Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *15/01/2021* **par :**

Sujay Mate

Développement d'un simulateur du ciel pour les instruments à grand champ de vue X/gamma en orbite terrestre basse - Application à l'évaluation des performances du spectro-imageur SVOM/ECLAIRs

---

JURY

Natalie Webb	Astronome	Présidente
Melania Del Santo	Chargé de Recherche	Rapportrice
Lorezno Natalucci	Senior Researcher	Rapporteur
Dipankar Bhattacharya	Professeur	Examinateur
Philippe Laurent	Inégnieur-Chercheur	Examinateur
Stéphane Schanne	Inégnieur-Chercheur	Examinateur
Philippe Guillemot	Ingénieur de Recherche	Invité
Laurent Bouchet	Ingénieur de Recherche	Directeur de thèse
Jean-Luc Atteia	Astronome	Co-directeur de thèse

École doctorale et spécialité :
SDU2E : Astrophysique, Sciences de l'Espace, Planétologie

Unité de Recherche :
Institut de Recherche en Astrophysique et Planétologie (IRAP), UMR5277, OMP/CNRS

Directeur(s) de Thèse :
Laurent Bouchet et *Jean-Luc Atteia*

Rapporteurs :
Melania Del Santo et *Lorezno Natalucci*

Acknowledgements

This thesis is a product of three years of a great professional and personal journey. I would like to thank Laurent and Jean-Luc for giving me this opportunity. I have learnt a great deal while working on the thesis and it has laid a solid foundation for my future research endeavours.

Working in a big collaboration like SVOM was a great experience and this thesis would not have been possible without the support of some of the members I had the pleasure to work with. First of all, I would like to thank Olivier for his time to time help and advice ; and thank you Benjamin for all the discussions which have been very useful to improve some aspects of the work. I would also like to thank members of the ECLAIRs DPIX team at IRAP and CNES (Roger, Carine, Karine, Pascale, Jean-Pascal and Philippe) for their time to time help. I am grateful to ECLAIRs team members at CEA (Stéphane, Nicolas, Henri, Aleksandra and Cyril), APC (Jean-Marc, Camille and Philippe), IAP (Frédéric and Maxime) and LUPM (Frédéric and Auélia) whose help and advice was crucial to carry out the thesis. I am also thankful to all the other members of the SVOM collaboration with whom I had a great pleasure to interact.

I would like to thank all the GAHEC members and administrative staff at IRAP for their help and advice. Particularly, the journey would not have been possible without the constant support of other doctoral students, many of whom are now great friends. In addition, I have had a great pleasure make some good friends here in Toulouse and I am thankful for their help and support. I am also thankful to the members of the thesis committee for their valuable suggestions during the thesis and the members of the jury for their suggestions to improve the manuscript.

I would also like to thank all the members of the AstroSat team for their support and help. Also, the support from the IISER fraternity was very important and I thank all my friends.

Finally and most importantly, I would like to thank my family for their support, not just for the past three years but for all the time up to achieving this personal success.

Résumé

Les sursauts gamma (Gamma-Ray Bursts ou GRBs) sont les explosions les plus lumineuses de l'univers. On les observe sous la forme de bouffées de rayons X/γ (d'une durée de quelques millisecondes à quelques dizaines de secondes) suivies d'une émission rémanente (généralement à de plus grandes longueurs d'onde). Ils résultent soit de la fusion de deux objets compacts (une paire d'étoiles à neutrons ou une étoile à neutrons et un trou noir), soit de l'effondrement du noyau d'une étoile massive ($> 15M_{\odot}$). Les GRBs sont d'excellents candidats pour étudier la physique aux énergies et densités extrêmes et un outil astrophysique pour sonder l'histoire de l'univers car ils sont observés à tous les âges de celui-ci.

La mission spatiale Sino-Française *SVOM* (lancement prévu en juin 2022) a pour objectif la détection et l'étude des GRBs à l'aide d'instruments spatiaux et terrestres dédiés afin d'obtenir une couverture multi-longueurs d'onde. Le principal instrument à bord du satellite *SVOM* est ECLAIRs, un imageur à masque codé à grand champ de vue (~ 2 sr) fonctionnant dans la bande d'énergie de 4 à 150 keV. ECLAIRs détectera et localisera les GRBs (ainsi que d'autres sources transitoires à hautes énergies) en temps quasi réel grâce à son « trigger » embarqué.

Le bruit de fond d'ECLAIRs est élevé et variable en raison du grand champ de vue et de la stratégie de pointage de *SVOM* qui amène la Terre à transiter dans le champ de vue. Une nouvelle méthode (appelée « Particle Interaction Recycling Approach » ou PIRA en anglais), basée sur des simulations de Monte-Carlo (GEANT4), a été développée pour estimer précisément et rapidement le bruit de fond variable. Les simulations du bruit de fond sont complétées avec des sources X et des sursauts gamma afin de générer des scénarios d'observation complets.

Le bruit de fond variable d'ECLAIRs pose des problèmes pour la détection des GRBs et affecte la sensibilité de l'instrument. Nous avons évalué les performances du « trigger » embarqué, notamment l'impact du bruit de fond sur la détection des sources transitoires et sa sensibilité aux caractéristiques des GRBs (durée, profil temporel, forme spectrale, position dans le champ de vue).

ECLAIRs enverra au sol tous les photons détectés. De plus, la disponibilité d'une plus grande puissance de calcul et une meilleure connaissance du contexte (par exemple, les variations du bruit de fond, les sources dans le champ de vue, etc.) au sol, nous ont conduits à développer un « trigger » sol pour surmonter les difficultés rencontrées par le « trigger » embarqué. Ainsi, nous proposons un algorithme basé sur des transformées en ondelettes pour détecter les GRBs dans le cadre du « trigger » sol.

Les travaux de cette thèse, à savoir le développement de PIRA, l'évaluation des performances et le développement d'un nouvel algorithme de détection de sursauts, fournissent une base solide pour construire un « trigger » sol efficace, qui complétera le « trigger » embarqué et améliorera les performances globales de la mission *SVOM*.

Abstract

Gamma-Ray Bursts (GRBs) are the most luminous explosions in the universe. They are observed as bright flashes of gamma/X-rays (lasting a few milliseconds to a few tens of seconds) followed by an "afterglow" emission (usually at longer wavelengths). They are produced either due to the merger of two compact objects (a pair of neutron stars or a neutron star and a black hole) or due to the core collapse of a massive star ($> 15M_{\odot}$). GRBs are excellent candidates to study physics at extreme energies and densities. They also constitute important astrophysical tools to probe the history of the universe as they are observed at all epochs.

The upcoming (June 2022) Sino-French mission *SVOM* (Space-based multi-band astronomical Variable Objects Monitor) aims to detect and study GRBs using dedicated space and ground based instruments to obtain multi-wavelength coverage. The primary instrument onboard *SVOM* spacecraft is ECLAIRs, a wide-field ($\sim$ 2 sr) coded-mask imager sensitive in the 4 - 150 keV energy range. ECLAIRs will detect and localise GRBs (and other high energy transients) in near real time using an onboard trigger.

ECLAIRs will encounter a high and variable background due to the wide field-of-view (FoV) and the pointing strategy of *SVOM* which makes the Earth transit through the FoV. A new method (called Particle Interaction Recycling Approach or PIRA), based on Monte-Carlo simulations (GEANT4), was developed to estimate the variable background accurately and rapidly. The simulations of the background are complemented with simulations of X-ray sources and gamma-ray bursts to generate complete observation scenarios.

The variable background of ECLAIRs poses challenges to detect GRBs and affects the sensitivity of the instrument. We use the simulated data to evaluate the performance of the onboard trigger, in particular, the impact of the variable background and its sensitivity to the GRB characteristics (duration, temporal profile, spectral shape, position in the FoV).

ECLAIRs will send all detected photons to the ground. In addition, the availability of a larger computational power and the better knowledge of the context (e.g. background variations, sources in the FoV, etc.) on the ground motivates us to develop an "offline trigger" to overcome the challenges faced by the onboard trigger. An algorithm based on wavelet transforms is proposed to detect GRBs as part of the offline trigger.

The work in this thesis, i.e. the development of PIRA, instrument's performance evaluation and development of a trigger method, provides a sound basis to build an effective offline trigger that will complement the onboard trigger and improve the overall performance of the *SVOM* mission.

Contents

List of Figures **7**

List of Tables **11**

1 Introduction **13**
1.1 The X-ray and Gamma-ray astronomy . . . 13
1.2 Past, present and future gamma-ray burst missions . . . 15
1.3 Motivation behind the work . . . 20

2 The high-energy transient sky **23**
2.1 The Gamma-ray Bursts . . . 23
2.1.1 GRBs science: from discovery till today . . . 23
2.1.2 GRB phenomenology and physics of the explosions . . . 26
2.1.2.1 GRB Prompt emission . . . 26
2.1.2.2 GRB Afterglow emission . . . 32
2.1.2.3 GRB progenitors and central engine . . . 34
2.1.3 The astrophysics of gamma-ray bursts . . . 36
2.1.3.1 GRBs as multi-messenger objects . . . 36
2.1.3.2 GRBs as cosmic rulers . . . 37
2.1.3.3 GRBs as cosmic and galactic probes . . . 38
2.1.4 GRBs: future prospects . . . 38
2.2 X-ray / Gamma-ray transients other than GRBs . . . 39
2.2.1 Terrestrial Gamma-ray Flashes . . . 39
2.2.2 Solar flares . . . 39
2.2.3 Transient X-ray binaries . . . 40
2.2.4 Soft Gamma Repeaters . . . 42
2.2.5 Tidal Disruption Events . . . 43
2.2.6 Supernovae and Active Galactic Nuclei . . . 44

3 The SVOM mission and the soft gamma-ray imager ECLAIRs **45**
3.1 The SVOM mission . . . 45
3.1.1 SVOM space segment . . . 47

3.1.1.1 ECLAIRs . . . 47
3.1.1.2 Gamma Ray Monitor . . . 47
3.1.1.3 Microchannel X-ray Telescope . . . 48
3.1.1.4 Visible Telescope . . . 48
3.1.2 SVOM ground segment . . . 49
3.1.2.1 Ground-based Wide Angle Camera . . . 49
3.1.2.2 Ground Follow-up Telescopes . . . 50
3.1.3 SVOM: Mission profile and operations . . . 50
3.1.3.1 Satellite orbit and pointing . . . 50
3.1.3.2 Transient alert system . . . 52
3.1.3.3 Observation programs . . . 52
3.1.4 SVOM: Scientific objectives and prospects . . . 53
3.2 The gamma-ray imager ECLAIRs . . . 56
3.2.1 ECLAIRs instrument description . . . 56
3.2.2 ECLAIRs geometry and coded mask imaging . . . 59
3.2.3 ECLAIRs performance and science prospects . . . 62

4 Simulations of the background and raw data for ECLAIRs 65
4.1 The astrophysical background in ECLAIRs . . . 65
4.2 ECLAIRs background simulations . . . 67
4.2.1 The GEANT4 simulations . . . 67
4.2.2 Background estimation using PIRA . . . 67
4.2.2.1 Generation of database of primaries . . . 68
4.2.2.2 Selection algorithms . . . 72
4.2.2.3 Assignment of time and dynamical simulation . . . 78
4.2.3 ECLAIRs Background in the static and dynamic cases . . . 79
4.2.3.1 ECLAIRs Background in static case . . . 79
4.2.3.2 ECLAIRs Background in the dynamic case . . . 82
4.2.4 Performance of PIRA and statistical considerations . . . 83
4.3 Simulation of X-ray sources and GRBs . . . 85
4.3.1 Source photon list simulations . . . 86
4.3.2 GRB photon list simulations . . . 88
4.3.3 Generation of source/GRB events . . . 89
4.4 ECLAIRs background and data simulators . . . 90
4.5 PIRA: future perspectives . . . 92

5 Performance of ECLAIRs trigger 95
5.1 Onboard trigger and GRB detection with ECLAIRs . . . 96
5.1.1 Onboard trigger: Requirements . . . 96
5.1.2 Onboard trigger: Algorithm and implementation . . . 96

5.1.2.1 The image trigger . . . 97
5.1.2.2 The count-rate trigger . . . 98
5.2 Performance of the onboard count-rate trigger . . . 99
5.2.1 Detection threshold for count-rate trigger . . . 100
5.2.2 Detection sensitivity of count-rate trigger . . . 101
5.2.3 Impact of trigger parameters on count SNR . . . 104
5.2.4 Count-rate trigger sensitivity to GRB characteristics . . . 110
5.3 Offline trigger for GRB detection with ECLAIRs . . . 110
5.3.1 Haar wavelet detection algorithm . . . 112
5.3.1.1 Wavelet transform properties . . . 113
5.3.1.2 Detection algorithm . . . 115
5.3.1.3 Performance of Haar trigger and perspectives . . . 122

Conclusions [English] **125**

Conclusions [Française] **129**

Bibliography **133**

Appendices **149**

A Coordinate frame transformations used in the Albedo background computation **149**
A.1 The GEANT4 frame to the “Local” frame transformation . . . 149
A.1.1 Definition and computation of ψ_E . . . 151
A.2 The “Local” frame to the “MAP” frame transformation . . . 151
A.3 The “MAP” frame to the geographic frame transformation . . . 152
A.4 The geographic frame to the geomagnetic frame transformation . . . 153
A.5 The celestial frame to the MAP frame transformation . . . 154

B PIRA codes for ECLAIRs **155**

List of Figures

1.1 Lightcurve of a GRB detected by Vela satellites . . . 15

2.1 All-sky map of BATSE GRBs . . . 24
2.2 Brightness distribution of GRBs from BATSE sample . . . 25
2.3 An artist's impression of the standard scenario for GRBs . . . 26
2.4 Sample lightcurves from BATSE GRB database . . . 27
2.5 Definition of GRB T_{90} . . . 28
2.6 T_{90} distribution of GRBs from BATSE database . . . 28
2.7 Hardness ratio vs. duration (T_{90}) distribution . . . 29
2.8 Prompt emission spectrum of GRB990123 . . . 30
2.9 Cartoon jet model for GRB depicting various emission processes and their site . . . 31
2.10 Canonical model explaining GRB afterglow lightcurve . . . 32
2.11 Observed afterglow lightcurves for a sample of GRBs . . . 34
2.12 The GW170817/GRB170817 detection . . . 36
2.13 TGF detected by *Fermi*/GBM . . . 40
2.14 SGR detected by *Swift*/BAT . . . 42
2.15 Representative picture of TDE scenario . . . 43

3.1 SVOM instrument overview . . . 45
3.2 GRB observation sequence by *SVOM* instruments . . . 46
3.3 The Gamma Ray Monitor . . . 47
3.4 The Micro-channel X-ray Telescope . . . 48
3.5 The Visible Telescope . . . 49
3.6 The Ground Wide Angle Camera . . . 49
3.7 The Ground Follow-up Telescopes . . . 51
3.8 *SVOM* pointing strategy . . . 52
3.9 The VHF network . . . 53
3.10 Space missions/ground facilities in *SVOM* era . . . 55
3.11 ECLAIRs and exploded view of its components . . . 56
3.12 ECLAIRs detector plane . . . 58

3.13 ECLAIRs mask pattern and structural model 58
3.14 Coded mask imaging principle and geometry of ECLAIRs 60
3.15 ECLAIRs sensitivity to GRB peak energy 62

4.1 ECLAIRs mass model 68
4.2 Indexing convention for pixels and orientation of ECLAIRs detector plane inside GEANT4 69
4.3 Input spectra for CXB, Reflection and Albedo simulations 71
4.4 Input spectra for charged particle simulations 72
4.5 PIRA: directional selection geometry 72
4.6 PIRA: spatial mapping geometry 75
4.7 PIRA: dynamical simulation extension flow chart 79
4.8 PIRA and direct GEANT4 detected spectra comparison for static simulations 80
4.9 PIRA and direct GEANT4 input photon distribution comparison for Albedo static simulations 81
4.10 PIRA and direct GEANT4 detected spectra comparison for Albedo static simulations 81
4.11 PIRA and direct GEANT4 detected spectra comparison for charged particle static simulations 82
4.12 ECLAIRs dynamic background (with CXB, Reflection and Albedo) simulated using PIRA 83
4.13 ECLAIRs dynamic background and detector plane image 84
4.14 ECLAIRs dynamic background due to the charged particles at the start of a SAA passage simulated using PIRA 84
4.15 Source sample used to simulate events in ECLAIRs 86
4.16 Comparison of photon drawn spectrum with the source model for Crab 88
4.17 Method of assigning photon arrival time to the GRB photon list 89
4.18 Schematic diagram illustrating source/GRB event simulation 90
4.19 ECLAIRs data simulator toolkit work flow 92
4.20 A complete (background + source + GRB) simulation of ECLAIRs data 93

5.1 Energy band and detector zone definitions for onboard trigger 97
5.2 Image trigger algorithm 97
5.3 Count-rate trigger algorithm 98
5.4 Background phase definition 100
5.5 Synthetic GRB light curves 101
5.6 Real GRB light curves 102
5.7 Background SNR distribution 103
5.8 Minimum count-rate for 6.5σ detection as function of orbit 104

5.9 Limiting flux as function of orbit . . . 104
5.10 Count-SNR dependence on timescale and energy bands for synthetic GRBs . . . 105
5.11 Count-SNR and image-SNR variation within FoV . . . 106
5.12 Relation between count-SNR and corresponding image-SNR for synthetic GRBs . . . 107
5.13 Count SNR dependence on timescale and energy bands for real GRBs . 108
5.14 Relation between count-SNR and corresponding image-SNR for real GRBs . . . 109
5.15 Limiting flux vs GRB properties . . . 111
5.16 Wavelet decomposition example for a sample time series . . . 114
5.17 Wavelet variance variation against decomposition level . . . 115
5.18 Normalised wavelet coefficient distribution for background only data . . 116
5.19 Coefficient significance distribution for data with GRB . . . 117
5.20 Maximum coefficient significance at each level . . . 118
5.21 Haar wavelet signal SNR timeseries . . . 119
5.22 Maximum Haar-SNR variation with the timescale . . . 120
5.23 Haar wavelet trigger for increasing and decreasing background phase . . 121
5.24 Comparison of Haar-SNR and count-SNR . . . 122

A.1 Definition of the Earth roll angle and frame transformations . . . 150

B.1 Organisation and sequence of execution of PIRA script and modules . . 156

List of Tables

1.1 Summary of notable past and current GRB missions 17
1.2 Summary of important upcoming GRB missions 18

3.1 Important characteristics of ECLAIRs 57

4.1 PIRA source file example 70
4.2 PIRA event file example 70
4.3 PIRA and direct GEANT4 performance comparison 85

5.1 Sample GRB properties 101

B.1 Description of PIRA scripts and modules 155

Chapter 1

Introduction

1.1 The X-ray and Gamma-ray astronomy

Astronomical observations have been carried out for centuries and across different cultures throughout human history. But for most periods in history, these observations were possible only in the visible part of the electromagnetic spectrum. The technological advances throughout the 20^{th} century enabled us to observe the sky at different wavelengths. After the start of the "space-age", it was possible to put telescopes into space and observe high-energy radiation (X-rays/gamma-rays) which would otherwise be impossible to detect from the surface of the Earth due to its absorption by the atmosphere. In 1962, during one such attempt to observe the higher energy radiation from space, the first extra-solar X-ray source, Sco X-1, was discovered by the detectors onboard a sounding rocket experiment (Giacconi et al., 1962). This discovery inspired the use of sustained satellite missions to observe the high-energy sky. The first dedicated X-ray satellite, *Uhuru*, catalogued around 340 X-ray sources in the energy band 2 - 6 keV (Forman et al., 1978). Detection of high energy gamma-rays ($\sim$ MeV) from the sky came a little later in 1972 [1] when 50 MeV gamma-rays were detected from the disk of the Milky Way by the *OSO-3* mission (Kraushaar et al., 1972). With deep and detailed all-sky surveys carried out by successive missions (e.g *ROSAT*, *CGRO*/EGRET, *Chandra*, *XMM-Newton*, *Fermi*/LAT and *SRG*/eROSITA) the number of known X-ray/gamma-ray sources are now in their millions. Along with space-based observations, the emergence of Cherenkov-shower telescopes in the last two decades has also opened up the possibility of observing very high energy ($\sim$ TeV) gamma-rays from the ground. Together with the observations at lower energies (e.g in radio, infrared, visible, UV), it is now possible to carry out multi-wavelength studies (covering almost 15 orders of magnitude in wavelength) of many astrophysical objects (such as supernova remnants, Active Galactic Nuclei or AGNs, pulsars etc.) giving a

1. The *Vela* constellation indeed detected MeV emission from gamma-ray bursts (GRBs) in 1967 but the discovery was announced in 1973 (see section 1.2)

better understanding of the astrophysics of these objects.

X/gamma-ray sources show diversity in terms of the size of the objects, energetics and distance at which they are observed in the universe. Sources like pulsars and X-ray binaries have an emission region spanning few solar radii and are observed in the Milky Way or its local environment (e.g in globular clusters); while AGNs span galactic scales ($\sim$ kpc) and they are observed up to very large (cosmological) distances [2]. Most high energy sources exhibit extreme physical phenomena, such as very high magnetic fields (e.g. $\sim 10^{15}$ G in the case of some pulsars), extreme gravitational fields, accretion of matter (e.g. X-ray binaries), emission via relativistic jets (e.g. AGNs) making them an ideal laboratory to study physical phenomena which can not be reproduced on Earth. Furthermore, as sources are observed at various distances relative to the Earth and the current epoch; the local sources play an important role in understanding the evolution of galaxies and local environments while the more distant sources are useful to map the history of the universe (see section 2.1.3.3 for one such application using GRBs).

Over the past half a century, X/gamma-ray astronomy has played an important role to shape our current understanding of the universe; answering many questions but also opening new ones. The transient phenomenon at high energies is one such subfield in which a lot of serendipitous discoveries have opened up more questions than they have answered (e.g. see section 2.2 and Chapter 2). Most transient phenomena have short durations ($\sim$ seconds to hours) limiting the observable photon flux, and they are unpredictable (in time and position) requiring constant monitoring of the sky to detect them. The limited photon flux and the necessity of constant monitoring of the sky imply the need for sensitive detectors with a wide field of view (FoV) to detect transients. Furthermore, many transients exhibit broadband emission (from soft X-rays to high energy gamma-rays) and require detectors that are sensitive in a broad (at least an order of magnitude) energy band.

Gamma-ray Bursts (GRBs) are one such transient that requires constant monitoring of the sky and sensitivity in a broad energy range for their detection. GRBs are non-repeating transients emitting bursts of γ-rays. They are the most luminous explosions in the universe lasting from a few milliseconds to a few thousand seconds. GRBs occur isotropically in the sky with most originating at large cosmological distances ($z > 1$). The transient nature and unpredictability of GRBs make their detection a challenging task. The burst of gamma-rays, called the prompt emission, is usually followed by a multi-wavelength afterglow emission. This emission is observed from X-rays at early times to radio wavelengths at late times (few hours to days or even months). A detailed overview of GRB properties is given in Chapter 2.

2. To give perspective, the most distant AGN observed till now, ULAS J1342+0928, has redshift $z \simeq 7.2$

Observations and understanding of transients like GRBs have been made possible due to dedicated and carefully designed high-energy space missions. The ECLAIRs instrument onboard the upcoming space mission *SVOM* (launch and operation expected in June 2022), see Chapter 3 and section 3.2) is a wide-field instrument dedicated to the study of GRBs and other high-energy transients. In the next section, I briefly discuss some of the important past, current and future high-energy space missions, particularly in the context of GRB detection.

1.2 Past, present and future gamma-ray burst missions

GRBs were first discovered by the United States' military satellite constellation *Vela*. The first detection of a GRB was on July 2, 1967 (Figure 1.1). In addition, the satellites *Vela 5a, 5b, 6a* and *6b* detected 16 GRBs (Klebesadel et al., 1973) during the period from 1967 to 1973. Following the initial discovery by the *Vela* satellites, many

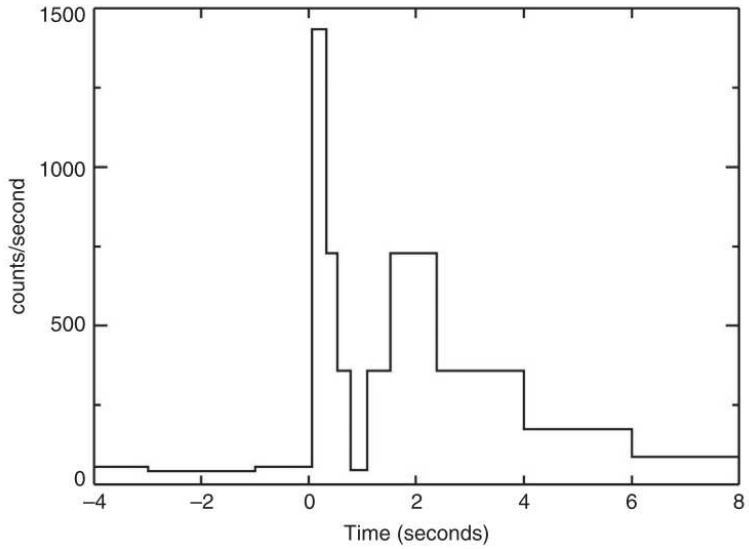


Figure 1.1 – The lightcurve of the first-ever GRB detected by the *Vela 4a, 4b* satellite on July 2, 1967. Credit: NASA/GSFC GRBCAT

other space-borne instruments such as *Venera*/Konus (Mazet et al., 1974) and Apollo 16 (Metzger et al., 1974) also detected GRBs. However, due to the weak localisation capabilities of most of the gamma-ray instruments, it was difficult to search for GRB counterparts at other wavelengths which could have given greater insights into their origins.

Dedicated efforts to understand GRBs started in early 1990s with the launch of the *Compton Gamma Ray Observatory* (*CGRO*, Gehrels et al., 1993). The Burst And Transient Source Experiment (BATSE) onboard *CGRO* was dedicated to the

detection of GRBs. BATSE had eight modules containing a Large Area Detector (LAD) and a Spectroscopy Detector (SD). BATSE was able to detect count-rate variations on timescales down to milliseconds and had a localisation accuracy of a few degrees (Briggs et al., 1999, see also table 1.1). BATSE played an important role in developing a basic understanding of the GRB prompt emission properties.

Multi-wavelength studies of the GRB phenomena were possible for the first time after the launch of *BeppoSAX* (Piro et al., 1995; Boella et al., 1997a) and later by the *High Energy Transient Explorer* (*HETE-2*, Ricker, 2003). *BeppoSAX* carried a Wide Field Instrument (WFI, see table 1.1) which included a set of two Wide Field Cameras (Jager et al., 1997, WFCs,) and a Gamma-Ray Burst Monitor (Feroci et al., 1997). It also carried Narrow Field Instruments (NFIs, see table 1.1) which included a set of four X-ray mirror telescopes (Parmar et al., 1997; Boella et al., 1997b) and two collimated instruments (Manzo et al., 1997; Frontera et al., 1997). *BeppoSAX* was able to follow-up GRBs after ground-based localisation. *HETE-2* carried the French Gamma-ray Telescope (FREGATE, Atteia et al., 2002) and a Wide Field X-ray Monitor (WXM, Shirasaki et al., 2003) and a Soft X-ray Camera (SXC, Villasenor, 2003). *HETE-2* was capable of detecting and localising GRBs onboard. The follow-up capabilities of both missions laid the foundations of our understanding of the GRB afterglows (see section 2.1.1).

The *Neil Gehrels Swift Observatory* (Gehrels, 2004) revolutionised the GRB field with its fast localisation, autonomous repointing and multi-wavelength follow-up capabilities. *Swift* has three instruments. The primary instrument is the Burst Alert Telescope (BAT, Barthelmy et al., 2005) – a hard X-ray wide-field coded-mask imager with a large collecting area (5200 cm^2). BAT is capable of rapid GRB detection (order of seconds) and accurate localisation (typically 3 arcmins). The other two instruments are the narrow field X-ray Telescope (XRT, Burrows et al., 2005) and the UV-Optical Telescope (UVOT, Roming et al., 2005) and they follow up (after the satellite repointing) BAT detected GRBs in the soft X-rays and UV/optical bands respectively (see table 1.1 for details). *Swift* has been crucial for significant progress in the GRB afterglow science during its 15 years of operations. Due to *Swift*'s accurate localisation capability (XRT localization < 2 arcsec), it is possible to follow-up GRBs with large telescopes equipped with spectrographs enabling the measurement of the redshift for many *Swift* detected GRBs.

One more important GRB mission which is still operational is the *Fermi Gamma-ray Space Telescope*. It carries two high-energy instruments; the Large Area Telescope (LAT, Atwood et al., 2009) and the all-sky Gamma-ray Burst Monitor (GBM, Meegan et al., 2009). The two instruments together cover around seven orders of magnitude in energy with good spectroscopic capabilities (see table 1.1). *Fermi* has been important for the very high energy broadband studies of GRB prompt emission

Table 1.1 – Summary of notable past and current GRB missions

Mission	Instrument(s)	Energy range	FoV (Loc. accuracy)	GRBs/yr‡	Selected important discovery	Years of operation
CGRO	BATSE - LAD BATSE - SD	20 keV - 1.9 MeV 10 keV - 100 MeV	4π sr (0.2° - 18°)	300	Isotropic distribution	1991 - 2000
BeppoSAX	2 WFCs 3 NFIs	2 - 30 keV 0.1 - 300 keV†	$40° \times 40°$ ($5'$) $\sim 1°$†	180	First afterglow detection	1996 - 2002
HETE-2	FREGATE WXM SXC	6 - 400 keV 2 - 25 keV 0.5 - 14 keV	$70° \times 70°$ (-) 1.6 sr ($< 19'$) 1.8 sr ($\sim 40''$)	40	SNe - GRB connection	2000 - 2006
Swift	BAT XRT UVOT	15 - 350 keV 0.3 - 10 keV 170 - 600 nm	1.4 sr ($1'$ - $4'$) $23.6' \times 23.6'$ ($18''$) $17' \times 17'$ ($\sim 0.5''$)	90	Extensive afterglow study, permitted hundreds of redshift measurements	2004 - present
Fermi	GBM LAT	8 keV - 40 MeV 20 MeV - 300 GeV	4π sr ($< 15°$) 2.4 sr ($< 3.5°$)	230	GRB - Gravitational waves connection	2008 - present
INTEGRAL	SPI IBIS	18 keV–8 MeV 15 keV–10 MeV	16° ($< 1.3°$) $9° \times 9°$ ($< 1'$)	9	GRB - Gravitational waves connection	2002 - present
GECAM	2 satellites with 25 GRDs each	6 keV – 5 MeV	all sky ($\sim 1°$)⋆	–	–	2020 - present

Notes:
Current operational missions are indicated in bold.
† Combined energy range and average FoV value for all NFIs (Narrow Field Instruments).
‡ It is an average rate estimated by dividing total number of detections over the mission lifetime.
⋆ Predicted localisation accuracy.

Table 1.2 – Summary of important upcoming GRB missions

Mission	Instrument(s)	Energy range	FoV (Loc. accuracy†)	GRBs/yr†	Launch date
SVOM	ECLAIRs	4 keV - 150 keV	$89° \times 89°$ ($\sim 13'$)	70	June 2022
	GRM	15 keV - 5 MeV	~ 2 sr ($\sim 5°$)	90	
	MXT	0.2 keV - 10 keV	$64' \times 64'$ ($\sim 13''$)	NA	
	VT	400 - 1000 nm	$26' \times 26'$ ($\sim 0.5''$)	NA	
Einstein Probe	WXT	0.5 - 4 keV	$36° \times 36°$ ($\sim 4'$)	–	End of 2022
	FXT	0.5-10 keV	$38'$ diam. ($< 20''$)		
eXTP	9 SFA	0.5-10 keV	$\geq 12''$ ($\sim 1'$)	NA	2027
	4 PFA	2-8 keV	$\sim 8''$ ($\leq 30''$)	NA	
	LAD	2-30 keV	$\sim 1°$ (-)	NA	
	WFM	2-50 keV	~ 4.1 sr ($< 1'$)	100	
THESEUS	XGIS	2 keV - 20 MeV	2 sr ($\sim 10'$)	387–870	Start of 2030 decade
	SXI	0.2 - 5 keV	0.5 sr ($< 1 - 2'$)		
	IRT	0.7 - 1.8 μm	$15' \times 15'$		
CubeSats	BurstCube	50 keV 1 MeV	all sky	–	Late 2021
	HERMES	3 keV - 2 MeV	all sky	–	–
	Glowburg	30 keV - 2 MeV	8 sr	70	Before 2025
	3UTransat⋆	20 - 200 keV	all sky	70	Before 2025

Notes:
NA - Not Applicable.
– indicate information is not available.
† All values indicate average predicted rate.
⋆ Under study at IRAP.

giving significant insights to constrain the prompt emission models (Kumar & Barniol Duran, 2009; Ackermann et al., 2010; Ghisellini et al., 2010; Zhang et al., 2011). The GRB170817A associated with the binary neutron star (BNS) merger GW170817 was first detected and localised by the *Fermi*/GBM onboard trigger (see section 2.1.3.1).

Another mission was recently launched (December 2020) specifically to detect GRBs associated with gravitational wave events. It is called the Gravitational wave high-energy Electromagnetic Counterpart All-sky Monitor (*GECAM*, Li et al., 2020b). *GECAM* consists of a set of two satellites each carrying 25 scintillator-based Gamma-Ray Detectors (GRDs). The two satellites operate in a low-earth orbit and in opposite orbital phases giving an all-sky coverage (see table 1.1).

Apart from the dedicated GRB missions, other high-energy missions such as the *INTEGRAL* observatory (Winkler et al., 2003), the *AstroSat* (Singh et al., 2014) have also played an important role in detecting and understanding GRBs. The SPI imaging-spectrometer (Vedrenne et al., 2003) and IBIS imager (Ubertini et al., 2003) onboard *INTEGRAL* and the Cadmium Zinc Telluride Imager (CZTI, Bhalerao et al., 2017) onboard *AstroSat* have been important to carry out polarization studies of some GRBs (e.g. see Kalemci et al., 2006; McGlynn et al., 2007, 2009; Götz et al., 2009, 2014a; Chattopadhyay et al., 2019) which have been crucial to give some clues about the prompt emission mechanism.

Although the current missions such as *Fermi* and *Swift* are performing up to the

mark, it is important to commission next-generation missions to better understand GRBs and answer some of the open questions. Many dedicated GRB missions are under development or have been proposed. One such mission which will be important for multi-wavelength studies of GRB phenomena is the Sino-French *SVOM* mission (Wei et al., 2016, see also Chapter 3). *SVOM* will have four instruments onboard a satellite and three dedicated follow-up instruments on the ground. Onboard it will carry two high-energy wide-field instruments – the ECLAIRs and the Gamma-Ray Monitor (GRM) and two narrow field instruments – the Micro-channel X-ray Telescope and the Visible Telescope (see table 1.2). *SVOM* will be very important for the broadband studies of the prompt emission with an energy coverage down to 4 keV. Additionally, *SVOM* is designed such that the redshift can be determined for a large fraction ($\sim 2/3$) of the GRBs detected by its space instruments (ECLAIRs and GRM). This makes *SVOM* an important mission in solving some of the open questions such as if GRBs can be used as cosmic rulers or as probes of the history of the universe (see also section 2.1.3).

In addition to *SVOM*, another mission called the *Einstein Probe* (EP, Weimin et al., 2016) will be commissioned around the same time (see table 1.2). It will carry two soft X-ray instruments, a Wide-field X-ray Telescope (WXT) and a Focusing X-ray Telescope (FXT). Due to its low energy coverage, EP will be important in detecting soft and highly redshifted GRBs.

Both *SVOM* and EP are small-sized satellite missions (total mass $\lesssim$1000 kg) and the sensitivity of the instruments onboard is constrained by the limited mass allocation (e.g. the collecting area of ECLAIRs is ~ 5 times smaller than that of *Swift*/BAT). To have more sensitivity to high-energy transients (including GRBs) and explore a discovery space that spans over the entire cosmic history of the universe, a medium-size mission (total mass $\gtrsim$2000 kg) called *THESEUS* (Amati et al., 2018) is proposed (see table 1.2). It is foreseen to have three instruments onboard, the X/Gamma-ray Imaging Spectrometer (XGIS), the Soft X-ray Imager (SXI) and the Infra-Red Telescope (IRT). It will play an important role in detecting GRBs originating from the very high redshift universe[3] ($z \sim 10$). The mission is under the assessment phase study of ESA and if selected it will be operational in the early 2030s. Other than *THESEUS*, another planned mission named the enhanced X-ray Timing Polarimeter (*eXTP*, Zhang et al., 2019) will have GRB detection as its secondary objective. It will carry a Spectroscopic Focusing Array (SFA), a Polarimetry Focusing Array (PFA), a Large Area Detector and a Wide Field Monitor (WFM). The WFM is expected to detect GRBs in real-time using an onboard trigger (similar to *SVOM*) and transmit the detection alert to the ground within 30 s after the trigger. The *eXTP* is planned to be launched

3. These GRBs might be associated with the first population of stars (population III) and could be important to probe the early epochs of the universe.

in 2027.

Apart from these missions, numerous nano-satellite mission (or CubeSats) such as *BurstCube* (Smith et al., 2019), *HERMES* (Fuschino et al., 2019), *LECX* (Braga et al., 2020), *Glowbug* (Grove et al., 2020), 3UTransat (under a study at IRAP) are also foreseen in the coming decade (see table 1.2). These missions are primarily designed to detect and localise short GRBs associated with Gravitational Wave events. They will play a crucial role in the domain of multi-messenger astrophysics involving GRBs.

1.3 Motivation behind the work

The success of the *SVOM* mission lies in the coordination between the space and ground-based instruments to study the complete picture of the GRB phenomenon. The first important step to achieve is the detection and localisation of GRBs. This task will be performed in real-time by the wide-field high-energy coded-mask imager ECLAIRs using an onboard trigger. Furthermore, ECLAIRs is designed to operate in "event-mode" and will send all the information about the detected photons to the ground. This enables the detection of transients on the ground using an "offline" trigger. Because both triggers are essential for the successful operation of ECLAIRs (and consequently that of *SVOM*), in a pre-launch phase it is important to assess the sensitivity of the onboard trigger and develop algorithms for the offline trigger keeping in mind the challenges (e.g. variable background) faced by ECLAIRs.

High-energy space instruments encounter backgrounds of astrophysical origin. In particular, it is known from previous missions (e.g. *INTEGRAL*/IBIS, Türler et al. (2010); *Swift*/BAT, Ajello et al. (2008)) that wide field-of-view (FoV) X/gamma-ray instruments experience a high level of background. ECLAIRs too will suffer from this due to its large FoV ($\sim$2 sr). The pointing strategy of *SVOM* (see Chapter 3, Cordier et al. (2008)) is chosen such that GRBs can be followed up from the ground as soon as they are detected by ECLAIRs. This requires ECLAIRs to always point towards the night-sky, i.e. in the anti-solar direction. Therefore, for all orbits, the Earth will pass through the FoV of ECLAIRs modulating the observed background.

ECLAIRs will be sensitive to a large number of bright and variable X-ray sources due to its low-energy threshold (4 keV). The detected counts from these sources will be modulated as well due to the passage of the Earth through the FoV. Therefore accurate knowledge of the source position, as well as their variability, is important to take into account their impact on the triggers. In particular, this has prompted the construction of an onboard catalogue of sources to assist the onboard trigger in the transient detection process (Dagoneau et al., 2020).

To estimate the background for any instrument, simulations can be carried out using a Monte-Carlo toolkit such a GEANT4. However, GEANT4 simulations are

time-consuming to determine the variable background for different observing cases. The development of an alternative method that can overcome the computational burden of the direct GEANT4 simulations is the main motivation behind this thesis work. The method developed (named Particle Interaction Recycling Approach or PIRA) extends the GEANT4 simulations to estimate the background (including its variation) in a fast and accurate way (see Chapter 4). In addition, we also keep in mind the goal to develop a general method so that it can be extended to any other high-energy instrument.

To assess the impact of the variable background on the performance of the instrument, it is important to simulate data representative of realistic observing cases. This motivated us to extend PIRA with simulations of X-ray sources (and GRBs if needed) to generate synthetic data as close as possible to the real data. These simulations also include complex cases such as slews of the satellite (changes in pointing direction). As part of this work, we use the simulated data to evaluate the sensitivity of the onboard count-rate trigger and the impact of the background variability on the trigger (see Chapter 5). Furthermore, the simulated data enable us to test algorithms that can be used as part of the offline trigger. One such algorithm is proposed as part of this thesis. The simulations of real observing cases have also found important applications for other technical and scientific performance tests (e.g. Arcier et al., 2020d).

Chapter 2

The high-energy transient sky

The transient sources in X-rays and gamma-rays have different types of progenitors. The observed energy, timescale, distance as well as size vary considerably with the progenitor. In this chapter, I will give an overview of high-energy transients (particularly in the context of *SVOM*/ECLAIRs). Because GRB study is the main objective of *SVOM*, I will give a detailed overview of the phenomenon while only briefly discussing other types of transients.

2.1 The Gamma-ray Bursts

I will begin with a discussion about gamma-ray bursts with the historical progress of the field and then outline some key aspects such as their observational properties, the physics of these explosions and their role in astrophysical perspective.

2.1.1 GRBs science: from discovery till today

After the initial discovery by the *Vela* satellites, no dedicated mission was launched for approximately 20 years and no real progress was made in terms of solving the mystery of the GRB phenomena. Only after the launch of *CGRO*/BATSE did the GRB science truly progress. One of the important conclusions drawn from the BATSE detected bursts was that the GRBs are isotropically distributed in the sky (Figure 2.1). Also, the lack of bursts at the faint end in the brightness distribution ($log\ N(>F)$ vs $log\ F$, see figure 2.2) indicated that GRBs are inhomogeneously distributed in space. Taken together with the observed isotropy, this observation hinted at a cosmological origin (Meegan et al., 1992). This result put rest to many theories proposed about the origin of GRBs (over hundred theories were proposed by early 1990s, see Nemiroff, 1994). BATSE data also revealed some important temporal properties of the GRBs by establishing two classes of GRBs (short and long GRBs) based on the duration (Kouveliotou et al., 1993). It also showed that GRBs have non-

thermal spectra which can be characterised empirically by two smoothly joined broken power-laws, called the "Band function", after the function's introduction by Band et al. (1993) (see also Eqn. 2.1).

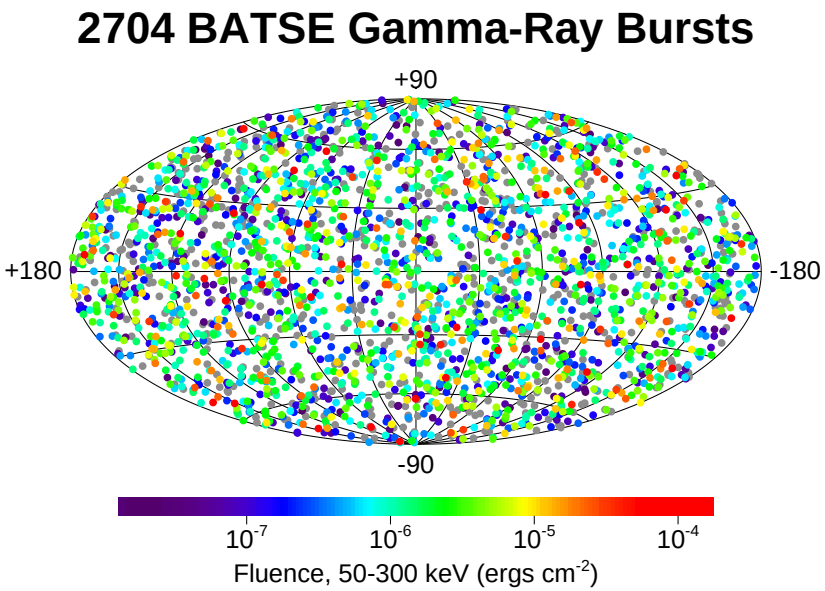


Figure 2.1 – The all sky distribution of 2704 BATSE-detected GRBs showing the isotropic nature of the bursts. Credit: NASA/NSSTC

BeppoSAX and *HETE-2* carried the GRB science into the multi-wavelength domain. On February 28, 1997, the *BeppoSAX*/WFCs detected and localised GRB970228. This was followed up by the *BeppoSAX*/NFIs (after a delay of ∼8 hours) which detected a fading X-ray counterpart in the 2 - 10 keV range confirming the first detection of a GRB "afterglow" (Costa et al., 1997). Simultaneously, an optical counterpart was also detected by ground-based telescopes (Van Paradijs et al., 1997). A few months later, optical afterglow observations of another event, GRB970508, showed absorption lines produced by the host-galaxy gas which allowed measurement of the spectroscopic redshift, $z = 0.835$ (Metzger et al., 1997) [1], and confirmed the cosmological origin of GRBs. A radio afterglow was also observed for this GRB (Frail et al., 1997).

The fast (∼ minutes times scale) onboard localisation capabilities of *HETE-2* enabled the first counterpart detections for long GRBs. This led to the confirmation of the connection between supernova (SNe) and GRB after the detection and localisation of GRB030329 and an associated supernova (Vanderspek et al., 2004). *HETE-2* also detected softer X-ray rich GRBs (more flux in X-rays than in gamma-rays) and X-ray flashes (Barraud et al., 2003; Sakamoto et al., 2005) extending the range of the GRB population in terms of their prompt emission spectral properties.

Swift's ability to slew rapidly (within minutes) has allowed quick follow-up of short GRBs which was not possible with previous missions. The afterglow, redshift

1. Estimates using the high energy flux and redshift indicate that the burst had an isotropic energy release, $E_{iso} \sim 7 \times 10^{51}$ erg during the prompt emission phase.

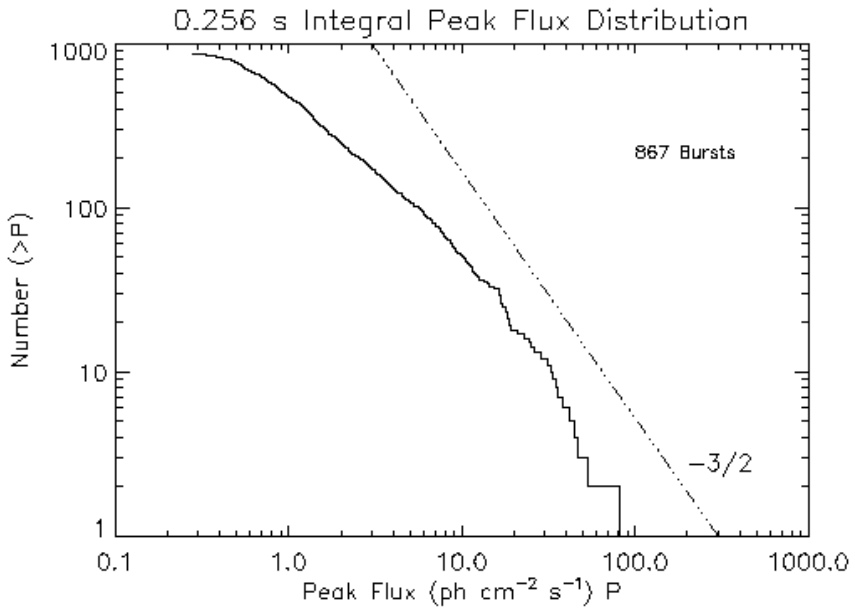


Figure 2.2 – The brightness distribution of 867 BATSE GRBs. If GRBs were uniformly distributed in an euclidean space, the number of GRB (N) above an observed flux F would follow the relation $N(>F) \propto F^{-3/2}$ giving rise to a distribution which follows the dotted line in this figure. However, the detected distribution show a lack of faint GRBs at the faint end hinting an inhomogeneity. Considering the fact that GRBs are observed isotropic on the sky (Figure 2.1), this distribution hints that GRBs are located at larger distances than local galaxies. Credit: NASA

and location properties (with regards to their host galaxy environment) of short GRBs detected by *Swift* has shown that short GRBs indeed have different progenitors than long GRBs (Berger et al., 2005; Nakar, 2007). The binary neutron star merger (BNS) theories proposed in the early 1990s (Eichler et al., 1989; Narayan et al., 1992) were the most consistent explanations of the *Swift* short GRB observations. This hypothesis was confirmed on August 17, 2017 with a coincident detection of GRB170817A (by *Fermi*/GBM and *INTEGRAL*/SPI-ACS) and gravitational wave signal (GW170817) from a BNS merger (see section 2.1.3.1). Regarding the afterglows of long GRBs, *Swift* studies have led to the most comprehensive understanding of the afterglow physics (Zhang et al., 2006). For example, *Swift* studies revealed the presence of features (e.g. X-ray flares, plateau) in the X-ray afterglow lightcurve of most GRBs (see section 2.1.2.2). The accurate localisation capability of *Swift* has enabled large optical telescopes to measure the redshifts of hundreds of GRBs ($\sim$ 450 GRBs till now) allowing to explore them in a cosmological context (see section 2.1.3.2 and 2.1.3.3)

Over the last 50 years, we have detected more than 5000 GRBs which have improved our understanding of these phenomena to a great extent. The space-based instruments have played a crucial role in detecting and localising the GRBs and measuring the prompt emission properties. The ground-based telescopes have been important to identify and follow-up GRB afterglows and their evolution at lower wave-

lengths (optical, infrared and radio). In particular, large optical telescopes have been key players in identifying GRB host galaxies and measuring their distance which is crucial for the GRB analysis. In the following section, I will give more details about our understanding of the physics of GRB phenomena.

2.1.2 GRB phenomenology and physics of the explosions

Currently, the well-accepted scenario (Figure 2.3) for the GRB emission (both prompt and afterglow) is explained by a fireball model (Piran, 1999). According to

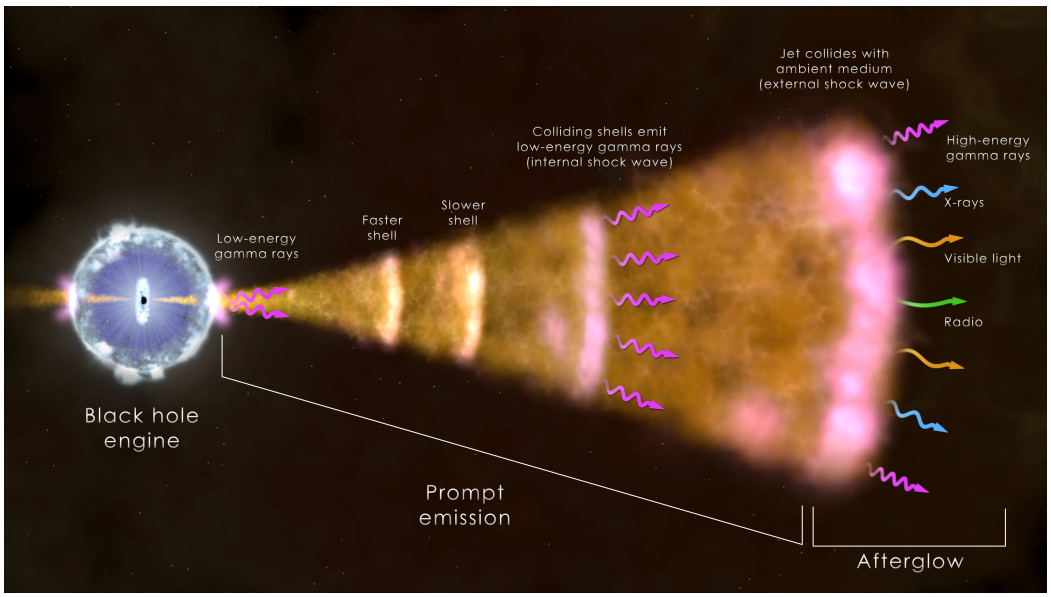


Figure 2.3 – A standard scenario for GRB emission. The compact central engine emits an ultra-relativistic jet which produces the prompt emission due to the internal dissipation and the afterglow emission due to the jet's interaction with the surrounding medium. Credits: NASA/GSFC.

this model, a transient ultra-relativistic jet is emitted by a compact central engine (a black hole or highly magnetized neutron star) which produces a GRB as it expands in the surrounding medium. The prompt emission is due to internal dissipation processes within the jet itself, while the afterglow emission is due to the interaction of the jet with the circumburst and interstellar mediums.

2.1.2.1 GRB Prompt emission

Lightcurves: The prompt emission phase is empirically defined as the time duration in which the emission in gamma-rays / hard X-rays is detected (typically in the tens of keV to few MeV range). The prompt phase of GRBs has very high temporal variability in terms of the observed photon flux. The prompt phase lightcurves are irregular with single or multiple peaks. The peaks can have a smooth profile or high variability component on top of a slowly varying profile (Figure 2.4). The duration

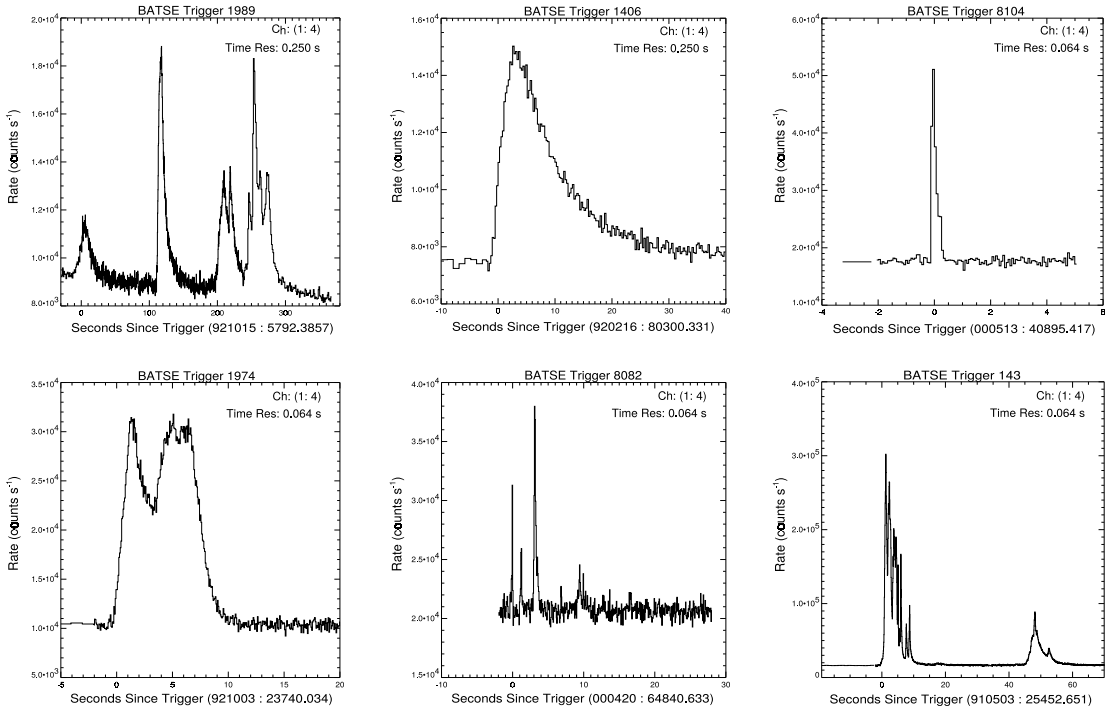


Figure 2.4 – Sample lightcurve from the BATSE database showing the irregular nature of the GRB lightcurves in the prompt emission phase. Credits: NASA/NSSTC

of the prompt emission typically varies from milliseconds to thousands of seconds. Conventionally, this duration is measured by the parameter called "T_{90}", defined as the difference between the times when the detected photon fluence of a GRB (photons collected per unit area in the detector bandpass) is between 5% and 95% of the total photon fluence (see figure 2.5). Initial observations by BATSE (Kouveliotou et al., 1993) and PHEBUS (Dezalay et al., 1992) showed that the duration distribution of the prompt emission has two distinct peaks with a separation at about 2 seconds. This indicated that there are two distinct types of GRBs, short-duration GRBs lasting < 2 seconds and long-duration GRBs lasting > 2 seconds. However, the analysis of GRBs detected by *Swift* and *Fermi* has shown that the separation between short and hard GRBs is not exactly at 2 seconds. Also, the duration distribution depends on the energy range in which the GRB duration is estimated (e.g see, Qin et al., 2012). Although the short/long divide does not have an unambiguous distinction purely based on the duration of the prompt emission, multi-wavelength observations by *Swift* have shown that the two classes do have a distinct physical origin with different kinds of progenitors. As mentioned before, the short GRBs most likely originate from the merger of binary neutron stars (or a neutron star and a black hole) while the long GRBs originate from the core collapse of a very massive star (> 15 $M_\odot$). Furthermore, the hardness ratio vs duration distribution (Kouveliotou et al., 1996; Dezalay et al., 1996) of GRBs have shown that short GRBs on average have a harder spectrum compared to long GRBs (e.g. see figure 2.7).

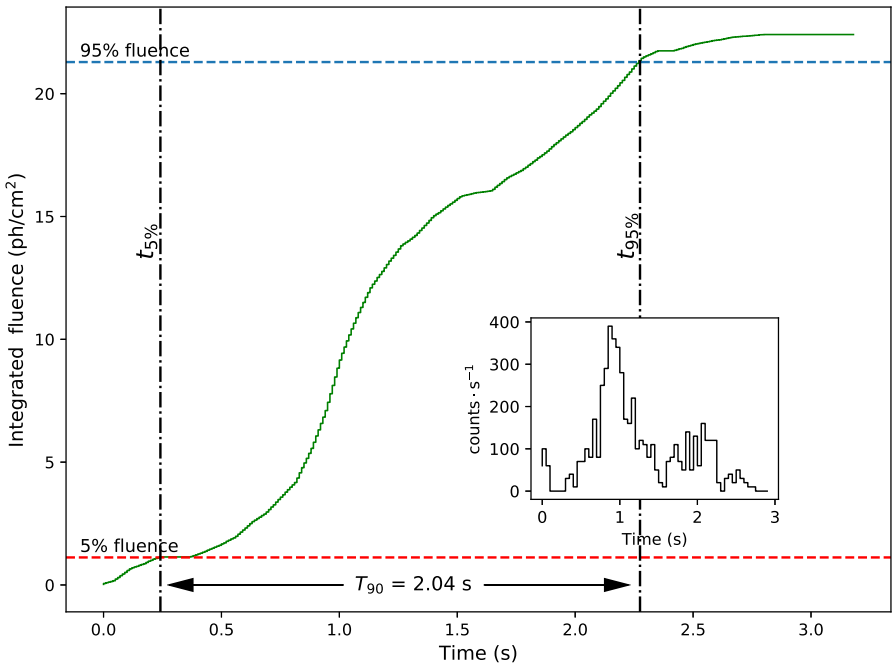


Figure 2.5 – An illustration showing the definition of the GRB duration parameter T_{90}: Here, the integrated fluence (ph/cm^2) of a GRB is plotted against the time. The T_{90} is defined as the duration between the epochs (black vertical lines) at which the fluence is between 5% (red line) and 95% (blue line) of the total fluence. The T_{90} is 2.04 seconds in this case. For reference, the counts lightcurve of the GRB is shown in the inset plot. Note: The observed fluence is also a function of the detector energy bandpass, hence the T_{90} is sensitive to the selected energy range.

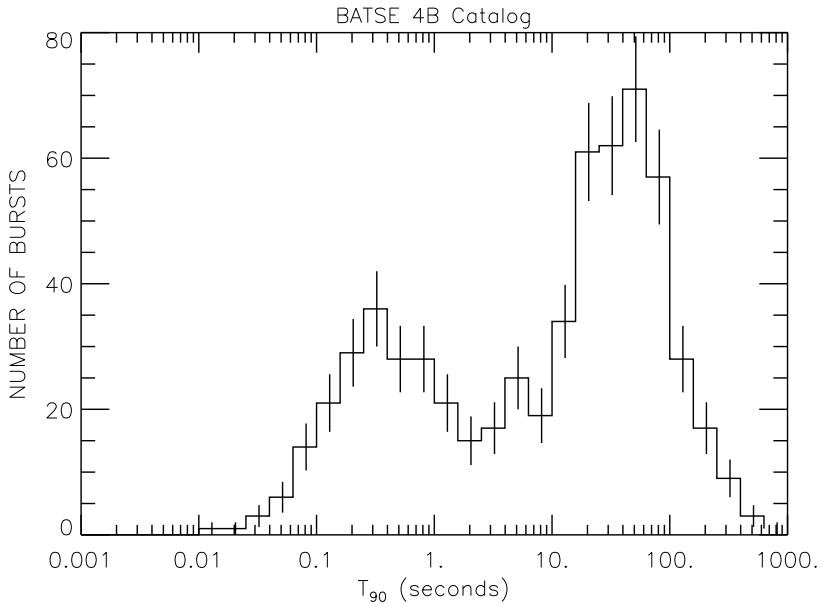


Figure 2.6 – The duration (T_{90}) distribution of BATSE GRBs. The distribution shows two clear peaks, one at ~ 0.3 seconds and another at ~ 30 seconds indicating two distinct classes of GRBs. Credits: NASA/NSSTC

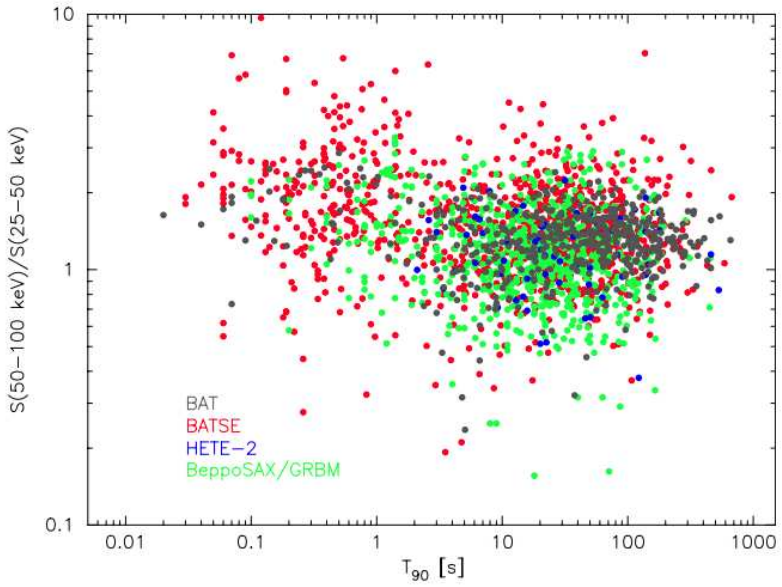


Figure 2.7 – The hardness ratio vs. duration (T_{90}) distribution of GRBs showing the short/ long divide. Figure from Sakamoto et al. (2011)

Spectra: The spectrum of the prompt emission (from keV to MeV energies) for most of the GRBs is non-thermal (Figure 2.8). It can be empirically fitted by the phenomenological model called the *Band* function (Band et al., 1993). The model describes the photon flux with two power-laws joining at a break energy:

$$N(E) = \begin{cases} A \left[\frac{E}{100\ \text{keV}}\right]^{\alpha} \exp\left(-\frac{E}{E_0}\right) & E < (\alpha-\beta)E_0 \\ A \left[\frac{(\alpha-\beta)E_0}{100\ \text{keV}}\right]^{(\alpha-\beta)} \exp(\beta-\alpha) \left(\frac{E}{100\ \text{keV}}\right)^{\beta} & E \geq (\alpha-\beta)E_0 \end{cases} \tag{2.1}$$

where A is the normalisation at 100 keV, E_0 is the break energy and α, β are the low and high energy photon spectral indices respectively. An important feature of the spectral shape is that for most GRBs, the energy spectrum ($E^2N(E)$) peaks at a characteristic energy $E_p = (\alpha+2)E_0$. The distribution of E_p ranges from few tens of keV to few MeV (Bošnjak et al., 2014; Gruber et al., 2014). The typical observed values of α have distribution between 0 and -2 while values of β have distribution between -2 to -4 for BATSE bursts (Preece et al., 2000). Similar results were also obtained by later missions (e.g. Barraud et al., 2003; Zhang et al., 2011; Bošnjak et al., 2014; Gruber et al., 2014). It should be noted that till now there is no absolute theoretical model predicting the spectral shape of *Band* function, however, the synchrotron-emission model with multiple segments predicts power-law indices consistent with observations. Moreover, although most GRB spectra can be well fitted with the *Band* function, it was found that some BATSE and *Fermi* GRBs require the combination of a thermal and a power-law component to fit the observed spectrum (Ryde, 2005; Ryde et al., 2010).

