

「すざく」
ファーストステップガイド
第4.0.2版 (Process Version 2.1–2.3)

「すざく」ヘルプデスク

平成20年11月17日

はじめに

第一版：このマニュアルは、ASCA時代の名著「はじめての表街道」に相当するものを意図して作りました。対象は、「すざく」の解析を始めようと思いついた新人と、最近解析から遠ざかっており、記憶があやふやになってきた人々です。誰にとっても、最初の一步は、たいへんハードルが高いものです。ましてや、近くに気軽に相談できる やさしい先輩がいない場合、どうすればよいか途方にくれることでしょう。一人でも多くの学生や新人が、このマニュアルを読み、「すざく」のデータに慣れ親しんで、すばらしい成果を世界に発信してくれることを祈ります。解析でわからない事があれば、公式な「すざく」ヘルプデスクが日本に立ち上がっていますので、<http://cosmic.riken.jp/suzaku/help/>にアクセスしてください。WWWページにある質問テンプレートに質問を記載のうえ、suzaku_help@crab.riken.jpに質問を投げるだけです。

2006年6月9日

第二版：SWG向けであったマニュアルを元に、Guest Observer向けのマニュアルを作りました。version 1.1,1.2 products用です。多くの方にとって「すざく」データ解析が、より敷居の低いものになることを望みます。

2006年8月22日

第三版：ver 1.x products用のマニュアルを改訂リリースします。「すざく」ヘルプデスクも、皆様のお陰で、一周年を迎えることができました。ヘルプデスクも併せてご利用ください。(2007年8月3日 小改訂しました)

2007年4月23日

すざくヘルプデスクー同

執筆メンバー

磯部直樹、浦田裕次、江副祐一郎、太田直美、川原田 円、久保田あや、鈴木素子、千田篤史、玉川 徹、寺田幸功、中島基樹、馬場 彩、平賀純子(あいうえお順・敬称略)

お願い

マニュアルの改良には、皆さんからのフィードバックが欠かせません。もしこのマニュアルが役に立ったと感じた人は、なんらかのフィードバックを「すざくヘルプデスク」(suzaku_help@crab.riken.jp)に送ってください。そのフィードバックが次のバージョンへの励みになります。このようなマニュアルは、高い公心心を持った人々により維持されています。もしあなたがこのマニュアルで巣立ち、後に続く人にも同じような環境を提供し続けたいと考えているのなら、すぐにマニュアルのメンテナーに参加してください！

目次

第 1 章 「すざく」衛星 (<i>Suzaku</i> SATELLITE)	6
第 2 章 搭載されている検出器 (SCIENTIFIC INSTRUMENTS)	8
2.1 概要 (Overview)	8
2.2 XRT	10
2.3 XIS	13
2.4 HXD	15
第 3 章 すざく衛星データ概要と解析ルートマップ (SUMMARY AND ROUTE MAP OF ANALYSES)	18
3.1 FITS 形式って? (FITS Data Format)	18
3.2 解析に必要な衛星データ (Data Types)	19
3.3 観測データの較正情報の最新化 (Overview of Processing)	20
3.4 標準的な解析 (Route Map of Your Analysis)	21
3.4.1 イメージを描きたい (Image analysis)	21
3.4.2 X 線スペクトルをみたい (Spectral analysis)	21
3.4.3 ライトカーブをみたい (Check light curve, and timing analysis)	22
3.4.4 より詳細な解析へのステップアップ (Advanced analysis)	22
第 4 章 とりあえずやってみよう (WALK THROUGH OF <i>Suzaku</i> ANALYSIS)	23
4.1 解析の前に真っ先にすべきこと:情報収集 (Getting Information)	23
4.2 データの取得 (Getting <i>Suzaku</i> Data)	23
4.3 XIS の解析 (XIS Analysis)	25
4.3.1 XIS 解析の準備 (Preparing for the XIS analysis)	25
4.3.2 cleaned event からの QL (QL with cleaned event)	25
4.3.3 XIS データに最新 CALDB をかけなおす	27
4.4 HXD の解析 (HXD Analysis)	28
4.4.1 データの取得 (Gathering datasets)	28
4.4.2 HXD データの再プロセス (GSO を解析したい場合 or PIN の一部のデータ)	31
4.4.3 GTI のマージ (Merge GTI)	34
4.4.4 スペクトルの抽出 (Extract spectral fits)	34
4.4.5 デッドタイム補正 (Dead time correction)	37
4.4.6 PIN バックグラウンドの積分時間補正 (Scale down of PIN BGD flux)	39
4.4.7 いざスペクトル解析! (Let's enjoy spectral analyses!)	39
4.5 より詳細なデータスクリーニングをしたい時 (Further Analysis)	40
4.5.1 (XIS/HXD) SAA, ELV, COR など hk 情報での screening (Screening with HK data (SAA, ELV, COR etc.))	40
4.5.2 (XIS/HXD) telemetry saturation の screening (Screenings with telemetry saturation time)	41
4.5.3 (XIS) cleansis による screening (Screenings with cleansis command)	42

4.5.4	(HXD) 地食を用いた NXB model のチェック (Check NXB model with earth occulted data)	42
第 5 章	XSELECT	43
5.1	初期設定 (Initial Settings)	43
5.1.1	XSELECT の立ち上げ (Start XSELECT)	43
5.1.2	event file の読み込み (Read event files)	44
5.1.3	XSELECT で使用する変数 (Variables in XSELECT)	45
5.2	ライトカーブの作成 (Generate Light Curves)	46
5.2.1	ライトカーブの抽出 (Extract light curves)	46
5.2.2	エネルギーバンドの指定 (Filter energy bands)	46
5.2.3	タイムリージョンの指定 (Filter time regions)	47
5.2.4	作成したライトカーブの保存 (Save light curves)	48
5.3	XIS イメージの作成 (Generate XIS Images)	48
5.3.1	タイムリージョン・エネルギーバンドの指定 (Filter time regions/energy bands)	48
5.3.2	イメージの抽出・保存 (Extract/save images)	49
5.4	スペクトルの作成 (Generate Spectra)	49
5.4.1	スペクトル取得領域の指定 (XIS) (Regions in extracting spectra)	49
5.4.2	スペクトルの抽出・保存 (Extract/save spectra)	50
5.5	その他便利な解析方法 (Tips)	50
5.5.1	event file の作成 (Extract filtered event files)	50
5.5.2	ライトカーブ・イメージ・スペクトルの同時作成 (Extract light curves, images, and spectra)	51
5.5.3	シェルコマンドの使用 (Using shell commands)	51
第 6 章	ds9	52
6.1	ds9 で出来ること (What can ds9 do ?)	52
6.2	ds9 の window (Window of ds9)	52
6.3	ds9 一通りの使い方 (how to use ds9)	53
6.3.1	ds9 を立ち上げる (Start up ds9)	53
6.3.2	イメージを自分好みに調整する (Adjusting images as you like)	53
6.3.3	スペクトル作成用の region を作成する (Make regions for spectral analysis)	56
6.3.4	画像を保存・印刷する (Saving and printing images)	57
6.3.5	ds9 の終了 (Shut down of ds9)	57
第 7 章	TIMING ANALYSIS; XRONOS	58
7.1	Overview of XRONOS	58
7.2	Barycentric Correction for Suzaku	58
7.3	Make Light Curve	60
7.4	Make Power Spectrum	61
7.5	Perform Periodic Search	63
7.6	Make Folded Light Curve	65
7.7	Additional Comments	66
第 8 章	XSPEC によるスペクトルフィッティング (XSPEC)	67
8.1	grppha でピンまとめ (Grouping of PI Bins with grppha)	67
8.1.1	方針 (Policy)	67
8.1.2	基本的な実行例 (Simple examples)	67

8.1.3	PHA channel を指定してピンまとめする方法 (How to group PHA channels by your favorite assignments)	68
8.1.4	systematic error を入れる (How to include systematic errors)	68
8.2	スペクトルを足す方法 (How to Add Spectra)	69
8.2.1	mathpha でスペクトルを足す (Add spectra via mathpha)	69
8.2.2	レスポンスの足し算 (Add response matrix files)	70
8.3	xspec でフィッティング (Fittings in XSPEC)	70
8.3.1	設定 (Settings)	70
8.3.2	xspec を走らせてみる (execute XSPEC)	70
8.3.3	data の読み込み (Read data)	71
8.3.4	スペクトルを表示 (Display spectra)	71
8.3.5	モデルフィット (Fittings with models)	71
8.3.6	フラックスを求める (Determine the flux)	73
8.3.7	検出器のエネルギー領域を超えてフラックスを求める (Determine the flux in out of energy ranges of a detector response)	73
8.3.8	モデルの追加と削除 (Add or Delete Models)	73
8.3.9	画面表示 etc. (Display etc.)	74
8.3.10	作業ファイルを保存 (Save files)	74
8.3.11	QDP でスペクトルをハードコピー (How to save hard copies via QDP)	75
8.3.12	その他 (Others)	75
第 9 章	便利な FTOOLS (USEFUL FTOOLS)	79
9.1	fhelpt	79
9.2	addarf / addrmf	79
9.3	fdump	79
9.4	mathpha / grppha	79
9.5	QDP のコマンド概要 (Overview of QDP commands)	80
9.5.1	ファイルの構成 (Files)	80
9.5.2	起動と終了 (Start and End)	80
9.5.3	全体 (Overview)	81
9.5.4	色 (Colors)	81
9.5.5	シンボル (Symbols)	81
9.5.6	ライン (Lines)	81
9.5.7	ラベル/タイトル (Labels/Titles)	81
9.5.8	ファイル出力関連 (File input/output)	82
第 10 章	「すざく」に固有な FTOOLS (Suzaku FTOOLS)	83
10.1	xisrmfgen	83
10.2	xissimarfgen	84
10.3	xiscontamicalc	86
10.4	xissim を用いた exposure map 作成	87
10.5	hxdarfgen	88
付録 A	便利な Web page (USEFUL WEB PAGES)	89

付録 B ソフトウェアのインストール・設定 (SOFTWARE INSTALLATION)	91
B.1 HEAsoft	91
B.1.1 バイナリディストリビューション (Binary distribution)	91
B.1.2 ソースコードディストリビューション (Source code distribution)	93
B.1.3 インストール後の設定 (Setup after installation)	94
B.1.4 ”ハマりがち”ばたーん (A common trap)	95
B.2 ds9	95
付録 C <i>Suzaku</i> HelpDesk	97
付録 D GLOSSARY	99

第1章 「すざく」衛星 (Suzaku SATELLITE)

「すざく」(朱雀; Astro-E2)は、ISAS/JAXAのM-V-6号ロケットにより、2005年7月10日に打ち上げられた、我が国5番目のX線天文衛星である。衛星は直径2.1mの八角柱の構体を基本とし、全長6.5m(軌道上で鏡筒伸展後)の大きさを持つ。太陽パネルを広げた幅は、5.4mになる。衛星の重量は1680kgにもなり、日本の科学衛星としては、これまでにない大型衛星である。姿勢は、太陽電池パネルが太陽から30度以内の方向に常に向くように三軸制御される。科学観測機器は太陽電池パネルの軸に垂直に向けられるので、観測できる範囲は太陽から60~120度の角度範囲に限定されることになる。近地点高度250km、遠地点高度550km、軌道傾斜角31度の楕円軌道に投入され、その後、搭載二次推進系により、高度約570kmの略円軌道へ最終投入された。「すざく」は1日に地球を15周するが、鹿児島県内之浦の地上局から衛星と通信できるのは、その内の5回のみである。そのため追跡オペレーションは、1日5回、約10分ずつ行われる。

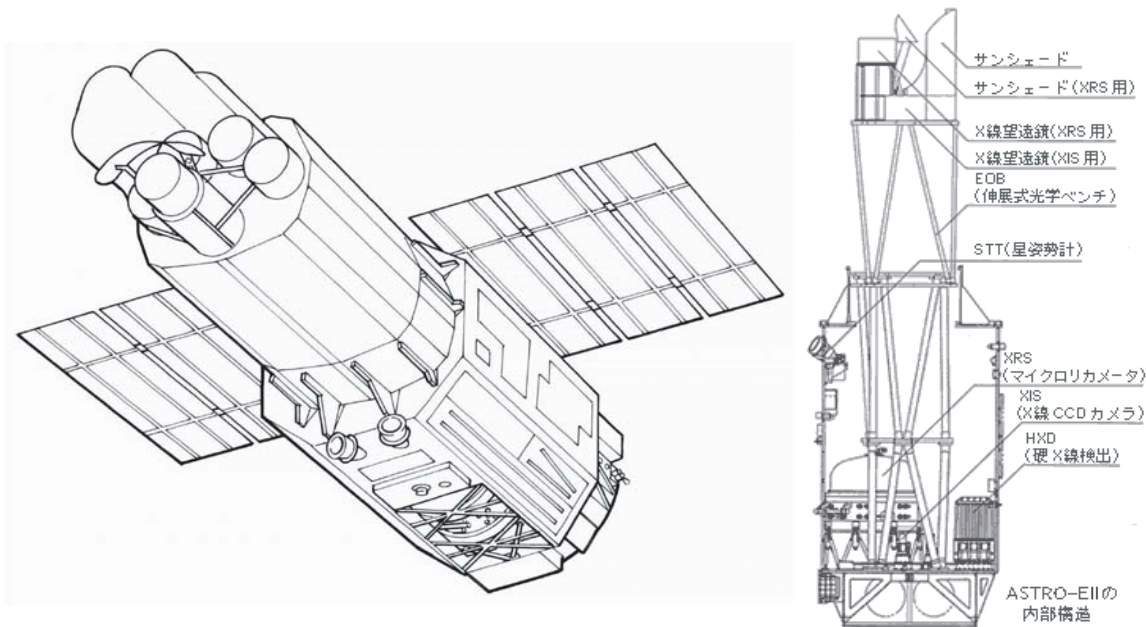


図 1.1: 「すざく」の概観

「すざく」衛星は、2000年2月に打ち上げロケットの不具合によって軌道投入できなかった Astro-E 衛星の再挑戦をかけたミッションで、「はくちょう(1979年)」「てんま(1983年)」「ぎんが(1987年)」「あすか(1993年)」に続く X 線天文衛星(図 1.2)として、日本国内の大学、諸機関、米国航空宇宙局(NASA)などの協力で開発が進められてきた。さまざまな X 線天体に対して、これまでになかった広いエネルギー領域(0.3-600keV)にわたり、より高いエネルギー分解能かつ高感度での観測ができることが、最大の特徴である。

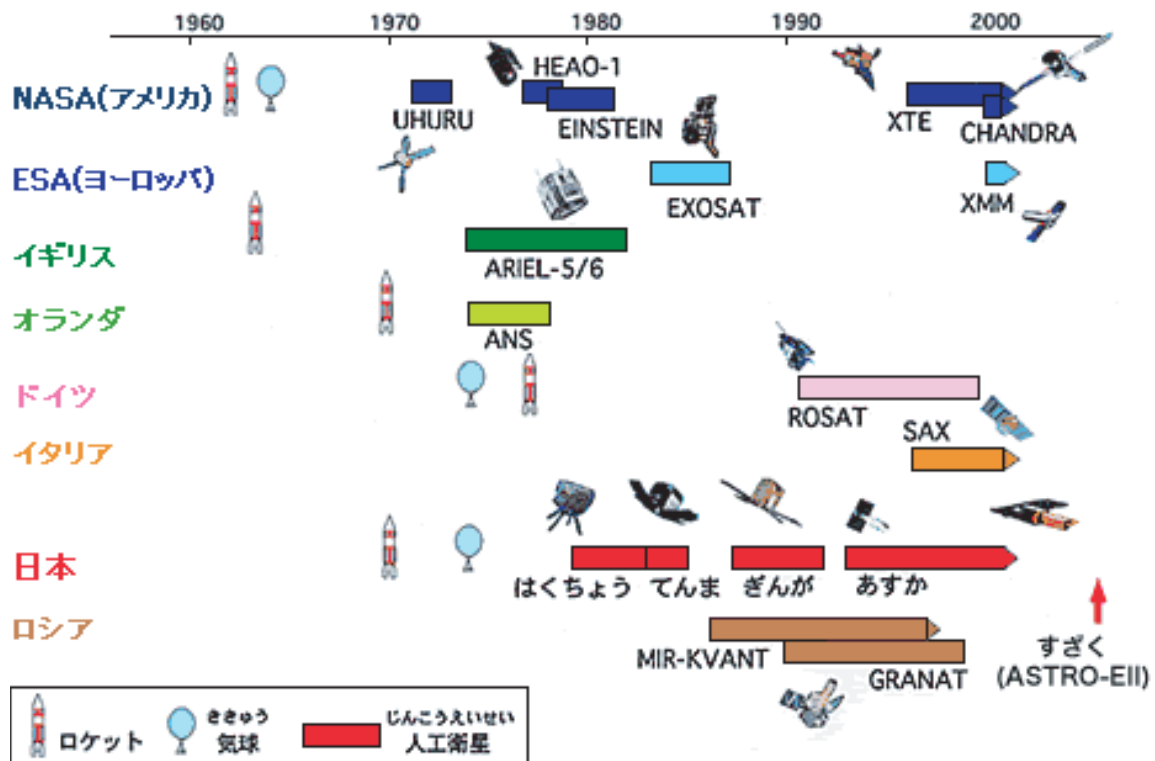


図 1.2: X 線天文学の歴史と日本の X 線天文衛星

「すざく」に関する最新の情報などは、以下の web サイトを参考のこと。

- 「すざく」プロジェクトのサイト
<http://www.astro.isas.jaxa.jp/suzaku/>
- 理化学研究所の「すざく」ヘルプデスクのサイト
http://cosmic.riken.jp/suzaku/help/index_j.html

第2章 搭載されている検出器 (SCIENTIFIC INSTRUMENTS)

天体のデータ解析を始める前に、観測機器についてよく知っておかないと、間違った結果を導くことになりかねず危険である。この章では、すざく衛星に搭載されている検出器の紹介と、その性能についてまとめる。

2.1 概要 (Overview)

すざく衛星には5つの軟X線検出器と1つの硬X線検出器が搭載されている。軟X線望遠鏡は、5つのX線反射鏡 (X-ray telescope; XRT) と5つの焦点面検出器 (4つのXIS検出器と1つのXRS検出器) からなる。XIS (X-ray Imaging Spectrometer) はX線 CCD カメラで、0.2–12 keV のエネルギー帯域をカバーし、典型的なエネルギー分解能は130 eV である。XRS (X-ray Spectrometer) はX線マイクロカロリメータで、エネルギー帯域はXISと同程度、典型的なエネルギー分解能は6 eV である。残念なことに、2005年8月8日、XRSで使用している液体ヘリウムが消失するという事故が発生し、XRSによる観測は不可能になった。さらに高いエネルギー (10–700 keV) のX線を観測するために開発されたのが硬X線検出器 (Hard X-ray Detector; HXD) である。「すざく」ではXIS4台とHXDで同じ天体を同時に観測することができ、広いエネルギー帯で高感度のX線分光が可能である。特に、(1) 硬X線領域 (10–300keV) においてこれまでで最高の感度、(2) 軟X線領域 (0.3–1 keV) で、これまでの CCD カメラに比べて高い感度と分解能、を実現している。各検出器の特徴や性能を表 2.1 にまとめる。衛星上での検出器の配置は図 2.1 に示した。

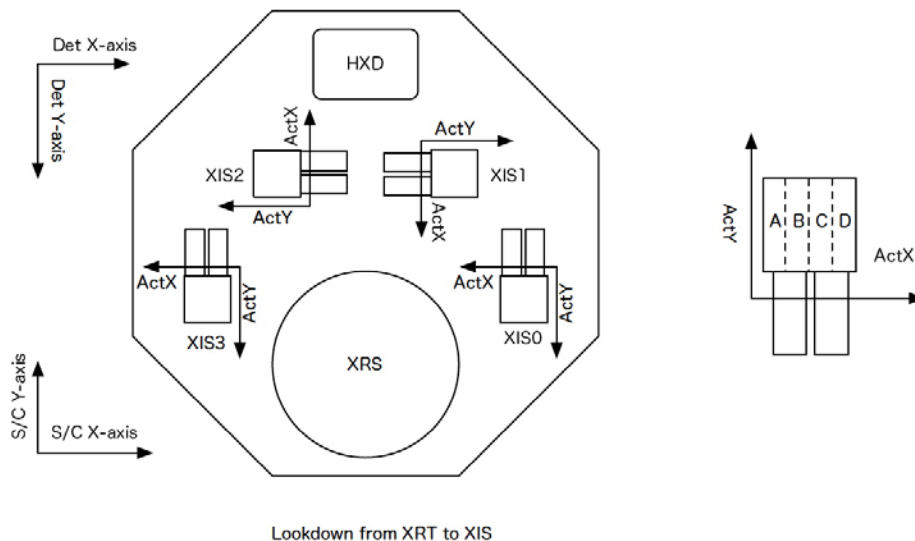


図 2.1: 衛星上での各検出器の配置 (Astro-E2 実験報告書より)。なお、「すざく」ではいくつかの座標系が用いられる。図中の S/C X, S/C Y は衛星座標系、DETX, DETY は検出器座標系である。この他に、観測時の衛星姿勢を考慮して、赤道座標系に投影したのが Sky 座標系 (X, Y) である。Sky 座標系を用いると、異なる検出器で得られたイメージを足し合わせる事ができる。

表 2.1: 「すざく」に搭載されている観測機器の概要

XRT	焦点距離	4.75 m
	視野 (FWHM)	17'@1.5 keV, 13'@8 keV
	Plate scale	0.724 arcmin/mm
	有効面積	440 cm ² @1.5 keV, 250 cm ² @8 keV
	角分解能	2' (HPD ¹)
XIS	視野	17'.8 × 17'.8
	エネルギー帯域	0.2–12 keV
	有効画素数	1024 × 1024
	1画素のサイズ	24 μm × 24 μm
	エネルギー分解能	~ 130 eV @ 6keV
	有効面積 (XRT-I 込み)	340 cm ² (FI), 390 cm ² (BI) @ 1.5 keV 350 cm ² (FI), 100 cm ² (BI) @ 8 keV
	時間分解能	8 s (Normal mode), 7.8 ms (P-Sum mode)
HXD	視野	34' × 34' (≲ 100 keV), 4°.5 × 4°.5 (≳ 100 keV)
	エネルギー帯域	10–600 keV (PIN 10–70 keV, GSO 40–600 keV)
	エネルギー分解能	PIN ~ 4 keV(FWHM), GSO 7.6/√E _{MeV} % (FWHM)
	有効面積	~ 160 cm ² @20 keV, ~ 260 cm ² @100 keV
	時間分解能	61 μs
HXD-WAM	視野	2π (non-pointing)
	エネルギー帯域	50 keV - 5 MeV
	有効面積	800 cm ² at 100 keV / 400 cm ² at 1 MeV
	時間分解能	31.25 ms for GRB, 1 s for All-Sky-Monitor

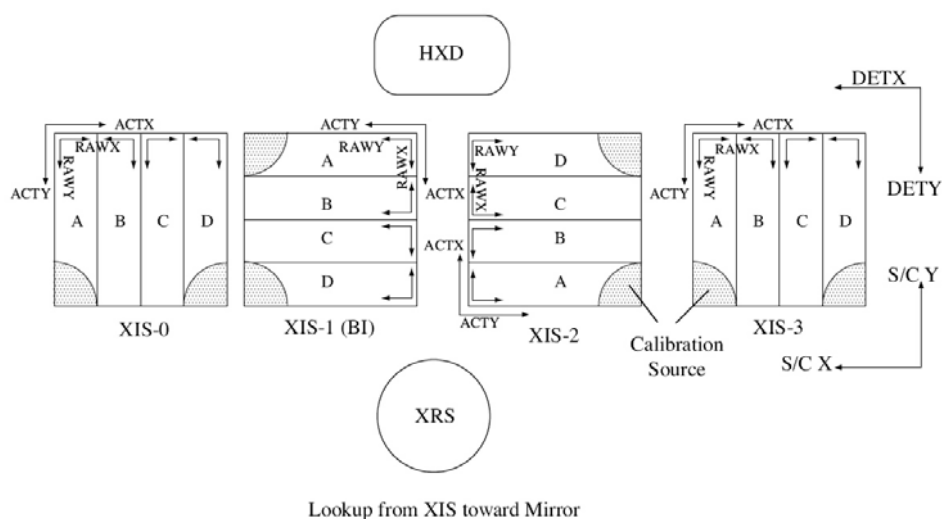


図 2.2: XIS の座標系と較正線源の位置 (Astro-E2 XIS Science FITS 確認書より)。この図は XIS から望遠鏡を見上げた図になっていることに注意。なお、RAW 座標系 (RAWX, RAWY) と Actual 座標系 (ACTX, ACTY) は XIS に特有の座標系である。

2.2 XRT

X線は物質中で強く吸収され、屈折率が1よりわずかに小さいという特徴を持つ。このことはX線の光学系を作る上で、屈折レンズが作れないこと、反射鏡は全反射²のみが利用できることを意味する。しかも屈折率の1からのずれが非常に小さいため、全反射は鏡面すれすれの角度の光線に対してのみおこる。例えば光子エネルギー数 keV の X 線が金の鏡面で全反射するのは鏡面から1度以下の入射角に限られる。この角度はX線のエネルギーが高くなるほど小さくなる。このような性質から、X線望遠鏡は全反射鏡と呼ばれる特殊な反射鏡を用いる必要がある。実際に日本で第4番目のX線天文衛星「あすか」にも金の全反射鏡を使った望遠鏡が搭載された。

「すざく」のX線望遠鏡(XRT)は、「あすか」のXRTよりもひとまわり大きいもので、口径40 cm、焦点距離4.75 mのXRT-I(焦点にXISを置くもの)が4台と、口径40 cm、焦点距離4.5 mのXRT-S(焦点にXRSを置くもの)が1台ある(外観と配置は図2.4)。反射鏡は、アルミ薄板にレプリカ法で鏡面を形成したレプリカミラーをそれぞれ175および168枚同心円状に並べて、小型超軽量だが高い効率のX線望遠鏡を構成している。この望遠鏡では光学系として、双曲面と放物面からなるWolter I型と呼ばれるものを円錐2段で近似して用いている(図2.3)。

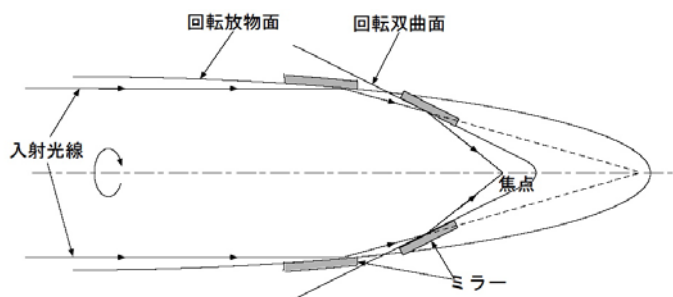


図 2.3: Wolter I 型 X 線反射鏡 (山下朗子 修士論文より)

レプリカ法の導入により鏡面形状精度が向上し、「あすか」に比べ約2倍優れた角分解能(HPD \sim 2')を達成した。また、焦点距離が長くなったことで、平均の斜入射角が小さく、エネルギーの高い側での反射率が2倍(@6 keV)程度向上した。「すざく」では、反射鏡の上にプリコリメータを加えることにより、多重薄板X線望遠鏡の問題であった迷光³を約1桁減少させた。

XRT-I + XIS 検出器で点源を観測すると、X線望遠鏡の特性(応答)により、ある広がりをもったイメージとして捉えられる(図2.5上)。これを方位角方向に平均化し、点像であるべき天体の像がその望遠鏡システムでどのように結像されるかを示した関数が Point Spread Function(PSF)である(図2.5下)。また、観測できる空の領域は、XRTと焦点面検出器の相対的な位置関係によって決まる。XRT-Iの光軸の位置⁴とXIS検出器の中心の相対的な関係を図2.6に示す。光軸の位置は検出器の中心と完全に一致はしていないが、4台とも約1'以内に入っている。

²もしくはブラッグ反射

³視野外にある明るいX線源からのもれこみ。

⁴光軸の位置は、その有効面積が最大になる場所として定義される。

表 2.2: XRT-I の性能

	Suzaku/XRT-I	ASCA/XRT
台数	4	4
焦点面距離	4.75 m	3.5 m
直径	399 mm	345 mm
重量 (一台あたり)	19.5 kg	9.8 kg
鏡面	Au	Au
鏡面数 (一台あたり)	1400	960
入射角	$0^{\circ}.18 - 0^{\circ}.60$	$0^{\circ}.24 - 0^{\circ}.70$
角分解能 (HPD)	$2'.0$	$3'.5$

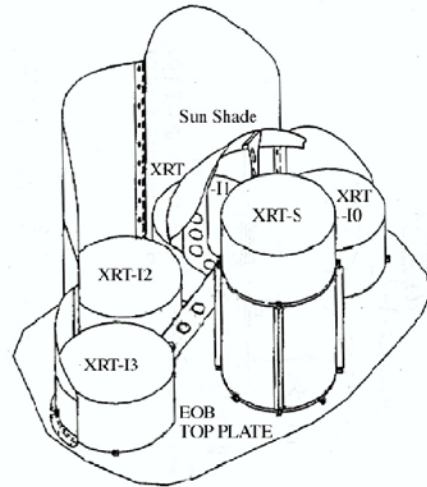


図 2.4: XRT の外観 (左) と配置図 (右) (The Suzaku Technical Description より)

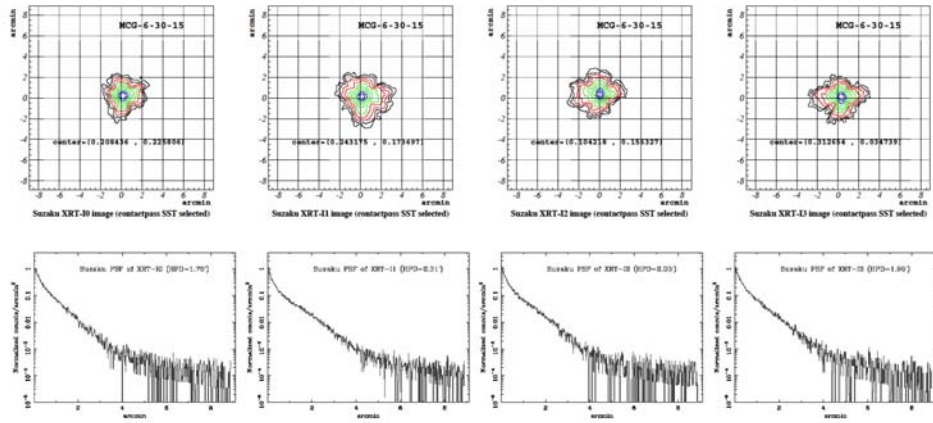


図 2.5: XRT-I + XIS による点源 (MCG-6-30-15) のイメージ (上) と Point Spread Function (下)(The Suzaku Technical Description より)。左から XRT-I0, I1, I2, I3。

