Searching for Evaporating Primordial Black Holes using Fermi Gamma Ray Telescope Data

Imperial College London

Jagdev Bains

CID 00552650

Horns Group, DESY

A thesis submitted for the degree of

MSci

May 2011

Abstract

All confirmed photon event data ($\geq$ 100MeV) recorded by the Fermi Gamma Ray Telescope's (FGRT's) Large Area Telescope (LAT) was used to attempt to locate photons emitted by evaporating primordial black holes (PBHs) via Hawking radiation. An 'event' is a single photon recorded by the LAT. HEALPix was used to group photons spatially into pixels with unique identification numbers, and all events within these pixels were sorted in time. The time interval between subsequent photons in each pixel was calculated and a histogram made for each pixel. The expected time distribution of events was compared with the measured for each pixel to identify any candidates for further analysis; if the expected varied significantly with the measured at the smallest time intervals ($26\mu s - 10s$). Several pixels were found that met the criteria for further analysis, however they were all found to include known high energy photon sources. No pixels were found with extra photons at short time intervals that could not be attributed to gamma-ray bursts (GRBs) or known sources. A limit was calculated on the range at which the LAT could detect a PBH burst, between 16.69 and 289.16pc.

Alle photonischen ($\geq$ 100MeV) und vom Large Area Telescope (LAT) des Fermi Gamma Ray Telescope's (FGRT's) dokumentierten, bestaetigten Ereignisse wurden benutzt, um Photonen zu lokalisieren, die durch sich verfluechtigende primordiale schwarze Loecher (PBHs) mit Hilfe der Hawking Strahlung emittiert wurden. HEALPix wurde benutzt, um einzelne Photonen raeumlich in Pixel zu gruppieren und alle Ereignisse innerhalb dieser zeitlich zu ordnen. Dabei wurde die Zeitspanne zwischen aufeinander folgenden Photonen in den einzelnen Pixel berechnet und ein Histogramm fuer jedes dieser Pixel erstellt. Die erwartete Zeitverteilung der Ereignisse wurde mit den gemessenen Ereignissen fuer jedes Pixel verglichen. Wenn die erwartete Verteilung von den gemessenen Daten abwich, wurden diese Pixel detaillierter untersucht. Unterschiedliche Pixel, die die Kriterien fuer eine weitere Analyse erreichten, wurden gefunden, jedoch haben sie alle bereits bekannte hochenergetische Photonquellen beinhaltet. Es wurden keine Pixel mit zusaetzlichen Photonen auf einem kurzen Interval ($26\mu s - 10s$) entdeckt, die nicht Gammastrahlenausbruechen (GRBs) oder bekannten Quellen zugeordnet werden konnten. Eine Grenze wurde fuer den Bereich berechnet in dem das LAT einen PBH Ausbruch aufspueren kann, diese liegt zwischen 16.69 und 289.16pc.

Acknowledgements

I would like to thank everybody in my group for their neverending patience and insight, along with speaking English when i was unable to understand in German; in particular Martin Raue and Milton Virgilio whom i barraged with questions on a regular basis. I would like to say a special thank you to my fellow Erasmus student Helena da Costa Kaufmann for translating frequently on my behalf, and listening to me constantly complain throughout our duration in Hamburg. Last but not least i have to thank my supervisor Professor Dieter Horns who gave me continual advice and support throughout my work, and without whom this would have not been possible. Thank you.

Contents

List of Figures

1

Introduction

1.1 Preface

This study was carried out at the Deutsches Elektronen-Synchotron (DESY) facility in Hamburg between November 2010 and June 2011. The research which went into this report was carried out in the Horns research group; a part of the experimental and astro-particle department. Professor Dieter Horns supervised all aspects of the research and advised where necessary on departures from the original plan due to unforeseen circumstances. My role was to carry out the spatial and temporal analysis required for this research and in the process write the necessary programs. The research involved learning and using Fermi Science Tools, C language, Python language, Linux operating systems and new maths where necessary. Almost all of the analysis was computational in nature, and it is these programs that reflect the work done for this study. This report has been written using TeXworks, the layout is that of a pre-existing freeware design template.