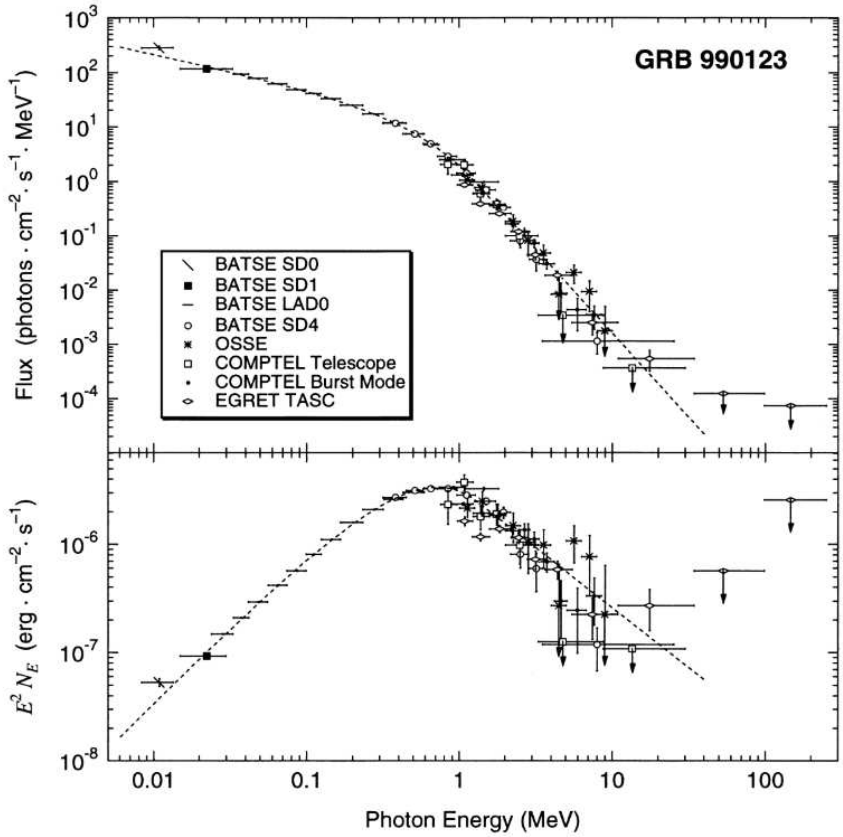


Figure 2.8 – The multi band spectrum of the prompt emission from the GRB990123. The spectrum is well described by the *Band* function. Figure from Briggs et al. (1999).

The interpretation of temporal and spectral properties of the prompt emission leads to the compactness problem (Ruderman, 1975) hinting towards a relativistic origin of GRBs. The argument of the compactness problem is the following – given the distance, energetics and variability timescale ($\sim$ 10 ms) of the GRB emission, the size of the emitting region must be small ($\sim 10^8$ cm) and should have an extreme density of high-energy photons. This implies a very high opacity of the medium due to the photon-photon interactions and copious formation of electron-positron pairs. An estimate for the pair production optical depth gives a value of $\tau_{\gamma\gamma} \sim 10^{15}$ (for typical GRB distances, see Piran, 1995) implying that no high-energy photons should escape as the region is optically thick and the spectrum should be thermal. However, as we detect photons at high energies, the emission region must be optically thin. The problem can be solved if the radiating outflow is moving relativistically towards the observer, as in this case two corrections should be applied, taking into account the Lorentz factor (Γ) of the outflow. The first is that the energy of observed photons in the source frame is reduced by a factor Γ and the second is that the emission region size is greater by a factor Γ^2 (because for a relativistic outflow moving towards an observer, the timescale in the observer frame is shrunken by the factor Γ^2). These corrections lead to an estimate of the optical depth which is less than unity typically for $\Gamma \geq 100$.

Hence there is now a consensus that GRBs are produced by a highly relativistic outflow (jet) originating from a compact central engine and the prompt emission is produced within this jet. The generally accepted theoretical model responsible for the prompt emission is the internal shock model (Rees & Meszaros, 1994; Sari & Piran, 1997, also see figure 2.9). In this framework, a fast-moving shell of matter collides with a slower-

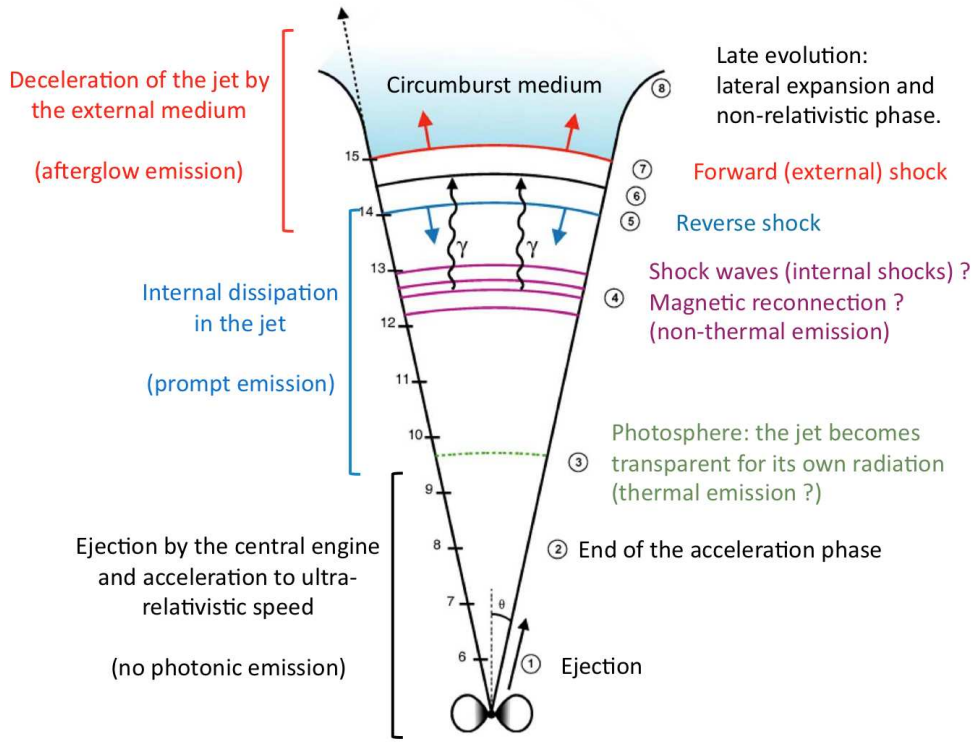


Figure 2.9 – A jet model showing various stages and processes involved in a GRB. Note that the scale on left is the log of distance (with base 10) in metres from the centre of the compact object. The differentially moving, highly relativistic shells of matter collide inside the jet producing the internal shocks, which emit the high energy photons observed in the prompt emission. Along with this, reconnection of magnetic field lines is also thought to contribute to the prompt emission. In some GRBs, observation of the thermal component in the spectrum has hinted at the visibility of the photosphere of the jet. When the ejecta interact with the surrounding medium, they start to decelerate rapidly due to the "forward shocks". These forward shocks are responsible for the afterglow emission, which is usually at lower energy as the ejecta start to decelerate into a non-relativistic regime. In some cases, a "reverse shock" is also observed where some shocked matter travels back into the jet producing a late time optical or UV flash. Image credit: SVOM/IAP team.

moving shell within the jet producing an internal shock. The relative kinetic energy between the two shells is dissipated producing high-energy photons. The most likely mechanism behind the generation of photons is due to the synchrotron (Mészáros & Rees, 1994) and/or synchrotron self-Compton (SSC, Kumar & McMahon, 2008) emission from the shocked electrons. The thermal emission observed in some GRBs is believed to be from the photosphere of the ejecta (Pe'er et al., 2005, 2006). Some studies have also shown that multiple emission processes might be at play in typical

GRBs (Zhang et al., 2011). However, we still lack a conclusive understanding of the prompt emission phenomena and more studies are needed to get the complete picture.

To summarise, though we have a better understanding of the prompt emission now than when the GRBs were first discovered, we still lack conclusive knowledge of many key aspects of the prompt emission (e.g. the composition of the jet, its opening angle, exact radiative processes leading to the high-energy emission etc). It is important to solve these questions because understanding the prompt emission is the first step to understand the GRB phenomena. Also, because the detection of GRB depends on the detection of the prompt emission, good knowledge of the prompt emission properties (e.g. energy range, duration, effects of redshift etc) is important to improve the detection capabilities of future instruments.

2.1.2.2 GRB Afterglow emission

The afterglow phase is empirically defined as the emission phase after the prompt sub-MeV emission has ended. The salient property of the afterglow is that it is a multi-wavelength emission. Unlike the prompt emission, the afterglow emission does not have high temporal variability and the flux density in a given energy band usually follows a power-law decay with time:

$$F_\nu(t) \propto t^{-\kappa} \tag{2.2}$$

where κ is the temporal decay index. The detection of the afterglow in a given energy

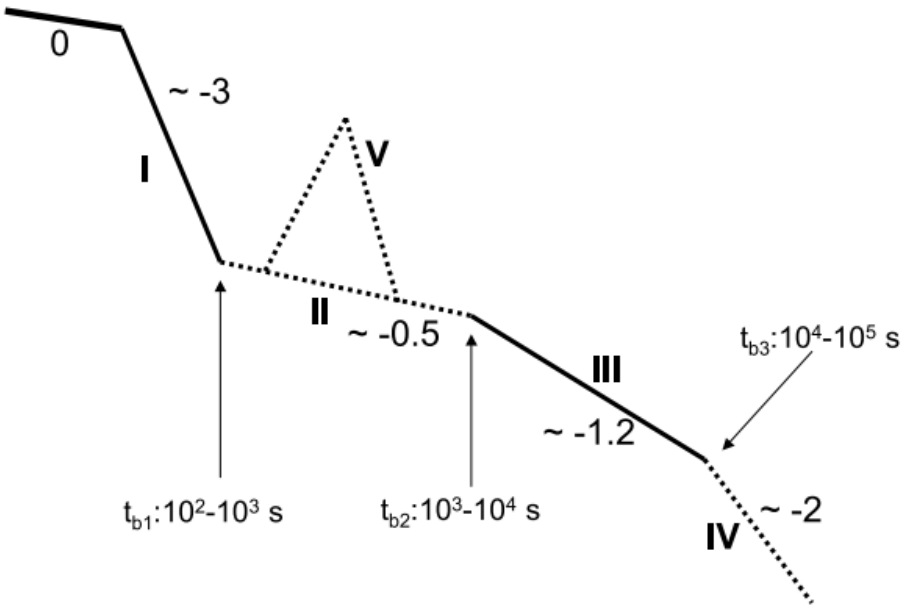


Figure 2.10 – A canonical afterglow lightcurve model for the X-ray flux based on *Swift*/XRT observations. Most of the GRBs observed by *Swift* show at least two of the above five components in their temporal evolution. Image credit: Zhang et al. (2006)

band depends on the brightness of the afterglow as well as the latency of the follow-up observations. The X-ray and very high energy (TeV range) afterglows usually fade away within few hours requiring a fast follow-up, while the optical afterglows are visible for days and radio afterglows last weeks to months. The launch of *Swift* enabled rapid progress in the GRB afterglow observations and understanding. The studies by Nousek et al. (2005) and Zhang et al. (2006) showed that the temporal evolution of the afterglows of most GRBs has one or more of the following (see figure 2.10) five components:

I A steep decay phase observed immediately after the prompt emission, extending up to few hundreds of seconds ($\sim$ 500 s). During this time the flux decreases rapidly with temporal index $3 \leq \kappa \leq 5$. This phase is observed in most GRBs.

II A plateau phase where the flux decay is slower with temporal index $0.2 \leq \kappa \leq 0.8$. This phase is observed only for a fraction of GRBs and lasts usually for $10^3 - 10^4$ s.

III A "normal" decay phase, where the temporal index is $0.7 \leq \kappa \leq 1.2$ which lasts up to $\sim 10^5$ s. This phase is observed in most GRBs.

IV A rapid decay following the "normal" decay phase is also observed in some GRBs with temporal index $2 \leq \kappa \leq 3$ lasting beyond 10^5 s [2].

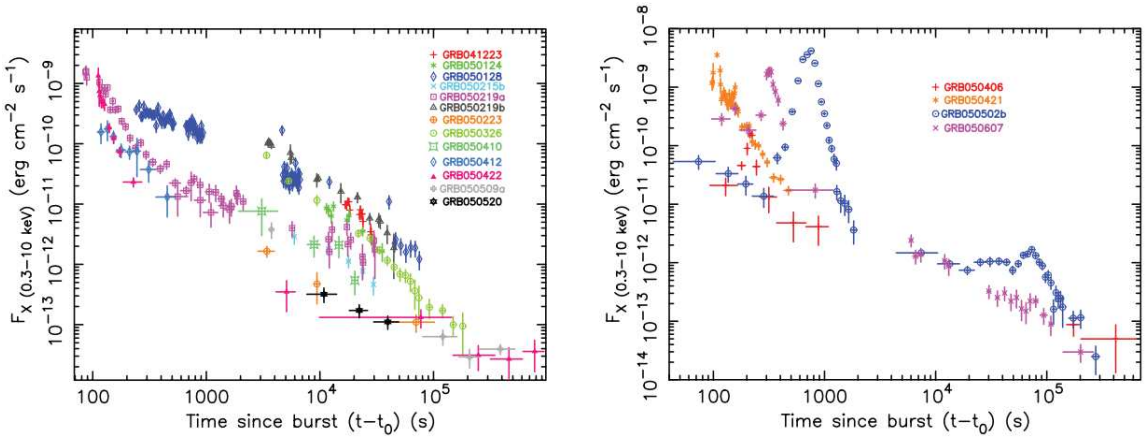
V About 50% of the afterglows show one or more X-ray flares superimposed on any of the above phases. The flares are observed at any time ranging from $\sim$ 100 s after the trigger [3] to as late as 10^5 s.

Observed afterglow lightcurves for a sample of GRBs are shown in figure 2.11. The optical and radio afterglows are fairly regular with most GRB afterglows showing an early rise with a power-law decay after reaching a peak luminosity (Oates et al., 2009; Chandra & Frail, 2012). Along with lower energy afterglows, high energy afterglows ($\sim$GeV) showing a power-law decay are also observed for many GRBs (Ghisellini et al., 2010). Recently, a TeV emission was also detected by the MAGIC Cherenkov telescopes for the GRB190114C (Acciari et al., 2019).

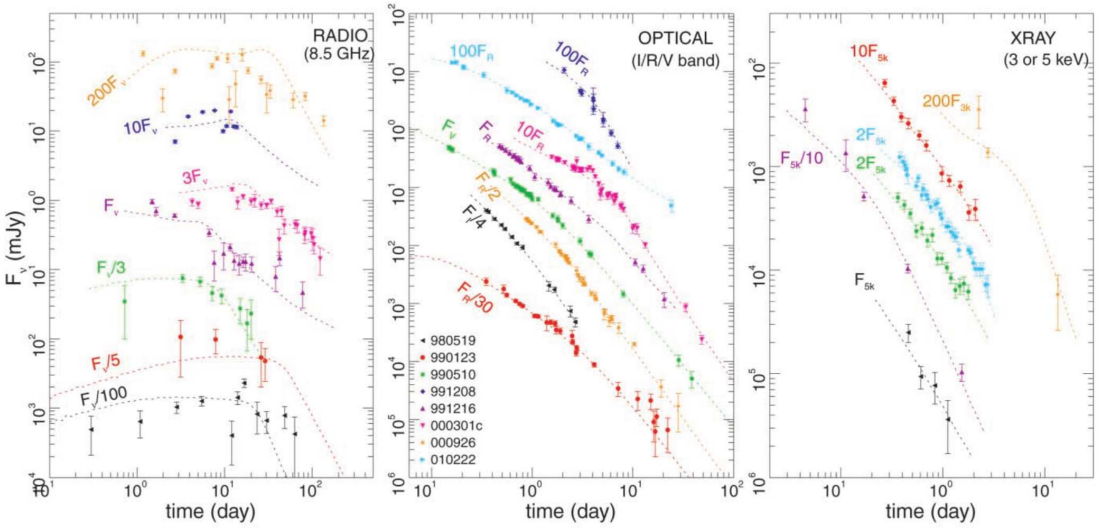
The afterglow emission observations are broadly consistent with the external shock being at the core of the emission (Meszaros & Rees, 1997; Sari et al., 1998). The spectrum at any given time is a power-law, with spectral indices suggesting synchrotron or synchrotron self-Compton emission in the external shock. Theoretically, the different

2. This decay is explained by the so-called "jet break" observed in many X-ray/Optical afterglow lightcurves. The explanation is the following: If we suppose that the observer line of sight is within the opening angle of the jet (θ_j). Due to the relativistic beaming, only the emission coming within a cone of opening angle $\theta_e \sim 1/\Gamma$ contributes to the observed flux. As the jet expands, at one moment in time, $\theta_e \geq \theta_j$ and the observer sees a sudden deficit in the observed flux. This is marked as a sudden steepening or "break" in the observed lightcurve.

3. Trigger here refers to the time at which a space-based GRB instrument detects the prompt emission signal above a certain detection threshold.



(a)



(b)

Figure 2.11 – Observed afterglow lightcurves for a sample of GRBs: (a) X-ray afterglow lightcurve of a sample of GRBs showing some of the five components depicted in the canonical picture shown in Figure 2.10, image from Nousek et al. (2005). (b) Optical, radio and X-ray lightcurves of sample of GRBs, image from Panaitescu & Kumar (2001)

temporal and spectral decay components observed in the afterglows can be attributed to different emission sites in the external shock (see figure 2.9) – the forward shock, the reverse shock or late time central engine activity dissipating internal energy in the outflow (e.g. in case of X-ray flares). Understanding the contributions from all the emission sites is important to understand the whole afterglow emission (Zhang, 2018).

2.1.2.3 GRB progenitors and central engine

The observations of the prompt emission show a phenomenological short/long duration separation indicating two different classes of GRBs. Although the separation is not sharp with observational biases depending on the instrument energy range, it

does indicate that there are two different types of progenitors for GRBs. Better evidence of the different progenitor hypothesis was obtained when follow-up afterglow observations were possible with *BeppoSAX*, *HETE-2* and *Swift*. Those observations showed that most of the long GRBs are found in dwarf galaxies with star-forming activity (Fruchter et al., 2006). Also, some long GRBs were associated with Type Ic SNe (e.g. Stanek et al., 2003; Modjaz et al., 2006). These observations suggested that long-duration GRBs are due to the core collapse of a massive star, and particularly, that such a GRB is associated with the supernova explosion of a Wolf-Rayet star as a progenitor, whose hydrogen and helium envelopes have been stripped off (Woosley & Bloom, 2006). Unlike long GRBs, *Swift* observations of short GRB afterglows showed no SNe connection (Berger, 2014). Also, the host environment of nearby short GRBs revealed that most short GRBs are found in elliptical or early-type galaxies with little or no star formation. Those which do occur in star-forming galaxies are usually localised in the low star-forming regions of the host galaxies. These observations indicated that most short GRBs have a non-collapsar origin and that their progenitors are most likely compact star binaries (black hole - neutron star or binary neutron stars), a hypothesis confirmed in 2017 (see section 2.1.3.1). Although progenitors of GRBs can be distinguished roughly based on the duration, there are some exceptions for this classification. For example, the GRB060614, a long duration burst ($\sim$ 102 s), has properties similar to short bursts except for its duration, casting doubt on the massive star collapse or merger scenario in this case (Gehrels et al., 2006). Similarly, it has been suggested that some short-duration bursts, particularly those observed at high redshift with high luminosity, are also associated with a massive star collapse (Zhang et al., 2009). These exceptions make GRB progenitor classification still a very open topic of study.

Based on the observations, two models are proposed for the central engine powering all GRBs (short or long) irrespective of their progenitors (see Kumar & Zhang, 2015, and references therein). The first model predicts a hyper-accreting black hole engine (Woosley, 1993). In this case, the jet is powered by the accretion disk around a stellar-mass black hole. The jet is thought to be produced either by Neutrino annihilation along the spin axis of the black hole (baryon dominated jet) or by the Blandford-Znajek mechanism (Poynting flux dominated jet, Lee et al., 2000). The second model for a central engine predicts a millisecond-magnetar engine (Metzger et al., 2011). In this case, a rapidly spinning (period $\sim$ 1 ms), highly magnetized (surface magnetic field $\sim 10^{15}$ G) neutron star ("magnetar") is believed to launch the GRB jet. This jet is baryon dominated at early times ($<$ 10 s) and gradually becomes Poynting flux dominated as the magnetar cools down.

With more observational evidence, such as detection and study of a larger population of SNe associated with GRBs and detection of more gravitational wave events

associated with GRBs, we will be able to get a clearer picture of the nature of GRB progenitors.

2.1.3 The astrophysics of gamma-ray bursts

2.1.3.1 GRBs as multi-messenger objects

A multi-messenger source emits radiation with different types of messengers such as photons, particles (cosmic rays and neutrinos) or gravitational wave radiation. GRBs are considered to be multi-messenger sources as additionally to their photon production, they can emit gravitational waves and are also thought to produce ultra-high-energy cosmic rays (UHECRs) along with high-energy neutrinos.

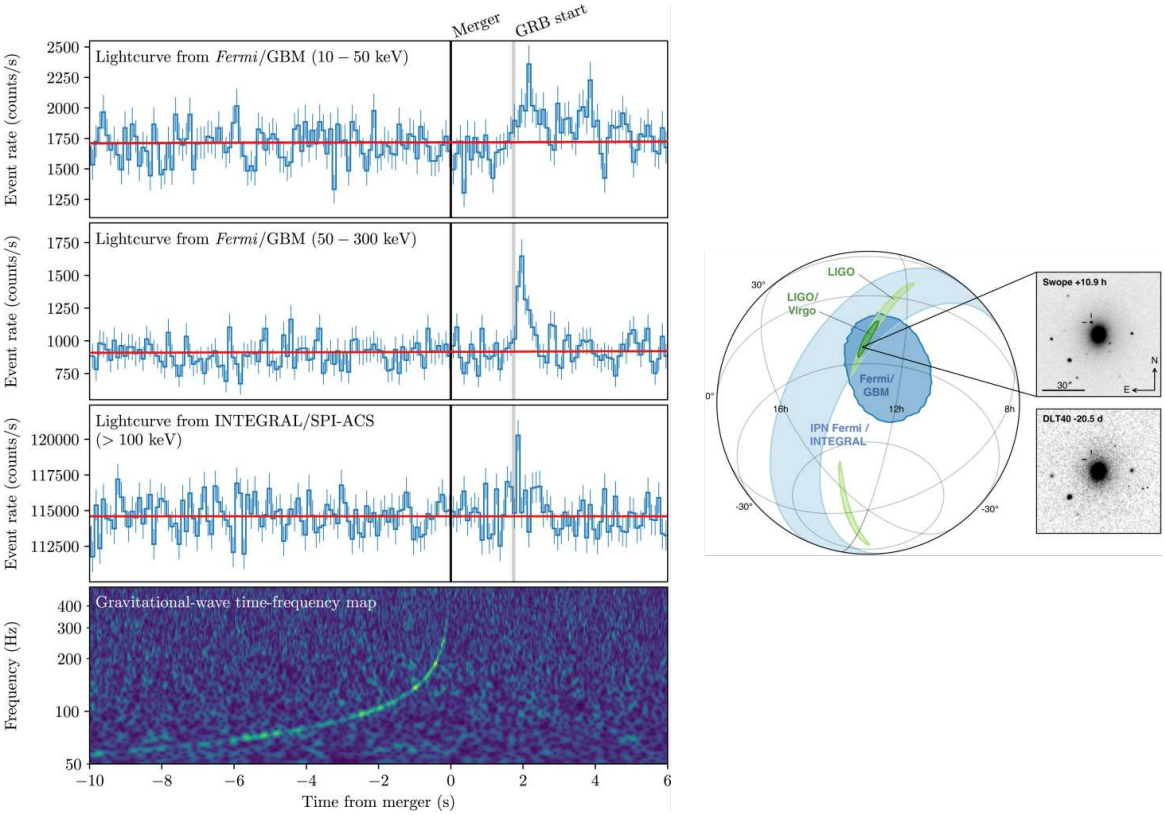


Figure 2.12 – The simultaneous detection of GRB 170817A and gravitational waves from a binary neutron star merger (GW 170817). Left: GRB signal in *Fermi*/GBM and *INTEGRAL*/SPI-ACS co-aligned in time with the gravitational wave signal in the LIGO detectors. The merger and GRB start times are marked with the black and grey line respectively. The delay of about 1.7 seconds of the GRB signal can be seen. Right: The initial localisation region of the LIGO/Virgo (dark green), *Fermi*/GBM only (dark blue) and interplanetary network (IPN) using *Fermi* and *INTEGRAL* (light blue) along with the optical images of the host galaxy NGC 4993 taken 10.9 hours (top) after the event and a reference image taken 20.5 days (bottom) before the event. The presence of a transient can be seen in the reticle marks in the top image. Figure from Abbott et al. (2017a)

The first conclusive proof of multi-messenger emission from a GRB came in 2017 when a short GRB (GRB170817) and a gravitational wave signal (GW170817) from

a binary neutron star (BNS) merger were observed simultaneously (Figure 2.12 - left). On 2017 August 17, the gravitational wave signal from the merger (at 12:41:04 UTC) was detected by the LIGO Hanford and Livingston detectors (Abbott et al., 2017b). The event was localised within a 28 deg^2 (90% probability) region on the sky (Figure 2.12 - right). Around the same time, with a delay of about ~ 1.7 s, the *Fermi*/GBM detectors were triggered by a GRB event ($T_{90} = 2 \pm 0.5$ s). The *INTEGRAL* observatory also detected the GRB in its offline analysis (Abbott et al., 2017c,a). Responding to the LIGO/Virgo and GRB triggers, an extensive multi-wavelength observation campaign was launched by many observatories around the world to look for a counterpart. The optical counterpart was found ∼10 hours after the trigger, localised in the galaxy NGC 4993. The most significant impact of this discovery was the confirmation of binary neutron star mergers as short GRB progenitors. The afterglow observations conclusively indicated the existence of a "kilonova" (e.g. see Kasliwal et al., 2017; Tanvir et al., 2017), an explosion event that has a brightness between the one of a classical nova and a supernova. The observations also showed that the kilonova is powered by the *r*-process [4] nucleosynthesis confirming theoretical predictions (Metzger et al., 2010). Another peculiar property of this event is that in spite of being the closest (∼40 Mpc) short GRB detected to date, it was much fainter than any other short GRB. This indicated the GRB170817 was viewed from a high off-axis angle (between the jet axis and Earth). Radio observations of the afterglow (Mooley et al., 2018) confirmed this hypothesis in which the viewing angle is predicted to be between $14° - 28°$.

Although for GRB170817A we have a coincident gravitational wave detection, we haven't detected high energy neutrinos coincident with this GRB as well as for any other GRB to date. Also, at the time of writing this thesis, there is no other confirmed multi-messenger detection. The O3 run of LIGO/VIRGO did have some binary neutron star candidates (e.g. GW190425), however, there was no confirmed electromagnetic counterpart detection.

2.1.3.2 GRBs as cosmic rulers

For the past two decades, type Ia supernovae have been good standard candles to study the cosmological parameters at different epochs. However, they suffer from an extinction problem and can be only observed up to redshift $z \sim 1.7$. On contrary, GRBs (in particular long GRBs, with tens to hundreds of seconds duration) do not suffer from this and can be observed up to very high redshifts (the farthest GRB

4. The *r*-process refers to a nuclear reaction where a heavy seed nucleus is subjected to rapid and successive neutron captures. The process is rapid in the sense that the nucleus doesn't have time to undergo radioactive decay before the next neutron is captured. The *r*-process is responsible for creating approximately half of the nuclei heavier than Iron.

with confirmed spectroscopic redshift is GRB090423A with $z = 8.2$). Although GRBs are not standard candles, as their intrinsic luminosity varies within few orders of magnitude ($\sim 10^{49} - 10^{52}$ ergs/s), they do show some correlations between measured parameters and intrinsic parameters (e.g Amati et al., 2002; Ghirlanda et al., 2004a). These relations can be used to estimate the luminosity distance and thereby obtaining the cosmological parameters at a given redshift (e.g. see Ghirlanda et al., 2004b; Schaefer, 2007). The key requirement here is the measurement of the GRB redshift. Some studies have disputed the observed correlations owing to the selection effects and the limited sample of GRBs with know redshift (e.g. see Heussaff et al., 2013; Turpin et al., 2016). Because the GRB sample with known redshift is still small (about a few hundred, compared to thousands in the case of type Ia SNe), more GRB redshift measurements are needed to conclusively use GRBs as cosmic rulers.

2.1.3.3 GRBs as cosmic and galactic probes

The cosmic nature of GRBs can also be exploited to study the universe at different epochs. The long GRBs are believed to have massive stars as progenitors (see section 2.1.2.3). As massive stars are usually found in star-forming regions, GRBs can be useful tools to probe the star-forming history of the universe (e.g Gou et al., 2007). Furthermore, the emission from a GRB travels through the host galaxy as well as through the intergalactic medium. Hence, the study of absorption/emission lines in the spectrum of a GRB afterglow can shed light on the host galaxy environment, its interstellar medium (ISM) as well as the environment along the line of sight in the intergalactic medium (IGM), providing clues about the metal enrichment of the universe (Prochaska et al., 2004).

2.1.4 GRBs: future prospects

Our understanding of GRBs is not at all complete and there are a lot of open questions yet to be answered (e.g. exact mechanism behind the prompt emission, properties of the central engine, standard picture for progenitors, GRB rates). The upcoming GRB missions (see section 1.2) will try to solve some of these questions. In addition, these missions will operate in a promising era of transient astrophysics. In particular, most of the future missions will benefit from more sensitive gravitational wave detectors improving our understanding of short GRBs and their connection with compact object mergers. This will be also complemented by more sensitive Neutrino detectors (e.g. KM3NeT, IceCube-Gen2) and cosmic-ray detectors (JEM-EUSO) making GRB science truly a multi-messenger field. Furthermore, sensitive large-scale survey facilities such as Rubin observatory/LSST (optical) and SKA (radio) will play an important role to complement the monitoring of the sky at wavelengths

other than X/gamma-rays[5]. The GRB studies will also benefit from more sensitive and fast follow-up observatories such as CTA (high-energy gamma-rays) and FAST (radio) improving our understanding of GRB afterglows.

In conclusion, the coming decade will be exciting for GRBs as well for high energy transient studies. The role of *SVOM* in this context is discussed in the next chapter.

2.2 X-ray / Gamma-ray transients other than GRBs

In this section, I will briefly talk about transients other than GRBs which will be detected by the *SVOM*/ECLAIRs. The discussion is structured in a roughly increasing order of distance scale at which the transients are located (from the closest to the Earth to the furthest).

2.2.1 Terrestrial Gamma-ray Flashes

One kind of high-energy transients observable by space instruments is produced right here on Earth in its atmosphere, it is called Terrestrial Gamma-ray Flash or TGF. It is a brief pulse of gamma-rays produced in the upper atmosphere during thunderstorm activity. The typical duration of a TGF is less than a millisecond with observed photons having energies up to few tens of MeV ($\sim$ 20 MeV). The TGFs were first detected serendipitously by the *CGRO*/BATSE instrument (Fishman et al., 1994). Later, other missions like *RHESSI* and *Fermi* have detected TGFs (Smith et al., 2005; Briggs et al., 2010). Currently, *Fermi* with its offline search detects about 800 TGFs per year (Roberts et al., 2018). A typical TGF detected by *Fermi* is shown in figure 2.13.

The origin of these flashes is thought to be due to the Relativistic Runaway Electrons Avalanche process (RREA) in the electric field produced by lightning thunderstorms (Gurevich et al., 1992; Dwyer, 2003). The electrons in this field are accelerated to high energies and interact with the nuclei of the atoms in the atmosphere generating gamma-ray photons via the bremsstrahlung process. The observation of such extreme phenomena in the Earth's atmosphere is one of the most surprising discoveries in the field of space-based observations.

2.2.2 Solar flares

The existence of solar flares was known for more than a century, however, the detection of gamma-rays from a solar flare was a surprising discovery made by the

5. For example, LSST is expected to detect "orphan afterglows" (an afterglow of GRB where the jet is not pointed towards the Earth and the prompt emission is not seen) and such observations can be important to put constraints on the true GRB rates as well as the beaming angle of GRB jets.

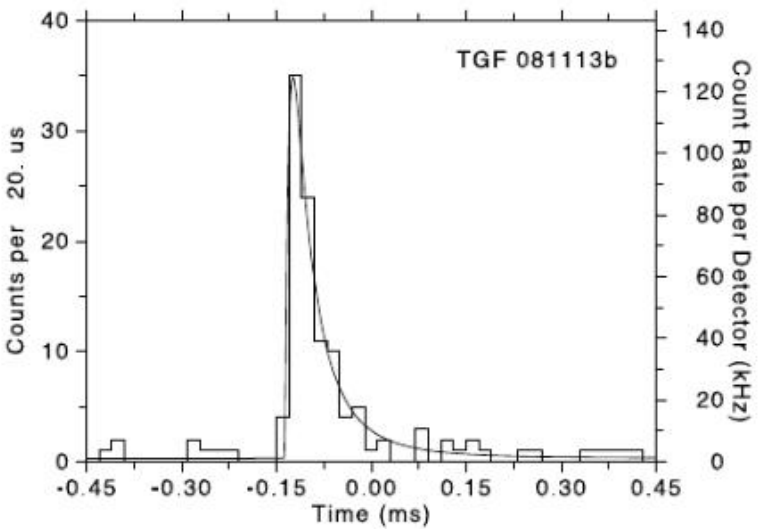


Figure 2.13 – A TGF detected by the *Fermi*/GBM. Notice the very short (< 1 ms) duration of the flash. Image from Briggs et al. (2010)

OSO-7 mission in 1972. The flares occur near the surface of the sun around the active region of sunspots. A sudden release of magnetic energy in the Sun's corona produces a solar flare. The emission observed in hard X-rays and soft gamma-rays is produced by bremsstrahlung radiation from the electron-ion pairs in the flare. The flares can release as much energy as 10^{32} ergs per second.

2.2.3 Transient X-ray binaries

X-ray binaries refer to a class of binary systems where one star is a compact object (neutron star or black hole) that accretes matter from a companion star. Several binaries are transient, i.e. they alternate between periods of quiescence and outburst (bright X-ray emission) with a large range of outburst timescales (minutes, days or even years). X-ray binaries are usually classified into two categories depending upon the mass of the donor star – Low Mass X-ray Binaries (LMXBs) and High Mass X-ray Binaries (HMXBs). The mechanism by which the accretion happens in the two classes is also different (e.g. by Roche-lobe overflow or by stellar wind respectively, Frank et al., 2002). The transient outbursts observed also have different properties depending on the mass of the donor star as well as the type of the compact object.

LMXB transients

In LMXBs, the donor star is usually a K or M-dwarf star. The X-ray emission in LMXBs comes from the accretion disk formed around the compact object due to the Roche-lobe overflow of the companion star. LMXBs with a black hole as a compact object usually show very bright outbursts, even up to the Eddington luminosity at the peak of the emission (see e.g. Remillard & McClintock, 2006; Done et al., 2007).

The outburst is due to the change in the mass accretion rate by the black hole.

The neutron star LMXB show different kinds of transient activities. There are X-ray bursters (e.g. see Lewin et al., 1993), producing the so-called Type I burst which is due to a thermonuclear runaway on the surface of the neutron star. The hydrogen accreted on the surface of the neutron star is steadily converted into helium forming an envelope around it. When this envelope reaches a critical mass, the helium is ignited explosively releasing large amounts of energy in form of blackbody radiation peaking in the X-rays. The duration of a single burst is usually between few seconds and few minutes. The bursts can be observed with an occurrence period varying between hours, days or longer. The source can become inactive for durations as long as several months. Apart from the Type I bursts, some sources (e.g. MXB 1730-335 called the Rapid Burster and GRO J1744-28, see Lewin et al., 1976; Daigne et al., 2002) show another type of bursts, called type II bursts. The distinction is based on the burst profiles[6]. The Type II bursts last for few seconds (up to ∼10 s) and are more frequent with an occurrence time interval of several seconds to minutes. Observations indicate that the Type II bursts have a different physical origin not related to a thermonuclear process but mostly attributed to instabilities in the accretion disk (e.g see van den Eijnden et al., 2017). Finally, there are also accretion powered pulsars that exhibit periodic pulses of X-ray emission (e.g. see Campana & Di Salvo, 2018; Patruno & Watts, 2021). The matter flow from the companion star is channelled onto the neutron star poles by the magnetic field producing a "hot-spot" pair. The bright hot-spots come in and out of the observer line of sight as the neutron star spins, producing the observed pulsation.

HMXB transients

The HMXBs have an O or B type donor star. The X-ray emission in HMXBs is due to the accretion of matter ejected by the donor star in the form of a stellar wind. Most of the transient HMXBs (about 2/3) are Be/X-ray transients (BeX transients) where the donor star is a main-sequence (Be) star and the compact object is a neutron star. The Be stars have strong equatorial outflows of matter (disk winds) mainly due to their rapid rotation. The BeX systems have eccentric orbits and near the periastron passage, the neutron star interacts with the stellar wind causing an X-ray outburst as the matter is accreted onto the neutron star. The duration of these outbursts is of the order of days and they occur fairly periodically with time intervals corresponding to the orbital period of the system (see Paul & Naik, 2011, and references therein).

Another class of HMXB transients, called Supergiant Fast X-ray Transients (SFXTs) are also observed. In these systems, the donor star is an OB type supergiant. Out-

6. Typically Type I bursts have a rapid rise - slow decline profile while Type II bursts start and stop abruptly without any gradual decay from the peak

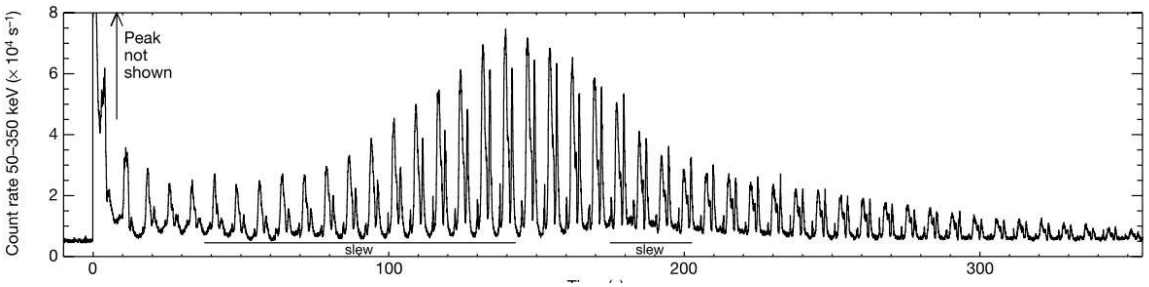


Figure 2.14 – The *Swift*/BAT lightcurve of SGR 18062-20 as observed on 27 December 2004. A giant flare was detected at the start of the burst activity. The "global" variation in the intensity of the burst peaks is due to the slewing of the spacecraft (manoeuvre to reorient the pointing axis of the telescope to a different direction). Image from Palmer et al. (2005)

bursts from these systems show very bright short-duration flares (of the order of hours) compared to the BeX systems. Furthermore, outside the outburst phase, the sources are often not detected (or detected as weak sources) due to their low luminosity in the X-ray band. Based on the observations of a few sources exhibiting periodic outbursts (e.g. IGR J11215-5952, Swank et al., 2007), the compact object is thought to be a neutron star. The bursts are likely to be produced by the accretion of a clumpy stellar wind of the supergiant companion onto the neutron star (Bozzo et al., 2017).

2.2.4 Soft Gamma Repeaters

Soft Gamma Repeaters (SGRs) are sources that emit short bursts ($\sim$ 100 ms) of gamma-rays and hard X-rays at irregular intervals during an active phase. In the period of inactivity, SGRs are ordinary quiescent X-ray sources. The first such source, SGR 0526-66, was observed by the Soviet *Venera 11* and *Venera 12* spacecrafts on March 5, 1979 (Mazets et al., 1979). It was localised in a supernova remnant in the Large Magellanic Cloud. Since then, especially after the launch of *Swift*, a handful of SGRs (about 15) have been detected[7] (Olausen & Kaspi, 2014), mostly located within the Milky Way (few tens of kiloparsecs). Many SGRs, including the March 1979 event, have shown a giant gamma-ray flare along with a bursting activity (e.g. see figure 2.14). SGRs are believed to be "magnetars" (neutron stars with magnetic field $> 10^{14}$ G) emitting high energy bursts due to the large scale magnetic rearrangements or due to the release of magnetic stresses within the neutron star's crust (Thompson & Duncan, 1995). The SGRs are much brighter compared to the X-ray bursts from LMXBs (discussed previously) and also have a harder spectrum than the X-ray bursts.

7. http://www.physics.mcgill.ca/~pulsar/magnetar/main.html

2.2.5 Tidal Disruption Events

The Tidal Disruption Events (TDEs) are transient events (flares) due to a tidal disruption of a star by a supermassive black hole. A theoretical prediction of such events (e.g. Frank & Rees, 1976) was done much prior to their detection in the 1990s. If a star's orbit passes below a critical radius near a supermassive black hole (mostly observed at the centre of galaxies), then the star is tidally disrupted [8] by the black hole forming an accretion disk which then emits electromagnetic radiation (see figure 2.15). The duration of these events varies from days to years. Most of the known TDEs [9]

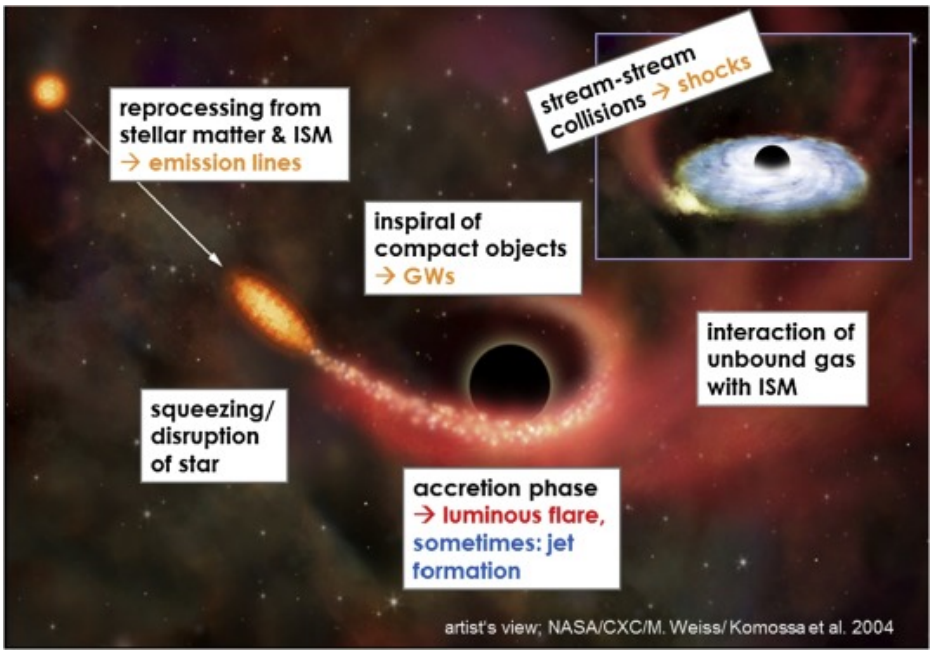


Figure 2.15 – A representative picture of a TDE. When a star passes very close to a supermassive black hole it is tidally disrupted and an accretion disk is formed. As the matter is accreted, radiation is emitted from the disk or in some cases from a jet. Image from Komossa (2015)

are observed in the optical, UV or soft X-ray bands (see a review by, Komossa, 2015), however *Swift* has observed two TDEs (Swift J1644+57, Bloom et al. (2011) and Swift J2058+0516 0.353, Cenko et al. (2012)) in the hard X-rays and the observations indicate a jet formation; a phenomena not observed in other low-energy TDEs. This discovery has opened discussions about the theoretical modelling of these sources requiring more observations of such jetted TDEs to understand the different scenarios (i.e. TDEs with or without jets).

8. Black hole mass must be between $\sim 10^5$ $M_\odot - 10^8$ $M_\odot$ for tidal disruption to be possible. For black holes with masses $> 10^8$ $M_\odot$, the star is directly "swallowed".
9. A TDE catalogue can be found here - https://tde.space/

2.2.6 Supernovae and Active Galactic Nuclei

Other transient sources at high-energy we can include are early-time emission from Supernovae (SNe) and flares in Active Galactic Nuclei (typically in Blazars). Supernovae of both classes, core-collapse and thermonuclear, are expected to produce transient X-ray emission, at early times (at the moment when the star explodes), in form of a flash due to the "shock breakout" of the ejecta (this emission is expected to last from few seconds to minutes). AGN flares are mostly due to the increased activity of the accreting supermassive black hole at the centre of the AGN host galaxy. The typical duration of these events can vary from hours to years (Gehrels & Cannizzo, 2013).

To summarise, the high-energy transient sky is diverse in terms of temporal, spatial and physical properties of the sources. Observations of these sources have improved our understanding of some of the most extreme phenomena in the universe. The diverse nature of sources causing transient events have also enabled the study of the evolution of the universe as well as the impact of some of these events on their local environment.

Chapter 3

The SVOM mission and the soft gamma-ray imager ECLAIRs

3.1 The SVOM mission

The *Space-based multi-band astronomical Variable Objects Monitor* or *SVOM* (Cordier et al., 2014; Wei et al., 2016), is a Sino-French mission dedicated to the multi-wavelength studies of GRBs. It is currently scheduled to be launched in June 2022 by the Chinese space agency's LM-2C rocket. The nominal duration of the mission will be three years with an additional extension of two years.

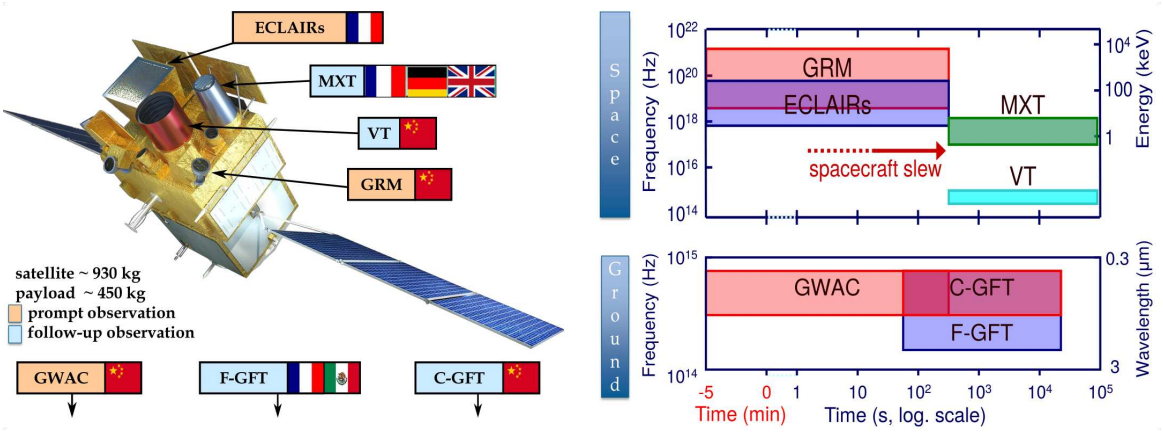


Figure 3.1 – Left: View of *SVOM* instruments and the partner countries responsible for each instrument. Right: Spectral coverage of *SVOM* instruments as function of time. Image credit: *SVOM* collaboration.

SVOM has seven instruments (Figure 3.1 left) separated into a space segment (4 instruments) and a ground segment (3 instruments). Together they cover seven orders of magnitude (with gaps) in wavelengths (Figure 3.1 right). The two high-energy wide-field instruments, ECLAIRs and Gamma Ray Monitor (GRM) will monitor the sky to detect and localise (mainly by ECLAIRs) GRBs. The narrow-field instruments,

Micro-channel X-ray Telescope (MXT) and Visible Telescope (VT) will follow-up the GRBs from space (when satellite slew is possible to bring the GRB location in their FoV, see figure 3.2) while the Ground Follow-up Telescope (GFT) will follow-up the GRBs from the ground. The Ground Wide Angle Camera (GWAC) will monitor a fraction of the sky simultaneously with ECLAIRs to detect prompt optical emission from GRBs.

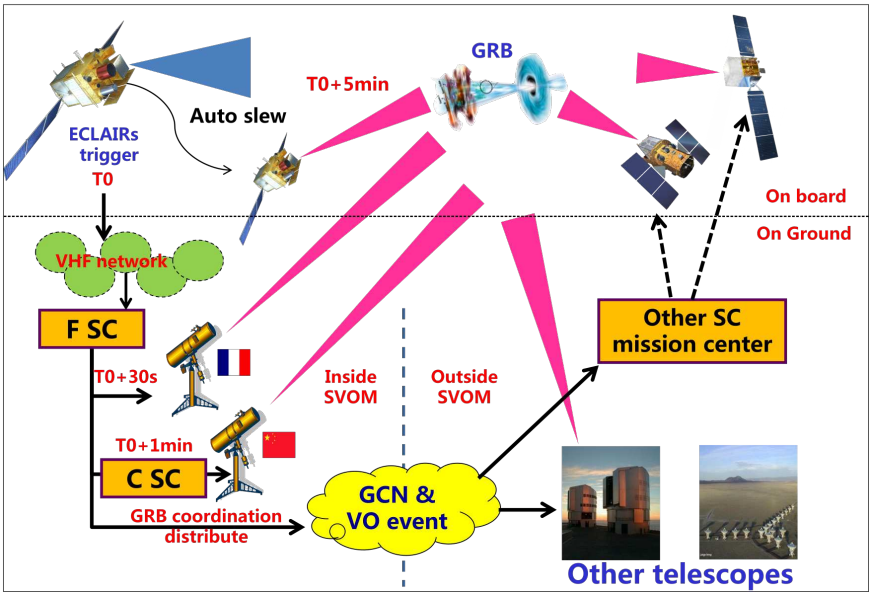


Figure 3.2 – A typical GRB observation sequence by *SVOM*. Image credit: *SVOM* collaboration.

SVOM is designed to rapidly detect and follow-up GRBs. A typical GRB observation sequence by *SVOM* is shown in figure 3.2. The GRB detection and localisation are carried out by ECLAIRs (using an onboard trigger, see Chapter 4). The localisation information (few arcmin accuracy, see section 3.2) is then sent to the ground telescopes for follow-up observations via a VHF network (see section 3.1.3.2). The localisation is also shared with the spacecraft to initiate an autonomous slew to bring the GRB location into the FoV of MXT and VT and begin a space-based follow-up. The VT and MXT search for the afterglow and localise it with better accuracy (less than arcsec accuracy with the VT, see section 3.1.1.3 and 3.1.1.4). Both pieces of localisation information (initial one by ECLAIRs and refined one by VT and MXT) are shared with the scientific community outside of *SVOM* collaboration via GCN or VOEvent for additional follow-up by other space-based or ground-based facilities.

In addition to GRBs, *SVOM* will study other high energy transients (e.g. TDEs, SGRs, SNe) and carry out multi-wavelength observations of known high energy sources (e.g. X-ray binaries, pulsars, AGNs).

In the following subsections, I'll give details about *SVOM* instruments, mission operation and its scientific objectives and prospects.

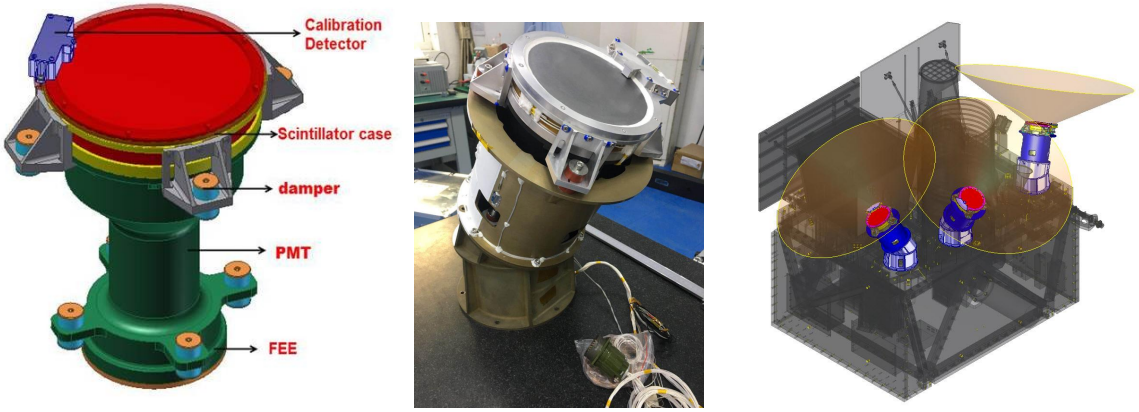
3.1.1 SVOM space segment

3.1.1.1 ECLAIRs

ECLAIRs (Godet et al., 2014), a high-energy coded-mask imager, is the primary instrument onboard *SVOM*. Its main goal is to detect and localise GRBs as well as other high-energy transients in real-time. It has a wide field of view ($\sim$ 2 sr) and it is sensitive in the energy range of 4 - 150 keV. A more detailed overview of ECLAIRs and its characteristics is given in section 3.2.

3.1.1.2 Gamma Ray Monitor

The Gamma Ray Monitor (GRM, Dong et al., 2010) consists of three spectroscopic Gamma Ray Detectors (GRDs, figure 3.3a and 3.3b) each placed on one side of the payload platform (Figure 3.3c). A GRD is made of Sodium Iodide, NaI(Tl),



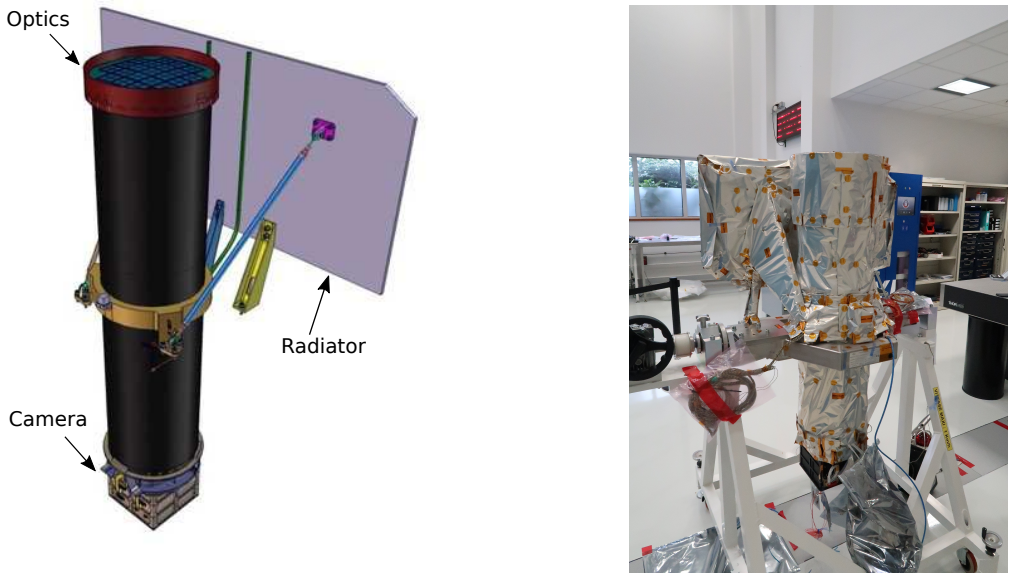
(a) (b) (c)

Figure 3.3 – The Gamma Ray Monitor (GRM) instrument onboard *SVOM*: (a) A schematic diagram of single Gamma Ray Detector (GRD) showing its main components. (b) Picture of the qualification model of a single GRD. (c) Figure showing placement of GRDs on the payload platform and the FoV of each GRD. Image credits: GRM instrument team (IHEP).

scintillator crystal which are 1.5 cm thick and 200 cm^2 in area. The GRD is sensitive from 15 keV to 5 MeV and has an energy resolution of 16% at 60 keV. It can record gamma-ray photons with a time resolution of 20 μs. Each GRD has an FoV (conical shape) of $\pm 60°$ ($\sim$ 3.14 sr) around the central axis. This results in a common FoV $\sim$2 sr which is about the same as ECLAIRs. Within the common FoV, GRM can roughly localise a transient with $\sim$ 5° accuracy. GRM will have its own onboard trigger which will share the time and crude localisation information with ECLAIRs for trigger enhancement. GRM is expected to detect >90 GRBs/yr.

3.1.1.3 Microchannel X-ray Telescope

The Microchannel X-ray Telescope (MXT, Götz et al., 2014b) is a focusing soft X-ray telescope (Figure 3.4) sensitive in the 0.2 - 10 keV range. MXT will use "Lobster Eye" optics (Angel, 1979) to focus the X-rays (focal length 1 m). The optics cover the FoV of 64 × 64 square arcmin. The pn-CCD [1] detectors (Meidinger et al., 2006) are used to record the focused X-rays. The detector has 256 × 256 pixels with an effective area of 23 cm^2 at 1 keV and an energy resolution of ∼80 eV at 1.5 keV.



(a) (b)

Figure 3.4 – The Micro-channel X-ray Telescope onboard *SVOM*: (a) A schematic diagram showing the main parts of MXT. (b) Picture of the final structural and thermal model of MXT. Image credit: MXT instrument team (CEA/Saclay, CNES).

MXT will be sensitive down to the flux of $\sim 10^{-12}$ erg cm^{-2} s^{-1} in 10 ks exposure in the 0.3 - 6 keV range. It will be able to localise the afterglows of the GRBs detected by ECLAIRs with an accuracy of less than 1 arcmin. The afterglow position will be estimated onboard by the MXT's data processing unit.

3.1.1.4 Visible Telescope

The Visible Telescope (VT, Fan et al., 2020), is a Ritchey-Chretien optical telescope (Figure 3.5) with a 44 cm diameter, focal length of 3.6 m and FoV of 26 × 26 square arcmin. VT will split the incoming light into two beams; one beam will pass through a blue band (400 nm - 650 nm) and the other will pass through a red band (650 nm - 1000 nm). Each band will have a 2048 × 2048 pixels back-illuminated CCD detector to record the image. VT will have a limiting magnitude of $M_v = 22.5$ for a 300

1. pn refers to the pn-junction used in the transfer registers of the CCD, see also http://www.pnsensor.de/Welcome/Detectors/pn-CCD/index.php

s exposure time and it will be able to localise optical afterglows of GRBs down to sub-arcsec accuracy. To localise the GRB afterglow rapidly, VT will analyse the images on-board.

3.1.2 SVOM ground segment

3.1.2.1 Ground-based Wide Angle Camera

The Ground-based Wide Angle Camera (GWAC, figure 3.6) system, located at the Ali observatory in China, is a set of 40 cameras each having 18 cm aperture and 22 cm focal length. Each camera has 4096 × 4096 back-illuminated CCD detectors sensitive in 500 - 800 nm. The FoV of each camera is 150 square degrees and the combined FoV of the system is ∼ 5400 square degrees with a localisation accuracy of 11 arcsec. For a 5σ detection in 10 s exposure, GWAC has a limiting magnitude of $M_v = 16$ (for a new moon night). GWAC has self trigger capabilities and it can detect and validate transients fully autonomously (Xu et al., 2020a,b). GWAC is operational since 2019 and was used to follow-up the gravitational wave alerts issued by the LIGO/VIRGO collaboration from their recent (April 2019 - March 2020) observing run O3.

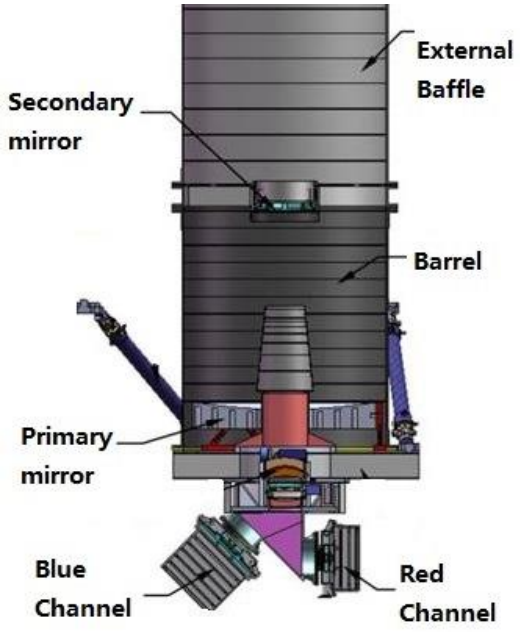


Figure 3.5 – A schematic diagram of the Visible Telescope onboard *SVOM*. Image credit: VT instrument team (NAOC).



Figure 3.6 – The prototype cameras of the GWAC system. Image credit: GWAC instrument team (NAOC).

3.1.2.2 Ground Follow-up Telescopes

The Ground Follow-up Telescopes (GFTs) are two automated robotic telescopes dedicated to the follow-up of *SVOM* detected transients. Both telescopes will be configured to receive the follow-up request directly via the VHF receiver network (see section 3.1.3.2).

COLIBRI : The COLIBRÍ (or French-GFT, Corre et al., 2018; Fuentes-Fernández et al., 2020, figure 3.7a) will be a 1.3 m diameter optical and near-infrared fast follow-up telescope build by a collaboration between France and Mexico. The telescope will be hosted at the national observatory in Sierra San Pedro Mártir, Baja California, Mexico. It will have two imaging instruments; a direct imaging instrument DDRAGO and a re-imaging instrument CAGIRE. DDRAGO will be equipped with 4096 × 4096 CCD and will observe in optical (with B filter [2] and Pan-STARRS $grizy$ filters [3]) and has a FoV of 26 × 26 square arcmin (equal to ECLAIRs source localisation error diameter). CAGIRE will be equipped with a 2048 × 2048 sensor and will observe in the infrared (Pan-STARRS J and H filters) with a slightly smaller FoV of 21.7 × 21.7 square arcmins compared DDRAGO. One notable feature of COLIBRÍ is that it will be able to point at any source in less than 20 s. COLIBRÍ can reach limiting magnitude (for 5σ detection) 20.5 in r band and 19 in J band in 60 s exposure time.

C-GFT : The Chinese Ground Follow-up Telescopes (C-GFT [4]) will be a 1 m diameter optical telescope with fast follow-up capability. It will be located at the Jiling Observatory in China. It will have an FoV of 21 × 21 square arcmins. C-FGT can observe in 3 wavelength channels defined by the SDSS [5] g, r, i (see Fukugita et al., 1996, for wavelength details) filters. The C-GFT will have a limiting magnitude (in r channel) of 19 (for 5σ confidence detection) in 100 s exposure. C-GFT will be able to process images and provide GRB locations to sub-arcsec precision within 5 min after it receives a follow-up request from ECLAIRs trigger.