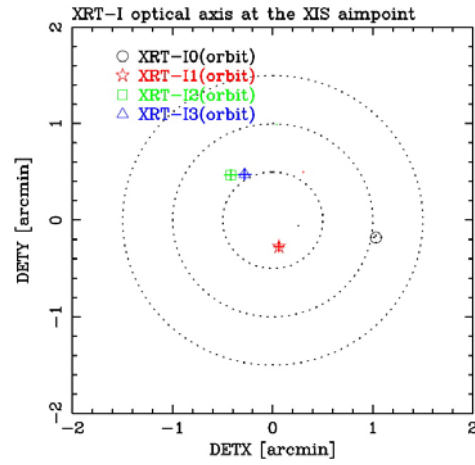


図 2.6: XIS 各センサーにおける光軸の位置 (検出器座標)

2.3 XIS

X 線検出に用いられる CCD⁵は、半導体検出器を 2 次元アレイ状に並べたものといえる。CCD のある画素に X 線が入射すると、ある確率で光電吸収が起こる。その結果生じた光電子はエネルギーを失うまで次々と Si 原子と衝突を繰り返し、電子・正孔対を作る。生じる電子・正孔対の数は、入射 X 線エネルギー E に比例し、およそ (E/W_{Si}) 個となる (ここで、 W_{Si} は Si の平均電離エネルギー ~ 3.65 eV)。こうしてできた電子の集まり (一次電子雲と呼ぶ) を正確に検出することによって、入射 X 線エネルギーを知ることができる。また、検出器の位置分解能は画素の大きさ (XIS では $24 \mu\text{m}$) によって決まり、比例計数管 (およそ $100 \mu\text{m}$) と比べて優れている。入射 X 線は空乏層内⁶で吸収されなければ正確なエネルギー測定ができないため、高いエネルギーの X 線の検出効率を上げるには、空乏層を厚くする必要がある。

「すざく」の XIS は 4 台の X 線 CCD カメラから構成され (図 2.7)、天体の撮像と X 線スペクトルの取得を目的としている。「あすか」に搭載された CCD カメラ (SIS) に比べて、空乏層の厚さが 2 倍になったため、高エネルギーの X 線に対する感度が向上している (≥ 7 keV で約 2 倍)。また、CCD の動作温度を -90°C まで下げたことで暗電流を押さえ、電荷転送非送率⁷を減少させるなど、様々な工夫がなされている。



図 2.7: XIS 検出器の外観 (The Suzaku Technical Description より)

XIS の 4 台のセンサーをそれぞれ X0, X1, X2, X3 と呼ぶ。また、CCD には表面照射型 (Frontside Illuminated; FI) と裏面照射型 (Backside Illuminated; BI) がある (図 2.8)。表面照射型 CCD では X 線を電極側から入射するため、低エネルギーの X 線は電極や絶縁層で吸収されてしまうのに対し、裏面照射型 CCD では X 線を電極の逆側から入射するため低エネルギーの X 線に対して高い検出効率を得ることができる。X0, X2, X3 の 3 台が FI-CCD、X1 が BI-CCD である。図 2.9 に XIS の模式図を示す。

XIS の観測モードは、Clock モードと Edit モードという異なる 2 つのモードから定義される。Clock モードには、Normal と Parallel-sum (P-sum) の 2 通りがある。

- Normal モードは CCD の全てのピクセルを通常 8 秒周期で読み出す。この場合、露光時間は 8 秒になる。Normal モードは Edit モード (5×5 , 3×3 , 2×2 のいずれか⁸) と組み合わせることができる。

⁵Charge Coupled Device(電荷結合素子) の略。

⁶空乏層の説明が必要であろう

⁷Charge Transfer Inefficiency (CTI) のこと。CCD ではゲート電圧を変化させることで、電荷をバケツリレーのように隣のピクセルに渡していく。転送中に何らかの形で電荷をトラップするピクセルがあると、電荷が完全に転送されず信号の劣化が起こる。

⁸例えば 5×5 モードでは、X 線イベントの中心座標とそれを取り巻く 24 ピクセルの合わせて 25 ピクセル分のパルスハイトがテレメトリ (人工衛星から地上に送られるデータ) に出力される。他のモードについては、The Suzaku Technical Description を参照。

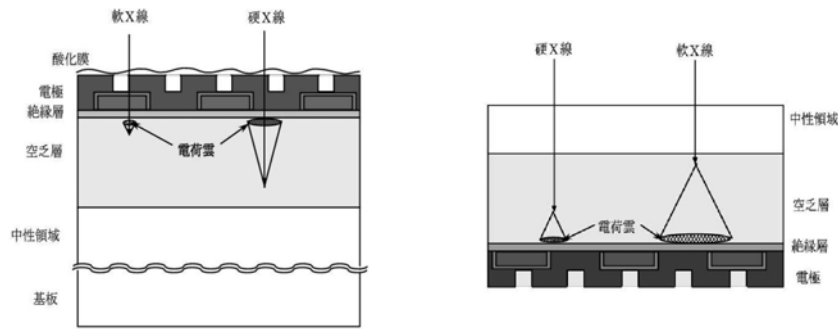


図 2.8: CCD の断面図 (東海林雅之 修士論文より)。表面照射型 CCD (左) と背面照射型 CCD (右) を模式的に示したもの。

また、Burst オプション⁹ や Window オプション¹⁰ を指定できる。

- P-sum モードは、撮像領域において縦方向に 64/128/256 列を加算し、一列分ずつデータを読み出す操作を行う。これにより、縦方向の位置情報は失うが、その代わりに時間情報を得ることができる。時間分解能は加算列数によらず 8/1024 sec (~ 8 ミリ秒) である。

X 線の入射により生成された電子雲は一つのピクセルにとどまる場合と、ピクセル境界付近に入射して 2 つ以上のピクセルにまたがる場合とがある。そこで、XIS では中心が最も波高の高い 3×3 ピクセルに注目して、スプリットの仕方ですべて 7 通りのグレードに分類する¹¹。そのうち、解析に用いられるのは Normal/Burst モードでは、グレード 0,2,3,4,6、P-sum モードでは、グレード 0,1,2 である。

XIS では軌道上でのエネルギーの絶対精度の測定のために、カメラごとに較正線源が取り付けられている。線源はいずれも ^{55}Fe (半減期 2.7 年) で、Mn $K\alpha$ (5.9 keV)、Mn $K\beta$ (6.5 keV) の特性 X 線を出す。較正線源の位置については、図 2.2 に示した。

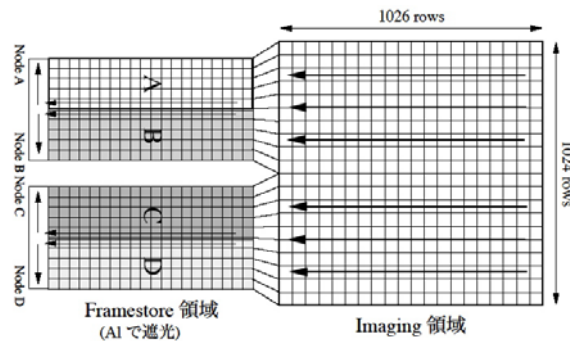


図 2.9: XIS センサーの模式図 (片山晴善 修士論文; 勝田 哲 修士論文より)。イメージング領域とフレームストア領域からなる。一枚の CCD には 4 つの読み出し口がある。

⁹Burst オプションの場合も読み出しの周期は 8 秒であるが、撮像領域の電荷を一度転送してクリアする操作が入るので実効的な露出時間は 8 秒よりも短くなる。Burst オプションは、明るい天体に対してバイルアップが起こるのを防ぐために利用する。

¹⁰Window オプションでは、CCD の指定した範囲にあるピクセルのみを短い周期で何度も読み出すことができる。このオプションは明るくて空間的ひろがり小さいソースに対してバイルアップを避け効率のよい観測を行うために利用する。

¹¹XIS グレード定義は、あすか SIS とほぼ同じ。詳細は Astro-E2 実験報告書などを参照。

2.4 HXD

HXDは、10～700 keVの広いエネルギー範囲の硬X線をこれまでにない高い感度で観測することを目的とした検出器である(外観は図2.10)。硬X線領域では、天体からの信号は典型的にエネルギーに対してべき関数的に減衰し、バックグラウンドに対して信号が微弱となる。よって高いエネルギーほど、検出器におけるバックグラウンドの低減が精度のよい観測をするうえで不可欠である。バックグラウンドには検出器の正面から入射するもの、視野外からシールドを通過して入射してくるもの、検出器に内在するものなどの成分がある。これらすべてを低減するようにもともと気球実験を通じて開発されたのが井戸型フォスウィッチカウンタであり、HXDではこの技術が応用されている。

HXDセンサーの構造を図2.11に示す。基本となる井戸型フォスウィッチカウンタは16本あり(Wellユニット)、その周りをBGO結晶のアンチカウンタ(Antiユニット)20本が取り囲む。Wellユニットの主検出部はPIN型半導体検出器(厚さ2mm)とGSOシンチレータ(厚さ5mm)を上下に重ねた形で構成され、10–700 keVという広帯域を実現する。さらにXISと組み合わせると、一つの衛星で3桁を超えるエネルギー帯域が同時にカバーされることになる。また、主検出部の周りのシールド部には深い井戸型をしたBGOシンチレータが用いられている。主検出部とBGOの反同時計測により、効率よくバックグラウンドを除去することができる。各ユニットについて簡単にまとめる。

なお、HXDの光軸はXIS nominal positionに対してDETX方向に $-3'.5$ ずれている。

- Well 検出器ユニット

天体からのX線は、 4×4 のマトリクス状に並べられた16本のWell検出器によって検出される。1本のWell検出器は約4.63 kgの重量をもつ。10～50 keV程度のエネルギーをもつX線は、PIN型半導体検出器で検出され、40～700 keVのX線はPINを突きぬけてその下のGSO結晶で検出される。バックグラウンド(線や荷電粒子)や視野外からのX線はBGOによって効率よく除去される。BGOは有効的な原子番号が大きく、線の阻止能に優れている。また、BGOの蛍光減衰時間は300 nsで、一方のGSOはより早い蛍光減衰時間(60 ns)をもつので、両者の違いを利用して、主検出部からの信号とシールド部の信号を波形弁別し反同時係数を行う。またマトリクス状に組まれた各ユニットは、シールドの薄い部分を通り抜けてくるバックグラウンドを互いに遮蔽しあう。

BGOの井戸部はアクティブコリメータの役割を持ち、視野を $4^\circ.6 \times 4^\circ.6$ に絞る。井戸部にはリン青銅製パッシブファインコリメータが挿入されていて、これが入射角の大きなX線を吸収し、低エネルギー側では視野は $0^\circ.56 \times 0^\circ.56$ に絞り込まれる。

- Anti 検出器ユニット

Anti検出器は、厚い(平均で2.6 cm厚)BGO結晶とフォトチューブを組み合わせた検出器である。このユニットをWellユニットの周辺に20本ならべることによって、放射線に対するアクティブシールドの役割を果たす。Anti検出器ユニットの有効面積は一面あたり $\sim 1200 \text{ cm}^2$ にもなり、厚いBGO結晶は1 MeVに対しても $\sim 600 \text{ cm}^2$ の面積をもつ。また、この検出器はあらゆる方向からくるX線・線にさらされており、ガンマ線バーストなどトランジェント天体の全天モニター(WAM: Wideband All-sky Monitor)として利用されている¹²。

¹²<http://www.astro.isas.ac.jp/suzaku/HXD-WAM/WAM-GRB/>



図 2.10: HXD の外観 (The Suzaku Technical Description より)

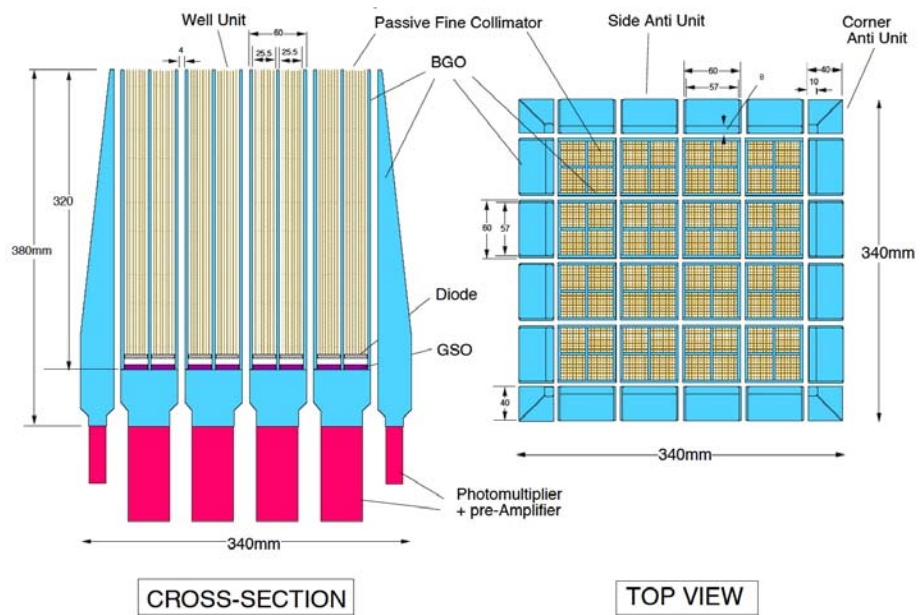


図 2.11: HXD の構造 (The Suzaku Technical Description より)

References

- ISAS/JAXA、「Astro-E2 実験計画書」
- ISAS/JAXA and NASA/GSFC, “The Suzaku Technical Description” ver. November 15, 2005
- 鶴 剛、松本 浩典、「Astro-E2 XIS Science FITS 確認書」2005/09/24 Ver.3e
- 山下 朗子、修士論文、1995 年 東京大学
- 片山 晴善、修士論文、1999 年 大阪大学
- 勝田 哲、修士論文、2004 年 大阪大学
- 東海林 雅幸、修士論文、2004 年 大阪大学
- 中嶋 大、修士論文、2004 年 京都大学
- 山口 弘悦、修士論文、2005 年 京都大学
- 寺田 幸功、修士論文、1999 年 東京大学

第3章 すざく衛星データ概要と解析ルートマップ

(SUMMARY AND ROUTE MAP OF ANALYSES)

3.1 FITS 形式って? (FITS Data Format)

「すざく」衛星は、公共天文台として機能し、国際的な競争力をつけるため、その観測データは、すべて、天文業界標準の FITS(Flexible Image Transport System) 型式に変換され、保存される。最終的には、ISAS/JAXA の DARTS や共同研究機関である NASA/GSFC の HEASARC グループによって管理され、人類共通の知的財産として残される。FITS 型式といっても難しいものではなく、図 3.1 のように、いろいろなコラム毎の値を複数の Row 詰めたテーブルに、そのデータの説明として、コラムの意味や、いろいろなキーワードの値を記述したヘッダ (ASCII) がついたファイルである。

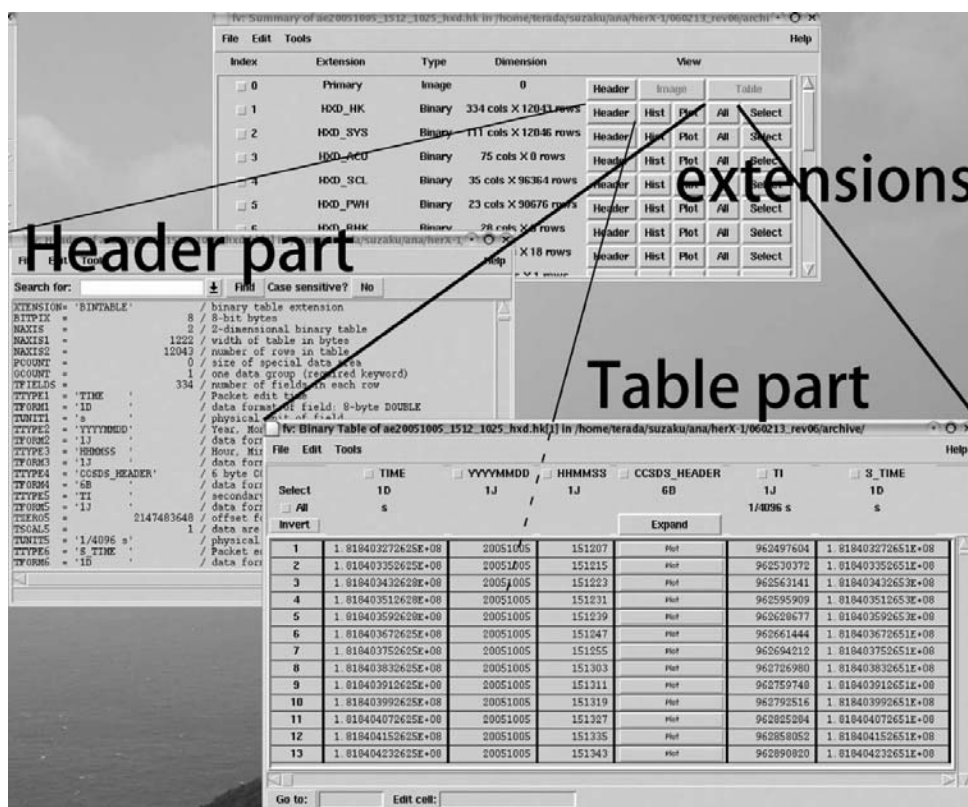


図 3.1: FITS ファイルを閲覧した画面。ヘッダ部とテーブル部のペアが、複数の extension として連なっているのが分かる。一番下表の table part にある 'TIME' や 'YYYMMDD' が「コラム」名、縦に 1,2,3,4.. とある列が「ROW」であり、Header part にある 'TIME / Packet edit time' などが各コラムの説明となっている。NAXIS2 などは「キーワード」と呼ぶ。この header+table のペアは、一番上の欄の 'HXD_HK' という「extension」を展開したものである。

3.2 解析に必要な衛星データ (Data Types)

衛星から出力されるデータには、衛星の状態を知るためのデータ (House Keeping) と、検出器からの観測データ (observation) との二種類がある。さらに、衛星の姿勢を記述した姿勢データと軌道データ、時刻データを加え、五種類のデータを扱うことになる。解析では、さらに、検出器の較正情報が必要となる。これは caldb (calibration database) と呼ばれ、検出器チームから供給され、他の衛星の較正情報とあわせ、GSFC が管理している。これらは全て FITS format で配布される。

- House Keeping Data
衛星共通系、HXD、XIS で、それぞれ、auxil/aeNNNNNNNNN.hk, hxd/hk/aeNNNNNNNNNhxhd_0.hk, xis/hk/aeNNNNNNNNNxis_0.hk といった HK Fits が存在する。解析の便宜のため、.hk や .ehk ファイルから必要な情報を一つの auxil/aeNNNNNNNNN.mkf というファイルにまとめている。したがって、解析では基本的に .mkf を使用すればよい (もし .mkf に解析に必要な項目が含まれていない場合は、GSFC に依頼して追加する必要がある)。
- Observation data
観測データは hxd/event_uf/aeNNNNNNNNNhxhd_0_wel_uf.evt, xis/event_uf/aeNNNNNNNNNxi[0-3]_0_3x3n030_uf.evt,... というファイル名で配付される。(XIS の 3x3030 部はマイクロコード)
- Attitude data
衛星の姿勢ファイル。auxil/aeNNNNNNNNN.att。
- Time data
時刻データ。衛星搭載のデータレコーダに記録されたデータに時刻づけする際などに使用される。auxil/aeNNNNNNNNN.tim。
- Orbital data
衛星の軌道要素ファイル。auxil/aeNNNNNNNNN.orb
- Extended HK
姿勢ファイルや軌道要素ファイルを元に、衛星の向いている方向が時々刻々とどう変化したか、また、衛星が飛翔している (地球上の) 場所が観測に与える影響がどう変化したか、などを計算しまとめたファイルが Extended HK file, auxil/aeNNNNNNNNN.ehk である。
auxil/aeNNNNNNNNN.ehk
- caldb
較正情報 FITS。観測者には直接に配付されるのではないので、
<http://www.astro.isas.jaxa.jp/suzaku/caldb/> もしくは、
<http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/suzaku/> から手に入れる。

ここで、配付される観測データ (Observation data) には複数の種類がある。衛星から出力されるデータを全て含むデータは all event (もしくは SFF, 3.3 節参照) と呼び、天体を観測している時間帯のデータだけでなく、地球を見ている時間帯、SAA 等観測の質が悪い時間帯のデータや、(天体からの X 線ではなく) 宇宙線由来のイベントなどが含まれている。いっぽう、cleaned event とは、これらの不用なデータを破棄したデータを指す。いかにゴミを破棄するか (data reduction) は、検出器の較正がどこまで進んだか等におおきく依存しており、可能ならば、自分で最新の較正情報 (caldb) と較正ツール (3.3 節にある「すざく」ftools) とを用いてリダクションしなおすのが望ましい。cleaned event からさらに自分の欲しいイベントを抽出する方法などは、第 5 章を参照のこと。

3.3 観測データの較正情報の最新化 (Overview of Processing)

3.2 節に述べた観測データ (Observation data) は、衛星で観測されたあと、図 3.2 のようなプロセスを経て FITS 化され、較正情報が付加され、配付される。最初に、衛星の生データを FITS 化しただけのものは、First FITS File (FFF) と呼ばれ、較正情報が付加されたものは Secound FITS File (SFF) と呼ばれる。両者は全く同じ format であるが、FFF の段階では、X 線の到来時刻や、イベント種別、天空上での座標などといった、較正が必要なカラムは空欄になっており、それを *Suzaku ftools* と呼ばれるツールを通すことで埋めて SFF にする。FFF と SFF のフォーマットが同じであるため、*Suzaku ftools* は SFF も入力とすることができる。すなわち、検出器の較正がすすみ、*caldb* や *Suzaku ftools* が更新された場合は、何度でも、ユーザが SFF をプロセスしなおすことができる。

解析の際には、「すざく」の検出器の較正がどこまで進んだかなどを常に把握するように心掛け、できるだけ最新の *caldb* と *Suzaku ftools* とをそろえ、最新の SFF (前節でいう all event) を用意することが大前提となる。参考までに、付録 B に *Suzaku ftools* を含む解析ツール群のインストールの仕方と、第 10 章に、2006 年 04 月現在の *Suzaku ftools* のリスト・簡単な説明とを載せておく。

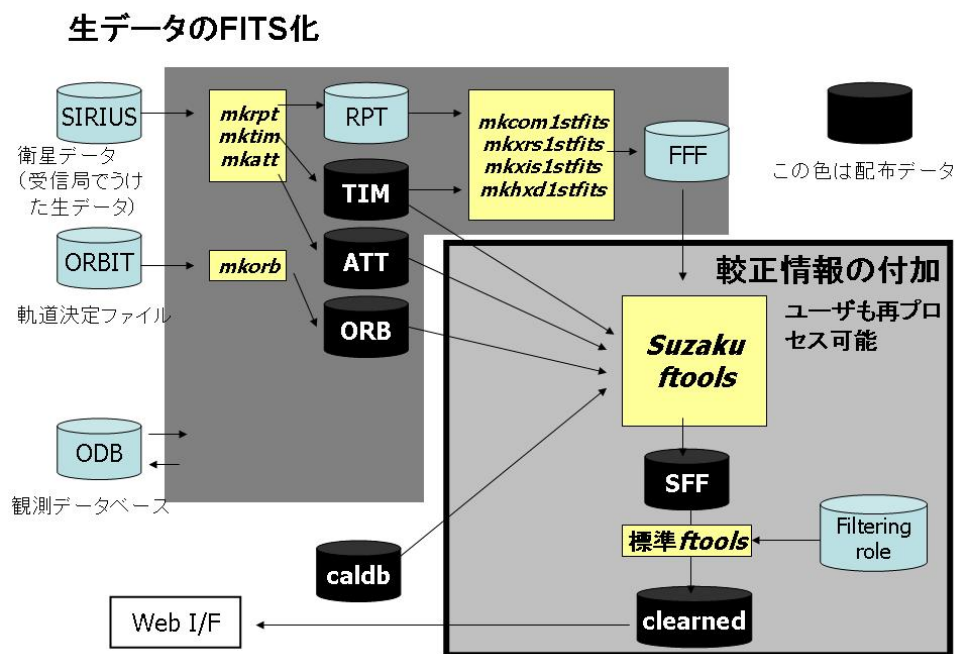


図 3.2: 「すざく」による観測が行なわれたあと、配付データが作成されるまでのプロセスを示した図。九州 USC にて取得されたデータは SIRIUS というデータベースに格納され、*mkrpt* を経て、Row Packet Telemetry data (RPT) という生データファイルに変換される。この RPT は生データであり公開しない。同様に、時刻データ、姿勢データ、軌道データなども生成される。RPT は、*mk***1stfits* というツールを用いて、より可読性のある FFF (本文参照) に変換される。その後、*Suzaku ftools* を用いて較正情報が付加され、SFF となり、さらに、余分なゴミデータを除去して、cleaned event となる。

3.4 標準的な解析 (Route Map of Your Analysis)

図 3.2 にある、SFF から cleaned event を作成するプロセスでは、衛星に依らない「標準の ftools」で処理される。SFF から先は、ユーザが行なう解析でも、同様に、「標準の ftools」で行なうことができる。図 3.3 にその概要を示す。

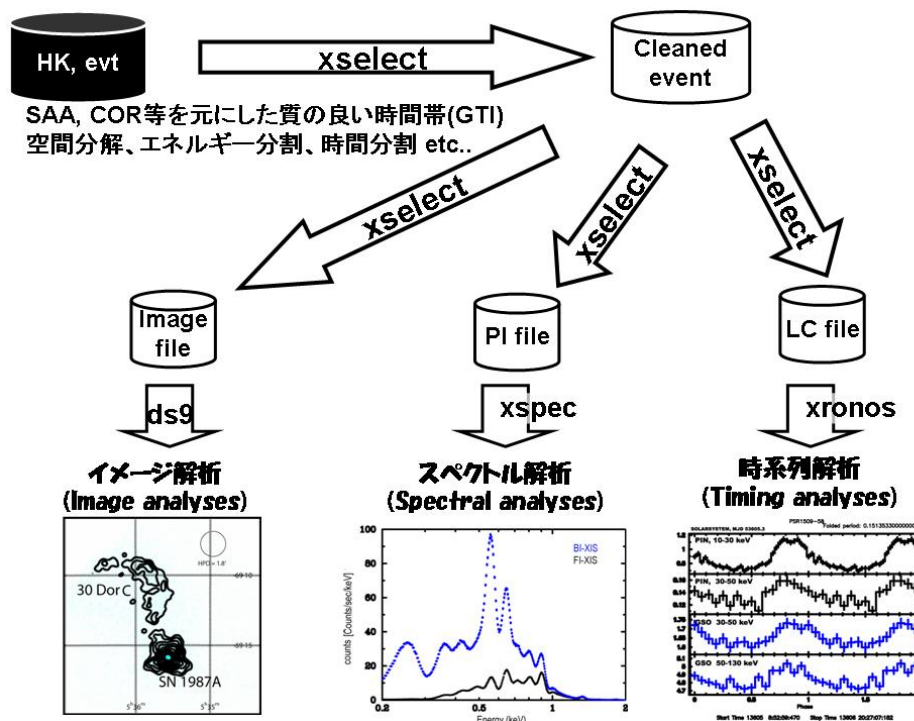


図 3.3: 標準の ftools を用いた衛星データ解析の概要

3.4.1 イメージを描きたい (Image analysis)

XIS のイメージ解析を行ないたい場合は、xselect (第 5 章) にて event file から image FITS file を抽出し、ds9(第 6 章) 等のツールを用いる。

3.4.2 X 線スペクトルをみたい (Spectral analysis)

スペクトル解析を行ないたい場合は、xselect (第 5 章) にて 波高空間でのヒストグラムファイル, PI (pulse height invariant) FITS file を作成し、xspec (第 8 章) にてスペクトルを解析する。

入射する X 線のエネルギー (E) は、検出器から出力される波高値 PHA と対応づけられているが、PHA のままでは、検出器のゲインや ADC の非線型性等といった個性がそのまま見えるので、こうした情報を補正し、個性を無くしたものが PI である。PI と E の関係は、必ずしも 1:1 対応ではなく、有限のエネルギー分解能でなまされ、また、低い PI 側に構造がでたりする。この E と PI の関係を記述したファイルをレスポンスファイル (rsp) と言う。また、PI FITS file にはバックグラウンドも含まれているので、バックグラウンドに相当する PI ヒストグラムを差し引いて解析を行なう必要がある。つまり、X 線スペクトルを解析する上で必要なファイルは、

- 観測した PI FITS file
 - Background PI file
 - レスポンスファイル (rsp, or arf and rmf)
- の三点セットである。

これらがそろった段階で、`xspec` を用いた解析ができる。実際の検出器では、PI file に、レスポンス関数の逆行列をかけて、天体の放射スペクトルにもどす事は原理的に不可能なので、物理的に意味のある放射モデルを仮定し、その放射モデルにレスポンス関数を掛け合わせてできる PI 分布と、実際に観測された PI 分布とを比較することで、天体の放射スペクトルを推定する。これを *fitting* と呼ぶ。詳細は、第 8 章にある。

3.4.3 ライトカーブをみたい (Check light curve, and timing analysis)

ライトカーブをチェックしたい場合は、`xselect` (第 5 章) にて event file から light curve (lc) FITS file を抽出し、ソフト `lcurve` にて描くことができる。実際の解析では、こうして描いたライトカーブをみて、太陽フレアや粒子線起原のフレアアップなどが無いか、検出器がおかしな動作をしていないか、などを 必ずチェックして欲しい。必要ならば、`xselect` に戻って、不用と考えられる時間帯のデータを破棄する、などの処理を行なう。

`lcurve` というコマンドは、標準ツールのなかでも、XRONOS (第 7 章) とよばれる時系列解析ツール群の一種である。XRONOS には、パワースペクトルを書くコマンド `powspec` や、周期的な変動を探查するコマンド `efserch`、周期で畳み込んだライトカーブを描く `efold` などがある。

3.4.4 より詳細な解析へのステップアップ (Advanced analysis)

以上の三種の解析 (イメージ, スペクトル, 時系列) は基本技である。第 4 章では、これらの基本技の *walk through* を、XIS, HXD についてもうすこし具体的に示している。これらの基本技をマスターし、組み合わせることで、