1.2 Aims

The initial aim of the research is to search all data collected by the LAT for bursts of photons with similar features to those expected by a PBH in the last stages of its existence. The success of this part of the research is vital as without it no other analysis can take place. The analysis should highlight some bursts of photons in time which can be referenced with known sources and GRBs, using pre-existing papers and data. If a burst occurs that is not attributable to the aforementioned then further analysis of the significance of such an event can be calculated. If all bursts are found to be from known sources and/or GRBs then volume density limits can be calculated on the frequency of evaporating PBHs. By not observing any evaporating PBHs it would be possible to estimate a limit of the number of PBHs in a given volume over time.

1.3 Overview

1.3.1 Theory

It has been theorised that PBHs end with a sudden burst of photons within a short time interval (1) due to Hawking radiation. This theory is widely accepted and it is believed that the LAT aboard the FGRT is well suited to detecting such an event(2). The telescope has a very low dead time, high energy range and allsky observation capabilities, this makes it suitable to detect and record bursts of high energy photons within a small time interval from any point in space (3). The term allsky refers to the complete 3D view of the local galaxy, as seen from the satellite. Relatively small PBHs of an original mass of $\sim 5 \cdot 10^{14}$g are expected to be evaporating at our current time providing they haven't accreted matter during their existence (1). The final burst of $10^{34} erg$ should consist of photons above 100MeV, all of which should be recorded by the LAT (4).

1.3.2 Experimental Procedure

The main procedure in this research was the statistical analysis of photon events. The analysis utilised data collected by the FGRT's LAT since its launch in June 2008. The weekly data collected by the LAT was processed using two Fermi Science Tools before being read into a C language program. The program's main function was to utilise HEALPix to spatially bin the photon events into pixels and then output a file with the event times and pixel identification numbers. This list was then processed into a Python language program which grouped events by pixel number, calculated the time interval between subsequent photons and created a histogram of time differences for each pixel. A second Python language program was used to fit an expected function to each histogram and compare this with the measured values; a two parameter maximisation of the poisson likelihood was used to calculate the expected values. Any large differences between expected and measured values in the shortest time bin were observed and recorded as possible candidates for further investigation. Candidates for further investigation were cross-checked with known photon sources and GRBs, and pixels including such sources were excluded from further analysis. The final results indicated that no evidence of evaporating PBHs had been recorded by the LAT since its launch.

2

Theory

2.1 Temporal Photon Distribution

If one assumes that the background rate of photons with respect to time is equal to some fixed probability multiplied by the number of photons, then it is simple to show that the expected photon distribution in time is exponential.

If one assumes that background photon events are poisson distributed with some fixed rate λ, the distribution of time intervals between events is a negative exponential (5). Note this is only the case for k=0.

$$D = n_o e^{-\lambda t} \tag{2.1}$$

D = distribution function, n_o = constant, t = time, λ = constant probability

The distribution in equation (2.1) has been observed in numerous studies and is generally accepted as the time distribution of background photons. To look for transient events it is therefore possible to compare measured distributions with the expected distribution; deviations from the expected values should be noticable.

2.2 Black Holes

Black holes are regions from which even light cannot escape, at their centre are infinitely dense singularities in space with very large gravitational fields. The area surrounding black holes, from which nothing can escape its gravitational field, is known as the event horizon(6). Anything that passes within the event horizon is pulled into the black hole;

this accretion of matter is one of the ways in which the event horizons of black holes can grow in size. It is usually the size of the event horizon that is used to define the size of the black hole and this measure will be used in this report. The only other way for a black hole to increase in size is for it to merge with another black hole, in this way supermassive black holes can be formed with event horizons greater than the distance between the Earth and Sun(7). The no-hair theorem(8) predicts that stable black holes can only be characterised by their mass, charge and spin. These three properties are the only external observables; all other information is lost within the black hole formation hence after a black hole becomes stable, no information about its early formation can be observed.