3.1.3 SVOM: Mission profile and operations

3.1.3.1 Satellite orbit and pointing

The *SVOM* satellite will have a Low Earth Orbit (LEO) with an inclination of 30°, an altitude of ∼ 625 km and an orbital period of ∼ 96 min. The altitude and inclination provide good protection (due to the Earth's magnetosphere) to the

2. see Bessell (2005) for wavelength details
3. see Tonry et al. (2012) for wavelength details
4. https://grandma.lal.in2p3.fr/observatories/svom-cgft/
5. Sloan Digital Sky Survey



(a) F-GFT **(b)** C-GFT

Figure 3.7 – The Ground Follow-up Telescopes (GFT) of *SVOM*. (a) The COLIBRÍ or French GFT (b) The Chinese GFT. Image credit: GFT instrument teams (LAM, UNAM, NAOC).

instruments from the energetic particles from space. However, due to the low altitude and high inclination, *SVOM* will pass through the South Atlantic Anomaly (SAA) several times a day ($\sim$ 6 orbits/day) encountering a high flux of trapped protons and electrons. These passages require temporary shut down of instruments which will result in a dead time of 13% - 17% per day.

Following some constraints such as avoidance of the sun, galactic sources in the FoV of instruments and thermal cooling requirements; *SVOM* satellite pointing will follow an attitude law called "B1 attitude law" (Cordier et al., 2008) for most of the observation time. In accordance with this law, the pointing axis of *SVOM* space instruments will point 45° away from the anti-solar direction (Figure 3.8) and away from the galactic plane. The pointing will also try to exclude the presence of Sco X-1 in the ECLAIRs FoV as it is very bright in the 4 - 20 keV range and can reduce the sensitivity of the instrument. The pointing strategy ensures that *SVOM* will always detect a GRB at night time so that it can be followed rapidly by ground-based instruments (by *SVOM* ground segment and/or by other facilities). Furthermore, with this pointing strategy, most GRBs detected by *SVOM* will have positions away from the galactic planes easing redshift determination[6].

However, as a consequence of the LEO and the nearly anti-solar pointing, the Earth will pass through the FoV of the instruments reducing the duty cycle of ECLAIRs to $\sim$ 65% and that of narrow field instruments (MXT and VT) to $\sim$ 50%. This also has a consequence on the background level observed by ECLAIRs whose characterisation is important for transient detection (see Chapter 3 and 4).

6. Redshift measurement is one of the key objectives of *SVOM*, see section 3.1.4

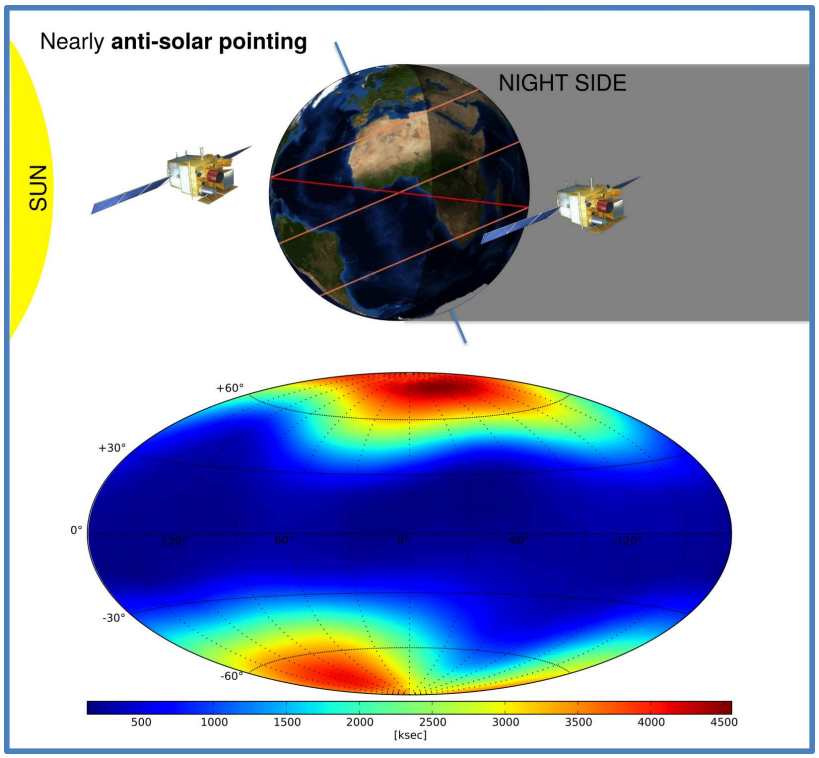


Figure 3.8 – Top: An illustration showing the passage of the Earth in the FoV of *SVOM* space segment instruments as consequence of the anti-solar pointing. Bottom: Simulated map for ECLAIRs exposure in galactic coordinates for the B1 law pointing. ECLAIRs will spend most of its time observing the regions near the galactic poles and avoid the galactic plane. Image credit: *SVOM* collaboration.

3.1.3.2 Transient alert system

The fast (order of minutes) afterglow follow-up with *SVOM* and complementary ground facilities requires a global alert system that can rapidly (order of seconds) transmit the essential information about the onboard GRB detection and localisation to the ground telescopes. The system consists of a VHF (Very High Frequency) emitter on the spacecraft and a network of VHF antennas spread across the globe below the path of the satellite (Figure 3.9). With the VHF system, *SVOM* will be able to transmit the GRB localisation information to the ground segment telescopes within 30 s (for about $2/3^{\text{rd}}$ of the GRBs) after the onboard detection.

3.1.3.3 Observation programs

SVOM observations are divided into three categories according to the type of the sources observed.

Core Program

The Core Program of *SVOM* includes detection and follow-up observations of GRBs and other high energy transients. A typical observation in this category starts with

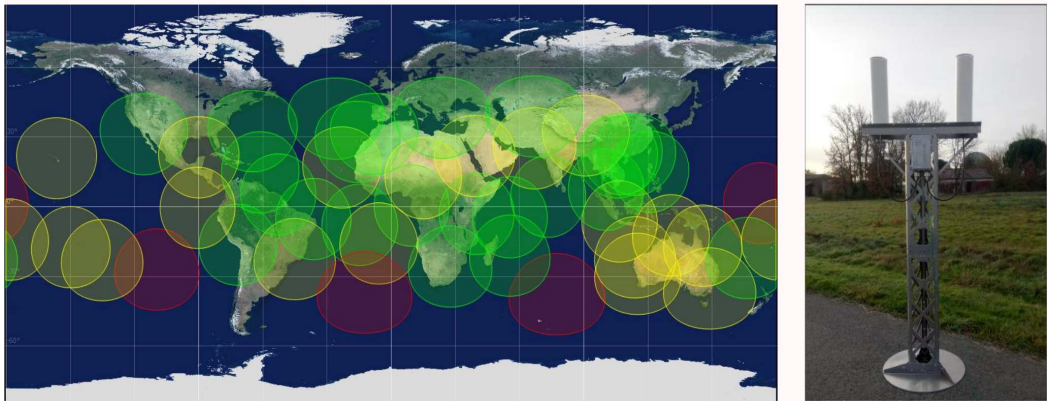


Figure 3.9 – Left: Location of ground VHF receiver network. At least one antenna will be visible to the satellite most of the time. Right: A prototype of the VHF receiver antenna. Credit: *SVOM* collaboration.

the detection of a GRB (or another transient) by ECLAIRs and GRM and finishes when VT and MXT end the follow-up observations which last about 14 orbits (∼22 hrs of observation time and ∼ 10 hrs of on-source time).

TOO Program

The TOO Program of *SVOM* includes unplanned observations of transient and variable sources. These observations will include follow-up of transients in response to the alerts received from other missions/ ground-based facilities (e.g. gravitational wave events, neutrino events etc.) or transients detected by the offline analysis of ECLAIRs data (see chapter 4). The typical duration of a TOO observation will be one orbit (∼ 96 min of total observation time out of which ∼45 min are on-source) but can be as long as 14 orbits (∼22 hrs of total observation time out of which ∼ 10 hrs are on-source) in case of an exceptional TOO events (e.g. gravitational wave alerts). The TOO observations are programmed from the ground with minimum delay.

General Program

The General Program (GP) of *SVOM* includes preplanned observations. These will be scheduled from the proposals submitted to the yearly announcement of opportunity call. The minimum duration of a GP observation will be one orbit (∼45 min of on-source time).

3.1.4 SVOM: Scientific objectives and prospects

SVOM aims to capture the complete picture of GRB (from prompt emission phase to late afterglow phase) and it will also play an important role to study other transients and known high energy sources. *SVOM* focuses on achieving the following objectives (see also Wei et al., 2016),

Detect all known types of GRBs: *SVOM* will detect all known types of GRBs, i.e. not only the classical short/hard and long/soft GRBs but also the X-ray rich GRBs and X-ray flashes (section 2.1.1). The 4 keV threshold of ECLAIRs also ensures that *SVOM* will be able to detect a larger fraction (compared to the current sample) of highly redshifted ($z > 5$) GRBs and soft/nearby GRBs.

Optical to the sub-MeV study of the prompt emission: *SVOM* will be able to study the temporal and spectral aspects of the GRB prompt emission in X-rays and gamma-rays owing to its broadband coverage at high energies (4 keV to 5 MeV) and good performance of ECLAIRs and GRM (e.g. good spectral resolution, good sensitivity to peak energy etc., see 3.2.3). For some GRBs (about 12%), *SVOM* will be able to study their prompt optical emission owing to the partial overlap (about 70%) of GWAC and ECLAIRs FoV (total FoVs are $\sim$ 5400 and $\sim$ 8000 square degrees respectively). These prompt optical emission observations will be crucial to understand the jet composition and to put constraints on the radiation mechanisms of the prompt phase. Until today, prompt optical emission has been detected only for few GRBs and *SVOM* will be crucial in increasing this sample.

Optical to X-ray study of the afterglow emission: *SVOM* will study the early and late afterglow from optical to X-rays. Due to its fast detection and alert capabilities, it will be possible to begin optical observations of the afterglow as early as 30 seconds after the ECLAIRs trigger (COLIBRÍ has a fast response time of $\sim$ 20 s, see section 3.1.2.2). The fast follow-up capabilities (order of seconds to minutes) will increase the chance of detecting afterglow of short GRBs. These studies will be important to understand the nature of the progenitors. In X-rays, MXT will study the temporal evolution of the afterglow and will be able to carry out detailed spectroscopic studies.

Provide redshift measurements for GRBs: One of the goals of *SVOM* is to obtain redshift measurement for most GRBs. VT is designed to provide redshift indicators for high redshift GRBs ($z > 6$) while COLIBRÍ can measure the photometric redshift with 10% accuracy (Corre, 2016). COLIBRÍ can provide sub-arcsec localisation within 5 minutes after it points at the afterglow; an order of magnitude improvement over current instruments. A complete sample of GRBs with redshift measurements will be useful for cosmological studies, to study the local environments of the galaxy and to put constraints on star formation rates throughout the history of the universe (e.g. see section 2.1.3.2 and 2.1.3.3).

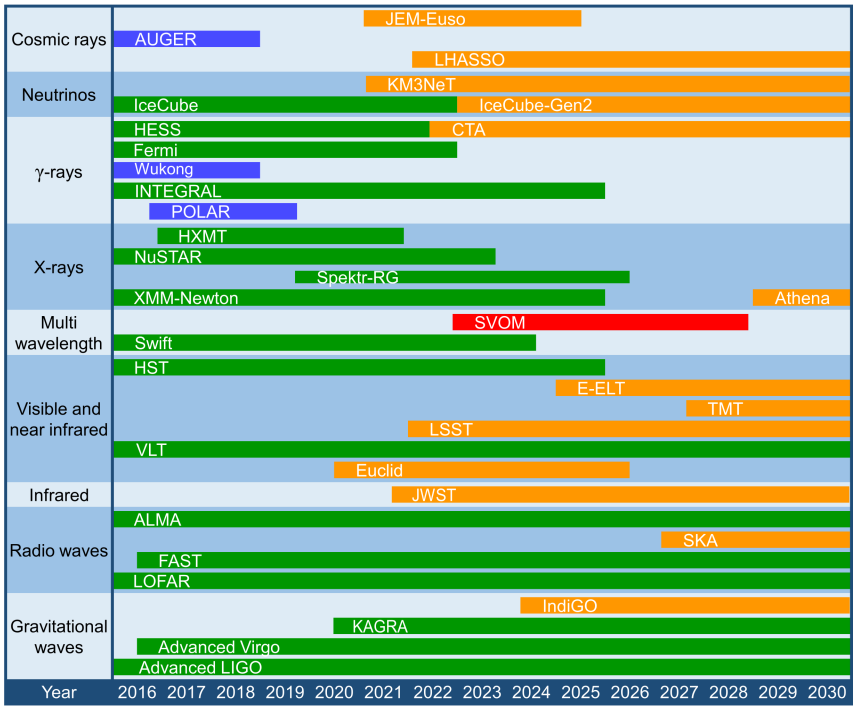


Figure 3.10 – Panorama of space missions/ground facilities in *SVOM* era. *SVOM* observations in synergy with these facilities will be crucial to understand many transient phenomena. Figure credit: *SVOM* collaboration.

Transient astronomy: *SVOM* will be operational in an era when many advanced facilities (see figure 3.10) will be observing the sky using electromagnetic radiation (e.g LSST, SKA), the gravitational wave radiation (e.g. LIGO/VIRGO) as well as the high-energy particles (e.g. IceCube and KM3NeT for neutrinos, JEM-Euso for cosmic-rays). With these facilities operational in the coming decade, the scientific community expects to detect many multi-wavelength and multi-messenger transients such as TDEs, FRBs, SGRs, electromagnetic counterparts of gravitational waves, neutrino events, orphan afterglows (i.e. GRBs viewed off-axis). *SVOM* will play a crucial role in detecting or following up these transients. Lastly, transients searches have always surprised astronomers and maybe *SVOM* will be able to do so by detecting previously unknown transient phenomena.

Observatory science: *SVOM* will be able to play an important role to measure the variability of known sources (e.g. AGNs, X-ray binaries, Cataclysmic Variables, ULXs) by monitoring them from optical to high energy X-rays over long durations. These studies will be important to understand the accretion processes and jet activity in these objects (see Goldwurm, 2017).

3.2 The gamma-ray imager ECLAIRs

The section is dedicated to the description of ECLAIRs which is the focus of the thesis. In this section, I'll give some details about the instrument, how it images the sky (coded mask imaging technique), its scientific performance and prospects.

3.2.1 ECLAIRs instrument description

A graphical rendering of ECLAIRs and its major parts are shown in figure 3.11a and 3.11b respectively. The total mass of the instrument is 86.5 kg with a maximum power usage of 84 W. The development (both technical and scientific) responsibility of ECLAIRs completely lies with the French partner institutes (IRAP (PI), CEA/Saclay, IAP, APC and CNES) of *SVOM*. In this subsection, I'll cover important characteristics of the instrument. A summary of these characteristics is given in the table 3.1.

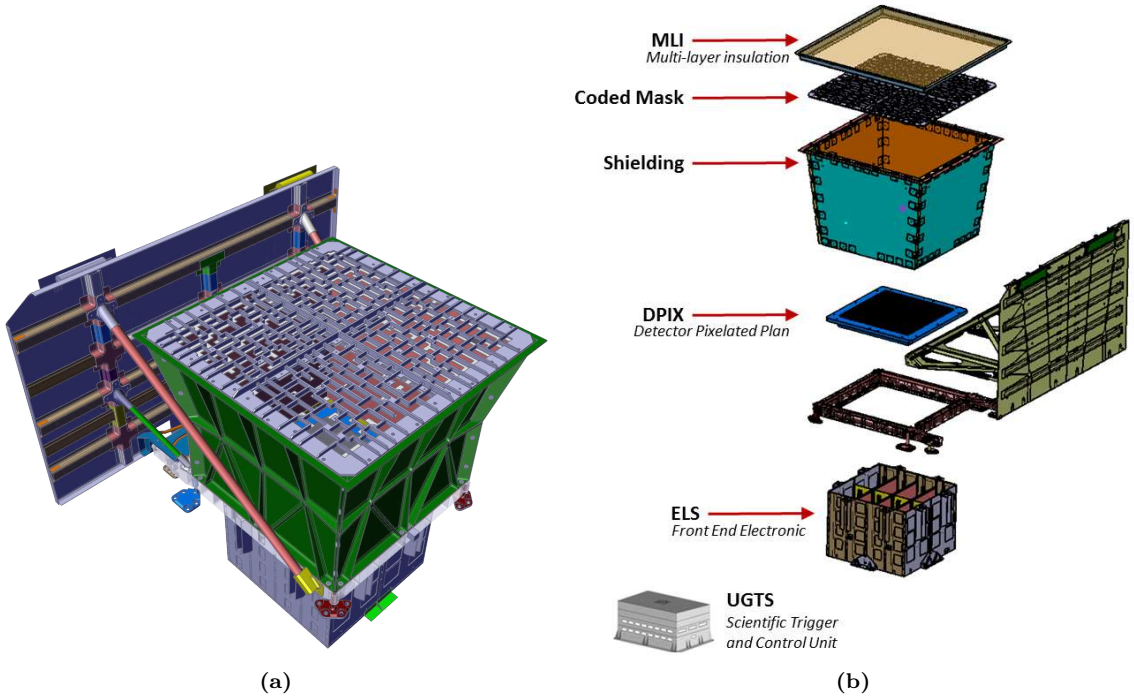


Figure 3.11 – View of ECLAIRs: (a) A graphical rendering. (b) An exploded view showing major components. Image credit: ECLAIRs instrument team (IRAP, CNES).

The detector plane and electronics

ECLAIRs detector plane is made of Schottky CdTe (Cadmium Telluride) detectors which enable the low energy threshold of 4 keV (Lacombe et al., 2013; Nasser et al., 2014; Godet et al., 2014). The detector plane is made of 6400 pixels with each pixel having size $4 \times 4 \times 1$ mm^3. The pixels are arranged into a 80×80 square with total geometrical area of ~ 1000 cm^2. The detector plane has 200 detector modules called

Table 3.1 – Important characteristics of ECLAIRs

Energy range	4 - 150 keV
Detectors	Schottky CdTe
Detector area	1000 cm^2
Effective area: at 6 keV	$\geq$ 200 cm^2
in 10 - 70 keV	$\geq$ 350 cm^2
Energy resolution at 60 keV	1.1 keV
Time resolution	20 μs
Field of View (FoV): Total	89° × 89°
Fully coded	23° × 23°
Source localisation error	$\sim$ 13 arcmin

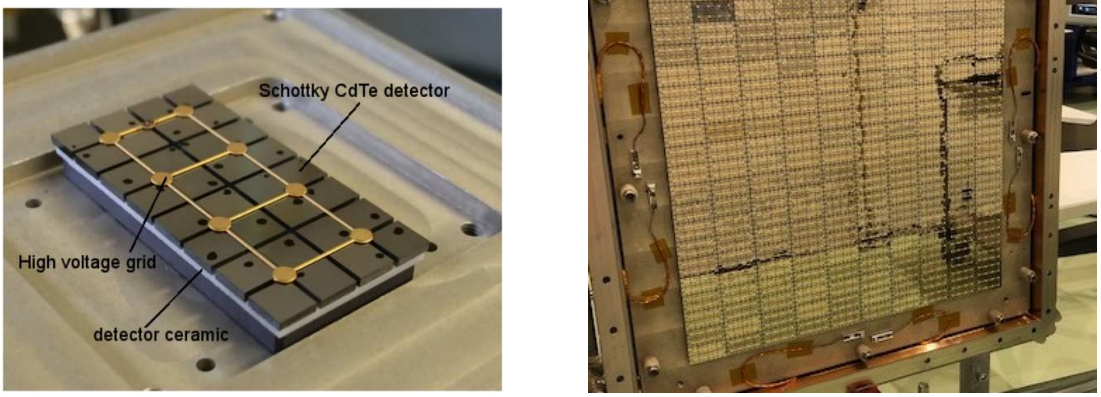
XRDPIX divided into 8 independent sectors (25 modules each). A single XRDPIX module has 32 pixels in 8×4 arrangement (Figure 3.12a). The ECLAIRs detectors offer an effective area of > 200 cm^2 at 6 keV and > 350 cm^2 in 10 - 70 keV range. The energy resolution of detectors is $\sim$1.1 keV at 60 keV (see table 3.1). The detectors will be operated below -18°C (passive cooling via radiator plate).

Each sector of the detector plane is read by an independent electronics chain (ELS) in a photon-counting mode registering time-tagged events with 20 μs resolution. The ELS will also classify events into single events or multiple events based on the number of pixels hits within the readout time. The entire electronics chain is designed to support (with a dead time of less than 5%) an event rate (on entire detector plane) of 10^5 events s^{-1} expected from the brightest GRB (e.g. GRB080319B [7]).

The coded mask and shielding

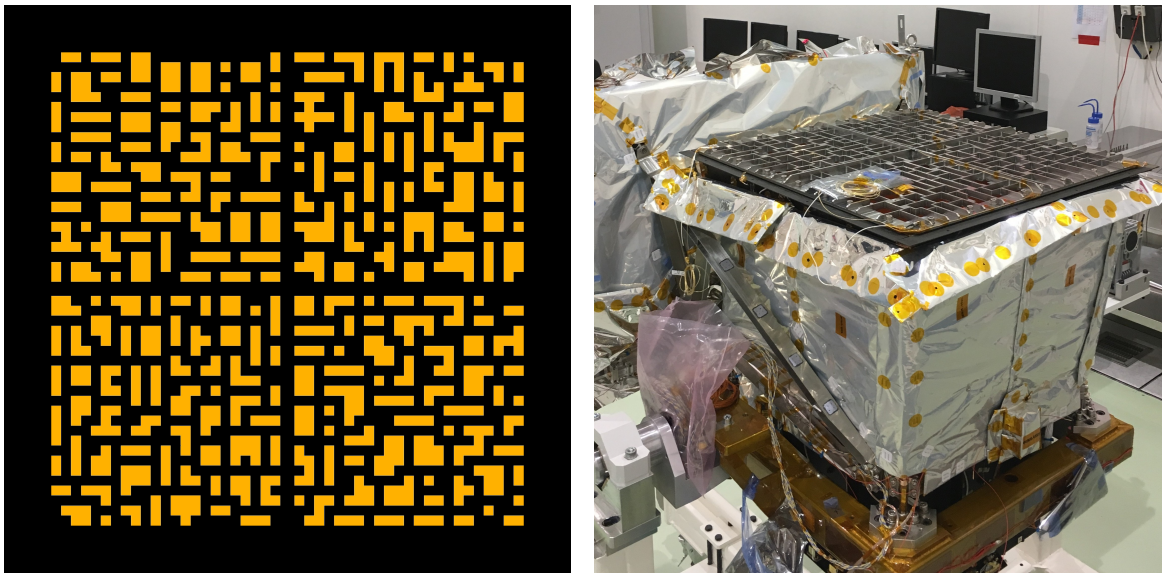
The coded mask of ECLAIRs is made of a sheet of Tantalum (0.6 mm) sandwiched between two layers of Titanium (Ti-Ta-Ti sandwich, Guillemot, 2020). The mask has a quasi-random pattern (see figure 3.13a) which can support itself. The open fraction of the mask is about 40%. Other geometrical details of the mask which are important for imaging are discussed in the next section. The detector - mask geometry gives a FoV of $89° \times 89°$ (or $\sim$ 2 sr) with a fully coded FoV of about $23° \times 23°$ (see section 3.2.2 for details). The localisation accuracy of ECLAIRs is 13 arcmin (see Gros, 2014). To reduce the optical and UV loading on the detectors, a thin multi-layer thermal coating insulation (MLI) will cover the mask. The mask-detector assembly is shielded by a structural support made of Lead, Copper and Aluminium layers.

7. https://swift.gsfc.nasa.gov/archive/grb_table/fullview/080319B/



(a) (b)

Figure 3.12 – View of ECLAIRs detectors: (a) A single XRDPIX module with 32 pixels in a 8×4 arrangement. (b) Complete detector plane with all 200 XRDPIX flight grade modules mounted on a cold plate for calibration purpose. Image credit: ECLAIRs instrument team (IRAP, CNES).



(a) (b)

Figure 3.13 – A figure showing (a) ECLAIRs mask pattern and (b) the structural and thermal model (STM) just before qualification tests. Image credit: ECLAIRs instrument team (IRAP, CNES).

The UGTS

The important characteristic of ECLAIRs is its capability to search for GRBs and transients in near real-time. This task as well as the management of the instrument parameters (e.g. temperature control, housekeeping data management, operating mode etc.) will be done by the UGTS (Unité de Gestion et Traitement Scientific or Scientific Management and Processing Unit in English, Schanne et al., 2013). The trigger

software in the UGTS consists of a count-rate and an image trigger algorithm to search for a transient in the data stream coming from the instrument in real-time (see Chapter 4 or Antier (2016) for more details). The UGTS will be in charge of sending the slew request and the VHF alert. The recorded raw data will be stored in the mass memory of the satellite for downlink when the ground stations are visible to the satellite (typically 4 - 6 times per day).

3.2.2 ECLAIRs geometry and coded mask imaging

The Coded Mask Imaging (CMI) is an effective way to image the sky in hard X-rays/gamma-rays as it can provide a large FoV (in comparison with focusing optics) with a good source localisation capability (order of few arcmins). The CMI technique has been successfully used on many large missions such as *BeppoSAX*/WFM, *INTEGRAL*/SPI and IBIS and *Swift*/BAT. A CMI telescope requires two important components:

— A coded mask with opaque (in the energy range of interest) and transparent elements arranged in a pre-defined pattern.

— A position-sensitive detector with a spatial resolution comparable to the mask element size.

The mask is placed above the detector at a fixed height. Within the FoV, photons that are coming from a given direction are either absorbed by the mask or pass through it, encoding the direction information with a unique pattern onto the detector plane (Figure 3.14 - (a)). The direction of the source can be recovered by decoding the detected pattern. A shielding material is placed around the mask - detector system (e.g. Lead-Copper-Aluminium shield in case of ECLAIRs) to block the photons which do not pass through the mask. The discussion here is focused on describing the geometry of ECLAIRs and imaging of the point sources by ECLAIRs, for a general review of coded mask systems one can refer to Gunson & Polychronopulos (1976); Caroli et al. (1987); In 't Zand (1992); Braga (2019).

Figure 3.14 - (b) shows a vertical cross-section of ECLAIRs. The mask dimensions, $L_m \times L_m$, are 54×54 cm^2 and the detector dimensions, $L_d \times L_d$, are 36×36 cm^2. The mask - detector vertical separation, h, is 45.84 cm. The mask has 46×46 elements and the size of each mask element, m, is 11.384 mm. The optical axis (or pointing axis) of the system ($\vec{z}$) goes vertically up from the centre of the detector (O) and intercepts at Z. With this geometry, the angular distance to any direction ($\vec{p}$) from the optical axis is then:

$$\theta = \tan^{-1} \frac{l}{h} \tag{3.1}$$

The entire FoV of ECLAIRs is divided into two regions. First is the Fully Coded

Field of View (FCFoV) where, for a given direction, all photons that pass through the mask reach the detector and cast a shadow on the entire detector. Second is the Partially Coded Field of View (PCFoV) where, for a given direction, only a fraction of photons that pass through the mask reach the detector and cast a shadow only on a part of the detector. For this geometry and on the either side of the pointing

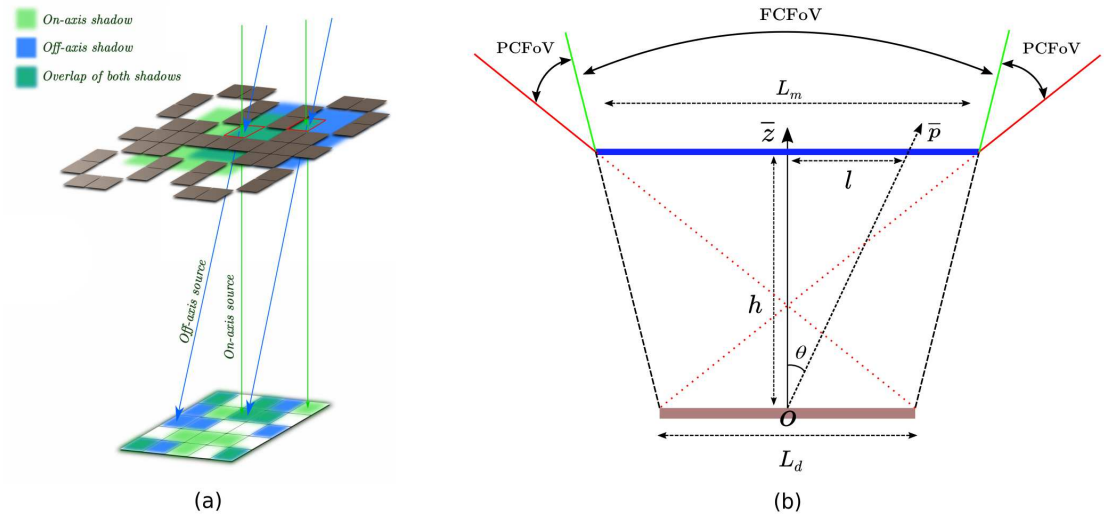


Figure 3.14 – Coded mask imaging principle and ECLAIRs geometry: (a) A schematic diagram showing the principle of coded mask imaging. The image shows the shadow pattern (or shadowgram) formed by the mask onto the detector for an on-axis as well as an off-axis source. Image credit: Braga (2019). (b) A vertical cross-section showing the geometry of ECLAIRs. The total FoV is marked by the Red lines. The directions within the Green line constitute the fully coded FoV (FCFoV) while the directions between the Green line and the end of the FoV (Red line) constitute the partially coded FoV (PCFoV).

axis, all directions with $l \leq (L_m - L_d)/2$ are within FCFOV while directions with $(L_m - L_d)/2 < l \leq (L_m + L_d)/2$ are within PCFoV.

An important step in the CMI system is to pixelate the FoV in order to localise the sources. The number of sky pixels depends upon the spatial resolution of the detector ($d = 4.5$ mm for ECLAIRs including the dead zones between two pixels) and the linear lengths of the mask and the detector. The total number of sky pixels are then $((L_m + L_d)/d - 1) \times ((L_m + L_d)/d - 1)$ which is equal to 199×199 pixels for ECLAIRs.

The process of encoding the incoming photon flux is represented mathematically as a convolution of the mask pattern with a source in the FoV. If we represent a mask pattern with an array M_{ij} which has values 1 or 0 corresponding to a transparent and an opaque element respectively and the detector counts are represented by the matrix D_{kl} then the coding process can be represented as:

$$D = S * M + B \tag{3.2}$$

where $*$ is the convolution operator. This can be written more explicitly as:

$$D_{kl} = \sum_{ij} S_{ij} M_{i+k,j+l} + B_{kl} \tag{3.3}$$

where, S_{ij} are the source intensities in the each sky pixel (or "sky image") and the B_{kl} represent the background counts which do not pass through the mask. The sky image can be reconstructed from the detector counts (or "shadowgram") by the inverse process called deconvolution (see also Gros, 2014; Goldwurm et al., 2001). For ECLAIRs, the linear size of mask element (m) is not equal to the spatial resolution of the detector (d). Therefore, before deconvolution, the mask must be rebinned with a ratio m/d, which is equal to 2.6 for ECLAIRs. The rebinned mask [8] has 120×120 elements with fractional values ranging between 0 to 1. Using this rebinned mask (M_r), we can define decoding arrays G^+ and G^- such that:

$$G = 2M_r - 1 \tag{3.4}$$

$$G^+ = \begin{cases} G & M_r > 0.5 \\ 0 & otherwise \end{cases} \qquad G^- = \begin{cases} G & M_r < 0.5 \\ 0 & otherwise \end{cases} \tag{3.5}$$

The decoding arrays are then used to reconstruct the source intensities (S_{ij} or "sky counts image") as follows:

$$S_{ij} = \frac{\sum_{kl} G^+_{i+k,j+l} W_{kl} D^{BC}_{kl} - \mathcal{B}_{ij} \sum_{kl} G^-_{i+k,j+l} W_{kl} D^{BC}_{kl}}{\mathcal{A}_{ij}} \tag{3.6}$$

where D^{BC} is the detector counts array corrected for any non flat background (by subtracting background count model), the W is the weights array (same size as D^{BC}) giving information about detector pixels (efficiency, dead pixel etc.). The $\mathcal{A}$ and $\mathcal{B}$ are the flux normalisation and balance array respectively. The flux normalisation array ensures that the final image has correct flux (cnt/pix) and the balance array ensures that the sky counts image is flat in absence of any source in the FoV (see Gros, 2014, for more details). The variance of each sky pixel is then given by the variance image as,

$$\sigma^2_{S_{ij}} = \frac{\sum_{kl} (G^+_{i+k,j+l} W_{kl})^2 Var(D^{BC}_{kl}) + \mathcal{B}^2_{ij} \sum_{kl} (G^-_{i+k,j+l} W_{kl})^2 Var(D^{BC}_{kl})}{\mathcal{A}^2_{ij}} \tag{3.7}$$

From the sky counts and variance image, we can define a sky signal to noise ratio image ($\mathrm{SNR_{img}}$) which determines the significance of each sky pixel relative to the

8. Rebinned mask can also be viewed as the shadow cast by a unit source (1 count/pixel) onto a virtual detector which has same size as the mask

background variance. The sky SNR_{img} is defined as,

$$SNR_{img,ij} = \frac{S_{ij}}{\sigma_{S_{ij}}} \tag{3.8}$$

The detection and localisation of transients with ECLAIRs is done by looking for a new source the SNR image (see Chapter 4 for details).

3.2.3 ECLAIRs performance and science prospects

The primary objective of ECLAIRs is to detect and localise all known types of GRBs and it is expected to detect about 70 GRBs/yr. It will be sensitive down to 2.5×10^{-8} erg cm^{-2} s^{-1} for 1 s long GRBs (in the 5 - 50 keV energy range and for a 6.5 sigma detection, see Atteia et al., 2016). Owing to its good effective area at low energies, ECLAIRs will have good sensitivity to GRBs which have peak energy (E_p, see section 2.1.2.1) below 30 keV.

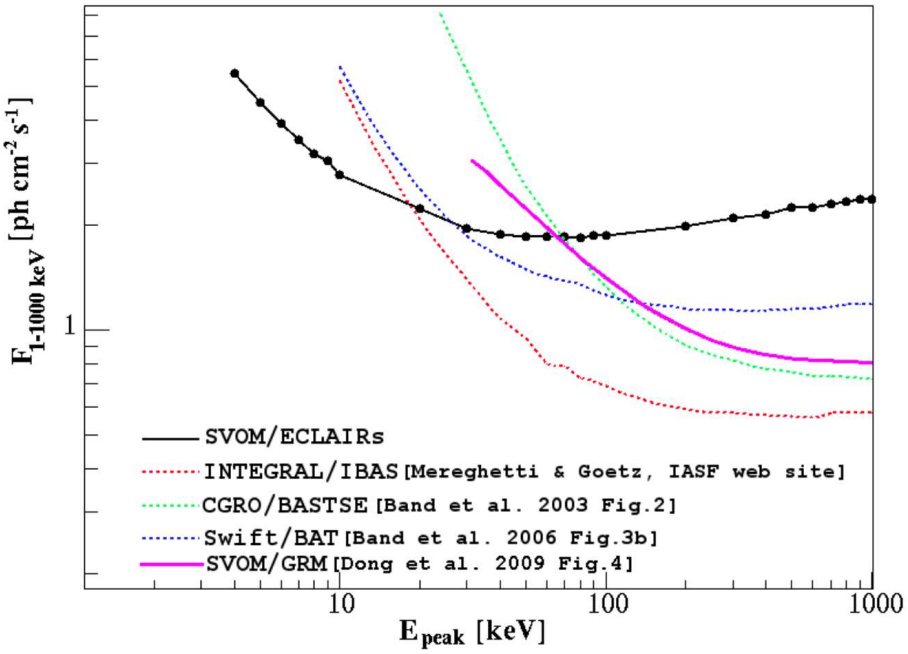


Figure 3.15 – ECLAIRs sensitivity as function of GRB peak energy for 1 s long GRBs. The sensitivity is represented by the upper limit of the flux (ph cm^{-2} s^{-1}) in the energy range 1 - 1000 keV. ECLAIRs is more sensitive than *Swift*/BAT to the GRBs with peak energy below 30 keV. Image credit: Godet et al. (2014)

ECLAIRs shares the lower end of its energy range with MXT and the higher end with GRM. The overlap at high energies with GRM will enable the study of the prompt emission in a broad (4 keV - 5 MeV) energy range. These broadband spectroscopic studies will be important to constrain the spectral parameters giving direct insights into the prompt emission mechanisms (e.g. see Bernardini et al., 2017), physics of

GRB progenitors and jet acceleration. It will also enable peak energy measurements for a large number of GRBs which, together with the redshift measurements (see section 3.1.4) are important for cosmological studies (see 2.1.3.2). At the lower end, the energy overlap with MXT will be important to study the transition from prompt emission to afterglow phase (see also section 3.1.4).

Along with GRBs, ECLAIRs will be important for detecting other high energy transients (e.g. TDEs, SGRs, TGFs, see section 2.2 and 3.1.4). In light of recent discoveries, a particularly interesting science case for ECLAIRs can be to search for coincident detection of SGRs and FRBs (see Kirsten et al., 2020; Li et al., 2020a; Mereghetti et al., 2020). Furthermore, thanks to its wide FoV and good sensitivity in soft and hard X-rays, ECLAIRs will be an important instrument for the *SVOM* objective (see section 3.1.4) of surveying and monitoring the known sources.

Chapter 4

Simulations of the background and raw data for ECLAIRs

Data which represent near-real scenario is important to design, develop and test the trigger algorithms (both onboard and offline triggers, see section 5.1 and section 5.3 for their description respectively) as well as to assess other instrument performance parameters (e.g. sensitivity of the instrument to a certain type of sources, testing data reduction process). The data must include the expected background, its variation, the observed X-ray sources and events from GRBs or transients. As part of the thesis, I have developed the tools to simulate data for ECLAIRs which are described in this chapter. I begin with a brief description of ECLAIRs background and its estimation using a method (Bouchet & Atteia, 2016) developed and improved as part of this thesis. I then describe simulations of GRBs and X-ray sources and the process of generating ECLAIRs data which represent near-real observing scenarios.

4.1 The astrophysical background in ECLAIRs

Space-based X-ray/gamma-ray telescopes experience a background of astrophysical origin. The background and its level depend on the energy range of the instrument, its geometry and the orbit of the satellite. Usually, non-focusing wide-field instruments such as *SVOM*/ECLAIRs experience a high background level, which in turn can affect the sensitivity of the instrument.

The background in ECLAIRs is due to the high energy photons and the energetic particles (charged or neutral particles e.g. protons, neutrons, ions etc). Given ECLAIRs energy range (4 - 150 keV), the diffuse cosmic X-ray background (CXB) photons (Giacconi et al., 1962) will be the dominant source of background. The origin of these photons is thought to be due to the distant unresolved Active Galactic Nuclei (AGNs). Due to its cosmological origin, the observed photon distribution of CXB is

isotropic in the sky. Furthermore, because *SVOM* will have a Low Earth Orbit (LEO, see section 3.1.3.1), the high energy photon emission from the atmosphere of the Earth will also contribute to the observed background. This emission mainly consists of two components, first is the reflection (e.g by Compton scattering) of the CXB photons from the atmosphere Churazov et al. (2008) and the second is the hard X-ray albedo produced by the interaction of cosmic-rays with the atmosphere (Sazonov et al., 2007). In the case of the CXB reflection (hereafter Reflection), the atmosphere (as viewed from the satellite) has uniform brightness and the emission intensity doesn't depend on the position of the satellite (geographical latitude, longitude) with respect to the Earth. While for the hard X-ray albedo (hereafter Albedo), the atmosphere has a non-uniform brightness and that the emission depends on the latitude and longitude of the satellite (Sazonov et al., 2007). Therefore, the observed intensity of albedo photons varies as the satellite position changes with respect to the Earth during the orbit.

The cosmic-rays will be the primary source of the background due to the energetic particles. The interaction of the cosmic-rays with the atmosphere of the Earth produces secondary particles (particle albedo) which will be detected in ECLAIRs as well. Along with the cosmic-rays, as *SVOM* will pass through the South Atlantic Anomaly (SAA) during some orbits, the very high flux (order of $10^3 - 10^4$ particles $\cdot$ cm$^{-2} \cdot$ s^{-1}) of trapped charged particles will generate high count-rates in ECLAIRs during the SAA ingress and egress phase [1].

The interaction of cosmic-rays or the SAA charged particles induce radioactivity in the satellite material. The radioactive decay of the elements will cause a delayed particle or photon emission which also contribute to the observed background (called activation background) in ECLAIRs (Bringer et al., 2013). The contribution of such an emission depends on the final impurities in the satellite elements which are not well known before the launch of the satellite.

The Earth will pass through the ECLAIRs FoV every orbit due to the *SVOM* pointing strategy (see section 3.1.3.1). This will result in a modulation of the observed background level (i.e. variations in the detected count rate) based on the position of the Earth with respect to the FoV. The variation of count rate and a partial presence of Earth in the FoV affect the sensitivity of both count and image trigger. This can lead to false detections or no detection of fainter transients. To study these effects, it is important to accurately estimate the background and its variation (dynamic modelling) as a function of the *SVOM* orbit.

1. ECLAIRs will cease operations when it is deep inside the SAA region to protect the detectors, see section 3.1.3.1

4.2 ECLAIRs background simulations

4.2.1 The GEANT4 simulations

The most accurate way to simulate background in high energy instruments is to use Monte-Carlo simulations. This can be done using the GEANT4 (Agostinelli et al., 2003) toolkit which simulates particle matter interactions accurately. GEANT4 was originally designed to simulate particle interactions in high energy physics, but now it is extensively used in other scientific disciplines such as medical, space and radiation instrumentation. A GEANT4 simulation of an instrument or detector has three major parts,

1. Definition of the mass model - The mass model of an instrument/ detector is the complete geometrical and elemental model. The accuracy of GEANT4 simulations significantly depends on how well the mass model is built compared to the real instrument/ detector.
2. Definition of physical processes - GEANT4 can simulate various types of interactions between particles such as photoelectric absorption, Compton scattering, bremsstrahlung etc.
3. Definition of input particles - GEANT4 simulations depend on the properties of the input particle, such as its type, its spectral and spatial (both angular and position) distribution, the definition of the source geometry etc.

The estimation of ECLAIRs background using GEANT4 is time consuming due to the computations involved to simulate the underlying physical processes. Additionally, the varying background of ECLAIRs necessitates dynamical simulations (continuous evolution of orbital parameters with time) and taking into account all possible orbital scenarios add additional computational burden. The method developed as part of this thesis, called the Particle Interaction Recycling Approach (or PIRA) extends the use of GEANT4 simulations to achieve the necessary dynamical simulations.

4.2.2 Background estimation using PIRA

The PIRA (Mate et al., 2019) was developed with the aim of providing another way that is as accurate as GEANT4 simulations but reduces the computation time and enables the calculation of the background for any possible geometrical configuration (e.g. change of Earth position with respect to the instrument pointing direction). The method utilises a large, precomputed database that stores the particles and their interactions with the detector and reuses these particles to simulate the background. This database is generated using GEANT4 simulations. The reuse of particles is based on selection algorithms that choose the particle (and its energy deposition

in the detector) based on its initial direction or energy. This way, a new GEANT simulation run can be avoided even if the orbital configuration is changed and a large amount of computation time (factor of $10^3 - 10^4$ in the case of ECLAIRs) can be saved. This approach also enables the computation of dynamic background.

In the following sections, I will describe the use of PIRA in the context of ECLAIRs background simulations with CXB, Reflection, Albedo and charged particles in the SAA as an example. A more generic and detailed description of PIRA can be found in Mate et al. (2019). Hereafter, the word primary will be used when referring to particles or photons unless mentioned specifically.

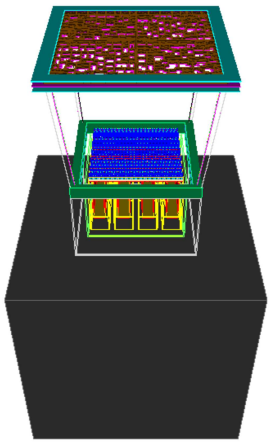


Figure 4.1 – Graphical rendering of the ECLAIRs mass model. The mask (Pink), the detector plane (Blue) and the boxes containing the electronic readout (Yellow) are shown in the image. The shield surface is not rendered to make internal components visible. In this version, the satellite platform (Dark Grey) is approximated by a cube.

4.2.2.1 Generation of database of primaries

The database of primaries depends on the mass model of the instrument. The ECLAIRs mass model (see Sizun, 2011, 2015a, for more details) used in our simulations is shown in figure 4.1. In the current version, the other SVOM instruments are not modelled and the satellite platform is approximated with a simple Aluminium cube.

We shine a large number of input primaries onto the mass model to generate the database of primaries. An important aspect to consider while generating the database is the spatial and spectral distribution of the input primaries. For simplicity, we take the spatial distribution of the input primaries to be isotropic. Furthermore, to increase the efficiency of the simulation, we use angle biasing (see Campana et al. (2013) for more details) to restrict the emission angle of incoming primaries. The input primaries are drawn from a spherical surface (hereafter "source sphere") of radius R_p which encloses the entire ECLAIRs mass model. The centre of this sphere is aligned with

the centre of the detector plane. Furthermore, we align the Z-axis of the GEANT4 frame with the ECLAIRs pointing axis. The GEANT4 XY plane is parallel to the detector plane. With this configuration, the definition of coordinates in the GEANT4 and in the ECLAIRs frame is the same.

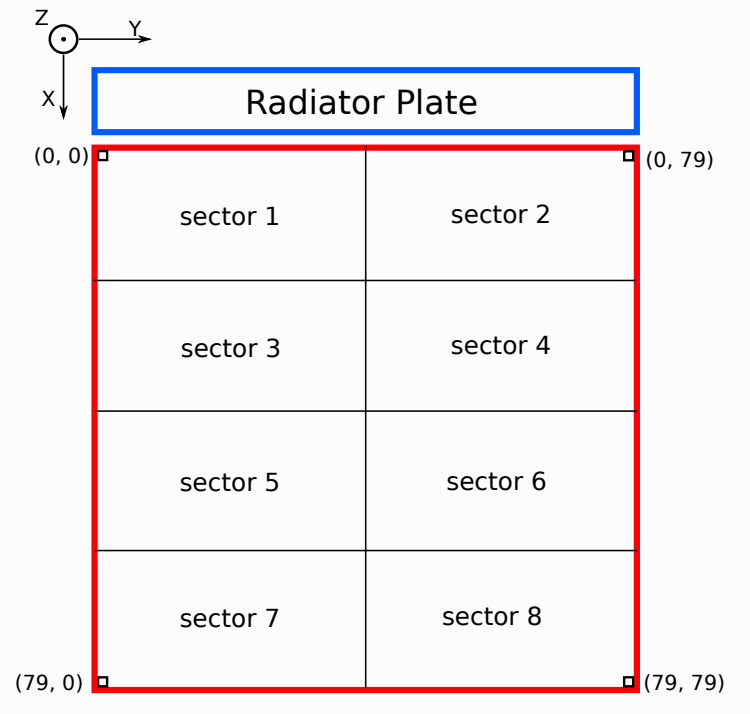


Figure 4.2 – Pixel orientation and indexing of the ECLAIRs detector plane inside GEANT4. The edge pixels are marked with their corresponding indices for reference. The co-ordinate frame for pixel indexing is shown in the top left.

The selection of primaries from the database is mainly based on their incoming direction and/or energy. Hence, for the database, we store the incoming direction and energy of the input primaries. Further, we also store the energy deposition by the primary (or its secondaries) and its location on the ECLAIRs detector plane (in pixel value, see figure 4.2 for pixel orientation and indexing). A primary (or its secondaries) can deposit energy in more than one pixel (multiple events); in such a case, we store all the energy deposits and their locations. To save space, we only store the information about a primary if it (or its secondaries) deposits energy in the detector. The information about the direction and the energy of the primary is stored in a "source" file (e.g. see table 4.1, Mate et al., 2019) and the energy deposits and their locations are stored in the "events" file (e.g see table 4.2, Mate et al., 2019). The 'EventID' in both files identifies the primary and it is used to match the primary and its energy deposit(s).

We generate a separate database for each background component (i.e. a separate database for the CXB, Reflection, Albedo, SAA electrons and SAA protons respectively) and the input spectral distribution is chosen depending upon the component.

Table 4.1 – Example of a source file in the case of ECLAIRs. The coordinates are expressed in the instrument frame. In the case of ECLAIRs, we have $x_p^2 + y_p^2 + z_p^2 = R_p^2$, where R_p (here= 500 cm) is the source sphere radius (see section 4.2.2.2 and figure 4.5 for the definition of the source sphere).

EventID	Input Energy (keV)	Coordinates on the Source Sphere†			Primary Direction	
		x_p (cm)	y_p (cm)	z_p (cm)	θ_p (deg)	ϕ_p (deg)
11209	10.169	269.150	-33.668	420.029	34.089	-8.925
1675257	27.621	62.495	-191.778	457.51	25.085	-72.204
...	...	...	...	...	...	...

† These parameters are not mandatory.

Table 4.2 – Example of an output event file in the case of ECLAIRs

EventID	Deposited Energy (keV)	Interaction Position (Pixels)	
		X	Y
11209	10.169	13	63
1675257*	17.213	33	66
1675257*	10.408	33	67
...	...	...	...

* In case of a multiple event, all the energy deposits and their pixel locations are stored.

For the CXB we use the spectral model given by Moretti et al. (2009) and Gruber et al. (1999) joined at 163 keV. The model is given by the following equation:

$$N(E) = \begin{cases} \dfrac{0.109}{\left(\frac{E}{29.0}\right)^{1.40} + \left(\frac{E}{29.0}\right)^{2.88}} & E < 163 \text{ keV} \quad (4.1a) \\ 0.0259 \left(\dfrac{E}{60}\right)^{-6.5} & (4.1b) \\ +0.504 \left(\dfrac{E}{60}\right)^{-2.58} & E \geq 163 \text{ keV} \\ +0.0288 \left(\dfrac{E}{60}\right)^{-2.05} & (4.1c) \end{cases}$$

The Reflection spectral model is obtained by multiplying the CXB spectrum with the reflection factor estimated by Churazov et al. (2008). The reflection factor is given as:

$$A(E) = \frac{1.22}{(\frac{E}{28.5})^{-2.54} + (\frac{E}{51.3})^{1.57} - 0.37} \times \frac{2.93 + (\frac{E}{3.08})^4}{1 + (\frac{E}{3.08})^4} \times \frac{0.123 + (\frac{E}{91.83})^{3.44}}{1 + (\frac{E}{91.83})^{3.44}} \quad (4.2)$$

For the Albedo, we use the spectral form suggested by Sazonov et al. (2007) to calculate the surface brightness of the atmosphere when observed from a spacecraft (valid for altitudes greater than 100 km). The Albedo spectrum (in units of

$\mathrm{ph \cdot cm^{-2} \cdot s^{-1} \cdot sr^{-1} \cdot keV^{-1}}$) is given as:

$$N(E) = \frac{C}{(\frac{E}{44})^{-5} + (\frac{E}{44})^{1.4}} \tag{4.3a}$$

$$C = \frac{3\mu(1+\mu)}{5\pi} \frac{1.47 \times 0.0178 \times [(\varphi/2.8)^{0.4} + (\varphi/2.8)^{1.5}]^{-1}}{\sqrt{1 + [\frac{R_{cut}}{1.3\varphi^{0.25}(1+2.5\varphi^{0.4})}]^2}} \tag{4.3b}$$

where C has units $\mathrm{ph \cdot cm^{-2} \cdot s^{-1} \cdot sr^{-1}}$ and $\mu = \cos\Theta$ with,

$$R_{cut}(\text{in GV}) = 59.4 \times \frac{\cos^4 \lambda_m}{[1 + (1 + \cos^3 \lambda_m \sin\Theta \sin\xi)^{1/2}]^2} \tag{4.3c}$$

Here, φ is the solar modulation potential (Usoskin et al., 2005), R_{cut} the geomagnetic cut-off rigidity, λ_m the magnetic latitude and Θ is the zenith angle (geographical frame) and ξ an azimuthal angle, with $\xi = \pi/2$ indicating the North - West direction. Following the simplification suggested by Sazonov et al. (2007), we assume $\xi = 0$. To generate the database of Albedo photons, the value of the constant 'C' is set to its maximum (0.025 $\mathrm{ph \cdot cm^{-2} \cdot s^{-1} \cdot sr^{-1} \cdot keV^{-1}}$). The spectral shape of these background component is shown in figure 4.3 (for details see, Zhao et al., 2012; Mate et al., 2019). For charged particles in the SAA, we use a spectral shape obtained by averaging the observed electron and proton spectra (see figure 4.4). Note that the choice to use the mean spectral form is arbitrary (we could have used another spectral form).

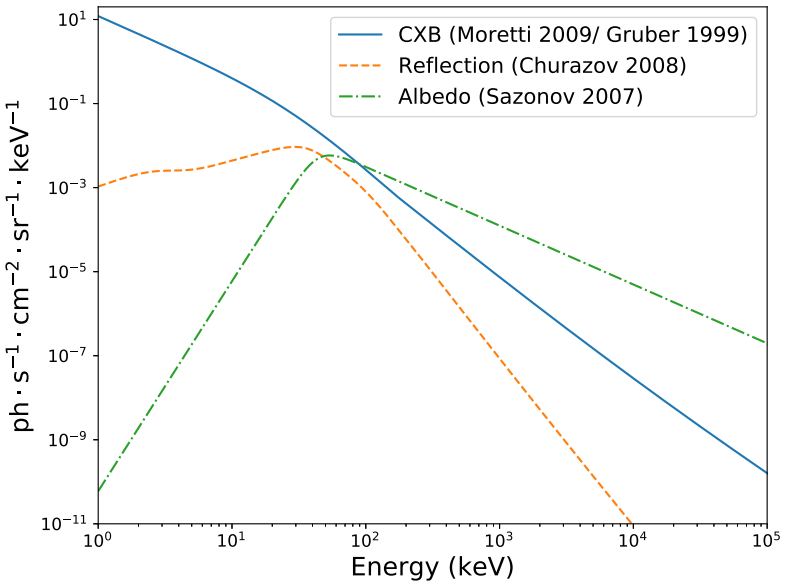


Figure 4.3 – The photon spectra of the CXB (Blue), Reflection (Orange) and Albedo (Green) emissions used in our simulations. The Albedo emission spectrum corresponds to the maximum emission observed at latitude of 30°N and longitude of 287°E.

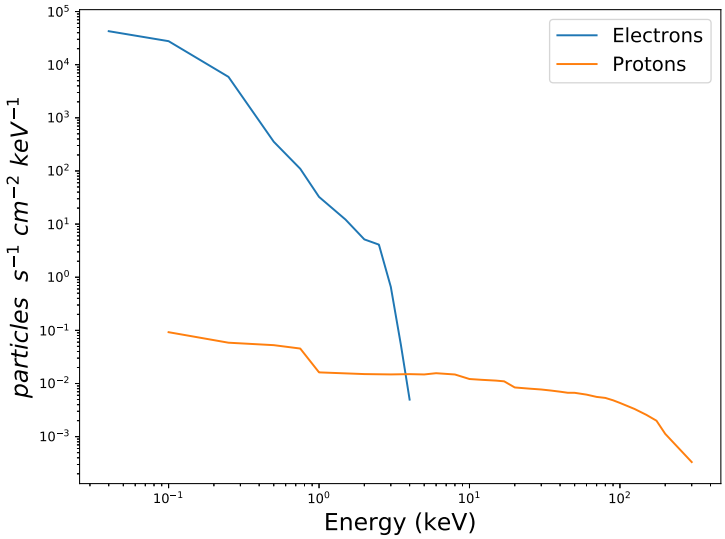


Figure 4.4 – Input spectral shape used to generate database of SAA electron and proton. These input spectra are the average of expected SAA electron and proton spectra over 400 orbits. The individual spectra are computed using the AE-8/AP-8 models [2]. Note that the choice of this spectral shape is arbitrary.

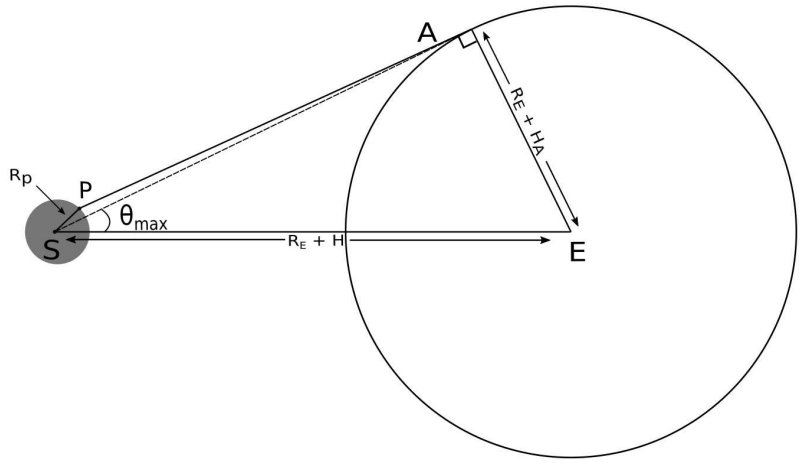


Figure 4.5 – Schematic representation of the relative position of the Earth and the satellite. **E** is the Earth centre, **S** is the satellite position, **A** is the point where the trajectory of incoming primary intersects with the atmosphere, R_E is the radius of the Earth ($\sim$ 6370 km), H_A is the height of the atmosphere ($\sim$ 100 km) and H is the satellite altitude ($\sim$ 625 km). The grey sphere centred around **S** depicts the source sphere with a radius R_p. The dashed line represents the case where the size of the source sphere is neglected compared to the satellite altitude i.e. $R_p << H$.

4.2.2.2 Selection algorithms

The crucial part of PIRA is the algorithms used to select the primaries from the database. We differentiate these algorithms into two categories, a) selection in the

2. https://ccmc.gsfc.nasa.gov/modelweb/models/trap.php

spatial domain and b) selection in the spectral domain. I explain these algorithms in the static case, i.e., when the satellite position (latitude, longitude) with respect to the Earth is fixed. This way, the explanation is easier and the results can be compared qualitatively with previous estimates of the background by Zhao et al. (2012).

Selection in the spatial domain

In the case of the selection in the spatial domain, the primaries are selected based on their spatial properties (mainly incoming direction). The selection in the spatial domain can be further separated into two cases. In the first case, the selection of primaries is based only on their incoming direction and in the second case, the selection is based on the directional selection combined with a modification of the original spatial distribution.

To understand the geometrical configuration, we assume the geometry shown in figure 4.5. In the figure, R_{Eff} is the effective Earth radius which is equal to the Earth radius (R_E) plus the height of the atmosphere (H_A). The satellite altitude is defined as H. The point **S** is the ECLAIRs (satellite) location and the point **E** is the centre of the Earth. The point **A** is the intersection of the primary's trajectory with the atmosphere of the Earth. All the coordinates are defined with respect to the ECLAIRs frame (same as the GEANT4 frame, see section 4.2.2.1).

Case 1: Directional selection only

This selection is applied to the CXB photons. The spatial distribution of CXB photons is isotropic in the sky. The presence of the Earth blocks CXB photons which would otherwise reach ECLAIRs. Hence a simple criterion based on the incoming direction of the photon can be applied to select (or reject) the photon from the database. To select the photon, consider the relation in Eqn. 4.4 which is true for each primary crossing the Earth.

$$\vec{SA} + \vec{AE} = \vec{SE} \tag{4.4}$$

where,

$$\vec{SA} = \vec{SP} + \vec{PA} = \begin{pmatrix} x_p \\ y_p \\ z_p \end{pmatrix} + \beta \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} \tag{4.5}$$

$$\vec{SE} = \begin{pmatrix} x_E \\ y_E \\ y_E \end{pmatrix} = (R_E + H) \begin{pmatrix} \sin\theta_E \cos\phi_E \\ \sin\theta_E \sin\phi_E \\ \cos\theta_E \end{pmatrix} \tag{4.6}$$

with,

$$\begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} \sin\theta_p \cos\phi_p \\ \sin\theta_p \sin\phi_p \\ \cos\theta_p \end{pmatrix} \tag{4.7}$$

and,

$$\|\vec{AE}\| = R_{Eff} = R_E + H_A \tag{4.8}$$

More explicitly, if we represent above equation in Cartesian coordinates (in ECLAIRs frame), it can be rewritten as a quadratic equation:

$$(x_p + \beta u_x - x_E)^2 + (y_p + \beta u_y - y_E)^2 + (z_p + \beta u_z - z_E)^2 = R_{Eff}^2 \tag{4.9}$$

where θ_E and ϕ_E are the zenith and azimuth angles corresponding to the Earth centre direction; θ_p and ϕ_p are the incident primary directions; $x_p,\ y_p,\ z_p$ are the primary coordinates on the source sphere (Table 4.1). The parameter β is the length of the vector $\vec{AP}$. For a given primary direction, if this equation (Eqn 4.9) has a positive solution in β, it implies that the primary has crossed the Earth and hence it is rejected.

If we assume that $R_p << H$, we can use another way of selecting the photons. Here, we first transform the directions ($\theta_p,\ \phi_p$) of all the photons in the database to a frame where the Z - axis is aligned with the direction $\vec{SE}$. In this new frame, all the photons having directions inside the cone defined by the polar angle θ_{max} (see figure 4.5) are rejected as their direction intercepts the Earth. This selection was used by Zhao et al. (2012) in their analysis.

It should be noted that the "directional selection" can also be applied to determine the cosmic-ray background as their flux is approximately (after neglecting the effect of the rigidity due to the Earth's magnetic field) isotropic in the sky.

Case 2: Directional selection with spatial distribution mapping

This method of selection is used to estimate the Albedo and Reflection components. In this case, along with a directional selection, the initial spatial distribution of primaries is modified. This is done due to two reasons, (a) the Earth's atmosphere, as seen from the satellite, is a convex surface while the source sphere is a concave surface; (b) the Albedo emission intensity (assuming it follows relation in Eqn. 4.3) is not isotropic within the visible part of the atmosphere. A mapping is used to associate the distribution of photons from the atmosphere with the distribution of photons on the source sphere. To modify the original distribution, a selection criterion based on

this mapping is used.

