/Public/Manuals/Princeton%20Instruments/WinView%202.6%20Imaging%20Software%20User%20Manual.pdf>. Accessed: 2017-04-06.
- Princeton Instruments. Princeton Instruments PIXIS-XO 2048B X-ray CCD Datasheet. Online, 2017. URL http://www.infim.ro/sites/default/files/u300/Princeton_Instruments_PIXIS_XO_2048B_RevB0.pdf. Accessed: 2017-04-06.
- Dimitrios Psaltis, Feryal zel, and Deepto Chakrabarty. Prospects for measuring neutron-star masses and radii with x-ray pulse profile modeling. *The Astrophysical Journal*, 787(2):136, 2014.
- S. Romaine, R. Bruni, B. Choi, et al. Development of light weight replicated x-ray optics. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 8861 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 0, October 2013. doi: 10.1117/12.2024897.
- Roper Scientific. VersArray Imaging Cameras. Online, 2017. URL <http://www.roperscientific.de/versarray.html>. Accessed: 2017-04-06.
- Sabol, E. J. X-ray Background Tool. Online, 2017. URL <https://heasarc.gsfc.nasa.gov/cgi-bin/Tools/xraybg/xraybg.pl>. Accessed: 2017-04-26.
- A. Savitzky and M. J. E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36:1627–1639, 1964.
- P. J. Serlemitsos. Conical Foil X-Ray Mirrors: Performance And Projections. In S. Bowyer and J. C. Green, editors, *Grazing incidence optics for astronomical and laboratory applications*, volume 830 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 179–184, August 1988. doi: 10.1117/12.942179.
- P. J. Serlemitsos and Y. Soong. Foil X-ray Mirrors. *Ap&SS*, 239:177–196, September 1996.
- P. J. Serlemitsos, L. Jalota, Y. Soong, et al. The X-ray telescope on board ASCA. *PASJ*, 47:105–114, February 1995.
- P. J. Serlemitsos, Y. Soong, T. Okajima, and D. J. Hahne. Foil x-ray mirrors for astronomical observations: still an evolving technology. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 7732 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 0, July 2010. doi: 10.1117/12.857355.

- Serlemitsos, P. J. Conical Foil X-ray Mirrors: Performance and Projections. *Applied Optics*, 27(8):1447–1452, 1988.
- S. L. Snowden, R. Egger, M. J. Freyberg, et al. ROSAT Survey Diffuse X-Ray Background Maps. II. *ApJ*, 485:125–135, August 1997.
- Y. Soong, L. Jalota, and P. J. Serlemitsos. Conical thin foil x-ray mirror fabrication via surface replication. In R. B. Hoover and A. B. Walker, editors, *X-Ray and Extreme Ultraviolet Optics*, volume 2515 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 64–69, June 1995.
- Y. Soong, T. Okajima, P. J. Serlemitsos, S. L. Odell, B. D. Ramsey, M. V. Gubarev, M. Ishida, Y. Maeda, R. Iizuka, T. Hayashi, Y. Tawara, A. Furuzawa, H. Mori, T. Miyazawa, H. Kunieda, H. Awaki, S. Sugita, K. Tamura, K. Ishibashi, T. Izumiya, S. Minami, T. Sato, K. Tomikawa, N. Kikuchi, and T. Iwase. ASTRO-H Soft X-ray Telescope (SXT). In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 9144 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 28, July 2014. doi: 10.1117/12.2056804.
- E. Spiller. *Soft X-Ray Optics*. SPIE Press monograph. SPIE Optical Engineering Press, 1994. ISBN 9780819416544.
- J.H. Swank. The rossi x-ray timing explorer. *Nuclear Physics B - Proceedings Supplements*, 69(13):12 – 19, 1999. The Active X-ray Sky Results from BeppoSAX and {RXTE}.
- H. Tsunemi, J. Hiraga, K. Yoshita, and K. Hayashida. Measurement of the charge cloud shape produced by an x-ray photon inside the {CCD} using a mesh experiment. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 421(12):90 – 98, 1999.
- Y. Tsusaka, H. Suzuki, K Yamashita, et al. Characterization of the Advanced Satellite for Cosmology and Astrophysics x-ray telescope: preflight calibration and ray tracing. *Appl. Opt.*, 34(22):4848–4856, Aug 1995.
- M. van der Klis. Timing Neutron Stars and the link with black-hole timing. In A. Baykal, S.K. Yerli, S.C. Inam, and S. Grebenev, editors, *The Electromagnetic Spectrum of Neutron Stars*, volume 210 of *NATO Science Series II: Mathematics, Physics and Chemistry*, pages 283–310. Springer, 2005.
- J.P.W. Verbiest, M. Bailes, W. van Straten, et al. Precision Timing of PSR J04374715: An Accurate Pulsar Distance, a High Pulsar Mass, and a Limit on the Variation of Newton’s Gravitational Constant. *The Astrophysical Journal*, 679(1):675, 2008. URL <http://stacks.iop.org/0004-637X/679/i=1/a=675>.

- K. Viironen and J. Poutanen. Light curves and polarization of accretion- and nuclear-powered millisecond pulsars. *A&A*, 426:985–997, November 2004.
- W. Voges, B. Aschenbach, Th. Boller, et al. The rosat all-sky survey bright source catalogue. *A&A*, 349:389, 1999.
- T. Wang, C. Otani, M. Matsuoka, et al. ASCA and ROSAT observations of the Seyfert 1 galaxy RX J0437.4-4711. *MNRAS*, 293:397, February 1998.
- H. Wolter. Optical image-forming mirror system providing for grazing incidence of rays, August 14 1956. US Patent 2,759,106.
- V. E. Zavlin. XMM-Newton Observations of Four Millisecond Pulsars. *ApJ*, 638: 951–962, February 2006.
- V. E. Zavlin. Theory of Radiative Transfer in Neutron Star Atmospheres and Its Applications. In W. Becker, editor, *Astrophysics and Space Science Library*, volume 357 of *Astrophysics and Space Science Library*, page 181, 2009.
- V. E. Zavlin and G. G. Pavlov. Soft X-rays from polar caps of the millisecond pulsar J0437-4715. *A&A*, 329:583–598, January 1998.
- V. E. Zavlin, G. G. Pavlov, and Y. A. Shibano. Model neutron star atmospheres with low magnetic fields. I. Atmospheres in radiative equilibrium. *A&A*, 315: 141–152, November 1996.
- V. E. Zavlin, G. G. Pavlov, D. Sanwal, et al. X-ray Radiation from the Millisecond Pulsar J0437-4715. *ApJ*, 569:894–902, April 2002.