2.2.1 Properties

2.2.1.1 Mass

The simplest black hole model is one that is spherically symmetric and has no spin or charge; these are known as Schwarzschild black holes. The mass of a black hole is its most defining and important feature due to its relation to event horizon size and Schwarzschild radius. The Schwarzschild radius defines the distance from the centre of an object such that if the mass of that object was compressed, the escape speed of a particle trying to leave the object's field of gravity at the surface, would be equal to that of light (9). Black holes have a Schwarzschild radius larger than the size of the singularity; this radius is the same as the event horizon for Schwarzschild black holes. The simplest relation of mass to event horizon size is given therefore by the Schwarzschild radius equation(10).

$$r = \frac{2 \cdot G \cdot m}{c^2} \tag{2.2}$$

r = Schwarzschild radius, G = Gravitational constant, m = mass of object, c = speed of light in vacuum

It is this radius that defines the classification of black holes; anything from micro to supermassive. As black holes pull in more matter their mass increases so, as seen in equation (2.2), their event horizon size increases proportionally. The mass of black holes is on the solar mass scale but can vary across several orders of magnitude. Whilst it is possible for a black hole to be charge neutral and spinless, they all must have a mass by definition.

2.2.1.2 Charge and Spin

A charged, spinless black hole is known as a Reissner-Nordstrom black hole, a charge neutral, spining black hole is known as a Kerr black hole, whilst a charged, spinning black hole is known as a Kerr-Newman black hole(11). The inclusion of spin and charge affects the size of the event horizon in each case and the Schwarzschild radius becomes a different quantity than the event horizon. The charge and spin of black holes is something studied in more detail in other papers, but for this report no further discussions on this topic are necessary.

2.2.2 Formation of Black Holes

Black holes have been forming continuously since the start of the universe, along with PBHs occuring within a very short time scale at the beginning of the universe. The types of black holes become indistuingishable once they become stable as outlined in the no-hair theorem(8), but their formations are very different.

2.2.2.1 Gravitational Collapse

A star remains stable whilst the gravitational pressure is in balance with the degeneracy pressure within the star. For varying reasons the degeneracy pressure within a star can begin to decline leading to that star collapsing under the force of its own gravity. This can result in a neutron star, white dwarf star or a black hole being formed; this depends on the initial star composition and mass. This process occurs as stars reach the end of their lives and have exhausted all stellar fuel sources(12). Stars above the Tolman-Oppenheimer-Volkoff limit may collapse into black holes; these are formed when the star collapses within its own Schwarzschild radius and therefore can only be compressed further into a singularity(11). The age of black holes formed in this way can vary widely as stars have been forming and collapsing since the start of the universe. Only stars above the Tolman-Oppenheimer-Volkoff limit may collapse into black holes, this imposes a lower mass limit on any black holes that form in this way.

2.2.2.2 Primordial Black Holes

PBHs are those which formed at the very start of the universe where fluctuations in the density profile of the initial matter could have formed dense enough regions to create black holes(13, 14). This idea was first proposed by Zeldovich and Novikov in 1969(15). The dense nature of the early universe still required sizable fluctuations to create these early black holes, but research since (notably by Hawking and Carr) has shown PBHs to be possible with smaller fluctuations than first thought(13).

It is very difficult to determine the number and size of black holes created in this way due to the uncertainty and lack of data from those early stages of the universe. Unlike black holes formed from gravitational collapse it is possible for very small PBHs to have formed(16). It is for this reason that this study searches for smaller PBHs and not ones created by gravitational collapse at a more recent time.

There have also been theories linking PBH creation and string theory; cosmic string loops allowing their formation. More in-depth analyses of the fluctuations required and other theories have been carried out frequently in different papers over the last few decades (see (13, 14, 16)).