To relate the photon distribution on a convex surface onto a concave surface, we use the geometrical configuration defined in Figure 4.6. We define two reference frames, first is called the "Local" which is centred on the satellite (defined by the dashed circle) and the second is called the "MAP" which is centred on the Earth. We make an assumption that the satellite dimensions are small with respect to the satellite altitude ($R_p << H$) and it can be approximated as a point source. The frames "Local" and "MAP" are related as follows:

$$\cos\theta_M = \frac{(1 - l \times \cos\theta_L)}{k} \tag{4.10}$$

where,

$$l = \cos\theta_L - \sqrt{k^2 - \sin^2\theta_L} \tag{4.11}$$

$$k = \frac{R_E + H_A}{R_E + H} \tag{4.12}$$

and,

$$\phi_M = \phi_L - \pi + \psi \tag{4.13}$$

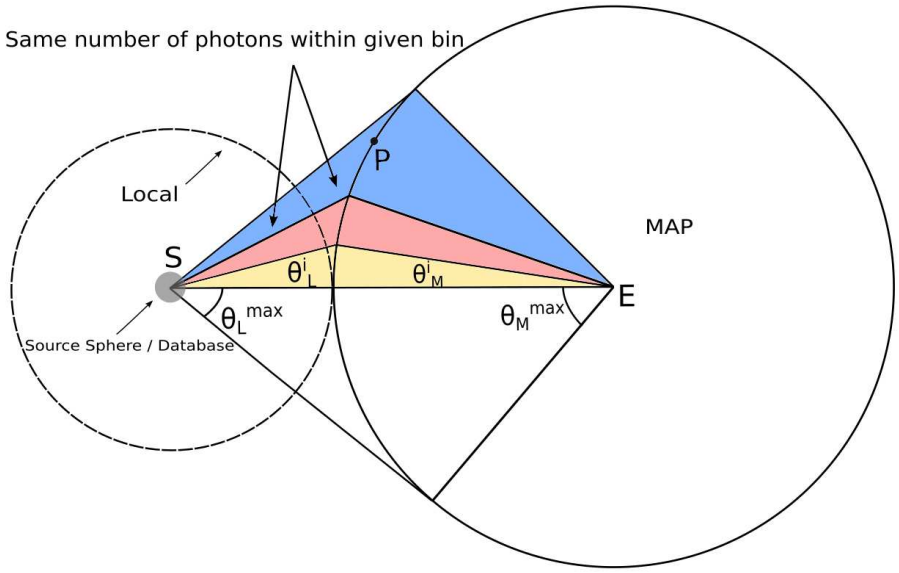


Figure 4.6 – Schematic diagram showing the geometrical configuration used to pixelate the visible part of the Earth's atmosphere to calculate the Albedo emission. **S** and **E** have the same meaning as in figure 4.5. The "Local" sphere (dashed line) is the expanded version of the "Database" source sphere. The source sphere (database) is shown as the grey circular region. In the "Local" frame, the visible part of the atmosphere (as seen from satellite) subtends a spherical cap of half-angle θ_L^{max}. The corresponding half opening angle in "MAP" frame is θ_M^{max}. θ_L^i and θ_M^i are the boundaries of the angular bin i in "Local" and "MAP" frames respectively. To illustrate the assumption of conservation of photons within the same angular bin in both frames ("Local" and "MAP"), same colours are used to show the number of photons emitted from the Earth Atmosphere towards the satellite.

where ψ is a roll angle and for convenience we take $\psi = 0$. Furthermore, the "Local" and "MAP" frames are pixelated into $\mathcal{N}_\theta \times \mathcal{N}_\phi$ angular bins. The bin boundaries in the "Local" frame are θ_L^i, ϕ_L^j and in "MAP" frame are θ_M^i, ϕ_M^j where $i \in [0, \mathcal{N}_\theta]$ and $j \in [0, \mathcal{N}_\phi]$.

We then assume that the photons emitted towards the satellite from a given angular bin in the "MAP" frame are conserved within the corresponding angular bin in the "Local" frame. By again invoking the point source approximation, we can consider the source sphere dimensions negligible and the effective cross-section as seen from the "MAP" frame is $\pi \mathrm{R}_\mathrm{p}^2$. Then the number of photons in each angular bin i, j can be given as (see Juul 1976; Watts 1965 for detail derivation):

$$N_{ij}^{map} = \pi R_p^2 k^2 \int_{\theta_M^i}^{\theta_M^{i+1}} \int_{\phi_M^j}^{\phi_M^{j+1}} I(\theta,\, \phi) \frac{\cos\theta - k}{(1 + k^2 - 2k\cos\theta)^{3/2}} \sin\theta \; d\theta \; d\phi \; dt \tag{4.14}$$

Here $I(\theta,\, \phi)$ is the angle dependant emission intensity. It has a constant value in the case of the Reflection while in the case of the Albedo it is given by the Eqn. 4.3b.

Next, we assume that the "Local" sphere in the figure 4.9 is an expanded version of the source sphere and the spatial distribution of photons on the "Local" surface and the source sphere is identical. If we take equal area bins on the "Local" sphere, the angular size is conserved on the source sphere as well. For the database, we have the isotropic distribution of photons and in the case of equal area bins, the number of photons in each angular bin is constant (N^{sim}). We can define a ratio r_{ij} for each bin such that:

$$r_{ij} = \frac{N_{ij}^{map}}{N^{sim}} \tag{4.15}$$

If we have enough number of photons in each angular bin such that $r_{ij} \leq 1$ for all bins, we can select the fraction r_{ij} in each bin and estimate the correct number of photons.

Before applying the above selection, we first apply the directional selection explained in the previous section. It is done exactly opposite to the CXB case as in the case of the Albedo we are interested in the photons coming from the direction of the Earth. Every photon coming from the direction of the Earth is then distinguished based on the angular bin it lies in and the ratio value r_{ij} corresponding to that angular bin is assigned to it. Then for each photon, a uniform random number $\kappa \in (0, 1)$ is drawn. If $r_{ij} > \kappa$, then the photon is selected, otherwise it is discarded. This way, we keep the fraction r_{ij} in every angular bin.

Selection in the spectral domain

The selection in the spectral domain is used when the spectral properties of the background component are different or vary compared to the database. Hence, the

selection in the spectral domain corresponds to modifying the spectral distribution of database primaries to a new distribution.

The selection in the spectral domain is applied in the case of charged particles in the SAA because the incoming flux and the spectral shape of incident particles vary as the satellite passes through the SAA. To simplify the selection, we ignore the pitch angle dependence of the particles and assume an isotropic distribution around the satellite.

To illustrate the selection, let us consider a simple case. Let us assume that we want to simulate a spectrum $S(E)$ using a database of particles which has been generated using spectrum $S(E)^{dat}$ (in units of particles $\cdot\, \mathrm{cm}^{-2} \cdot \mathrm{keV}^{-1}$). We first define $\mathbf{s}(E)^{dat}$ which is the per particle database spectrum defined as the total database spectrum $S(E)^{dat}$ normalised to one incident particle, that is:

$$\mathbf{s}(E)^{dat} = \frac{S(E)^{dat}}{N_\circ} \quad [\mathrm{cm}^{-2} \cdot \mathrm{keV}^{-1}] \tag{4.16}$$

Here E is the energy and $N_\circ$ is the total number of particles of a given type drawn to generate the database. Then the spectrum $S(E)$, can be rewritten in term of a re-sampled version of the database spectrum as:

$$S(E) = r(E) N_{sim} \mathbf{s}(E)^{dat} \tag{4.17}$$

where $r(E)$ is an energy dependent ratio and N_{sim} is the total number of particles needed to simulate the spectrum $S(E)$. If we take N_{sim} such as,

$$N_{sim} = max\left(\frac{S(E)}{\mathbf{s}(E)^{dat}}\right) \tag{4.18}$$

then for each energy we have $r(E) \leq 1$. That is at each energy we have a larger number of particles in the database than predicted from $S(E)$. Using a similar approach of a random draw as in the case of the spatial distribution mapping, at each energy we select only fraction $r(E)$ from the total particles to extract the correct number of particles [3].

Note that although we store detected particles in the PIRA database, the energy selection is done using the input electron/proton GEANT4 spectrum and the expected electron/proton spectrum. This is possible because we also store the initial information (energy and direction, see table 4.1) about the detected primary in the database.

3. Note that in real case the spectral information is available in finite energy bins. If information is available with large bin-size, the spectrum is rebinned with smaller energy bins by interpolating between adjacent bins in the original spectrum (e.g. see Line 72-81 of ECLAIRs PIRA code for SAA electrons).

4.2.2.3 Assignment of time and dynamical simulation

In the selection process explained above, it is also possible to assign an arrival time to each primary. Physically speaking, arrival of primaries is a process which follows the Poisson distribution. Hence the difference (dt) between the arrival time of two successive primaries follows the exponential distribution. This is given by the following probability distribution function:

$$f(dt, \lambda) = \begin{cases} \lambda e^{-\lambda dt} & dt \geq 0 \\ 0 & dt < 0 \end{cases} \tag{4.19}$$

where λ is the average rate expressed as the number of primaries per unit time. We can determine the arrival time of each primary by computing the average primary rate for each background component. For example, if a given primary arrives at time t and $\dot{N}(t)$ is the rate of primaries at time t, then the arrival time of next primary is given by $t + dt$, where dt is computed by setting $\lambda = \dot{N}(t)$ in Eqn. 4.19.

However, because in the database we only store the detected primaries, to assign the time and pick primaries from the database, the average primary rate must be converted to the detected rate of the primaries. This is done by multiplying the average primary rate by the average detection efficiency ($\bar{\eta}$). The rate parameter is modified as $\lambda = \bar{\eta}\dot{N}(t)$. The average efficiency [4] is defined in the following manner (see Mate et al., 2019, for more details):

$$\bar{\eta} = \frac{\text{No. of detected database primaries}}{\text{No. of simulated database primaries}} \tag{4.20}$$

Using the above approach of assigning time, we can easily extend the PIRA algorithm to carry out a time-varying estimation of the background where the orbital parameters vary as a function of time. Major steps of this extension are shown in the figure 4.7. At each time t, the rate of primary ($\lambda(t)$) is updated (step 1). Based on this rate, the time is updated every time a primary is drawn from the database (step 2 and 3). Then depending upon the selection type, parameters such as Earth direction, flux ratio (Eqn. 4.15) or spectrum ratio (Eqn. 4.17) are estimated at time t to keep or reject the primary (step 4). This process is repeated for each component of the background. In the end, all the components are merged to obtain the overall background.

It should be noted that, in the case of CXB, Reflection and Albedo, the database is created with a component specific spectrum hence no selection in the spectral domain is required.

4. Note that the efficiency depends on the input spectrum of the primaries and will not be same for databases created using different input spectra.

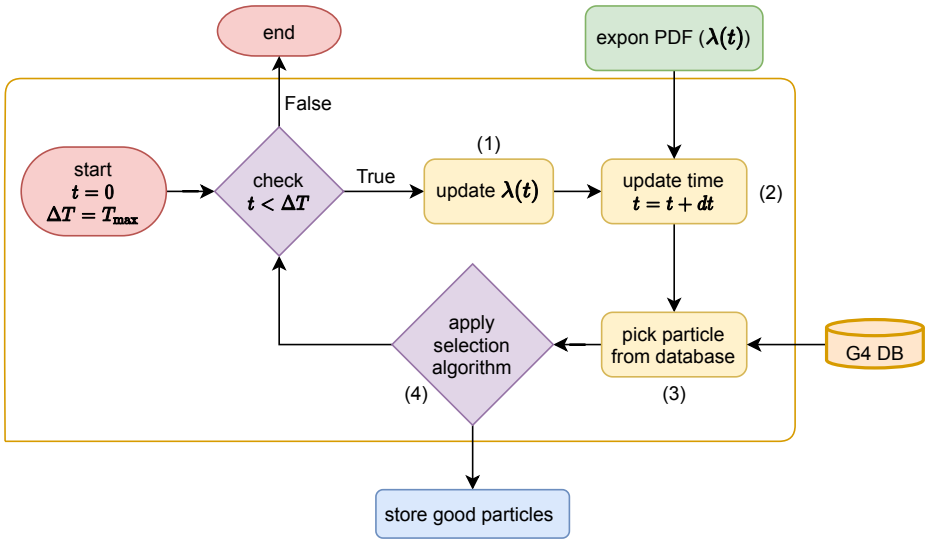


Figure 4.7 – A flow diagram showing the extension of PIRA algorithms to perform dynamical simulation.

4.2.3 ECLAIRs Background in the static and dynamic cases

Here I present the results we obtain for the ECLAIRs background in the static case and the dynamic case. We estimate the ECLAIRs background in the static case and simultaneously verify the PIRA in the case of Albedo and for charged particle spectrum. Then using the PIRA algorithms for the dynamic case we estimate the orbital variation of the ECLAIRs background.

4.2.3.1 ECLAIRs Background in static case

The ECLAIRs background in the static case is estimated at various zenith angles of the Earth with respect to the pointing direction of ECLAIRs. Such an analysis has been done previously by Zhao et al. (2012). To ease the qualitative comparison, we estimate the static background at the same zenith angles. The background contribution by individual components and the total background as a function of the Earth zenith angle are shown in figure 4.8. Here in the case of Albedo, we fix the satellite position to be at latitude $lat_{sat} = 30°\text{N}$ and longitude $lon_{sat} = 287°\text{E}$, where the maximum Albedo emission is observed [5] (as in Zhao et al. (2012); Bouchet & Atteia (2016)). We assume that the solar modulation potential (φ) is 0.25 GV in Eqn. 4.3b.

From the figure 4.8 it can be noticed that the CXB is the dominant background component when the Earth is not in the FoV of ECLAIRs. While the Albedo and Reflection flux is comparable when the Earth is in the FoV. In the case of the Albedo, the flux is not affected significantly even if the Earth is completely out of the FoV. This is because the Albedo spectrum is hard and consists of more high-energy photons

5. This is assuming the Geomagnetic frame with a reference year of 2015

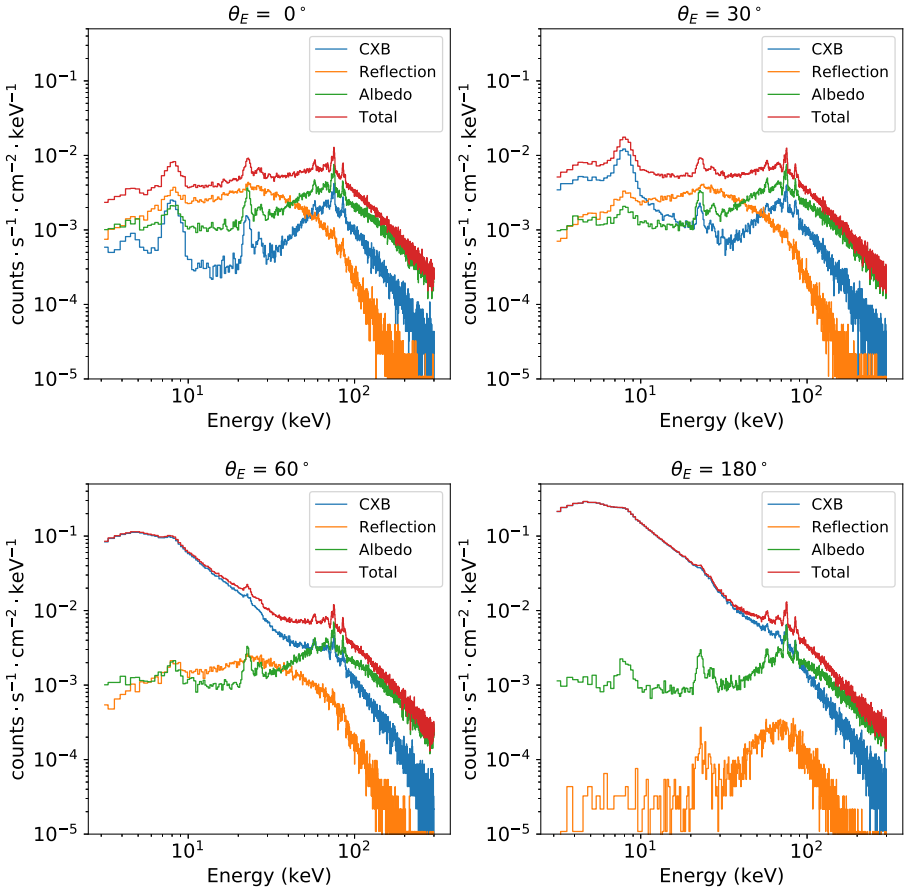


Figure 4.8 – ECLAIRs spectra of individual photon background components (CXB, Reflection and Albedo) as well as their sum for different Earth zenith angles $\theta_{\mathrm{E}} = 0°$ (top left), 30° (top right), 60° (bottom left), 180° (bottom right) The Albedo is estimated at the position where the maximum emission is observed (30°N and 287°E).

compared to the CXB and Reflection. In all cases, the fluorescence lines from the instrument elements such as mask, shield and detector are visible. These lines are useful for in-flight energy calibration of the detectors.

Albedo

To verify that the PIRA algorithms are correctly implemented, we verify them for the complex case of the Albedo emission (in a similar manner as done in Bouchet & Atteia, 2016). For that, we compare the spatial distribution of selected photons on the source sphere with a direct GEANT4 simulation. This is done for each Earth zenith angle. For each angle, we compare the distribution of the initial direction, energy and the source sphere coordinates of the selected particles. Figure 4.9 shows this comparison for the Earth zenith angle $\theta_E = 60°$ and azimuthal angle $\phi_E = 0°$. Furthermore, we compare the spectrum as a function of the Earth zenith angle in the case of PIRA and the direct GEANT4 simulation (Figure 4.10). Both the comparisons agree within expected statistical uncertainty.

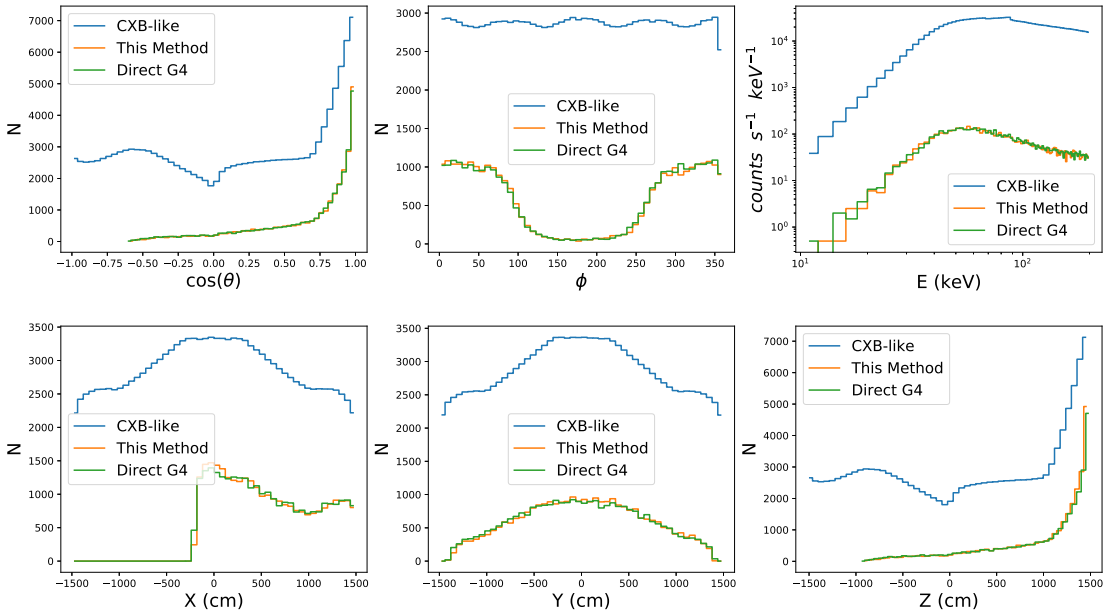


Figure 4.9 – Distribution of detected Albedo photons (N_D) using PIRA (Orange) and the related direct GEANT4 simulations (Green), for 60° Earth zenith angle. Distribution of incoming photon's zenith angle (Top-left) and azimuth angle (Top-right). Distribution of incoming photon's energy (Middle left). Initial photons position distribution (Middle right and bottom panels). In all the plots, "Database" (Blue) is the distribution of all the detected photons stored in the Albedo database (without any selections). It is arbitrarily scaled down to ease the comparison.

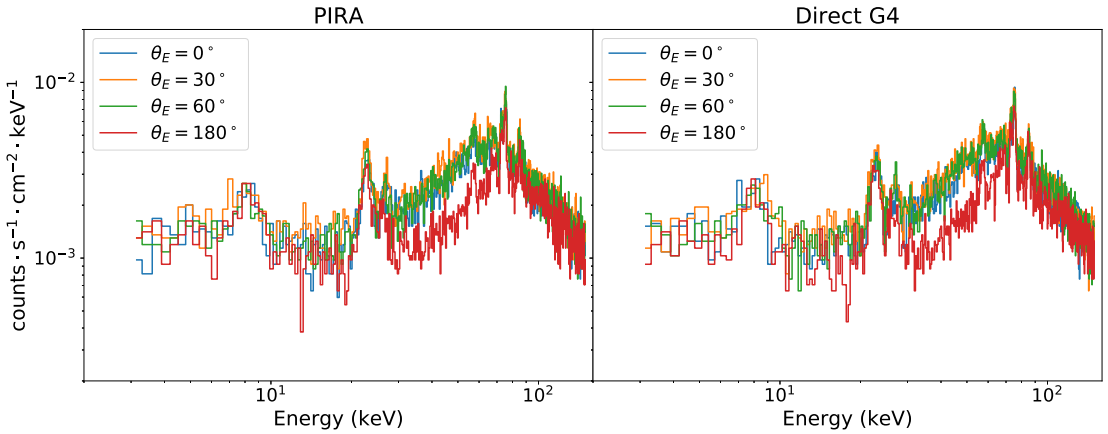
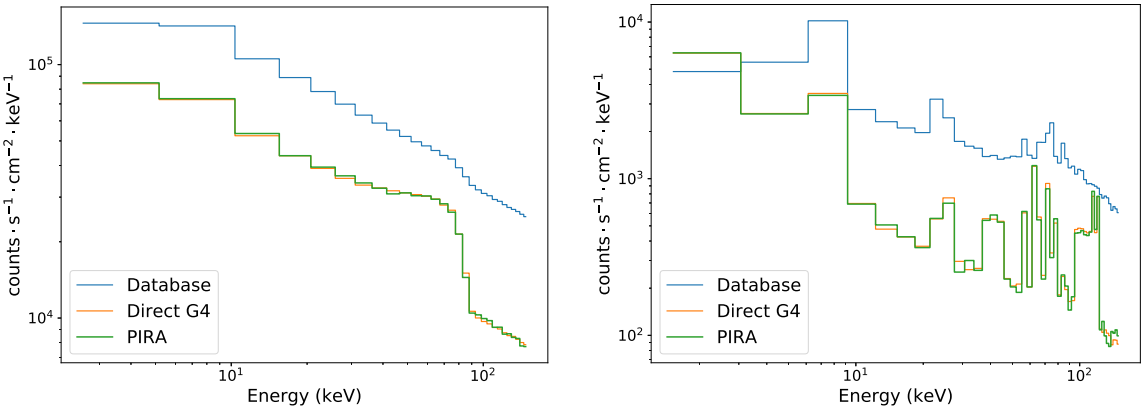


Figure 4.10 – Albedo background spectra for several Earth zenith angles. (Left) the output of PIRA, (Right) the output of a direct GEANT4 simulation. In both cases the Earth azimuth angle (ϕ_E) is equal to 0°.

Charged particle

As for the Albedo case, we verify the selection in the spectral domain by comparing the result of PIRA simulations of electrons and protons with a direct GEANT4 simulation. For that, we assume an incoming spectrum of known shape and intensity, in particular a power-law of particle index $\gamma = -2$ and $\gamma = -3$. We then compare the detected spectra estimated from the PIRA simulation with the direct GEANT4. Figure 4.11

shows this comparison for the input spectral index $\gamma = -2$ for both electrons and protons. The direct GEANT4 and PIRA spectra of detected particles match within expected statistical uncertainty.



(a) Electrons **(b)** Protons

Figure 4.11 – Spectra obtained using PIRA (Green) and direct GEANT4 simulations (Orange) for electrons (left) and protons (right). For both particles the input spectrum is power-law with $\gamma = -2$. The "Database" (Blue) is the spectrum of all the events stored in the electrons (and protons) database (arbitrarily scaled down to ease a comparison)

4.2.3.2 ECLAIRs Background in the dynamic case

We make use of the PIRA algorithms in dynamic case to estimate the ECLAIRs background as a function of the position of *SVOM* along its orbit. We use the orbital parameters from a simulation of the *SVOM* orbit by Jaubert et al. (2017).

The variation of the background photon components in a typical *SVOM* orbit is shown in figure 4.12. The pointing direction of ECLAIRs is constant during this orbit. The modulation of the background level is primarily due to the passage of the Earth within the FoV. From the figure, it can be seen that the CXB photons dominate the counts with an average rate of about ~ 2700 counts $\cdot$ s^{-1} when ECLAIRs is pointing away from the Earth. When the Earth completely occults the FoV, all three background components, i.e. CXB, Reflection and Albedo have comparable flux. Furthermore, the figure 4.13 shows the counts detected on the detector plane during the various Earth positions in the figure 4.12. It can be seen that the count intensity and its variation across the detector plane are significantly different for the four different Earth positions.

We use the models of Perry et al. (2008) and Ginet et al. (2007) to compute the electrons and protons spectra in the SAA respectively. The Systems Tool Kit (STK) [6] software is used to generate these spectra as a function of the *SVOM* position in the SAA.

6. http://www.agi.com/home

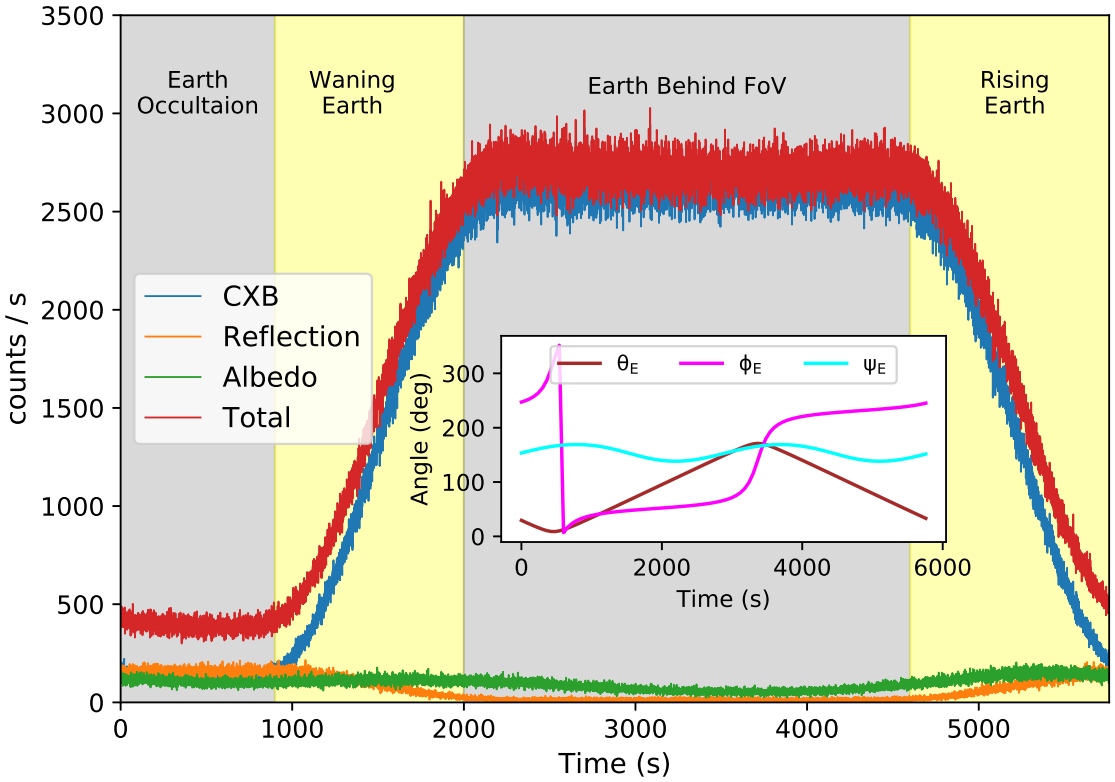


Figure 4.12 – Typical modulation of the photon background: CXB (Blue), Reflection (Orange) and Albedo (Green) for an orbit (duration $\sim$ 98 minutes). The variation of the Earth angles with respect to the pointing axis for this orbit is shown in the inset plot. The θ_E and ϕ_E are the polar and azimuthal angles while the ψ_E is the Earth roll angle which defines the direction of north in the ECLAIRs plane (For the definition of the angles and related transformations used for Albedo computation see Appendix A).

The detected count rate due to electrons and protons at the entrance of the SAA is shown in the figure 4.14. Although ECLAIRs will cease operations (see section 3.1.3.1) in the SAA; the study of the variation of the count-rate during the start and the end of the SAA is necessary to put constraints on the operations (e.g. to test the limits of the onboard trigger and memory buffers).

4.2.4 Performance of PIRA and statistical considerations

One of the salient advantages of the PIRA is its computational efficiency. As a consequence of reusing the particle interactions, the PIRA is $10^3 - 10^4$ times faster compared to the direct GEANT4 simulations. To illustrate the computation efficiency, in the table 4.3 (taken from Mate et al., 2019) we compare the performance of the PIRA and the direct GEANT4. The comparison is done for two cases – CXB and Albedo. It should be noted that for PIRA, the creation of the database can take a large amount of computation time, however, once this computation is done, the PIRA algorithms can estimate the background in few minutes.

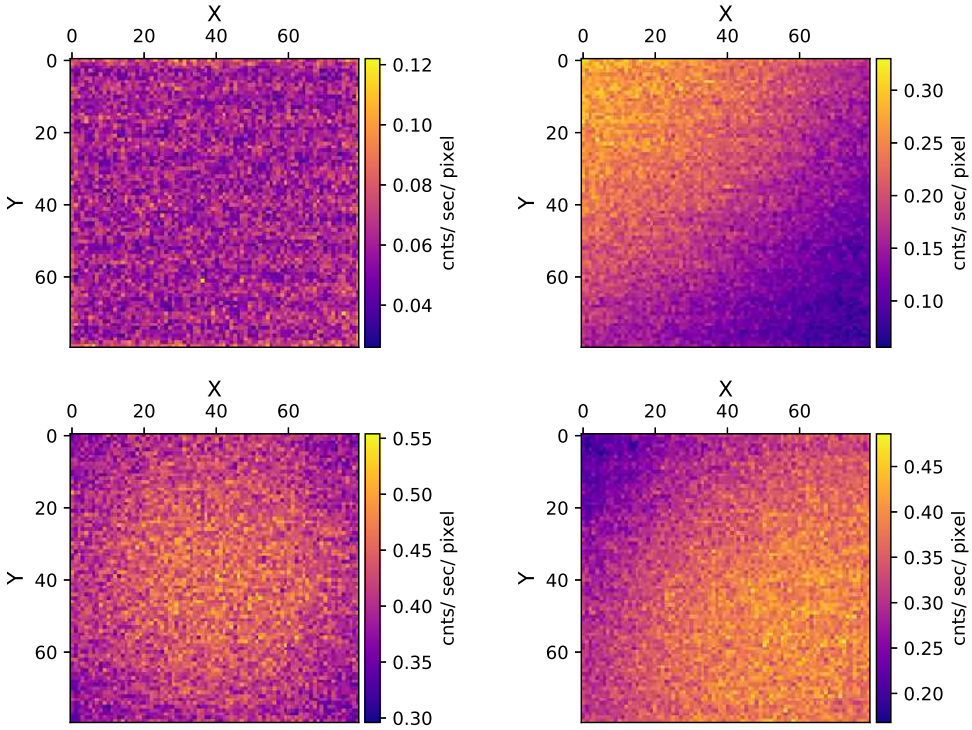


Figure 4.13 – Impact of the dynamic background on the detector plane image. The images show (clockwise, starting from top left) detected count-rate for each pixel when the Earth is completely in FoV, waning out of the FoV, completely behind the FoV and rising into the FoV respectively. The durations corresponding to each detector plane histogram are marked in figure 4.12.

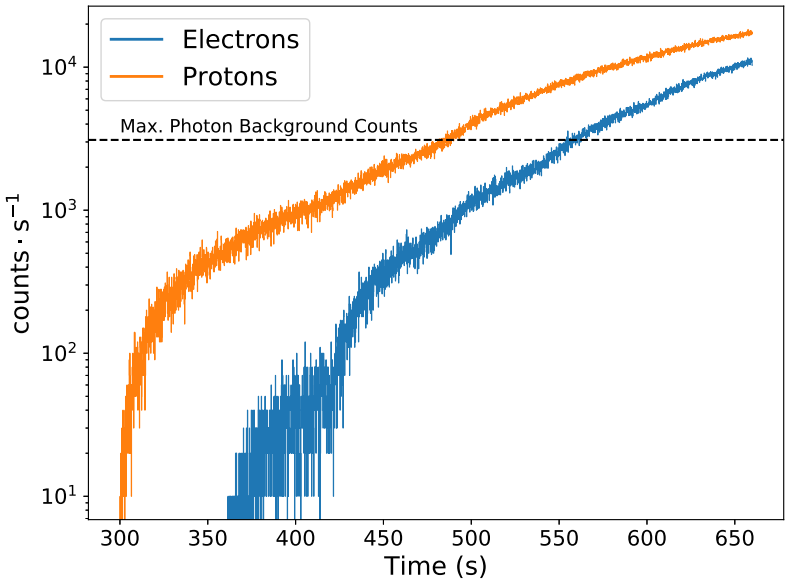


Figure 4.14 – Typical variation of the background (single events only) due to the electrons (Blue) and protons (Orange) when ECLAIRs enters the SAA. For comparison, the maximum photon background count rate (sum of the CXB, Reflection and Albedo components) is shown as the dotted line (see also Bouhargani, 2017a).

Table 4.3 – Comparison of the time taken by PIRA and a direct GEANT4 simulation to compute the background in the static case (without taking into account the time take to generate the databases).

Duration	Background Component	Computation Time (in minutes)	
		Direct GEANT4	PIRA
1 min	CXB	951	0.58
	Albedo	925	0.38
98 min	CXB	93150.88	5.9 (4.42)*
	Albedo	90625.75	16.28 (6.68)*

* The value in the parenthesis indicate the computation time in the dynamic case i.e, when the orbital parameters are evolving with time. The dynamic simulation takes less time compared to the static simulation because the number of photons to process are less in former case due to the modulation caused by the Earth (see figure 4.12.)

The database has a large but finite number of primaries. Therefore, the algorithms of PIRA should respect the original statistics of the database while picking up a primary from it. This is achieved if we do not pick a primary more than once for a given PIRA simulation. This puts a constraint on the maximum duration which can be simulated using PIRA. This duration is limited by the size of the database, i.e number of primaries simulated. The duration can be increased by simply increasing the size of the database.

For this study, we have used databases that allow a maximum simulation duration of ∼300 min. (∼3 orbits) for CXB, ∼570 min. (∼6 orbits) for Reflection and ∼135 min.(∼2 orbits) for Albedo. Therefore it is possible to simulate two orbits in a single simulation with the three components without running into limitations of statistics.

4.3 Simulation of X-ray sources and GRBs

The generation of ECLAIRs data also requires simulating the photons from bright X-ray sources (e.g. X-ray binaries, pulsars and AGNs etc.) visible to ECLAIRs as well as various types of GRBs. ECLAIRs, due to its large FoV, can detect multiple sources for a given pointing direction. Additionally, similar to the modulation of the background, the Earth passage through the FoV will modulate the source counts as a source rises/sets behind the Earth. Simulations of different types of GRBs with different observation scenarios are important to assess the sensitivity of the trigger to such observing cases.

Here, I will explain how the X-ray sources and GRB events are simulated in the case of ECLAIRs. A typical simulation involves two steps,

— Generation of a “photon list” which gives the arrival time of each photon and its energy at the entrance of the mask.

— A ray-tracing simulation and application of the instrument response to create an events list detected by the instrument. A GEANT4 simulation for this step will be done in the future. We use ray-tracing as it is a faster way (compared to GEANT4) to generate events for observing scenarios with GRB and sources having different position (in time or space).

I begin with an explanation of generating a photon list for X-ray sources and GRBs.

4.3.1 Source photon list simulations

A comprehensive catalogue of X-ray sources observable in the ECLAIRs energy range has been created by Dagoneau et al. (2020). The catalogue has source flux and spectral information (including variability) which can be used to model the source as observed by ECLAIRs. The catalogue is created using the (mean) flux information in the MAXI/GSC (2 - 20 keV, with 3 sub-channels) and SWIFT/BAT (14 - 195 keV, 8 sub-channels) catalogues where the fluxes are used to fit a spectral model for each source. A power-law [7] or a broken power-law [8] model is used for the fitting. The

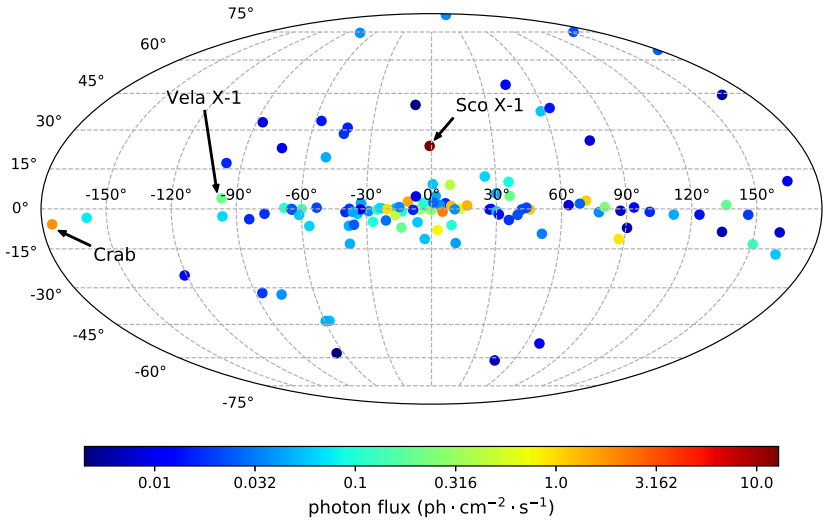


Figure 4.15 – A sample of relatively bright (SNR $> 6\sigma$ in 20 min) X-ray sources observable by ECLAIRs. The flux value is computed in the 4 -150 keV range. The figure is generated using the catalogue of N. Dagoneau (Dagoneau et al., 2020).

catalogue gives information such as the position, the flux in various energy bands, the spectral parameters, the variability. for each source. A sample of relatively bright sources, with signal to noise ratio (SNR) above 6σ in 20 minutes exposure, is shown in figure 4.15.

7. https://heasarc.gsfc.nasa.gov/xanadu/xspec/manual/node211.html
8. https://heasarc.gsfc.nasa.gov/xanadu/xspec/manual/node142.html

The spectral information in the catalogue is used to draw photons (with time and energy) coming from a given source. To understand the photon draw, consider a source with constant flux and a photon number spectrum $N(E)$. For a given energy range (E_1 to E_2), the total incoming flux is given by:

$$\mathrm{F}_{12} = \int_{E_1}^{E_2} N(E)\, dE \tag{4.21}$$

With this definition, the cumulative spectral probability density for the given source in the given energy band is:

$$\bar{\mathcal{S}}(E) = \frac{\int_{E_1}^{E} N(\tilde{E}) d\tilde{E}}{\mathrm{F}_{12}} \tag{4.22}$$

Using the uniform probability distribution between $(0, 1)$ we can invert the Eqn 4.22 to assign an energy (between E_1 - E_2) to each photon. If the instrument has a collecting area A (cm^2), an arrival time of the photon is estimate with the rate parameter $\lambda = \mathrm{F}_{12} \times \mathrm{A}$ using the method explained in section 4.2.2.3. The total number of photons ($\mathcal{N}$) arriving from the source in exposure of Δt are given by:

$$\mathcal{N} = \mathrm{F}_{12} \times \Delta\mathrm{t} \times \mathrm{A} \tag{4.23}$$

For example, if the photon spectrum is a power-law with the form:

$$N(E) = k \times E^{\alpha_{PL}} \tag{4.24}$$

where k is the norm ($\mathrm{ph} \cdot \mathrm{cm}^{-2} \cdot \mathrm{s}^{-1} \cdot \mathrm{keV}^{-1}$) at 1 keV and α_{PL} is the photon index. The inversion of Eqn. 4.22 is given by:

$$\mathrm{E}_{\mathrm{ph}} = \sqrt[p]{\frac{\mu \times p \times \mathrm{F}_{12}}{k} + E_1^p} \tag{4.25}$$

where μ is an uniform random number between $(0, 1)$ and $p = \alpha_{PL} + 1$. Figure 4.16 shows a comparison of the spectral model of Crab (from the catalogue) and spectrum of the photons obtained using above method. The drawn spectrum agrees with the model within the statistical fluctuations. In these simulations, we do not consider the variability in the source flux and spectral shape. The GRB photon-list simulation, which is a more general case with variable flux and spectral shape, is explained in the next section.

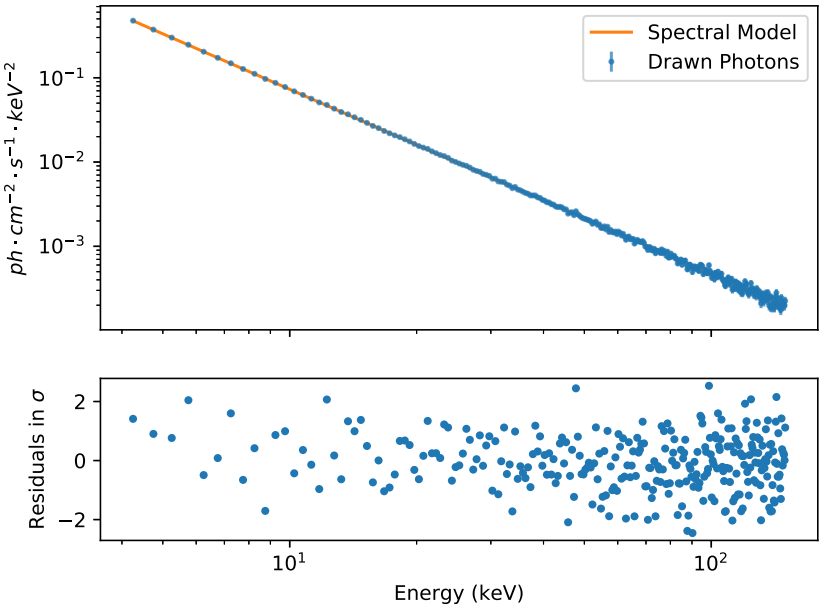


Figure 4.16 – Comparison of photon drawn spectrum with the source model for Crab. The residuals in terms of sigma are shown in the bottom panel. The residuals are estimated as (counts - model)/error. The drawn spectrum agrees with the model within the statistical fluctuations.

4.3.2 GRB photon list simulations

The GRB photon list simulations use the "méthode statique" explained in Antier (2016, chap. 5). The necessary tools[9] for these simulations are developed by the SVOM team at Institut d'astrophysique de Paris (IAP). The GRB sample is built from the GRBs detected by previous missions. In particular, GRBs in the catalogues by *CGRO*/BATSE (Kaneko et al., 2006a; Goldstein et al., 2013), *HETE-2*/FREGRATE (Pélangeon et al., 2008), *Swift*/BAT (Sakamoto et al., 2011; Heussaff, 2015), *Fermi*/GBM (Gruber et al., 2014) are used. The data, i.e the lightcurve (LC) and the spectrum (SP) of each GRB in these catalogues, are first converted to the ECLAIRs energy range (4-250 keV). For spectrum extrapolation, the best fit model from Power-law, Band function (Band et al., 1993), Smoothly Broken Power Law or Comptonised Power Law (Kaneko et al., 2006b) is used.

For the photon time draw, the lightcurve ($\mathrm{ph} \cdot \mathrm{s}^{-1}$) is first normalised using the extrapolated fluence in the ECLAIRs energy range. The lightcurve is converted to the cumulative probability density (see figure 4.17). Then a random number (μ) is drawn from a uniform distribution between [0, 1] and the cumulative density function is inverted to get the value of the arrival time of a single photon. This is repeated for the total number of photons (N) which are determined by the area at the entrance

9. http://svom.iap.fr/index.html

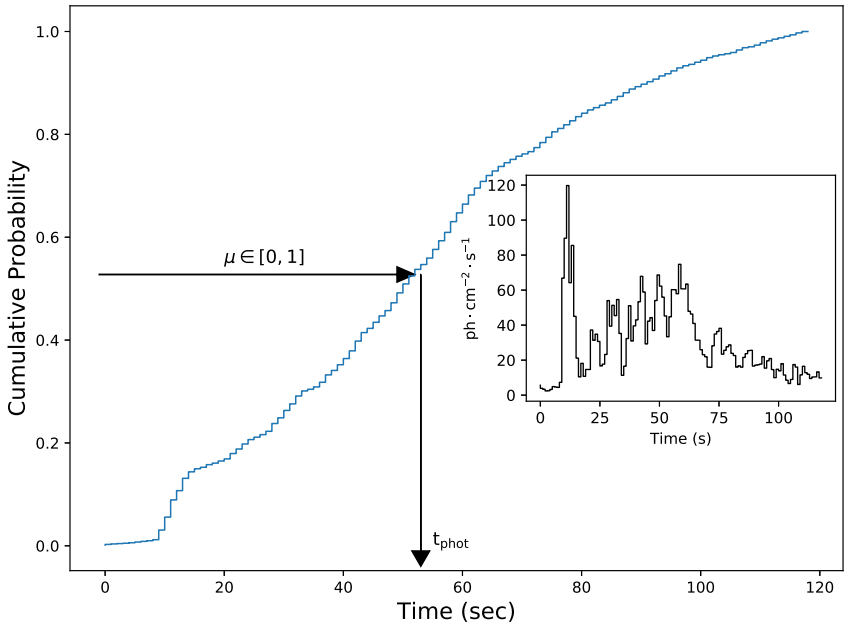


Figure 4.17 – The method of assigning photon arrival time to the GRB photon list. The lightcurve of the GRB (inset plot) is used to construct a cumulative distribution (blue curve). Then a uniform random number (μ) between 0 and 1 is drawn which is used to invert the cumulative distribution to determine the photon arrival.

of the mask (~ 1296 cm^2) and the duration of the given GRB. To assign energy to each photon, the cumulative distribution of the best fit spectral model is used. The method to invert the spectral distribution is similar to the one explained in the previous section. If in the original data of the GRB, a time-resolved spectrum is available, then based on the drawn photon arrival time the spectral model for the corresponding time is used to draw the energy of the photon. Furthermore, if the redshift is known in the original data, then it is also possible to correct the spectrum and lightcurve for the effect of the redshift and simulate the photon-list of the GRB at a different redshift. More details about the simulation parameters and calculations can be found in Daigne (2019).

4.3.3 Generation of source/GRB events

The photon-list (both GRB and source) is passed through a ray-tracing simulation[10] which gives an event-list similar to the background with each event having a time, energy and pixel position on the detector. The simulation also takes into account the detection efficiency of the CdTe detectors. We do not consider any energy redistribution and hence for these simulations, all events have single pixel interaction.

A geometrical model of ECLAIRs (with a mathematical representation of the

10. The ray-tracing simulations are adapted from the codes written as part of the ECLAIRs general program pipeline (Goldwurm, 2018).

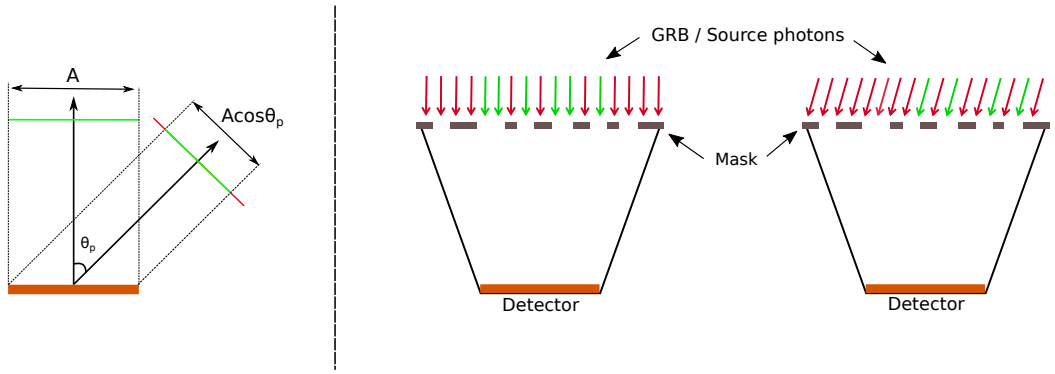


Figure 4.18 – A schematic showing the process of selecting source / GRB photons to generate events in ECLAIRs. Left: Illustration of off-axis flux ($\propto \cos\theta_p$) variation due to a lesser projected area. Right: Illustration of the photon selection process through the mask (for on-axis and off-axis source); photons passing through the mask hole are picked (green arrows) and photons hitting the mask opaque element or shield are rejected (red arrows).

mask pattern and detector geometry) is used to carry out the ray-tracing simulations. First, based on the source location relative to the pointing of ECLAIRs, each photon in the photon-list is assigned a direction. The incoming photon flux is then corrected based on the photon direction (θ_p) as the incoming flux varies $\propto \cos\theta_p$ compared to the on-axis flux (Figure 4.18: Left). After this correction, a position at the entrance of the ECLAIRs mask is randomly drawn for each photon. Based on the position and the direction, the intersection of the photon path on the mask (hole or opaque element) and the detector (pixel values) is determined. The photons which encounter opaque elements are discarded and the ones encountering holes are kept along with their detector pixel values (Figure 4.18: Right). The time and energy of the event are the same as of the photon in the photon list.

The photons reaching the detector are further corrected for the detection efficiency ($\eta(E)$) which is a function of the incident energy. The detection efficiency for ECLAIRs is given in the Ancillary Response File (ARF, Sizun, 2015b). The photon arriving at the detectors are then further selected in the following way (see also figure 4.19),

1. For each photon arriving at the detector having energy E, find the value of the efficiency at that energy : $\eta(E)$.
2. Draw a uniform random number between 0 and 1 : ρ
3. If $\eta(E) > \rho$, select the photon or else reject it.

4.4 ECLAIRs background and data simulators

The PIRA and source/GRB simulation methods are used to develop ECLAIRs background and data simulators. The background simulator computes the expected

background events for a given orbit or a part of an orbit. The simulator uses the orbit and attitude information (Jaubert et al., 2017, e.g. pointing direction, Earth position in the FoV) and the GEANT4 databases as an input. Currently, the simulator can generate events for CXB, Reflection, Albedo and charged particles in the SAA.

The output of the background simulator is passed to the ECLAIRs data simulator to generate complete data by adding X-ray source and GRB events. The ECLAIRs data simulator, along with the background events, takes the source catalogue (section 4.3.1) and GRB photon list as inputs. It takes the GRB position in the sky (Right Ascension, Declination), GRB time (with respect to the start time) as user parameters. Based on the orbit and attitude information, it determines the sources in the FoV and generates a photon list for them. These photon lists (for sources and GRB alike) are used to generate events as explained in section 4.3.3. In the output of the simulator, every event is tagged with an ID to indicate which (e.g. CXB, X-ray source or GRB) photon (or particle) generated the event. This makes it possible to separate events for different types of analysis. The simulator is generic and can handle stable as well as variable (slew) pointing. A flow chart showing the algorithm behind the data simulator is shown in figure 4.19. Figure 4.20 shows one orbit lightcurve of events simulated using the data simulator for a stable pointing and a pointing involving a slew.

The events data generated by these simulators is primarily used to test and optimise the trigger algorithms (see Chapter 4). It also has other applications, for example,

— To test data reduction process in the ECLAIRs pipeline.

— To test the in-flight calibration methods using the instrument lines (This can be done via direct GEANT4 simulation as well but using PIRA reduces the computation time.).

— To study the performance of the detector properties such pixel efficiency, noisy pixel removal process, expected data rates etc. (Bouhargani, 2017b; Atteia et al., 2018; Arcier et al., 2020a,b).

— To carry out end to end simulations using an electronic simulator (ELS) to test UGTS configurations (Garnier, 2019; Arcier et al., 2020c).

— To study the scientific performance of ECLAIRs, for example, determining the sensitivity of ECLAIRs to nearby GRB population (Arcier et al., 2020d).

The PIRA, X-ray source and GRB simulations have a enabled generation of data representing close to real observation cases. Such data enable us to study the performance of ECLAIRs. Particularly, in the context of the thesis, the simulated data is used to test the performance of the onboard trigger and to test a possible offline trigger algorithm as discussed in the following chapter.

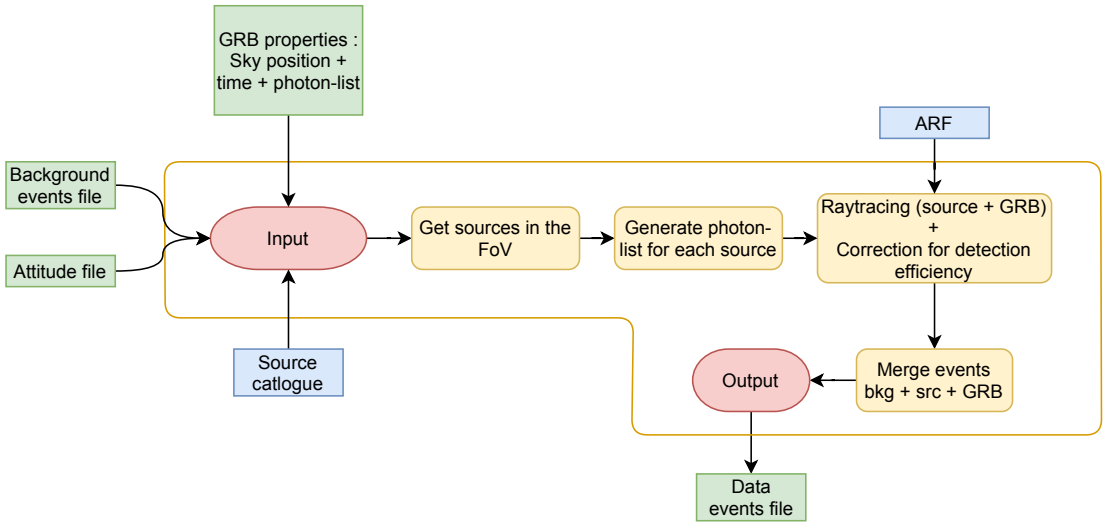


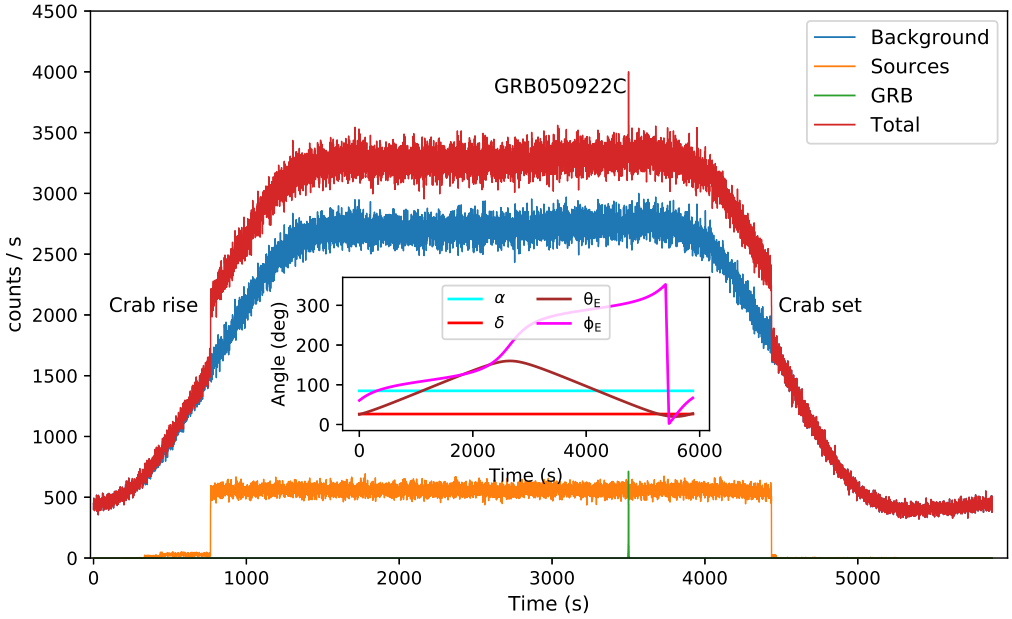
Figure 4.19 – Organisation of the data simulator tool showing the input, output and important processes.

4.5 PIRA: future perspectives

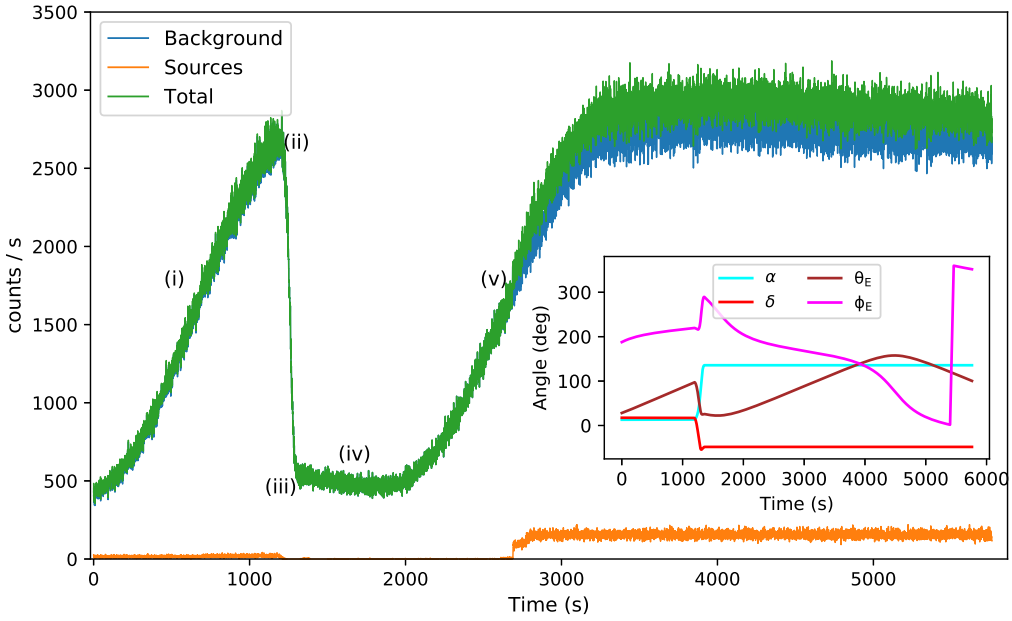
The application of PIRA to ECLAIRs has demonstrated the accuracy and computational efficiency of the method. In these PIRA simulations, we have used the ECLAIRs mass model and a very rough mass model of the satellite platform. When the complete *SVOM* mass model will be available, the PIRA database will be updated. Note that because of the generic nature of the approach, changes in the mass model will not necessitate a change in the processing codes of PIRA. In addition, the cosmic-ray component will be added to the PIRA simulations as well.

A background component that is not discussed here is the activation background. The activation background contribution depends on the impurities in the satellite/instrument materials. Before the instrument launch, the impurities are not well known and only a rough estimate of the activation background is possible. The knowledge of the impurities (type and relative abundance) can be deduced more accurately in flight by looking at the decay lines in the observed spectrum. For these simulations, an approach used by missions such as *INTEGRAL* (Weidenspointner et al., 2003) or *Hitomi* (Sato et al., 2012; Odaka et al., 2018) may be used.

Furthermore, the generic nature of PIRA is already being exploited in the context of *SVOM*. PIRA will be used to determine the background (and its variation) for the *SVOM*/GRM instrument and this work has been already started by the *SVOM*/GRM team at the Laboratoire Univers et Particules de Montpellier (LUPM).



(a) Orbit with stable Pointing



(b) Orbit with variable pointing (slew)

Figure 4.20 – A complete simulation of ECLAIRs data. Figure (a): Light curve of an orbit with a stable pointing and X-ray sources (Crab) and a GRB in the FoV. Figure (b): Light curve of an orbit with a variable (slew) pointing. In this figure, various stages of the orbits are marked with Roman numerals with (i) and (v) indicating the Earth exiting the FoV, (ii) indicating the start of the slew, (iii) indicating the end of the slew and (iv) indicating the Earth occultation. For both plots, the variation of the orbital parameters is shown in the inset plots.

Chapter 5

Performance of ECLAIRs trigger

The real-time detection of GRB and their follow-up is the key objective of ECLAIRs. This can be only achieved if ECLAIRs is equipped with an onboard detection mechanism that can detect the transients. The onboard analysis (done by an onboard trigger) is necessary mainly because it is not possible to have a real-time downlink (e.g. to download the data) and uplink (e.g. to issue slew commands from the ground) with the satellite. However, the onboard trigger will suffer from sensitivity variations due to the varying background of ECLAIRs (see Chapter 4). Therefore it is important to evaluate the sensitivity of the trigger in the context of accurate data simulations.

In the first part of this chapter, I summarise the performance and sensitivity analysis of the onboard trigger carried out using the simulated data (section 5.2). The onboard trigger performance analysis is restricted to the count-rate trigger[1]. The analysis is carried out by making a simplified model of the onboard trigger, independently of the existing onboard software. The trigger model is designed to be compatible with the output of the ECLAIRs data simulator (see section 4.4) and has been also used to perform other scientific performance tests (e.g. see Arcier et al., 2020d).

In addition to the onboard trigger, ECLAIRs will have an "offline" trigger to detect transients on the ground (see section 5.3). I propose a method that can be used as part of the offline trigger. The applicability of the method is demonstrated using the simulated data in the section 5.3. I begin the chapter with a brief description of the onboard trigger algorithm.

1. A sensitivity analysis for the image trigger can be found in the thesis of N. Dagoneau (Dagoneau, 2020)

5.1 Onboard trigger and GRB detection with ECLAIRs

The onboard trigger of ECLAIRs is one of the key components of *SVOM*. The trigger will analyse the data stream coming from ECLAIRs in real-time and search for GRBs and other high energy transients (TDEs, SGRs etc.). In this section, I review some of the main properties (requirements, algorithm, implementation etc.) of the onboard trigger.

5.1.1 Onboard trigger: Requirements

The onboard detection of GRBs is a complex task due to their diverse temporal and spectral characteristics, variable instrument background (see chapter 3) and the limited resources (memory, data processing power) availability. Given these constraints, for efficient detection and localisation of GRBs (as well as other transients), the onboard trigger should satisfy the following scientific and technical requirements:

— Universality: It should be sensitive to all known types (short/hard, long/soft, X-ray rich etc.) of GRBs irrespective of their temporal shape of spectral characteristics.

— Robustness: It should not be strongly affected by the background variation and should have minimum false alarm rates.

— Optimised computation time: It should be able to analyse the data rapidly and should provide the time and localisation information with a minimum delay.

5.1.2 Onboard trigger: Algorithm and implementation

The onboard trigger employs two methods to detect GRBs: a count-rate trigger (better suited for shorter timescales, from 10 ms up to 20 s) and an image trigger (better suited for longer time scales, from 20 s to 20 min). Both methods will be applied on four energy bands (Figure 5.1 top) to ensure that the trigger is sensitive to the different GRB spectral characteristics (e.g. soft or hard, X-ray Rich GRBs etc. see Chapter 2). In addition, to obtain a better sensitivity for off-axis[2] GRBs, the count-rate trigger method will be applied on 9 detector zones (Figure 5.1 bottom). Here I will describe both algorithms and their onboard implementation briefly. Details about the implementation of the trigger (software and prototype hardware system description) can be found in Schanne et al. (2008, 2013); Le Provost et al. (2013).

2. Off-axis angle is the angle between a given direction and the pointing axis of the telescope.

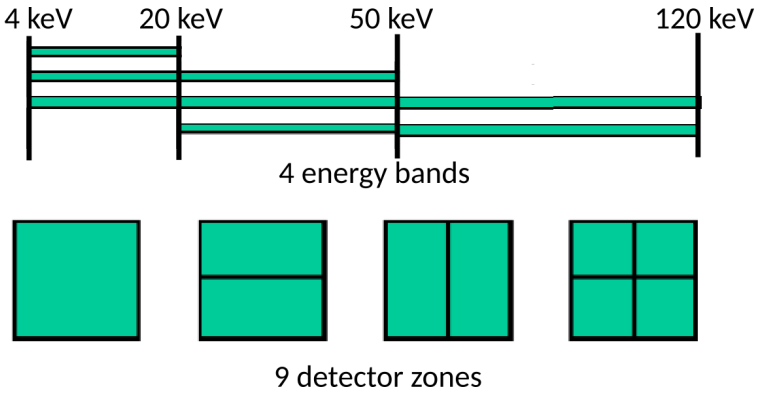


Figure 5.1 – An example of four energy bands for the onboard trigger (both count-rate and image trigger) and the 9 detector zones (full detector, 4 detector halves, and 4 quadrants) used by the count-rate trigger. The energy bands are programmable and the energy boundaries can be uploaded from the ground. Image credit: Schanne et al. (2013)

5.1.2.1 The image trigger

The image trigger works as a cyclic process running every 20.48 seconds on 4 energy bands independently. At every cycle, a shadowgram (or detector count image, see section 3.2.2) is constructed using the photons detected over the 20.48 seconds time interval before the start of the cycle. The shadowgram is used to construct a sky counts image and a sky variance image (see section 3.2.2) which are stored in the memory of the UGTS (see section 3.2.1). These new sky images are stacked with the previous ones to construct sky images on 7 timescales ranging from 20.48 to 1310.72 seconds (∼20 min). For each timescale (except for the shortest one), a set of two images that are half-overlapping in time are also generated (see figure 5.2).