- 場所毎のスペクトルを解析したり、
- エネルギー毎のイメージを描いたり、
- phase 毎のスペクトルを描いたり、
- 場所毎に周期性探查をしたり、

といった解析ができるようになる。第 5 章以降では、標準的な各ツールについて、より詳細な解説を載せているので、自分で組み合わせて使ってみて欲しい。

第4章 とりあえずやってみよう (WALK THROUGH OF *Suzaku* ANALYSIS)

本章では、「すざく」衛星のデータを実際に解析する場合の walk through を示す。Processing version 2.2.x.x のデータを対象に、HEAsoft v6.5 を用いた解析についての解説となっている。下記のホームページを参考にしてほしい。

- Software (HEAsoft) ¹:
<http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/>
- Calibration files:
<http://www.astro.isas.jaxa.jp/suzaku/caldb/>
- ABC guide:
<http://heasarc.gsfc.nasa.gov/docs/suzaku/analysis/abc/>

4.1 解析の前に真っ先にすべきこと:情報収集 (Getting Information)

- 「すざく」衛星の観測記録は下記の WWW page から辿れる。観測天体リスト、衛星運用に関する記録、XIS や HXD に関する記録、そして HK に関する記録をチェックすることができる。
<http://www.astro.isas.jaxa.jp/suzaku/log>
- データを解析するうえでのキャリブレーションの制限は下記にあるのでよく把握すべし。
<http://www.astro.isas.jaxa.jp/suzaku/process/>
- 検出器チームからのアナウンスを確認することも重要。
<http://www.astro.isas.jaxa.jp/suzaku/analysis/>
- 現在の Calibration Status などの確認は下記が便利
<http://www.astro.isas.jaxa.jp/suzaku/process/caveats/>
<http://heasarc.gsfc.nasa.gov/docs/suzaku/analysis/watchout.html>
<http://www.astro.isas.jaxa.jp/suzaku/doc/>

4.2 データの取得 (Getting *Suzaku* Data)

まずはじめに解析したい天体のデータの取得を行なう。すざく観測データは、

- DARTS (ISAS), <http://darts.isas.jaxa.jp/astro/suzaku>
- HEASARC (NASA), <ftp://legacy.gsfc.nasa.gov>

¹HEAsoft 6.0.6 以前には「すざく」のツールは含まれない。また、改訂は3カ月に一度のペースで起こるので、解析にはなるべく新しいバージョンの HEAsoft を使用するのが望ましい (2008-11-17 現在での最新版は HEAsoft v.6.5.1)。

から取得できる。

まず、すざく衛星のデータ解析を行なうディレクトリに移動し、下記のように、wget コマンドを使って一度にデータを取得することもできる。データを持って来る前に、解析を行なう場所のディスク容量が充分にあることを df コマンド等で確認しておくこと。

```
unix% wget -rL -nH --cut-dirs=4 \  
ftp://ftp.darts.isas.jaxa.jp/pub/suzaku/ver2.0/101005040/  
(注意: 一行目の最後の \ は、その行が次の行に続くことを意味している。一行に入らなかったため、表  
記上改行したに過ぎない。実際には \ やリターンを打たずに、続けて次の行を打つこと.)
```

ダウンロードがすべて終ると、

```
auxil/    hxd/    xis/    log/
```

といったディレクトリができ、その下に、データが置かれているはず²

データは、PGP key で暗号化され保護されているので、

```
http://www.astro.isas.jaxa.jp/suzaku/process/decrypt/
```

にある decrypt_data.pl をダウンロードし、データを復号する必要がある。

```
unix% chmod +x decrypt_data.pl  
unix% ./decrypt_data.pl -d .  
Enter the password: (パスワードを入力)
```

ここで password は、データを受け取った時のメールに記載されているパスワードを使用する。すると、aeXXXXXhxd_0.hk.gz.gpg などと gpg で暗号化されていたファイルが解読され、aeXXXXXhxd_0.hk.gz となる (XXXXXX は 観測 ID で、10 ではじまるものは、公開データなので、暗号化されていない)。なお、decrypt_data.pl -p 'password' として、パスワードを直接入力するオプションが存在するが、コマンドラインでパスワードを書くと、そのソフトを実行中に ps コマンドでパスワードが読めてしまうので、使用してはいけない。

Comment:

次のセクションからは、解析環境を自前で準備し (付録 B)、ローカルな環境で解析する方法を紹介するが、より簡単には、ISAS PLAIN センターに用意された analysis server を利用するという方法もある。詳細は、http://plain.isas.jaxa.jp/ana_servers/を参照。

²こうしたディレクトリ構造など、データプロセスの詳細は、GSFC と日本側の pipe-line process のインターフェイスを規定した Interface Control Document (ICD) に詳しい。 http://heasarc.gsfc.nasa.gov/docs/astroe/archive/astroe_icd_sdc_v1.2.pdf

4.3 XISの解析 (XIS Analysis)

4.3.1 XIS 解析の準備 (Preparing for the XIS analysis)

では xis/ ディレクトリの中身を確認しよう。以下には、

```
xis/event_uf/ ... スクリーニングされていないデータ
                ae001122330xiN_0_5x5n000z_uf.evt.gz (N=0,1,2,3 for XIS0,1,2,3)
xis/event_cl/ ... スクリーニングされたデータ
                ae001122330xiN_0_5x5n00a_cl.evt.gz (N=0,1,2,3 for XIS0,1,2,3)
xis/products/ ... イメージなど
                ae001122330xiN_0_pi.gif etc..
xis/hk/        ... hk ファイルなど
                ae001122330xiN_0.hk.gz
```

のようなファイルがある。

4.3.2 cleaned event からの QL (QL with cleaned event)

とりあえずデータを QL したい時には、SAA、地没データなどが既に除去されている cleaned event がよい。xis/event_cl/ の下にある。event file からライトカーブやイメージ、スペクトルを作るには、xselect を実行しよう。xselect の説明は第 5 章に書いてあるので、詳細はそちらを参照のこと。

以下に実際に xselect を実行したときの出力を載せる。括弧書き () は注釈を表わしている。

注: _cl.evt.gz の直前のアルファベットは、area discri などの設定を表している。このアルファベットの違うデータ同士を足し合せて使用することは出来無い。

```
unix% xselect
```

```
** XSELECT V2.3 **
```

```
> Enter session name >[xsel] xsel
Setting plot device to /xw
xsel:SUZAKU > read events ae001122330xi0_5x5n000a_cl.evt.gz
```

```
> Enter the Event file dir >[./]
Got new mission: SUZAKU
> Reset the mission ? >[yes]
```

```
Notes: XSELECT set up for      SUZAKU
Time keyword is TIME          in units of s
Default timing binsize =     16.000
```

```
Setting...
```

```
Image keywords  = X           Y           with binning =   8
WMAP keywords   = X           Y           with binning =   8
Energy keyword   = PI                               with binning =   1
```

```
Getting Min and Max for Energy Column...
```

```
Got min and max for PI:      0    4095
```

Got the minimum time resolution of the read data: 8.0000
MJDREF = 5.1544000742870E+04 with TIMESYS = TT
Number of files read in: 1

***** Observation Catalogue *****

Data Directory is: /YOUR_ANALYSIS_DIRECTORY/xis/

HK Directory is: /YOUR_ANALYSIS_DIRECTORY/xis/

OBJECT	DATE-OBS	TIME-OBS	DATE-END	TIME-END	EDITMODE
1 A0535+26	2005-09-14	13:49:10	2005-09-15	00:59:50	2x2

xsel:SUZAKU-XIS0-STANDARD > extract all

extractor v4.62 31 Jan 2006

Getting FITS WCS Keywords

Doing file: /YOUR_ANALYSIS_DIRECTORY/xis/ae001122330xi0_5x5n000a_cl.evt

100\% completed

Total	Good	Bad: Region	Time	Phase	Grade	Cut
72558	72558	0	0	0	0	0

=====

Grand Total	Good	Bad: Region	Time	Phase	Grade	Cut
72558	72558	0	0	0	0	0

in 7419.2 seconds

Fits light curve has 72558 counts for 9.780 counts/sec

Thresholding removed more than half the bins

Try exposure=0.0 on the extract command in xselect

or lcthresh=0.0 if running extractor stand-alone

Spectrum has 72558 counts for 9.780 counts/sec

... written the PHA data Extension

Image has 72558 counts for 9.780 counts/sec

(観測されたイメージをみるため、saoimage で ds9 の window を立ち上げてみよう。)

xsel:SUZAKU-XIS0-STANDARD > saoimage

(これでソースのイメージが見られる。

ds9 の詳細な説明は別章に記述があるので、ここでは省く。

ソースのイメージファイルを保存するには、以下のようにすればよい。)

xsel:SUZAKU-XIS0-STANDARD > save image

> Give output file name >[] xis0

Wrote image to file xis0.img

(ライトカーブを見て、保存する。)

```
xsel:SUZAKU-XIS0-STANDARD > plot curve
```

(qdp window が立ち上がり、qdp のプロンプトが表示される。
qdp を終了するときは q と打とう。)

```
xsel:SUZAKU-XIS0-STANDARD > save curve
> Give output file name >[xis0] xis0
Wrote FITS light curve to file xis0.lc
```

(最後にスペクトルを見て、保存しよう。)

```
xsel:SUZAKU-XIS0-STANDARD > plot spectrum
```

(ライトカーブの時と同様に、qdp window が立ち上がり、
qdp のプロンプトが表示される。)

```
xsel:SUZAKU-XIS0-STANDARD > save spectrum
> Give output file name >[] xis0
Wrote spectrum to xis0.pha
```

(xselect を終了する時は、以下のように打つ。)

```
xsel:SUZAKU-XIS0-STANDARD > quit
> Save this session? >[no]
```

ここで作ったライトカーブ、スペクトルのファイルは XRONOS や XSPEC を用いて解析を行なえる。詳しくは第 7、8 章を参照してほしい。スペクトル解析のための rmf 作成、arf 作成はそれぞれ 10.1、10.2 にある。

XIS の バックグラウンドのうち、Non X-ray background (NXB) 成分については、地没のデータベースをもとにして、観測中の Cut off rigidity (COR) で重みをつけて作成する。詳細については、
<http://www.astro.isas.jaxa.jp/suzaku/analysis/xis/nte>
を参照のこと。

4.3.3 XIS データに最新 CALDB をかけなおす

最新のキャリブレーションを解析にとりこむためには、手持ちの XIS データに最新の CALDB をかけ直す必要がある。以下のような手順で行なう。

CALDB をインストールした後、

```
example% xispi infile=ae101005070xi0_0_3x3n066z_uf.evt.gz \
          outfile=ae101005070xi0_0_3x3n066z_uf_new.evt \
          hkfile=./hk/ae101005070xi0_0.hk.gz
```

注意!: 3x3 モードのデータと 5x5 モードのデータは、必ず別々に xispi をかけること。

4.4 HXD の解析 (HXD Analysis)

4.4.1 データの取得 (Gathering datasets)

4.4.1-a データファイルのチェック (Checking data files)

HXD のデータを取得すると、下記のようなファイル群が得られる。

```
auxil/          <----- 機器で共通なファイル
  ae101005040.hk.gz
  ae101005040.ehk.gz  (EHK ファイル)
  ae101005040.cat.gz
  ae101005040.mkf.gz  (MKF ファイル; xselect で使用)
  ae101005040.att.gz  (姿勢ファイル)
  ae101005040.orb.gz  (軌道ファイル)
  ae101005040.tim.gz  (時刻ファイル)

hxd/hk/         <----- HXD の HK ファイルと 一時的 gain history, GTI
  ae101005040hxd_0.hk.gz
  ae101005040hxd_0_pin_ghf.gz      (PIN gain history; 不使用)
  ae101005040hxd_0_gso_ghf.gz      (PIN gain history; 一時的なゲイン)
  ae101005040hxd_0_wam_ghf.gz      (WAM gain history; 一時的なゲイン)
  ae101005040hxd_0_wel_uf.gti.gz    (cleaned event 抽出用の時間帯 GTI)
  ae101005040hxd_0_tel_uf.gti.gz    (テレメトリ飽和のない時間帯)
                                     cleaned event には両 GTI を適用

hxd/event_uf/  <----- Unscreened Event File
  ae101005040hxd_0_wel_uf.evt.gz    (WEL = PIN + GSO + Pseudo + noise)
  ae101005040hxd_0_wam_uf.evt.gz    (WAM イベントファイル)
  ae101005040hxd_0_bst01_uf.evt.gz  (WAM, BST data)
  ae101005040hxd_0_bstidt.fits.gz   (WAM 時刻づけ用ファイル)
  イベントのファイル名にある _0_ の部分は、ファイルサイズが 2 Gbyte よりも
  大きくなれば _1_, _2_, _3_, ..... というファイル群に分割される。ここに示し
  た _0_ は "統合されたファイル"を意味し、2 G byte より小さい場合に 1 ファイ
  ルにまとめられる。

hxd/event_cl/  <----- Screened Event File
  ae101005040hxd_0_gsono_cl.evt.gz  (GSO cleaned events; たいてい再プロセス必要)
  ae101005040hxd_0_pinno_cl.evt.gz  (PIN cleaned events)
  ae101005040hxd_0_pse_cl.evt.gz    (Pseudo events; dead time 補正)

hxd/products/ <----- Pipe-line process の生成物
  ae101005040hxd_0_gsono_sr.lc ae101005040hxd_0_gsono_sr.pi
  ae101005040hxd_0_pinno_sr.lc ae101005040hxd_0_pinno_sr.pi
  ae101005040hxd_0_wam.lc      ae101005040hxd_0_bst01_sr.lc
  ae101005040hxd_0_wel_lc.gif  ae101005040hxd_0_wel_pi.gif
```

4.4.1-b CALDB files の取得 (Getting CALDB files)

正確に解析したい場合は、HXDのデータは次節 (§4.4.2) のように再プロセスする必要があるので、CALDB ファイルも必要となる。CALDB ファイルは、

<http://www.astro.isas.jaxa.jp/suzaku/caldb/>

もしくは、

<http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/suzaku/>

から取得できる。HXD に関しては下記。

```
ae_hxd_gsoght_20081001.fits <---- GSO のゲイン変化を表すモデルのパラメータテーブル
ae_hxd_gsoghf_20070704.fits <---- GSO バックグラウンドラインのフィット結果 (不要)
ae_hxd_pinghf_20051125.fits (PIN):ダミー
ae_hxd_gsolin_20051209.fits <---- PI 決定に使用する線形性情報 (GSO)
ae_hxd_pinlin_20060724.fits (PIN)
ae_hxd_gsopsd_20071010.fits <---- イベント種別 (GRADE) 決定用情報 (GSO)
ae_hxd_pinthr_20080717.fits (PIN)
ae_hxd_gsoart_20051126.fits <---- arf builder 用データベース (GSO)
ae_hxd_pinart_20070611.fits (PIN)
ae_hxd_teldef_20060810.fits <---- arf builder 用 Teldef file
ae_hxd_wampht_20061027.fits <---- HXD-WAM 用 CALB (ここでは不使用)
ae_hxd_bstidt_20070512.fits <---- HXD-WAM バーストデータの ID テーブル (不使用)
```

ここで、GSO のゲイン履歴ファイル (ae_hxd_gsoght_XXXXXXXXX.fits) は³、約 1ヶ月に一度のペースで更新されるので (毎月初旬)、自分の観測日時が含まれるか否か、常によく CALDB 領域を監視し取得しておく⁴。

HEADAS の Suzaku ftools が CALDB に自動アクセスするためには、caldb のディレクトリに、「すざく」特有のファイル以外にも、共通ツールをインストールしておく必要がある。詳細は、http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_install.html を参照のこと。ftools では、下記のように、CALDB ファイルを直接指定するかわりに、'CALDB' と記せば自動アクセスされる。
(例) `hxdpi hxd_gsoght_fname=ae_hxd_gsoght_20081001.fits ...`
→ `hxdpi hxd_gsoght_fname=CALDB ...`

³Process version 1.x では ae_hxd_gsoghf_XXXXXXXXX.fits を使用していたが、version 2.0 以降、メンテナンスされていない。Version 2.x 用のツールと ae_hxd_gsoght_XXXXXXXXX.fits を使用すること。

⁴Version 2.0 からは、プロセス用ソフトが、CALDB の適切なファイルを自動的に選択するようになった。正しい gsoght が使用されたかは、生成された event fits のヘッダを確認する。なお、ネットワーク経由で CALDB にリモートアクセスする方法は、http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_remote_access.html を参照のこと。

4.4.1-c. Non X-ray background の取得 (Getting NXB files)

HXD のエネルギー帯域では、多くの場合、観測された信号中におけるバックグラウンドの寄与が無視できない。HXD では、観測ごとに ON/OFF 観測を行なって OFF 観測をバックグラウンドとして差し引くという方法は採用していない。そのかわりに、バックグラウンドの大部分を占める検出器由来のバックグラウンド、すなわち Non X-ray background (NXB) を観測ごとにモデル化し、公開している。このほかに、天球面上にほぼ一様に存在している Cosmic X-ray Background (CXB) や、我々の銀河からの寄与を加えたものが、最終的なバックグラウンドとなる。

NXB は軌道 (COR/SAA 通過) や実際に浴びたプロトン数 (PIN でカウント)、およびセンサー内部の重元素 (Gd, Bi...) の放射化によって時間変化する。これらの情報をもとに、NXB は 2008 年 4 月現在、2 通りの方法でモデル化されており、HXD 検出器チームから下記の URL を紹介⁵、観測毎に配布されている⁶。

```
·http://www.astro.isas.jaxa.jp/suzaku/analysis/hxd/pinnxb, gsonxb
·http://heasarc.gsfc.nasa.gov/docs/suzaku/analysis/pinbgd.html, gsobgd.html
  (ftp://legacy.gsfc.nasa.gov/suzaku/data/background/)
```

科学論文に、いずれの NXB model を使用したかを記載したい場合は、自分の使用した NXB event FITS のヘッダーにある 'METHOD' (モデル化の手法名) と 'METHODV' (バージョン) とを書くといよい。

```
METHOD = 'PINUDLCUNIT' / Method of the background modeling
METHODV = '2.0pre20071002' / Version of the the background modeling method
```

HXD のバックグラウンドとしては、前述のとおり、NXB 以外に Cosmic X-ray Background (CXB) も考慮しないといけない。CXB のスペクトルは過去の衛星で詳細に測定されており、その放射モデルを元に推定する。その際のレスポンス関数は、点源レスポンスではなく、天球からフラットに放射されたときの応答関数を利用する。CXB の推定方法およびスペクトルの作りかたは <http://heasarc.gsfc.nasa.gov/docs/suzaku/analysis/pin-cxb.html> に記載されている。

4.4.1-d. PIN レスポンスについて (Choosing PIN response)

PIN のレスポンスは、基本的には時間とともに変化しない。しかし打ち上げ後、一部の PIN の印加電圧を変更したり、Lower threshold レベルを変えている。これらの変更に応じて、下のように 4 つの時期ごとのレスポンスが CALDB 中に用意されている。データ解析の際には、観測時期に応じた ae_hxd_pinthr_XXX.fits でプロセスしたデータを用いる必要があり、また、観測時期に応じてレスポンスを選択しなければならない。詳細は

<http://www.astro.isas.jaxa.jp/suzaku/analysis/hxd/pinnxb/>

を参照のこと。なお、Version1.x の時は、印加電圧が 400 V の PIN に対応したレスポンスが用意されておらず、500 V の PIN のみを使用した解析を推奨していたが、version2.x では、64 個すべての PIN を使ってデータ解析することができる。

Epoch	File(s)
2005.8.17 -- 2006.5.13	ae_hxd_pinXXXXXe1_20080129.rsp (XXXXX=xinom,hxnom,flat)
2006.5.13 -- 2006.10.2	ae_hxd_pinXXXXXe2_20080129.rsp (XXXXX=xinom,hxnom,flat)
2006.10.2 -- 2007.7.28	ae_hxd_pinXXXXXe3_20080129.rsp (XXXXX=xinom,hxnom,flat)
2007.7.28 -- 2008.9.1	ae_hxd_pinXXXXXe4_20080129.rsp (XXXXX=xinom,hxnom,flat)
2007.9.1 -- **	ae_hxd_pinXXXXXe5_20080716.rsp (XXXXX=xinom,hxnom,flat)

⁵version 1.2.x.x では 観測データとは別に NXB data を配布しているが、将来、NXB 推定ツール一式 (hxdnxbgen) がリリースされれば (2008 年夏の予定)、pipe-line process に組みこめるので、クイック版の NXB file は、DARTS や HEASARC から取得できるようになるだろう。詳細版の NXB はプロセスから 1.5 カ月ほど後に配布される。

⁶PIN については、観測時期ごとに使用するべきバックグラウンドが PIN background の web ページに表にしてまとめてある。

4.4.2 HXDデータの再プロセス (GSOを解析したい場合 or PINの一部のデータ)

パイプラインプロセスでは、GSOのPIを決めるのに使用するゲインは一時的なものにせざるを得ないので、科学的解析には使用できない。すなわち、GSOの解析を行なう場合はこのセクションは必須である。また、V2.0のプロセスのPINのデータは、PLPINのキャリブレーションが一部間違っているので、再プロセスが必要である。プロセスのバージョンは、event FITSのヘッダにある'PROCVER'で知る事ができる。まとめると下記。no needとなっている場合は、次の§4.4.3にスキップしても構わない。

	ver 2.0.x.	ver 2.1.x.x	ver 2.2.x.x	それ以降
PIN	Reprocess	no need	no need	no need
GSO	Reprocess	Reprocess	Reprocess	Reprocess

「GSOのゲインが変わるだけ」といっても、PIが変化すると、GRADEを用いたイベントセレクションの判定結果(clean or dirty)も変化する。そこで、パイプラインで生成された cleaned event は一旦わすれ、

- unscreened event に戻って再プロセスし、
- それを screening する

という手順が必要である。

4.4.2-a reprocess an HXD unscreened event

例として、ae101005040hxd_0_wel_uf.evt という名の unscreened event file をプロセスし、ae101005040hxd_0_wel_uf2.evt を作成する方法を示す。HEADASのHXD ftools を使用する前に、§4.4.1-b に記した CALDB 環境の設定を行っておく必要がある。

まず、作業ディレクトリを作成し、必要なファイルをリンクする。

```
unix% mkdir event_cl2/; cd event_cl2/
unix% ln -s ../event_uf/ae101005040hxd_?_wel_uf.evt.gz .
           ? が 1,2,3... に分割されていれば、それぞれに対して再プロセスする。
unix% ln -s ../hk/ae101005040hxd_0.hk.gz .
unix% ln -s ../../auxil/ae101005040.ehk.gz .
unix% ln -s ../../auxil/ae101005040.tim.gz .
```

次に、HEADAS が動作する環境で、下記の3つを実行。

まずは時刻づけ (実は不用だが念のため)。

```
unix% hxdtime input_name=ae101005040hxd_0_wel_uf.evt.gz \
  create_name=ae101005040hxd_0_wel_uf2.evt \
  tim_filename=ae101005040.tim.gz \
  hklist_name=ae101005040hxd_0.hk.gz \
  leapfile=CALDB \
  read_iomode=create time_change=y grade_change=n pi_pmt_change=n \
  pi_pin_change=n gtimode=y gti_time=S_TIME time_convert_mode=4 \
  use_pwh_mode=n num_event=-1 event_freq=10000 anl_verbose=-1 anl_profile=yes
```

次に PI づけ。

```
unix% hxdpi input_name=ae101005040hxd_0_wel_uf2.evt \
  hklist_name="@hk_file.list" \
  hxd_gsoght_fname=CALDB hxd_gsolin_fname=CALDB \
```



```

hxd_pinghf_fname=CALDB hxd_pinlin_fname=CALDB \
create_name=hxdpi.out read_iomode=overwrite \
time_change=n grade_change=n pi_pmt_change=y pi_pin_change=y \
gtimode=n gti_time=S_TIME rand_seed=7 rand_skip=0 use_pwh_mode=n \
num_event=-1 event_freq=10000 anl_verbose=-1 anl_profile=yes

```

<----- ここで、hxd_gsoght_fname でアクセスされる GSO のゲイン履歴ファイルに、自分の観測データの時期が含まれていない場合は、hxdpi は何もせず終了する。CALDB が更新するのを待ってから、再挑戦する。

最後にイベント選別用のカラム GRADE/ DET_TYPE をうめる。

```

unix% hxdgrade input_name=ae101005040hxd_0_wel_uf2.evt \
hxdgrade_psdssel_fname=CALDB \
hxdgrade_pinthres_fname=CALDB \
leapfile=CALDB \
hxdgrade_psdssel_criteria=2.1 \
read_iomode=overwrite create_name=hxdgrade.out \
time_change=n grade_change=y pi_pmt_change=n pi_pin_change=n \
gtimode=n gti_time=S_TIME use_pwh_mode=n num_event=-1 \
event_freq=10000 anl_verbose=-1 anl_profile=yes

```

これで、ae101005040hxd_0_wel_uf2.evt が得られた。

4.4.2-b select cleaned events

次に ae101005040hxd_0_wel_uf2.evt をスクリーニングする。HXD の標準的な条件は、下記の条件⁷を満たす時間帯 (Good Time Interval; GTI) にあるイベントのうち、'DET_TYPE' コラムを用いて pure な PIN や GSO event を選び出す、というものである。(DET_TYPE = 0 が GSO, 1 が PIN, 2 が Pseudo。)

```

"AOCU_HK_CNT3_NML_P==1",
"ANG_DIST < 1.5",
"HXD_DTRATE < 3", (GSO only)
"SAA_HXD == 0",
"T_SAA_HXD > 500",
"TN_SAA_HXD > 180",
"ELV > 5",
"COR >6",
"HXD_HV_Wn_CAL>700" (n=0,1,2,3)
"HXD_HV_Tn_CAL>700" (n=0,1,2,3)
"テレメトリが飽和していない"

```

まずは、上記の条件の時間変化を記録したファイル群を、自分の作業ディレクトリに置く。

```

unix% cp ../../auxil/ae101005040.mkf.gz .
unix% gunzip ae101005040.mkf.gz      (gz を解凍した形で、作業ディレクトリにある必要がある)
unix% ln -s ../hk/ae101005040hxd_0_tel_uf.gti.gz .

```

選択条件は、GSFC の下記のディレクトリに置いてあるので、これを利用。

⁷正確な表現は、
http://www.astro.isas.jaxa.jp/suzaku/process/v2changes/criteria_hxd.html
で確認のこと。

```
unix% wget http://suzaku.gsfc.nasa.gov/docs/suzaku/analysis/pin_mkf.sel
unix% wget http://suzaku.gsfc.nasa.gov/docs/suzaku/analysis/gso_mkf.sel
```

次に xselect でイベントを抽出。PIN の場合を下記に記す。

```
unix% xselect
> Enter session name >[xsel] xsel
xsel > read event
xsel > .
xsel > ae101005040hxd_0_wel_uf2.evt <---- まずはイベント読み込み

xsel > select mkf @pin_mkf.sel      <---- GSO なら gso_mkf.sel を使用
xsel > .                            ae101005040.mkf を読んでくれる。

xsel > filter time file ae101005040hxd_0_tel_uf.gti.gz
                                     <---- テレメトリ飽和のない時間帯を選別
xsel > filter column                 <---- event FITS 内のカラムで選別
xsel > "DET_TYPE=1:1"                <---- GSO なら "DET_TYPE=0:0"

xsel > show filter                    <----- フィルター条件を確認
xsel > extract event                 <===== 抽出
xsel > save event                    <===== 保存
xsel > ae101005040hxd_0_pinno_cl2.evt
xsel > yes
xsel > exit
```

これで、cleaned event が完成。ただし、PIN が GSO かを示す 'DETNAM' というキーワードが WEL のままになっているので、抽出したイベント種別に応じ、下記のようにヘッダ調整。(GSO なら value="WELL_GSO" を使用のこと)

```
unix% fparkey fitsfile="ae101005040hxd_0_pinno_cl2.evt+0" \
      value="WELL_PIN" keyword="DETNAM" comm="detector name" add=no
unix% fparkey fitsfile="ae101005040hxd_0_pinno_cl2.evt+1" \
      value="WELL_PIN" keyword="DETNAM" comm="detector name" add=no
```

また、hxdtime がたまに TIMEDEL の計算を誤り、"-999" という値が入る事がある。TIMEDEL = -999 だと、時系列解析ツール (Xronos; see §7) がうまく動作しない⁸。そこで、

```
unix% fparkey fitsfile="ae101005040hxd_0_pinno_cl2.evt+0" \
      value="6.1E-05" keyword="TIMEDEL" \
      comm="finest time resolution (time between frames)" add=no
unix% fparkey fitsfile="ae101005040hxd_0_pinno_cl2.evt+1" \
      value="6.1E-05" keyword="TIMEDEL" \
      comm="finest time resolution (time between frames)" add=no
```

のようにヘッダを調整する。以上で、event_cl2/ ディレクトリに、ae101005040hxd_0_pinno_cl2.evt と ae101005040hxd_0_gsono_cl2.evt という cleaned event が完成。

⁸ うまく動作しないどころか、不良動作に気づかず、間違った結果を出す！最悪

4.4.3 GTIのマージ (Merge GTI)

バックグラウンドは、長期間にわたる夜地球のデータベースから作成した Non X-ray Background (NXB) イベントファイルであり、通常の観測データと同様に `xselect` 等でスペクトルを抽出することができる。ただし、HXD のバックグラウンドは、時間的に変動するため、観測データと NXB データとで、スペクトルを作成する時間帯 (GTI) を完全に一致させる必要がある。各々の GTI は、3 番目の Extension (see §3.1) に書かれており、`mgtime` というツールを用いて、

```
unix% mgtime \  
  ingtis="ae101005040hxd_0_pinno_cl2.evt+2,ae101005040hxd_0_pinnox.evt.gz+2" \  
  outgti="ae101005040hxd_wel_pin.gti" merge="AND"
```

といった感じに、両者をマージした GTI を作成しないとイケない。

4.4.4 スペクトルの抽出 (Extract spectral fits)

スペクトルの抽出は、XIS と同じである。ただし、スペクトルで使用する 波高値 PI の名前が、PLPIN (PIN), PLSLOW (GSO) と異なるので注意が必要である。

PIN HV=400V のユニットを解析から外すなど、PIN 毎にスペクトルを確認したい場合は、下記のように `fselect` したものを使用すべし。(第 4.4.1-d 節の通り V2.x では 全 PIN 解析可能)

- 2006 May 24 – 2006 Oct 3
`fselect infile=ae101005040hxd_0_pinno_cl2.evt \
 outfile=ae101005040hxd_0_pinno_cl2_w123.evt \
 expr='UNITID>3'`
- 2006 Oct 3 –
`fselect infile=ae101005040hxd_0_pinno_cl2.evt \
 outfile=ae101005040hxd_0_pinno_cl2_w23.evt \
 expr='UNITID>7'`
`xselect` 中で `filter column "UNITID=8:15"` もしくは、`select event "UNITID=8:15"` というやりかたもある。" " が無いと正しく動作しないので注意。

```
unix% xselect
```

```
** XSELECT V2.3 **
```

```
> Enter session name >[xsel] psr1509src  
(イベントを読みこむ; 前のセクションで作成した cleaned event)  
(session name は何でもよい。ここでは 一例として psr1509src とする)
```

```
psr1509src:SUZAKU > read event  
> Enter the Event file dir >[.] .  
> Enter Event file list >[] ae101005040hxd_0_pinno_cl2.evt
```

```
Notes: XSELECT set up for      SUZAKU  
Time keyword is TIME          in units of s  
Default timing binsize =     16.000
```

```
Setting...
```

```

Image keywords = UNITID      UNITID      with binning = 1
WMAP keywords  = UNITID      PIN_ID     with binning = 1
Energy keyword  = PI_SLOW                    with binning = 1

```

Getting Min and Max for Energy Column...