2.2.3 Hawking Radiation

In 1974 Stephen Hawking wrote in an article to Nature his first predictions of Hawking radiation being emitted from black holes throughout their lifetime(1). Hawking radiation is the emission of thermal radiation due to quantum effects. It is theorised that black holes can create and emit particles, under which circumstances the black hole mass and temperature would decrease. For most black holes this effect is negligible due to their enormous masses, but for smaller PBHs with masses less than $\sim 10^{16}$g, this cumulative effect would be more important. It is predicted that PBHs with a mass of $\sim 5 \cdot 10^{14}$g which have not accreted matter since their creation would have lost enough mass through Hawking radiation to be at the final stages of evaporation. The initial equation theorised by Stephen Hawking is as follows(17).

$$t \approx \frac{\hbar \cdot c^4}{G^2 \cdot m^3} \approx 10^{64} \cdot \left(\frac{m}{m_\odot}\right)^3 years \tag{2.3}$$

t = Lifetime of PBH, m = mass of PBH, $m_\odot$ = solar mass unit

One can see PBHs with a mass of $5 \cdot 10^{14}$g would be evaporating in the current epoch of $\sim 10^{17}$s. The different stages of Hawking radiation are illustrated in 2.1.

Photons of energies higher than 100Mega-Electronvolts (MeV) are expected to be emitted in the photon burst occuring at the end of the PBH lifetime(17). This sudden burst of photons is expected to happen over a short time interval in the order of seconds; depending on the spin and charge of the black hole the photon flux distribution will differ for each individual PBH.

FIG. 1 The parameter α counting the degrees of freedom of the Hawking mass radiation as a function of the black-hole temperature and lifetime. Λ_{qh} is the quark–hadron deconfinement scale. The contribution from each particle species is indicated.

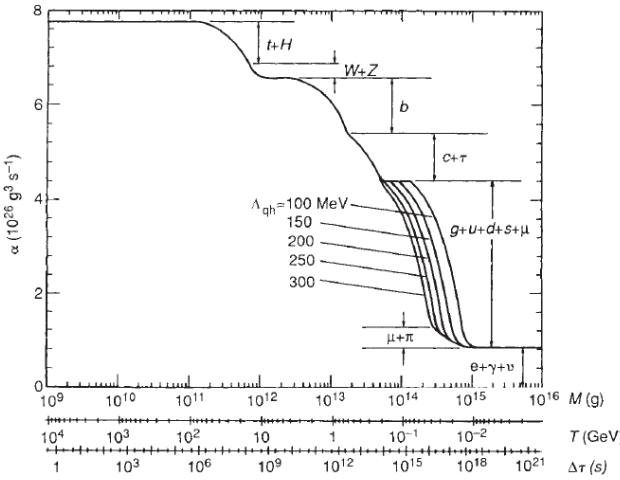


Figure 2.1: Profile of PBH evaporation - The graph shows the stages through which a PBH would evaporate. One can see that PBHs of mass $5 \cdot 10^{14}$g are expected to end in this epoch with a sudden photon burst(16)

2.3 Fermi Gamma Ray Telescope

The FGRT was originally launched in June 2008 and was named the Gamma-ray Large Area Space Telescope (GLAST) before its name was changed as tribute to Enrico Fermi. The telescope orbits earth at an altitude of 550km with a single orbit taking approximately 90minutes. The National Aeronautics and Space Administration (NASA) initially aimed for the FGRT to have a minimum lifetime of 5years possibly stretching to 10years: the collected information over the duration of the mission time is available for free on the FGRT website(3).

There are two instruments aboard the FGRT, the LAT and the GLAST Burst Monitor (GBM), both of which detect and analyse photon events. The LAT is the only relevant instrument for this study due to the energy range over which the GBM is aimed; with the burst of photons from Hawking radiation being primarily above 100MeV the GBM would be unable to detect them, hence the specific parameters of that instrument are not discussed here(17).