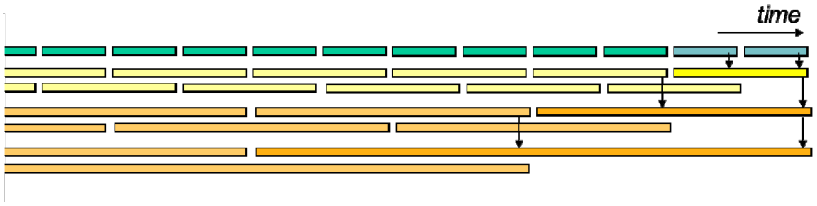


Figure 5.2 – Sky image summation process for the image trigger. Longer timescale sky images are constructed by summing consecutive images (Green). Except for the shortest timescale, all timescales have half-overlapping time intervals. Image credit: Schanne et al. (2013)

At the end of each cycle, the sky counts and variance images are processed to construct the sky SNR image ($SNR_{img,ij}$, see Eqm. 3.8) at all timescales. The sky SNR image is analysed to find locations (pixel values) of the excess which are above a certain threshold (currently set to 6.5σ). These locations are then converted to sky coordinates and are compared with an onboard catalogue (Dagoneau et al., 2020). If an excess is at a location that is not found in the catalogue, a transient identification is confirmed.

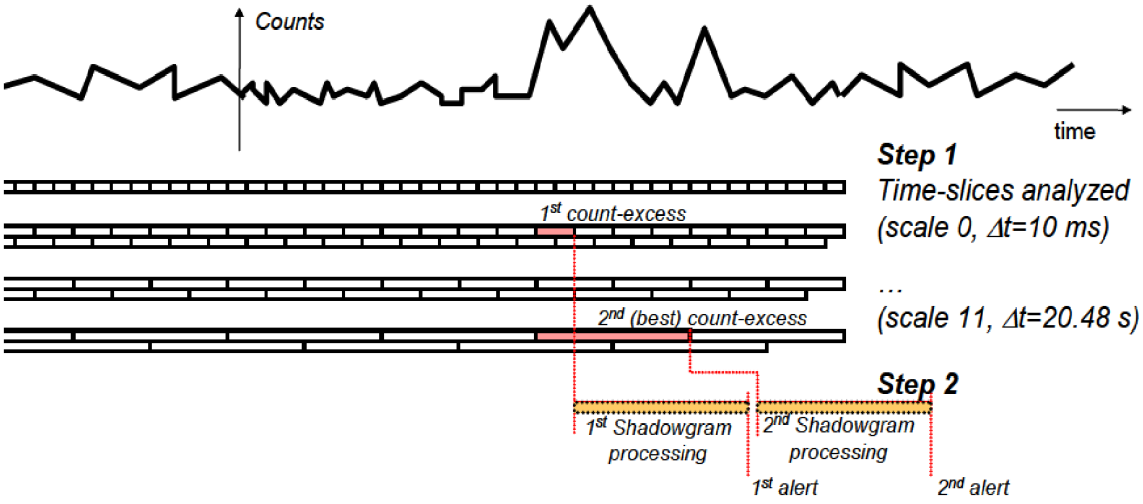


Figure 5.3 – Count-rate trigger algorithm: Counts are binned at a minimum timescale of 10 ms (scale 0) and then are added pairwise (with semi overlapping intervals) to build longer duration timescales up to 20.48 ($= 2^{11} \times 10$ ms or scale 11). Counts at each timescale are analysed to find excess above the background. When a first best excess is found, imaging is performed to find the localisation information. The "alert" in the figure refers to the slew alert sent to the satellite and the detection alert sent via the VHF network (section 3.1.3.2) to the ground. Image credit: Antier (2016)

5.1.2.2 The count-rate trigger

The count-rate trigger works as a two step process cyclically running every 2.5 seconds (Figure 5.3). In the first step, the detected counts are integrated over fixed duration time intervals (called time bins) and are compared to the predicted background counts to find excesses with a significance above a predefined threshold. For a given timescale (duration) Δt, the counts excess significance (in σ) at the i^{th} time bin is defined as:

$$\text{SNR}_{\text{cnt,i}} = \frac{C_i - B_i \Delta t}{\sqrt{B_i \Delta t}} \tag{5.1}$$

where C_i are the detected counts in the bin and B_i is the extrapolated background rate (cnt/s) for that bin. To extrapolate the background, a degree two polynomial fit is performed on the counts measured over 300 s duration before the start of the current cycle. This step is independently performed on the 4 energy bands, 9 detector zones, 12+11 timescales (12 timescales from 10 ms to 20.48 s and 11 semi-overlapping timescales from 20 ms to 20.48 s). All the detected excesses are stored in the memory along with their time duration (start and end times of the time bin).

In the second step, the best excess among the excesses not analysed yet is picked. The time interval corresponding to this excess is used to construct a sky SNR image ($\text{SNR}_{\text{img,ij}}$, Eqn. 3.8). The sky SNR image is processed similarly to the image trigger to look for a new source.

Note that the image processing associated with the count-rate trigger is indepen-

dent of the image trigger. Also, the model of the onboard count-rate trigger differs from the existing onboard software in terms of the coding principle.

5.2 Performance of the onboard count-rate trigger

I discuss the method and results of the count-rate trigger performance in this section. The analysis includes the impact of the background on the detection threshold and detection sensitivity, the impact of the trigger parameters (e.g. timescales, energy bands) on the count-SNR and the sensitivity of the count-rate trigger to GRB characteristics. Before going into the details of the various analyses I carried out, I explain some of the common parameters and terminologies used throughout the rest of the chapter.

Background phases: For this analysis, we choose an orbit with a pointing direction close to the galactic pole such that no persistent sources are present in the FoV[3]. For this orbit, we define two "background phases" to indicate the background count variations with respect to the time (Figure 5.4). These phases are:

— Increasing/ decreasing phase: This is defined by the duration for which the background counts are increasing or decreasing. During this phase, the Earth partially occults the FoV but it is slowly going out (increasing counts) or coming into (decreasing counts) the FoV. The phase corresponds to the duration when the count-rate is between 25% to 100% of the maximum count-rate.

— Flat phase: This is defined by the duration for which the Earth is completely out of the FoV and background counts are approximately constant.

Ideal SNR: The ideal-SNR ($\mathrm{SNR_{ideal}}$), for each time bin i, is defined as:

$$\mathrm{SNR_{ideal,i}} = \frac{n_i^{grb}}{\sqrt{n_i^{bkg}}} \tag{5.2}$$

where, n_i^{grb} and n_i^{bkg} are the number of detected GRB counts and background counts respectively. The number of GRB counts and background counts in each time bin can be separated because in the simulations, every detected count has an unique ID representing the origin of the photon (see section 4.4). The $\mathrm{SNR_{ideal}}$ is a good indicator to quantitatively compare the accuracy of the $\mathrm{SNR_{cnt}}$.

3. This is chosen to avoid the additional complexity in the interpretation of the analysis due to the presence of sources

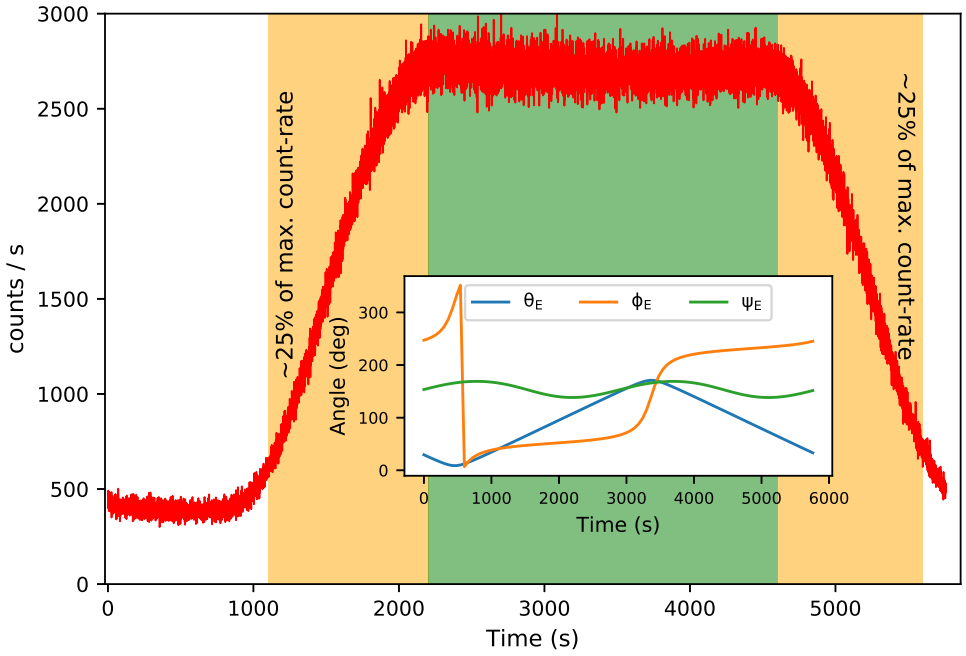


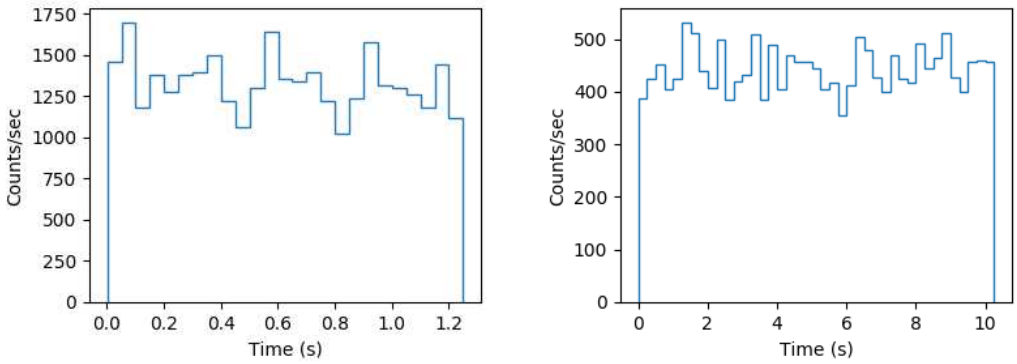
Figure 5.4 – The definition of background phases. Shaded Orange region: increasing/decreasing phase; shaded Green region: flat phase. The inset plot shows the variation of Earth angles (defined with respect to the pointing axis) theta (θ_E or zenith angle), phi (ϕ_E or azimuthal angle) and roll (ψ_E) as function of time (For the definition of the roll angle see Appendix A).

GRBs sample For the analysis, we use a sample of GRBs that have different spectral and temporal characteristics. We have two types of samples, the first one consists of synthetic GRBs. These GRBs have durations equal to the search timescales and a square shaped lightcurve (Figure 5.5). The input spectrum of these GRBs is fixed to be a powerlaw with a known photon index (α_{PL}). This sample is useful to quantify the impact of the GRB parameters on the count-rate trigger as these parameters can be varied manually.

The second sample consists of real GRBs taken from the IAP GRB database (see section 4.3.2). This sample is selected to cover GRBs with different durations and spectral properties (Table 5.1).

5.2.1 Detection threshold for count-rate trigger

The first step to characterise the onboard trigger performance is to estimate the SNR distribution obtained purely due to the background count variation. This analysis is important to determine the SNR threshold value above which any excess can be considered as a detection. For this analysis, we choose 5 orbits similar to those in figure 5.4). The count-rate trigger is run on the simulated background data and the SNR values are combined to obtain the distribution of the background SNR for different timescales and different background phases.



(a) A short synthetic GRB, $\Delta T = 1.28$ s **(b)** A long synthetic GRB, $\Delta T = 10.24$ s

Figure 5.5 – Representative count lightcurves of short (left) and long (right) synthetic GRBs.

Table 5.1 – Spectral and temporal properties of real GRB sample‡

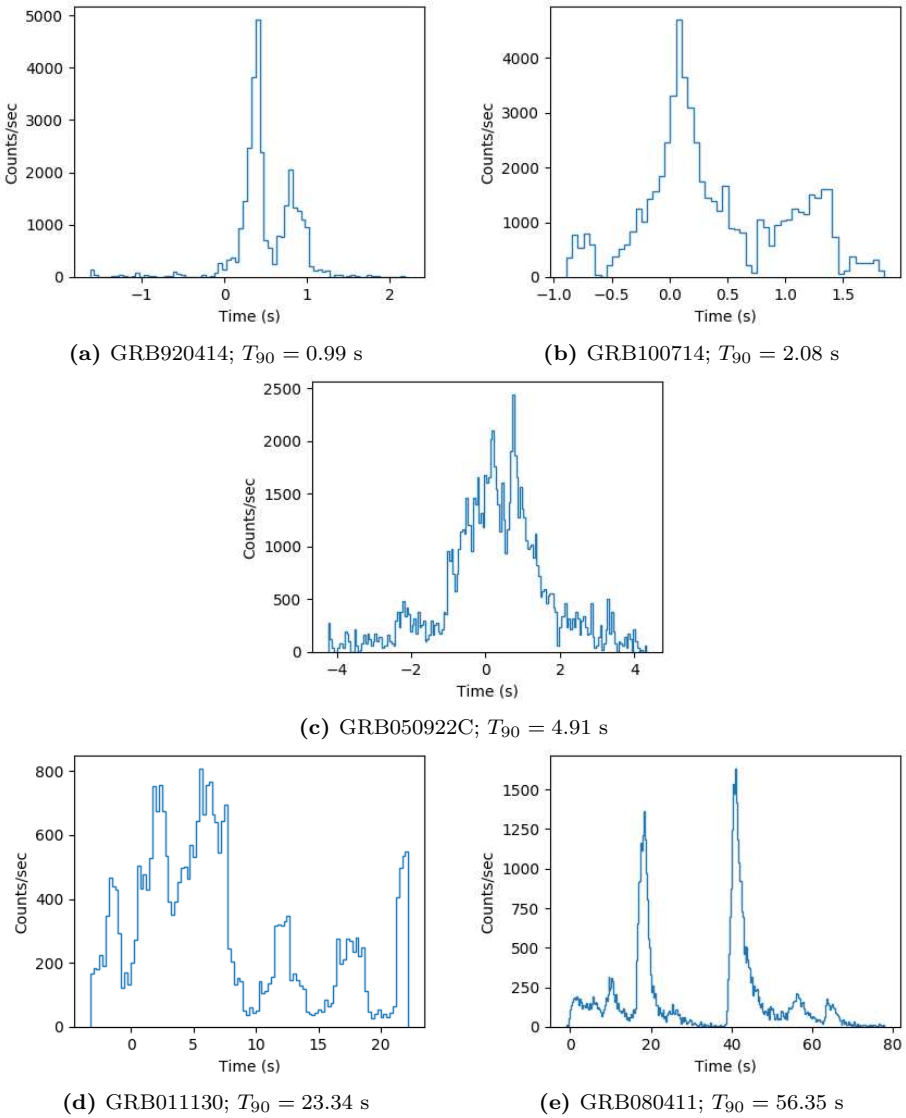
Name	T_{90}† (s)	α	β	E_p (keV)
GRB920414	0.99	-0.96	-2.14	769.4
GRB100714	2.08	-1.13	-	89.9
GRB050922C	4.91	-0.83	-	130.5
GRB011130	23.34	-1.74	-2.7	3.9
GRB080411	56.34	-1.51	-	259.0

† The T_{90} is in 4 - 120 keV energy range.
‡ More details about these GRBs can be found at http://svom.iap.fr/GRB_REFERENCE/php/movegrb.php

Figure 5.7 shows the background SNR distribution in the 4 - 120 keV energy band for four timescales and for different background phases. As expected, due to the random fluctuations in the counts, the SNR distributions have a nearly Gaussian shape centred around zero. From the analysis, we can conclude that for orbits with no X-ray sources in the FoV, 5.5σ is a conservative threshold to separate the SNR due to the presence of a GRB/transient from the SNR due to purely background counts. However, due to the background component not accounted for in the current PIRA simulations (e.g. charge particle hits, activation effect), we take a slightly higher count-rate threshold as 6.5σ. This is chosen to be consistent with the image-SNR threshold determined by other analysis (Gros, 2016a).

5.2.2 Detection sensitivity of count-rate trigger

One of the important properties of the count-rate trigger is the counts detection limit for a given significance. It is quantified by the minimum excess counts required



(a) GRB920414; $T_{90} = 0.99$ s

(b) GRB100714; $T_{90} = 2.08$ s

(c) GRB050922C; $T_{90} = 4.91$ s

(d) GRB011130; $T_{90} = 23.34$ s

(e) GRB080411; $T_{90} = 56.35$ s

Figure 5.6 – Count light curves of real GRBs taken from IAP database.

to detect a GRB with a significance of n-σ. The detection limit varies along the orbit due to the variable background. It also depends on the energy band as the background counts vary across energy bands. To quantify the impact of the background variation, we do the following analysis,

— We set the minimum required detection significance to be 6.5σ.

— We run the count-rate trigger (excluding the imaging step) on the entire orbit shown in the figure 5.4, excluding the parts where the Earth completely occults the FoV and obtain the $\mathrm{SNR_{cnt}}$ for all energy bands and all timescales as a function of time.

— We select two timescales, one long (10.24 s) and one short (1.28 s) and two energy bands (4 - 20 keV and 20 - 120 keV) and we estimate the minimum count-rate excess required.

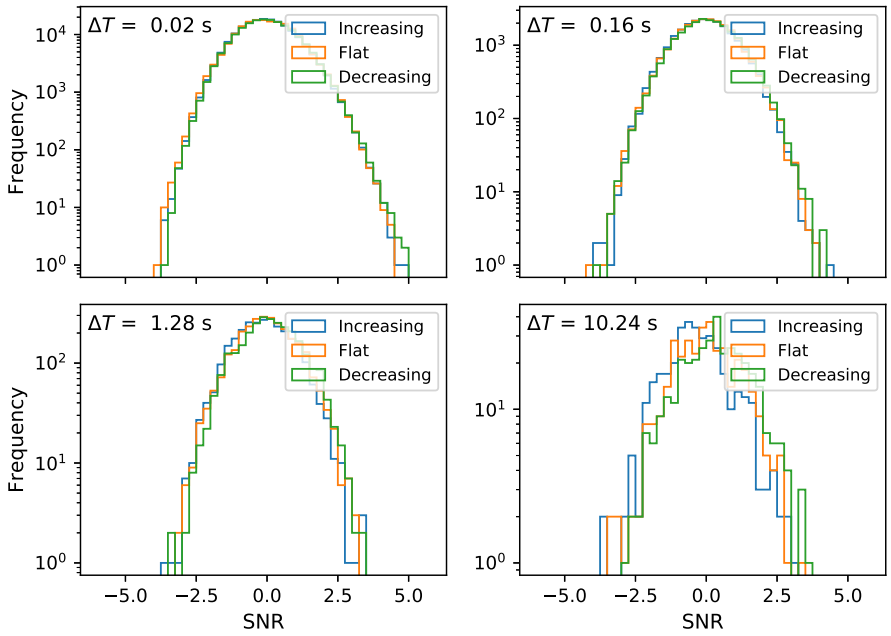


Figure 5.7 – Distribution of background count-rate SNR for different background phases and timescales (simulation of 5 orbits.). The distribution is for 4 - 120 keV and entire detector zone

Figure 5.8 shows the variation of the minimum count-rate excess required as function of time (see figure 5.4 for variation of orbital parameters). The analysis shows the detection limit dependence on the energy band and the GRB duration.

We further extend this analysis to determine the sensitivity of the trigger as a function of the *SVOM* orbit. The sensitivity is quantified by estimating the on-axis flux detection limit as a function of time (for n-σ detection). The minimum count-rate obtained from the detection limit analysis is converted to flux using the instrument effective area (ARF, see section 4.3.3) and by assuming a powerlaw spectral shape for the GRBs. For the powerlaw photon index (α_{PL}), the average values of short and long GRB indices (-1.2 and -1.6 respectively) found in the latest *Swift*/BAT catalogue (Lien et al., 2014) are used. Figure 5.9 shows the variation of the minimum flux needed for a 6.5σ detection as a function of time for typical 1 s and 10 s GRBs (duration in T_{90}). The figure shows that the sensitivity (in the flat part of the orbit) for a typical long/soft GRB is approximately 5 times better compared to the sensitivity for a short/hard GRB. It should be noted that during the increasing/decreasing background phases (i.e. for time $\lesssim$ 2000 s and $\gtrsim$ 5000 s in figure 5.8 and 5.9) the sensitivity to GRB is better in terms of detecting count excess. However, the presence of the Earth in the FoV pose problems for imaging and the better sensitivity to detect count excess doesn't necessarily imply GRB will be always detected and localised in the image.

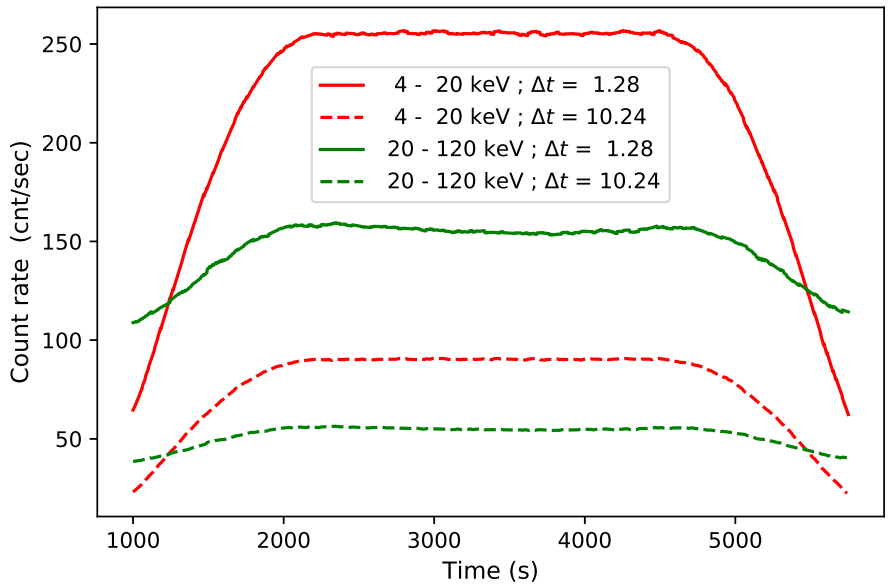


Figure 5.8 – Minimum count-rate (cnt/sec) for 6.5σ detection as function of the orbit for two energy bands (4 - 20 keV and 20 - 120 keV) and two timescales (1.28 s and 10.24s).

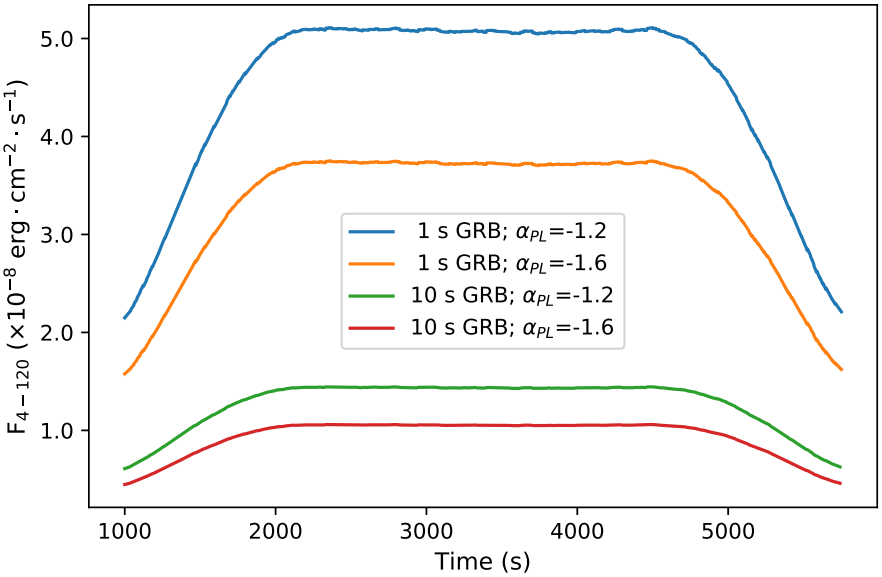


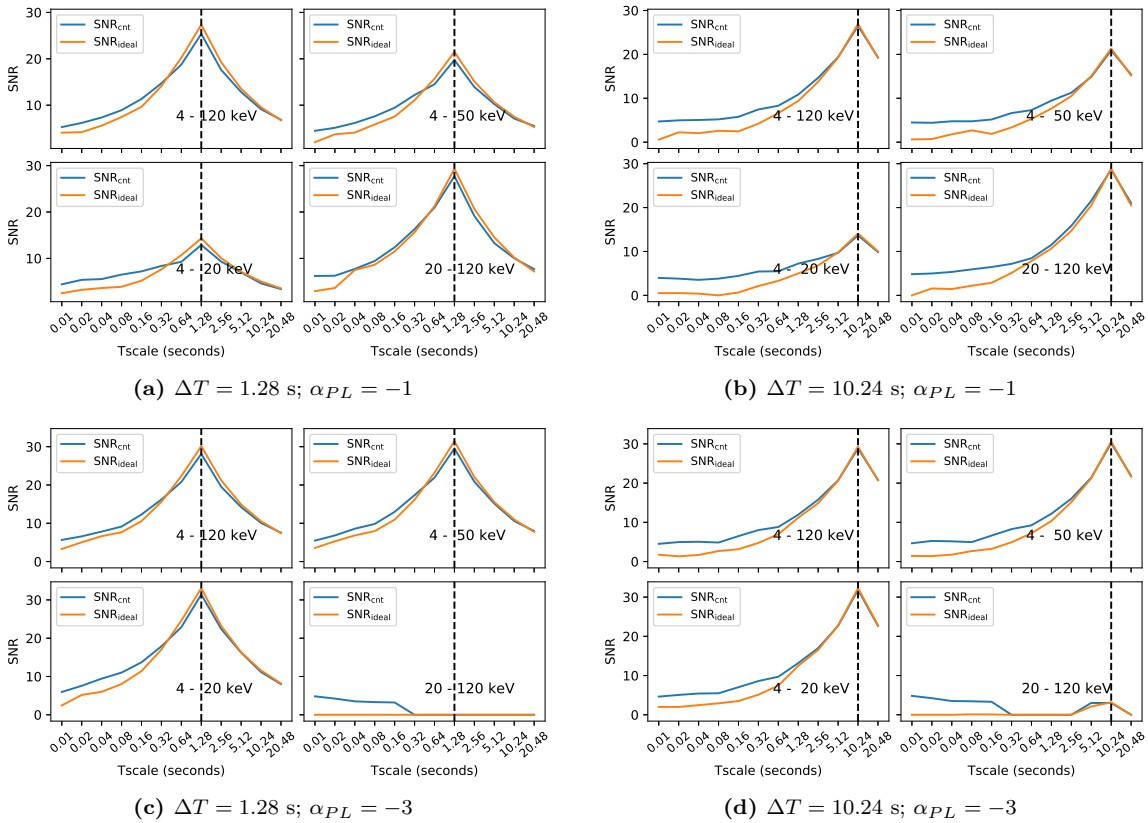
Figure 5.9 – Minimum on-axis flux (erg cm^{-2} s^{-1}) needed to detect a typical 1 s and 10 s GRB above 6.5σ significance in the 4 - 120 keV energy band.

5.2.3 Impact of trigger parameters on count SNR

Here we evaluate qualitatively and quantitatively how the count-SNR depends on the various trigger parameters. This includes the count-SNR dependence on timescale, energy band, off-axis position and corresponding image-SNR (SNR$_{\rm img}$). We carry out this analysis on both synthetic and real GRBs.

Synthetic GRBs

The first analysis concerns the dependence of the count-SNR ($\mathrm{SNR_{cnt}}$) on the search timescale and energy band. For this, the synthetic GRBs are simulated during the flat part of the orbit with photons incident on-axis. The count-rate trigger algorithm is run to find the maximum count-SNR at different timescales and energy bands. The variation is shown in figure 5.10 for four GRBs having two different durations and two different photon indices. This figure shows that the count-rate trigger correctly recovers the duration of the GRB with $\mathrm{SNR_{cnt}}$ being maximum at the timescale corresponding to the GRB duration. Concerning the energy bands, for harder spectrum ($\alpha_{PL} = -1$), the $\mathrm{SNR_{cnt}}$ is maximum in the high energy band (20 - 120 keV) while for softer spectrum ($\alpha_{PL} = -3$, typical for an X-ray flash or XRF) the $\mathrm{SNR_{cnt}}$ is higher in the low energy bands (maximum in the 4 - 50 keV band) with no detection in the high energy band. These results are consistent with the expectations.



(a) $\Delta T = 1.28$ s; $\alpha_{PL} = -1$

(b) $\Delta T = 10.24$ s; $\alpha_{PL} = -1$

(c) $\Delta T = 1.28$ s; $\alpha_{PL} = -3$

(d) $\Delta T = 10.24$ s; $\alpha_{PL} = -3$

Figure 5.10 – Maximum count-SNR variation with respect to the timescale and energy band for synthetic GRBs. The vertical dashed line indicate the timescale of the GRB. For comparison, the ideal-SNR ($\mathrm{SNR_{ideal}}$, Eqn. 5.2) is also plotted.

The second analysis involves the impact of the GRB position in the FoV on the count-rate trigger and the relation between count-SNR ($\mathrm{SNR_{cnt}}$) and corresponding image-SNR ($\mathrm{SNR_{img}}$). For this, we simulate a GRB at random sky positions (2500 positions, uniformly distributed in the FoV) within the FoV during the flat phase of

the orbit. For each position, we estimate the maximum count-SNR (in all 9 zones and 4 energy bands, see section 5.1.2) and the corresponding image-SNR[4]. Figure 5.11 shows this distribution of count-SNR and image-SNR within the FoV for 1.28 s long synthetic GRB. Both SNR values are nearly constant within the fully coded FoV (marked by Red square in the figure) and then decrease rapidly towards the edges. This result is qualitatively consistent (e.g. in terms of the shape of the distribution) with a purely theoretical estimate given by Gros (2016b).

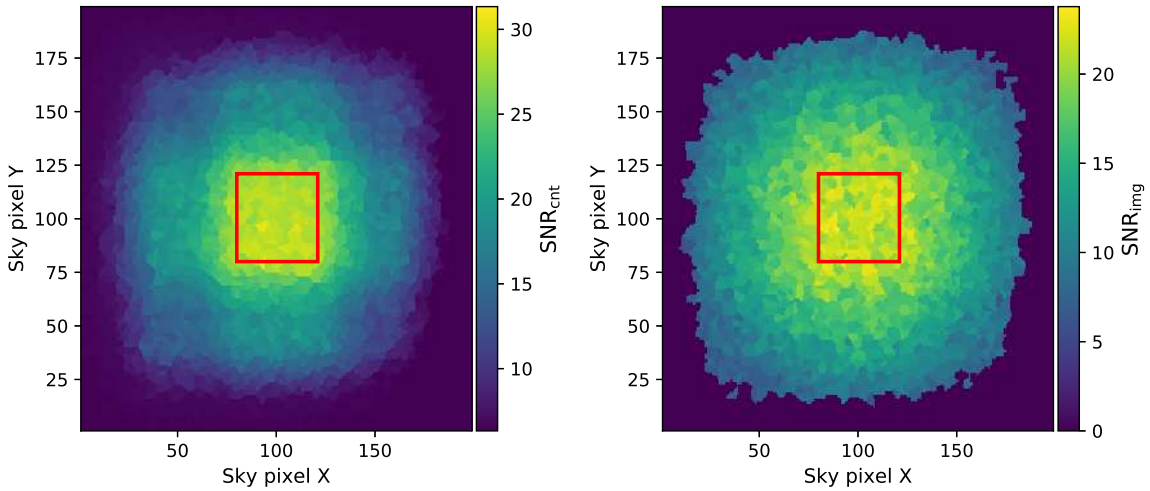


Figure 5.11 – The variation of count-SNR and corresponding image-SNR within the FoV for 1.28 s synthetic GRB. The bounds of the fully coded FoV are indicated by the Red square.

Furthermore, figure 5.11 shows that although, the count-SNR and image-SNR vary in a similar fashion within the FoV, the maximum value of count-SNR and image-SNR are not equal (see also, Schanne, 2017). The relation between the two is shown in the figure 5.12a and 5.12b where count-SNR values are plotted against the corresponding image-SNR for short and long synthetic GRBs. This relation between the count-SNR and image-SNR can be theoretically expressed using Eqn. 5.1 and Eqn. 3.8 as:

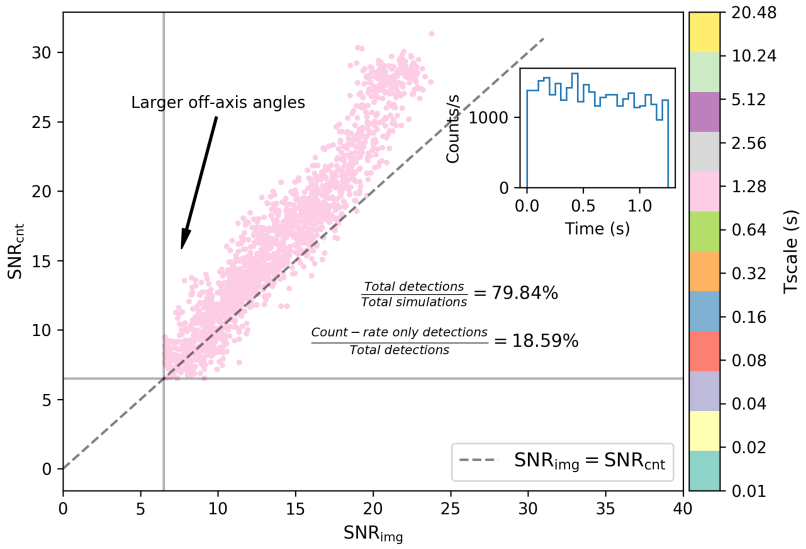
$$\mathrm{SNR}_{\mathrm{cnt}} \approx \mathrm{SNR}_{\mathrm{img}} \times \sqrt{1 + \frac{S}{B}} \tag{5.3}$$

where S and B are source and background counts respectively. From this, we can expect two possible behaviours, (a) If the background counts dominate over the source counts, the count-SNR and image-SNR values are approximately equal and (b) If the source counts dominate over background counts, count-SNR and image-SNR values are not comparable.

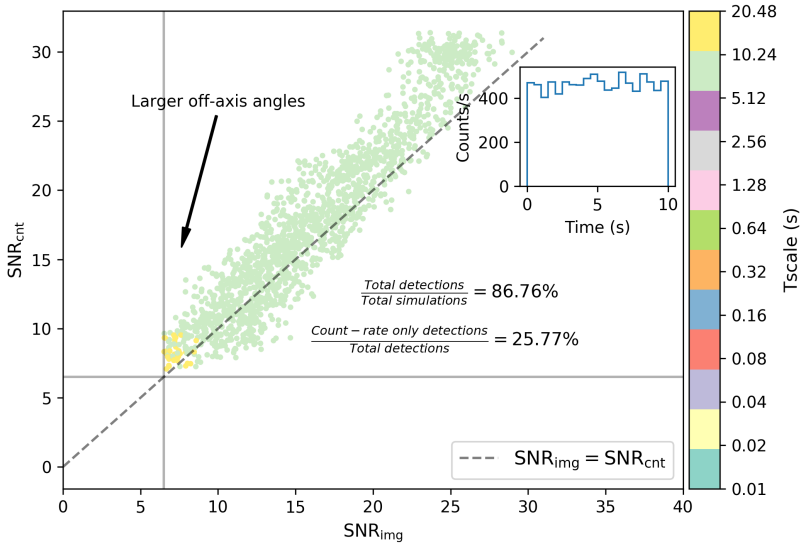
Case (a) is observed for fainter long GRB as the background counts are dominant over the source counts in this case (Figure 5.12b). Case (b) is observed in the

4. For both SNRs we set a detection threshold of 6.5σ.

case of short GRBs because source counts dominate over the background counts due to shorter integration time (figure 5.12a). It is also observed for bright long GRB ($SNR_{cnt} \gtrsim 20$, Figure 5.12b) as in this case source counts are higher compared to the background counts. This behaviour implies that for some short GRBs which are well above the count-rate detection threshold (e.g. GRBs with $SNR_{cnt} = 10\sigma$), there is a possibility that the corresponding image-SNR is below the detection threshold and the localisation of such GRBs will not be possible.



(a) 1.28 s GRB

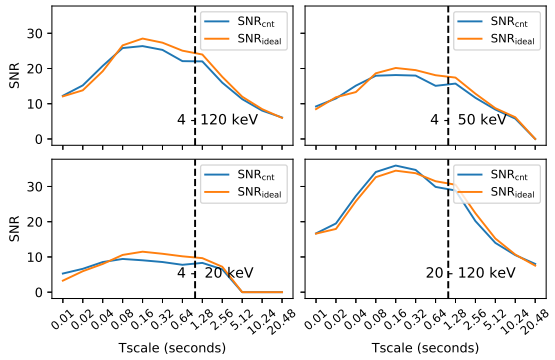


(b) 10.24 s GRB

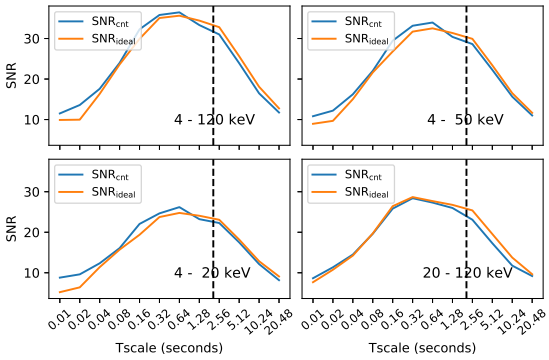
Figure 5.12 – The relation between the count-SNR and the image-SNR for two synthetic GRBs. Figure (a) is for a 1.28 s GRB and figure (b) is for 10.24 s GRB. The points are colour coded based on the timescale at which both the SNRs were computed. The horizontal and vertical grey lines show the SNR threshold for count-rate and image trigger respectively (both 6.5σ).

Real GRB sample

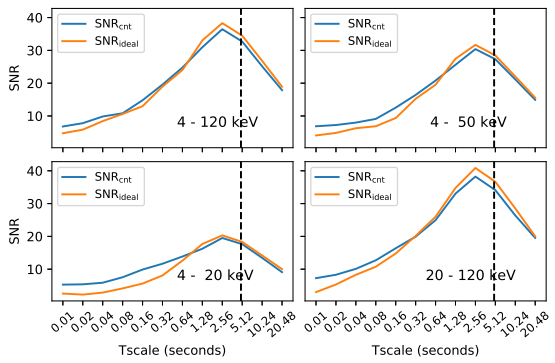
We repeat the analysis carried out in the case of synthetic GRBs for the real GRBs. Figure 5.13 shows the variation of the count-rate trigger SNR against the timescale. The figure indicates the count-rate trigger is not sensitive to the duration (T_{90}) of the GRB but to the duration of the GRB peak. This is seen in the case of GRB920414, GRB100714 and GR080411 where the maximum count-rate SNR is observed at lower timescales (approximately corresponding to the timescale of the brightest peak in the GRB lightcurve). The sensitivity of the count-rate trigger to the energy band is consistent with the previous analysis. This is seen in the case of GRB920414 and



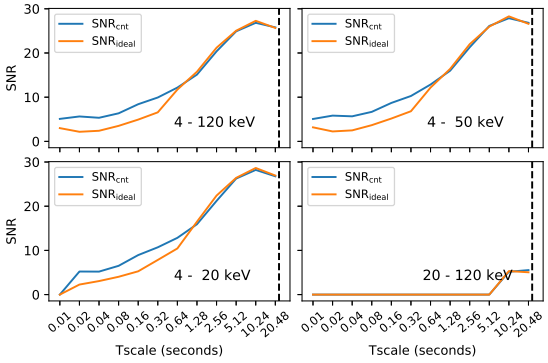
(a) GRB920414; $T_{90} = 0.99$ s



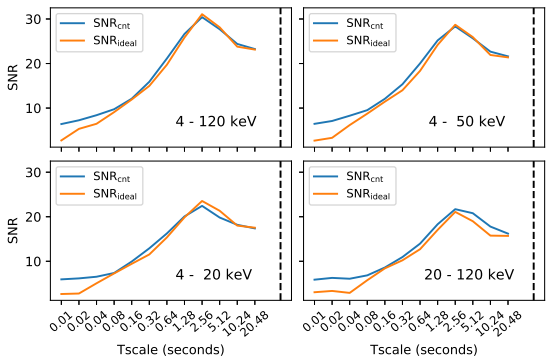
(b) GRB100714; $T_{90} = 2.08$ s



(c) GRB050922C; $T_{90} = 4.91$ s



(d) GRB011130; $T_{90} = 23.34$ s

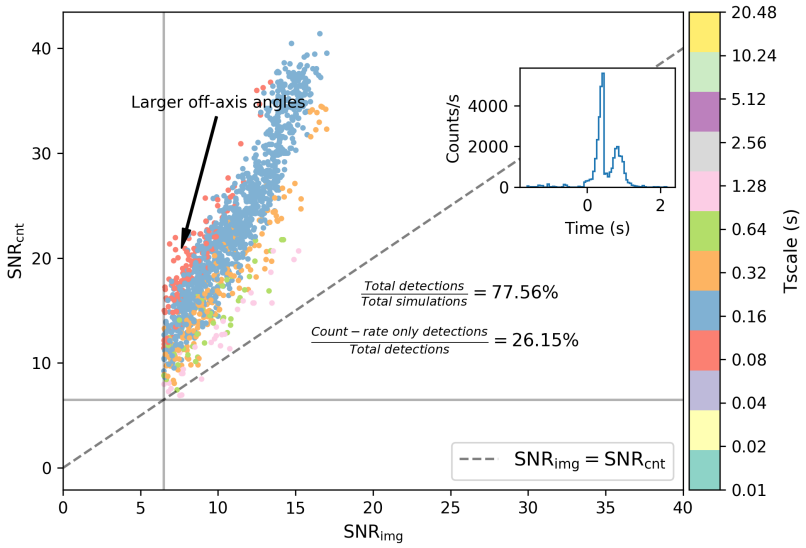


(e) GRB080411; $T_{90} = 56.35$ s

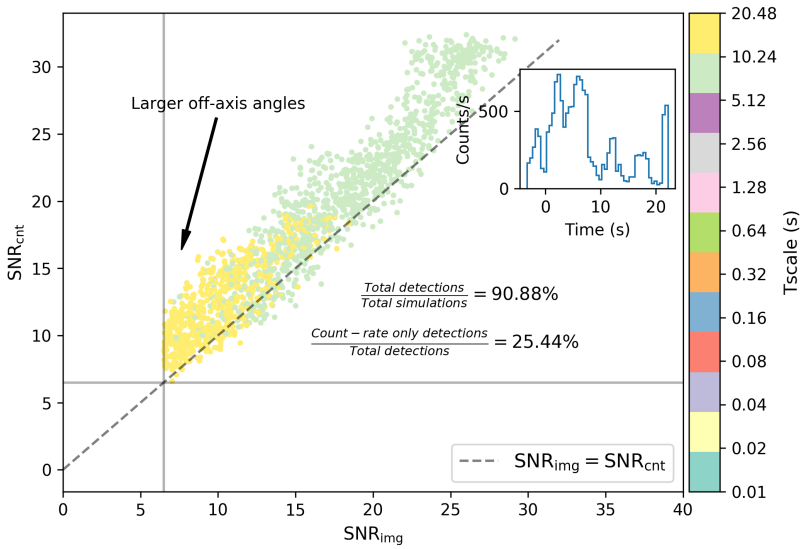
Figure 5.13 – Maximum count SNR variation as function of the timescale and energy bands for the sample of real GRBs. The vertical dashed line indicate the T_{90} duration of the GRBs.

GRB011130, where the former, a classical short-hard GRB gives maximum count-SNR in 20 - 120 keV band while the latter, an X-ray flash, gives the maximum count-SNR in 4 - 20 keV band.

Similar to the synthetic GRBs, we also compare the count-SNR and the corresponding image-SNR for two real GRBs (one short and one long). We obtain a consistent result that shows that for shorter timescales the count-SNR and image-SNR follow the case (a) explained in the previous subsection. This is particularly



(a) GRB920414



(b) GRB011130

Figure 5.14 – The relation between the count-SNR and the image-SNR for two real GRBs. Figure (a) is for GRB920414 (T_{90} = 0.99 s) and figure (b) is for GRB011130 (T_{90} = 23.34 s). The points are colour coded based on the timescale at which both the SNRs were computed. The horizontal and vertical grey lines show the SNR threshold for count-rate and image trigger respectively (both 6.5σ).

highlighted in the case of the GRB920414 which is a multi-peak GRB and the maximum count-SNRs are detected at different timescales. The count and image SNRs for longer timescale deviate less compared to the shorter timescale SNRs (Figure 5.14a). For the long burst, GRB011130, the count-SNR and image-SNR distribution follow the case (a) for high SNR (bright burst) value and follow the case (b) for small SNR values (faint bursts, Figure 5.14b).

5.2.4 Count-rate trigger sensitivity to GRB characteristics

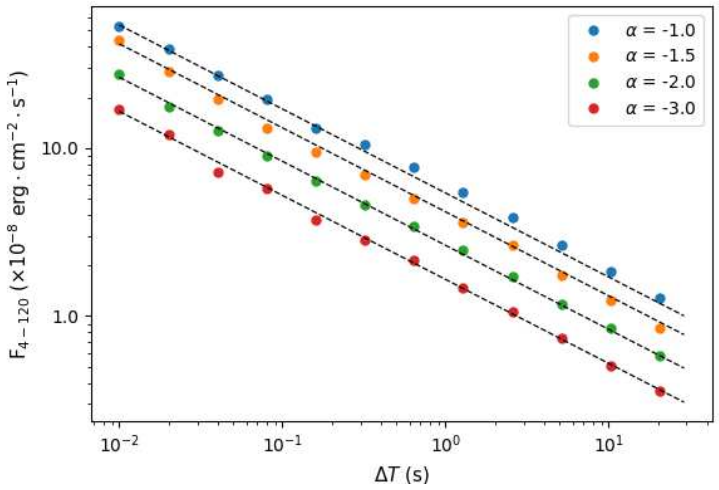
It is important to quantify how the sensitivity of the count rate trigger depends on the various GRB characteristics e.g. the duration, photon index and position in the FoV. We have carried out analyses to determine how the sensitivity depends on these parameters. We quantify the sensitivity by estimating the limiting on-axis flux (in units of erg cm^{-2} s^{-1}) required for 6.5σ detection in the 4 - 120 keV energy band.

Figure 5.15 show the limiting flux as function of GRB duration (Figure 5.15a), photon index (Figure 5.15b) and the off-axis angle (Figure 5.15c). For a GRB with fixed fluence, the limiting flux is expected to be proportional to the inverse of the square root of the GRB duration due to the definition of count-SNR (Eqn. 5.1). Figure 5.15a also shows that for a given duration, the limiting flux is lower for softer GRB (lower value of α_{PL}) compared to the harder GRB (higher value of α_{PL}) because softer GRBs have more photons in the 4 - 120 keV energy band. The limiting flux varies quadratically with the photon index and increases as the photon index increases (larger limiting flux for harder GRBs). The dependence of limiting flux on the off-axis angle is given by the inverse of the function which represents the convolution of the mask response and the $\cos\theta$ area projection factor (see section 4.3.3).

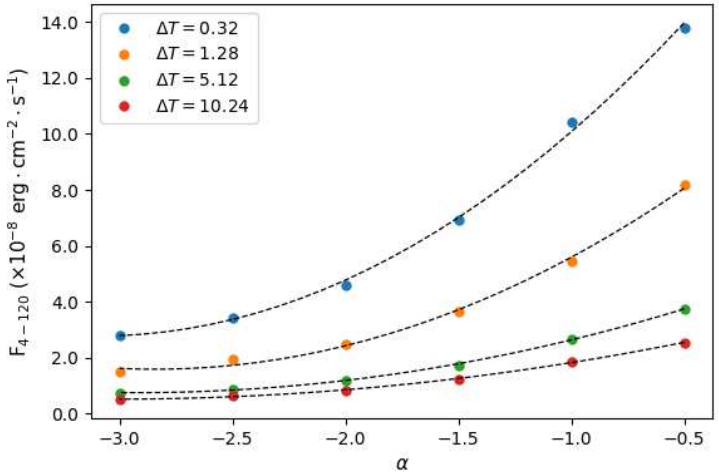
5.3 Offline trigger for GRB detection with ECLAIRs

ECLAIRs will transfer all the photons (i.e. all detected events with time, position (pixel) and energy information) to the ground. Therefore it is possible to search for GRBs (and other transients) using an "offline" or "ground" trigger. The offline trigger will benefit from a larger computational power and knowledge of the background variation before and after the transient search window and the context (e.g. space weather conditions). The offline trigger will be important for the following reasons,

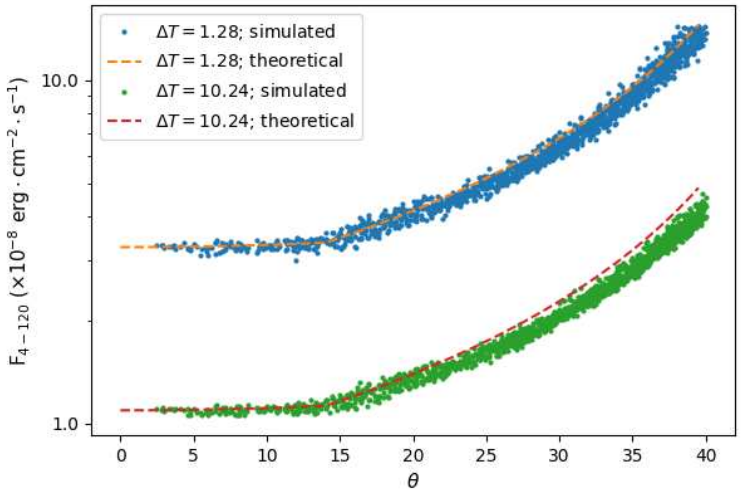
1. To detect GRBs (or transients) that are below the detection threshold of the onboard trigger (sub-threshold search).



(a)



(b)



(c)

Figure 5.15 – Dependence of the limiting flux (erg cm^{-2} s^{-1}) on the GRB properties: (a) Duration, (b) Photon index and (c) Off-axis angle. The dotted lines in (a) and (b) represent the fit to the data while the dotted lines in (c) represent the theoretical estimate of limiting flux computed by multiplying the mask response and the area projection factor $\cos\theta$.

2. To detect GRBs (or transients) when onboard trigger operation is ceased (e.g. during satellite slews, SAA ingress/egress phase)

3. To detect very long duration (spanning a duration of few orbits) transients such as ultra-long GRBs or TDEs.

4. To search for counterparts of the transients detected by other mission/facilities (e.g. *Swift*, *Fermi*, LSST, SKA etc.) in ECLAIRs data.

5. To detect transient events close to the Earth such as TGFs or solar flares (although ECLAIRs will point away from the Sun, solar flares are bright and energetic enough to penetrate the satellite material and can be detected even if the Sun is not in the FoV of ECLAIRs. And overall, ECLAIRs will detect their reflection on the Earth atmosphere).

6. Long term monitoring of data which will be important to optimise the parameters for the onboard trigger.

As part of the thesis, we test one possible algorithm which can be used as part of the offline trigger algorithms. The algorithm uses the count-rate data. In this section, I describe the algorithm, its implementation and compare its performance to the onboard count-rate trigger.

5.3.1 Haar wavelet detection algorithm

The wavelet transform is used to decompose a time-dependent signal into frequency components at different timescales to perform a time-frequency analysis. A typical transient signal usually exhibits variations across different timescales and the wavelet transform can be a promising tool to detect the transient signal on top of the background variation. The detection algorithm described here utilises the Haar wavelet (Haar, 1910) to decompose a timeseries data (in our case the detected counts per time bin). We expect that the transient signal is contained in the high frequency component (short timescales) and the background, which varies relatively slowly, is contained in the lower frequency component (long timescales). The choice of Haar wavelet is due to its simplicity and its support on a dyadic interval which keeps the difference between two successive timescales by a factor of two (a property similar to the onboard count-rate trigger). The decomposed "signal" component is subsequently analysed to detect an "excess". To decompose the timeseries into wavelet coefficients, we use the "à trous" algorithm (e.g. see Shensa, 1992; Starck et al., 2007). I will first describe some basic and relevant properties of the Haar wavelet transform and then describe the detection algorithm.

5.3.1.1 Wavelet transform properties

Consider a time series (or counts per time bin) data with a time resolution $\delta t^{(0)}$ which is the highest resolution. The data (i.e. counts in each time bin) at this level are then represented as $c^{(0)} = [c_0^{(0)}, c_1^{(0)}, c_2^{(0)}, ..., c_{N-1}^{(0)}]$ where N are the number of bins and 0 represents the current level ($l = 0$). Using the "à trous" algorithm, the data can be decomposed into a set of wavelet coefficients $c^{(L)}$ and $\mathcal{W} = \{w^{(1)}, w^{(2)}, ..., w^{(L)}\}$ where the $c^{(L)}$ are the average wavelet coefficients at the coarsest time resolution (level L) and $\mathcal{W}$ are the detail wavelet coefficients at intermediate levels. The progression from one level to next is given by:

$$c_i^{(l)} = \sum_j h_j^{(l)} c_{i+j}^{(l-1)} \tag{5.4}$$

$$w_i^{(l)} = \sum_j g_j^{(l)} c_{i+j}^{(l-1)} \tag{5.5}$$

where, $c^{(l)}$ represent the average (or smooth) coefficients at level l, $w^{(l)}$ represent the detail coefficients at level l. The h and g are low and high pass filter for the wavelet transform. For the Haar wavelet, the first level filters (i.e to go from $l = 0$ to $l = 1$) are given by, $h^{(1)} = [\frac{1}{2}, \frac{1}{2}]$ and $g^{(1)} = [-\frac{1}{2}, \frac{1}{2}]$. Following the "à trous" algorithm, to go to the next level, a zero is inserted between consecutive elements of the filter, e.g., $h^{(2)} = [\frac{1}{2}, 0, \frac{1}{2}, 0]$ and $g^{(2)} = [-\frac{1}{2}, 0, \frac{1}{2}, 0]$ and so on for the following levels. The original data can be reconstructed from the coarsest resolution level L by summing all the detail wavelet coefficients and the average coefficient at the level L (Eqn. 5.6). The wavelet decomposition of a sample time series is represented in the figure 5.16.

$$c_i^{(0)} = c_i^{(L)} + \sum_{k=1}^{L} w_i^{(k)} \tag{5.6}$$

An important property of the Haar wavelet coefficients is that in case of a purely "noisy" data (purely background counts in our case) the detail wavelet coefficients ($w^{(l)}$) at each level are symmetrically distributed with a zero mean. In particular for the Haar transform, the variance of the wavelet coefficients $\sigma^2_{w^{(l)}}$ at each level l is related to the variance of the original data ($\sigma^2_{c^{(0)}}$) by the following relation (see also figure 5.17):

$$\sigma^2_{w^{(l)}} = \sigma^2_{c^{(0)}} \times 2^{-l} \tag{5.7}$$

Furthermore, because the original data follow the Poisson statistics, the variance of the original data is equal to the mean of the data ($\bar{c}^{(0)}$). This makes it possible to determine the variance at any level by just using the statistics of the original data.

Additionally, because wavelet transform is a linear operation and due to the relation in Eqn. 5.7 the variance follows the relation:

$$\sigma^2_{c^{(0)}} = \sigma^2_{c^{(l)}} + \sum_{k=1}^{l} \sigma^2_{w^{(k)}} \tag{5.8}$$

Combining all these properties, we can see that distribution of wavelet coefficients for purely background counts is mostly constrained within a fixed value of σ for each

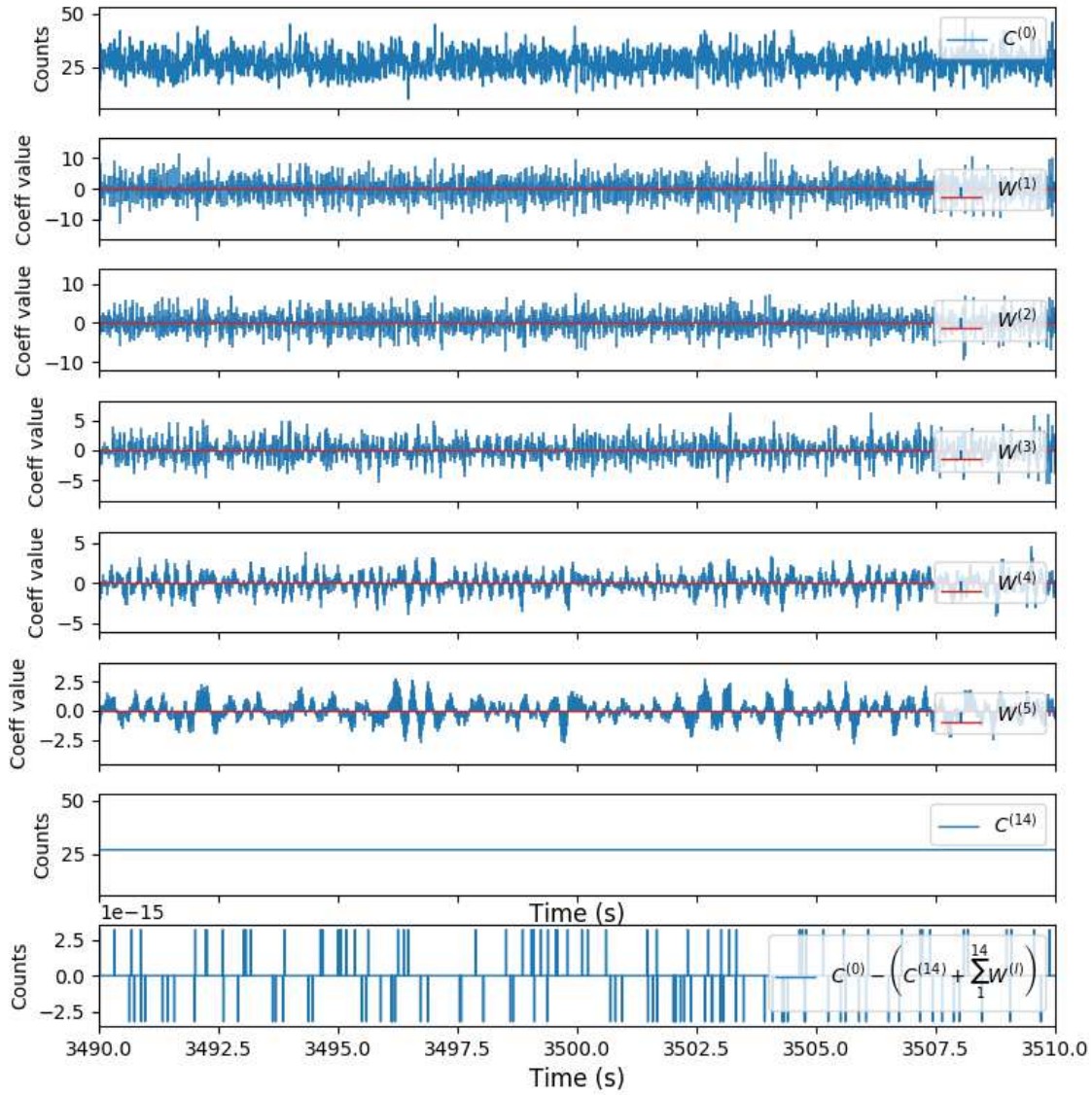


Figure 5.16 – Example of wavelet decomposition using the à trous algorithm. The topmost panel shows the original time series data which is then subsequently decomposed into $L = 14$ levels. For representation, only first five, i.e $W^{(1)}$ to $W^{(5)}$ detail coefficients (panel 2 to 6) are shown. The average coefficients at the final level $C^{(14)}$ is shown in panel 7. The summation of the wavelet coefficients subtracted from the original data is shown in the last panel. As excepted, this value is almost zero except for the small spikes which are due to the error in the floating point precision.

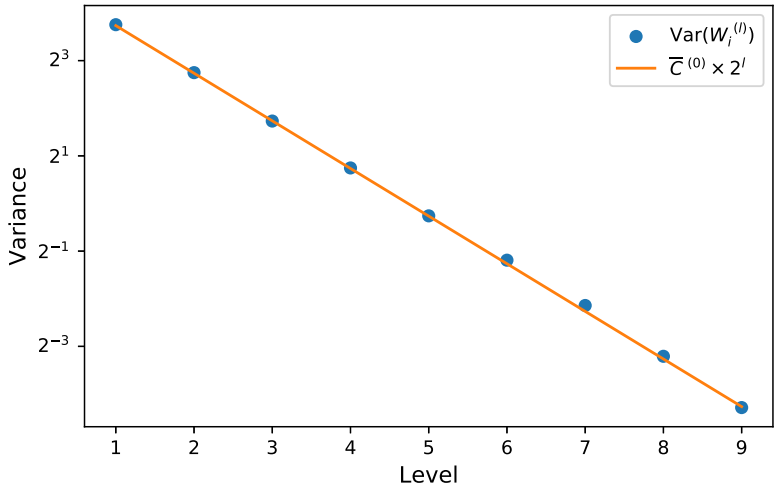


Figure 5.17 – Variation of the variance of the detail coefficients against the decomposition level. The blue points show the variance computed from the coefficients obtained at each level after decomposition while the orange line indicates the predicted variance following the Eqn. 5.7.

level (Figure 5.18a and 5.18b). Following the figure 5.18 and to ease the discussion below, we define the coefficient significance ($\epsilon_i^{(l)}$) which represent the detail coefficients normalised to their standard deviation ($\sigma_{w^{(l)}}$), i.e.:

$$\epsilon_i^{(l)} = \frac{w_i^{(l)}}{\sigma_{w^{(l)}}} \tag{5.9}$$

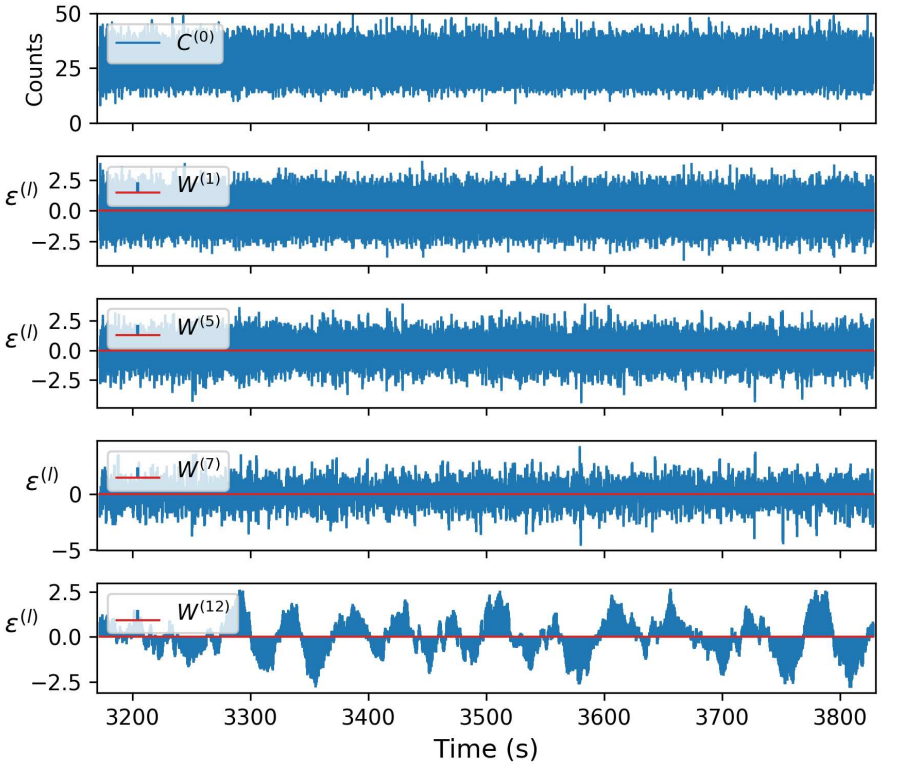
5.3.1.2 Detection algorithm

The detection of an "excess" in the data is done in two steps. First, we detect the excess at each level based on the distribution of the wavelet coefficients at that level. This is possible because, if the data contains a GRB (or transient) signal, the wavelet coefficient distribution in the figure 5.18 is modified. In such a case, sharp peaks are observed in the wavelet coefficients (Figure 5.19a) for levels at which the GRB signal is present. The coefficient significance distribution [5] (Figure 5.19) at these levels usually has outliers at large σ value compared to data containing only background.