Got min and max for PI_SLOW: 0 511

could not get minimum time resolution of the data read

MJDREF = 5.1544000742870E+04 with TIMESYS = TT

Number of files read in: 1

***** Observation Catalogue *****

Data Directory is: /home/terada/suzaku/ana/psr1509/060405_rev07/aaa/

HK Directory is: /home/terada/suzaku/ana/psr1509/060405_rev07/aaa/

	OBJECT	DATE-OBS	TIME-OBS	DATE-END	TIME-END
1	PSR1509-58	2005-08-23	08:26:15	2005-08-24	20:37:57

(次に 4.4.3 章で作成した GTI でフィルター)

psr1509src:SUZAKU-HXD-WELL > filter time file

> Enter list of GTI files >[] ae101005040hxd_wel_pin.gti

(フィルタがかかったかをチェック)

psr1509src:SUZAKU-HXD-WELL > show filter

*** SELECTIONS ***

NONE

*** FILTERS ***

Fits timing files have been read from:

ae101005040hxd_wel_pin.gti

(ライトカーブをチェック)

psr1509src:SUZAKU-HXD-WELL > extract curve

extractor v4.61 28 Nov 2005

No image X-axis TCRPX, set to 0

No image Y-axis TCRPX, set to 0

No image X-axis TCRVL, set to 0

No image Y-axis TCRVL, set to 0

No image X-axis TCDLT, set to 1

No image Y-axis TCDLT, set to 1

Doing file: /home/terada/suzaku/ana/psr1509/060405_rev07/aaa/evt/ae20050823_0826_2038_hxd_wel_pin_cl.e

100% completed

Total	Good	Bad: Region	Time	Phase	Grade	Cut
50788	50788	0	0	0	0	0

```

=====
Grand Total      Good  Bad: Region    Time  Phase  Grade  Cut
50788            50788          0      0      0      0      0

```

```

in 44581.      seconds
Fits light curve has 50788 counts for 1.139 counts/sec
Thresholding removed significant counts
Try exposure=0.0 on the extract command in xselect
or lcthresh=0.0 if running extractor stand-alone
Thresholding removed more than half the bins
Try exposure=0.0 on the extract command in xselect
or lcthresh=0.0 if running extractor stand-alone
psr1509src:SUZAKU-HXD-WELL > save cur
> Give output file name >[psr1509_pin_evtb] psr1509_pin_evtb
> File already exists, overwrite it? >[yes]
Wrote FITS light curve to file psr1509_pin_evtb.lc

```

(最後に、スペクトルを作成し保存する)

```

psr1509src:SUZAKU-HXD-WELL > set PHANAME PI_PIN
Getting Min and Max for Energy Column...
Got min and max for PI_PIN:      0      255
                                (GSO の場合は set PHANAME PI_SLOW です)

```

(HEADAS 6.1 では不用)

```

psr1509src:SUZAKU-HXD-WELL > extract spec
extractor v4.61      28 Nov 2005
No image X-axis TCRPX, set to 0
No image Y-axis TCRPX, set to 0
No image X-axis TCRVL, set to 0
No image Y-axis TCRVL, set to 0
No image X-axis TCDLT, set to 1
No image Y-axis TCDLT, set to 1
No wmap X-axis TCRPX, set to 0
No wmap Y-axis TCRPX, set to 0
No wmap X-axis TCRVL, set to 0
No wmap Y-axis TCRVL, set to 0
No wmap X-axis TCDLT, set to 1
No wmap Y-axis TCDLT, set to 1
Doing file: /home/terada/suzaku/ana/psr1509/060405_rev07/aaa/evt/ae101005040hxd_0_pinno_c12.evt
100% completed

```

Total	Good	Bad: Region	Time	Phase	Grade	Cut
50788	50788	0	0	0	0	0

```

Grand Total      Good  Bad: Region      Time      Phase      Grade      Cut
50788      50788      0      0      0      0      0
in 44581.      seconds
Spectrum      has 50788 counts for 1.139 counts/sec
... written the PHA data Extension

```

```

psr1509src:SUZAKU-HXD-WELL > save spec
> Give output file name >[psr1509_pin_evtb]
> File already exists, overwrite it? >[yes]
Wrote spectrum to psr1509_pin_evtb.pha
psr1509src:SUZAKU-HXD-WELL > exit
> Save this session? >[no]

```

同じ事を、GSO event, PIN background, GSO background について、合計 4 種、行なう。以後、PI file は、それぞれ、

```

psr1509_gso_evtb.pha
psr1509_pin_evtb.pha
psr1509_gso_bgdb.pha
psr1509_pin_bgdb.pha

```

で保存したとする。

4.4.5 デッドタイム補正 (Dead time correction)

検出器には、不感時間 (dead time) が存在するため、正確な X 線フラックスを見積もるためには、これを補正する必要がある。HXD では、センサーからの信号でトリガーされるイベントとは別に、デッドタイム補正用に、エレクトロニクスで、周期的に擬似イベント (pseudo event) を発生させている。この擬似イベントは、実際の信号による不感時間中にトリガーされた場合、テレメトリにでてこないもので、発生させたであろうイベント数と、実際にテレメトリにでたイベント数との比を取る事で、その観測で不感時間が占める割り合いを算出している。Dead time 補正用のツールでは、この不感時間の割り合いに応じて⁹、PI file の Exposure を書き変えている。

以下の要領で、補正ツール `hxddtcor` を、観測 PI file 毎にかける。PIN Background に関しては、たいていは dead time 補正されたデータベースであるので、dead time 補正は行なわないことが多い。dead time 補正が必要な NXB event か否かは、ダウンロードした WWW page をよくよく確認のこと。これを間違えると大変!

```

unix% cp psr1509_gso_evtb.pha psr1509_gso_evtb_dtcor.pha
unix% hxddtcor event_fname="ae101005040hxd_0_wel_uf2.evt" \
      pi_fname="psr1509_gso_evtb_dtcor.pha" \
      save_pseudo=no chatter=2

```

```

unix% cp psr1509_pin_evtb.pha psr1509_pin_evtb_dtcor.pha
unix% hxddtcor event_fname="ae101005040hxd_0_wel_uf2.evt" \
      pi_fname="psr1509_pin_evtb_dtcor.pha" \
      save_pseudo=no chatter=2

```

Version 2.0 以降は、Pseudo event だけを抽出したファイルを用意しているので、上記の、
`event_fname="ae101005040hxd_0_wel_uf2.evt"`
の部分は、

```
event_fname="ae101005040hxd_0_pse_cl.evt.gz"
```

で代用しても OK である。

⁹hxddtcor では、エレクトロニクスの不感時間の割合に加え、GRADE による event セレクションの効率も含めて EXPOSURE を補正している。DET_TYPE =2 で用意されている Pseudo event FITS (aeYYYYMMDDhxd_0_pse_cl.evt.gz) は、純粋に UFF から Pseudo event だけを抽出したのではなく、cleaned criteria (see §4.4.2-b) もかかった Pseudo event である。

```

HEADAS version 6.1.3 以降に含まれる hxdtdcor は、複数の pi file, event file の入力に対応している。
たとえば、event_uf/ 中の uf event が下記のように複数ある場合、
ae101005040hxd_1_wel_uf.evt
ae101005040hxd_2_wel_uf.evt
ae101005040hxd_3_wel_uf.evt
たとえば、ae101005040_uf.list というファイルに上記のファイル名を列挙しておき、
unix% hxdtdcor event_fname="@ae101005040_uf.list" \
      pi_fname="psr1509_gso_evtb_dtdcor.pha" \
      save_pseudo=no chatter=2
とすると 複数の uf file を読みこんでくれる。

```

先の hxdtdcor のパラメータ 'chatter' の値を 0 以外にしておく、プロセスの状態を表示してくれるので、何% 程度、dead time 補正されたかがわかるであろう。バックグラウンドメインな観測の場合、典型的に 5% 程度である。

念には念を入れたい場合は、fkeyprint で EXPOSURE キーワードの値をダンプして確認する。

```

unix% fkeyprint infile="psr1509_gso_evtb.pha" keynam="EXPOSURE"
# FILE: psr1509_gso_evtb.pha
# KEYNAME: EXPOSURE
# EXTENSION: 0
EXPOSURE= 4.448161003002524E+04 / Exposure time
# EXTENSION: 1
EXPOSURE= 4.448161003002524E+04 / Exposure time
# EXTENSION: 2
EXPOSURE= 4.448161003002524E+04 / Exposure time
hxdfaq/hxd_dtdcor_lc.html
unix% fkeyprint infile="psr1509_gso_evtb_dtdcor.pha" keynam="EXPOSURE"
# FILE: psr1509_gso_evtb_dtdcor.pha
# KEYNAME: EXPOSURE
# EXTENSION: 0
EXPOSURE= 42773.250000 / Exposure time
# EXTENSION: 1
EXPOSURE= 42773.250000 / Exposure time
# EXTENSION: 2
EXPOSURE= 42773.250000 / Exposure time

```

割り算すると、live time が 96.2 % と求まるので、大間違いはなからう

(注意)
HXD の event FITS から lcurve (see section7) でライトカーブを描いたとしても、それは、デッドタイム込みのライトカーブであることに注意。デッドタイムはカウントレートに依存した値をとるので、デッドタイム補正したライトカーブを描く場合は、各時間ビンで補正する必要がある。つまり、

1. event_uf/aeXXXXXXXXXhxd_0_wel_uf.evt.gz から Pseudo event の抜き出し
これは hxdtdcor で save_pseud=yes オプションをつけることができる。
2. Pseudo event を Clean と同じ criteria でセレクションし抽出。(fselect)
version 2.x では aeYYYYMMDDhxd_0_pse_cl.evt.gz として、step-2 まで既に用意されている。
3. Pseudo のライトカーブ作成。標準で 0.25 Hz で Pseudo event が発生している、Pusedo 値/4.0 がデッドタイムとなる。
4. イベントのライトカーブを、(Pusedo/4.0) のライトカーブで割り算する。

というステップが必要となる。これらの詳細は、HXD の FAQ ページ、
http://www.astro.isas.jaxa.jp/suzaku/analysis/hxd/hxdfaq/hxd_dtdcor_lc.html
に記載されている。

4.4.6 PIN バックグラウンドの積分時間補正 (Scale down of PIN BGD flux)

PIN のバックグラウンドファイルに関しては、統計負けしないように、10 倍の flux でイベントファイルを作成している。そのため、出来た BGD の PI file の Exposure を 10 倍して、BGD の flux を 0.1 倍にしないとイケない。下記のように fparkey で書き変える。

```
unix% cp psr1509_pin_bgdb_dtcor.pha psr1509_pin_bgdb_expcor.pha
unix% fkeyprint infile="psr1509_pin_bgdb_expcor.pha" keynam="EXPOSURE"
# FILE: psr1509_pin_bgdb_expcor.pha
# KEYNAME: EXPOSURE
# EXTENSION: 0
EXPOSURE= 4.448161003002524E+04 / Exposure time
# EXTENSION: 1
EXPOSURE= 4.448161003002524E+04 / Exposure time
# EXTENSION: 2
EXPOSURE= 4.448161003002524E+04 / Exposure time
unix% fparkey
Keyword value[] 4.448161003002524E+05
Name of FITS file and [ext#] [] psr1509_pin_bgdb_expcor.pha+0
Keyword name[EXPOSURE]
unix% fparkey
Keyword value[] 4.448161003002524E+05
Name of FITS file and [ext#] [] psr1509_pin_bgdb_expcor.pha+1
Keyword name[EXPOSURE]
unix% fparkey
Keyword value[] 4.448161003002524E+05
Name of FITS file and [ext#] [] psr1509_pin_bgdb_expcor.pha+2
Keyword name[EXPOSURE]
```

4.4.7 いざスペクトル解析! (Let's enjoy spectral analyses!)

以上にて xspec に使える PI file が完成した。あとは、第 8 章などを見て楽しむべし。

GSO のスペクトルを描く場合は NXB model と同じビンまとめをしておく必要がある。やりかたは下記のとおり。

```
unix% grppha ae101005040hxd_gsono_cl2.pha ae101005040hxd_gsono_cl2bin.pha
GRPPHA[] group 0 24 25 25 26 2 27 28 2 29 31 3 32 35 4 36 38 3
GRPPHA[] group 39 42 4 43 46 4 47 51 5 52 56 5 57 62 6
GRPPHA[] group 63 68 6 69 75 7 76 83 8 84 91 8 92 100 9
GRPPHA[] group 101 110 10 111 121 11 122 134 13 135 147 13 148 162 15
GRPPHA[] group 163 178 16 179 196 18 197 216 20 217 238 22 239 262 24
GRPPHA[] group 263 288 26 289 317 29 318 349 32 350 384 35 385 422 38
GRPPHA[] group 423 465 43 466 511 46
GRPPHA[] exit
```


4.5 より詳細なデータスクリーニングをしたい時 (Further Analysis)

cleaned event からではなく、もう少し screening を自分でやってデータを作成したい時も、XSELECT が活用できる。この様な時、XXX/event_uf/ (XXX=XIS, HXD) の下にある event file から自分で screening を行なう。

4.5.1 (XIS/HXD) SAA, ELV, COR など hk 情報での screening (Screening with HK data (SAA, ELV, COR etc.))

XIS の場合 (XIS case)

SAA, ELV などの情報は、auxil/ae001122330.mkf に書かれているので、あらかじめ gunzip しておこう。XSELECT 内で、

```
xsel:SUZAKU-XIS0-STANDARD > select mkf "ELV>5 && DYE_ELV>20 && SAA==0 && T_SAA>436"  
(mkf file のディレクトリを聞かれたら、ディレクトリ名を答える)  
xsel:SUZAKU-XIS0-STANDARD > extract event  
xsel:SUZAKU-XIS0-STANDARD > save event
```

とすると、cleaned event の標準的な data reduction の一部を行なったことになる。

ELV, DYE_ELV, COR などは、観測毎、あなたのやりたい解析によって、最適な条件が違いうる。ぜひいろいろ試して最適な条件を見つけて欲しい。

HXD の場合 (HXD case)

HXD のデータを再プロセスしたい場合は、§4.4.2 に示したように行なえばよい。ここで、COR の条件などを変えたい場合は、Event データだけでなく、background もそれにあわせて変える必要があることに注意する。

4.5.2 (XIS/HXD) telemetry saturation の screening (Screenings with telemetry saturation time)

XIS の場合 (XIS case)

XIS の場合、観測している天体が明るすぎる時や、週末運用で data rate を落としている時は、telemetry saturation とよばれる現象が起こることがある。XIS ではデータ量が telemetry limit を越えた時点で当該フレームのデータ転送は中断されるので、saturation の起こったフレームは CCD の Segment の一部を欠いた不完全なデータとなる。

XIS データから telemetry saturation の起こった時間帯を除去するための GTI フィルタ¹⁰ は XIS チームにより用意されている。スクリーニングをしたい event file に対応したセンサーおよび edit mode の GTI ファイルをカレントディレクトリにリンクまたはコピーする。XSELECT 内で、

```
xsel:SUZAKU-XISO-STANDARD > filter time file gti_xis0_3x3.txt
xsel:SUZAKU-XISO-STANDARD > extract event
xsel:SUZAKU-XISO-STANDARD > save event
```

で、gti file が適用されたイベントを作成できる。

HXD の場合 (HXD case)

HXD では、HXD HK file をもとに、テレメトリーが飽和した時間帯を計算するツール *hxdgtigen* がある。テレメトリー飽和がない GTI を作成するには、

```
hxdgtigen hk_dir="./hk" hk_file="ae100006010hxd_0.hk" \
          gti_fname="ae100006010hxd_0_tlm.gti"
```

のようにすればよく、その後、xselect でフィルターするとよい¹¹

```
xsel > filter time file ae100006010hxd_0_tlm.gti
```

なお、default の cleaned event では、このようなテレメトリー飽和は除外されている。

¹⁰2006/08/01 現在の入手先は以下 (京都大学 山口弘悦氏の homepage)。

<http://www-cr.scphys.kyoto-u.ac.jp/member/hiroya/xis/gtifile.html>

¹¹PIN HV 400V への対応は、HEADS v6.1.3 以降に含まれる hxdgtigen を使用して、

```
./hxdgtigen hk_dir="." hk_file="ae101005040hxd_0.hk.gz" \
          gti_fname="hxdgtigen.gti" WPU="0123"
```

といった感じで WPU オプションをつけて撰択する。

4.5.3 (XIS) cleansis による screening (Screenings with cleansis command)

XIS の event file 中にある hot pixel や flickering pixel を取り除くためには、update version の cleansis を実行すればよい。

```
unix% punlearn cleansis
unix% cleansis chipcol=SEGMENT
CLEANSIS_V1.7
Name of SIS file to be cleaned (input) [] ae001122330xi0_0_5x5n000a_cl.evt
Name for cleaned event file (output) [] ae001122330xi0_0_5x5n000a_cl2.evt
Poisson clean cell size (odd integer > 1; e.g. 5) [5]
Log Poisson probability threshold (e.g. -5.3) [-5.3]
Zero background threshold ( > ) [3]
Minimum PHA value for clean (inclusive) [0]
Maximum PHA value for clean (inclusive) [4095]
```

これで ae001122330xi0_0_5x5n000a_cl2.evt という名前の hot pixel などの除去された event file ができる。

4.5.4 (HXD) 地食を用いた NXB model のチェック (Check NXB model with earth occulted data)

衛星が地球を見ている間は、Cosmic X-ray Background (CXB) も天体からの Signal もない Non X-ray Background (NXB) だけが検出される時間帯である。この地食中の時間帯を用いて、HXD 検出器チームが供給する NXB model の検証をおこなうことができる。

地食中のイベントは、unscreened event FITS を用いて、xselect で下記のように作成することができる。

```
xsel > read event
xsel > ./
xsel > ae101005040hxd_0_wel_uf2.evt

xsel > read hk
xsel > .
xsel > ../../auxil/ae101005040.mkf.gz
xsel > yes

xsel > select hk
xsel > ELV<-5 && COR>8 && SAA_HXD == 0 && T_SAA_HXD > 500 && ANG_DIST < 1.5 && HXD_DTRATE < 3
&& AOCU_HK_CNT3_NML_P==1 && HXD_HV_WO_CAL>700 && HXD_HV_TO_CAL>700 && HXD_HV_W1_CAL>700 &&
HXD_HV_T1_CAL>700 && HXD_HV_W2_CAL>700 && HXD_HV_T2_CAL>700 && HXD_HV_W3_CAL>700
&& HXD_HV_T3_CAL>700

xsel > extract events
xsel > save event ae101005040hxd_wel_pin_nxb.evt

xsel > exit
```

これ以降は、4.4.3 章から後の手順を行えばよい。NXB Event も、地食イベントとマージした GTI で切りだせば、地食と NXB model のスペクトルを比較することができる。

第5章 XSELECT

XSELECT は、event ファイルに様々なフィルタをかけて、ライトカーブ・イメージ・スペクトルを抽出することを主な目的としているソフトウェアである。コマンドラインインタフェースによって、内部で FTOOLS を動かしながら必要なタスクを行う。XSELECT が備える機能は以下の 4 つに大別される。

- 入力として event FITS を読み込み Observation Catalogue を作成する。
- GTI(Good Time Interval) を作成し、もしくは読み込んで、目的に応じた時間帯のイベントを抽出する。
- 入力した event FITS に様々なフィルタ (GTI, PHA, region, grade など) をかける。
- フィルタをかけた event、イメージ、ライトカーブ、および スペクトルファイルを出力する。

本章では、「すざく」の実際のデータ解析に即した XSELECT の使用法を述べる。本章の目標は、XSELECT を通じて「すざく」の cleaned event file から自分の目的に応じたライトカーブ、イメージ (XIS のみ) およびスペクトルを一通り作成できるようになることである。

XSELECT の詳細なマニュアルについては以下を参照してほしい。

<http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/ftools/xselect/xselect.html>

また、shell のコマンドライン上で

```
$ fhhelp xselect
```

あるいは XSELECT 上で

```
xsel> help ?
```

とするとヘルプを参照することができる。なお、以下の記述は XSELECT Version 2.3 に基づいている。

5.1 初期設定 (Initial Settings)

5.1.1 XSELECT の立ち上げ (Start XSELECT)

コマンドラインから

```
$ xselect
```

と入力すると、XSELECT が立ち上がり、以下のような表示とプロンプトが表示される。

```
** XSELECT V2.3 **
```

```
> Enter session name >[xsel]
```

ここで入力されている session name は、XSELECT がカレントディレクトリに作成する一時ファイル名の先頭に使われる。これらは XSELECT 終了と同時に消去されるが、session を途中でセーブしたいときは、ターゲット名など、後で見ても分かりやすい名前を付けるとよい。

ここでは GC_SRC1(銀河中心)の観測を例とする。

```
> Enter session name >[xsel] GC_SRC1
```

ちなみに、XSELECT のコマンドラインでは、[] の中はデフォルト値を示す。プロンプトは以下のようになる。

```
GC_SRC1:SUZAKU >
```

5.1.2 event file の読み込み (Read event files)

カレントディレクトリにある event file を読み込むには以下のように書く。

```
GC_SRC1:SUZAKU > read event ae100027010xi0_0_3x3n000_cl.evt
> Enter the Event file dir >[./] ./
```

正しく読み込まれると、続いて以下のような表示がされる。

```
Notes: XSELECT set up for      SUZAKU
Time keyword is TIME          in units of s
Default timing binsize =     16.000
```

Setting...

```
Image keywords   = X          Y          with binning =    8
WMAP keywords    = X          Y          with binning =    8
Energy keyword   = PI                               with binning =    1
```

Getting Min and Max for Energy Column...

```
Got min and max for PI:      0  4095
```

```
Got the minimum time resolution of the read data:  8.0000
```

```
MJDREF = 5.1544000742870E+04 with TIMESYS = TT
```

```
Number of files read in:  1
```

```
***** Observation Catalogue *****
```

```
Data Directory is: /YOUR_ANALYSIS_DIRECTORY/xis/
```

```
HK Directory is: /YOUR_ANALYSIS_DIRECTORY/xis/
```

	OBJECT	DATE-OBS	TIME-OBS	DATE-END	TIME-END	EDITMODE
1	GC_SRC1	2005-09-23	07:16:30	2005-09-24	10:39:58	3x3

複数の event ファイルを読み込ませたい場合、たとえば 同じターゲットの 2 つの異なる時期の観測について、それぞれの 3x3 mode と 5x5 mode の XIS のデータを読み込ませたい場合は、以下のように書くことができる。

```
GC_SRC1:SUZAKU > read event "ae100027010xi0_0_3x3n000_cl.evt
ae100027010xi0_0_5x5n000_cl.evt ae100037040xi0_0_3x3n000_cl.evt
ae100037040xi0_0_5x5n000_cl.evt"
```

あるいは上記 event ファイルを列挙したファイル (ここでは evt.lis) をあらかじめ用意しておき、これを読み込ませる方法もある。

```
GC_SRC1:SUZAKU > read event "@evt.lis"
```

ここでは以下のような evt.lis を読み込ませた。

```
$ cat evt.lis
ae100027010xi0_0_3x3n000_cl.evt
ae100027010xi0_0_5x5n000_cl.evt
ae100037040xi0_0_3x3n000_cl.evt
ae100037040xi0_0_5x5n000_cl.evt
```

ただし、異なる INSTRUMENT の event files を同時に読み込ませることは出来ないので注意。

5.1.3 XSELECT で使用する変数 (Variables in XSELECT)

XSELECT で使用する変数の一覧は以下の通り。

```
GC_SRC1:SUZAKU > set
BINSIZE          --> set the binsize for light curves
DATA*DIR         --> set the data directory
DATAMODE         --> setup Xselect for particular data modes
DEVICE           --> set the plot device
DUMPCAT          --> set make obscat to display or not
IMAGE            --> set the coordinates for the image to sky or det.
INSTRUMENT       --> set the instrument name
MISSION          --> set the mission name
MKFDIR           --> set the directory for the MKF file
OBSDIR           --> set the observation catalogue directory (default is pwd)
PHAREBIN         --> set the pha bin size
PAGEWIDTH        --> set the pagewidth for show commands
PHANAME          --> set the column name for PHA
QUIT             --> quit set
WMAPNAME         --> set the column for the WMAP
WMAPBINSIZE      --> set the bin size for the weighted map
XYBINSIZE        --> set the bin size for the image
XYNAME           --> sets the column for images
XYCENTER         --> set the center for the image
XYSIZE           --> sets the size of the image
```

「すざく」のデータを取り扱う上でとくに注意すべき変数は以下である。

BINSIZE	出力ライトカーブの binsize。単位は sec。default は 16.0。
INSTRUMENT	検出器名。すざくでは XIS0, XIS1, XIS2, XIS3 および HXD のいずれか。
PHAREBIN	出力スペクトルファイルの bin まとめ数。default は 1。
XYNAME	使用する座標軸。X Y で 天空座標を、DETX DETY で検出器座標をそれぞれ指定。default は X Y。
XYBINSIZE	出力イメージの bin まとめするサイズ。単位は pixel。default は 8。

各変数がどのような値をとっているかは、

```
GC_SRC1:SUZAKU-XIS0-STANDARD > show status
```

の以下の行で確認できる。

```
*** MISSION ***

SUZAKU NONE XIS0 STANDARD                                MISSION INSTRUMENT DATAMODE
Time keyword is TIME          in units of s
Default timing binsize =      16.000                      BINSIZE
Minimum timing resolution:     8.0000
Energy keyword is PI           with binning      1        PHAREBIN
Max and min for PI           :      0    4095
Image keywords = DETX          DETY              with binning = 8    XYNAME XYBINSIZE
WMAP keywords = X             Y                  with binning = 8
```

各変数に default と異なる値を指定したい場合は以下のように書く。

```
GC_SRC1:SUZAKU > set binsize 64.0
```

例えば XIS の各 sensor 上で同じ位置からスペクトルを取得したい場合は、天空座標ではなく検出器座標でリージョンを指定する必要があるので、以下のように設定する。

```
GC_SRC1:SUZAKU > set xynome DETX DETY
```

5.2 ライトカーブの作成 (Generate Light Curves)

events file を読み込んでまず確認すべきことは、観測全体のライトカーブである。観測中、検出器に何らかの異常が起こっていないか、太陽活動起源のフレアアップが起きていないか、あるいは突発天体がバーストを起こしていないか、などの情報をライトカーブから得ることができる。以下では、XIS のデータを例に説明をする。

5.2.1 ライトカーブの抽出 (Extract light curves)

ライトカーブを抽出するには extract コマンド¹を用いる。

```
GC_SRC1:SUZAKU-XISO-STANDARD > extract curve
```

...(中略)...

```
      Total      Good  Bad: Region      Time      Phase      Grade      Cut
      126063     126063           0           0           0           0           0
=====
Grand Total      Good  Bad: Region      Time      Phase      Grade      Cut
      126063     126063           0           0           0           0           0
in 37052. seconds
Fits light curve has 126063 counts for 3.402 counts/sec
```

今は特にフィルタリングをかけていないので、すべてが Good と判定されている。つまり、読み込んだイベントファイル中の全 event のライトカーブが作成されたことを示している。さっそく見てみる。

```
GC_SRC1:SUZAKU-XISO-STANDARD > plot curve
```

とすると内部で qdp が呼び出され、作成されたライトカーブがディスプレイに表示される。横軸は観測開始時刻からの経過時間 (sec)、縦軸はカウントレート (counts/s)。ライトカーブでとところどころデータが抜けているのは、cleaned event になっている時点で、地没や SAA の 時間帯があらかじめ取り除かれているためである。

5.2.2 エネルギーバンドの指定 (Filter energy bands)

切り出すエネルギーバンドの指定は PI で行う。

PI とエネルギーの関係はそれぞれの検出器で以下の通りである。

XIS: $E = 3.65PI$ [eV]

HXD-PIN: $E = 0.375(PI_{PIN} + 1.0)$ [keV]

HXD-GSO: $E = 2(PI_{SLOW} + 0.5)$ [keV]

例えば XIS で 2.0–6.0 keV のエネルギーバンドの event を切り出したい場合は以下のように書く。

¹旧 bin コマンド。extract is the new name for the old bin command. (本家マニュアル抜粋)

```
xsel:SUZAKU-XISO-STANDARD > filter pha_cutoff 547 1644
```

PI channel は幅をもっているために、隣接する 2 つのエネルギーバンドで切りたい場合は、PI channel に重複がないように指定する。たとえば、HXD-PIN で 15-40 keV と 40-70 keV の二つのエネルギーバンドを切りたい場合は、それぞれ

```
filter pha_cutoff 40 106
filter pha_cutoff 107 186
```

もしくは、

```
filter pha_cutoff 40 107
filter pha_cutoff 108 186
```

とする。もし 2 つのバンドの境目で同じ PI channel(上記の例では 107 ch) を指定すると、PI=107 のエネルギーに対応するイベントは、両方のエネルギーバンドに含まれてしまう。特に 1 ch の幅が大きい HXD の解析の場合には注意が必要である。

filter をクリアしたい場合は、

```
clear pha_cutoff
```

でよい。

5.2.3 タイムリージョンの指定 (Filter time regions)

XSELECT では以下の方法で任意のタイムリージョンを指定して event を切り出すことができる。

- | | |
|------------|-------------------------|
| (1) CURSOR | --> タイムリージョンをマウスで指定 |
| (2) FILE | --> GTI ファイルを読み込ませて指定 |
| (3) MJD | --> MJD time で指定 |
| (4) SCC | --> SpaceCraft time で指定 |
| (5) UT | --> UT で指定 |

本節では (1) による方法を述べる。まず、extract curve コマンドによってライトカーブが抽出されている状態で、

```
GC_SRC1:SUZAKU-XISO-STANDARD > filter time cursor
```

とすると内部で qdp が立ち上がり、現時点で抽出されているライトカーブがディスプレイに表示される。qdp プロンプトで

```
PLT> quit
```

とすると、マウスでタイムリージョンを指定できる状態になる。表示されているライトカーブの上にカーソルを持っていき、start time にしたい場所でクリックをすると、ライトカーブ上方に緑の十字印が表示される。続いて end time にしたいところでクリックすると、start-end time 間に白い線分がひかれる。この線分の範囲内の時間帯がタイムリージョンとして選ばれたことを示している。複数のタイムリージョンを指定したい場合は、上の動作を繰り返せばよい。指定したタイムリージョンの間に不要な時間帯がある場合は、除きたい時間の start / end time にカーソルを合わせてそれぞれ “e” を押すと、この間の時間帯はタイムリージョンから削除される。その他のコマンドは以下の通り。

“p” qdp のプロンプトに戻る。

“c” 指定したタイムリージョンをキャンセルして XSELECT プロンプトに戻る。

“x” 指定したタイムリージョンをセーブして XSELECT プロンプトに戻る。

“x” で XSELECT プロンプトに戻った後に extract すればいま指定したタイムリージョンのフィルタをかけた状態でイベントを抽出することができる。なお、タイムリージョンは GC_SRC1.cursor-gti001.xsl の

名前でセーブされているが (GC_SRC1 は初めに指定したセッション名)、XSELECT 終了時にセッションをセーブしない場合はこのファイルは消去されてしまう。このため、タイムリージョンを残しておきたい場合は以下のようにして別名で保存しておくとい。

```
GC_SRC1:SUZAKU-XISO-STANDARD > save time cursor good <-- FITS 形式で保存
GC_SRC1:SUZAKU-XISO-STANDARD > save goodtime good <-- ASCII 形式で保存
```

それぞれ good.curs_gti (FITS)、 good.gti (ASCII) の名前で保存される。これらは適宜、別のセッションでも

```
GC_SRC1:SUZAKU-XISO-STANDARD > filter time file good.gti
```

などとして、GTI フィルタとして読み込ませることができる。読み込んでいるフィルタを clear したい場合は

```
GC_SRC1:SUZAKU-XISO-STANDARD > clear time cursor
```

とする。GTI ファイルとして読んでいるフィルタは

```
GC_SRC1:SUZAKU-XISO-STANDARD > clear time file good.gti
```

などとすれば clear できる。

5.2.4 作成したライトカーブの保存 (Save light curves)

上記のようにして作成したライトカーブを FITS 形式で保存するには

```
GC_SRC1:SUZAKU-XISO-STANDARD > save curve lc_xis0_3x3.fits
```

とすればよい。ファイル名は自分の好きなように付けることができるが、あとで見ても分かるような名前を付けるとよい。ライトカーブを qdp 形式で保存したい場合には

```
GC_SRC1:SUZAKU-XISO-STANDARD > plot curve
PLT> wdata lc_xis0_3x3
PLT> whead lc_xis0_3x3
```

とすると、lc_xis0_3x3.qdp と lc_xis0_3x3.pco が作成される。

```
PLT> wenviron lc_xis0_3x3
```

で一度に作成することもできる。

5.3 XIS イメージの作成 (Generate XIS Images)

本節では XIS イメージを作成する方法について述べる。イメージを作成するうえで注意すべき変数は XYNAME、XYBINSIZE の 2 つである。

5.3.1 タイムリージョン・エネルギーバンドの指定 (Filter time regions/energy bands)

ある特定のタイムリージョンや、特定のエネルギーバンドのイメージを作成したい場合は、それぞれに応じたフィルタをかける必要がある。

タイムリージョンとして、たとえば前節で作成した GTI を指定する場合は、

```
xsel:SUZAKU-XISO-STANDARD > filter time file good.gti
```

でよい。一方で、特定のエネルギーバンドで切りたい場合は、切り出すエネルギーを PI で指定したフィルターをかける。PI と エネルギーの関係については、第 5.2.2 節を参照。例えば 2.0–6.0 keV のエネルギーバンドのみを切り出したい場合は以下のように書く。

```
xsel:SUZAKU-XISO-STANDARD > filter pha_cutoff 547 1643
```

5.3.2 イメージの抽出・保存 (Extract/save images)

続いて以下のコマンドで、上記のフィルタをかけたイメージが抽出される。

```
GC_SRC1:SUZAKU-XISO-STANDARD > extract image
```

...(中略)...