2.3.1 Large Area Telescope

The LAT is “an imaging, wide field of view, high energy gamma-ray telescope, covering the energy range from below 20MeV to more than 300Giga-Electronvolts (GeV)(18)”. The LAT is considered a natural successor to the Energetic Gamma Ray Experiment Telescope (EGRET), with improvements including higher energy ranges, allsky view capabilities and more accurate angular and energy resolutions(19). The LAT was built in modules and assembled by an international collaboration; contributions were made by high-energy physics institutes, universities across the world and NASA. The LAT is primarily in “scanning” mode which continually sweeps across the sky, but it can be turned towards specific events of interest if required.

The way in which the LAT detects photon events is through electron pair conversion. Through this conversion the direction of the original photon’s source and its energy can be reconstructed. The LAT is equipped with self-triggering tracker modules, calorimeter modules, an anticoincidence detector (ACD) and a data aquisition system (DAQ). Photons pass firstly into the tracker where they are converted into e^+e^- pairs and their motion tracked, followed by the calorimeter to measure the energy of the electron shower that follows(18).

2.3.1.1 Converter-Tracker

The tracker is also known as the converter-tracker due to the nature of its composition. The tracker-converter is comprised of interleaving layers of position-sensitive detectors and tungsten. The tungsten’s use is to increase the likelihood of photons being converted into e^+e^- pairs; these can be tracked by the single-sided silicon strip detectors (SSDs). Tungsten is chosen as the conversion material due to its high Z value which makes the process more probable. The SSDs are made of silicon which is important as it allows them to be self-triggering and not require any external triggers(18). The triggering mechanism acts to inform the tracker when to record information and from what region. Once the photons have been converted, the SSDs deliver the tracking information to the DAQ with a very high level of precision. The SSDs and tungsten foils are arranged in such a way as to avoid secondary scattering where possible. Secondary scattering would make it harder to reconstruct the original photon events due to the uncertainty involved in each collision. The tungsten covers only areas where detectors are present to avoid premature conversion, similarly the distance between the tungsten and SSDs is minmised to reduce the lever arm for further scattering. The material between SSDs can also cause premature conversion, but to minimise this problem this material is reduced in thickness(18). The configuration of the SSDs is shown in 2.2, with the photon events entering from the top.

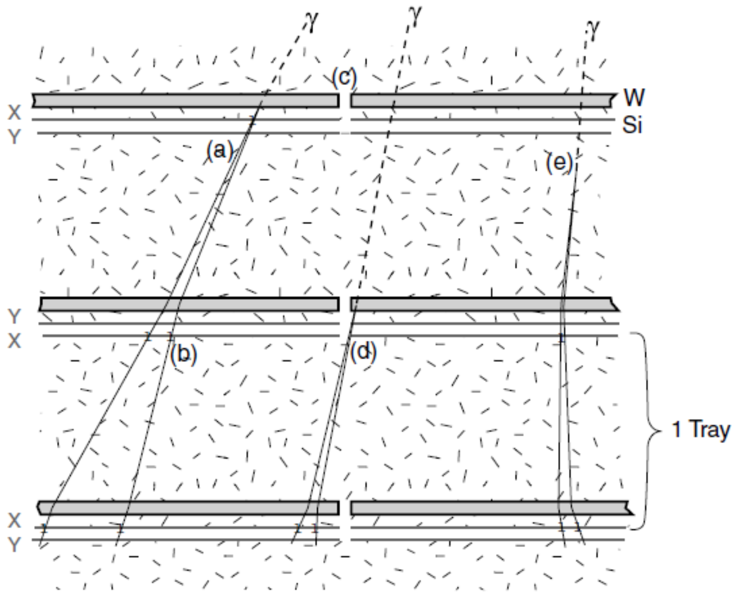


Figure 2.2: LAT Converter-Tracker Cross-section - Photons are more likely to convert into electron pairs at the tungsten layers and directly flow into the SSDs to be tracked(18)