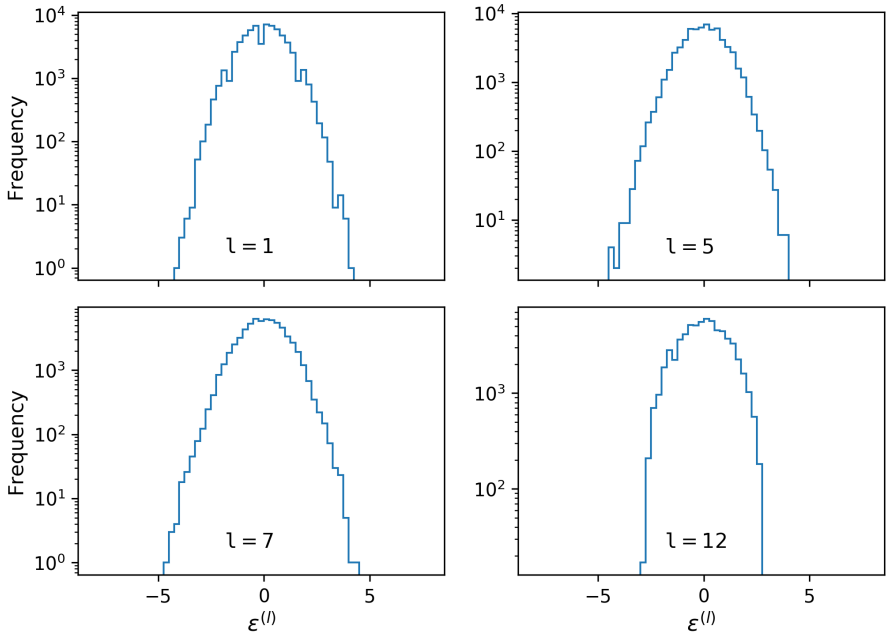
To detect the presence of a GRB signal, we first set a 5σ threshold [6] for each level

5. Note: In the case of data containing a GRB, to compute the variance of each scale (Eqn. 5.7) we assume that we have a sufficiently long timeseries and that there is no signal in the data other than the GRB. Additionally, we assume that the GRB is not located at the beginning or at the end of the timeseries. The $\sigma^2_{c^{(0)}}$ in Eqn. 5.7 is then computed by taking the mean of the counts in a fixed interval at the start and end of the timeseries.

6. This is chosen because the coefficient significance for data containing only the background is usually constrained within 5σ at each level, e.g. see Figure 5.18

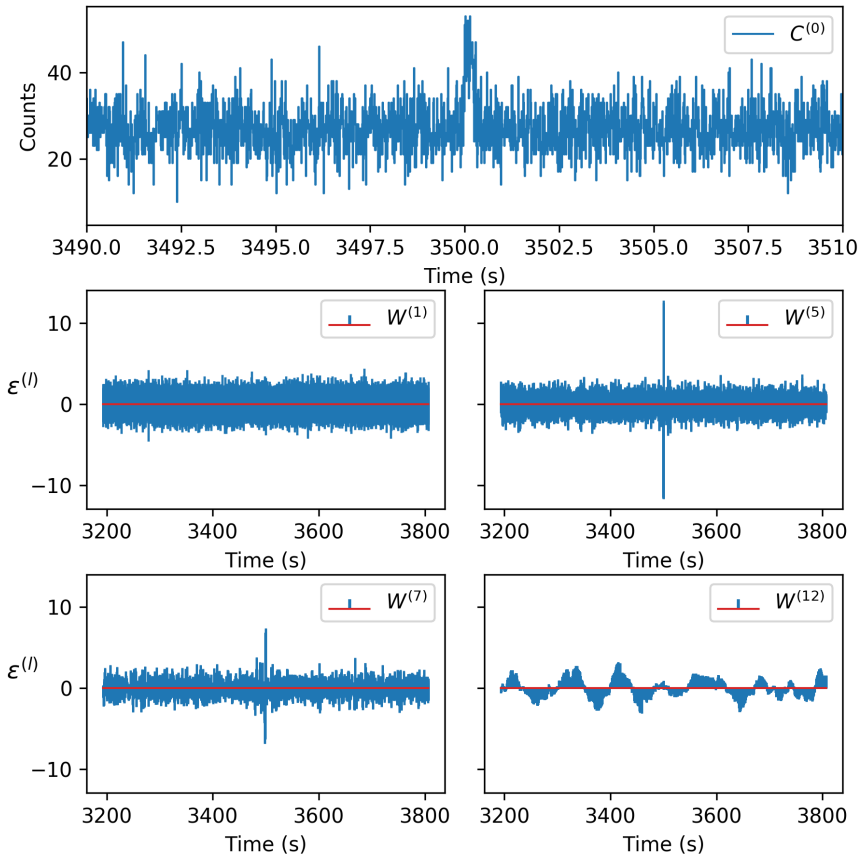


(a)

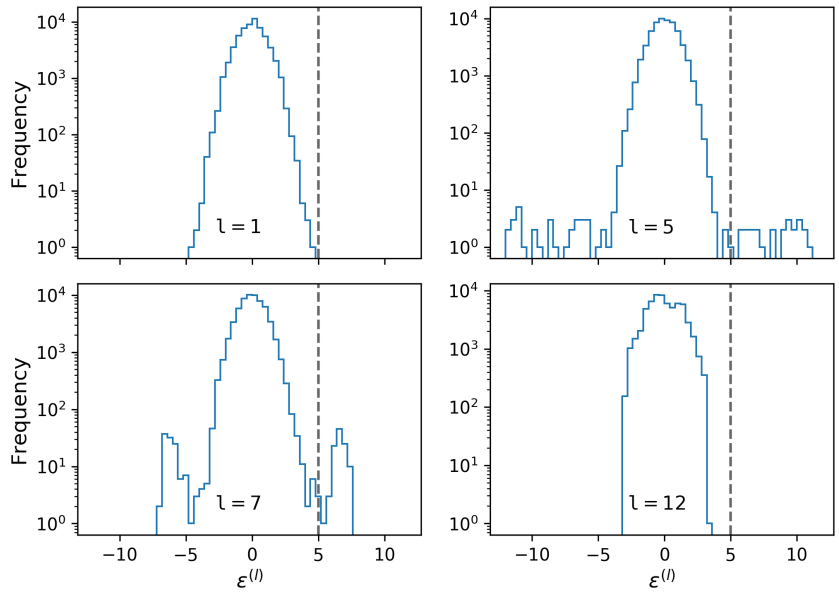


(b)

Figure 5.18 – Detail wavelet coefficients at different scales normalised (see Eqn. 5.9) to the standard deviation (σ^l) of each scale for background only data. Figure (a) shows the time variation of the coefficients at selected scales (panel 2 to 4) along with the original data (topmost panel). Figure (b) shows the histogram of the coefficients for the scales in figure (a). Note: The total duration of the data is 600 s, but for figure (a) only part is shown as an illustration. The minimum time resolution used to bin data is 0.01 s.



(a)



(b)

Figure 5.19 – Coefficient significance distribution for data with a synthetic GRB (0.32 s duration). Figure (a) shows the time variation of the coefficient significance at selected scales (panel 2 to 4) along with the original data (topmost panel). Sharp peaks can be seen at some levels at the position of the GRB. Figure (b) shows the histogram of the coefficients for the scales in figure (a). Large values (up to 10σ) compared to the purely background distribution (Figure 5.18) are seen in two levels.

and assume that any coefficient with significance above this threshold is due to the GRB. We select the maximum coefficient significance (max $\epsilon^{(l)}$) at each level and find the global maximum, i.e. the maximum coefficient significance among all levels [7]. The level at which the maximum is obtained is called as the l_{max} (Figure 5.20).

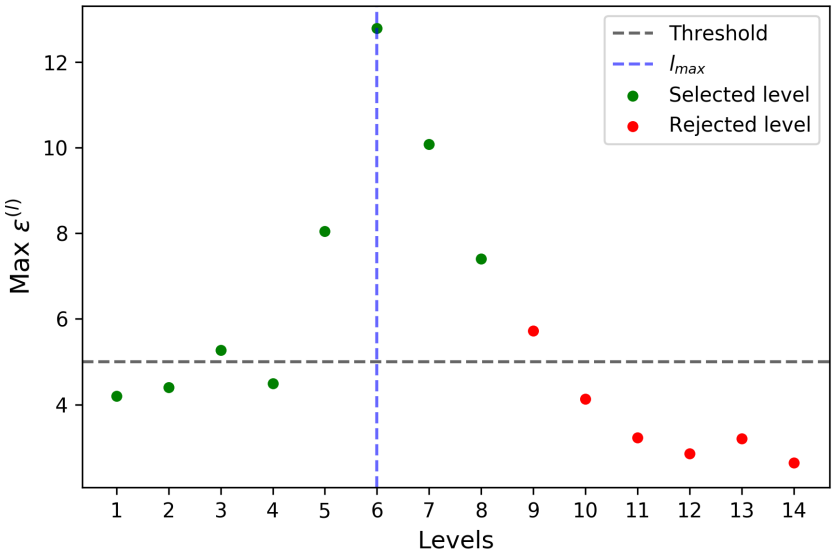


Figure 5.20 – Maximum coefficient significance at each level for the GRB in figure 5.19a. The global maximum of the coefficient significance is obtained at level $l_{max} = 6$ is shown by the vertical dotted line. The horizontal dotted line indicated the threshold used for identifying the detection (5σ in this case). All the wavelet coefficients of the levels coloured Green are used to construct the GRB signal following Eqn. 5.10.

After the detection, to define the SNR for an excess, we use the following method: We consider that all the levels up to the level $l_{max} + 2$ contain the contribution from the GRB. All the remaining levels are considered to contain only the background. The choice of adding the contribution from 2 additional levels above the l_{max} is a conservative compromise between gaining signal and rejecting the background. With this assumption, we first define a total excess signal in the following way (see figure 5.21a middle panel):

$$s_i = \sum_{1}^{l_{max}+2} w_i^{(k)} \tag{5.10}$$

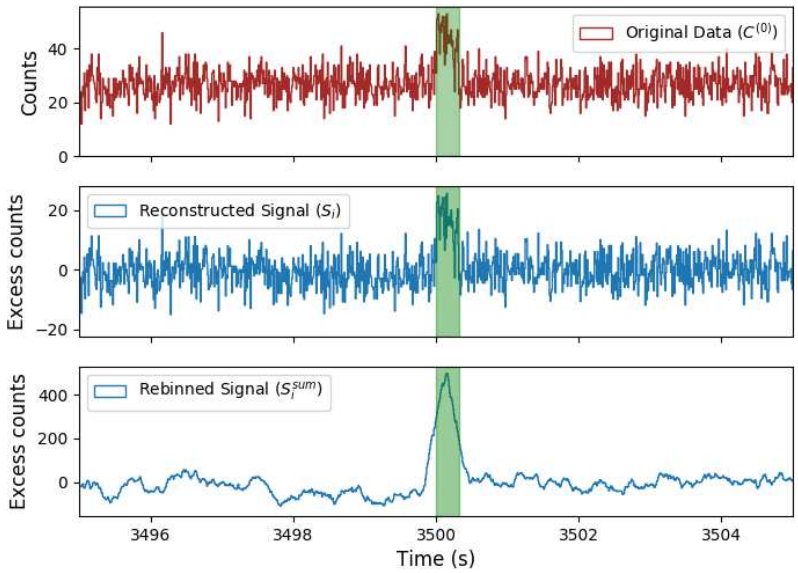
This signal definition has time resolution equivalent to the original data (in our case 0.01 s). We analyse this signal further at timescales similar to the count-rate trigger (i.e. 0.01 s to 20.48 s). For each timescale the signal is rebinned to compute the SNR. The timescales are related to a level l as $\delta t^{(0)} \times 2^{l-1}$ where $\delta t^{(0)}$ is the timescale (or

7. Note: To find the maximum value we only analyse the positive values of the coefficient significance as the positive value are located within the GRB duration.

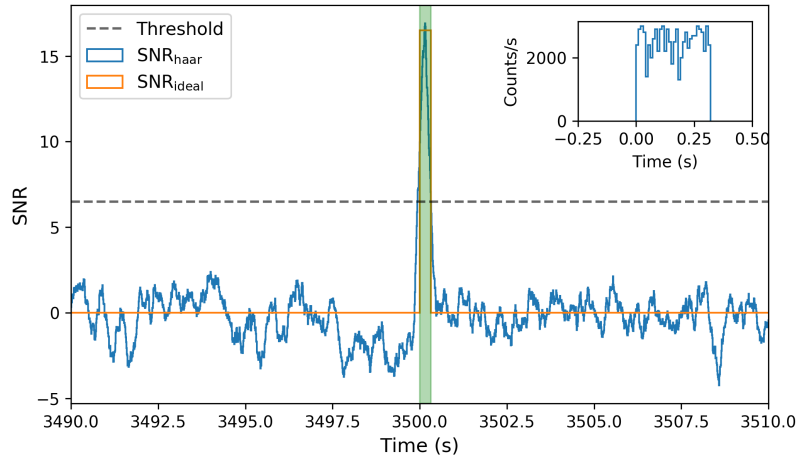
time resolution) of the original data. To do the rebinning we use a moving sum filter, the rebinned signal is represented as (see figure 5.21a bottom panel):

$$s_i^{sum} = \sum_{j=-\frac{M}{2}}^{\frac{M}{2}-1} s_{i+j} \tag{5.11}$$

where the length of the filter is $M = 2^{l-1}$. The SNR is then defined by normalising



(a) Haar signal time series



(b) Haar-SNR time series

Figure 5.21 – A figure showing the Haar wavelet signal and SNR computation for a sample GRB. The duration of the GRB is 0.32 s which is marked by the shaded Green area in all cases. Figure (a): The first (from top) panel shows the original data (Red) binned at highest resolution timescale ($\delta t^{(0)} = 0.01s$). The second panel shows the signal computed using Eqn 5.10. The third panel shows the signal in the second panel binned at the GRB timescale using Eqn. 5.11. Figure (b): The Haar-SNR and ideal-SNR computed at the GRB timescale. Both the SNRs are comparable and match in terms of location in time. The dotted horizontal line indicates the threshold used to verify the detection.

the rebinned signal with the total variance of all scales i.e. variance of the total background ($\sigma^2_{c(0)}$). This normalisation makes SNR comparable to the definition of ideal-SNR where the signal excess is compared to the background variance. The Haar-SNR ($\mathrm{SNR_{haar}}$) is then defined as:

$$\mathrm{SNR_{haar,i}} = \frac{s_i^{sum}}{\sqrt{M} \times \sigma_{c(0)}} \tag{5.12}$$

After computing the Haar-SNR, the detection is validated only if the maximum value of Haar-SNR is above a 6.5σ threshold [8].

The figure 5.21b shows the Haar-SNR timeseries for the timescale corresponding to the GRB duration. The location of the maximum Haar-SNR is within the GRB duration and it is consistent with the ideal-SNR. The time (t) at which the maximum Haar-SNR is obtained is considered as the trigger time of the burst and the start and end time of the signal is given by $t \pm \delta t/2$ respectively, where δt is the timescale at which the Haar-SNR is computed. This interval can be used as a reference point for further analysis (e.g. imaging to identify the GRB location in the sky).

For verification, we compare the Haar-SNR computed at different timescales with the ideal-SNR (see Eqn. 5.2) at the same timescale as shown in the figure 5.22. It shows that usually the $\mathrm{SNR_{haar}}$ is consistent with the ideal-SNR for all timescales and the global maximum is obtained at the timescale corresponding to the GRB duration.

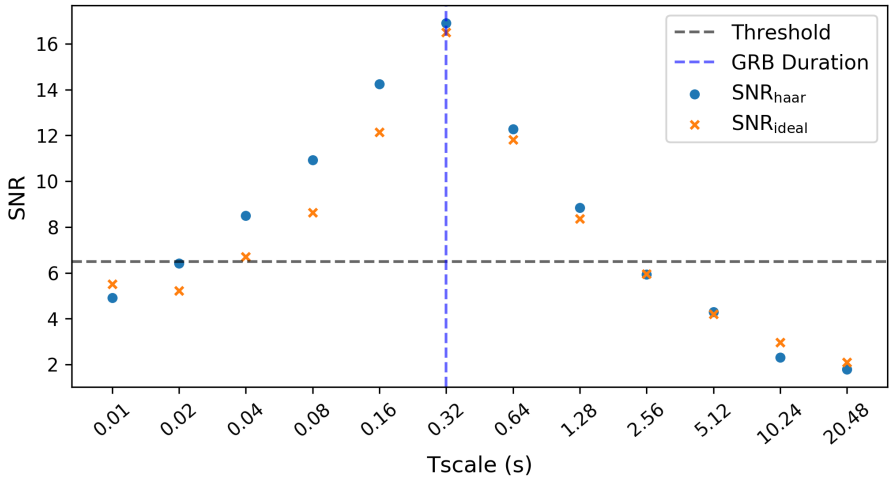
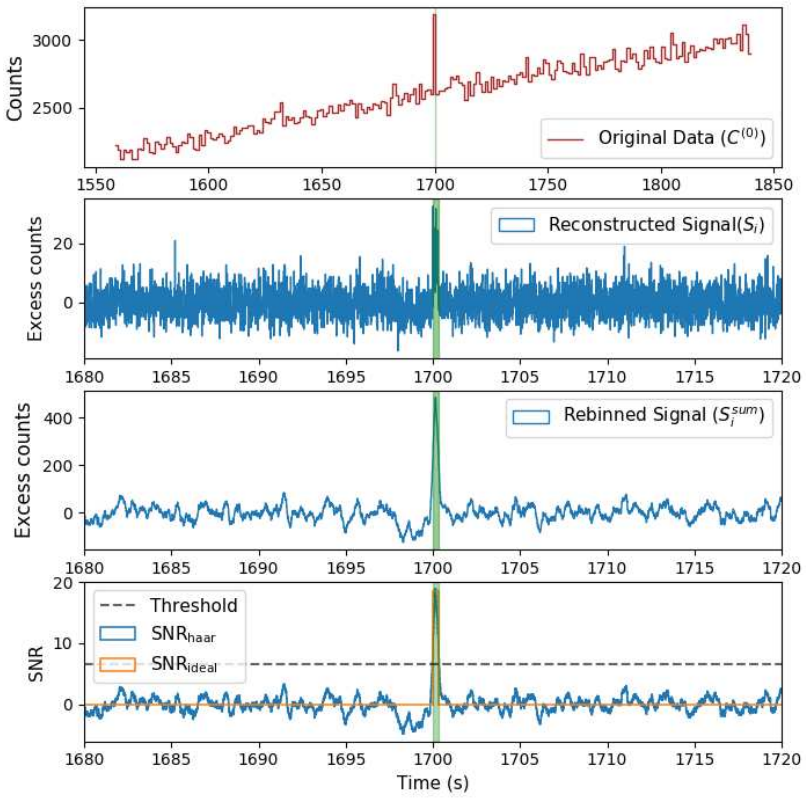
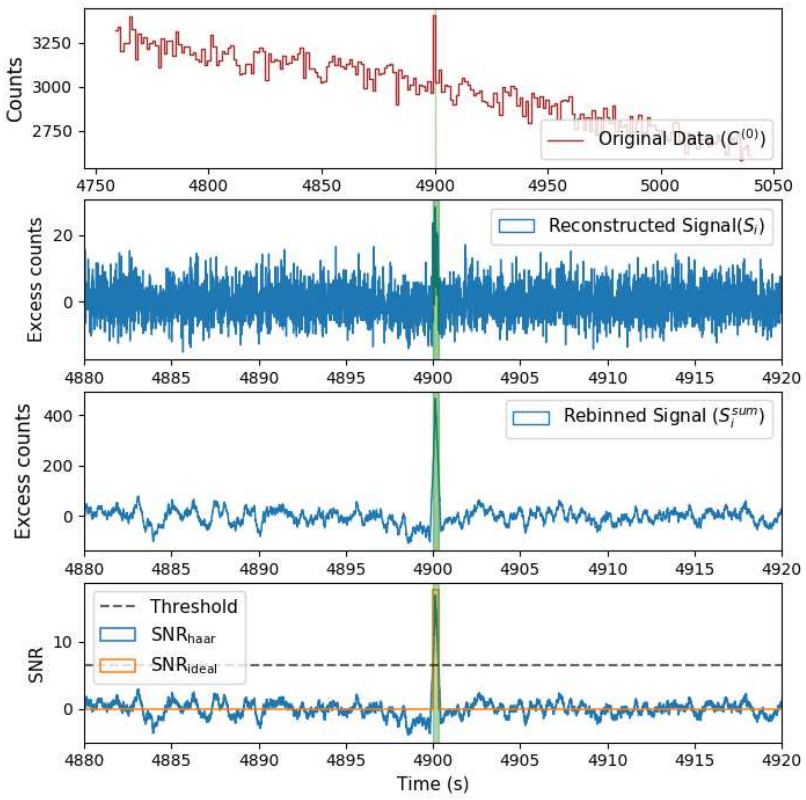


Figure 5.22 – Maximum Haar-SNR against the timescale for the GRB in figure 5.21. The global maximum is obtained at the timescale corresponding to the GRB duration (indicated by the dotted vertical line). The detection is validated only if the maximum SNR at a given timescale is above the 6.5σ threshold (marked by the dotted horizontal line)

8. This is currently chosen to be same as the count-rate trigger threshold (A detailed analysis of statistics of the SNR must be done to see if this value can be lowered)



(a)



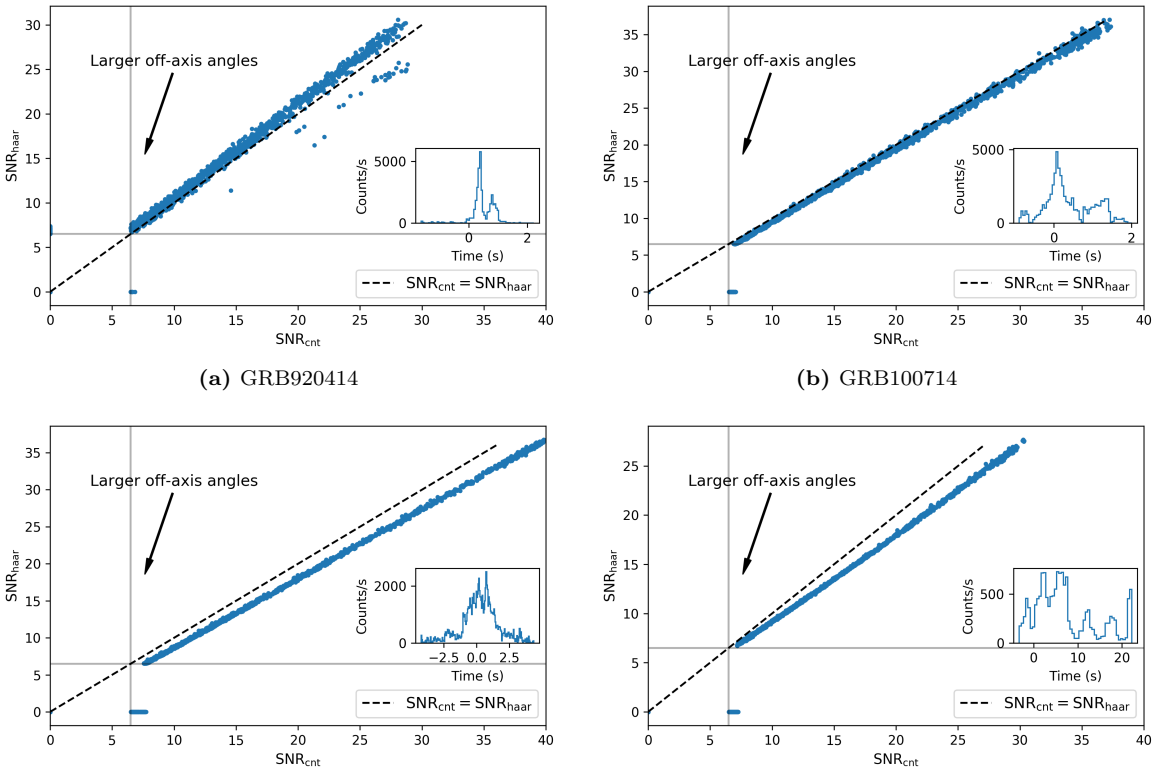
(b)

Figure 5.23 – Haar wavelet trigger verification for (a) increasing and (b) decreasing background count phase.

We also check the validity of the method for the increasing and decreasing background phase of the orbit. Figure 5.23a and 5.23b show the signal reconstruction and comparison of the Haar-SNR with the ideal-SNR for the increasing and decreasing background phase respectively. In both cases, the Haar-SNR agrees with the ideal-SNR reasonably well. The potential of using wavelet transform is highlighted here as no background model fitting is required even in the case of variable background count-rate.

5.3.1.3 Performance of Haar trigger and perspectives

To evaluate the performance of the Haar wavelet trigger algorithm, we compare the Haar-SNR with the count-SNR. For this, we simulate GRBs at various positions in the sky to obtain different observed intensities. In case of the Haar wavelet trigger algorithm, we take the minimum timescale, $\delta t^{(0)} = 0.01$ s and we evaluate the Haar SNR up to $l = 12$, i.e $\delta t^{(l)} = 20.48$ s. This is exactly the same as the count-rate trigger timescales and eases the comparison.



(a) GRB920414 **(b)** GRB100714

(c) GRB050922C **(d)** GRB011130

Figure 5.24 – Comparison of Haar-SNR and the count-SNR for four real GRBs (top pane: short GRBs, bottom panel: long GRBs). The horizontal and vertical grey line indicate the threshold use for the detection for each method. The dashed black line indicate the equality between the two SNRs. The on-axis lightcurve of each GRB is shown in the inset plot.

The distribution of Haar-SNR and count-SNR for four real GRBs (two short and two long) is shown in the figure 5.24. For all GRBs analysed here, the Haar-SNR behaves similarly to the count-SNR. However, for the given GRB intensity, there is a discrepancy in the two SNR values which is due to the fact that our signal reconstruction (Eqn. 5.10) is not perfect. A better way to reconstruct the signal is currently under investigation.

In conclusion, we demonstrate the effectiveness of the Haar trigger, keeping in mind that its first goal is to provide the location in time and duration of GRB-like events for further analysis. The method is promising, however, the current study concentrates on the method itself and we need to perform more investigations, in particular, studies with faintest GRBs, evaluation of true and false detections, GRBs located in different background phases (Figure 5.4) to conclusively prove the reliability and the sensitivity of the method.

Conclusions [English]

The upcoming Sino-French astrophysics mission - *SVOM* (currently planned to be launched in June 2022), is dedicated to the study of Gamma-ray Bursts (GRBs) and other high-energy transients. Onboard *SVOM*, the ECLAIRs instrument, a hard X-ray wide-field coded-mask imager will detect and localise the high-energy transients in real-time using an onboard trigger. Furthermore, ECLAIRs will have an "offline" trigger which will search for transients on the ground after the data is available there. This PhD thesis is focused on the simulation of ECLAIRs data and the evaluation of the performance of the ECLAIRs triggers using the simulated data.

A wide-field instrument like ECLAIRs suffers from a high background level which affects its sensitivity. Additionally, due to the pointing strategy of *SVOM*, the Earth will pass through the ECLAIRs field-of-view (FoV) modulating the observed background. We have developed a method (named Particle Interaction Recycling Approach or PIRA), based on the Monte-Carlo GEANT4 simulations, which can simulate the variation of the background accurately and rapidly. PIRA reuses particle interaction information stored in a large pre-computed GEANT4 database to estimate the background for any orbital configuration. The use of GEANT4 ensures the accuracy and the reusing of interactions increases the computation speed. Additionally, the algorithms used as part of PIRA can be extended to any high-energy instrument provided the input database is computed for the instrument.

In the thesis, we have described and verified the PIRA algorithms using ECLAIRs as a test case. We illustrate the algorithms in the case of the four most dominant background components affecting ECLAIRs, namely, the cosmic X-ray background (CXB), the reflection of CXB from the Earth's atmosphere (Reflection), the hard X-ray emission from the Earth's atmosphere (Albedo) and the charged particle background in the South Atlantic Anomaly (SAA). We have also shown that PIRA is $10^3 - 10^4$ times computationally more efficient than the direct GEANT4 simulations. This work has been published in a refereed journal (Mate et al., 2019).

Using PIRA, we simulate background events observed by ECLAIRs as a function of the *SVOM* orbit. The background events are then used to simulate synthetic data as close as possible to the real data by adding simulations of X-ray sources and GRBs. As part of the thesis, a simulation toolkit has been created to simulate ECLAIRs data

which has been shared with the *SVOM* collaboration. The simulated data have been used to carry out technical as well as scientific performance studies of ECLAIRs. We use the simulated data to assess the impact of the variable background on the onboard trigger performance and to test the basic functioning of an algorithm that can be used as part of the offline trigger.

Concerning the assessment of the performance of the onboard trigger, the study is limited to the count-rate part of the onboard trigger and for the observing scenarios where no X-ray sources are present in the field of view of ECLAIRs. The count-SNR (Signal to Noise Ratio) is used as a reference parameter for our study. In a first analysis, we evaluate the impact of the background on the count-rate trigger and its GRB detection capabilities. Based on the statistics of the background we determine the count-SNR (in σ) threshold to be used onboard to distinguish GRBs (or transient signal) from the background. Given the properties of the background obtained from the simulation, we estimate a threshold of 5σ. However, because some background components are currently not taken into account in PIRA, we have used a slightly higher SNR threshold of 6.5σ (this value is chosen to be the same as the image-SNR threshold determined by Gros, 2016a). We use this threshold to determine the sensitivity of the count-rate trigger for a short (1 s) and a long (10 s) GRB triggering at different locations (in time) on the orbit. The sensitivity is quantified with two parameters: the minimum required count-rate (cts/s) and the minimum detectable on-axis flux (erg cm^{-2} s^{-1}).

We have assessed the impact of trigger parameters (energy band, timescale, image SNR etc.) on the count-rate SNR. This analysis is carried out using a sample of synthetic GRBs with known spectral and temporal properties and a sample of real GRBs. We find that, for a given GRB, the energy band in which the count-rate SNR is maximum depends on the spectral shape of the GRB. The timescale at which the maximum count-rate SNR is obtained depends on the temporal features of the GRB (e.g. peaks). Both results are consistent with the expectations.

Furthermore, we have evaluated the sensitivity of the trigger to the GRB properties. The sensitivity is determined by evaluating the minimum detectable on-axis flux as a function of the GRB timescale, spectral shape (power-law with different photon index) and its position in the FoV. For a GRB with a given duration, the minimum detectable flux increases quadratically as the photon index increases (i.e. transition from softer to harder spectrum). The variation of the minimum detectable flux with respect to the GRB position in the FoV is consistent with the expected theoretical calculation done using the geometrical response of the coded mask.

Finally, we have discussed the "offline" trigger for ECLAIRs, which analyses the data on the ground. The offline trigger will benefit from the larger computational power and better knowledge of the context of the transient search (e.g. knowledge of

space weather, sources in the FoV, etc.). It will be used to detect transients missed by the onboard trigger (e.g events below the onboard detection threshold, events during a temporary shutdown) and also to find transients reported by other missions (e.g. *Swift*, *Fermi*) or facilities (e.g. LIGO/VIRGO events, LSST, SKA) in ECLAIRs data. We have proposed an algorithm that can be used as part of the offline trigger. The algorithm is based on the wavelet transform of the count-rate signal. We have explained and tested the algorithm using the simulated data.

Conclusions [Française]

La mission spatiale sino-française *SVOM* (dont le lancement est prévu en juin 2022), est consacrée à l'étude des sursauts gamma (GRBs) et autres sources transitoires de haute énergie. À bord du satellite *SVOM*, l'instrument ECLAIRs, un imageur-rayons-X à masque codé à large champ de vue, jouera un rôle essentiel pour détecter et localiser les sursauts gamma et les sources transitoires de haute énergie. La détection et la localisation se feront en temps quasi réel à l'aide d'un « trigger » embarqué. En outre, les données de ECLAIRs seront analysées à l'aide d'un « trigger » sol qui recherchera les sources qui auront échappé au « trigger » embarqué. Cette thèse de doctorat est axée sur la simulation des données ECLAIRs et l'évaluation des performances des différents « trigger » ECLAIRs.

Un instrument à grand champ de vue comme ECLAIRs souffre d'un niveau de bruit de fond élevé qui affecte la sensibilité de l'instrument. De plus, en raison de la stratégie de pointage de *SVOM*, le bruit de fond dans ECLAIRs est variable car la Terre passe dans le champ de vue toutes les 96 minutes. Nous avons développé une méthode, appelée « Particle Interaction Recycling Approach » (ou PIRA), basée sur des simulations de Monte-Carlo, qui permet de simuler le fond dynamique avec précision et rapidité. PIRA réutilise les interactions de particules stockées dans une grande base de données GEANT4 pré-calculée pour estimer le bruit de fond pour n'importe quelle configuration orbitale. L'utilisation de GEANT4 assure la précision et la réutilisation des interactions augmente la vitesse de calcul. En outre, les algorithmes utilisés dans le cadre de PIRA peuvent être étendus à tout instrument à haute énergie.

Dans le cadre de la thèse, nous avons développé les algorithmes PIRA dans le cadre de l'instrument ECLAIRs. Nous avons développé les algorithmes dans le cas de quatre composantes du bruit de fond qui affectent principalement ECLAIRs, à savoir, le bruit de fond des rayons X cosmiques (CXB), la réflexion du CXB par l'atmosphère terrestre (Reflection), l'émission de rayons X durs de l'atmosphère terrestre (Albedo) et le bruit de fond des particules chargées dans l'anomalie de l'Atlantique Sud (SAA). Pour chaque composante, nous avons vérifié PIRA en comparant le bruit de fond dans un cas statique obtenu à l'aide de PIRA et en utilisant une simulation directe de GEANT4. Nous avons également montré que PIRA est $10^3 - 10^4$ fois plus rapide que les simulations directes de GEANT4. Ce travail a été publié dans une journal

scientifique.

Avec PIRA, nous simulons les événements de bruit du fond observés par ECLAIRs en fonction de l'orbite *SVOM*. Les événements de bruit de fond sont ensuite utilisés pour simuler des données pour des cas d'observation quasi-réels en ajoutant des simulations de sources de rayons X et de GRB. Dans le cadre de la thèse, une boîte à outils a été créée pour simuler les données d'ECLAIRs, qui a été partagée avec la collaboration *SVOM*. Les données simulées ont été utilisées pour réaliser des études techniques et scientifiques sur les performances d'ECLAIRs. Ainsi, nous utilisons les données simulées pour évaluer l'impact de la bruit de fond sur les performances du « trigger » embarqué et pour tester le fonctionnement de base d'un algorithme qui peut être utilisé dans le cadre du « trigger » sol.

En ce qui concerne l'évaluation des performances du « trigger » embarqué, l'étude se limite à la partie du « trigger » embarqué relative au taux de comptage et aux scénarios d'observation dans lesquels aucune source de rayons X n'est présente dans le champ de vue des ECLAIRs. Le rapport signal/bruit (SNR) du « trigger » taux de comptage a été utilisé comme paramètre de référence pour notre étude. Dans une première analyse, nous évaluons l'impact du bruit de fond sur la capacités de détection des GRB par le « trigger » embarqué. Sur la base des statistiques du bruit de fond, nous déterminons le seuil du taux de comptage SNR (en σ) à utiliser à bord pour distinguer les GRBs du bruit de fond. Nous concluons que, compte tenu des propriétés du bruit de fond obtenues par simulation et d'éventuelles systématiques instrumentales non prises en compte dans les simulations PIRA, la valeur SNR de $6{,}5\sigma$ peut être utilisée sans risque comme seuil à bord. Nous avons utilisé cette valeur pour déterminer la sensibilité du « trigger » du taux de comptage pour un sursaut court (1 s) et un sursaut long (10 s) détectés à différentes position sur l'orbite. La sensibilité est quantifiée à l'aide de deux paramètres : le taux de comptage minimum requis (cts/s) et le flux minimum détectable sur l'axe (erg cm^{-2} s^{-1}).

Nous avons évalué l'impact des paramètres de déclenchement (bande d'énergie, échelle de temps, SNR de l'image, etc.) sur le SNR du taux de comptage. Cette analyse est réalisée à l'aide d'un échantillon des GRBs synthétiques ayant des propriétés spectrales et temporelles connues et d'un échantillon de GRBs réel. Nous concluons que, pour un GRB donné, la bande d'énergie dans laquelle le taux de comptage SNR est maximal dépend de la forme spectrale du GRB. L'échelle de temps à laquelle le taux de comptage SNR maximum est obtenu dépend des caractéristiques temporelles du GRB (par exemple, les pics). Notre analyse montre que le SNR taux de comptage et le SNR de l'image correspondant sont comparables pour des échelles de temps plus longues (> 10 s), mais s'écartent significativement de l'égalité pour des échelles de temps plus courtes.

En outre, nous avons évalué la sensibilité du « trigger » aux propriétés du GRB

en déterminant le flux minimum détectable sur l'axe en fonction de l'échelle de temps du GRB, de sa forme spectrale et de sa position dans le champ de vue. Nous avons conclu que le flux limite diminue comme la racine carrée de la durée du sursaut, un comportement attendu basé sur la définition du SNR taux de comptage. Pour un GRB de durée donnée, le flux limite augmente de façon quadratique à mesure que l'indice photonique augmente (c'est-à-dire lors de la transition d'un spectre plus doux à un spectre plus dur). La variation du flux limite par rapport à la position du GRB dans le FoV est cohérente avec le calcul théorique effectué à l'aide de la réponse angulaire du masque.

Enfin, nous avons discuté du « trigger » sol pour ECLAIRs, qui effectue une analyse plus poussée des données au sol. Le « trigger » sol bénéficiera d'une plus grande puissance de calcul et d'une meilleure connaissance du contexte de la recherche transitoire (par exemple, connaissance du contexte, des sources dans le champ de vue, etc.). Il permettra de détecter les transitoires manqués par le « trigger » embarqué (par exemple, les événements pendant l'arrêt temporaire du « trigger » embarqué, les événements en dessous du seuil, etc.). Nous avons proposé un algorithme qui peut être utilisé dans le cadre du « trigger » sol. L'algorithme est basé sur la transformée en ondelettes du signal. Nous avons testé l'algorithme en utilisant des données simulées.

Bibliography

Abbott, B. P., Abbott, R., Abbott, T. D., et al. 2017a, Astrophys. J., 848, L12, doi: 10.3847/2041-8213/aa91c9

Abbott, B. P., et al. 2017b, doi: 10.1103/PhysRevLett.119.161101

Abbott, B. P., Abbott, R., Abbott, T. D., et al. 2017c, Astrophys. J., 848, L13, doi: 10.3847/2041-8213/aa920c

Acciari, V. A., Ansoldi, S., Antonelli, L. A., et al. 2019, Nature, 575, 455, doi: 10.1038/s41586-019-1750-x

Ackermann, M., Asano, K., Atwood, W. B., et al. 2010, Astrophys. J., 716, 1178, doi: 10.1088/0004-637X/716/2/1178

Agostinelli, S., Allison, J., Amako, K., et al. 2003, Nucl. Instruments Methods Phys. Res. A, 506, 250, doi: 10.1016/S0168-9002(03)01368-8

Ajello, M., Greiner, J., Sato, G., et al. 2008, Astrophys. J., 689, 666, doi: 10.1086/592595

Amati, L., Frontera, F., Tavani, M., et al. 2002, Astron. Astrophys., 390, 81, doi: 10.1051/0004-6361:20020722

Amati, L., O'Brien, P., Götz, D., et al. 2018, Adv. Sp. Res., 62, 191, doi: 10.1016/j.asr.2018.03.010

Angel, J. R. P. 1979, Astrophys. J., 233, 364

Antier, S. 2016, PhD thesis, Université Paris-Saclay

Arcier, B., Atteia, J.-L., & Pons, R. 2020a, Simulation des taux de comptages attendus en vol pour les pixels de SVOM / ECLAIRs, Tech. rep., IRAP

Arcier, B., Bouchet, L., Godet, O., & Atteia, J.-L. 2020b, Calcul d 'efficacité à partir des données EQM ELS2, Tech. rep., IRAP

Arcier, B., Mate, S., Bouchet, L., & Maestre, S. 2020c, Simulation d ' évènements ECLAIRs avec ELSimulator, Tech. rep.

Arcier, B., Atteia, J. L., Godet, O., et al. 2020d, Astrophys. Space Sci., 365, doi: 10.1007/s10509-020-03898-z

Atteia, J.-L., Schanne, S., Cordier, B., & Guillemot, P. 2018, ECLAIRs Data Rates, Tech. rep.

Atteia, J.-L., Boer, M., Cotin, F., et al. 2002, Gamma-Ray Burst Afterglow Astron. 2001 A Work. Celebr. First Year HETE Mission, 662, 17, doi: 10.1063/1.1579292

Atteia, J. L., Bouchet, L., Claret, A., et al. 2016, ECLAIRs Science Requirements Document – V2, Tech. rep.

Atwood, W. B., Abdo, A. A., Ackermann, M., et al. 2009, Astrophys. J., 697, 1071, doi: 10.1088/0004-637X/697/2/1071

Band, D., Matteson, J., Ford, L., et al. 1993, Astrophys. J., 413, 281, doi: 10.1086/172995

Barraud, C., Olive, J.-F., Lestrade, J. P., et al. 2003, Astron. Astrophys., 400, 1021, doi: 10.1051/0004-6361:20030074

Barthelmy, S. D., Barbier, L. M., Cummings, J. R., et al. 2005, Space Sci. Rev., 120, 143, doi: 10.1007/s11214-005-5096-3

Berger, E. 2014, Annu. Rev. Astron. Astrophys., 52, 43, doi: 10.1146/annurev-astro-081913-035926

Berger, E., Kulkarni, S. R., Fox, D. B., et al. 2005, Astrophys. J., 634, 501, doi: 10.1086/491667

Bernardini, M. G., Xie, F., Sizun, P., et al. 2017, Exp. Astron., 44, 113, doi: 10.1007/s10686-017-9551-4

Bessell, M. S. 2005, Annu. Rev. Astron. Astrophys., 43, 293, doi: 10.1146/annurev.astro.41.082801.100251

Bhalerao, V., Kasliwal, M. M., Bhattacharya, D., et al. 2017, Astrophys. J., 845, 152, doi: 10.3847/1538-4357/aa81d2

Bloom, J. S., Giannios, D., Metzger, B. D., et al. 2011, Science (80-.)., 333, 203, doi: 10.1126/science.1207150

Boella, G., Butler, R. C., Perola, G. C., et al. 1997a, Astron. Astrophys. Suppl. Ser., 122, 299, doi: 10.1051/aas:1997136

Boella, G., Chiappetti, L., Conti, G., et al. 1997b, Astron. Astrophys. Suppl. Ser., 122, 327, doi: 10.1051/aas:1997138

Bošnjak, Ž., Götz, D., Bouchet, L., Schanne, S., & Cordier, B. 2014, A&A, 561, 25, doi: 10.1051/0004-6361/201322256

Bouchet, L., & Atteia, J.-L. 2016, SIMULATEUR DE DONNEES " BRUIT " SVOM / ECLAIRs, Tech. rep.

Bouhargani, H. E. 2017a, Simulation et analyse de l ' entrée SAA Réunion GhOST, Tech. rep., IRAP

—. 2017b, Simulation de données ECLAIRs Réunion GhOST, Tech. rep., IRAP

Bozzo, E., Bernardini, F., Ferrigno, C., et al. 2017, Astron. Astrophys., 608, 128, doi: 10.1051/0004-6361/201730398

Braga, J. 2019, Publ. Astron. Soc. Pacific, 132, doi: 10.1088/1538-3873/ab450a

Braga, J., Durão, O. S. C., Castro, M., et al. 2020, Mon. Not. R. Astron. Soc., 493, 4852, doi: 10.1093/mnras/staa500

Briggs, M. S., Pendleton, G. N., Kippen, R. M., et al. 1999, Astrophys. J. Suppl. Ser., 122, 503, doi: 10.1086/313221

Briggs, M. S., Fishman, G. J., Connaughton, V., et al. 2010, J. Geophys. Res. Sp. Phys., 115, doi: 10.1029/2009JA015242

Bringer, O., Van Lauwe, A., & Goutte, H. 2013, Etude de la Validité d'une Hypothèse d'Irradiation Continue dans le Cadre de Calculs d'Activation Résiduelle du Détecteur ECLAIRs, Tech. rep.

Burrows, D. N., Hill, J. E., Nousek, J. A., et al. 2005, Space Sci. Rev., 120, 165, doi: 10.1007/s11214-005-5097-2

Campana, R., Feroci, M., Del Monte, E., et al. 2013, Exp. Astron., 36, 451, doi: 10.1007/s10686-013-9341-6

Campana, S., & Di Salvo, T. 2018, in Phys. Astrophys. Neutron Stars (Springer, Cham), 149–184. https://link.springer.com/chapter/10.1007/978-3-319-97616-7{_}4

Caroli, E., Stephen, J. B., Di Cocco, G., Natalucci, L., & Spizzichino, A. 1987, Space Sci. Rev., 45, 349, doi: 10.1007/BF00171998

Cenko, S. B., Krimm, H. A., Horesh, A., et al. 2012, Astrophys. J., 753, 77, doi: 10.1088/0004-637X/753/1/77

Chandra, P., & Frail, D. A. 2012, Astrophys. J., 746, 156, doi: 10.1088/0004-637X/746/2/156

Chattopadhyay, T., Vadawale, S. V., Aarthy, E., et al. 2019, Astrophys. J., 884, 123, doi: 10.3847/1538-4357/ab40b7

Churazov, E., Sazonov, S., Sunyaev, R., & Revnivtsev, M. 2008, Mon. Not. R. Astron. Soc., 385, 719

Cordier, B., Desclaux, F., Foliard, J., & Schanne, S. 2008, in AIP Conf. Proc., Vol. 1000 (AIP), 585–588. http://aip.scitation.org/doi/abs/10.1063/1.2943538

Cordier, B., Wei, J., Atteia, J. L., et al. 2014, Proc. Sci., doi: 10.22323/1.233.0005

Corre, D. 2016, Photometric redshift estimation for the GFT, Tech. Rep. November

Corre, D., Basa, S., Klotz, A., et al. 2018, in Model. Syst. Eng. Proj. Manag. Astron. VIII, ed. G. Z. Angeli & P. Dierickx, Vol. 10705 (SPIE), 65. https://www.spiedigitallibrary.org/conference-proceedings-of-spie/10705/2313127/End-to-end-simulations-for-COLIBRI-ground-follow-up-telescope/10.1117/12.2313127.full

Costa, E., Frontera, F., Heise, J., et al. 1997, Nature, 387, 783, doi: 10.1038/42885

Dagoneau, N. 2020, PhD thesis, Université Paris-Saclay

Dagoneau, N., Schanne, S., Rodriguez, J., Atteia, J. L., & Cordier, B. 2020, Astron. Astrophys., doi: 10.1051/0004-6361/202038995

Daigne, F. 2019, ECL+GRM Simulations: technical note on the tools developed at IAP, Tech. rep.

Daigne, F., Goldoni, P., Ferrando, P., et al. 2002, Astron. Astrophys., 386, 531, doi: 10.1051/0004-6361:20020223

Dezalay, J.-P., Barat, C., Talon, R., et al. 1992, in AIP Conf. Proc., Vol. 265 (AIP Publishing), 304–309. http://aip.scitation.org/doi/abs/10.1063/1.42799

Dezalay, J. P., Lestrade, J. P., Barat, C., et al. 1996, Astrophys. J., 471, L27, doi: 10.1086/310321

Done, C., Gierliński, M., & Kubota, A. 2007, Astron. Astrophys. Rev., 15, 1, doi: 10.1007/s00159-007-0006-1

Dong, Y., Wu, B., Li, Y., Zhang, Y., & Zhang, S. 2010, Sci. China Physics, Mech. Astron., 53, 40, doi: 10.1007/s11433-010-0011-7

Dwyer, J. R. 2003, Geophys. Res. Lett., 30, n/a, doi: 10.1029/2003GL017781

Eichler, D., Livio, M., Piran, T., & Schramm, D. N. 1989, Nature, 340, 126, doi: 10.1038/340126a0

Fan, X., Zou, G., Qiu, Y., et al. 2020, Appl. Opt., 59, 3049, doi: 10.1364/ao.386177

Feroci, M., Frontera, F., Costa, E., et al. 1997, EUV, X-Ray, Gamma-Ray Instrum. Astron. VIII, 3114, 186, doi: 10.1117/12.283768

Fishman, G. J., Bhat, P. N., Mallozzi, R., et al. 1994, Science (80-.)., 264, 1313, doi: 10.1126/science.264.5163.1313

Forman, W., Jones, C., Cominsky, L., et al. 1978, Astrophys. J. Suppl. Ser., 38, 357, doi: 10.1086/190561

Frail, D. A., Kulkarnit, S. R., Nicastro, L., Feroci, M., & Taylor, G. B. 1997, Nature, 389, 261, doi: 10.1038/38451

Frank, J., King, A., & Raine, D. 2002, Accretion Power in Astrophysics: Third Edition, doi: 10.1063/1.2815178. http://adsabs.harvard.edu/abs/2002apa..book.....F

Frank, J., & Rees, M. J. 1976, Mon. Not. R. Astron. Soc., 176, 633, doi: 10.1093/mnras/176.3.633

Frontera, F., Costa, E., Dal Fiume, D., et al. 1997, Astron. Astrophys. Suppl. Ser., 122, 357, doi: 10.1051/aas:1997140

Fruchter, A. S., Levan, A. J., Strolger, L., et al. 2006, Nature, 441, 463, doi: 10.1038/nature04787

Fuentes-Fernández, J., Watson, A. M., Cuevas, S., et al. 2020, J. Astron. Instrum., 9, doi: 10.1142/S2251171720500014

Fukugita, M., Ichikawa, T., Gunn, J. E., et al. 1996, Astron. J., 111, 1748, doi: 10.1086/117915

Fuschino, F., Campana, R., Labanti, C., et al. 2019, Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip., 936, 199, doi: 10.1016/j.nima.2018.11.072

Garnier, G. 2019, Débit du lien photon ELS : Simulation d ' entrée dans la SAA, Tech. rep.

Gehrels, N. 2004, Astrophys. J., 611, 777, doi: 10.1086/422091

Gehrels, N., & Cannizzo, J. K. 2013, Philos. Trans. R. Soc. A Math. Phys. Eng. Sci., 371, 20120270, doi: 10.1098/rsta.2012.0270

Gehrels, N., Chipman, E., & Kniffen, D. 1993, Astron. Astrophys. Suppl. Ser., 97, 5

Gehrels, N., Norris, J. P., Barthelmy, S. D., et al. 2006, Nature, 444, 1044, doi: 10.1038/nature05376

Ghirlanda, G., Ghisellini, G., & Lazzati, D. 2004a, Astrophys. J., 616, 331, doi: 10.1086/424913

Ghirlanda, G., Ghisellini, G., Lazzati, D., & Firmani, C. 2004b, Astrophys. J., 613, L13, doi: 10.1086/424915

Ghisellini, G., Ghirlanda, G., Nava, L., & Celotti, A. 2010, Mon. Not. R. Astron. Soc., 403, 926, doi: 10.1111/j.1365-2966.2009.16171.x

Giacconi, R., Gursky, H., Paolini, F. R., & Rossi, B. B. 1962, Phys. Rev. Lett., 9, 439, doi: 10.1103/PhysRevLett.9.439

Ginet, G. P., Madden, D., Dichter, B. K., & Brautigam G, D. H. 2007, in IEEE Radiat. Eff. Data Work. (IEEE), 1–8. http://ieeexplore.ieee.org/document/4342532/

Godet, O., Nasser, G., Atteia, J. L., et al. 2014, Sp. Telesc. Instrum. 2014 Ultrav. to Gamma Ray, 9144, 914424, doi: 10.1117/12.2055507

Goldstein, A., Preece, R. D., Mallozzi, R. S., et al. 2013, ApJS, 208, 21, doi: 10.1088/0067-0049/208/2/21

Goldwurm, A. 2017, MISSION RATIONALE AND REQUIREMENTS, Tech. rep.

—. 2018, Architecture Design Document of the General and ToO Program ECLAIRs Pipeline, Tech. rep.

Goldwurm, A., Goldoni, P., Gros, A., et al. 2001, Eur. Sp. Agency, (Special Publ. ESA SP, 497

Götz, D., Laurent, P., Antier, S., et al. 2014a, Mon. Not. R. Astron. Soc., 444, 2776, doi: 10.1093/mnras/stu1634

Götz, D., Laurent, P., Lebrun, F., Daigne, F., & Bonjak, E. 2009, Astrophys. J., 695, L208, doi: 10.1088/0004-637X/695/2/L208

Götz, D., Osborne, J., Cordier, B., et al. 2014b, Sp. Telesc. Instrum. 2014 Ultrav. to Gamma Ray, 9144, 914423, doi: 10.1117/12.2054898

Gou, L., Fox, D. B., & Meszaros, P. 2007, Astrophys. J., 668, 1083, doi: 10.1086/520699

Gros, A. 2014, Eclairs : Geometry , Deconvolution and PSLE, Tech. rep.

—. 2016a, Expected burst number : a tool to evaluate Eclair ' s performance, Tech. rep.

—. 2016b, Eclairs : SNR from bricks, Tech. rep.

Grove, J. E., Cheung, C. C., Kerr, M., et al. 2020

Gruber, D., Goldstein, A., Von Ahlefeld, V. W., et al. 2014, Astrophys. Journal, Suppl. Ser., 211, 12, doi: 10.1088/0067-0049/211/1/12

Gruber, D. E., Matteson, J. L., Peterson, L. E., & Jung, G. V. 1999, Astrophys. J., 520, 124

Guillemot, P. 2020, ECLAIRs - Dossier de Définition, Tech. rep.

Gunson, J., & Polychronopulos, B. 1976, Mon. Not. R. Astron. Soc., 177, 485, doi: 10.1093/mnras/177.3.485

Gurevich, A. V., Milikh, G. M., & Roussel-Dupre, R. 1992, Phys. Lett. A, 165, 463, doi: 10.1016/0375-9601(92)90348-P

Haar, A. 1910, Math. Ann., 69, 331, doi: 10.1007/BF01456326

Heussaff, V. 2015, PhD thesis. http://www.theses.fr/2015TOU30205/document

Heussaff, V., Atteia, J. L., & Zolnierowski, Y. 2013, Astron. Astrophys., 557, 100, doi: 10.1051/0004-6361/201321528

In 't Zand, J. 1992, PhD thesis

Jager, R., Mels, W. A., Brinkman, A. C., et al. 1997, Astron. Astrophys. Suppl. Ser., 125, 557, doi: 10.1051/aas:1997243

Jaubert, J., Morand, V., & Jouret, M. 2017, Realistic Mission Scenarios for Satellite Power Analysis, Tech. rep.

Juul, N. H. 1976, Lett. Heat Mass Transf., 3, 513, doi: 10.1016/0094-4548(76)90006-0

Kalemci, E., Boggs, S. E., Kouveliotou, C., Finger, M., & Baring, M. G. 2006, Astrophys. J. Suppl. Ser., 169, 75, doi: 10.1086/510676

Kaneko, Y., Preece, R. D., Briggs, M. S., et al. 2006a, Astrophys. J. Suppl. Ser., 166, 298, doi: 10.1086/505911

—. 2006b, Astrophys. J. Suppl. Ser., 166, 298, doi: 10.1086/505911

Kasliwal, M. M., Nakar, E., Singer, L. P., et al. 2017, Science (80-.)., 358, 1559, doi: 10.1126/science.aap9455

Kirsten, F., Snelders, M., Jenkins, M., et al. 2020

Klebesadel, R. W., Strong, I. B., & Olson, R. A. 1973, OBSERVATIONS OF GAMMA-RAY BURSTS OF COSMIC ORIGIN, Tech. rep.

Komossa, S. 2015, J. High Energy Astrophys., 7, 148, doi: 10.1016/j.jheap.2015.04.006

Kouveliotou, C., Koshut, T., Briggs, M. S., et al. 1996, in Am. Inst. Phys. Conf. Ser., Vol. 384 (AIP Publishing), 42–46

Kouveliotou, C., Meegan, C. A., Fishman, G. J., et al. 1993, Astrophys. J., 413, L101, doi: 10.1086/186969

Kraushaar, W. L., Clark, G. W., Garmire, G. P., et al. 1972, Astrophys. J., 177, 341, doi: 10.1086/151713

Kumar, P., & Barniol Duran, R. 2009, On the generation of high-energy photons detected by the Fermi Satellite from gamma-ray bursts, Oxford Academic, doi: 10.1111/j.1745-3933.2009.00766.x. https://academic.oup.com/mnrasl/article-lookup/doi/10.1111/j.1745-3933.2009.00766.x

Kumar, P., & McMahon, E. 2008, Mon. Not. R. Astron. Soc., 384, 33, doi: 10.1111/j.1365-2966.2007.12621.x

Kumar, P., & Zhang, B. 2015, Phys. Rep., 561, 1, doi: 10.1016/j.physrep.2014.09.008

Lacombe, K., Nasser, G., Amoros, C., et al. 2013, Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip., 732, 122, doi: 10.1016/j.nima.2013.07.003

Le Provost, H., Schanne, S., Flouzat, C., et al. 2013, IEEE Nucl. Sci. Symp. Conf. Rec., doi: 10.1109/NSSMIC.2013.6829557

Lee, H. K., Wijers, R. A., & Brown, G. E. 2000, The Blandford-Znajek process as a central engine for a gamma-ray burst, Elsevier, doi: 10.1016/S0370-1573(99)00084-8

Lewin, W. H., Van Paradijs, J., & Taam, R. E. 1993, Space Sci. Rev., 62, 223, doi: 10.1007/BF00196124

Lewin, W. H. G., Doty, J., Clark, G. W., et al. 1976, Astrophys. J., 207, L95, doi: 10.1086/182188

Li, C. K., Lin, L., Xiong, S. L., et al. 2020a

Li, X. Q., Wen, X. Y., An, Z. H., et al. 2020b, Sci. Sin. Phys. Mech. Astron., 50, 129508, doi: 10.1360/SSPMA-2019-0417

Lien, A., Sakamoto, T., Barthelmy, S. D., et al. 2014, Proc. Sci., 829, doi: 10.22323/1.233.0038

Manzo, G., Giarrusso, S., Santangelo, A., et al. 1997, Astron. Astrophys. Suppl. Ser., 122, 341, doi: 10.1051/aas:1997139

Mate, S. 2020, ECLAIRs background and data simulator document ECLAIRs background simulator, Tech. rep.

Mate, S., Bouchet, L., Atteia, J.-L., et al. 2019, Exp. Astron., 48, 171, doi: 10.1007/s10686-019-09643-x

Mazet, E. P., Golenetskij, S. V., & IlInskij, V. N. 1974, Pisma v Zhurnal Eksp. i Teor. Fiz., 19, 126

Mazets, E. P., Golenetskii, S. V., Il'inskii, V. N., Aptekar', R. L., & Guryan, Y. A. 1979, Nature, 282, 587, doi: 10.1038/282587a0

McGlynn, S., Clark, D. J., Dean, A. J., et al. 2007, Astron. Astrophys., 466, 895, doi: 10.1051/0004-6361:20066179

McGlynn, S., Foley, S., McBreen, B., et al. 2009, Astron. Astrophys., 499, 465, doi: 10.1051/0004-6361/200810920

Meegan, C., Lichti, G., Bhat, P. N., et al. 2009, Astrophys. J., 702, 791, doi: 10.1088/0004-637X/702/1/791

Meegan, C. A., Fishman, G. J., Wilson, R. B., et al. 1992, Nature, 355, 143, doi: 10.1038/355143a0

Meidinger, N., Andritschke, R., Hälker, O., et al. 2006, Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip., 568, 141, doi: 10.1016/j.nima.2006.05.268

Mereghetti, S., Savchenko, V., Ferrigno, C., et al. 2020

Mészáros, P., & Rees, M. J. 1994, MNRAS, 269, 41

Meszaros, P., & Rees, M. J. 1997, Astrophys. J., 476, 232, doi: 10.1086/303625

Metzger, A. E., Parker, R. H., Gilman, D., Peterson, L. E., & Trombka, J. I. 1974, Astrophys. J., 194, L19, doi: 10.1086/181660

Metzger, B. D., Giannios, D., Thompson, T. A., Bucciantini, N., & Quataert, E. 2011, Mon. Not. R. Astron. Soc., 413, 2031, doi: 10.1111/j.1365-2966.2011.18280.x

Metzger, B. D., Martínez-Pinedo, G., Darbha, S., et al. 2010, Mon. Not. R. Astron. Soc., 406, 2650, doi: 10.1111/j.1365-2966.2010.16864.x

Metzger, M. R., Djorgovski, S. G., Kulkarni, S. R., et al. 1997, Nature, 387, 878, doi: 10.1038/43132

Modjaz, M., Stanek, K. Z., Garnavich, P. M., et al. 2006, Astrophys. J., 645, L21, doi: 10.1086/505906

Mooley, K. P., Deller, A. T., Gottlieb, O., et al. 2018, Nature, 561, 355, doi: 10.1038/s41586-018-0486-3

Moretti, A., Pagani, C., Cusumano, G., et al. 2009, Astron. Astrophys., 493, 501

Nakar, E. 2007, Short-hard gamma-ray bursts, North-Holland, doi: 10.1016/j.physrep.2007.02.005

Narayan, R., Paczyński, B., & Piran, T. 1992, Astrophys. J., 395, L83, doi: 10.1086/186493

Nasser, G., Godet, O., Atteia, J.-L., et al. 2014, Sp. Telesc. Instrum. 2014 Ultrav. to Gamma Ray, 9144, 91443X, doi: 10.1117/12.2056354

Nemiroff, R. J. 1994, in AIP Conf. Proc., Vol. 307 (AIP Publishing), 730–734. http://aip.scitation.org/doi/abs/10.1063/1.45838

Nousek, J. A., Kouveliotou, C., Grupe, D., et al. 2005, Astrophys. J., 642, 389, doi: 10.1086/500724

Oates, S. R., Page, M. J., Schady, P., et al. 2009, Mon. Not. R. Astron. Soc., 395, 490, doi: 10.1111/j.1365-2966.2009.14544.x

Odaka, H., Asai, M., Hagino, K., et al. 2018, Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip., 891, 92, doi: 10.1016/J.NIMA.2018.02.071

Olausen, S. A., & Kaspi, V. M. 2014, Astrophys. J. Suppl. Ser., 212, 6, doi: 10.1088/0067-0049/212/1/6

Palmer, D. M., Barthelmy, S., Gehrels, N., et al. 2005, doi: 10.1038/nature03525

Panaitescu, A., & Kumar, P. 2001, Astrophys. J., 560, 49

Parmar, A. N., Martin, D. D., Bavdaz, M., et al. 1997, Astron. Astrophys. Suppl. Ser., 122, 309, doi: 10.1051/aas:1997137

Patruno, A., & Watts, A. L. 2021, in Timing Neutron Stars Pulsations, Oscil. Explos. (Springer, Berlin, Heidelberg), 143–208. https://link.springer.com/chapter/10.1007/978-3-662-62110-3{_}4

Paul, B., & Naik, S. 2011, Bull. Astron. Soc. India, 39, 429

Pe'er, A., Meszaros, P., & Rees, M. J. 2005, Astrophys. J., 635, 476, doi: 10.1086/497360

—. 2006, Astrophys. J., 642, 995, doi: 10.1086/501424

Pélangeon, A., Atteia, J. L., Nakagawa, Y. E., et al. 2008, Astron. Astrophys., 491, 157, doi: 10.1051/0004-6361:200809709

Perry, K. L., Ginet, G., Quigley, S., & Madden, D. 2008, in Am. Geophys. Union, Fall Meet. 2008, Abstr. id.SM11A-1584. http://adsabs.harvard.edu/abs/2008AGUFMSM11A1584P

Piran, T. 1995

—. 1999, Phys. Rep., 314, 575, doi: 10.1016/S0370-1573(98)00127-6

Piro, L., Scarsi, L., & Butler, R. C. 1995, in X-Ray EUV/FUV Spectrosc. Polarim., ed. S. Fineschi, Vol. 2517 (SPIE), 169–181. http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1003664

Preece, R. D., Briggs, M. S., Mallozzi, R. S., et al. 2000, THE BATSE GAMMA-RAY BURST SPECTRAL CATALOG. I. HIGH TIME RESOLUTION SPECTROSCOPY OF BRIGHT BURSTS USING HIGH ENERGY RESOLUTION DATA, Tech. rep.