Total	Good	Bad:	Region	Time	Phase	Grade	Cut
126063	84548	0		164	0	0	41351
=====							
Grand Total	Good	Bad:	Region	Time	Phase	Grade	Cut
126063	84548	0		164	0	0	41351
in 36936.	seconds						
Image	has	84548	counts	for 2.289		counts/sec	

Time, Cut の欄の数字は 除外されたタイムリージョンのイベント数および指定した範囲外のエネルギーを持つイベント数をそれぞれ示している。

続いて

```
xsel:SUZAKU-XISO-STANDARD > saimage
```

とすると ds9 が立ち上がり、作成したイメージを見ることができる。ds9 の詳細については 第 6 章 (ds9) を参照してほしい。

XSELECT の default の座標系は天空座標 (X, Y) なので、とくに指定しない限りイメージの XY 軸はそれぞれ (RA, Dec) に沿ったものとなる。また、イメージの bin サイズ (xybinsize) は default で 1bin = 8 pixels となっている。これらの値は 第 5.1.3 節で述べた通り set コマンドで変えることができる。

```
xsel:SUZAKU-XISO-STANDARD > save image 2060_8bin_sky.fits
```

すると、イメージが FITS 形式で保存される。ファイル名は自分があとで見ても分かるような名前を付けておくと良い。

5.4 スペクトルの作成 (Generate Spectra)

5.4.1 スペクトル取得領域の指定 (XIS) (Regions in extracting spectra)

XIS のスペクトルを作成する場合、スペクトルを切り出す領域を region file で指定する必要がある。手順は以下の通り。

1. まず 第 5.3.1 節の手順と同様に extract image で抽出したイメージを ds9 で表示させる。または 5.3.2 節の手順で保存した image FITS を ds9 で表示させる。
2. ds9 上でスペクトルを切り出したい領域の region file を作成する。ds9 については第 6 章を参照してほしい。また、region file の format に関する注意は 第 6.3.3 節にまとめられている。

3. 2. で作成した region file を xselect で読み込む。region file 名を src.reg として

```
GC_SRC1:SUZAKU-XISO-STANDARD > filter region src.reg
```

とすると、リージョンフィルタがかかる。ここで、region file の座標系・bin サイズは、それぞれ XSELECT の XYNAME・XYBINSIZE の値に一致している必要がある。

フィルタが正しい位置にかかっているかどうかを確かめるには

```
GC_SRC1:SUZAKU-XISO-STANDARD > extract image
...
GC_SRC1:SUZAKU-XISO-STANDARD > saimage
```

でイメージを見てみるとよい。

5.4.2 スペクトルの抽出・保存 (Extract/save spectra)

スペクトルを抽出するには

```
GC_SRC1:SUZAKU-XISO > extract spec
```

とする。取得したスペクトルは

```
GC_SRC1:SUZAKU-XISO-STANDARD > plot spec
```

で簡単に確かめることができるが XSELECT ではこれ以上のことはできない。以降の解析は XSPEC を用いて行う。得られたスペクトルを保存するには

```
GC_SRC1:SUZAKU-XISO-STANDARD > save spec src.pha
```

とすると、PI file として保存される。続いて以下のように「output のスペクトルを bin まとめるかどうか」を訊かれるが、binning は XSELECT 終了後に grppha などの FTOOLS を用いて行うことができるので、この時点では no で問題ない。

```
> Group ( or rebin ) the spectra before outputting? >[no] no
Wrote spectrum to src.pha
```

grppha の使い方については 8 章を参照してほしい。

5.5 その他便利な解析方法 (Tips)

5.5.1 event file の作成 (Extract filtered event files)

よく用いるフィルタ条件をかけた event file をあらかじめ作成しておくところから解析を始めれば、フィルタをかけ直す手間が省ける。

```
GC_SRC1:SUZAKU-XISO-STANDARD > extract events
```

で event file を作成できる。保存は以下。

```
GC_SRC1:SUZAKU-XISO-STANDARD > save event filtered_evt.fits
> Use filtered events as input data file ? >[yes]
```

最後のプロンプトで yes と答えると、以後のセッションでいま保存した event file を 入力 event file として読み込んでくれる。

5.5.2 ライトカーブ・イメージ・スペクトルの同時作成 (Extract light curves, images, and spectra)

```
GC_SRC1:SUZAKU-XISO-STANDARD > extract all
```

で、一度にライトカーブ・イメージ・スペクトルを抽出できる。ただし 3 つとも同一のフィルタ条件で描かれることに注意 (これが第 4 章での方法である)。

5.5.3 シェルコマンドの使用 (Using shell commands)

XSELECT 内でシェルコマンドを用いたい場合は、\$+シェルコマンド で使用することができる。

```
GC_SRC1:SUZAKU > $!s
```

など。読み込ませるファイル名が分からなくなったときなどに便利。

第6章 ds9

ds9 とは、FITS イメージや binary table などを可視化することを大きな目的としたソフトウェアで、天文学ではミッションを問わず広く用いられている。ここでは、「すざく」の解析で使いそうな ds9 テクニックを紹介する。なお、ds9 の詳細なマニュアルは、

<http://hea-www.harvard.edu/RD/ds9/>

を参照してほしい。

6.1 ds9 で出来ること (What can ds9 do ?)

ds9 で出来ることは、主に以下の通りである。

1. FITS を可視化する

event FITS, image FITS を可視化する。この目的で使うケースが一番多い。イメージは、linear/log スケール、カラーマップを調整することが出来たり、座標や領域などを書き込むことが出来る。出来たイメージを ps, jpeg にすることも出来る。

2. 領域 (region) を作成する

スペクトル解析などに用いる領域 (region file) を作成、保存する。

6.2 ds9 の window (Window of ds9)

ds9 では、図 6.1 のような画面が立ち上がる。この画面を本マニュアルでは main window と呼ぶ。画面上部にはファイル名・マウスのいる位置の座標などが、下部には実際の画像がある。最上部にあるボタンをクリックしながら、コマンドを実行していく。よく使用されるコマンドは画面中程にもある。

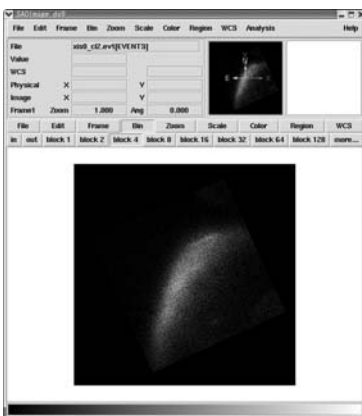


図 6.1: ds9 の window。

6.3 ds9 一通りの使い方 (how to use ds9)

6.3.1 ds9 を立ち上げる (Start up ds9)

ds9 を立ち上げる時は、

```
$ ds9 (infile)
```

とする。または、

```
$ ds9
```

としたあと、file open とボタンをクリックし、infile を入力する。ここで、infile に来るのは、event FITS または image FITS である。

6.3.2 イメージを自分好みに調整する (Adjusting images as you like)

1. binning (event file の場合のみ)

入力ファイルが event file 形式の場合は、イメージの binning が出来る。特に、表面輝度の暗い広がった天体のイメージを見る時は binning が効果的である。

```
Bin block xx
```

の順にクリックすると、イメージが xx bin ごとに binning される。例えば、図 6.2 左を 32bin まとめると、図 6.2 中央のようになる。また、smooth をクリックすると、画像が指定した bin size で smoothing される。smoothing の方法は、boxcar, tophat, gaussian から選択出来る。図 6.2 右は、同図中央を gaussian で smoothing した例である。

なお、ds9 の描画画面は default では 1024x1024 pixel である。そのため、ほとんど binning をしない大きなイメージを見たければ、

```
Bin 4096x4096
```

などと、画像サイズを変える必要がある。

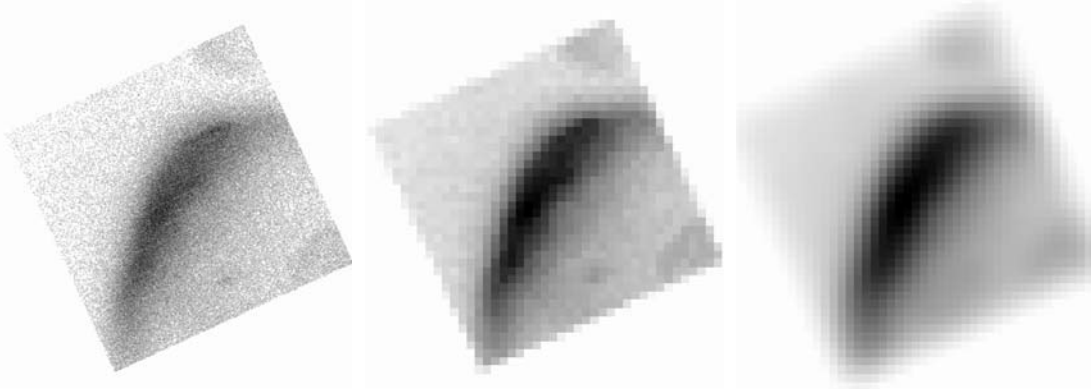


図 6.2: 左: binning していない画像。中央: 32bin まとめした画像。右: さらに gaussian で smoothing した画像。

2. zoom in, zoom out する

画像の zoom in, out を行なう。

```
Zoom (in or out)
```

と順にクリックすればよい。

3. scaling を変える

画像が見やすいよう、scaling を変える。

Scale (linear)

などとして、scale を変えることが出来る。() 内は、linear, log, squared, squared root から選択できる。scaling の最小・最大値は、default では、データ点の最小・最大値になっている。これを変更したい場合は、

Scale Scale parameter

とクリックすると、パラメータ設定用の window が開くので、そこで調節する。

4. カラーを調整する

画像のグラデーションを変え、調節する。

Color (gray)

などとすると、様々なグラデーションに変更することが出来る。gray scale、b (青から赤、黄へと変わるグラデーション)、bb (茶系のグラデーション) などがよく使われる。右クリックしながら画面上をぐりぐりドラッグすると、コントラストなどを変えすることも出来る。コントラストを反転させたい時は、

Color Invert

とすればよい。

5. 領域 (region) を書く

領域 (以下 region) とは、画面上に書き込める幾何学図形や直線、文字などを指す。window 真ん中にあるボタンからたどる時は

Region more ... 、

window 上のボタンからたどる時は

Region shape ...

とクリックし、好みの形の region を選ぶ。右クリック ドラッグで、好みの大きさの region が描ける。出て来た region をクリックすると、4 角に四角い点が現われるので、これをクリック-ドラッグすることで大きさ・位置を調整できる。

さらに大きさや位置、色などの調整をしたい時は、region をダブルクリックして、図 6.3 のような sub window を出す。ここで図形の中心座標や大きさなどを任意に変えることが出来る。default では physical 座標指定になっているが、

Coord WCS

Radius WCS

などとクリックすることで、sky 座標での指定も可能である。region の色や太さは、

Color/Width ...

などとクリックして指定する。

複数の region を組み合わせて一つの region にすることも可能である。その際、sub window メニューで

Property exclude

を選ぶと、その領域をくり抜くことが出来る。くり抜き指定が入った領域は、画面上では赤の斜線が入る。

出来上がった region は、main window 上で

Region Save

とすることでテキストファイルとして保存できる。同様に

Region Load

で保存した region file を読み出すことも可能である。

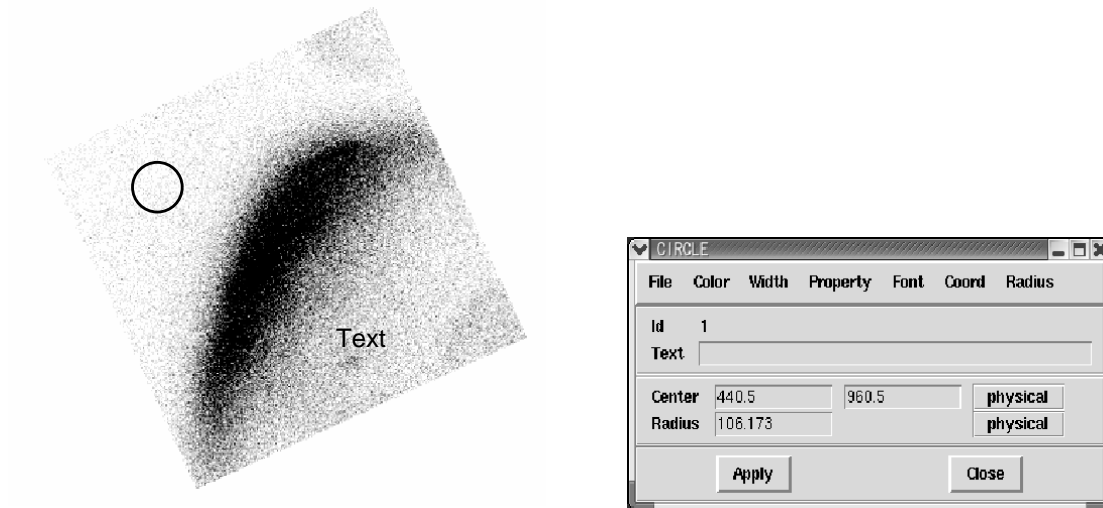


図 6.3: 左: region を書いた一例。右: region パラメータを指定する window の例。

region file はテキストファイルなので、editor を使って編集することが可能である。例えば、

```
circle(64.5,64.5,46.875)
-circle(64.5,64.5,31.25)
```

という region file は、半径 46.875pix の円から半径 31.25pix の円をくり抜いたものになる。

なお、スペクトル解析のための region file の保存の際の注意事項に関しては、6.3.3 節に改めてまとめる。

6. 座標を書く

様々な座標軸を画面上に書きたい時は、

Analysis Display Coordinate Grid

の順にクリックすると、画面上に座標を書くことが出来る。図 6.4 左は、ds9 上で実際に J2000 座標が書かれたところである。座標の種類や色などを指定したい時は、

Analysis Coordinate Grid Parameters ...

とすると、図 6.4 右のような座標用の window が立ち上がる。座標システムや軸間隔を default 以外にしたい時は

Coordinate (目的の座標システム)

Coordinate (目的の座標間隔)

とする。座標軸などの色や太さなどの変更方法は、region の場合と同じである。

座標は、検出器の座標系、sky 座標共にこのコマンドで引くことが出来る。しかし、sky 座標情報の入っていない FITS (例えば、detector 座標で作成された image FITS) の場合は、sky 座標を書くことは出来ない。

7. contour を書く

Analysis Display Contours

とすると、図 6.5 左のように等高線を引くことが出来る。等高線の本数、scaling、smoothingなどを調整したい時は、

Analysis Contours Parameters ...

をクリックし、図 6.5 右の window を立ち上げる。この window の中で、Ccontour Levels は等高線

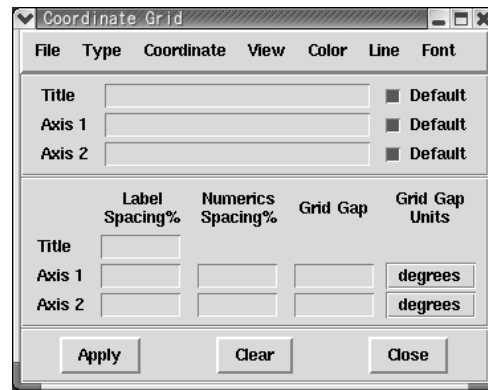
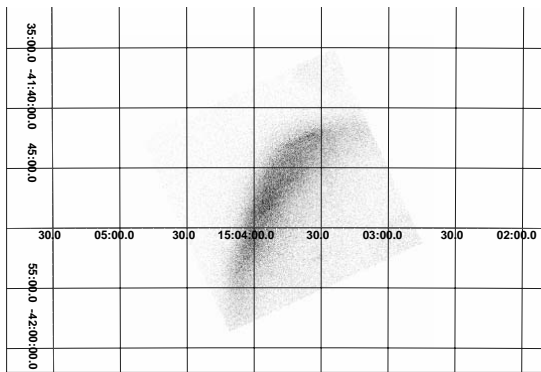


図 6.4: 左: 座標を引いた一例。右: 座標づけパラメータを指定する window。

の本数、Contour Smoothness は、smoothing をかける bin 数である。また、等高線を引く最小値・最大値は Limits を書き換えることで変更できる。等高線間隔を log スケールなどに変更するには、この window の Scale をクリックし、希望のスケールを選ぶ。contour の色や太さの変更方法は、region の場合と同じである。調節が終わったら、Generate ボタンをクリックした後 Apply ボタンを押すと等高線が書ける。

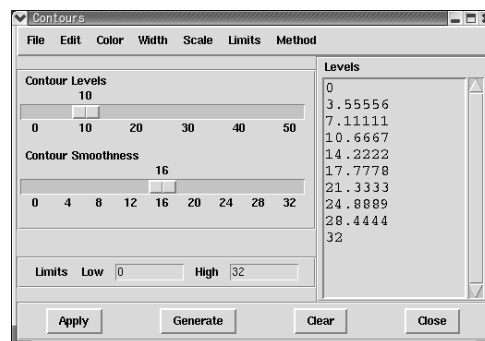
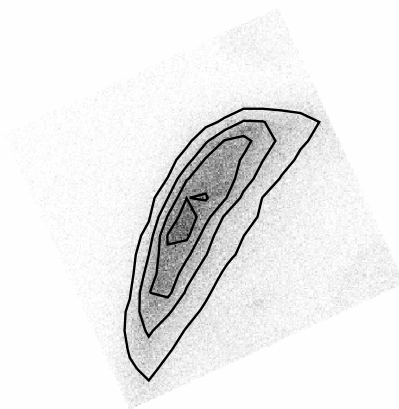


図 6.5: 左: 等高線を引いた一例。右: 等高線パラメータを指定する window。

等高線のパラメータは、contour window 上で

File Save Contours ...

とすると、テキストファイルにセーブしておける。同様に

File Load Contours ...

で、セーブしておいた contour file をロード出来る。

豆知識: 等高線だけのイメージを作成したい時は、カラーをぐりぐりして、真っ白にしてしまえばよい。

6.3.3 スペクトル作成用の region を作成する (Make regions for spectral analysis)

XIS のスペクトルを作る時は、XSELECT 上で region filter をかけることは 5.4.1 節にまとめられている。ここでは、スペクトル作成の際の region file format の注意事項をまとめておく。

1. sky 座標で region を作成したい場合

sky 座標で region を作るのは、

- ・ 複数観測した時のデータを合わせたい時
- ・ 他の衛星のデータ、論文の座標などと合わせたい時

に便利である。このような場合は、

```
Region File Format DS9/Funtools
```

```
Region Region Coordinate System WCS
```

```
Region Region Coordinate System (好きな sky 座標)
```

としてセーブし、region を作成する。

2. physical 座標で region を作成したい場合

physical 座標で region を作るのは、

- ・ background を CCD の同じ位置から取りたい時

などに便利である。この場合は、

```
Region Region Format DS9/Funtools
```

```
Region Region Coordinate System Image
```

```
Bin block 8
```

として region を作成する。ここで注意しなければいけないのは、

!! イメージの binning を XSELECT 上の set xybinsize で決められているものと同じにする !!

ことである。default では、xybinsize = 8 となっている。

6.3.4 画像を保存・印刷する (Saving and printing images)

出来た画像を保存する時は、

```
File Save Images as ...
```

とクリックし、好みのフォーマットをクリックすれば良い。保存できるフォーマットは、FITS, jpeg, TIFF, PNG, PPM, MPEG。

PS file に落としたければ、

```
File Print
```

で出て来る sub window で、"Print To file" を選べばよい。ここで Printer を選べば、printer に出力される。

6.3.5 ds9 の終了 (Shut down of ds9)

ds9 を終了するには、

```
File exit
```

の順にクリックする。

第7章 TIMING ANALYSIS; XRONOS

7.1 Overview of XRONOS

ライトカーブを描いたり、時系列解析を行なう場合は、HEADAS に含まれる XANADU package の XRONOS ツール群を使用する。「あすか」の頃は、X 線解析のための独立したツール群となっていて、XRONOS 環境に入ってから使う仕様になっていたが、今は、個別の ftools として、コマンドラインで使えるようになってる。詳細は、HEASARC の大元のページである、

<http://heasarc.gsfc.nasa.gov/docs/xanadu/xronos/xronos.html>

を参考のこと。記述に不一致があれば、HEASARC の方を信じ、suzaku_local@crab.riken.jp に連絡ねがいます。大まかには下記のようなことができる。

- ライトカーブの描画 (lcurve)
- パワースペクトルの作成 (powspec)
- 周期的変動の探査 (efsearch)
- 畳み込みライトカーブの作成 (efold)

7.2 Barycentric Correction for Suzaku

衛星は地球のまわりをまわっており、地球は太陽のまわりをまわっている。光は地球半径分すすむのに 20 msec かかり、太陽まで 8 分かかかる。他の衛星と時刻を合わせたい場合は、衛星の軌道運動による効果を打ち消すために、時刻を地球中心の値に変換し (Geocentric correction)、さらに、観測時期による違い (地球の公転運動の効果) を打ち消すために、太陽中心での時刻に補正する必要がある。これを Heliocentric correction と呼ぶ。周期の速い X 線パルサーや電波パルサー、ミリ秒パルサーの Pulse の到着時刻について、他の時期の観測と比較するためには、さらに、惑星の運動をすべて加味した太陽系重心への変換が必要となる。これを barycentric correction と呼ぶ。

XRONOS package にも barycentric correction tool は含まれているが、「すざく」の軌道要素が読めるツールとして aebarycen を HEADAS v6.1.x からリリースするに至った¹。ただし、まだ、絶対時刻にバグがある可能性が高いので全面的には信用しないと共に、なにかバグをみつけたら suzaku_help@crab.riken.jp まで報告してほしい。用意するファイルは、時刻補正したいイベントファイル (aeXXXXXX.evt) と軌道ファイル (aeXXXXXX.orb) である。

aebarycen version 2006-07-24

Written by Y.Terada (RIKEN), T.Enoto (UT), Y.ISHISAKI (TMU)

Input/output file or @filelist to be corrected[] ae101005040hxd_0_wel_uf2.evt<--入力File

Orbit ephemeris file for Suzaku[]ae101005040.orb <--軌道要素

leapsec file name [] : /usr/local/lheasoft/6.06/headas/ftools/refdata/leapsec.fits

<--- 閏秒のファイル (HEADAS をインストールした場所にある)

(CALDB と書けば CALDB から適切に取って来てくれる)

¹prepared by Y.Terada, T.Enoto, and Y.Ishisaki et.al

```

Right Ascension (HeaderKey/NNhNNmNNs/deg): 500 <--- 天体の RA
Declination (HeaderKey/+NNdNNmNNs/deg): 50 <--- 天体の DEC
(デフォルト値は用意されていない。自分の興味のある天体の座標をいれる。)
(このように定義外の値を与えると, event FITS のヘッダの RA_NON, DEC_NOM が使われる)
....

```

こうして出来た FITS file を見てみると、

```

HISTORY -----
HISTORY aebarycen version 2006-07-24 at 2006-08-02T07:34:53
HISTORY -----
HISTORY filelist='ae101005040hxd_0_wel_uf2.evt'
HISTORY orbit='ae101005040.orb'
HISTORY leapfile='/usr/local/lheasoft/6.06/headas/ftools/refdata/leapsec.fits'
HISTORY (RA,DEC) = (500,500) = ( 300.0000 , 40.0000 )
HISTORY time_col='TIME', start_col='START', stop_col='STOP'
PLEPHEM = 'JPL-DE200' / Solar System ephemeris used for baryctr corr.