2.3.1.2 Calorimeter

One calorimeter module is made up of 96 caesium iodide crystals split into 8 layers, with 12 crystals on each layer. The modules are arranged in alternating perpendicular layers forming a hodoscope(18). Each crystal is connected to pin diodes at each end which measure the difference in light levels. The measured differences provides information on the position of energy deposition which allows for more accurate event reconstruction. The calorimeter is designed not only to measure the energy deposited by electron showers but to reconstruct the image of the shower, this helps improve the background rejection rate. The high accuracy in the energy resolution of the calorimeter is vital to the main aims of the telescope, it is achieved by applying shower leakage corrections(18)

2.3.1.3 Anticoincidence Detector and Data Acquisition System

The ACD's main function is to help reject the background rate of charged-particles which could cause problems by giving false positives. The ACD is made of plastic scintillator tiles due to a number of variables including cost, weight, available technology and efficiency(18). Further details are given in (20).

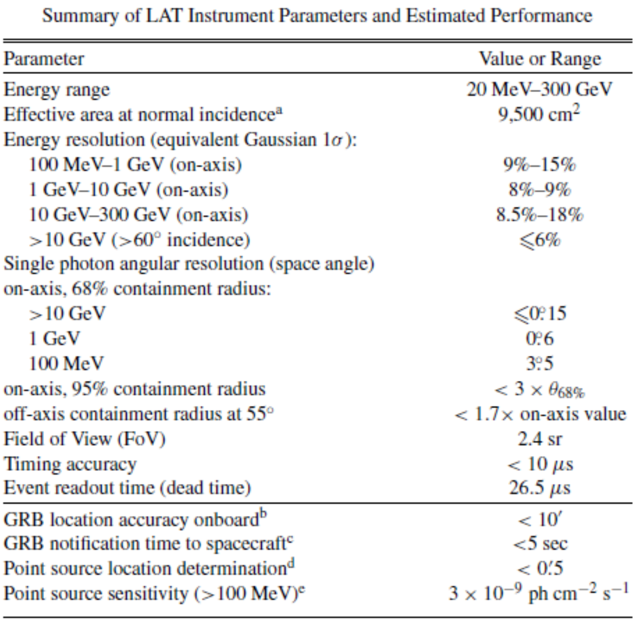
The DAQ simply collects the data from the other modules, runs necessary algorithms and provides an onboard platform to search for transient events. Specifics on the nature of the electronics within the DAQ are not necessary to discuss in this paper, but more details can be found in (18).

2.3.1.4 Specifications

All of the LAT specifications are discussed in far more detail in the Atwood et al 2009 paper(18); for the purposes of this report only a brief summary of the most pertinent specifications are outlined, see 2.3.

The excellent angular resolution at higher energies along with good temporal resolution allows for accurate photon source localisation, particularly relevant for this study. One of the most important parameters however is the very low dead time between events, the dead time is the minimum time required for the detector to process the last event and reset itself ready for the next. For the LAT the deadtime is 26μs which is important considering Stephen Hawking's first prediction of the length of the final photon burst at 0.1s(1). The field of view is also significant as it allows for a large portion of the sky to be viewed at any one time which would give a broader view on any bursts occuring.

The suitability of searching for final photon bursts by PBHs with the LAT has been discussed in more depth in a paper written by T. Ukwatta in 2010(2), the paper suggests that the parameters of the LAT are well within those necessary to detect a burst by a PBH.