Prochaska, J. X., Bloom, J. S., Chen, H., et al. 2004, Astrophys. J., 611, 200, doi: 10.1086/421988

Qin, Y., Liang, E.-W., Liang, Y.-F., et al. 2012, Astrophys. J., 763, 15, doi: 10.1088/0004-637X/763/1/15

Rees, M. J., & Meszaros, P. 1994, Astrophys. J., 430, L93, doi: 10.1086/187446

Remillard, R. A., & McClintock, J. E. 2006, Annu. Rev. Astron. Astrophys., 44, 49, doi: 10.1146/annurev.astro.44.051905.092532

Ricker, G. R. 2003, in AIP Conf. Proc. (AIP Publishing), 3–16

Roberts, O. J., Fitzpatrick, G., Stanbro, M., et al. 2018, J. Geophys. Res. Sp. Phys., 123, 4381, doi: 10.1029/2017JA024837

Roming, P. W., Kennedy, T. E., Mason, K. O., et al. 2005, Space Sci. Rev., 120, 95, doi: 10.1007/s11214-005-5095-4

Ruderman, M. 1975, Ann. N. Y. Acad. Sci., 262, 164, doi: 10.1111/j.1749-6632.1975.tb31430.x

Ryde, F. 2005, Astrophys. J., 625, L95, doi: 10.1086/431239

Ryde, F., Axelsson, M., Zhang, B. B., et al. 2010, Astrophys. J. Lett., 709, L172, doi: 10.1088/2041-8205/709/2/L172

Sakamoto, T., Lamb, D. Q., Kawai, N., et al. 2005, Astrophys. J., 629, 311, doi: 10.1086/431235

Sakamoto, T., Barthelmy, S. D., Baumgartner, W. H., et al. 2011, Astrophys. J. Suppl. Ser., 195, 2, doi: 10.1088/0067-0049/195/1/2

Sari, R., & Piran, T. 1997, Astrophys. J., 485, 270, doi: 10.1086/304428

Sari, R., Piran, T., & Narayan, R. 1998, Astrophys. J., 497, L17, doi: 10.1086/311269

Sato, T., Odaka, H., Hiragi, K., et al. 2012, in IEEE Nucl. Sci. Symp. Conf. Rec., 1727–1730

Sazonov, S., Churazov, E., Sunyaev, R., & Revnivtsev, M. 2007, Mon. Not. R. Astron. Soc., 377, 1726, doi: 10.1111/j.1365-2966.2007.11746.x

Schaefer, B. E. 2007, Astrophys. J., 660, 16, doi: 10.1086/511742

Schanne, S. 2017, Simulation signal à bruit counts & image & par zone, Tech. rep.

Schanne, S., Provost, H. L., Kestener, P., et al. 2008, Technical design report of the Scientific processing and Trigger Unit onboard ECLAIRs (UGTS), Tech. rep.

Schanne, S., Le Provost, H., Kestener, P., et al. 2013, IEEE Nucl. Sci. Symp. Conf. Rec., doi: 10.1109/NSSMIC.2013.6829408

Shensa, M. J. 1992, IEEE Trans. Signal Process., 40, 2464, doi: 10.1109/78.157290

Shirasaki, Y., Kawai, N., Yoshida, A., et al. 2003, Publ. Astron. Soc. Japan, 55, 1033, doi: 10.1093/PASJ/55.5.1033

Singh, K. P., Tandon, S. N., Agrawal, P. C., et al. 2014, in Sp. Telesc. Instrum. 2014 Ultrav. to Gamma Ray, ed. T. Takahashi, J.-W. A. den Herder, & M. Bautz, Vol. 9144 (SPIE), 91441S. http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2062667

Sizun, P. 2011, Synthesis of ECLAIRs Geant4 Simulations, Tech. rep.

—. 2015a, Status of model and simulations Other background components, Tech. rep.

—. 2015b, About ECLAIRs ' s ARF & RMF version 6, Tech. rep.

Smith, D. M., Lopez, L. I., Lin, R. P., & Barrington-Leigh, C. P. 2005, Science (80-.)., 307, 1085, doi: 10.1126/science.1107466

Smith, J. R., Briggs, M. S., Bruno, A., et al. 2019, in ICRC. http://pos.sissa.it/

Stanek, K. Z., Matheson, T., Garnavich, P. M., et al. 2003, Astrophys. J., 591, L17, doi: 10.1086/376976

Starck, J. L., Fadili, J., & Murtagh, F. 2007, IEEE Trans. Image Process., 16, 297, doi: 10.1109/TIP.2006.887733

Swank, J. H., Smith, D. M., & Markwardt, C. B. 2007, ATel, 999, 1

Tanvir, N. R., Levan, A. J., González-Fernández, C., et al. 2017, doi: 10.3847/2041-8213/aa90b6

Thompson, C., & Duncan, R. C. 1995, Mon. Not. R. Astron. Soc., 275, 255

Tonry, J. L., Stubbs, C. W., Lykke, K. R., et al. 2012, Astrophys. J., 750, 99, doi: 10.1088/0004-637X/750/2/99

Türler, M., Chernyakova, M., Courvoisier, T. J.-L., et al. 2010, Astron. Astrophys., 512, A49, doi: 10.1051/0004-6361/200913072

Turpin, D., Heussaff, V., Dezalay, J.-P., et al. 2016, Astrophys. J., 831, 28, doi: 10.3847/0004-637x/831/1/28

Ubertini, P., Lebrun, F., Di Cocco, G., et al. 2003, Astron. Astrophys., 411, L131, doi: 10.1051/0004-6361:20031224

Usoskin, I. G., Alanko-Huotari, K., Kovaltsov, G. A., & Mursula, K. 2005, J. Geophys. Res., 110, A12108, doi: 10.1029/2005JA011250

van den Eijnden, J., Bagnoli, T., Degenaar, N., et al. 2017, Mon. Not. R. Astron. Soc. Lett., 466, L98, doi: 10.1093/mnrasl/slw244

Van Paradijs, J., Groot, P. J., Galama, T., et al. 1997, Nature, 386, 686, doi: 10.1038/386686a0

Vanderspek, R., Sakamoto, T., Barraud, C., et al. 2004, Astrophys. J., 617, 1251, doi: 10.1086/423923

Vedrenne, G., Roques, J.-P., Schönfelder, V., et al. 2003, Astron. Astrophys., 411, L63, doi: 10.1051/0004-6361:20031482

Villasenor, J. N. 2003, in AIP Conf. Proc., Vol. 662 (AIP Publishing), 33–37

Watts, R. G. 1965, J. Heat Transf., 87, 369, doi: 10.1115/1.3689118

Wei, J., Cordier, B., Antier, S., et al. 2016, 1

Weidenspointner, G., Kiener, J., Gros, M., et al. 2003, Astron. Astrophys., 411, L113, doi: 10.1051/0004-6361:20031209

Weimin, Y., P, O. J., Chen, Z., & Richard, W. 2016, arXiv, 36, 117, doi: 10.11728/CJSS2016.02.117

Winkler, C., Courvoisier, T. J., Di Cocco, G., et al. 2003, Astron. Astrophys., 411, L1, doi: 10.1051/0004-6361:20031288

Woosley, S., & Bloom, J. 2006, Annu. Rev. Astron. Astrophys., 44, 507, doi: 10.1146/annurev.astro.43.072103.150558

Woosley, S. E. 1993, Astrophys. J., 405, 273, doi: 10.1086/172359

Xu, Y., Xin, L. P., Wang, J., et al. 2020a, Publ. Astron. Soc. Pacific, 132, doi: 10.1088/1538-3873/ab7a73

Xu, Y., Xin, L., Han, X., et al. 2020b

Zhang, B. 2018, The Physics of Gamma-Ray Bursts (Cambridge University Press), doi: 10.1017/9781139226530

Zhang, B., Fan, Y. Z., Dyks, J., et al. 2006, Astrophys. J., 642, 354, doi: 10.1086/500723

Zhang, B., Zhang, B.-B., Virgili, F. J., et al. 2009, Astrophys. J., 703, 1696, doi: 10.1088/0004-637X/703/2/1696

Zhang, B.-B., Zhang, B., Liang, E.-W., et al. 2011, Astrophys. J., 730, 141, doi: 10.1088/0004-637X/730/2/141

Zhang, S. N., Santangelo, A., Feroci, M., et al. 2019, Sci. China Physics, Mech. Astron., 62, doi: 10.1007/s11433-018-9309-2

Zhao, D., Cordier, B., Sizun, P., et al. 2012, Exp. Astron., 34, 705, doi: 10.1007/s10686-012-9313-2

Appendix A

Coordinate frame transformations used in the Albedo background computation

The computation of the Albedo background for ECLAIRs requires coordinate transformations between several frames. These transformations are from the GEANT4 frame (or ECLAIRs frame) to the "Local" frame (Section A.1), from the "Local" frame to the "MAP" frame (Section A.2), from the "MAP" frame to the geographic frame (Section A.3) and from the geographic frame to the geomagnetic frame (Section A.4). The figure A.1 shows some of the frames involved in these computations. The sections below explain each transformation and give the transformation matrix needed in each case.

A.1 The GEANT4 frame to the "Local" frame transformation

The directions of the incoming photons are recorded in the GEANT4 (or ECLAIRs) frame. To correctly use the Eqn. 4.14 and Eqn. 4.15, the photon coordinates should be transformed into the "Local" frame. The transformation from the GEANT4 frame to the "Local" frame requires three Euler angles (see the inset diagram in the figure A.1), θ_{E}, ϕ_{E} and ψ_{E}. The first two angles are simply the spherical coordinates of the direction of the Earth's centre in the GEANT4 frame (same as the Z-axis of the "Local" frame). Rotation using these angle align the Z-axis of the GEANT4 frame with the Z-axis of the "Local" frame. The third angle, called Earth roll, is needed (see section A.1.1 for the definition) to align the X-axes of both the frames. The transformation matrix corresponding to the GEANT4 frame to the "Local" frame

transformation is given as:

$$R_{G4->Local} = \begin{bmatrix} c_1c_2c_3 - s_1s_3 & c_1s_3 + c_2c_3s_1 & -c_3s_2 \\ -c_3s_1 - c_1c_2s_3 & c_1c_3 - c_2s_1s_3 & s_2s_3 \\ c_1s_2 & s_1s_2 & c_2 \end{bmatrix} \tag{A.1}$$

where,

$$c_1 = \cos\phi_{\rm E},\ c_2 = \cos\theta_{\rm E},\ c_3 = \cos\psi_{\rm E} \tag{A.2}$$

$$s_1 = \sin\phi_{\rm E},\ s_2 = \sin\theta_{\rm E},\ s_3 = \sin\psi_{\rm E} \tag{A.3}$$

Using this, any coordinate in the GEANT4 frame, e.g. the photon's incoming direction $(\theta_p,\ \phi_p)$ can be transformed to the "Local" frame coordinate $(\theta_{p,L},\ \phi_{p,L})$ as:

$$\begin{bmatrix} \sin\theta_{p,L}\cos\phi_{p,L} \\ \sin\theta_{p,L}\sin\phi_{p,L} \\ \cos\theta_{p,L} \end{bmatrix} = R_{G4->Local} \cdot \begin{bmatrix} \sin\theta_p\cos\phi_p \\ \sin\theta_p\sin\phi_p \\ \cos\theta_p \end{bmatrix} \tag{A.4}$$

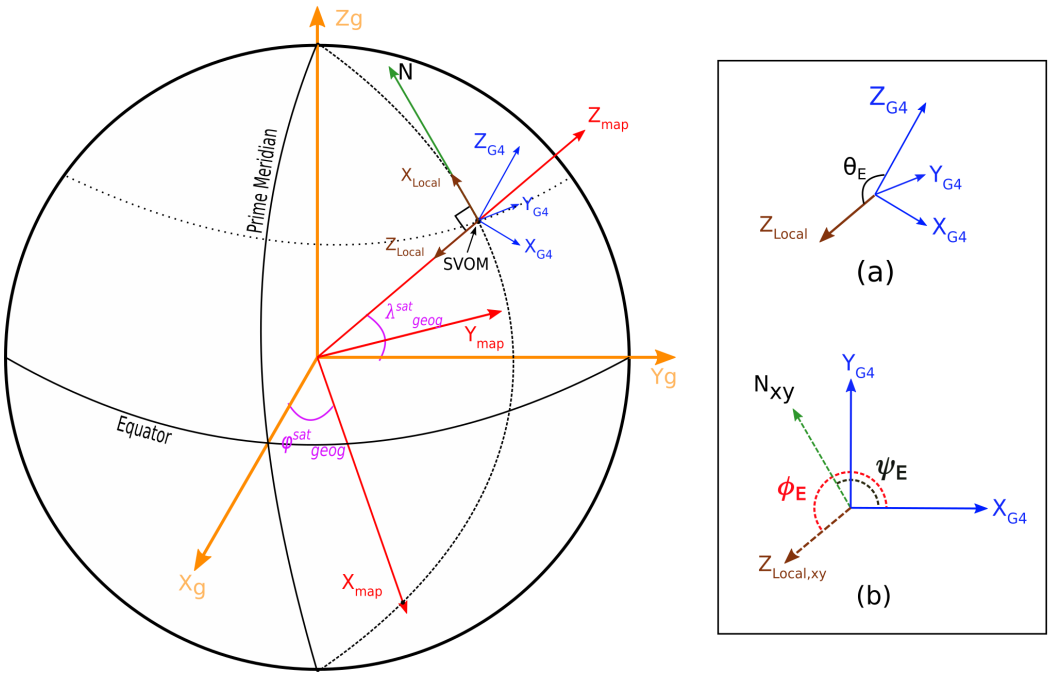


Figure A.1 – A schematic showing the orientation of the coordinate systems used in the Albedo background computation at a given satellite latitude (curved horizontal dotted line) and longitude (curved vertical dashed line). The GEANT4 (or ECLAIRs) frame is shown in Blue, the "Local" frame is shown in Brown (Y-axis not shown to avoid confusion in the projection), the "MAP" frame is shown in Red and the geographic frame is shown in Orange. The direction of the local North is shown by the Green vector. The inset diagrams (a) and (b) show the definition of the $\theta_{\rm E}, \phi_{\rm E}$ and $\psi_{\rm E}$ angles. Note that in the inset diagram (b), the dotted vectors $\rm N_{xy}$ and $\rm Z_{Local,xy}$ are the projection of the North vector (N) and $\rm Z_{Local}$ axis in the XY plane of the GEANT4 frame respectively.

A.1.1 Definition and computation of ψ_{E}

The Earth roll angle (ψ_{E}) is defined as: *The angle between the direction of the local North and the X-axis of the GEANT4 frame counted positive in the direction of the Y-axis.* The local North is defined as the tangent along the longitude of the satellite and pointing towards the North (see vector "N" in figure A.1). The computation of ψ_{E} requires knowledge of the pointing direction of ECLAIRs in the celestial coordinate system (J2000 in case of *SVOM*). If we assume that $Q = [q_0, q_1, q_2, q_3]$ is the attitude quaternion going from the Z-axis of the J2000 to the pointing direction of ECLAIRs, then transformation matrix from J2000 frame to GEANT4 (or ECLAIRs) frame ($R_{J2000->G4}$) is given as [1]:

$$R_{J2000->G4} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_3q_0) & 2(q_1q_3 - q_2q_0) \\ 2(q_1q_2 - q_3q_0) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_1q_0) \\ 2(q_1q_3 + q_2q_0) & 2(q_2q_3 - q_1q_0) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{A.5}$$

And the direction of the local North in the J2000 frame is given as:

$$N_{J2000} = \begin{bmatrix} -\sin\lambda^{sat}_{J2000}\cos\varphi^{sat}_{J2000} \\ -\sin\lambda^{sat}_{J2000}\sin\varphi^{sat}_{J2000} \\ \cos\lambda^{sat}_{J2000} \end{bmatrix} \tag{A.6}$$

where λ^{sat}_{J2000} and φ^{sat}_{J2000} are the latitude and longitude of the satellite measured in the J2000 frame [2]. With Eqn. A.5 and Eqn. A.6 we can find the local North vector in the GEANT4 frame (N_{G4}) and then ψ_{E} as:

$$N_{G4} = R_{J2000->G4} \cdot N_{J2000} \quad \text{and} \quad \psi_{\mathrm{E}} = \arctan\left(\frac{N_{y,G4}}{N_{x,G4}}\right) \tag{A.7}$$

where, $N_{x,G4}$ and $N_{y,G4}$ are the x and y components of the N_{G4} vector respectively. Note that the range for ψ_{E} is from 0° to 360°.

A.2 The "Local" frame to the "MAP" frame transformation

This transformation is same as explained in the section 4.2.2.2 in Eqn. 4.10 to Eqn. 4.13. It is repeated here for completeness and all the definitions remain the

1. For more details on quaternions and rotation matrix see https://www.tu-berlin.de/fileadmin/fg169/miscellaneous/Quaternions.pdf

2. Note that normally orbital parameter files give satellite position in the J2000 frame and in the Cartesian (x, y, z) coordinates. The latitude and longitude can be computed from these coordinates using the usual definition.

same as in the main section. The θ_M and ϕ_M are the spherical coordinates in the "MAP" frame, the θ_L and ϕ_L are the spherical coordinates in the "Local" frame, R_E is the radius of the Earth, H_A is the height of the atmosphere and H is the altitude of the satellite.

$$\cos\theta_M = \frac{(1 - l \times \cos\theta_L)}{k} \tag{A.8}$$

where,

$$l = \cos\theta_L - \sqrt{k^2 - \sin^2\theta_L} \tag{A.9}$$

$$k = \frac{R_E + H_A}{R_E + H} \tag{A.10}$$

and,

$$\phi_M = \phi_L - \pi \tag{A.11}$$

A.3 The "MAP" frame to the geographic frame transformation

The Albedo emission depends on the geographic latitude and longitude of the emitting region of the atmosphere (Eqn. 4.3). Hence, the "MAP" frame coordinates must be transformed to the geographic frame to determine the expected number of photons. The orientation of the MAP frame relative to the geographic frame is given as: The Z_{MAP} axis passes through the centre of the Earth and the satellite position and in the direction of the sky. The X_{MAP} axis is parallel to the local meridian of the satellite and in the direction of the South and the Y_{MAP} axis is along the East direction completing the coordinates according to the right hand rule (Figure A.1). Unlike the previous transformation, here only two angles, namely the geographic latitude and longitude of the direction corresponding to the Z-axis of the MAP frame (which are also the latitude and longitude of the satellite, see figure A.1) are necessary. This transformation matrix for this is give as:

$$R_{MAP->geog} = \begin{bmatrix} c_1c_2 & -s_1 & c_1s_2 \\ c_2s_1 & c_1 & s_1s_2 \\ -s_2 & 0 & c_2 \end{bmatrix} \tag{A.12}$$

where,

$$c_1 = \cos\varphi^{sat}_{geog},\ c_2 = \cos\left(90^\circ - \lambda^{sat}_{geog}\right) \tag{A.13}$$

$$s_1 = \sin\varphi^{sat}_{geog},\ s_2 = \sin\left(90^\circ - \lambda^{sat}_{geog}\right) \tag{A.14}$$

where λ_{geog}^{sat} and φ_{geog}^{sat} are the geographic latitude and longitude of the satellite.[3] Using this, any coordinate in the MAP frame (θ_M, ϕ_M) can be transformed to the geographic coordinate $(\lambda_{geog}, \varphi_{geog})$ as:

$$\begin{bmatrix} \cos\lambda_{geog}\cos\varphi_{geog} \\ \cos\lambda_{geog}\sin\varphi_{geog} \\ \sin\lambda_{geog} \end{bmatrix} = R_{MAP->geog} \cdot \begin{bmatrix} \sin\theta_M\cos\phi_M \\ \sin\theta_M\sin\phi_M \\ \cos\theta_M \end{bmatrix} \tag{A.15}$$

A.4 The geographic frame to the geomagnetic frame transformation

This transformation is needed to compute the cutoff rigidity (R_{cut}, see Eqn. 4.3) in the Albedo "C" factor. The transformation requires the current position of the geomagnetic North pole. The transformation is given by the following matrix:

$$R_{geog->geomag} = \begin{bmatrix} c_1c_2 & s_1c_2 & -s_2 \\ s_1 & c_1 & 0 \\ c_1s_2 & s_1s_2 & c_2 \end{bmatrix} \tag{A.16}$$

where,

$$c_1 = \cos\varphi_{geog}^{gmag_north},\ c_2 = \cos(90^\circ - \lambda_{geog}^{gmag_north}) \tag{A.17}$$

$$s_1 = \sin\varphi_{geog}^{gmag_north},\ s_2 = \sin(90^\circ - \lambda_{geog}^{gmag_north}) \tag{A.18}$$

where $\lambda_{geog}^{gmag_north}$ and $\varphi_{geog}^{gmag_north}$ are the latitude and longitude of the geomagnetic North pole expressed in the geographic frame[4]. Using this, any coordinate in the geographic frame $(\lambda_{geog}, \varphi_{geog})$ can be transformed to the geomagnetic coordinate $(\lambda_{geomag}, \varphi_{geomag})$ as:

$$\begin{bmatrix} \cos\lambda_{geomag}\cos\varphi_{geomag} \\ \cos\lambda_{geomag}\sin\varphi_{geomag} \\ \sin\lambda_{geomag} \end{bmatrix} = R_{geog->geomag} \cdot \begin{bmatrix} \cos\lambda_{geog}\cos\varphi_{geog} \\ \cos\lambda_{geog}\sin\varphi_{geog} \\ \sin\lambda_{geog} \end{bmatrix} \tag{A.19}$$

Note that for computational purpose, the "Local" and "MAP" frames are pixelated into $\mathcal{N}_\theta \times \mathcal{N}_\phi$ angular bins. This also imply pixelatation of latitudes and longitudes

3. Note that the geographic latitude and longitude defined here are different than the J2000 latitude and longitude in the section A.1.1. The J2000 is an Earth centred inertial frame (ECI) while the geographic frame is Earth centred rotational (ECR) frame.

4. The position of the geomagnetic North pole varies due to the shift in the magnetic poles. It is usually estimated every year and in the code these values can be updated as input parameters depending on the year of the input attitude. To get latest values, see http://wdc.kugi.kyoto-u.ac.jp/poles/polesexp.html

in both, geographic and geomagnetic frames. However, this process doesn't affect the transformations and Eqn. A.4, A.15 and A.19 are valid for each angular bin ij where $i \in [0, \mathcal{N}_\theta]$ and $j \in [0, \mathcal{N}_\phi]$.

A.5 The celestial frame to the MAP frame transformation

The MAP frame coordinates can be directly obtained from the celestial (J2000) frame using the satellite latitude and longitude in the J2000 frame $(\lambda^{sat}_{J2000}, \varphi^{sat}_{J2000})$. This transformation is given by the following matrix:

$$R_{J2000->MAP} = \begin{bmatrix} s_1 c_2 & s_1 s_2 & -c_1 \\ -s_1 & c_2 & 0 \\ c_1 c_2 & c_1 s_2 & s_1 \end{bmatrix} \tag{A.20}$$

where,

$$c_1 = \cos \varphi^{sat}_{J2000}, \; c_2 = \cos \lambda^{sat}_{J2000} \tag{A.21}$$

$$s_1 = \sin \varphi^{sat}_{J2000}, \; s_2 = \sin \lambda^{sat}_{J2000} \tag{A.22}$$

Using this, any coordinate in the celestial frame $(\alpha_{J2000}, \delta_{J2000})$ can be transformed to the MAP frame coordinate (θ_M, ϕ_M) as:

$$\begin{bmatrix} \sin\theta_M \cos\phi_M \\ \sin\theta_M \sin\phi_M \\ \cos\theta_M \end{bmatrix} = R_{J2000->MAP} \cdot \begin{bmatrix} \cos\delta_{J2000} \cos\alpha_{J2000} \\ \cos\delta_{J2000} \sin\alpha_{J2000} \\ \sin\delta_{J2000} \end{bmatrix} \tag{A.23}$$

Note that this transformation is not used in the PIRA codes but can be useful to obtain geographic latitude and longitude directly from the celestial frame by combining it with the Eqn A.12.

Appendix B

PIRA codes for ECLAIRs

This appendix gives the PIRA codes used to simulate ECLAIRs background. The codes are written in *python3.6* using an object-oriented approach. The following table lists the name of the script/module with its purpose and each class is documented below in the listings. Figure B.1 shows the organisation of the overall package with a sequence of procedures within each module. To know how to run the codes, please see the data simulation manual (Mate, 2020).

Table B.1 – Description of PIRA scripts and modules

Script/module name	Description
config.ini	Configuration file with necessary input parameters
eclbkgsim.py	Main script to run the simulation
background.py	Module to handle calls to individual background process and merge events
cxb.py	Module to compute CXB background
reflection.py	Module to compute Reflection background
albedo.py	Module to compute Albedo background
electrons.py	Module to compute SAA electron background
protons.py	Module to compute SAA protons background
functions.py	Common functions used by all component modules

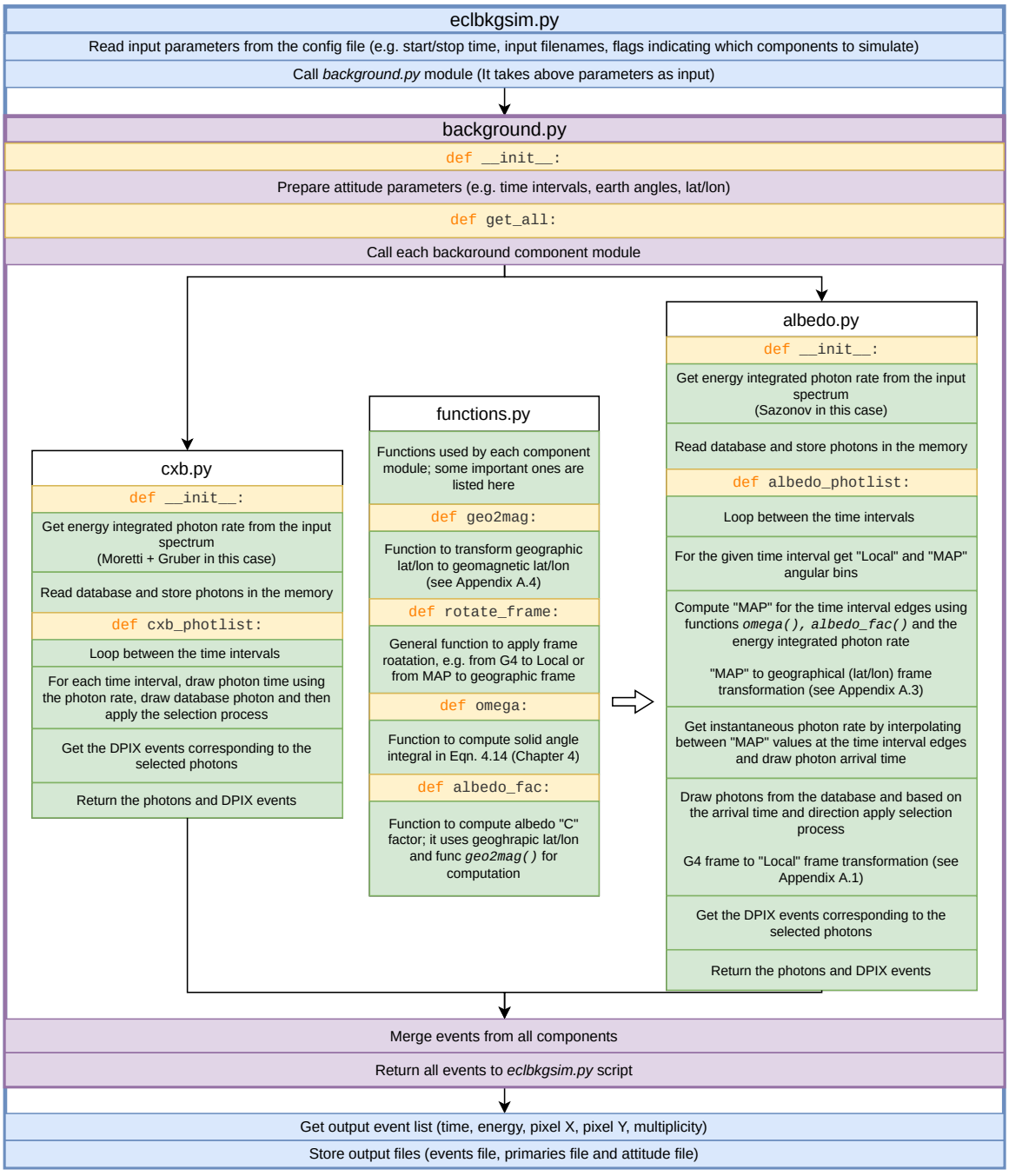


Figure B.1 – A diagram showing how the modules in PIRA code are organised, most important functions within them and sequence of execution. For illustration purpose, a case of only two background components is shown. In real case, the call from `background.py` is made to all user requested background components. The logical structure of other background component modules is similar.

Listing B.1 – config.ini

```
[PATHS]

# Input data dir
G4DataDir = /path/to/G4data
AttDir = /path/to/Attdata

# Orbit files
Attitude = %(AttDir)s/SVOM_Attitude.att
Att_WFlux = %(AttDir)s/SVOM_Attitude_wFlux.att

# CNES att and orbit files
CNES_att = %(AttDir)s/ephem_att_nominal_2020.dat
CNES_orb = %(AttDir)s/ephem_orb_nominal_2020.dat

# G4 Files
CXBSRC = %(G4DataDir)s/CXBMG_Source.fits
CXBDPIX = %(G4DataDir)s/CXBMG_Dpix.fits
REFSRC = %(G4DataDir)s/REFLEC_Source.fits
REFDPIX = %(G4DataDir)s/REFLEC_Dpix.fits
ALBSRC = %(G4DataDir)s/ALBEDO_Source.fits
ALBDPIX = %(G4DataDir)s/ALBEDO_Dpix.fits
ELECSRC = %(G4DataDir)s/database_electrons_particles.fits
ELECDPIX = %(G4DataDir)s/database_electrons_dpixevents.fits
PROTSRC = %(G4DataDir)s/database_protons_particles.fits
PROTDPIX = %(G4DataDir)s/database_protons_dpixevents.fits
ELECFLUX = %(G4DataDir)s/GEANT4_input_electrons.fits
PROTFLUX = %(G4DataDir)s/GEANT4_input_protons.fits

# Output
OutDir = /path/to/output/data

[VARIABLES]

###  Simulation Variables (To be set by user for each run as required)  ###


# Select Time range (in ISOT format)
TSTART = 2020-01-01T00:00:00.000
TSTOP = 2020-01-01T01:38:00.000

#If using CNES input files, update this time and set USE_CNES flag to 1
TSTART_CNES = 25567.0
TSTOP_CNES  = 25567.065972222223
USE_CNES = 0

#Particle computation flag
PARTICLES = 0

#Select components to calculate (only for array proceesing)
CXB = 1
REFLECTION = 1
ALBEDO = 1
ELECTRONS = 0
PROTONS = 0

# Pixel convention for output events, choose bwtween
# G4: Default Geant 4;
# DPIX: DPIX like (compatible with CEA and ELS);
```

```
# ECLGP: ECL General program pipeline like (compatible with data simulator)
PIXEL_CONV = G4

# GEANT4 coordinate parameters
X_OFFSET = 0 # DPIX centre offset from origin in X axis (in cm)
Y_OFFSET = 0 # DPIX centre offset from origin in Y axis (in cm)
Z_OFFSET = -22.885 # DPIX top surface offsetn from the G4 origin (in cm)

## Albedo C factor params
# Magnetic South pole coord (in degrees) forattitude file year (2020, change when
    needed)and solar modulation potential (in GV)
MPOLE_LAT = 80.37
MPOLE_LON = 287.59
PHI_SOL = 0.25

# Particle rate limit to restrict computation beyond detector saturation and to save
    time
ELECMAXRATE = 100000
PROTMAXRATE = 50000

###  Algorithm parameters (should not be changed often)  ###


# Earth radius and atmospheric height in km
R_EARTH = 6370
H_ATM = 100

 # ALBEDO MAP pixellation
NTHETA = 20
NPHI = 120

# Delta T for SAA particle loop. This is needed to save computaion time.Should not be
     changed often
DELTAT_PARTICLE = 5
```

Listing B.2 – eclbkgsim.py

```
#!/usr/bin/env python3

"""

Main script to be run to estimate background events for ECLAIRs
using the PIRA method described in the article Mate et. al 2019"

The output of this script produces DPIX eventlist, attitude information file
and primaries file for the duration of the SVOM orbit and the criteria
selected by the user.

For details about how to run the script see README.md

Authors : Sujay Mate, Benjamin Arcier
Credits : Sujay Mate, Laurent Bouchet, Jean-Luc Atteia, Benjamin Arcier
Version : 3.0 (Apr 2021)
Email : Sujay.Mate@irap.omp.eu, sujay.mate@gmail.com

"""

import os
import argparse
```

```
23 from configparser import ConfigParser, BasicInterpolation
24
25 import numpy as np
26 from astropy.io import fits
27 from astropy.time import Time
28 from astropy.table import Table
29
30 from background import CallProcess
31
32 # Define argparser and commandline arguments
33 parser = argparse.ArgumentParser()
34 parser.add_argument("--configfile", type=str, help="Custom config file name")
35 parser.add_argument("--tstart", type=str,
36                     help="Start Time in YYYY-MM-DDTHH:MM:SSS format or CNES time
       depending on input attfile")
37 parser.add_argument("--tstop", type=str,
38                     help="Start Time in YYYY-MM-DDTHH:MM:SSS format or CNES time
       depending on input attfile")
39 parser.add_argument('--particles', dest='particles',
40                     action='store_true', help="Computes particles if passed")
41 parser.add_argument("--outdir", type=str, help="Directroy to store outputfiles")
42 parser.add_argument("--outname", type=str, help="Custom outname for the output files"
       )
43 parser.add_argument("--use_CNES", dest='use_CNES',
44                     action='store_true', help="Use CNES att and orb files as input")
45 args = parser.parse_args()
46
47 # Read Config file
48 scriptdir = os.path.dirname(os.path.realpath(__file__))
49 if args.configfile:
50     configfile = scriptdir + "/" + args.configfile
51     print("---> Using " + args.configfile + " as configuration file. <---\n")
52 else:
53     configfile = scriptdir + "/config.ini"
54     print("---> Using config.ini as configuration file. <---\n")
55
56 # Pass the configfile through the configparser to access the variables
57 config = ConfigParser(interpolation=BasicInterpolation())
58 config.read(configfile)
59
60 # Read flag to check if CNES input file is to be used
61 use_CNES = config.getboolean('VARIABLES', 'USE_CNES')
62 if args.use_CNES is not None:
63     use_CNES = args.use_CNES
64
65 # READ variables
66 # Read tstart and tstop depending on the type of input file
67 if use_CNES:
68     t1 = float(config['VARIABLES']['TSTART_CNES'])
69     t2 = float(config['VARIABLES']['TSTOP_CNES'])
70     if args.tstart is not None:
71         t1 = args.tstart
72     if args.tstop is not None:
73         t2 = args.tstop
74 else:
75     t1 = config['VARIABLES']['TSTART']
76     t2 = config['VARIABLES']['TSTOP']
77     if args.tstart is not None:
78         t1 = args.tstart
```

```
79      if args.tstop is not None:
80          t2 = args.tstop
81
82  # Read particle flag to check if particle simulation is to be performed
83  particles = config.getboolean('VARIABLES', 'PARTICLES')
84
85  # Read path to store the output
86  outdir = config['PATHS']['OutDir']
87
88  # Set up process mask to only simulate required processes
89  CXB = config.getboolean('VARIABLES', 'CXB')
90  Reflection = config.getboolean('VARIABLES', 'REFLECTION')
91  Albedo = config.getboolean('VARIABLES', 'ALBEDO')
92  Electrons = config.getboolean('VARIABLES', 'ELECTRONS')
93  Protons = config.getboolean('VARIABLES', 'PROTONS')
94  processes = [CXB, Reflection, Albedo, Electrons, Protons]
95
96  # ref date when CNES file is used
97  refdate = Time('1950-01-01T00:00:00.000', format='isot', scale='utc')
98
99  # Over-ride config file variables if commandline arguments are passed
100 if args.particles is not None:
101     particles = args.particles
102 if args.outdir is not None:
103     outdir = args.outdir
104
105 # Define orbit ID
106 if args.use_CNES:
107     deltaT = Time(t2 + refdate.jd, format='jd').unix - Time(t1 + refdate.jd, format='
        jd').unix
108 else:
109     deltaT = Time(t2, format='isot', scale='utc').unix - Time(t1, format='isot',
        scale='utc').unix
110
111 # Append custom output name to the orbit ID if it is provided in the input
112 if args.outname is not None:
113     orbID = "{0:s}_{1:06.1f}_".format(str(t1), deltaT) + args.outname
114 else:
115     orbID = "{0:s}_{1:06.1f}".format(str(t1), deltaT)
116
117
118 # Get all events and photons
119 call = CallProcess(t1, t2, configfile, particles, args.use_CNES)  # Initiate the
        class intance
120 events, photons = call.get_all(processes)  # Call the function to simulate the events
121
122 # Get attitude information for the simulation duration
123 attinfo = call.get_attInfo()
124
125 # Apply pixel convention
126 pixel_conv = config['VARIABLES']['PIXEL_CONV']
127
128 if pixel_conv == 'DPIX':
129     # flip X and Y axis along the centre
130     events[:, 2] = 79 - events[:, 2]
131     events[:, 3] = 79 - events[:, 3]
132 if pixel_conv == 'ECLGP':
133     # interchange X and Y along with a centre flip for new Y axis
134     x_tmp = np.copy(events[:, 2])
```

```
135     y_tmp = np.copy(events[:, 3])
136 
137     events[:, 2] = y_tmp
138     events[:, 3] = 79 - x_tmp
139 
140 # Convert array into astropy table
141 events = Table(events, names=('eventID', 'energy', 'X', 'Y', 'effMult', 'procID'),
142                dtype=['f8', 'f4', 'uint8', 'uint8', 'uint16', 'uint8'])
143 events.sort('eventID')
144 events['eventID'].unit = 's'
145 events['energy'].unit = 'keV'
146 photons = Table(np.array(photons), names=('procID', 'photIndex'), dtype=['uint8', '
    uint32'])
147 
148 # Create common header
149 hdr = fits.Header()
150 if args.use_CNES:
151     hdr['TSTART_CNES'] = (t1, 'Start time in CNES days')
152     hdr['TSTOP_CNES'] = (t2, 'Stop time in CNES days')
153     hdr['TSTART_ISOT'] = (Time(t1 + refdate.jd, format='jd').isot, 'Start time in ISO
     8601 format')
154     hdr['TSTOP_ISOT'] = (Time(t2 + refdate.jd, format='jd').isot, 'Stop time in ISO
    8601 format')
155     hdr['TSTART_MJD'] = (Time(t1 + refdate.jd, format='jd').mjd, 'Start time in mean
    julian days')
156     hdr['TSTOP_MJD'] = (Time(t2 + refdate.jd, format='jd').mjd, 'Stop time in mean
    julian days')
157 else:
158     hdr['TSTART_CNES'] = (Time(t1, format='isot').jd - refdate.jd, 'Start time in
    CNES days')
159     hdr['TSTOP_CNES'] = (Time(t2, format='isot').jd - refdate.jd, 'Stop time in CNES
    days')
160     hdr['TSTART_ISOT'] = (t1, 'Start time in ISO 8601 format')
161     hdr['TSTOP_ISOT'] = (t2, 'Stop time in ISO 8601 format')
162     hdr['TSTART_MJD'] = (Time(t1, format='isot', scale='utc').mjd, 'Start time in
    mean julian days')
163     hdr['TSTOP_MJD'] = (Time(t2, format='isot', scale='utc').mjd, 'Stop time in mean
    julian days')
164 
165 hdr['MJDREF'] = (2400000.5, 'MJD reference date; unit: julian days')
166 hdr['CNESREF'] = (refdate.jd, 'CNES reference date; unit: julian days')
167 hdr['EXPOSURE'] = (deltaT, 'Eventfile exposure; unit : seconds')
168 hdr['PIXEL_CONV'] = (pixel_conv, 'Pixel convention used')
169 hdr['comment'] = 'ISO 8601 format is given as YYYY:MM:DDTHH:MM:SS.SSS'
170 hdr['comment'] = 'Pixel convention types: G4 (geant), DPIX (ELS), ECLGP (pipeline)'
171 
172 # Store the att, event and photon idx files
173 
174 # Create attitude HDUList
175 attHDU = fits.HDUList()
176 attHDU.append(fits.PrimaryHDU())
177 attHDU.append(fits.BinTableHDU(data=attinfo, header=hdr, uint=True, name='ATTINFO'))
178 
179 # Create attitude file specific header comments
180 atthdr = attHDU[1].header
181 atthdr.comments['TTYPE1'] = 'Time of parameter calculation (wrt TSTART)'
182 atthdr.comments['TFORM1'] = 'Real*8 (double precision)'
183 atthdr.comments['TTYPE2'] = 'Earth theta in the FoV frame'
184 atthdr.comments['TFORM2'] = 'Real*4 (floating point)'
```

```
185 atthdr.comments['TTYPE3'] = 'Earth phi in the FoV frame'
186 atthdr.comments['TFORM3'] = 'Real*4 (floating point)'
187 atthdr.comments['TTYPE4'] = 'Earth roll in the FoV frame (defn : see below)'
188 atthdr.comments['TFORM4'] = 'Real*4 (floating point)'
189 atthdr.comments['TTYPE5'] = 'Satellite altitude'
190 atthdr.comments['TFORM5'] = 'Real*4 (floating point)'
191 atthdr.comments['TTYPE6'] = 'Right Ascension of ECL pointing (X-) axis'
192 atthdr.comments['TFORM6'] = 'Real*4 (floating point)'
193 atthdr.comments['TTYPE7'] = 'Declination of ECL pointing (X-) axis'
194 atthdr.comments['TFORM7'] = 'Real*4 (floating point)'
195 atthdr.comments['TTYPE8'] = 'Right Ascension of ECL Y-axis (in DPIX plane)'
196 atthdr.comments['TFORM8'] = 'Real*4 (floating point)'
197 atthdr.comments['TTYPE9'] = 'Declination of ECL Y-axis (in DPIX plane)'
198 atthdr.comments['TFORM9'] = 'Real*4 (floating point)'
199 atthdr.comments['TTYPE10'] = 'Image roll in FoV frame (defn : see below)'
200 atthdr.comments['TFORM10'] = 'Real*4 (floating point)'
201 atthdr.comments['TTYPE11'] = 'Quaternion 0'
202 atthdr.comments['TFORM11'] = 'Real*8 (double precision)'
203 atthdr.comments['TTYPE12'] = 'Quaternion 1'
204 atthdr.comments['TFORM12'] = 'Real*8 (double precision)'
205 atthdr.comments['TTYPE13'] = 'Quaternion 2'
206 atthdr.comments['TFORM13'] = 'Real*8 (double precision)'
207 atthdr.comments['TTYPE14'] = 'Quaternion 3'
208 atthdr.comments['TFORM14'] = 'Real*8 (double precision)'
209 atthdr.comments['TTYPE15'] = 'Satellite position (J2000 frame) X-coord'
210 atthdr.comments['TFORM15'] = 'Real*8 (double precision)'
211 atthdr.comments['TTYPE16'] = 'Satellite position (J2000 frame) Y-coord'
212 atthdr.comments['TFORM16'] = 'Real*8 (double precision)'
213 atthdr.comments['TTYPE17'] = 'Satellite position (J2000 frame) Z-coord'
214 atthdr.comments['TFORM17'] = 'Real*8 (double precision)'
215 atthdr.comments['TTYPE18'] = 'SAA flag; 1=inside SAA, 0=outside SAA'
216 atthdr.comments['TFORM18'] = 'Integer (unsigned integer)'
217 atthdr['comment'] = 'Earth roll : Angle between the local North (in Geographical
    frame) and'
218 atthdr['comment'] = 'the ECL Y axis'
219 atthdr['comment'] = 'Image roll : The angle between the North (in Celestial frame)
    and'
220 atthdr['comment'] = 'the ECL Z axis, counted positive to the East'
221 atthdr['comment'] = 'For the detail definition of both roll angles see the code
    documentation'
222
223 # write the attitude file
224 attHDU.writeto(outdir + "/attitude_{0:s}.fits".format(orbID), overwrite=True)
225
226 # Generate event file HDU list
227 eventsHDU = fits.HDUList()
228 eventsHDU.append(fits.PrimaryHDU())
229 eventsHDU.append(fits.BinTableHDU(data=events, header=hdr, uint=True, name='tupleDpix
    '))
230
231 # Create event file specific header comments
232 evthdr = eventsHDU[1].header
233 evthdr.comments['TTYPE1'] = 'Time of each event relative to the TSTART'
234 evthdr.comments['TFORM1'] = 'Real*8 (double precision)'
235 evthdr.comments['TTYPE2'] = 'Detected energy'
236 evthdr.comments['TFORM2'] = 'Real*4 (floating point)'
237 evthdr.comments['TTYPE3'] = 'DPIX position - pixel X'
238 evthdr.comments['TFORM3'] = 'Integer (unsigned integer)'
239 evthdr.comments['TTYPE4'] = 'DPIX position - pixel Y'
```

```
evthdr.comments['TFORM4'] = 'Integer (unsigned integer)'
evthdr.comments['TTYPE5'] = 'Event multiplicity'
evthdr.comments['TFORM5'] = 'Integer (unsigned integer)'
evthdr.comments['TTYPE6'] = 'Origin of the event (CXB, Ref, etc.)'
evthdr.comments['TFORM6'] = 'Integer (unsigned integer)'
evthdr['comment'] = 'procID key : CXB=0; Ref=1; Alb=2; SAA Elec=3; SAA Prot=4; src=5;
    grb=6'

# Write the event file
eventsHDU.writeto(outdir + "/events_{0:s}.fits".format(orbID), overwrite=True)

# Create primaries HDUList
photonsHDU = fits.HDUList()
photonsHDU.append(fits.PrimaryHDU())
photonsHDU.append(fits.BinTableHDU(data=photons, header=hdr, uint=True, name='
    tupleSource'))
phothdr = photonsHDU[1].header
phothdr.comments['TTYPE1'] = 'Origin of the event (CXB, Ref, etc.)'
phothdr.comments['TFORM1'] = 'Integer (unsigned integer)'
phothdr.comments['TTYPE2'] = 'Database event ID'
phothdr.comments['TFORM2'] = 'Integer*8 (integer long)'
phothdr['comment'] = 'procID key : CXB=0; Ref=1; Alb=2; SAA Elec=3; SAA Prot=4;'

# Write the priamries file
photonsHDU.writeto(outdir + "/primaries_{0:s}.fits".format(orbID), overwrite=True)
```

Listing B.3 – background.py

```
"""

Primary module to handle input, output and processing of the PIRA simulations

"""

import glob
from configparser import ConfigParser, BasicInterpolation

import numpy as np
from scipy.interpolate import interp1d
from astropy.io import fits
from astropy.table import Table, Column, hstack
from astropy.time import Time
from astropy.utils import iers
import astropy.coordinates as coo
import astropy.units as u
from pyquaternion import Quaternion

import mode_list as mult

# Overwrite datatable path for iers data
iers.Conf.iers_auto_url.set('ftp://cddis.gsfc.nasa.gov/pub/products/iers/finals2000A.
    all')


class CallProcess(object):
    """Class to handle background processing.

        Init call to the class selects necessary part of the attitude
        file based on give start and stop time.
```

```
31
32          Parameters
33          ----------
34          tstart : Start time
35          tstop : Stop time
36          configfile : configfile path string
37
38          Keyword Arguments
39          -----------------
40          particles : Flag to decide particle calculation (default=False)
41      """
42
43      def __init__(self, tstart, tstop, configfile, particles=False, use_CNES=False):
44          """Init call to access initialise CallProcess"""
45
46          # Read Config file
47          self.configfile = configfile
48          self.config = ConfigParser(interpolation=BasicInterpolation())
49          self.config.read(self.configfile)
50          self.use_CNES = use_CNES  # Flag to use CNES input file
51
52          # Get attitude parameteres in .att file is used
53          if not self.use_CNES:
54              # Parameters and File Reading
55              self.particles = particles
56
57              # Select correct attitude file (with or without flux information) based
    on whether SAA particle
58              # simulation is selected or not
59              if not self.particles:
60                  attpath = self.config['PATHS']['Attitude']
61                  self.att_hdu = fits.open(attpath, mmap=True)
62              else:
63                  attpath = self.config['PATHS']['Att_WFlux']
64                  self.att_hdu = fits.open(attpath, mmap=True)
65                  check = (self.att_hdu[1].data['ISOT'][0] <= tstart) & (self.att_hdu
    [1].data['ISOT'][-1] >= tstop)
66                  if not check:
67                      print("Particle Info Doesn't Exist for Given time, Setting
    Particle Flag to False")
68                      self.particles = False
69                      attpath = self.config['PATHS']['Attitude']
70                      self.att_hdu = fits.open(attpath, mmap=True)
71
72              # Attitude file initialisation and interpolation
73              # Select desired time
74              self.sel_att = (self.att_hdu[1].data['ISOT'] >= tstart) & (self.att_hdu
    [1].data['ISOT'] <= tstop)
75              t_orb = self.att_hdu[1].data['ISOT'][self.sel_att]
76
77              # Adjust time such that orbit starts at zero
78              time = (np.round(Time(t_orb, format='isot', scale='utc').unix)
79                      - np.round(Time(t_orb[0], format='isot', scale='utc').unix))
80
81              # Extract attitude parameters and create attitude table to store in
    memory to be accessed by the
82              # simulation classes.
83              self.att = Table()
84              self.att.add_column(Column(time, name='Time'))
```

```
85
86          # Extract data from attitude file corresponding to given duration
87          theta = np.radians(self.att_hdu[4].data['ETheta'][self.sel_att])
88          phi = np.radians(self.att_hdu[4].data['EPhi'][self.sel_att])
89          roll = np.radians(self.att_hdu[4].data['ERoll'][self.sel_att])
90          lat = np.radians(self.att_hdu[3].data['Latitude'][self.sel_att])
91          lon = np.radians(self.att_hdu[3].data['Longitude'][self.sel_att])
92          lon[lon < 0.] = lon[lon < 0.] + 2*np.pi
93          alt = self.att_hdu[3].data['Altitude'][self.sel_att]*1e5
94
95          # Add Columns to the attitude table with the extracted data
96          self.att.add_column(Column(theta.round(5), name='E_Theta'))
97          self.att.add_column(Column(phi.round(5), name='E_Phi'))
98          self.att.add_column(Column(roll.round(5), name='E_Roll'))
99          self.att.add_column(Column(lat.round(5), name='Lat'))
100         self.att.add_column(Column(lon.round(5), name='Lon'))
101         self.att.add_column(Column(alt.round(0), name='Alt'))
102
103         # Append SAA particle flux information to the table if SAA particle flux
    simulation is
104         # to be carried out
105         if self.particles:
106             # Store flux and energy infromation in arrays
107             self.ebins_elec = self.att_hdu[7].data['Energy']
108             self.ebins_prot = self.att_hdu[9].data['Energy']
109
110             # Interpolate flux to new time binning
111             interp_fluxelec = interp1d(time, self.att_hdu[8].data[self.sel_att,
    :], axis=0)
112             self.fluxint_elec = interp_fluxelec(self.att['Time'])
113
114             interp_fluxprot = interp1d(time, self.att_hdu[10].data[self.sel_att,
    :], axis=0)
115             self.fluxint_prot = interp_fluxprot(self.att['Time'])
116
117     # If CNES file is used, process the input data to convert to the attitude
    table using dedicated functions.
118     # defined in this module
119     else:
120         # Read CNES file names
121         attdir = self.config['PATHS']['AttDir']
122         attfile = self.config['PATHS']['CNES_att']
123         orbfile = self.config['PATHS']['CNES_orb']
124
125         # Load input data from the files
126         attinfo = np.loadtxt(attfile, usecols=(0, 1, 2, 3, 4))
127         orbinfo = np.loadtxt(orbfile, usecols=(0, 1, 2, 3, 7))
128
129         # Convert the CNES input params to necessary input params for the
    simulation
130         get_output = self.__convert_CNES(attdir, attinfo, orbinfo, tstart, tstop)
131
132         # Store the necessary parameters in the table format and load them into
    the memory
133         # Var other than self.att are necessary to dump attitude information (
    func get_attInfo())
134         self.att, self.quat, self.Xdirn, self.Ydirn, self.Sky_roll, self.satpos =
     get_output
135
```

```
136     def get_all(self, processes):
137         """Function to return all background events and photons
138            in given time interval
139 
140             Parameters
141             ----------
142             processes : Boolean array with information about which
143                         processes to compute. The computation is performed
144                         only if the value corresponding to the index which is
145                         equal to the ProcessID it true. For processID mapping
146                         see below.
147 
148             Returns
149             -------
150             event : (N,6) array as one DPIX event. Here 'N' is the total
151                     number of events detected in DPIX for given time
152                     interval
153 
154                     Each column corresponds to,
155                     col0 : EventID (time)
156                     col1 : Energy (keV)
157                     col2 : Pixel X
158                     col3 : Pixel Y
159                     col4 : Multiplicity
160                     col5 : ProcessID (which physical process generated it)
161 
162                     ProcessID mapping,
163                     CXB = 0, Reflection = 1, Albedo = 2,
164                     Electrons = 3, Protons = 4
165 
166             photons : (n,2) array with photon/particle database index and
167                       ProcessID. Here 'n' is total number of incident
168                       photons/particles
169 
170             Note : In general N > n as one photon/particle can create
171                     multiple intercations
172         """
173 
174         # Initiate empty arrays to be appended by each process call
175         events = np.array([])
176         photons = np.array([])
177 
178         # Call to CXB simulation if required by the user input
179         if processes[0]:
180             cxbc = mult.cxb.CXB(self.att, self.configfile)
181             events_cxb, photons_cxb = cxbc.cxb_photlist()
182 
183             events = events_cxb
184             photons = photons_cxb
185 
186         # Call to Reflection simulation if required by the user input
187         if processes[1]:
188             refl = mult.reflection.REFLECTION(self.att, self.configfile)
189             events_refl, photons_refl = refl.refl_photlist()
190 
191             if events.any():
192                 events = np.vstack((events, events_refl))
193                 photons = np.vstack((photons, photons_refl))
194             else:
```

```
                events = events_refl
                photons = photons_refl

        # Call to Albedo simulation if required by the user input
        if processes[2]:
            alb = mult.albedo.ALBEDO(self.att, self.configfile)
            events_alb, photons_alb = alb.albedo_photlist()

            if events.any():
                events = np.vstack((events, events_alb))
                photons = np.vstack((photons, photons_alb))
            else:
                events = events_alb
                photons = photons_alb

        # Call to SAA electrons simulation if required by the user input
        if processes[3] and self.particles:
            elec = mult.electrons.ELECTRONS(self.att['Time'], self.fluxint_elec,
                                            self.ebins_elec, self.configfile)
            events_elec, photons_elec = elec.elec_photlist()

            if events.any():
                events = np.vstack((events, events_elec))
                photons = np.vstack((photons, photons_elec))
            else:
                events = events_elec
                photons = photons_elec

        # Call to SAA protons simulation if required by the user input
        if processes[4] and self.particles:
            prot = mult.protons.PROTONS(self.att['Time'], self.fluxint_prot,
                                        self.ebins_prot, self.configfile)
            events_prot, photons_prot = prot.prot_photlist()

            if events.any():
                events = np.vstack((events, events_prot))
                photons = np.vstack((photons, photons_prot))
            else:
                events = events_prot
                photons = photons_prot

        return events, photons

    def get_attInfo(self):
        """Function to return the attitude information for the given PIRA simulation
    duration

            Parameters
            ----------
            None.

            Returns
            ---------
            All the attitude parameters for the given simulation duration.
        """

        # Initiate table to be populated by the paramaters for the given duration
        attinfo = self.att.copy()
        attinfo.remove_columns(['Lat', 'Lon'])  # Remove latitude and longitude
```

```
    inforation
253
254         # Add Earth angle information
255         attinfo['E_Theta'] = np.degrees(attinfo['E_Theta'])
256         attinfo['E_Phi'] = np.degrees(attinfo['E_Phi'])
257         attinfo['E_Roll'] = np.degrees(attinfo['E_Roll'])
258
259         # Extraact more parameters based on the type of the input att file (fits
    format or CNES format)
260         if not self.use_CNES:
261             Xdirn = Table([self.att_hdu[2].data['RA(X)'][self.sel_att], self.att_hdu
    [2].data['DEC(X)'][self.sel_att]],
262                           names=['RA(X)', 'DEC(X)'])
263             Ydirn = Table([self.att_hdu[2].data['RA(Y)'][self.sel_att], self.att_hdu
    [2].data['DEC(Y)'][self.sel_att]],
264                           names=['RA(Y)', 'DEC(Y)'])
265             skyroll = Table([self.att_hdu[2].data['Roll'][self.sel_att]], names=['
    Roll'])
266             quat_table = Table(self.att_hdu[5].data[self.sel_att])
267             satpos = Table(self.att_hdu[6].data[self.sel_att], dtype=['f8', 'f8', 'f8
    '])
268             saa = Table(np.array([self.att_hdu[3].data['SAA'][self.sel_att]]).T,
    names=['SAA'], dtype=['uint8'])
269             sat_pos = hstack([satpos, saa])
270         else:
271             Xdirn = Table(self.Xdirn, names=['RA(X)', 'DEC(X)'])
272             Ydirn = Table(self.Ydirn, names=['RA(Y)', 'DEC(Y)'])
273             skyroll = Table(np.degrees(self.Sky_roll), names=['Roll'])
274             quat_table = Table(self.quat, names=['Q0', 'Q1', 'Q2', 'Q3'])
275             sat_pos = Table(self.satpos, names=['x', 'y', 'z', 'SAA'], dtype=['f8', '
    f8', 'f8', 'uint8'])
276
277         # Append all attitude parameters into one sigle table to be returned to the
    main script for output
278         attinfo = hstack([attinfo, Xdirn, Ydirn, skyroll, quat_table, sat_pos])
279
280         # Define unit (where applicable) and dtype of the columns
281         attinfo['Time'].unit = 's'
282         for colname in attinfo.colnames[1:4]:  # unit and dtype for angle columns
283             attinfo[colname].unit = 'deg'
284             attinfo[colname] = attinfo[colname].astype('f4')
285
286         attinfo['Alt'] = (attinfo['Alt']/1e5).astype('f4')
287         attinfo['Alt'].unit = 'km'
288         attinfo['Alt'].name = 'Altitude'
289
290         for colname in attinfo.colnames[5:10]:  # unit and dtype for angle columns
291             attinfo[colname].unit = 'deg'
292             attinfo[colname] = attinfo[colname].astype('f4')
293
294         for colname in attinfo.colnames[14:17]:  # unit for satellite position
    columns
295             attinfo[colname].unit = 'm'
296
297         return attinfo
298
299     def __get_timeinterval(self):
300         '''
301             Function to get tbounds of slew durations in the CNES file
```

```
        '''

        # Define tstart,tstop array
        t_bounds = np.empty((len(self.slew_files), 2))

        # read each file and store the start and end time
        for i, fname in enumerate(self.slew_files):
            f_read = np.loadtxt(fname, usecols=(0,))
            t_bounds[i] = f_read[[0, -1]]

        return t_bounds

    def __get_delidx(self, attinfo, tbounds):
        '''
            Function to get idx of common time in ephm_att file
            and in the maneuver files. These indices will be deleted
            from the ephm_att file
        '''
        # Define idx array
        idx = []

        # Get time values
        time = attinfo[:, 0]

        # Correct for common time intervals
        for i in range(tbounds.shape[0]):
            tmp_id = np.where((time >= tbounds[i, 0]) & (time <= tbounds[i, 1]))[0]
            idx.extend(list(tmp_id))

        return idx

    def __merge_slew(self, attinfo):
        '''
            Merge ephm_att and maneuver files and return
            a final sorted att file.
        '''
        # Get del indices
        tbounds = self.__get_timeinterval()
        idx = self.__get_delidx(attinfo, tbounds)

        # create del mask
        mask = np.ones(attinfo.shape[0], dtype=bool)
        mask[idx] = False

        # Delete masked indices
        new_att = attinfo[mask]

        # read each file
        for fname in self.slew_files:
            slew = np.loadtxt(fname, usecols=(0, 1, 2, 3, 4))
            new_att = np.vstack((new_att, slew))

        # sort
        attinfo_out = new_att[new_att[:, 0].argsort()]

        return attinfo_out

    def __convert_CNES(self, attdir, attinfo, orbinfo, tstart, tstop):
        '''
```

```
361             Method to convert CNES file params to earth angles and usable by this
    program
362         '''
363
364         # Detect if detail slew maneuvers available
365         self.slew_files = glob.glob(attdir + "/maneuver*")
366
367         # If slew maneuvers exist, merge them in attinfo and interpolate sat
    positions
368         if self.slew_files:
369             print("Detail slew data found, merging info")
370             self.slew_files.sort()
371             attinfo = self.__merge_slew(attinfo)  # Merge detail slew data
372
373             # Interpolate sat postion for the slew data
374             fit = interp1d(orbinfo[:, 0], orbinfo[:, 1:], axis=0)
375             orbinfo = np.zeros((attinfo[:, 0].shape[0], 5))
376             orbinfo[:, 0] = np.copy(attinfo[:, 0])
377             orbinfo[:, 1:] = fit(attinfo[:, 0])
378             orbinfo[:, 4] = orbinfo[:, 4].round()
379
380         # Get time interval of interest and convert to seconds
381         tmask_att = (attinfo[:, 0] >= tstart) & (attinfo[:, 0] <= tstop)
382         tmask_orb = (orbinfo[:, 0] >= tstart) & (orbinfo[:, 0] <= tstop)
383         refdate = Time('1950-01-01T00:00:00.000', format='isot', scale='utc')
384         times_jd = Time(attinfo[tmask_att, 0] + refdate.jd, format='jd')
385         times = times_jd.unix.round()
386         times = times - times[0]  # Get relative time to tstart
387
388         # Apply tmask on orbfile and attfile
389         orbinfo = orbinfo[tmask_orb]
390         attinfo = attinfo[tmask_att]
391
392         # Get quaternions
393         quat = attinfo[:, 1:]
394         N = quat.shape[0]
395
396         # Define ECL ref
397         XECL = np.array([1, 0, 0])
398         YECL = np.array([0, 1, 0])
399         ZECL = np.array([0, 0, 1])
400
401         # Initiate emtpy arrays
402         X_dirn = np.empty((N, 2))
403         Y_dirn = np.empty((N, 2))
404         Sky_roll = np.empty((N, 1))
405         E_dirn = np.empty((N, 3))
406
407         print("Converting Quaternions to angle")
408
409         for i in range(N):
410             Q = Quaternion(quat[i, :])
411
412             # Transform Pointing axis from ECL frame to J2000
413             cartrepXI = coo.CartesianRepresentation(Q.rotate(XECL))
414             XI = coo.GCRS(cartrepXI, obstime='J2000')
415
416             # Transform first axis (Y) in image plane from ECL frame to J2000
417             cartrepYI = coo.CartesianRepresentation(Q.rotate(YECL))
```

```
            YI = coo.GCRS(cartrepYI, obstime='J2000')

            # Store Pointing Directions
            X_dirn[i, 0] = XI.ra.value
            X_dirn[i, 1] = XI.dec.value

            # Store first axis (Y) axis directions
            Y_dirn[i, 0] = YI.ra.value
            Y_dirn[i, 1] = YI.dec.value

            # Get Sky roll (Definition in doc : SVOM-FSC-GPT-ECPI-ADD)
            # Define local north vector in celestial frame
            North_sky = np.array([-np.sin(XI.dec)*np.cos(XI.ra),
                                  -np.sin(XI.dec)*np.sin(XI.ra),
                                  np.cos(XI.dec)])
            North_sky = coo.CartesianRepresentation(North_sky)

            # Compute East vector
            East_sky = -1*XI.cartesian.cross(North_sky)

            # Find Z-axis
            cartrepZI = coo.CartesianRepresentation(Q.rotate(ZECL))  # Transform Y-
    dirn from ECL to GCRS J2000 frame
            ZI = coo.GCRS(cartrepZI, obstime='J2000')

            # Compute the sky roll angle
            skyroll = -1*np.arctan2(ZI.cartesian.dot(East_sky), ZI.cartesian.dot(
    North_sky))

            # Correct the angle convention (0 to 360 deg)
            if skyroll < 0:
                skyroll += 2*np.pi*u.rad

            Sky_roll[i, 0] = skyroll.value

            # Get Earth Angles
            sat_gcrs = np.array([orbinfo[i, 1], orbinfo[i, 2], orbinfo[i, 3]])
            mag = np.sqrt(sat_gcrs.dot(sat_gcrs))
            Earth_gcrs = -sat_gcrs/mag
            Earth_sat = Q.inverse.rotate(Earth_gcrs)

            # Store E_dirn
            # Compute Earth theta
            E_dirn[i, 0] = np.arccos(Earth_sat[0])

            # Compute Earth phi
            phi = np.arctan2(Earth_sat[2], Earth_sat[1])
            if phi < 0:
                phi += 2*np.pi

            E_dirn[i, 1] = phi

            # Get Earth Roll - Defined as dirn of Local North projected in the ECL YZ
     plane,
            # measured from ECL Y-axis (increasinf towards Z-axis).

            # Get local north lat/lon in GCRS coordinate.
            gcrs_lat = np.arcsin(sat_gcrs[2]/mag)
            gcrs_lon = np.arctan2(sat_gcrs[1], sat_gcrs[0])
```

```
            if gcrs_lon < 0:
                gcrs_lon += 2*np.pi

            # Find local north vector in GCRS J2000. The direction is same in ECL frame
            North_gcrs = np.array([-np.sin(gcrs_lat)*np.cos(gcrs_lon),
                                   -np.sin(gcrs_lat)*np.sin(gcrs_lon),
                                   np.cos(gcrs_lat)])

            # Tranform the direction coordinates in ECL frame
            North_sat = Q.inverse.rotate(North_gcrs)

            # Find the angle of the north direction projection in the ECL YZ frame.
            roll = np.arctan2(North_sat[2], North_sat[1])

            # Correct the angle convention (0 to 360 deg)
            if roll < 0:
                roll += 2*np.pi

            E_dirn[i, 2] = roll

        print("Quaternion conversion finished\n")

        # Get satellite position
        cartrep = coo.CartesianRepresentation(orbinfo[:, 1], orbinfo[:, 2], orbinfo[:, 3], unit=u.m)
        gcrs = coo.GCRS(cartrep, obstime=times_jd)
        itrs = gcrs.transform_to(coo.ITRS(obstime=times_jd))

        print("Lattitude and Longitude computation finished\n")

        # Create attitude table
        att = Table()
        att.add_column(Column(times, name='Time'))

        # Add Columns to the attitude table
        att.add_column(Column(E_dirn[:, 0].round(5), name='E_Theta'))
        att.add_column(Column(E_dirn[:, 1].round(5), name='E_Phi'))
        att.add_column(Column(E_dirn[:, 2].round(5), name='E_Roll'))
        att.add_column(Column(np.radians(itrs.earth_location.lat).round(5), name='Lat'))
        att.add_column(Column(np.radians(itrs.earth_location.lon).round(5), name='Lon'))
        att.add_column(Column(itrs.earth_location.height*1e2, name='Alt'))

        att['Lon'][att['Lon'] < 0] = att['Lon'][att['Lon'] < 0] + 2*np.pi

        return (att, quat, X_dirn, Y_dirn, Sky_roll, orbinfo[:, 1:])
```