```

といった履歴がかかれ、

```

TIMESYS = 'TDB' / Coordinate Reference System
TIMEREF = 'SOLARSYSTEM' / Times are pathlength-corrected to barycenter
TSTART = 206432797.694330 / time start (+243.395389 s barycen corrected)
TSTOP = 206487866.716041 / time stop (+244.418146 s barycen corrected)
DATE = '2006-08-02T07:34:53' / file creation date (YYYY-MM-DDThh:mm:ss UT)

```

のように TSTART や TSTOP の値などが書き換わり、TIMEREF が、'LOCAL' から 'SOLARSYSTEM' に変わっているはずである。もちろん、event 本体の時刻や GTI も変更されているはずである。確認のため、補正前の時刻との差で最もよく効く値として、Hericentric correction の値を計算したい場合、天体の座標 (RA,DEC) と地球の位置とから、天体へのベクトルを太陽公転面の垂線に射影した距離から図 7.1 を参考にするとよい。

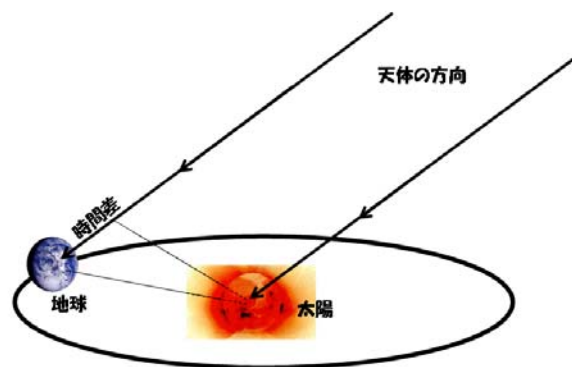


図 7.1: 太陽中心補正で変更される時間差

7.3 Make Light Curve

下記の要領でライトカーブが描ける。

```
% lcurve
lcurve 1.0 (xronos5.21)

Number of time series for this task[1] 1
      <----- (1 つのファイルだけをプロットするなら 1, 2 つ以上ならその数)
Ser. 1 filename +options (or @file of filenames +options) [] crabhxdnor_hxd_wel_pin.evt
      <----- (Event FITS もしくは light curve FITS の名をいれる)
.....
Default Newbin Time is:   39.632403   (s) (to have 1 Intv. of   512 Newbins)
Type INDEF to accept the default value

Newbin Time or negative rebinning[100] 100.0
      <----- ライトカーブの一ピンの時間幅を入力。今の場合 100 秒。
      当然、最小の時間分解能より小さくすることは出来ない。
Newbin Time .....   100.00000   (s)
Maximum Newbin No.   203
Default Newbins per Interval are:   203
(giving   1 Interval of   203 Newbins)
      <----- イベントのカバーする時間範囲と、先程入力した時間ピンから、
      トータルで何ピンのライトカーブになるかを教えてくれている。

Number of Newbins/Interval[239] 203
      <----- 描画する時間領域 (Interval) あたり何ピンを表示するか入力。
      今は、一画面でみたいので、先に教えてもらった 203 を入力。
      たとえば、20 を指定すると、21 枚分 描画される。

Maximum of   1 Intvs. with   203 Newbins of   100.000   (s)
      <----- 単なる確認。1 Interval となっていますね。
Name of output file[default] <----- このあたりは default 、リターンで OK
Do you want to plot your results?[yes] <----- リターン
Enter PGPLOT device[/XW] <----- リターン
.....
PLT> q <----- QDP に入る。自由にどうぞ。
```

複数のファイルを 1 シリーズとしてまとめる時は、ファイルをリストアップした flist というファイルを作り、Ser.1. ... [] の答えを @flist とすれば OK。また、2 シリーズ以上を与えた時は、最後にプロットする段階で、複数プロットモードか、比を取るモードか、などを聞かれるので、素直に答えてやればよい。

7.4 Make Power Spectrum

周期的な変動がある場合は、ライトカーブの各周期毎のパワーを描いた「パワースペクトル」を描いたときに、その特定の周波数に高いパワーがあらわれる。「すざく」の場合は、SAA や COR、地球の蝕などによって、データが離散的であるので、人為的にでた「嘘のパワー」を見極め、天体固有の周期変動を探查する必要がある。

```
% powspec
Ser. 1 filename +options (or @file of filenames +options)[] crab_hxd_wel_pin.evt
      <----- (Event FITS もしくは light curve FITS の名をいれる)
...
Name of the window file ('-' for default window)[] -
      <----- デフォルトでよい。リターン
...
Default Newbin Time is:  2.4743037      (s) (to have 1 Intv. of  8192 Newbins)
      <----- 時間分解能の適当な値を教えてください、参考程度にするだけ。
Newbin Time or negative rebinning[0.1] 0.01
      <----- パワースペクトルを描くための、元となるライトカーブの時間分解
                能を入力。ミリ秒を探查したいのに 大きな値 10.0 sec として
                もダメだし、500 sec 程度の周期変動を探查したいのに、あまり
                細かな時間ビン指定しても、一ビンあたりの統計が悪く、やたら
                ビン数も増えるばかりでよろしくない。
Newbin Time ..... 0.10000000E-01 (s)
Maximum Newbin No. 2026802
      <----- 指定した時間分解能だと、全体で何ビンのライトカーブかを教えて
                くれる。(参考値)

つぎに、「計算の元となるライトカーブ」を何枚分用意するか、を考える。
Default Newbins per Interval are: 8192
(giving 248 Intervals of 8192 Newbins each)
      <----- 参考値。8192 bin づつのライトカーブなら 248 枚できる、と
                言っている。
Number of Newbins/Interval[8192] 204800
      <----- 一枚 (1 Interval) あたりのビン数を入力。ここでは、約 10 枚
                (10 Interval) を予定して、(8192x248/10) 程度。

**** Warning: No. of Newbins/Intv. reset to 262144
Maximum of 8 Intvs. with 262144 Newbins of 0.100000E-01 (s)
Default intervals per frame are: 8
      <----- 適切な調整がなされ、だいたい予定どおり、8 Interval の計
                算となった。
```

最後に、Interval 毎のパワーの計算を何枚 (Frame) づつ重ねて描画するか、を決める。

Number of Intervals/Frame[25] 8

<----- 8 Interval 全部を 1 枚 (1 Frame) のパワースペクトルにまとめたいので、ここでは 8 を指定。たとえば 2 とすると $8/2 = 4$ Frame 結果がでる。

Rebin results? (>1 const rebin, <-1 geom. rebin, 0 none)[0] <----- デフォルト

Name of output file[default] <----- デフォルト

Do you want to plot your results?[yes] <----- デフォルト

Enter PGPLOT device[/XW] <----- デフォルト

...

PLT> q <----- QDP に入る。自由にどうぞ。

こんな感じでやると、図 7.2 のようなパワースペクトルがかける。

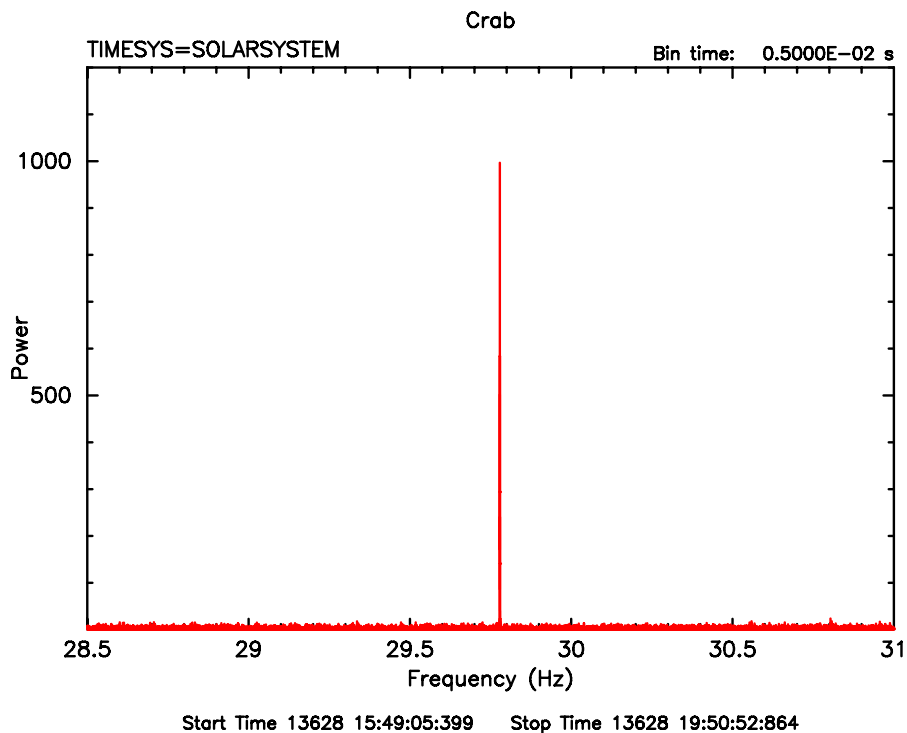


図 7.2: Crab パルサーのパワースペクトル

7.5 Perform Periodic Search

前の節で、およその周期が分かったので、その精度をあげる。ベストの周期で、天体のライトカーブを畳み込んでやると、ひじょうに変動の大きなライトカーブになるはずである。逆に、無関係な周期で畳み込むと、平らなライトカーブとなるであろう。サーチプログラムでは、次節 7.6 で述べるような畳み込みのライトカーブを、いろいろな周期で描き、それが一定なのか否かを統計的に χ^2 検定し、その χ^2 が最大なるような周期を算出する。下記の例では、Crab パルサーに対し、33.5810 msec を中心に 0.0002 msec の分解能で +/- 128 個分のトライアルを行なってみる。

```
% efsearch
Ser. 1 filename +options (or @file of filenames +options)[file1] evt/crab_hxd_wel_pin.evt
      <----- (Event FITS もしくは light curve FITS の名をいれる)
...
Name of the window file ('-' for default window)[-]
      <----- デフォルトでよい。リターン
...

計算の元となる 畳み込みライトカーブの時刻原点と、探査したい周期を与える
Default Epoch is: 13628.00000
Epoch format is days.
Epoch[34 234.23] 13628 <----- 原点: MJD から 40000 引いた値を用いる (*注)
Period format is seconds.
Period[88.87] 0.0335810 <----- 周期: default 設定 なら 秒単位で与える
Expected Cycles ..          603555.91 <---- 何周期分のデータに相当するか表示 (参考値)
...

畳み込みライトカーブの 0 ~ 1 phase を何ビンに分けるかを設定
Default phase bins per period are:          8      <--- 参考値
Phasebins/Period {value or neg. power of 2}[-3] 32 <--- 自分の設定値
...

次に、ひとつの計算で用いる時間範囲を指定する
(giving          1 Interval of          19313790 Newbins) <----- 全体を 1 Interval にす
Type INDEF to accept the default value          るならこの値 (参考値)
Number of Newbins/Interval[10] 19313790 <----- 設定値
Maximum of          1 Intvs. with          19313790 Newbins of          0.104941E-02 (s)
...

最後に探査する時間分解能とビン数を指定する。
Resolution for period search {value or neg. power of 2}[.03] 0.0000002
      <----- 0.0002 ミリ秒ステップで計算
Number of periods to search[100] 128 <----- 前後 128 x 0.0002 msec.
...
Name of output file[default] <----- デフォルトで OK
Do you want to plot your results?[yes] <----- デフォルトで OK
Enter PGPLOT device[/XW] <----- デフォルトで OK
PLT> q      <----- QDP に入る。自由にどうぞ。
```


結果は下記のような感じである。これを繰り返かし、時間分解能をあげて探査するとよい。

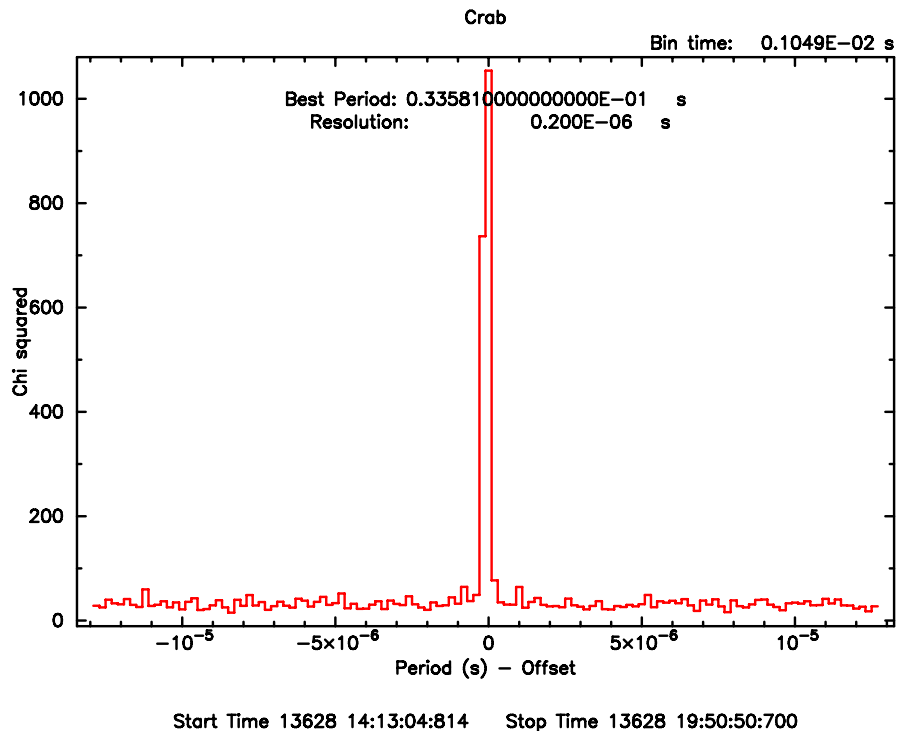


図 7.3: Crab パルサーの周期探査の結果画面

周期的変動をサーチする際、検出器の時間分解能以上の時間が探査できることに疑問を持つかもしれない。たとえば、1.00 秒しか時間分解能のない検出器があったとしても、100 秒の間に 100 pulse が含まれるか、99 pulse しかふくまれないか、が区別できれば、入射イベントが 1.00 秒周期なのか 1.01 秒周期なのかの区別がつく。すなわち、周期的な変動の探査には、検出器の時間分解能だけでなく、観測時間が何サイクル分カバーしているかも効いてくる。

(*注) XRONOS のデフォルトの時間軸の単位は TJD (Truncated Julian Days $TJD=JD-2440000.5$) である。一方、MJD (Modified Julian Days) = $JD-2400000.5$ [day] なので、 $TJD = MJD - 40000$ [day] となり、40000 days のずれができる。

7.6 Make Folded Light Curve

周期が求まったので、次に、その周期のなかで天体がどのような波形で変動しているのかを知るために、この時間で畳み込んだライトカーブを描いてみる。周期 P で変動すると考える天体からある時刻 T にイベントが到来し、その時刻 T が、ある時刻原点 T_0 から数えて N (整数) + x (0..1) サイクル目に相当する場合 ($T = T_0 + P \times (N + x)$)、 x は phase と呼ばれ、0 から 1 の値をとる。ふつうのライトカーブは横軸 T 、縦軸 カウントのヒストグラムであるが、横軸 x 縦軸カウントとしたライトカーブは、畳み込みライトカーブと呼ぶ。(周期 P で折り畳んで表示しているようなもの) 解析では下記のように efold というツールを用いて描画できる。

```
% efold
Number of time series for this task[1] 1
      <----- (1 つのファイルだけをプロットするなら 1, 2 つ以上ならその数)
Ser. 1 filename +options (or @file of filenames +options)[] crabhxdnor_hxd_wel_pin.evt
      <----- (Event FITS もしくは light curve FITS の名をいれる)
...
Name of the window file ('-' for default window)[-]
      <----- デフォルトでよい。リターン
...

Default Epoch is: 13628.00000
Epoch format is days.
Epoch[] 13628 <----- 畳み込みの原点: Phase = 0 の値: MJD - 40000
Period format is seconds.
Period[] 33.5809E-3 <----- 畳み込み周期: default 設定 なら 秒単位で与える
Period derivative [0.00] <----- P_dot (default なら聞かれない)

Phasebins/Period {value or neg. power of 2}[17] 257
      <--- 畳み込みライトカーブの 0 ~ 1 phase を何ビンに分けるかを設定
...

Default Newbins per Interval are: 155114332
(giving 1 Interval of 155114332 Newbins)
      <---- 計算を すべてまとめて 1 枚 (1 Interval) でおさめる場合の
      ビン数が表示される (参考値)
Number of Newbins/Interval[10260482] 155114332
      <----- 計算は、1 Interval で行ないたいので、参考値そのまま。
Default intervals per frame are: 1 <---- 描画の枚数 (Frame) の参考値
      今は 1 Interval しかないので 1 枚。
Number of Intervals/Frame[1] 1 <----- 1 枚にまとめて表示と入力
...
Name of output file[] <----- デフォルトで OK
Do you want to plot your results?[yes] <---- デフォルトで OK
Enter PGPLOT device[/xw] <----- デフォルトで OK
PLT> q <----- QDP に入る。自由にどうぞ。
```

結果は、次の図 7.4 に示すようなかたちで表示される。一番はじめの箇所で、複数のシリーズを指定した場合は、画面が縦に分割されて表示される。

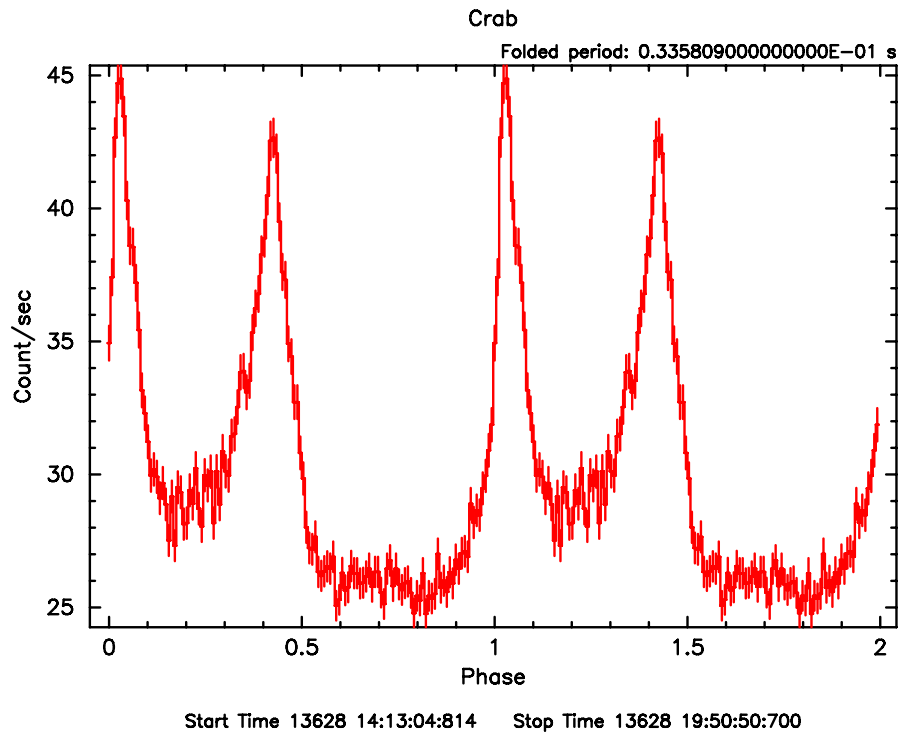


図 7.4: Crab パルサーのライトカーブを 33.5809 ミリ秒で畳み込んだライトカーブ

7.7 Additional Comments

次の段階としては、図 7.4 の phase で分割してスペクトルを描きたい、などの要求がでてくるだろう。詳細は xselect の章 (§5) にまかせるが、

xselect 上で filter する際、

```
xsel:SUZAKU-HXD-WELL_PIN > filter PHASE
> Enter what filter? >[TIME] PHASE
> Enter the epoch for phase selection ( in MJD ) >[] 53628.0
> Enter the period for phase selection ( in DAYS ) >[] 3.88668e-07
> Enter phases from 0-1 (e.g. 0.1-0.25,0.3-0.45) >[] 0-0.5
```

といった感じでイベントを抜きだせる。なお、時刻原点は XRONOS と MJD では 値が 40000 ずれていること (§7.5 注)、Period が秒ではなく Day であることに注意する。また、Period があまりに短いものは xselect 上では切り出せないことがあるようである。(bug でしょう) 一度 filter したものを event におとし、efold などでも正しく切り出されているかを確認して次のステップにすすむのがよからう。

第8章 XSPECによるスペクトルフィッティング (XSPEC)

8.1 grpphaでビンまとめ (Grouping of PI Bins with grppha)

XSELECT で作られたスペクトルは、そのままではビンが細かく、一つのビンあたりの統計がよくない。そこで実際にスペクトルフィットを行う前に grppha を用いて binning する。

8.1.1 方針 (Policy)

xspec では通常 χ^2 検定を用いてフィッティングを行っている。ここで、チャンネルあたりのデータの標準偏差 σ はポアソン統計に基づいて \sqrt{N} (N はチャンネルのカウント数の期待値) に従うが¹、データにつくエラーは観測されたカウント数 N に基づいて \sqrt{N} としている。フィットは $\sqrt{N} \approx \sqrt{N}$ の仮定のもとに行われるが、実際には N が小さくなると、この仮定が崩れ不定性が大きくなる。 $\sqrt{N} \approx \sqrt{N}$ の仮定が妥当性を持つのは $N > 10$ 程度であり、これが保たれるようにビンまとめする。だいたい $N > 20$ くらいでビンまとめしているのが普通のようなのだが、バックグラウンドなどを考慮して最適なビンまとめをする。また、ラインの構造など調べたければ、検出器の分解能をつぶすほどにビンまとめしてはいけない。

8.1.2 基本的な実行例 (Simple examples)

grppha の実行形式は

```
$ grppha (infile) (outfile)
```

である。以下の例では、1ビンあたりの最小のカウント数 N を 20cts としてビンまとめしている。

```
$grppha xis0.pha xis0_grp.pha
```

```
-----
MANDATORY KEYWORDS/VALUES
-----
-----
EXTNAME      - SPECTRUM      Name of this BINTABLE
TELESCOP     - SUZAKU        Mission/Satellite name
INSTRUME     - XISO          Instrument/Detector
FILTER       - NONE         Instrument filter in use
EXPOSURE     - 17225.         Integration time (in secs) of PHA data
AREASCAL    - 1.0000         Area scaling factor
BACKSCAL    - 1.0000         Background scaling factor
BACKFILE     - none             Associated background file
CORRSCAL    - 1.0000         Correlation scaling factor
CORRFILE     - none             Associated correlation file
RESPFILE    - none             Associated redistribution matrix file
ANCRFILE    - none             Associated ancillary response file
```

¹ここら辺は人によるらしく、ガウス統計のエラーをデータにつけている場合もあるようだ。

```

POISSERR - TRUE          Whether Poissonian errors apply
CHANTYPE - PI           Whether channels have been corrected
TLMIN1   - 0           First legal Detector channel
DETCCHAN - 4096        No. of legal detector channels
NCHAN    - 4096        No. of detector channels in dataset
PHAVERSN - 1.2.0       OGIP FITS version number
STAT_ERR - FALSE       Statistical Error
SYS_ERR  - FALSE       Fractional Systematic Error
QUALITY  - TRUE        Quality Flag
GROUPING - FALSE       Grouping Flag

```

```

-----
GRPPHA[] group min 20          <--- ( 1 ) 1bin あたり少なくとも 20cts はあるようにビンまとめ
GRPPHA[exit] exit
... written the PHA data Extension
..... exiting, changes written to file : xis0_grp.pha
** grppha 3.0.0 completed successfully

```

xis0_grp.pha の中身を fdump してみると以下のように GROUPING というコラムができていのがわかる。'1' から '-1' をはさんで次の '1' がくるまでが一つのビンにまとめられている。今 min 20 でビンまとめをしたので、0-96channel の足し合わせが 28 となりここまでが一つのビンにまとめられている。

```

----- xis0_grp.pha の中身 -----
      CHANNEL      COUNTS      QUALITY  GROUPING
              count
      1           0           0      0      1
      2           1           0      0     -1
      3           2           0      0     -1
      4           3           0      0     -1
      5           4           0      0     -1

... skip ...

      94          93           0      0     -1
      95          94           0      0     -1
      96          95          13      0     -1
      97          96          15      0     -1
      98          97          18      0      1
      99          98          13      0     -1

```

8.1.3 PHA channelを指定してビンまとめする方法 (How to group PHA channels by your favorite assignments)

起動などは min 20 の場合とほとんど同じですが、(1) の group のあとに具体的に bin まとめを指定します。明るい天体などは min20 ではほとんどビンまとめされないのこちらの方法がお勧めです。

```

GRPPHA[] group 0 511 1 512 1023 2 1024 2047 4 2048 4095 8
<--- ( 2 ) 0-511 channel までは 1bin まとめ、512-1023 channel までは 2bin まとめ、...

```

8.1.4 systematic errorを入れる (How to include systematic errors)

暗い天体なら systematic error は無視できますが、銀河系内の明るい天体であれば、statistic error にたいして systematic error が無視できなくなります。何%の systematic error を入れるかは今後の cal に依存

しますが、ここでは grppha を使ってピンまとめしたあとに 1bin あたり 1% の systematic error を入れる方法を紹介します。

ピンまとめと systematic error の関係を示すファイルを用意 (definition file between systematics and PHA bins)

"0 511 1 512 1023 2 1024 2047 4 2048 4095 8" でピンまとめすることにして、以下のようにして syserror_1.dat というファイルを用意します。ファイルの中身の形式は

[最初の channel] [最後の channel] [ピンまとめ前のその channel の systematic error]

である。したがって、上記のピンまとめでは以下のように syserror_1.dat というファイルを用意し、ピンまとめ後に 1% の systematic error が定義する。

```
$ echo "0 511 1" | awk '{print $1,$2,$3/100}' > syserror_1.dat
$ echo "512 1023 2 " | awk '{print $1,$2,$3**0.5/100}' >> syserror_1.dat
$ echo "1024 2047 4 " | awk '{print $1,$2,$3**0.5/100}' >> syserror_1.dat
$ echo "2048 4095 8' " | awk '{print $1,$2,$3**0.5/100}' >> syserror_1.dat
```

--syserror_1.dat の中身--

```
0 511 0.01
512 1023 0.0141421
1024 2047 0.02
2048 4095 0.0282843
```

grppha でピンまとめし、systematic error を入れる (grouping by grppha and include systematic errors)

```
GRPPHA[] group 0 511 1 512 1023 2 1024 2047 4 2048 4095 8
GRPPHA[] sys syserror_1.dat <---( systematic error のファイルを読み込む)
```

8.2 スペクトルを足す方法 (How to Add Spectra)

統計が足りない場合、スペクトルを足したくなることもある。以下は xis0, xis2, xis3 の pha ファイルを足して、fi_ave.pha というファイルを作る場合を示す。

8.2.1 mathpha でスペクトルを足す (Add spectra via mathpha)

fhelp mathpha で詳細を確認してみること。

```
$ mathpha ERRMETH=POISS-0 <--- POISS-0 ではエラーの計算はされない。
Expression to be evaluated[] xis0.pha+xis2.pha+xis3.pha <--- input file
Units algebraic expression to be performed in (C,R,F or Help)[C] C <--- (C:Counts, R:Rate, F:File)
O/p PHA filename[] fi_ave.pha <--- output file
Exposure time flag/value ({value},{file},CALC,NULL or Help)[] CALC <---EXPOSURE を足し合わせる。
Areascal flag/value ({value},{file},NULL or Help) [%] NULL
Number of comment strings to be added (up to 4)[1]
Comment 1[]
..... processing file: xis0.pha
..... processing file: xis2.pha
..... processing file: xis3.pha
... Following keywrds set to null values in o/p file:
```

```

..... TELESCOP, INSTRUME, FILTER
... ... performing algebra in units of COUNTS
... written the PHA data Extension
** mathpha 4.5.7 completed successfully

```

注意 上の実行例では BACKSCAL の値はリセットされるので、後から書き直してやる必要がある。足し合わせたいファイルの BACKSCAL の値が全て 1%以内でそろっていれば、実行時に `mathpha backscal='%` として書き込まれる。

8.2.2 レスポンスの足し算 (Add response matrix files)

arf, rmf それぞれの足し合わせには第 9 章の `addarf`, `addrmf` という `ftool` が使える。ただし、検出器間のレスポンスを足す場合、先に検出器ごとの arf と rmf をかけて $rsp = arf \times rmf$ にしておいてから (`marfrmf` という `ftools` を使うとよい)、`rsp` 同士を足さなければならない (`addrmf` を使う)。arf と rmf を別々に足してからかけると、 $arf1 \times rmf2$ などの cross term の分だけ間違えることになる。また、`rsp` を足す際にはどのような重みで足すかを十分に考える必要がある。それによって足したスペクトルの exposure に入れるべき値も変わってくる。

8.3 xspec でフィッティング (Fittings in XSPEC)

ここでは `xspec11` のみ紹介する。`xspec12` を使いたい場合、`heasarc` のマニュアルを参照すること。

8.3.1 設定 (Settings)

詳しくは、

<http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/install.html>
の "INITIALIZATION AND SETUP" を参照すればよい。ここでは `tcsch` の場合の設定例を示す。

```

$ setenv HEADAS /path/to/your/installed/headas/<PLATFORM>
                                                    <-- HEADAS をインストールした場所
$ source $HEADAS/headas-init.csh

```

which `xspec11` して確認する。

```

$ which xspec11
/path/to/your/installed/headas/<PLATFORM>/bin/xspec11

```

画面表示の設定もしておきます。ここでは `/xw` の場合ですが、重くて嫌な場合は `/xt` なども使える。

```

$ setenv XSPEC_TYPE cpd/xw
$ setenv PGPLOT_TYPE /xw

```

8.3.2 xspec を走らせてみる (execute XSPEC)

```
$ xspec11
```

```
Xspec 11.3.2p 21:31:51 27-Jan-2006
```

```
For documentation, notes, and fixes see http://xspec.gsfc.nasa.gov/
```

```
Current plot device /xw
```

```
Type "help" or "?" for further information
Auto-save has been disabled.
```

```
Abundances set to Anders & Ebihara
```

```
XSPEC>exit
Do you really want to exit? (y)
XSPEC: quit
Type <RETURN> for next page:
```

8.3.3 dataの読み込み (Read data)

```
XSPEC>data 1:1 xis0_grp.pha
XSPEC>back 1 xis0_bkg.pha
XSPEC>resp 1 xis0.rmf <--- 検出器の適切なレスポンスを使用してください。
XSPEC>arf 1 xis0.arf <--- 検出器の適切な arf を使用してください。
```

とすればよい。正式のレスポンスは NASA/GSFC の以下のページから取得できる。
<http://suzaku.gsfc.nasa.gov/docs/heasarc/caldb/suzaku/>

ここで、data, back, resp, arf の”1:1”や”1”は 1 番目のファイルであることを示す。2 番目のデータを同時に読み込みたければ、data 2:2 ***, back 2 ***, resp 2 ***, arf 2 *** と続ければよい。以下のようになればきちんと読み込まれている。

```
XSPEC>show files

Information for file 1
  belonging to plot group 5, data group 5
  telescope = SUZAKU , instrument = XIS0 , channel type = PI
  Current data file : xis0_grp.pha
    with integration time 1.7225E+04
  Background file : xis0_bkg.pha
  No current correction
  Response (RMF) file : xis0.rmf
  Auxiliary (ARF) file : xis0.arf
  Weighting method is standard
```

8.3.4 スペクトルを表示 (Display spectra)

```
XSPEC>setplot ch <--- 横軸のスケールを channel にする。( default はこっちなのでうたなくてもよい)
XSPEC>plot ldata <--- 横軸 channel で表示されるはず
XSPEC>setplot energy <--- 横軸のスケールを energy にする。
XSPEC>plot <--- 横軸 energy [keV] で表示されるはず
XSPEC>ignore *-1.0 10.0-* <--- 1keV 以下、10keV 以上は使わないという意味。( case by case )
XSPEC>notice 0.3-10.0 <--- 0.3--10keV を使うという意味( case by case )
XSPEC>plot <---0.3--10keV のスペクトルが表示されるはず。
```

8.3.5 モデルフィット (Fittings with models)

xspec では通常 χ^2 検定を用い、「モデルがデータをよく再現する」とき reduced χ^2 の値 χ^2/dof は 1 に近くなる。詳しくは、Bevington などの教科書を読んで勉強すること。試しに「吸収*ベキ関数」のモデルでフィットする。吸収は wabs、ベキ関数は powerlaw というモデルを使う。

```
XSPEC>model wabs*power <--- (1) モデルを決める。
Model: wabs<1>( powerlaw<2> )
Input parameter value, delta, min, bot, top, and max values for ...
```


1 0.001 0 0 1E+05 1E+06 <--- default の値が表示される (最初の"1"が default 値、次の 0.001 がフィットする際のステップ幅、次の 0 と 1E+5 はベストフィットを探す最小/最大値)

1:wabs:nH>0.1 <--- (2a) wabs の初期値を入力ここでは 0.1 を入力してみる (何もいれなければ default)

1 0.01 -3 -2 9 10
2:powerlaw:PhoIndex>2 <--- (2b) power-law の光子指数の初期値を 2 にしてみる。

1 0.01 0 0 1E+24 1E+24
3:powerlaw:norm> <--- (2c) normalization は default 値を入れてみる。

Model: wabs<1>(powerlaw<2>)

Model Fit	Model Component	Parameter	Unit	Value
par par comp				
1 1 1	wabs	nH	10 ²²	0.100000 +/- 0.000000
2 2 2	powerlaw	PhoIndex		2.000000 +/- 0.000000
3 3 2	powerlaw	norm		1.000000 +/- 0.000000