Listing B.4 – cxb.py

```
"""
    Module to compute CXB events using PIRA
"""

from configparser import ConfigParser, BasicInterpolation

import numpy as np
from astropy.io import fits
```

```
9  from scipy.integrate import simps
10 from scipy.interpolate import interp1d
11
12 from common import functions as f
13
14
15 class CXB(object):
16     '''
17         Class to handle CXB computations.
18     '''
19
20     def __init__(self, att, configfile):
21         '''
22             Init call to the CXB class.
23         '''
24
25         # Read config file
26         config = ConfigParser(interpolation=BasicInterpolation())
27         config.read(configfile)
28
29         # Read and store algorithm parameters
30         self.Re = config.getfloat('VARIABLES', 'R_EARTH') * 1e5  # Earth radius in cm
31         H_atm = config.getfloat('VARIABLES', 'H_ATM') * 1e5  # Height of the
    atmosphere in cm
32         self.Reff = self.Re + H_atm
33
34         # Read G4 Files
35         cxbsrc = config['PATHS']['CXBSRC']
36         cxbdpix = config['PATHS']['CXBDPIX']
37         sourceHDU = fits.open(cxbsrc, mmap=True)
38         dpixelHDU = fits.open(cxbdpix, mmap=True)
39
40         # Read and store database parameters
41         # G4 Energy Variables
42         Emin_sim = sourceHDU[1].header['EMIN']
43         Emax_sim = sourceHDU[1].header['EMAX']
44         Ecut_sim = sourceHDU[1].header['ECUT']
45
46         # G4 Geometry Variables
47         Rp_CXB = sourceHDU[1].header['RADIUS']  # Source radius
48         Theta_max = sourceHDU[1].header['THETAMAX']   # Opening angle
49
50         # G4 Coord offsets
51         self.x0 = config.getfloat('VARIABLES', 'X_OFFSET')
52         self.y0 = config.getfloat('VARIABLES', 'Y_OFFSET')
53         self.z0 = config.getfloat('VARIABLES', 'Z_OFFSET')
54
55         # Get avg. detection efficiency
56         N_SIMU = sourceHDU[1].header['NPH_SIMU']
57         N_INTE = sourceHDU[1].header['NPH_INTE']
58         Eff_CXB = np.float32(N_INTE / N_SIMU)
59
60         # Integrate over energy
61         logEbins = np.linspace(np.log10(Emin_sim), np.log10(Emax_sim), 1000)  #
    Define energy bins to integrate
62         Ebins = 10**logEbins
63         vMorettiGruber = np.vectorize(f.MorettiGruber)  # Vectorize spectral function
64         N_E = vMorettiGruber(Ebins, Ecut=Ecut_sim)  # Get photon spectrum
65         fluxCXB = simps(N_E, Ebins)  # Integrate the spectrum over the energy band to
```

```
    get the total no. of photons

        # Find CXB rate
        self.rate_cxb = np.float32(4 * np.pi**2 * Rp_CXB**2 * fluxCXB * np.sin(
    Theta_max)**2 * Eff_CXB)

        # Read Geant4 simulation data
        self.source = sourceHDU[1].data
        self.source['phi'][self.source['phi'] < 0] = (self.source['phi'][self.source[
    'phi'] < 0]
                                                      + 360)
        self.source['theta'] = np.radians(self.source['theta'])
        self.source['phi'] = np.radians(self.source['phi'])
        self.dpixel = dpixelHDU[1].data
        self.data_max = self.source['runID'].size
        self.source_idx = np.random.permutation(self.data_max)

        # Map src to dpix index
        self.idxmap = np.zeros((self.source['eventID'].size, 2))
        evt, self.idxmap[:, 0], self.idxmap[:, 1] = np.unique(self.dpixel['eventID'],
                                                              return_index=True,
                                                              return_counts=True)
        self.idxmap = self.idxmap.astype(int)
        del evt

        # set attitude file
        self.att = att

        # define interpolation functions for Earth angles and sat altitude
        self.interp_theta = interp1d(self.att['Time'], self.att['E_Theta'],
    fill_value='extrapolate')
        self.interp_phi = interp1d(self.att['Time'], np.unwrap(self.att['E_Phi']),
    fill_value='extrapolate')
        self.interp_alt = interp1d(self.att['Time'], self.att['Alt'], fill_value='
    extrapolate')

    def cxb_photlist(self):
        '''
            Function to return the CXB event list.
        '''

        # Initialise global arrays
        idx_global = []
        t_global = []

        # Compute photons for delta T
        tend = self.att['Time'][0]
        p = 0
        cycle = 0

        # Loop over consecutive time bins in the attitude file
        for i in range(self.att['Time'][:-1].size):

            # Assign time bounds
            t1 = self.att['Time'][i]
            t2 = self.att['Time'][i + 1]
            orb_delta = t2 - t1

            # Calculate Number of Photons
```

```
            delT = t1 + orb_delta - tend
            Nphot = int(delT * self.rate_cxb)

            # Correct for fractional rate
            mu = np.random.uniform(0, 1, 1)
            if mu <= ((delT * self.rate_cxb) - Nphot):
                Nphot += 1

            # Get Poisson time
            dt = np.random.exponential(scale=1/self.rate_cxb, size=Nphot)
            t = dt.cumsum() + tend
            del dt

            # Interpolate coordinates for all times

            Etheta_t = self.interp_theta(t)
            Ephi_t = self.interp_phi(t) % (2*np.pi)  # Take modulo to correct for
    interpolation above 2*pi
            Ealt_t = self.interp_alt(t)

            # Select Photons
            Nrem = self.data_max - p

            # Check remaining number of photons in order to not repeat them
            if Nphot > Nrem:
                idx = self.source_idx[p:]
                self.source_idx = np.random.permutation(self.data_max)
                idx = np.append(idx, self.source_idx[:Nphot - Nrem])
                p = Nphot - Nrem
                cycle += 0
            else:
                idx = self.source_idx[p:p+Nphot]
                p += Nphot

            # Extract photons from the source file
            src = self.source[idx]

            # Get incoming photon direction
            theta = src['theta']
            phi = src['phi']

            # Get incoming photon source sphere coordinates
            x = src['x'] - self.x0
            y = src['y'] - self.y0
            z = src['z'] - self.z0

            # Find whether photon hits the Earth or not
            Rh = self.Re + Ealt_t
            k = self.Reff / Rh

            del k, Ealt_t  # memory management

            # Find dirn of the Earth in ECL frame
            xe = Rh*np.sin(Etheta_t)*np.cos(Ephi_t)
            ye = Rh*np.sin(Etheta_t)*np.sin(Ephi_t)
            ze = Rh*np.cos(Etheta_t)

            # Unit vector in direction of the photon
            ux = -np.sin(theta)*np.cos(phi)
```

```
177             uy = -np.sin(theta)*np.sin(phi)
178             uz = -np.cos(theta)
179 
180             # Solve Eqn. 11 in Mate et. al. 2019
181             delx = x - xe
182             dely = y - ye
183             delz = z - ze
184 
185             bprime = ux*delx + uy*dely + uz*delz
186             c = delx**2 + dely**2 + delz**2 - self.Reff**2
187             delta = bprime**2 - c
188 
189             # Memory management
190             del theta, phi, Etheta_t, Ephi_t
191             del x, y, z, xe, ye, ze, ux, uy, uz, delx, dely, delz, c
192 
193             # Create a mask for earth crossing
194             mask = np.zeros(Nphot)
195 
196             # Fill earth mask with photons which cross earth
197             deltamask = delta > 0.
198 
199             lambda1 = -bprime[deltamask] + np.sqrt(delta[deltamask])
200             lambda2 = -bprime[deltamask] - np.sqrt(delta[deltamask])
201 
202             # Find photons which cross the Earth (in order to reject them)
203             earthcrossing = (lambda1 < 0.) | (lambda2 < 0.)
204 
205             mask[np.where(deltamask)[0][earthcrossing]] = 1
206             mask = mask.astype(bool)
207 
208             # Take the negation as we need photons from sky
209             mask = ~mask
210 
211             # Select indexes
212             idx_global = np.append(idx_global, idx[mask])
213             t_global = np.append(t_global, t[mask])
214             tend = t_global[-1]
215 
216         print(cycle)
217 
218         # Select DPIX indices corresponding to the selected photons
219         dpix_idx = self.idxmap[idx_global.astype(int), 0]
220         mult = self.idxmap[idx_global.astype(int), 1]
221 
222         # Memory management
223         del mask, idx, deltamask, lambda1, lambda2, bprime, earthcrossing
224         del self.source, self.source_idx, self.idxmap
225 
226         dpix_idx = np.hstack([list(range(x, y)) for x, y in zip(dpix_idx, dpix_idx +
    mult)])
227 
228         # Store output data in 2-D array to return
229         events = np.zeros((dpix_idx.size, 6))
230         events[:, 0] = np.repeat(t_global, mult)
231         events[:, 1] = self.dpixel[dpix_idx]['energy'].data
232         events[:, 2] = self.dpixel[dpix_idx]['X'].data
233         events[:, 3] = self.dpixel[dpix_idx]['Y'].data
234         events[:, 4] = np.repeat(mult, mult)
```

```
        events[:, 5] = 0

        # Store ID for selected photon
        photon = np.insert(idx_global.reshape(idx_global.size, 1), 0, 0, axis=1)

        print("CXB eventlist returned")
        return events, photon
```

Listing B.5 – reflection.py

```
"""
    Module to compute Reflection events using PIRA
"""

from configparser import ConfigParser, BasicInterpolation

import numpy as np
from astropy.io import fits
from scipy.integrate import simps
from scipy.interpolate import interp1d

from common import functions as f


class REFLECTION(object):
    '''
        Class to handle Reflection computations.
    '''

    def __init__(self, att, configfile):
        '''
            Init call to the Reflection class.
        '''

        # Read config file
        config = ConfigParser(interpolation=BasicInterpolation())
        config.read(configfile)

        # Read and store algorithm parameters
        self.Re = config.getfloat('VARIABLES', 'R_EARTH') * 1e5  # Earth radius in cm
        H_atm = config.getfloat('VARIABLES', 'H_ATM') * 1e5  # Earth radius in cm
        self.Reff = self.Re + H_atm

        # Read G4 Files
        refsrc = config['PATHS']['REFSRC']
        refdpix = config['PATHS']['REFDPIX']
        sourceHDU = fits.open(refsrc, mmap=True)
        dpixelHDU = fits.open(refdpix, mmap=True)

        # Read and store database parameters
        # G4 Energy Variables
        Emin_sim = sourceHDU[1].header['EMIN']
        Emax_sim = sourceHDU[1].header['EMAX']
        Ecut_sim = sourceHDU[1].header['ECUT']

        # G4 Geometry variables
        Rp_Ref = sourceHDU[1].header['RADIUS']  # Source radius
        Theta_max = sourceHDU[1].header['THETAMAX']  # Opening angle

```

```
50          # G4 Coord offsets
51          self.x0 = config.getfloat('VARIABLES', 'X_OFFSET')
52          self.y0 = config.getfloat('VARIABLES', 'Y_OFFSET')
53          self.z0 = config.getfloat('VARIABLES', 'Z_OFFSET')
54
55          # Get avg. detection efficiency
56          N_SIMU = sourceHDU[1].header['NPH_SIMU']
57          N_INTE = sourceHDU[1].header['NPH_INTE']
58          Eff_Ref = np.float32(N_INTE / N_SIMU)
59
60          # Integrate over energy
61          logEbins = np.linspace(np.log10(Emin_sim), np.log10(Emax_sim), 1000)  #
    Define energy bins to integrate
62          Ebins = 10**logEbins
63          vRefl = np.vectorize(f.reflection)  # Vectorise spectral function
64          N_E = vRefl(Ebins, Ecut=Ecut_sim)   # Get photon spectrum
65          fluxREF = simps(N_E, Ebins)  # Integrate the spectrum over the energy band to
     get the total no. of photons
66
67          # Find reflection rate
68          self.rate_refl = np.float32(4 * np.pi**2 * Rp_Ref**2 * fluxREF * np.sin(
    Theta_max)**2 * Eff_Ref)
69
70          # Read Geant4 simulation data
71          self.source = sourceHDU[1].data
72          self.source['phi'][self.source['phi'] < 0] = (self.source['phi'][self.source[
    'phi'] < 0]
73                                                        + 360)
74          self.source['theta'] = np.radians(self.source['theta'])
75          self.source['phi'] = np.radians(self.source['phi'])
76          self.dpixel = dpixelHDU[1].data
77          self.data_max = self.source['runID'].size
78          self.source_idx = np.random.permutation(self.data_max)
79
80          # Map src to dpix index
81          self.idxmap = np.zeros((self.source['eventID'].size, 2))
82          evt, self.idxmap[:, 0], self.idxmap[:, 1] = np.unique(self.dpixel['eventID'],
83                                                                return_index=True,
84                                                                return_counts=True)
85          self.idxmap = self.idxmap.astype(int)
86          del evt
87
88          # set attitude file
89          self.att = att
90
91          # define interpolation functions for Earth angles and sat altitude
92          self.interp_theta = interp1d(self.att['Time'], self.att['E_Theta'],
    fill_value='extrapolate')
93          self.interp_phi = interp1d(self.att['Time'], np.unwrap(self.att['E_Phi']),
    fill_value='extrapolate')
94          self.interp_alt = interp1d(self.att['Time'], self.att['Alt'], fill_value='
    extrapolate')
95
96      def refl_photlist(self):
97          '''
98              Function to return the Reflection event list.
99          '''
100
101         # Initialise global arrays
```

```
        idx_global = []
        t_global = []

        # Compute photons for delta T
        tend = self.att['Time'][0]
        p = 0
        cycle = 0

        # Loop over consecutive time bins in the attitude file
        for i in range(self.att['Time'][:-1].size):

            # Assign time bounds
            t1 = self.att['Time'][i]
            t2 = self.att['Time'][i + 1]
            orb_delta = t2 - t1

            # Calculate Number of Photons
            delT = t1 + orb_delta - tend
            Nphot = int(delT * self.rate_refl)

            # Correct for fractional rate
            mu = np.random.uniform(0, 1, 1)
            if mu <= ((delT * self.rate_refl) - Nphot):
                Nphot += 1

            # Get Poisson time
            dt = np.random.exponential(scale=1/self.rate_refl, size=Nphot)
            t = dt.cumsum() + tend
            del dt

            # Interpolate coordinates for all times

            Etheta_t = self.interp_theta(t)
            Ephi_t = self.interp_phi(t) % (2*np.pi)  # Take modulo to correct for
    interpolation above 2*pi
            Ealt_t = self.interp_alt(t)

            # Select Photons
            Nrem = self.data_max - p

            # Check remaining number of photons in order to not repeat them
            if Nphot > Nrem:
                idx = self.source_idx[p:]
                self.source_idx = np.random.permutation(self.data_max)
                idx = np.append(idx, self.source_idx[:Nphot - Nrem])
                p = Nphot - Nrem
                print(t[0], t[-1])
                cycle += 1
            else:
                idx = self.source_idx[p:p+Nphot]
                p += Nphot

            # Extract photons from the source file
            src = self.source[idx]

            # Get incoming photon direction
            theta = src['theta']
            phi = src['phi']

```

```
160             # Get incoming photon source sphere coordinates
161             x = src['x'] - self.x0
162             y = src['y'] - self.y0
163             z = src['z'] - self.z0
164 
165             # Find whether photon hits the Earth or not
166             Rh = self.Re + Ealt_t
167             k = self.Reff / Rh
168 
169             del k, Ealt_t  # memory management
170 
171             # Find dirn of the Earth in ECL frame
172             xe = Rh*np.sin(Etheta_t)*np.cos(Ephi_t)
173             ye = Rh*np.sin(Etheta_t)*np.sin(Ephi_t)
174             ze = Rh*np.cos(Etheta_t)
175 
176             # Unit vector in direction of the photon
177             ux = -np.sin(theta)*np.cos(phi)
178             uy = -np.sin(theta)*np.sin(phi)
179             uz = -np.cos(theta)
180 
181             # Solve Eqn. 11 in Mate et. al. 2019
182             delx = x - xe
183             dely = y - ye
184             delz = z - ze
185 
186             bprime = ux*delx + uy*dely + uz*delz
187             c = delx**2 + dely**2 + delz**2 - self.Reff**2
188             delta = bprime**2 - c
189 
190             # Memory management
191             del theta, phi, Etheta_t, Ephi_t
192             del x, y, z, xe, ye, ze, ux, uy, uz, delx, dely, delz, c
193 
194             # Create a mask for earth crossing
195             mask = np.zeros(Nphot)
196 
197             # Fill earth mask with photons which cross earth
198             deltamask = delta > 0.
199 
200             lambda1 = -bprime[deltamask] + np.sqrt(delta[deltamask])
201             lambda2 = -bprime[deltamask] - np.sqrt(delta[deltamask])
202 
203             # Find photons which cross the Earth (in order to select them)
204             earthcrossing = (lambda1 < 0.) | (lambda2 < 0.)
205 
206             mask[np.where(deltamask)[0][earthcrossing]] = 1
207             mask = mask.astype(bool)
208 
209             # Select indexes
210             idx_global = np.append(idx_global, idx[mask])
211             t_global = np.append(t_global, t[mask])
212             tend = t_global[-1]
213 
214         print(cycle)
215 
216         # Select DPIX indices corresponding to the selected photons
217         dpix_idx = self.idxmap[idx_global.astype(int), 0]
218         mult = self.idxmap[idx_global.astype(int), 1]
```

```
219
220         # Memory management
221         del mask, idx, deltamask, lambda1, lambda2, bprime, earthcrossing
222         del self.source, self.source_idx, self.idxmap
223
224         dpix_idx = np.hstack([list(range(x, y)) for x, y in zip(dpix_idx, dpix_idx +
    mult)])
225
226         # Store output data in 2-D array to return
227         events = np.zeros((dpix_idx.size, 6))
228         events[:, 0] = np.repeat(t_global, mult)
229         events[:, 1] = self.dpixel[dpix_idx]['energy'].data
230         events[:, 2] = self.dpixel[dpix_idx]['X'].data
231         events[:, 3] = self.dpixel[dpix_idx]['Y'].data
232         events[:, 4] = np.repeat(mult, mult)
233         events[:, 5] = 1
234
235         # Store ID for selected photon
236         photon = np.insert(idx_global.reshape(idx_global.size, 1), 0, 1, axis=1)
237         print("Reflection eventlist returned")
238         return events, photon
```

Listing B.6 – albedo.py

```
1  '''
2      Module to compute Albedo events using PIRA
3  '''
4
5  from configparser import ConfigParser, BasicInterpolation
6  from random import expovariate
7
8  import numpy as np
9  from astropy.io import fits
10 from scipy.integrate import simps
11 from scipy.interpolate import interp1d
12
13 from common import functions as f
14
15
16 class ALBEDO(object):
17     '''
18         Class to handle Albedo computations.
19     '''
20
21     def __init__(self, att, configfile):
22         '''
23             Init call to the Albedo class.
24         '''
25
26         # Read config file
27         config = ConfigParser(interpolation=BasicInterpolation())
28         config.read(configfile)
29
30         # Read and store algorithm parameters
31         self.Re = config.getfloat('VARIABLES', 'R_EARTH') * 1e5
32         H_atm = config.getfloat('VARIABLES', 'H_ATM') * 1e5
33         self.Reff = self.Re + H_atm
34
35         # Number of angular bins for Local and MAP frame
```

```
36          self.nth = config.getint('VARIABLES', 'NTHETA')
37          self.nph = config.getint('VARIABLES', 'NPHI')
38
39          # Variables for alb factor (C in Sazanove et al.)
40          self.mpole_lat = config.getfloat('VARIABLES', 'MPOLE_LAT')
41          self.mpole_lon = config.getfloat('VARIABLES', 'MPOLE_LON')
42          self.phi_sol = config.getfloat('VARIABLES', 'PHI_SOL')
43
44          # Initiate index array for angular bins
45          self.tn = np.arange(0, self.nth, 1)
46          self.pn = np.arange(0, self.nph, 1)
47
48          # Read G4 simulation files
49          albsrc = config['PATHS']['ALBSRC']
50          albdpix = config['PATHS']['ALBDPIX']
51          sourceHDU = fits.open(albsrc, mmap=True)
52          dpixelHDU = fits.open(albdpix, mmap=True)
53
54          # Read and store database parameters
55          # G4 Energy Variable
56          Emin_sim = sourceHDU[1].header['EMIN']
57          Emax_sim = sourceHDU[1].header['EMAX']
58
59          # G4 Geometry Variables
60          Rp_ALB = sourceHDU[1].header['RADIUS']  # Source radius
61          Theta_max = sourceHDU[1].header['THETAMAX']   # Opening angle
62
63          # G4 Coord offsets
64          self.x0 = config.getfloat('VARIABLES', 'X_OFFSET')
65          self.y0 = config.getfloat('VARIABLES', 'Y_OFFSET')
66          self.z0 = config.getfloat('VARIABLES', 'Z_OFFSET')
67
68          # Get avg. detection efficiency
69          N_SIMU = sourceHDU[1].header['NPH_SIMU']
70          N_INTE = sourceHDU[1].header['NPH_INTE']
71          Eff_ALB = np.float32(N_INTE / N_SIMU)
72
73          # Integrate over energy
74          logEbins = np.linspace(np.log10(Emin_sim), np.log10(Emax_sim), 1000)  #
    Define energy bins to integrate
75          Ebins = 10**logEbins
76          N_E = f.albedo(Ebins)  # Get photon spectrum
77          fluxALB = simps(N_E, Ebins)  # Integrate the spectrum over the energy band to
     get the total no. of photons
78
79          # Find constant normalisation factor
80          self.MAP_Norm = np.float32(2 * np.pi * Rp_ALB**2 * np.sin(Theta_max)**2 *
    fluxALB *
81                                     self.nth * self.nph * Eff_ALB)
82
83          # Read Geant4 simulation data
84          self.source = sourceHDU[1].data
85          self.source['phi'][self.source['phi'] < 0] = (self.source['phi'][self.source[
    'phi'] < 0]
86                                                        + 360)
87          self.source['theta'] = np.radians(self.source['theta'])
88          self.source['phi'] = np.radians(self.source['phi'])
89          self.dpixel = dpixelHDU[1].data
90          self.data_max = self.source['runID'].size
```

```
        self.source_idx = np.random.permutation(self.data_max)

        # Map src to dpix index
        self.idxmap = np.zeros((self.source['eventID'].size, 2))
        evt, self.idxmap[:, 0], self.idxmap[:, 1] = np.unique(self.dpixel['eventID'],
                                                              return_index=True,
                                                              return_counts=True)
        self.idxmap = self.idxmap.astype(int)
        del evt

        # set attitude file
        self.att = att

        # define interpolation functions for Earth angles and sat altitude
        self.interp_theta = interp1d(self.att['Time'], self.att['E_Theta'],
    fill_value='extrapolate')
        self.interp_phi = interp1d(self.att['Time'], np.unwrap(self.att['E_Phi']),
    fill_value='extrapolate')
        self.interp_roll = interp1d(self.att['Time'], np.unwrap(self.att['E_Roll']),
    fill_value='extrapolate')
        self.interp_alt = interp1d(self.att['Time'], self.att['Alt'], fill_value='
    extrapolate')

    def albedo_photlist(self):
        '''
            Function to return the Albedo event list.
        '''

        # Initialise global arrays
        idx_global = []
        t_global = []

        # Compute photons for delta T
        tend = self.att['Time'][0]
        p = 0
        cycle = 0

        # Loop over consecutive time bins in the attitude file
        for i in range(self.att['Time'][:-1].size):

            # Assign time bounds
            t1 = self.att['Time'][i]
            t2 = self.att['Time'][i + 1]
            orb_delta = t2 - t1

            # Calculate MAP photons for each angular bin and
            # at start and end time of the given bin
            MAP1 = MAP2 = np.zeros((self.nth, self.nph))

            Rh = self.Re + self.interp_alt((t1 + t2)/2)
            k = self.Reff / Rh
            theta_max = np.arccos(np.sqrt(1 - k**2))

            # Generate equal area bins in the Local frame
            thetaL = np.arccos(np.linspace(np.cos(0), np.cos(theta_max), self.nth +
    1))
            phiL = np.radians(np.linspace(0, 360., self.nph + 1))
            lval = np.cos(thetaL) - np.sqrt(k**2 - (np.sin(thetaL)**2))
            if np.isnan(lval).any():
```

```
145                 lval[np.isnan(lval)] = np.cos(thetaL)[np.isnan(lval)]
146
147             # Define corresponding bins in the MAP frame
148             thetaM = np.arccos((1 - lval*np.cos(thetaL)) / k)
149             phiM = np.pi - phiL
150             phiM[phiM < 0] = phiM[phiM < 0] + 2*np.pi
151
152             # Get bin centres for the MAP frame
153             thetaMC = (thetaM[:-1]+thetaM[1:]) / 2
154             phiMC = (phiL[:-1]+phiL[1:]) / 2
155             delphi = phiMC[1] - phiMC[0]
156
157             # Compute MAP photons for lower time bound of the bin
158             for l in self.tn:
159                 for m in self.pn:
160                     # Transform MAP frame coord to get corresponding lat/lon
161                     MAPlat, MAPlon = f.rotate_frame(self.att['Lon'][i],
162                                                     np.pi/2 - self.att['Lat'][i], 0,
163                                                     thetaMC[l], phiMC[m], inverse=
    False)
164                     MAPlat = np.pi/2 - MAPlat
165
166                     # compute the integral in Eqn. 16 in Mate et. al. 2019
167                     omega = (f.omega(thetaM[l+1], k) - f.omega(thetaM[l], k)) *
    delphi
168
169                     # Compute the albedo C factor
170                     MAP1[l, m] = f.albedo_fac(MAPlat, MAPlon, thetaMC[l],
171                                               self.mpole_lat, self.mpole_lon, self.
    phi_sol) * omega
172
173             # Normalise the MAP array to get actual photon rate in each bin (in ph/s)
174             MAP1 = self.MAP_Norm / (1 - np.cos(theta_max)) * MAP1
175
176             # Compute MAP photons for upper time bound of the bin
177             for l in self.tn:
178                 for m in self.pn:
179                     # Transform MAP frame coord to get corresponding lat/lon
180                     MAPlat, MAPlon = f.rotate_frame(self.att['Lon'][i + 1],
181                                                     np.pi/2 - self.att['Lat'][i + 1],
     0,
182                                                     thetaMC[l], phiMC[m], inverse=
    False)
183                     MAPlat = np.pi/2 - MAPlat
184
185                     # Compute the integral in Eqn. 16 in Mate et. al. 2019
186                     omega = (f.omega(thetaM[l+1], k) - f.omega(thetaM[l], k)) *
    delphi
187
188                     # Compute the albedo C factor
189                     MAP2[l, m] = f.albedo_fac(MAPlat, MAPlon, thetaMC[l],
190                                               self.mpole_lat, self.mpole_lon, self.
    phi_sol) * omega
191
192             # Normalise the MAP array to get actual photon rate in each bin (in ph/s)
193             MAP2 = self.MAP_Norm / (1 - np.cos(theta_max)) * MAP2
194
195             # Find the MAP ratio (Eqn. 18 Mate. et. al 2019) at the time bin bounds
196             ratio1 = MAP1/MAP1.max()
```

```
            ratio2 = MAP2/MAP2.max()
            slope_ratio = (ratio2-ratio1) / orb_delta
            const_ratio = ratio1

            # Prepare interpolation params to interpolate MAP array for the times in
            # between the bounds of the given time bin
            slope_MAP = (MAP2-MAP1) / orb_delta
            const_MAP = MAP1

            # Calculate Number of Photons expected within the given time bin
            delT = t1 + orb_delta - tend
            tt = tend - t1
            t = []
            while tt < delT:
                MAP_t = slope_MAP*tt + const_MAP
                rate_alb = MAP_t.max()
                tt += expovariate(rate_alb)
                t.append(tt)

            t = np.array(t) + tend
            Nphot = t.size

            # Interpolate coordinates for all times
            Etheta_t = self.interp_theta(t)
            Ephi_t = self.interp_phi(t) % (2*np.pi)  # Take modulo to correct for
    interpolation above 2*pi
            Roll = self.interp_roll(t) % (2*np.pi)  # Take modulo to correct for
    interpolation above 2*pi
            Ealt_t = self.interp_alt(t)

            # Select Photons
            Nrem = self.data_max - p

            # Check remaining number of photons in order to not repeat them
            if Nphot > Nrem:
                idx = self.source_idx[p:]
                self.source_idx = np.random.permutation(self.data_max)
                idx = np.append(idx, self.source_idx[:Nphot - Nrem])
                p = Nphot - Nrem
                cycle += 1
            else:
                idx = self.source_idx[p:p+Nphot]
                p += Nphot

            # Extract photons from the source file
            src = self.source[idx]

            # Get incoming photon direction
            theta = src['theta']
            phi = src['phi']

            # Get incoming photon source sphere coordinates
            x = src['x'] - self.x0
            y = src['y'] - self.y0
            z = src['z'] - self.z0

            # Find whether photon hits the Earth or not
            Rh = self.Re + Ealt_t
            k = self.Reff / Rh
```

```
254             theta_max = np.arccos(np.sqrt(1 - k**2))
255 
256             del k  # memory management
257 
258             # Find dirn of the Earth in ECL frame
259             xe = Rh*np.sin(Etheta_t)*np.cos(Ephi_t)
260             ye = Rh*np.sin(Etheta_t)*np.sin(Ephi_t)
261             ze = Rh*np.cos(Etheta_t)
262 
263             # Unit vector in direction of the photon
264             ux = -np.sin(theta)*np.cos(phi)
265             uy = -np.sin(theta)*np.sin(phi)
266             uz = -np.cos(theta)
267 
268             # Solve Eqn. 11 in Mate et. al. 2019
269             delx = x - xe
270             dely = y - ye
271             delz = z - ze
272 
273             bprime = ux*delx + uy*dely + uz*delz
274             c = delx**2 + dely**2 + delz**2 - self.Reff**2
275             delta = bprime**2 - c
276 
277             # Memory management
278             del x, y, z, xe, ye, ze, ux, uy, uz, delx, dely, delz, c
279 
280             # Create a mask for earth crossing
281             mask = np.zeros(Nphot)
282 
283             # Fill earth mask with photons which cross earth
284             deltamask = delta > 0.
285 
286             lambda1 = -bprime[deltamask] + np.sqrt(delta[deltamask])
287             lambda2 = -bprime[deltamask] - np.sqrt(delta[deltamask])
288 
289             # Find photons which cross the Earth (in order to select them)
290             earthcrossing = (lambda1 < 0.) | (lambda2 < 0.)
291 
292             mask[np.where(deltamask)[0][earthcrossing]] = 1
293             mask = mask.astype(bool)
294 
295             # Select photons coming from Earth and reject other
296             idx = idx[mask]
297             t = t[mask]
298             Etheta_t = Etheta_t[mask]
299             Ephi_t = Ephi_t[mask]
300             Ealt_t = Ealt_t[mask]
301             Roll = Roll[mask]
302 
303             theta = theta[mask]
304             phi = phi[mask]
305 
306             # Transform Geant4 coord to Local to find the corresponding MAP
    coordinate
307             rotate_frame = np.vectorize(f.rotate_frame)
308             phot_thetaL, phot_phiL = rotate_frame(Ephi_t, Etheta_t, Roll,
309                                                   theta, phi, inverse=True)
310             phot_phiM = np.pi - phot_phiL
311 
```

```
            phot_phiM[phot_phiM < 0] = phot_phiM[phot_phiM < 0] + 2*np.pi
            theta_idx = np.digitize(phot_thetaL, thetaL) - 1
            theta_idx[theta_idx == self.nth] = self.nth - 1
            phi_idx = np.digitize(phot_phiM, phiL) - 1

            # Interpolate the map to get ratio
            phot_ratio = slope_ratio[theta_idx, phi_idx] * (t-tend) + const_ratio[
    theta_idx, phi_idx]

            # Throw uniform random numbers and select photons
            eta = np.random.uniform(0, 1, mask.sum())
            ratio_sel = phot_ratio > eta

            # Select indexes
            idx_global = np.append(idx_global, idx[ratio_sel])
            t_global = np.append(t_global, t[ratio_sel])
            tend = t_global[-1]

        print(cycle)

        # Select DPIX indices corresponding to the selected photons
        dpix_idx = self.idxmap[idx_global.astype(int), 0]
        mult = self.idxmap[idx_global.astype(int), 1]

        # Memory management
        del mask, idx, deltamask, lambda1, lambda2, bprime, earthcrossing
        del self.source, self.source_idx, self.idxmap

        dpix_idx = np.hstack([list(range(x, y)) for x, y in zip(dpix_idx, dpix_idx +
    mult)])

        # Store output data in 2-D array to return
        events = np.zeros((dpix_idx.size, 6))
        events[:, 0] = np.repeat(t_global, mult)
        events[:, 1] = self.dpixel[dpix_idx]['energy'].data
        events[:, 2] = self.dpixel[dpix_idx]['X'].data
        events[:, 3] = self.dpixel[dpix_idx]['Y'].data
        events[:, 4] = np.repeat(mult, mult)
        events[:, 5] = 2

        # Store ID for selected photon
        photon = np.insert(idx_global.reshape(idx_global.size, 1), 0, 2, axis=1)

        print("Albedo eventilist returned")
        return events, photon
```

Listing B.7 – electrons.py

```
"""
    Module to copmute SAA electron events using PIRA
"""

from configparser import ConfigParser, BasicInterpolation

import numpy as np
from astropy.io import fits
from scipy.interpolate import interp1d

from common import functions as f
```

```
12
13
14  class ELECTRONS(object):
15      '''
16          Class to handle SAA electron computations.
17      '''
18
19      def __init__(self, time, fluxint, E_bins, configfile):
20          '''
21              Init call to the ELECTRONS class.
22          '''
23
24          # Read config file
25          config = ConfigParser(interpolation=BasicInterpolation())
26          config.read(configfile)
27
28          # Algorithm variables
29          delt = config.getfloat('VARIABLES', 'DELTAT_PARTICLE')
30          self.maxrate = config.getfloat('VARIABLES', 'ELECMAXRATE')
31
32          # Read G4 Files, store database and initialise date
33          elecsrc = config['PATHS']['ELECSRC']
34          elecdpix = config['PATHS']['ELECDPIX']
35          elecflux = config['PATHS']['ELECFLUX']
36
37          # Get input spec info
38          input_spec_info = fits.open(elecflux)
39
40          # Read params needed for flux normalisation
41          total_flux = input_spec_info[1].header['FLUX']
42          Rp_ELE = input_spec_info[1].header['RADIUS']
43          Theta_max = input_spec_info[1].header['THETAMAX']
44
45          # Read input spectrum data
46          sim_spec_int = input_spec_info[1].data['Integrated_flux(cm-2.s-1)']
47          sim_energy = input_spec_info[1].data['E(MeV)']*1000
48
49          # Read G4 Files
50          sourceHDU = fits.open(elecsrc, mmap=True)
51          dpixelHDU = fits.open(elecdpix, mmap=True)
52
53          # Get avg. detection efficiency
54          N_SIMU = sourceHDU[1].header['NPH_SIMU']
55          N_INTE = sourceHDU[1].header['NPH_INTE']
56          self.Eff_ELE = np.float32(N_INTE / N_SIMU)
57
58          # Read Geant4 simulation data
59          self.source = sourceHDU[1].data
60          self.dpixel = dpixelHDU[1].data
61          self.data_max = self.source['eventID'].size
62
63          # Map src to dpix index
64          self.idxmap = np.zeros((self.source['eventID'].size, 2))
65          evt, self.idxmap[:, 0], self.idxmap[:, 1] = np.unique(self.dpixel['eventID'],
66                                                                return_index=True,
67                                                                return_counts=True)
68          self.idxmap = self.idxmap.astype(int)
69          del evt
70
```

```
71          # Compute weights and initialise global variables
72          # Define timebin intervals
73          self.time = np.arange(time[0], time[-1] + delt, delt)
74
75          # Correct for simulated and input flux information energy bins
76          sim_Emax = sim_energy.max()
77          if (E_bins > sim_Emax).any():
78              emaxID = np.where(E_bins > sim_Emax)[0][1] - 1
79              E_bins = E_bins[:emaxID]
80              fluxint = fluxint[:, :emaxID]
81
82          # Define finer energy bins for particle selection
83          self.ebins = 10**np.linspace(np.log10(E_bins[0]), np.log10(E_bins[-1]),
    10000)
84
85          # Find expected differential spectrum from the integrated spectrum
86          self.weight_exp = np.zeros((time.size, E_bins.size))
87          self.weight_exp[:, 1:] = (fluxint[:, :-1] - fluxint[:, 1:])
88
89          # Interpolate the expected spectrum with finer energy bins
90          self.prob_exp = f.interpolate_prob(self.weight_exp.cumsum(1), self.ebins,
    E_bins)
91
92          # Find the simulated differential spectrum from the integrated spectrum
93          weight_sim_tmp = (sim_spec_int[:-1] - sim_spec_int[1:])
94          weight_sim_tmp = np.insert(weight_sim_tmp, 0, 0)
95          prob_sim = weight_sim_tmp.cumsum()/total_flux
96          prob_sim = f.interpolate_prob(prob_sim, self.ebins, sim_energy)
97          self.w_sim = np.insert(np.diff(prob_sim), 0, 0)
98
99          # Normalise the spectrum to estimate expected particle rate per energy bin
100         self.prob_exp = self.prob_exp*Rp_ELE**2*np.pi*np.sin(Theta_max)**2
101
102         # Interpolate the particle rate per energy bin to a smaller time interval
103         interp_prob = interp1d(time, self.prob_exp, axis=0)
104         self.prob_exp = interp_prob(self.time)
105
106     def elec_photlist(self):
107         '''
108             Function to return the Electrons event list.
109         '''
110
111         # Initialise global arrays
112         idx_global = []
113         t_global = []
114         orb_delta = self.time[1] - self.time[0]
115
116         # Compute photons for delta T
117         tend = self.time[0]
118
119         # Loop over consecutive time bins in the attitude file
120         for i in range(self.time[:-1].size):
121             # Assign time and prob bounds
122             t1 = self.time[i]
123             t2 = self.time[i + 1]
124
125             prob1 = self.prob_exp[i]
126             prob2 = self.prob_exp[i + 1]
127
```

```
128             if prob1[-1] == 0 and prob2[-1] == 0:
129                 tend = t2
130                 continue
131 
132             # Calculate Number of Photons
133             delT = t1 + orb_delta - tend
134 
135             # Compute expected differential spectrum
136             w_exp1 = np.insert(np.diff(prob1), 0, 0)
137             w_exp1[w_exp1 < 0] = 0
138             w_exp2 = np.insert(np.diff(prob2), 0, 0)
139             w_exp2[w_exp2 < 0] = 0
140 
141             # Scale the simulated spectrum
142             if w_exp1.cumsum()[-1] > w_exp1.cumsum()[-1]:
143                 ratio_tmp = w_exp1/self.w_sim
144             else:
145                 ratio_tmp = w_exp2/self.w_sim
146             ratio_tmp[np.isnan(ratio_tmp)] = 0
147             wsim_new = self.w_sim*ratio_tmp.max()
148 
149             # Compute rate and Nphot
150             rate = wsim_new.cumsum()[-1]*self.Eff_ELE
151             if (self.maxrate > 0 and rate > self.maxrate):
152                 rate = self.maxrate  # Set max rate
153             Nphot = int(rate * delT)
154 
155             # Correct for fractional rate so that there is at least one particle
    selected
156             mu = np.random.uniform(0, 1, 1)
157             if mu <= ((delT * rate) - Nphot):
158                 Nphot += 1
159 
160             # Get Poisson time
161             dt = np.random.exponential(scale=1/rate, size=Nphot)
162             t = dt.cumsum() + tend
163             del dt
164 
165             # Select particles
166             idx = np.random.randint(0, self.data_max - 1, Nphot)
167             src = self.source[idx]
168 
169             # Bin energy and find ratio for each particle
170             ene_idx = np.digitize(src['energy'], self.ebins)
171             ene_idx[ene_idx == 10000] = 9999
172 
173             w_phot = (w_exp2[ene_idx]-w_exp1[ene_idx]) / orb_delta * (t - t1) +
    w_exp1[ene_idx]
174             ratio_phot = w_phot/wsim_new[ene_idx]
175             ratio_phot[np.isnan(ratio_phot)] = 0
176 
177             # Select/ discard photon based on ratio
178             eta = np.random.uniform(0, 1, Nphot)
179             sel = ratio_phot > eta
180 
181             # Select indexes
182             idx_global = np.append(idx_global, idx[sel])
183             t_global = np.append(t_global, t[sel])
184             tend = t_global[-1]
```

```
185
186         # Select DPIX indices corresponding to the selected particles
187         dpix_idx = self.idxmap[idx_global.astype(int), 0]
188         mult = self.idxmap[idx_global.astype(int), 1]
189         dpix_idx = np.hstack([list(range(x, y)) for x, y in zip(dpix_idx, dpix_idx +
    mult)])
190
191         # Store output data in 2-D array to return
192         events = np.zeros((dpix_idx.size, 6))
193         events[:, 0] = np.repeat(t_global, mult)
194         events[:, 1] = self.dpixel[dpix_idx]['energy'].data
195         events[:, 2] = self.dpixel[dpix_idx]['X'].data
196         events[:, 3] = self.dpixel[dpix_idx]['Y'].data
197         events[:, 4] = np.repeat(mult, mult)
198         events[:, 5] = 3
199
200         # Store ID for selected particles
201         photon = np.insert(idx_global.reshape(idx_global.size, 1), 0, 3, axis=1)
202
203         print("Electron eventlist returned")
204         return events, photon
```

Listing B.8 – protons.py

```
1  """
2      Module to copmute SAA proton events using PIRA
3  """
4
5  from configparser import ConfigParser, BasicInterpolation
6
7  import numpy as np
8  from astropy.io import fits
9  from scipy.interpolate import interp1d
10
11 from common import functions as f
12
13
14 class PROTONS(object):
15     '''
16         Class to handle SAA protons computations.
17     '''
18
19     def __init__(self, time, fluxint, E_bins, configfile):
20         '''
21             Init call to the PROTONS class.
22         '''
23
24         # Read config file
25         config = ConfigParser(interpolation=BasicInterpolation())
26         config.read(configfile)
27
28         # Algorithm variables
29         delt = config.getfloat('VARIABLES', 'DELTAT_PARTICLE')
30         self.maxrate = config.getfloat('VARIABLES', 'PROTMAXRATE')
31
32         # Read G4 Files, store database and initialise date
33         protsrc = config['PATHS']['PROTSRC']
34         protdpix = config['PATHS']['PROTDPIX']
35         protflux = config['PATHS']['PROTFLUX']
```

```
36
37          # Get input spec info
38          input_spec_info = fits.open(protflux)
39
40          # Read params needed for flux normalisation
41          total_flux = input_spec_info[1].header['FLUX']
42          Rp_PRO = input_spec_info[1].header['RADIUS']
43          Theta_max = input_spec_info[1].header['THETAMAX']
44
45          # Read input spectrum data
46          sim_spec_int = input_spec_info[1].data['Integrated_flux(cm-2.s-1)']
47          sim_energy = input_spec_info[1].data['E(MeV)']*1000
48
49          # Read G4 Files
50          sourceHDU = fits.open(protsrc, mmap=True)
51          dpixelHDU = fits.open(protdpix, mmap=True)
52
53          # Get avg. detection efficiency
54          N_SIMU = sourceHDU[1].header['NPH_SIMU']
55          N_INTE = sourceHDU[1].header['NPH_INTE']
56          self.Eff_PRO = np.float32(N_INTE / N_SIMU)
57
58          # Read Geant4 simulation data
59          self.source = sourceHDU[1].data
60          self.dpixel = dpixelHDU[1].data
61          self.data_max = self.source['eventID'].size
62
63          # Map src to dpix index
64          self.idxmap = np.zeros((self.source['eventID'].size, 2))
65          evt, self.idxmap[:, 0], self.idxmap[:, 1] = np.unique(self.dpixel['eventID'],
66                                                                return_index=True,
67                                                                return_counts=True)
68          self.idxmap = self.idxmap.astype(int)
69          del evt
70
71          # Compute weights and initialise global variables
72          # Define timebin intervals
73          self.time = np.arange(time[0], time[-1] + delt, delt)
74
75          # Define finer energy bins for particle selection
76          self.ebins = 10**np.linspace(np.log10(E_bins[0]), np.log10(E_bins[-1]),
    10000)
77
78          # Find expected differential spectrum from the integrated spectrum
79          self.weight_exp = np.zeros((time.size, E_bins.size))
80          self.weight_exp[:, 1:] = (fluxint[:, :-1] - fluxint[:, 1:])
81
82          # Interpolate the expected spectrum with finer energy bins
83          self.prob_exp = f.interpolate_prob(self.weight_exp.cumsum(1), self.ebins,
    E_bins)
84
85          # Find the simulated differential spectrum from the integrated spectrum
86          weight_sim_tmp = (sim_spec_int[:-1] - sim_spec_int[1:])
87          weight_sim_tmp = np.insert(weight_sim_tmp, 0, 0)
88          prob_sim = weight_sim_tmp.cumsum()/total_flux
89          prob_sim = f.interpolate_prob(prob_sim, self.ebins, sim_energy)
90          self.w_sim = np.insert(np.diff(prob_sim), 0, 0)
91
92          # Normalise the spectrum to estimate expected particle rate per energy bin
```

```
93          self.prob_exp = self.prob_exp*Rp_PRO**2*np.pi*np.sin(Theta_max)**2
94
95          # Interpolate the particle rate per energy bin to a smaller time interval
96          interp_prob = interp1d(time, self.prob_exp, axis=0)
97          self.prob_exp = interp_prob(self.time)
98
99      def prot_photlist(self):
100         '''
101             Function to return the Protons event list.
102         '''
103
104         # Initialise global arrays
105         idx_global = []
106         t_global = []
107         orb_delta = self.time[1] - self.time[0]
108
109         # Compute photons for delta T
110         tend = self.time[0]
111
112         # Loop over consecutive time bins in the attitude file
113         for i in range(self.time[:-1].size):
114             # Assign time and prob bounds
115             t1 = self.time[i]
116             t2 = self.time[i + 1]
117
118             prob1 = self.prob_exp[i]
119             prob2 = self.prob_exp[i + 1]
120
121             if prob1[-1] == 0 and prob2[-1] == 0:
122                 tend = t2
123                 continue
124
125             # Calculate Number of Photons
126             delT = t1 + orb_delta - tend
127
128             # Compute expected weights
129             w_exp1 = np.insert(np.diff(prob1), 0, 0)
130             w_exp1[w_exp1 < 0] = 0
131             w_exp2 = np.insert(np.diff(prob2), 0, 0)
132             w_exp2[w_exp2 < 0] = 0
133
134             # Scale the simulated spectrum
135             if w_exp1.cumsum()[-1] > w_exp1.cumsum()[-1]:
136                 ratio_tmp = w_exp1/self.w_sim
137             else:
138                 ratio_tmp = w_exp2/self.w_sim
139             ratio_tmp[np.isnan(ratio_tmp)] = 0
140             wsim_new = self.w_sim*ratio_tmp.max()
141
142             # Compute rate and Nphot
143             rate = wsim_new.cumsum()[-1]*self.Eff_PRO
144             if (self.maxrate > 0 and rate > self.maxrate):
145                 rate = self.maxrate  # Set max rate
146             Nphot = int(rate * delT)
147
148             # Correct for fractional rate so that there is at least one particle
    selected
149             mu = np.random.uniform(0, 1, 1)
150             if mu <= ((delT * rate) - Nphot):
```

```
                Nphot += 1

            # Get Poisson time
            dt = np.random.exponential(scale=1/rate, size=Nphot)
            t = dt.cumsum() + tend
            del dt

            # Select particles
            idx = np.random.randint(0, self.data_max - 1, Nphot)
            src = self.source[idx]

            # Bin energy and find ratio for each particle
            ene_idx = np.digitize(src['energy'], self.ebins)
            ene_idx[ene_idx == 10000] = 9999

            w_phot = (w_exp2[ene_idx]-w_exp1[ene_idx]) / orb_delta * (t - t1) + w_exp1[ene_idx]
            ratio_phot = w_phot/wsim_new[ene_idx]
            ratio_phot[np.isnan(ratio_phot)] = 0

            # Select/ discard photon based on ratio
            eta = np.random.uniform(0, 1, Nphot)
            sel = ratio_phot > eta

            # Select indexes
            idx_global = np.append(idx_global, idx[sel])
            t_global = np.append(t_global, t[sel])
            tend = t_global[-1]

        # Select DPIX indices corresponding to the selected particles
        dpix_idx = self.idxmap[idx_global.astype(int), 0]
        mult = self.idxmap[idx_global.astype(int), 1]
        dpix_idx = np.hstack([list(range(x, y)) for x, y in zip(dpix_idx, dpix_idx + mult)])

        # Store output data in 2-D array to return
        events = np.zeros((dpix_idx.size, 6))
        events[:, 0] = np.repeat(t_global, mult)
        events[:, 1] = self.dpixel[dpix_idx]['energy'].data
        events[:, 2] = self.dpixel[dpix_idx]['X'].data
        events[:, 3] = self.dpixel[dpix_idx]['Y'].data
        events[:, 4] = np.repeat(mult, mult)
        events[:, 5] = 4

        # Store ID for selected particles
        photon = np.insert(idx_global.reshape(idx_global.size, 1), 0, 4, axis=1)

        print("Protons eventlist returned")
        return events, photon
```

Listing B.9 – functions.py

```
"""

Common functions used by the PIRA background simulation codes

"""


```

```
8  from math import sin, cos, acos, atan, atan2, pi, sqrt
9  import numpy as np
10
11
12 def geo2mag(glat, glon, mpole_lat=80.3, mpole_lon=287.59):
13     '''
14         Function to convert geographic coordinates to geomagnetic coordinates
15         The function is copy of IDL routine GEO2MAG (see the link below for details)
16
17         https://idlastro.gsfc.nasa.gov/ftp/pro/astro/geo2mag.pro
18     '''
19
20     # Define current geomagnetic North pole lat and lon in geographic coordinates
21     MPlon = np.radians(mpole_lon)
22     MPlat = np.radians(mpole_lat)
23
24     # Coordinate to be transformed represented in Cartesian system (in form of a 3
       vector)
25     xyz = np.array([[cos(glat)*cos(glon)], [cos(glat)*sin(glon)], [sin(glat)]])
26
27     # First rotation
28     geolong2maglong = np.zeros((3, 3))
29     geolong2maglong[0, 0] = cos(MPlon)
30     geolong2maglong[0, 1] = sin(MPlon)
31     geolong2maglong[1, 0] = -sin(MPlon)
32     geolong2maglong[1, 1] = cos(MPlon)
33     geolong2maglong[2, 2] = 1.
34
35     # Second roatation
36     tomaglat = np.zeros((3, 3))
37     tomaglat[0, 0] = cos(pi/2 - MPlat)
38     tomaglat[0, 2] = -sin(pi/2 - MPlat)
39     tomaglat[2, 0] = sin(pi/2 - MPlat)
40     tomaglat[2, 2] = cos(pi/2 - MPlat)
41     tomaglat[1, 1] = 1.
42
43     # Rotate the vector to get its coordinates in new the geomagnetic frame
44     matrix = np.matmul(tomaglat, geolong2maglong)
45     out = np.matmul(matrix, xyz)
46
47     # Convert from Cartesian to polar to get the geomagnetic latitude
48     mlat = atan(out[2]/sqrt(out[1]**2 + out[0]**2))
49
50     return mlat
51
52
53 def rot_mat(alpha, beta, gamma):
54     '''
55         General roatation matrix used to transform coordinates between different
56         frames i.e. between MAP and Geographic and between G4 and Local.
57
58         The matrix has form corresponding to a ZYZ elemental rotation (see below for
       details)
59         https://en.wikipedia.org/wiki/Euler_angles#Rotation_matrix
60     '''
61
62     # Initiate empty matrix
63     matrix = np.zeros((3, 3))
64
```

```
65      # Compute sin and cos of all euler angles
66      c1 = cos(alpha)
67      s1 = sin(alpha)
68
69      c2 = cos(beta)
70      s2 = sin(beta)
71
72      c3 = cos(gamma)
73      s3 = sin(gamma)
74
75      # Populate the matrix elements based on the ZYZ rule
76      matrix[0, 0] = c1*c2*c3 - s1*s3
77      matrix[0, 1] = -c3*s1 - c1*c2*s3
78      matrix[0, 2] = c1*s2
79
80      matrix[1, 0] = c1*s3 + c2*c3*s1
81      matrix[1, 1] = c1*c3 - c2*s1*s3
82      matrix[1, 2] = s1*s2
83
84      matrix[2, 0] = -c3*s2
85      matrix[2, 1] = s2*s3
86      matrix[2, 2] = c2
87
88      return matrix
89
90
91  def rotate_frame(alpha, beta, gamma, theta, phi, inverse=False):
92      '''
93          Function to apply frame rotation for a given coordinate vector
94      '''
95
96      # Convert the polar coordinates theta, phi to the Cartesian vector
97      vec1 = np.array([[sin(theta)*cos(phi)], [sin(theta)*sin(phi)], [cos(theta)]])
98
99      # Get rotation matrix based on the rotation angles
100     matrix = rot_mat(alpha, beta, gamma)
101     if inverse:
102         matrix = matrix.T
103
104     # Apply the matrix to the vector to transform it to the new coordinate system
105     vec2 = np.matmul(matrix, vec1)
106
107     # Get the polar coordinates in the new system
108     theta2 = acos(vec2[2])
109     phi2 = atan2(vec2[1], vec2[0])
110     if phi2 < 0:
111         phi2 += 2*pi
112
113     # The theta and phi are rounded off to 4 decimal places as the function is mostly
        used to transform bin
114     # boundaries from one frame to another and to bin photon directions it is
        importat to have finite decimal
115     # points in the the bin boundary
116     return round(theta2, 4), round(phi2, 4)
117
118
119 def omega(theta, k):
120     '''
121         Compute the integral factor in the Eqn. 16 of Mate et. al. 2019
```

```
122     '''
123     l2 = 1 + k**2 - 2*k*np.cos(theta)
124     omega = (k*np.cos(theta) - 1)/np.sqrt(l2)
125     return omega
126 
127 
128 def albedo_fac(lat, lon, theta, mpole_lat=80.3, mpole_lon=287.59, phi_sol=0.25):
129     '''
130         Compute the albedo 'C' factor in Sazonov et. al. 2007
131     '''
132     epsilon = 0
133     lambdam = geo2mag(lat, lon, mpole_lat, mpole_lon)  # Get the magnetic latitude
134 
135     mu = cos(theta)
136     mu_val = mu*(1 + mu)
137 
138     const = 3*1.47*0.0178/5/np.pi
139 
140     # compute cut-off rigidity
141     R = 59.4 * cos(lambdam)**4 / (1 + (1 + cos(lambdam)**3 * sin(theta) * sin(epsilon
    ))**.5)**2
142 
143     num = ((phi_sol/2.8)**0.4 + (phi_sol/2.8)**1.5)**(-1)
144     denom = sqrt(1 + (R / (1.3*phi_sol**0.25*(1 + 2.5 * phi_sol**0.4)))**2)
145 
146     C = const*mu_val*num/denom
147     return C
148 
149 
150 def Gruber(E):
151     '''
152         Gruber et. al. 1999 CXB spectral model
153     '''
154     if E < 60.:
155         gruber = 7.877*E**(-.29)*np.exp(-E/41.13)
156     else:
157         gruber = 0.0259*(E/60)**(-5.5) + 0.504*(E/60)**(-1.58) + 0.0288*(E/60)
    **(-1.05)
158     return gruber/E
159 
160 
161 def Moretti(E):
162     '''
163         Moretti et. al. 2009 CXB spectral model.
164     '''
165     norm = 0.109
166     pow1 = 1.4
167     pow2 = 2.88
168     Ebreak = 29.
169 
170     return norm/((E/Ebreak)**pow1 + (E/Ebreak)**pow2)
171 
172 
173 def MorettiGruber(E, Ecut=163.708):
174     '''
175         CXB spectral model used for ECLAIRs Geant4 simulation.
176         Combination of Moretti and Gruber models.
177     '''
178     if E < Ecut:
```

```
179         cxbmg = Moretti(E)
180     else:
181         cxbmg = Gruber(E)
182 
183     return cxbmg
184 
185 
186 def Ae(E):
187     '''
188         Reflection component in the Churazov et. al. 2008 CXB reflection spectral
    model.
189     '''
190     return (1.22/((E/28.5)**(-2.54) + (E/51.3)**1.57 - .37) *
191             (2.93 + (E/3.08)**4)/(1 + (E/3.08)**4) *
192             (0.123 + (E/91.83)**3.44)/(1 + (E/91.83)**3.44))
193 
194 
195 def reflection(E, Ecut=163.708):
196     '''
197         Total Reflection spectrum in case of the ECLAIRs Geant4 simulations
198     '''
199     if E < Ecut:
200         ref = Ae(E)*Moretti(E)
201     else:
202         ref = Ae(E)*Gruber(E)
203 
204     return ref
205 
206 
207 def albedo(E):
208     '''
209         Energy part of the albedo spectrum.
210     '''
211     return 1. / ((E/44)**(-5) + (E/44)**1.4)
212 
213 
214 def interpolate_prob(prob, ebins, E_edges):
215     '''
216         Function to interpolate input SAA particle spectrum between available bins
    and the new finer bins
217     '''
218     idx = np.digitize(ebins[1:-1], E_edges)
219     pr = (ebins[1:-1]-E_edges[idx-1])/(E_edges[idx] - E_edges[idx-1])
220     if prob.ndim == 2:
221         pint = (1.0-pr)*prob[:, idx-1] + pr*prob[:, idx]
222         pint = np.insert(pint, 0, 0, axis=1)
223         return np.append(pint, pint[:, -1].reshape(pint.shape[0], 1), axis=1)
224 
225     else:
226         pint = (1.0-pr)*prob[idx-1] + pr*prob[idx]
227         pint = np.insert(pint, 0, 0)
228         return np.append(pint, pint[-1])
```