Chi-Squared = 3507078. using 118 PHA bins. <--- 自乗値と自由度(まったく合っていない)

Reduced chi-squared = 31313.20 for 112 degrees of freedom
Null hypothesis probability = 0.00 <--- 信頼度

XSPEC>plot ldata delchi <--- スペクトル、モデルとの residual を表示。

XSPEC>renorm <--- (3) モデルの形は変えずに normalization だけ振ってみる。
あまりにもモデルとデータが離れすぎていると
フィットに時間がかかるため。

Chi-Squared = 1422.628 using 118 PHA bins.
Reduced chi-squared = 12.70203 for 112 degrees of freedom
Null hypothesis probability = 0.00

XSPEC>show <--- パラメータがどうなっているか見てみる。

Model: wabs<1>(powerlaw<2>)

Model Fit	Model Component	Parameter	Unit	Value
par par comp				
1 1 1	wabs	nH	10 ²²	0.100000 +/- 0.000000
2 2 2	powerlaw	PhoIndex		2.000000 +/- 0.000000
3 3 2	powerlaw	norm		0.314087 +/- 0.000000 <--- ここがかわった。

Chi-Squared = 1422.628 using 118 PHA bins. <--- 少し良くなったけど合っていない。
Reduced chi-squared = 12.70203 for 112 degrees of freedom
Null hypothesis probability = 0.00

XSPEC>fit <--- (4) 全てのパラメータをフィットしてみる。

Number of trials exceeded - last iteration delta = 0.1023

Continue fitting? (Y)

と聞かれるので、フィットを続けたければ yes、止めたければ no をうつ。いちいち聞かれるのが手間な場合は、fit の前に "query yes"もしくは"query no"と打っておけばよい。
fit が終わると次のようにでる。通常 fit は数回繰り返すべき。

Model: wabs<1>(powerlaw<2>)

Model Fit	Model Component	Parameter	Unit	Value
par par comp				
1 1 1	wabs	nH	10 ²²	0.304457 +/- 0.112850E-02
2 2 2	powerlaw	PhoIndex		1.73741 +/- 0.195044E-02
3 3 2	powerlaw	norm		0.415500 +/- 0.101485E-02

```
-----  
-----  
Chi-Squared =      115.9575      using   118 PHA bins.  
Reduced chi-squared =      1.035335      for   112 degrees of freedom <--- 良くなった。  
Null hypothesis probability = 0.380
```

8.3.6 フラックスを求める (Determine the flux)

ベストフィットのモデルに対して、フラックスを求めてみる。例えば、上記のモデルフィットの結果に対して、

```
XSPEC>flux 0.5 10
```

と打つと、以下のように 0.5-10keV のフラックスが求まる。

```
Model flux  0.4657      photons ( 2.0272E-09 ergs)cm**-2 s**-1 ( 0.500- 10.000)
```

8.3.7 検出器のエネルギー領域を超えてフラックスを求める (Determine the flux in out of energy ranges of a detector response)

ベストフィットのモデルに基づいて、検出器のレスポンスが定義されているエネルギー領域を超えてフラックスを求めたくなることがある。その場合、DUMMYRSP というコマンドが便利。以下は 0.1-100keV のフラックスを求めたい場合の実行例である。

```
XSPEC> dummy 0.1 100 1000 <-- 0.1keV から 100keV まで 1000bin でダミーのレスポンスを作成  
XSPEC> flux 0.1 100  
Model flux  0.5590      photons ( 5.8868E-09 ergs)cm**-2 s**-1 ( 0.100-100.000)
```

8.3.8 モデルの追加と削除 (Add or Delete Models)

追加、削除にはそれぞれ ADDCOMP、DELCOMP というコマンド、もしくは EDITMOD というコマンドを使う。

addcomp

たとえば、現在のモデル

```
Model: wabs<1>( powerlaw<2> )
```

に対して、

```
XSPEC> add 2 diskbb
```

とすると

```
Model: wabs<1>( diskbb<2> + powerlaw<3> )
```

のように diskbb というモデルが 2 番目の成分として足される。

delcomp

上記のモデルに対して

```
XSPEC> del 2
```

とすれば、

Model: wabs<1>(powerlaw<2>)
のように diskbb が消されます。

editmod

もう少し複雑な場合、たとえば、

```
Model: wabs<1>( diskbb<2> + powerlaw<3> + gaussian<4>)
```

にたいして、diskbb成分とpowerlaw成分にのみ吸収端 (edge) をかけたいような場合、editmodのコマンドが便利である。

```
XSPEC> editmod wabs((diskb + power) edge + gau)
```

と打ってみると

```
XSPEC>editmod wa((diskb+pow)*edge+gau)
```

```
Model: wabs<1>( ( diskbb<2> + powerlaw<3> )edge<4> + gaussian<5> )
```

Input parameter value, delta, min, bot, top, and max values for ...

```
7      0.05      0      0      100      100
```

```
6:edge:edgeE>
```

```
1      0.01      0      0      5      10
```

```
7:edge:MaxTau>
```

```
-----  
-----  
Model: wabs<1>( ( diskbb<2> + powerlaw<3> )edge<4> + gaussian<5> )
```

...

のように、新しい成分のパラメータのみを聞いてくれる。

8.3.9 画面表示 etc. (Display etc.)

```
XSPEC> plot ?
```

と打てば、何が表示できるかわかる。以下のようなものが一般的。

```
XSPEC>pl ld delc      :スペクトル、モデルとの residual を表示。  
XSPEC>setplot add    :モデルが2成分以上ある場合に全ての成分を表示する (setplot noad で戻る)。  
XSPEC>pl ld ra      : スペクトル、モデルとの比を表示。  
XSPEC>pl u          : unfolded spectrum を表示 (縦軸は photon flux)  
XSPEC>pl eeu        : unfolded spectrum を表示 (縦軸は F )  
XSPEC>pl mo         : モデルを表示  
XSPEC>pl eemo       :モデルを F で表示
```

8.3.10 作業ファイルを保存 (Save files)

xspec の中で、「save option file 名」とうつ。option によって保存される内容が変わる。

```
XSPEC>save all outfile.xcm      :ファイル、レスポンス、モデルすべてを outfile.xcm というファイルに保存  
XSPEC>save model outfile.xcm    :モデルのみを outfile.xcm というファイルに保存  
XSPEC>save files outfile.xcm    :ファイルのみを outfile.xcm というファイルに保存
```

次からは、xspec を立ち上げて、

```
XSPEC>@outfile.xcm
```

とすれば良い。

8.3.11 QDPでスペクトルをハードコピー (How to save hard copies via QDP)

例えば以下のような感じにやればよい。より美しい表示をしたければ、後述の QDP のコマンド概要 (9.5) を参照すること。

```
XSPEC>iplot      <--- QDP の環境に入る。
PLT> plot
PLT> la ro       <--- 縦軸の数字を回転。
PLT> fo ro       <--- フォントを roman にする。
PLT> cs 1.5      <--- ラベルのサイズ
PLT> hard spec.ps/cps
      <--- spec.ps という postscript ファイルに出力。( /cps はカラーで表示の意味。 /ps とすればモノクロ)
PLT> wd spec     <--- spec.qdp ができる。
PLT> wh spec     <--- spec.pco ができる。
PLT> q
XSPEC>
```

8.3.12 その他 (Others)

どんなモデルがあるか? (List of Models)

```
XSPEC>model ?
```

Possible additive models are :

```
apec      bapec      bbody      bbodyrad  bextrav   bextriv   bknpower   bkn2pow
bmc       brems      bvapex     c6mekl    c6pmekl   c6pvmkl   c6vmekl   cemekl
cevmkl    cflow     compbb     compLS    compPS    compST    compTT    cutoffpl
disk      diskbb    diskline   diskm     disko     diskpn    equil     expdec
ezdskbb   gaussian  gnei       grad      grbm      kerrbb    kerred    laor
lorentz   meka      mekal      mkcflow   nei       npshock   nsa       nteea
pegpwlw   pextrav   pextriv    plcabs    powerlaw  posm      pshock    raymond
redge     refsch    sedov      smaug     srcut     sresc     step      vaped
vbremss   vequil    vnei       vmeka     vmekal    vmcflow   vnei      vnpshock
vpshock   vraymond  vsedov     zbody     zbremss   zgauss    zpowerlw  eqpair
eqpail    eqpait    compPS     thCompml  thComp1   thComp2   sxcomp    thComp0
smemm     xsgacp    dkbbth     dkbbfth   dkbbnth   dkbbbeq   bplc      photemis
swindemm  adisk2    atable
```

Possible multiplicative models are :

```
absori    acisabs   constant  cabs      cyclabs   dust      edge      expabs
expfac    gabs     highecut  hrefl     notch     pcfabs    phabs     plabs
pwab      redden   smedge    spline    SSS_ice   TBabs     TBgrain   TBvarabs
uvred     varabs   vphabs    wabs      wndabs    xion      zedge     zhighest
zpcfabs   zphabs   zredden   zTBabs    zvarabs   zvfeabs   zvphabs   zwabs
zwndabs   smfix    smfix2    smfix3    smfix4    smfixn    smabs     nick
warmabs   swindabs xisabs    mpowerla  mtable    etable
```

Possible mixing models are :

```
ascac     project   xmmpsf
```

Possible convolution models are :

```
gsmooth   lsmooth  reflect   rgxsxsrc  reflxion  reflxi2   reflxil   reflbal
reflbal2  reflbal3  ref2      conline   nslines
```

Possible pile-up models are :

```
pileup
```

Additional models are available at :
legacy.gsfc.nasa.gov/docs/xanadu/xspec/newmodels.html

どういうモデルか? (What kind of models?)

(例) powerlaw モデルのヘルプを試みる。

```
XSPEC>help models pow
      XSPEC_11.3_commands
      Models
      powerlaw
```

Simple photon power law.

```
-----
A(E) = K (E/1 keV)**(-par1)
-----
```

where :

```
par1  = photon index of power law (dimensionless)    <--- パラメータ1は光子指数
K      = photons/keV/cm**2/s at 1 keV.                <--- normalization(パラメータ2)の定義
```

```
help>      <--- リターンでもとにもどる。
XSPEC>
```

いろいろなコマンド (Summary of commands in XSPEC)

help commands とすると使えるコマンドの一覧が出てくる。

```
XSPEC>help commands
```

```
      XSPEC_11.3_commands
      Commands

'abund'   - Set the Solar abundance table.
'addcomp' - Add a component to the model.      <--- モデル成分を加える
'arf'     - Read an auxiliary response file.   <--- arfを読み込む
'autosave' - Periodically save XSPEC status.
'backgrnd' - Reset the files to be used for background subtraction. <--- バックグラウンドを
読み込む

'bayes'   - Set up for Bayesian inference.
'chatter' - Control the verbosity of XSPEC.
'corfile' - Reset the files to be used for background correction.
'cornorm' - Reset the normalization to be used in correcting the
           background.
'cosmo'   - Set H0 and q0.

'cpd'     - Alias for 'setplot device'.
'data'    - Input one or more pha data files.   <--- dataを読み込む
'delcomp' - Delete a component from the model. <--- モデル成分を取り除く
'diagrsp' - Diagonalise the current response matrix for ideal response.
'dummyrsp' - Create a 'dummy' response, covering a given energy range.
           <--- 仮のレスポンスを作る。bolometric fluxを求めたいときなどに使う。
```

'dump' - Write out history packages of the observed spectrum and model.
 'editmod' - Add, delete or change one component in the model.
 'eqwidth' - Calculate a model component's equivalent width.<--- line などの等価幅を求める。
 'error' - Determine a single parameter confidence region. <--- パラメータのエラー。
 'exec' - Execute a shell command from within XSPEC.

 'exit' - Wind up any hardcopy plots and exit from XSPEC. <--- xspec から抜ける
 'extend' - Extend the energy range over which models are calculated.
 <--- エネルギー領域を使用しているスペクトルの範囲以上にのばす。
 convolution model などを使うときに重要。
 'fakeit' - Produce simulated data files for sensitivity studies.
 <--- モデルとレスポンスに基づいてシミュレーション。
 'fit' - Find the best fit model parameters. <--- フィットする
 'flux' - Calculate the current model's flux over an energy range. <--- flux を求める。

 'freeze' - Do not allow a fit parameter to vary during the fit. <--- パラメータを固定
 'ftest' - Calculate the F statistic and probability. <--- F 検定の値を計算
 'gain' - Perform a simple modification of the response gain.
 'genetic' - Set parameters for the genetic algorithm.
 'goodness' - Monte Carlo calculation of goodness-of-fit

 'hardcopy' - Print the current plot to your default printer.
 'help' - Obtain help on XSPEC commands.
 'identify' - List possible lines in the specified energy range.
 'ignore' - Ignore a range of pha channels in future fit operations. <--- 無視するエネルギー範囲の設定
 'improve' - Try to find a better minimum (requires MINUIT).

 'iplot' - As plot command but interactive using PLT. <--- QDP に入る
 'log' - Open the log file to save output. <--- ログファイルに落とす
 'lumin' - Calculate the current model's luminosity over a given rest frame energy range and redshift.
 'margin' - Calculate the multi-dimensional probability distribution
 'method' - Set the minimization method.

 'model' - Define the model to be used when fitting the data.
 'newpar' - Modify the model parameters.
 'notice' - Restore a range of pha channels for future operations. <--- エネルギー範囲の設定

 'pause' - Exit via the FORTRAN PAUSE statement.
 'plot' - Plot various information on the current plot device.

 'query' - Switch on/off prompt to continue fitting. <--- fit の際の答えを yes/no に設定。

 'quit' - An alias for 'exit'.
 'readline' - Enable/disable the gnu readline command editing/history facility.
 'recornrm' - Adjust correction norms to minimize the fit statistic, holding the model fixed.
 'renorm' - Adjust the model norms, and/or allow automatic renorming. <--- モデルの normalization を変える

 'response' - Reset the files used to determine the detector responses.
 <--- レスポンスを読み込む
 'save' - Save aspects of the current state to a command file. <--- xcm file に保存。

 'script' - Open the script file to save all commands input.

 'setplot' - Modify the plot device and other values used by the plot routines.

```

'show'      - Display current file and model information.

'source'    - Execute a script file.
'statistic' - Change the fit statistic in use.
'steppar'   - Step through a range of parameter values; perform a fit    <--- パラメータを
               at each step.          ステップさせ、それぞれについて 自乗値を計算。
'suggest'   - Make a suggestion or report a bug to the XSPEC gnomes.
'syscall'   - Execute command in a shell.

'systematic' - Set the model systematic error.
'tclout'    - Write xspec information to a Tcl variable.
'thaw'      - Allow a fit parameter to vary during the fit. <--- パラメータを freeにする (<-->freez)
'thleqw'    - Calculates expected fluorescent line equivalent width.
'time'      - Display elapsed time and other statistical information.

'uncertain' - Alias for 'error'.
'untie'     - Untie linked parameters.
'weight'    - Change the weighting function used for chi-squared fits.
'xhistory'  - Open a history file, in order to save fit results.
'xsect'     - Changes the absorption cross-sections in use.

```

第9章 便利な FTOOLS (USEFUL FTOOLS)

9.1 fhhelp

fhhelp は ftools の man のようなものであると思えばよい。ftool でこんなことがしたいが、どういうコマンドを使ってよいかわからないという場合には、

```
fhhelp ftools
```

と打つと、コマンドリストとその簡単な説明が表示される。また、個々の ftools の使いかたなどを知りたいときには

```
fhhelp (コマンド名)
```

とすればよい。

9.2 addarf / addrmf

arf / rmf ファイルを重みをつけて足すことができる。

```
addarf 'foo.arf goo.arf' '0.5 0.5' hoo.arf
addrmf foo.rmf,goo.rmf 0.5,0.5 hoo.rmf
```

とすると foo.arf と goo.arf に同じ (0.5 と 0.5) 重みをつけて足し hoo.arf に書き出すことができる。また foo.rmf と goo.rmf に同じ (0.5 と 0.5) 重みをつけて足し hoo.rmf に書き出すことができる。

9.3 fdump

fits file を ASCII として書き出すことができる。

```
fdump infile outfile columns rows
```

というかたちで使う。columns と rows には、出力したい columns と rows の名前を指定できる。最も簡単な使いかたは

```
fdump file.fits
```

とし、出力先やどの column, row をとりだすのかきかれる度に答えればよい。特定の extension だけ取り出したいときには (ここでは 2 を取り出す)

```
fdump file.fits[2]
```

などとすればよい。実行すると

```
Name of optional output file[STDOUT]
```

ときかれるので、ここで出力するファイル名を指定できる。STDOUT とすれば、標準出力に書き出される。

9.4 mathpha / grppha

第8章に詳しく書かれているのでそちらを参照。

9.5 QDP のコマンド 概要 (Overview of QDP commands)

簡単なコマンドの実行例を示します。詳しくは正規のマニュアルを参照¹。

9.5.1 ファイルの構成 (Files)

qdp のファイルはデータファイル (file.qdp) と書式ファイル (file.pco) からなる。file.qdp の中身は”X Y1 Y2...”とか”X Xerr Y Yerr”とか”X X+err X-err Y...”の書式になっている。いくつかの例をあげておく。

例 1) X Xerr Y の場合の file.qdp

```
read serror 1
1 0.1 2
2 0.15 1.8
3 0.1 2
....
```

例 2) X Xerr Y1 Y1+err Y1-err Y2

```
read serror 1
read terror 2
1 0.1 2 0.1 -0.2 3
2 0.15 1.8 0.1 -0.15 2.5
3 0.1 2 0.1 -0.2 2.8
.....
```

例 3) X Xerr Y で二つのデータセットがある場合。no で区切られる。

```
read serror 1
1 0.1 2
2 0.15 1.8
3 0.1 2
....
no no no
1 0.1 3
2 0.2 2
3 0.12 4
4 0.1 5
```

9.5.2 起動と終了 (Start and End)

```
$ qdp file.qdp
PLT> @file.pco      <--- 書式ファイルの読み込み (あれば)
PLT> plot           <--- プロット
PLT> quit           <--- 終了
```

¹<http://heasarc.gsfc.nasa.gov/docs/software/ftools/others/qdp/qdp.html>

9.5.3 全体 (Overview)

```
PLT> plot <--- プロット
PLT> skip on <--- no で区切られたデータセットの番号が 1,2,3 の順番になる。
PLT> view 0.1 0.1 0.9 0.9 <--- 画面表示の範囲指定。(x0 y0 x1 y1 の順番)
```

9.5.4 色 (Colors)

```
PLT> color ?
PGPLOT colors are:
  0=Black,    1=White,    2=Red,    3=Green,
  4=Blue,     5=Light blue,  6=Magenta,  7=Yellow,
  8=Orange,   9=Yel.+Green, 10=Green+Cyan, 11=Blue+Cyan,
 12=Blue+Mag, 13=Red+Mag,   14=Dark Grey, 15=Light Grey
PLT> color 3 on 4 <--- 4 番目のデータセットの色を緑にする。
```

9.5.5 シンボル (Symbols)

```
PLT> marker ? <--- シンボルの番号の対応が表示される。
PLT> marker 17 on 4 <--- 4 番目のデータセットのマークを にする。
```

9.5.6 ライン (Lines)

```
PLT> line on 2 <--- 2 番目のデータセットをラインでつなぐ。
PLT> lstyle ?
PGPLOT line styles are:
  1=Solid, 2=Dash, 3=Dash-dot, 4=Dot, 5=Dash-dot-dot-dot
PLT> lstyle 1 on 2 <--- 2 番目のデータセットのラインの種類は solid
```

9.5.7 ラベル/タイトル (Labels/Titles)

```
PLT> label x "xtitle" <--- x 軸のタイトル
PLT> label y "yttitle" <--- y 軸のタイトル
PLT> label t "top title" <--- 図のタイトル
PLT> label f "filename" <--- ファイル名を表示させたいければ
PLT> label pos y 2.5 <--- y 軸のタイトルの位置を指定
PLT> window 1 <--- 1 番目の window (もともと一つなら関係ない)
PLT> label 1 p 1.5 2.5 "hoge" jus left col 3 cs 1.5
<--- 1 番目のラベル (hoge) を (x,y)=(1.5, 2.5) に貼る。(左揃え、サイズは 1.5、色は 3)
PLT> label 2 p 1.5 1 " " li 90 1 ls 1
<--- 2 番目のラベルは (x,y)=(1.5, 1) から垂直に上にのびる直線。
PLT> csize 1.5 <--- ラベルのサイズの変更
PLT> label rotate <--- 縦軸の数字を回転。
PLT> font roman <--- フォントを roman にする。
```

9.5.8 ファイル出力関連 (File input/output)

PLT> hardcopy output.ps/cps

<--- output.ps という postscript ファイルに出力。(/cps はカラーで表示の意味。 /ps とすればモノクロ)

PLT> wdata output <--- output.qdp ができる。

PLT> whead output <--- output.pco ができる。

第10章 「すざく」に固有な FTOOLS (*Suzaku* FTOOLS)

ここでは、「すざく」の解析に固有かつ重要な FTOOLS を紹介する。

10.1 xisrmfgen

XIS のエネルギー分解能、gain などは、観測時期・観測モード・CCD 上のイベントの位置などによって異なる。従って、XIS の channel 波高値 (PHA) とエネルギー (E) との対応づけが必要である。生の PHA から gain を 3.65 eV/ch に揃えた PI (Pulse Invariant) への変換は、xispi という critical ftools によってユーザーに渡る Second Fits File の時点ですでに補整されている。この PI と E との関係は rmf で記述されており (§3.4.2 参照)、あるエネルギー E の光子が入射したときの PI の分布を行列としておさめられている。xisrmfgen は、xselect で作成したスペクトルファイルに対応する rmf file を作成してくれる ftools である。ここでは最も簡単な使い方の例を説明する。詳しくは

```
> fhelptools xisrmfgen
```

を参照してほしい。

1. 準備するもの
 - .pha file (スペクトルファイル: ここでは tmp.pha とする)
 - CALDB の設定 (§A 参照)
2. 生成されるもの
 - .rmf file (レスポンスファイル: ここでは tmp.rmf とする)
3. 使用方法

```
> xisrmfgen phafile=tmp.pha outfile=tmp.rmf
```

10.2 xissimarfgn

天体からの X 線のフラックスを正確に求めるには、正確な (望遠鏡を含む) 検出器の有効面積 (arf) が必要である。XIS の場合、XIS/HXD nominal position で観測された点源については、CALDB で arf が公開されている。しかし、

- XRT の有効面積が光軸からの距離に依存すること。
- 積分領域が小さくなると、積分領域の外にでてしまう X 線光子の割合が増えるため、有効面積が減少すること。
- XIS の OBF 上に附着している汚染物質 (contamination) の影響 (空間的に非一様でかつ時間変化) が CALDB の arf には含まれていないので自分で `xiscontamicalc` を使って計算する必要があること。
- 広がった天体の場合には、そもそも有効面積の計算は単純ではないこと。

などの理由から、自分の解析しようとする天体にあわせて arf を計算した方が多い場合が多い。キャリブレーション情報をもとに、モンテカルロ計算を行なって arf を求める tool が `xissimarfgn` であり、ここではその使い方を簡単に解説する。`xissimarfgn` は、HEADAS 6.1.1 以降には正式に HEADAS に含まれている。すでにながりのパラメタが CALDB やスペクトルファイルから自動で読み込まれるようになってるので、CALDB を正しくインストールしておかなければならない。

なお、`xissimarfgn` を使おうと思う人は、Ishisaki et al. 2007, PASJ, 59, S113 を必ず参照すること。また、各パラメタの意味については

```
unix> fhelpt xissimargen
```

を確認して欲しい。参考のため、以下に簡単な arf 計算の計算例を紹介しておく。

その 1 : 点源のスペクトルを SKY 座標で積分した場合の arf

点源の Sky 座標を (SKY, SKY) = (700, 800) とし、そのスペクトルを (SKYX, SKYY) = (700, 800) を中心に、半径 115 pixel (約 2 arcmin) で積分した場合の XIS1 の arf (`source_xis1.arf`) を作る。積分領域ファイル (`source_xis1.reg`¹) スペクトルファイル (`source_xis1.pi`)、レスポンスファイル (`source_xis1.rmf`)、GTI ファイル (`source_xis1.gti`, GTI extension のついたイベントファイルでも良い)²、attitude ファイル (配付データの `auxil/` 以下にある。とりあえず `source.att` とする) を用意する。

ここで `xissimarfgn` で使う座標には、bin まとめていない状態のイメージの座標を指定しなくてはならないことに注意が必要である。XSELECT 中でデフォルトでは、XIS のイメージは 8 bin まとめられているため、そこで作ったイメージから座標を決めると、`xissimarfgn` では座標や半径などの値が正しい値の 1/8 になってしまうのである。正しくは例えば XSELECT 中で `set xybinsize 1` と打ってイメージの bin サイズを変更し、その後 `extract` して作ったイメージで座標を決めればよい。

以上のファイルの準備ができたなら以下のように実行する。

```
unix> xissimarfgn \  
  instrum=XIS1 source_mode=SKYXY pointing=AUTO source_x=700 source_y=800 \  
  num_region=1 region_mode=SKYREG regfile1=source_xis1.reg arffile1=source_xis1.arf \  
  limit_mode=MIXED num_photon=100000 accuracy=0.005 phafile=source_xis1.pi \  
  detmask=/CALDB/data/suzaku/xis/bcf/ae_xi1_calmask_20051105.fits \  
  gtifile=source_xis1.gti attitude=source.att rmffile=source_xis1.rmf estepfile=default
```

¹text 形式で `circle(700,800,115)` と書いておけば良い

²GIT ファイルを入力しない場合には、`date_obs=2006-03-01T12:00:00` のように日時を指定する。

その 2 : 一様な輝度分布の広がった天体に対する arf

天体の大きさを半径 20 arcmin の円とし、明るさは一様とする。2 通りの積分領域で XIS2 の arf を作る。出力される arf は天体全体のフラックスを求めることができるように規格化されているので、積分領域に比べて天体が大きい場合には注意すること。

```
unix> xissimarfgn instrume=XIS2 pointing=AUTO source_mode=UNIFORM source_rmin=0 \
      source_rmax=20 num_region=2 region_mode=SKYREG \
      regfile1=source_xis2_a.reg arffile1=source_xis2_a.arf \
      regfile2=source_xis2_b.reg arffile2=source_xis2_b.arf \
      limit_mode=MIXED num_photon=1000000 accuracy=0.005 phafile= source_xis2.pi \
      detmask=/CALDB/data/suzaku/xis/bcf/ae_xi2_calmask_20051105.fits \
      gtifile=source_xis2.gti attitude=source.att rmffile=source_xis2.rmf estepfile=medium
```

ここで estepfile はどの位細かいエネルギーステップで arf の計算を行なうかを表わすパラメタであり、full, dense, medium, sparse の 4 通りが用意されている。まず最初に、比較的是やく計算が終了する estepfile=sparse で設定などの確認を行なってから、好みの estepfile で計算することを勧める。

特に自分で指定をしないかぎり、xissimarfgn では contamination 考慮した arf が作られる。このとき、最新の HEADAS と CALDB を正しくインストールしてあれば、最新の contamination 情報が CALDB から読みだされる³。もし、この影響を考慮しない arf を作りたいのであれば、contamifile=none と設定する必要がある。また、ここでファイル名を直接指定すれば、古い(とはかぎらないが...)バージョンの contamination 情報をもとに arf を作ることも可能である。

Source の位置、積分領域の位置が適切かどうかを確認するには、ds9 で arf の最初の extension に入っているイメージを見るのが非常に有効である。

³xissimarfgn の parameter file (おそらく /pfiles/xissimarfgn.par) をみればわかるが contamifile,f,h,"CALDB" ,,, "Name of XIS contamination file" のように、デフォルトで CALDB を読みに行くようになっている。

10.3 xiscontamicalc

XIS の Contamination を含んでいない arf がすでにあるときには⁴、xissimarfgn によって Contamination を含んだ arf を作りなおさなくても、xiscontamicalc を使えば比較的簡単に contamination の評価が出来る。xiscontamicalc には、以下に示すような 2 通りの使い方がある。

その 1 : XSPEC で使う補正因子を計算する

以下の [] の部分に対して、自分の観測に対応したパラメタを入力する。

```
unix> xiscontamicalc
xiscontamicalc version 2006-11-26
Written by Y.ISHISAKI (TMU)
Built on ANL HEADAS converter 1.71 for ANL version 1.72
Instrument Name (XIS0,XIS1,XIS2,XIS3) [XIS0]
Suzaku TIME, or date string in UTC, 'yyyy-mm-ddThh:mm:ss.sss' [2006-03-01T00:00:00]
X-ray energy in keV[1]
incident X-ray DETX position[310]
incident X-ray DETY position[512.5]
ARF file name to multiply the transmission[none]
```

以下の様な出力が出る。XSPEC で スペクトルフィッティングを行なう際に、vphabs を使って、C(CARBON) と O(OXYGEN) の吸収量を補正すれば良い。

```
xis_contami: reading '/inst/soft/caldb//data/suzaku/xis/bcf/ae_xi0_contami_20061016.fits' ...
  ngp=522, t0=177242000.0, t1=492342800.0, nen=7900, e0=0.201, e1=15.999
```

```
TELESCOP = 'SUZAKU'
INSTRUME = 'XIS0'
CONTAMIFILE = '/inst/soft/caldb//data/suzaku/xis/bcf/ae_xi0_contami_20061016.fits' (CALDB)
LEAPFILE = '/inst/soft/caldb//data/gen/bcf/leapsec_010905.fits' (CALDB)
DATE-OBS = '2006-03-01T00:00:00' ( TIME = 194486401.000000 )
DAYS = 199.588 [dy] after 2005-08-13T09:53:20
ENERGY = 1.000 [keV]
R_OFFSET = 3.517 [arcmin] at ( DETX , DETY ) = ( 310.0 , 512.5 ) [pixel]
CARBON = 1.877231 [10**18 cm**-2]
OXYGEN = 0.312872 [10**18 cm**-2]
TRANSMIS = 8.843071e-01
```

その 2 : arf を 補正する

上の例の入力の最後の行で、Contamination を含んでいない arf を入力する。

```
ARF file name to multiply the transmission[none] source_nocontami.arf
scale column density of contaminant from nominal value?[no]
```

これで、入力した arf に対して、補正が行なわれる⁵。

⁴例えば CALDB には、XIS/HXD nominal position に対する arf が用意されている。

⁵入力 arf に上書きとなる。現状では、何度でも xiscontamicalc をかけることが、ソフトウェアの上では可能なので、そういうことのないように各自注意すること

10.4 xissimを用いた exposure map 作成

xissim は、XIS image などのシミュレーションを行なうことができる ftools である。例えば以下のような場合、xissim は有用である。広がった天体のイメージを描きたい時、視野中心と視野の端では vignetting の影響で実効的な exposure は異なるため、生イメージでは正しい強度 map にはなっていない。そこで、xissim を用いて視野内の exposure の map を作成することで、イメージの exposure 補正をすることが可能である。ここでは、xissim を用いた XIS0, 2.0 keV での exposure map の作り方を紹介する。

0. 必要な物

元の image fits: input.img

オイラー角情報: (=ea1, ea2, ea3)

オイラー角情報は、イベントファイルにも書いてあるので、fdumpなどで調べよう。

レスポンスファイル: (ここでは xis0.rmf)

1. 空が一様に光っている image fits を用意

元の image fits を input.img、一様な空 image fits を flatinput.img とする。

```
fcarith infile="input.img" const="0" outfile="tmp.img" ops="*"
fcarith infile="tmp.img" const="1" outfile="flatinput.img" ops="+"
```

これで全ピクセルに 1 が入った image fits が作成された。

2. 2 keV の photon list を作成

```
mkphlist photon_flux=1 flux_emin=1.0 flux_emax=10.0 \
spec_mode=1 image_mode=0 time_mode=0 limit_mode=0 \
fits_image_file=flatinput.img nphoton=200000000 \
energy=2.0 outfile=flat_2keV.pht
```

mkphlist とは、xissim のための photon list を作成する ftool である。詳しくは

```
>fhhelp mkphlist
```

で勉強してほしい。

3. photon list のイベントを空 image fits に降らせてイベントファイルを作成

```
xissim instrume=XIS0
infile1=flat_2keV.pht infile2=none \
ea1=999.9 ea2=999.9 ea3=999.9 \
xis_rmffile=xis0.rmf \
outfile=flat.evt
```

4. xselect などで image fits を作成

第 5 章などを参照して、image fits を作る。これが exposure map と等価である。絶対値は違うことに注意すること。

なお、2008 年 3 月現在、xisexpmapgen という FTOOLS がリリースされている。これは、exposure map を direct に作ってくれるツールである。

10.5 hxdarfgn

HEADAS 6.0.6 以降には HXD の arf file を作成する hxdarfgn なる ftools が含まれている。これは、HXD nominal position のレスポンスに対して用いる arf を作成するツールである。(XIS nominal position のレスポンスに対して用いてはダメ)

正確な表現をするならば、HXD のエネルギー応答関数は、入射角度が変わると、パルスハイト分布 (rmf) の形自体が変わるので、rmf × arf という体系では表現しきれないものであるが、第零近似として、arf を用いた解析が行なえるようになっている。HXD の arf/rmf は、通常の X 線望遠鏡 (arf) → 焦点面検出器 (rmf) という切り分けは行なっておらず、HXD nominal position の時に arf の全ての成分が 1.0 となるように定義されている。すなわち、HXD arf は、HXD nominal position のレスポンスからの補正項として定義されている。

hxdarfgn の詳細な使い方は、

```
unix $ fhelph hxdarfgn
```

の記述をよく読んでほしい。

ae101005040.hxd_pin.pha という PIN の PI file を用いていて、(RA, DEC) = (274.0554, 49.8675) の天体に対する PIN の arf が欲しい場合は、

```
unix $ hxdarfgn hxd_arf_pinid=64 hxd_arf_gsoid=17 \  
teldef_name="ae_hxd_teldef_20060516.fits" \  
attitude_name="ae101005040.att" \  
hxd_arfdb_name="ae_hxd_pinart_20051126.fits" \  
input_pi_name="ae101005040_hxd_pin.pha" \  
point_ra=274.0554 point_dec=49.8675
```

とすればよい。ここで hxd_arf_pinid=64 なら PIN-0,1,2,...63 全てをマージしたものである。PIN arf を作成する場合は、hxd_arf_gsoid を 16 より大きい値にする必要がある。衛星の向いていた RA, DEC は、PI file のヘッダに書かれている 'RA_NOM' 'DEC_NOM' から読まれる⁶。また、ae101005040.hxd_gso.pha という GSO の PI file に対する GSO の arf が欲しい場合は、

```
unix $ hxdarfgn hxd_arf_pinid=65 hxd_arf_gsoid=16 \  
teldef_name="ae_hxd_teldef_20060516.fits" \  
attitude_name="ae101005040.att" \  
hxd_arfdb_name="ae_hxd_gsoart_20051126.fits" \  
input_pi_name="ae101005040_hxd_gso.pha" \  
point_ra=274.0554 point_dec=49.8675
```

とすれば生成される。ここで、CALDB 中の arf data base の名前 (hxd_arfdb_name) は、

```
PIN なら ae_hxd_pinart_XXXXXXXXX.fits、  
GSO なら ae_hxd_gsoart_XXXXXXXXX.fits
```

としないと正常に動作しないので注意が必要である。

⁶V1.x 仕様。将来的には attitude file から計算する

付録A 便利な Web page (USEFUL WEB PAGES)

ここでは、このマニュアルの中に出て来た Web page をはじめとして、「すざく」のデータ解析をする上で知っている便利な Web page を挙げておく。

「すざく」に関するページ

「すざく」プロジェクトのサイト (宇宙研)

<http://www.astro.isas.ac.jp/suzaku/index.html.ja>

「すざく」プロジェクトのサイト (GSFC)

<http://suzaku.gsfc.nasa.gov/>

「すざく」ヘルプデスク

理研 (日本語)

http://cosmic.riken.jp/suzaku/help/index_j.html

理研 (英語)

<http://cosmic.riken.jp/suzaku/help/>

GSFC

http://suzaku.gsfc.nasa.gov/docs/suzaku/astroe_helpdesk.html

FAQ ページ

理研 (英語)

<http://cosmic.riken.jp/suzaku/help/faq/index.html>

GSFC

http://suzaku.gsfc.nasa.gov/docs/suzaku/helpdesk/analysis_faq20070101.html

「すざく」の較正情報/データプロセスに関する情報

較正情報 (宇宙研)

<http://www.astro.isas.jaxa.jp/suzaku/caldb/>

較正情報 (GSFC)

<http://suzaku.gsfc.nasa.gov/docs/heasarc/caldb/suzaku/>

CALDB の設定方法 (GSFC)

http://suzaku.gsfc.nasa.gov/docs/heasarc/caldb/caldb_install.html

データプロセス

<http://www.astro.isas.jaxa.jp/suzaku/process/>

v1.2, v1.3 データのファイル名等の定義に関する詳細 (宇宙研, GSFC)

http://darts.isas.ac.jp/astro/suzaku/seq_data.html.v1.2

http://darts.isas.ac.jp/astro/suzaku/seq_data.html.v1.3

http://suzaku.gsfc.nasa.gov/docs/astroe/archive/astroe_icd_sdc_v1.2.pdf

データのダウンロード

DARTS (宇宙研)

<http://darts.isas.jaxa.jp/astro/suzaku>

HEASARC (GSFC)

<ftp://legacy.gsfc.nasa.gov>

HXD の NXB

<http://www.astro.isas.jaxa.jp/suzaku/analysis/hxd/hxdnxb/>

ソフトウェアのダウンロード

HEASoft

<http://heasarc.gsfc.nasa.gov/docs/software.html>

ds9

<http://hea-www.harvard.edu/RD/ds9/>

xissimarfgen の最新版

<http://www-x.phys.metro-u.ac.jp/~ishisaki/xissim/>

ソフトウェアのマニュアル

HEADAS

<http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/>

qdp

<http://heasarc.gsfc.nasa.gov/docs/software/ftools/others/qdp/qdp.html>

xspec (インストールとセットアップ)

<http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/install.html>

xspec

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

xronos

<http://heasarc.gsfc.nasa.gov/docs/xanadu/xronos/xronos.html>

xselect

<http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/ftools/xselect/xselect.html>

xissimarfgen

<http://heasarc.gsfc.nasa.gov/docs/suzaku/analysis/xissimarfgen/>

その他

観測ログ

http://darts.isas.ac.jp/astro/tables/SUZAKU_LOG.html

観測ログ検索

<http://darts.isas.ac.jp/suzakulog/top.do>

ISAS PLAIN センター analysis server について

http://plain.isas.jaxa.jp/ana_servers/

WAM のページ

<http://www.astro.isas.jaxa.jp/suzaku/HXD-WAM/WAM-GRB/>

付録B ソフトウェアのインストール・設定

(SOFTWARE INSTALLATION)

この章では解析に必要なソフトのインストール方法を説明する。B.1 と B.2 でそれぞれ HEASOFT と ds9 のインストール方法について述べる。これらのソフトは HEASARC のホームページ

<http://heasarc.gsfc.nasa.gov/docs/software.html>

からダウンロードすることができる。

インストール前の確認。(上から下に向かってマニアックに...)

どこにインストールするか。

インストールするディスクに十分空きがあるか。

インストールするディストリビューション(つまりコンパイルするか、バイナリで持ってくるか)。

(バイナリ派のあなたは) インストールするマシンの OS。

(コンパイル派のあなたは) コンパイラのバージョン。

これらはあなたに解析を指導してくれる人の指示に従うべきである。が、「よくわからないけどとりあえず解析を始めてみよう(^o^)/」というあなたにはバイナリディストリビューションをお薦めする。バイナリディストリビューションの方が簡単だが(?) コンパイルすると Xspec で local model を使えるというメリットがある。

B.1 HEAsoft

HEASOFT は解析に必要な様々なソフトをまとめたものである。

<http://heasarc.gsfc.nasa.gov/docs/software.html>

で HEAsoft というリンクをクリックしてみよう。はじめの方に HEAsoft に含まれているそれぞれのソフトについて記述があるが、その下に Downloads と書いてある部分がある。大きな Download の文字をクリックするとダウンロードページに飛べる。ここからは、使用するディストリビューションによって作業が変わるので分けて説明する。ソースコード派の方は B.1.1 を飛ばして B.1.2 を読めばよい。

B.1.1 バイナリディストリビューション (Binary distribution)

バイナリの場合は簡単である。図 B.1 のステップ 1 で自分の OS を、ステップ 2 で All にチェックする (All にチェックすると自動で全部にチェックが入る)。「すざく」の解析だけを行う場合には Mission-Specific Tools 中の Suzaku をチェックするという方法もある。こうすると「すざく」の解析に必要なもの全てに自動でチェックが入る。¹そして Submit をクリックし tar でかためたファイルを適当な場所に置いて、インストールしたい場所で展開すればよい。展開するとまず headas というディレクトリができ、その中に

```
attitude/  heagen/    image/      tcltk/ demo/    heatools/  
integral/  ftools/    spectral/  Xspec/  heacore/  swift/
```

¹ただし、Xspec v11 にはチェックが入らない!!このマニュアルではいまのところ Xspec v11 を使った解析を説明しているので、これはチェックしておくべきだろう。将来的に、これらの記述は変更される可能性がある

ぐらいのディレクトリといくつかのファイルがあるはずである。さて、お手元には実は上のリストにはないディレクトリがあるはずである。例えば、linux の人は i686-pc-linux-gnu-libc2.2 とかいう名前の²ディレクトリである。このディレクトリの下にある BUILD_DIR に移動し、

```
./configure > config.out
```

と打つ。これで、インストール作業は終わりである。ただ、このままでは使えず、いくつかの設定をする必要がある。これは、B.1.3 で述べる。

ある条件のマシンでは、バイナリで持ってきた場合にうまくいかないことが分かっている。

```
locate libg2c
```

と打ってみよう。ここで、何も出力されずにプロンプトが帰って来た場合には対処が必要である。方法はこの節の最後 (B.1.4) に解決方法を載せたので、参照してほしい。

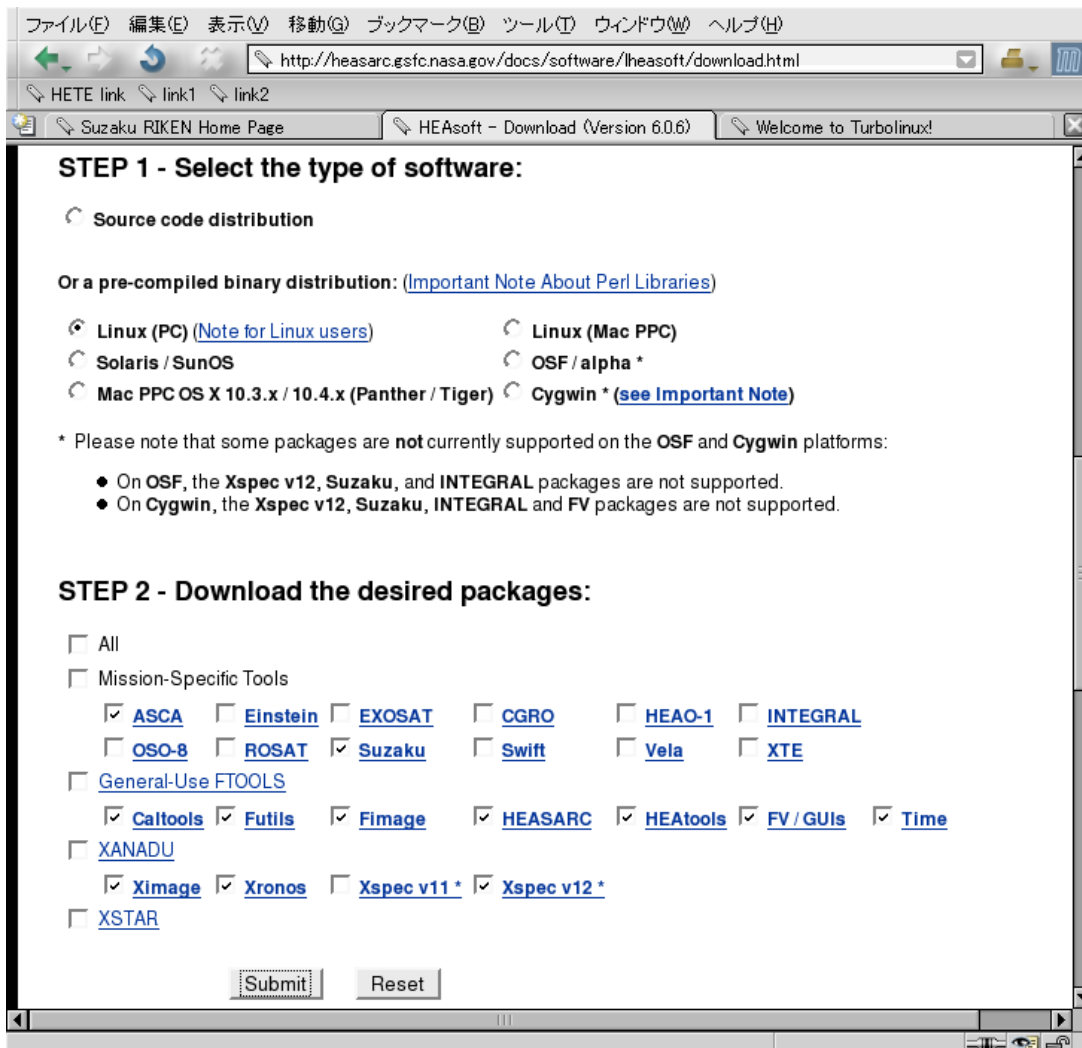


図 B.1: HEASoft ダウンロードのページ

²バージョンによって多少異なる可能性あり

B.1.2 ソースコードディストリビューション (Source code distribution)

ソースコードディストリビューションの場合は、図B.1の画面で、ステップ1はSource code distributionのボタンを、ステップ2はAllをチェックする。「すざく」の解析だけを行う場合には Mission-Specific Tools 中の Suzaku をチェックするという方法もある。こうすると「すざく」の解析に必要なもの全てに自動でチェックが入る。³そして Submit をおし、tar でかためたファイルを適当な場所に置いて、インストールしたい場所で展開すればよい。展開するとまず headas というディレクトリができ、その中に

```
BUILD_DIR/          README.CYGWIN      attitude/  heagen/    swift/
CMakeLists.txt*    Release_Notes_6.0.4  ftools/   heatools/  tcltk/
HEASOFT-INSTALL.TXT Xspec/             heacore/  integral/
```

というようなものがあるはずである。ここからは、ここにある HEASOFT-INSTALL.TXT に沿ってインストールを進めればよい。⁴

まず、展開してできた headas ディレクトリの中の BUILD_DIR に移動する。ここで

```
./configure > config.out
```

と打つ (./configure だけでも良いが、これだと後でうまくいかなかったときにどこが良くないのかわからない)。これが終わったら、次にコードのコンパイルを行う。具体的には、

```
make > build.log
```

を実行する。そして最後に

```
make install > install.log
```

ところで、HEASOFT-INSTALL.TXT というガイドに、VERY IMPORTANT と書かれている項目がある。それは、build.log や install.log にエラーが出ていないかをチェックすることである。エラーがある場合には、これらのファイルの中に "***" を含む行が存在するはずなので、これがないかをチェックするのが手っ取り早い方法である。ただし、次のふたつの場合は "***" を含むが、実質問題はない、と書かれている。

(その1)

```
read_subs.f:292: warning:
                GOTO 100
                1
read_subs.f:349: (continued):
100    call fcerr('*** FADMAP EXITED IN SUBROUTINE READ_EXPOSURE ***')
2
Reference to label at (1) is outside block containing definition at (2)
```

(その2)

³ただし、Xspec v11 にはチェックが入らない!!このマニュアルではいまのところ Xspec v11 を使った解析を説明しているの、これはチェックしておくべきだろう。将来的に、これらの記述は変更される可能性がある

⁴と書いて分かるくらいなら始めからこのセクションは読まないであろうからもう少し丁寧に解説する...

```
*** Warning: This library needs some functionality provided by -ltermLib.  
*** I have the capability to make that library automatically link in when  
*** you link to this library. But I can only do this if you have a  
*** shared version of the library, which you do not appear to have.  
*** The inter-library dependencies that have been dropped here will be  
*** automatically added whenever a program is linked with this library  
*** or is declared to -dlopen it.
```

さて、これ以外の場所に "***"が見付かったり、そもそもどこかでエラーが出て止まった場合はどうすればよいだろう。お近くの詳しくそうな人に相談するのがよからう。お近くに詳しくそうな人が見当たらなかったり、頼りなかった場合には、HEASOFT-INSTALL.TXT の始めの方にある CONTACT INFORMATION の部分を参照して、解決方法を探すこともできる。とりあえず、ソースをコンパイルするのは諦めて、バイナリを持ってくるという手もある。

無事エラーがないことが確認されればインストールはおしまいである。B.1.3 に従ってインストール後の設定を行えばよい。

B.1.3 インストール後の設定 (Setup after installation)

インストール後に必要になる手続きがある。まずあなたが HEASOFT をインストールしたディレクトリ (headas というディレクトリの親ディレクトリ) をもう一度チェックしよう。ここでは、/YOUR_DIRECTORY にインストールしたとして話をすすめる。また、headas の下にあなたの使っている OS に応じたディレクトリがあるはずだがこれもチェックしておこう。例えば、i686-pc-linux-gnu-libc2.2 alphaev56-dec-osf4.0d sparc-sun-solaris2.8 powerpc-apple-darwin7.5.0 などがそうである。ここでは YOUR_PLATFORM であるとしておく。あなたがお使いの Shell は何だろう？

```
echo $SHELL
```

とか打つと分かるかもしれない。csh や tcsh の方は、\$HOME/.cshrc に

```
setenv HEADAS /YOUR_DIRECTORY/headas/YOUR_PLATFORM  
alias heainit "source $HEADAS/headas-init.csh"
```

を追加する。sh、ash、ksh、bash の方は \$HOME/.login に (と言っている bash の私は \$HOME/.bashrc を使うが)

```
export HEADAS=/YOUR_DIRECTORY/headas/YOUR_PLATFORM  
alias heainit=". $HEADAS/headas-init.sh"
```

を書いておく。

そして、解析を始める前に "heainit" と打つと諸々の設定がなされ、解析に必要なソフトが使えるようになる、というわけである。

以上で HEASOFT のインストールの解説は終わりである。

B.1.4 ”ハマりがち”ぱたーん (A common trap)

古い linux やとても新しい linux ではバイナリを持ってきてもうまくゆかないことがある。これは実行時に起こる不具合なので、インストール時には気付かないかもしれない。例えば、Xspec を立ち上げようとすると怒られる。

```
/YOUR_DIRECTORY/headas/i686-pc-linux-gnu-libc2.2.4/bin/xspec12:  
error while loading shared libraries: libgcc_s.so.1:  
cannot open shared object file: No such file or directory
```

とか。エラーメッセージの中に libstdc++.so.5 libg2c.so.0 libgcc_s.so のいずれか（上の例では libgcc_s.so）が含まれる場合には以下の方法で対処できる。

まず、

```
http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/linux.html
```

に行く。そして "Compiler library patch for Linux" をクリックして tar ファイルをダウンロードし、headas ディレクトリで解凍すればよい。これで先程「ないよ」と言われた libgcc_s.so などが headas/i686-pc-linux-gnu-libc2.2.4/lib/ に追加される。

もし解凍したとき、あなたの \$HEADAS と異なるディレクトリができてしまった場合⁵は、これだけでは動かない。解凍してできた lib の中身を \$HEADAS/lib に移動させて初めて動く。まずは

```
echo $HEADAS
```

などとしてしっかり headas の環境が設定されていることを確認した上で

```
cd (解凍してできた)lib  
mv * $HEADAS/lib
```

を行うこれで動くようになる。空になった lib やその上のディレクトリは削除してしまって構わない。

後から追加されたディレクトリをそのまま分けておきたい場合、この場所をソフトを動かしたときに参照するように設定して解決できる。これには \$LD_LIBRARY_PATH に今持ってきた i686-pc-linux-gnu-libc2.2.4/lib も追加すれば良いわけである。使っている Shell に応じて

```
setenv LD_LIBRARY_PATH "/YOUR_DIRECTORY/headas/i686-pc-linux-gnu-libc2.2.4/lib:$LD_LIBRARY_PATH"
```

もしくは

```
export LD_LIBRARY_PATH="/YOUR_DIRECTORY/headas/i686-pc-linux-gnu-libc2.2.4/lib:$LD_LIBRARY_PATH"
```

である。

B.2 ds9

ds9 も同じく

```
http://heasarc.gsfc.nasa.gov/docs/software.html
```

から SAOImage DS9 のリンクをたどって必要なファイルをダウンロードする。とりあえず最新版をダウン

⁵インストールされている headas が微妙に古かったりするとこんなことになってしまう

ロードすればよいので、リンク先である

<http://hea-www.harvard.edu/RD/ds9/>

で最も上の方にあるバージョンを持ってくれば良い。自分の OS に合わせてバイナリをダウンロードし、解凍する。そうすると ds9 というファイルが 1 つできるので、これを PATH の通ったディレクトリに置けば良い（もしくはこれを置いたディレクトリに PATH を通せばよい）。

付録C Suzaku HelpDesk

解析に困った事があれば、「すざくヘルプデスク」に質問をなげるのも良からう。日本枠で観測提案が採択されたメンバー（日本、アジア、ヨーロッパ）は、主に日本の「すざくヘルプ」に、US 枠は NASA GSFC が運営する GOF に、質問を投げるのがよい。それぞれ、WWW Page は、

http://cosmic.riken.jp/suzaku/help/index_j.html（日本語）

<http://cosmic.riken.jp/suzaku/help/>（英語）

が日本のヘルプデスク、

http://suzaku.gsfc.nasa.gov/docs/suzaku/astroe_helpdesk.html

が NASA のヘルプデスクである（下の図 C.1）。

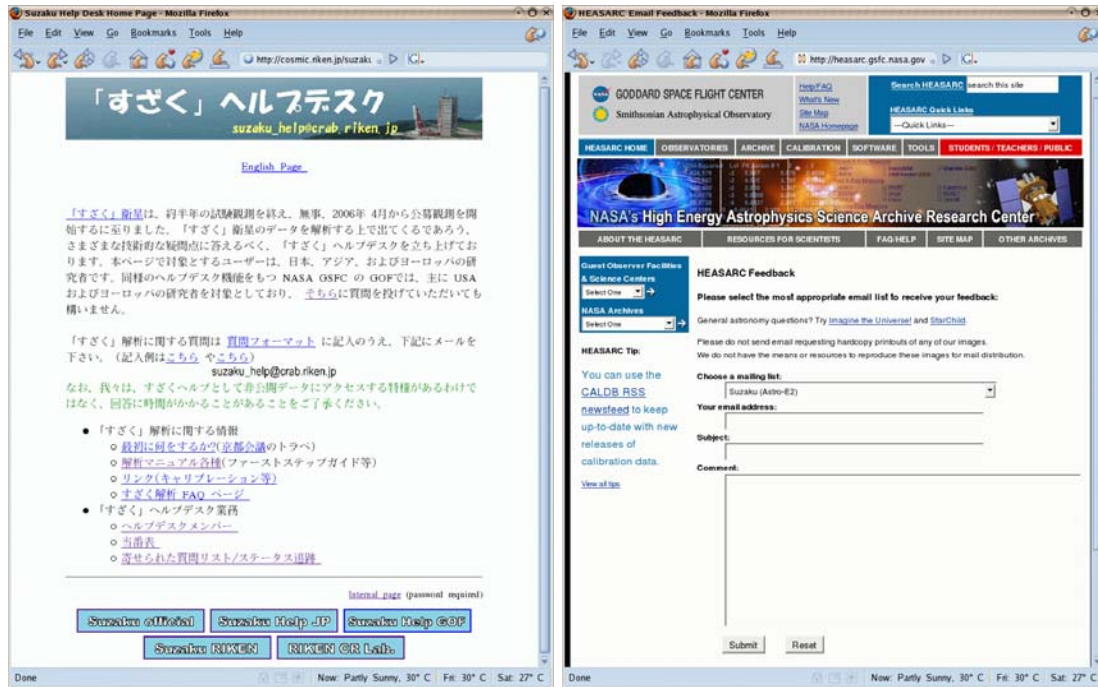


図 C.1: 理研メンバーが主となって運営している日本の「すざく」ヘルプデスク (左) と、NASA GSFC が運営するヘルプデスク (右)

以下、日本のヘルプデスクの場合を述べる。

下記のような質問フォーマットが WWW ページから取れるので、ここに質問その他を記載し、
suzaku_help@crab.riken.jp

宛にメールを投げるとよい。

***** すざくヘルプ質問票 (v070718)*****

質問者氏名 (your name):

所属機関 (your institute):

学年/職種 (your title):

質問のホームページ掲載許可 (Can this question be made public?)

[Yes/No; default Yes]:

守秘義務の有無 (Secret idea included?)

[有/無; Y/N]:

特記事項 (Additional information):

解析環境 (Analysis environment):

Linux kernel x.x.x

gcc version x.x.x

HEADAS version x.x.x

解析データ, 可能なら記載 (Favorite Target; if possible):

Obs ID:

Obs date:

Target Name:

Process Version:

HK FITS Header の 'PROCVER' を参照ください。

fkeyprint keynam="PROCVER" で確認できます。

質問事項 (Question; Free format):

質問は、時には検出器チームやオペレーションチームなどに展開する必要があり、返答に時間がかかることもある。質問後、「受理」メールが返信された時点で、このプロセスに入る。もし、「WWW に公開してもよい」質問ならば、WWW page で、質問のステータスを追跡できる。ヘルプデスク内部で質問の回答が出来そうな方がみつかれば、ステータスが「担当者決定」に、そうでなく、さらに各チームに回覧が必要となり、回覧された状態なら「回覧」となる。問題が解決されれば「解決」となり、「完了」となる。

時には、質問内容自体に、新しい科学的アイデアなどが含まれており、検出器メンバーにさえ内容をあかしたくない場合がある。その場合は「守秘義務」を YES にしておけば、すざくヘルプデスクメンバー以外には、該当部分の情報は回覧されないが、逆に、問題解決に時間を要することもあるので、よく考えて YES/NO を決定していただきたい。

付録D GLOSSARY

少しずつ書き足していきます。皆さんが初心者だった頃に困った略語を (suzaku_help@crab.riken.jp) までお寄せください。

- ARF: Aucillary Response File
- CALDB: Calibration Database
- CCD: Charge Coupled Device (BI-CCD: Back Illuminated CCD, FI-CCD: Front Illuminated CCD)
- CL: CLeaned event
- COR: Cut-Off Rigidity
- CTI: Charge Transfer Inefficiency
- CXB: Cosmic X-ray Background
- DARTS: Data ARchive and Transmission System
- FITS: Flexible Image Transport System
- FFF: First FITS File
- GO: Guest Observer
- GOF: Guest Observer Facility
- GSFC: Goddard Space Flight Center
- GTI: Good Time Interval
- HEADCC: High Energy Astrophysics Data Centers Coordination
- HEASARC: High Energy Astrophysics Science Archive Research Center
- HEAssoft: High Energy Astrophysics software
- HK: House Keeping
- HXD: Hard X-ray Detector
- NXB: Non X-ray Background
- PHA: Pulse Hight Amplitude
- PI: Pulse hight Invariant
- PSF: Point Spread Function
- QL: Quick Look

- RMF: Redistribution Matrix File
- RPT: Raw Packet Telemetry
- SAA: South Atlantic Anomaly
- SFF: Second FITS File
- SWG: Science Working Group
- WAM: Wideband All-sky Monitor
- XIS: X-ray Imaging Spectrometer
- XRS: X-Ray Spectrometer
- XRT: X-Ray Telescope

Acknowledgment

皆さん *suzaku* の解析を楽しんでいただけましたでしょうか。ファーストステップガイド初版へは、日本全国の研究室の（主に）学生のみなさんから意見を寄せていただきました。これらのフィードバック無しには、本ガイドを出版することはできませんでした。この場を借りて、改めてご協力に感謝いたします。貢献してくださった方の名前を、ひとりひとり列挙することはできませんでした。ご容赦ください。（青学大 吉田研、愛媛大 粟木研、埼玉大 田代研、中央大 坪井研、東大 牧島研、立教大 北本研、理研 牧島宇宙放射線研）

Guest Observer 向け ver.2.0 出版の際には、海老沢さん、藤本さん、山岸さん（ISAS/JAXA）、寺島さん（愛媛大）、濱口さん（GSFC）はじめ、多くの方々にご協力をいただきました。ありがとうございました。この先も休み無く改訂を続けていきます。今後とも、「すざく」に関する皆さん、「すざく」を解析している皆さんのご協力をよろしくお願いいたします。

第三版に向け、国分さん、前田さん、岡田さん、尾崎さん（ISAS/JAXA）、村上さん、海老沢さん（PLAINセンター）、石崎さん（首都大）、森さん（宮崎大）、粟木さん、寺島さん（愛媛大）、h 高橋さん（広島大）、藤本さん（金沢大）ほかの皆さんから、マニュアルの内容について、貴重なご意見をいただきました。ありがとうございました。