Summary of LAT Instrument Parameters and Estimated Performance

Parameter	Value or Range
Energy range	20 MeV–300 GeV
Effective area at normal incidence[a]	9,500 cm^2
Energy resolution (equivalent Gaussian 1σ):	
100 MeV–1 GeV (on-axis)	9%–15%
1 GeV–10 GeV (on-axis)	8%–9%
10 GeV–300 GeV (on-axis)	8.5%–18%
>10 GeV (>60° incidence)	$\leqslant$6%
Single photon angular resolution (space angle)	
on-axis, 68% containment radius:	
>10 GeV	$\leqslant 0.^{\circ}15$
1 GeV	$0.^{\circ}6$
100 MeV	$3.^{\circ}5$
on-axis, 95% containment radius	$< 3 \times \theta_{68\%}$
off-axis containment radius at 55°	$< 1.7\times$ on-axis value
Field of View (FoV)	2.4 sr
Timing accuracy	$< 10\ \mu$s
Event readout time (dead time)	26.5 μs
GRB location accuracy onboard[b]	$< 10'$
GRB notification time to spacecraft[c]	<5 sec
Point source location determination[d]	$< 0.'5$
Point source sensitivity (>100 MeV)[e]	3×10^{-9} ph cm^{-2} s^{-1}

Figure 2.3: Official LAT Parameters - The salient paramaters that pertain to this study are all noted in the table, particularly the spatial and temporal resolution and dead time(18)

2.3.2 Fermi Science Tools

The Fermi Science Tools are a set of freely available programs and subroutines that allow the manipulation and refinement of the raw data collected by the LAT and GBM, provided by the FGRT website(21). The data from the LAT used in this research comes in weekly .fits files from the FGRT which can be joined together to make larger files representing all data collected since June 2008. Each weekly .fits file contains all of the collected information about all of the events that occured within that week; this includes but is not limited to angular direction, time of event and energy. For this study the tools were used to create a single clean set of photon event data from June 2008 until the present time. The two tools used were gtselect and gtmktime whose uses within this study are explained below in more detail.

2.3.2.1 gtselect

gtselect makes a number of user-defined cuts from the input data and outputs a .fits file with only events that fit within the given parameters. The available cuts include time intervals, locations in the sky and energies. There is an option for a “zenith angle cut” which is recommended by the FGRT website in almost every type of analysis. The zenith angle cut accounts for the presence of the Earth, the cut is necessary as it eliminates any kind of background radiation from Earth being counted as photon events. The generally accepted zenith angle cut is 105^o, with the Earth at 100^o (21). The output of gtselect is another .fits file with only the selected photon events included.

2.3.2.2 gtmktime

The second tool used in this study was gtmktime which is similar to gtselect in that it filters photon events using a number of optional filters. Any input file into gtmktime is cross-referenced with a “spacecraft” file obtained from the FGRT website. The spacecraft file contains information on “good time intervals”. Good time intervals are times when the LAT was fully operational and not being influenced by any problems and/or local events that could contaminate any data collected in that interval. The output of gtmktime is a .fits file, excluding events within some time intervals that were deemed unsuitable for data analysis by the filters.

There are several highly recommended filters available when using gtmktime, all of which were used in this study. One filter only allows photon events of the highest quality i.e. events with a high probability of being due to a photon, another accounts for the rocking motion of the FGRT as it views the allsky whilst a third only allows time intervals when the LAT was fully operational(21). These filters create the aforementioned good time intervals, and these intervals are listed in the spacecraft file. The use of gtmktime severely reduces the amount of available data but with the benefit of leaving only high quality data.

2.3.3 HEALPix

HEALPix is an acronym representing Hierarchical, Equal Area, Isolatitude Pixelisation. As a general definition, HEALPix breaks a curved surface into pixels of equal area which can be viewed as a simple 2D image. Spherical topology makes for very difficult analysis and observations due to the nature of the shape, HEALPix eliminates the complications involved with that topology and instead represents the area in the analysis as a flat projection(22). Within this study HEALPix was used to transform the view of the allsky as seen by the LAT into a flat image observable in two dimensions. A HEALPix photon counts map would show the number of photons detected within each pixel across a given length of time.

2.3.3.1 Pixelisation Scheme

The number of pixels a spherical shell is split into, is determined by a user-set variable called “Nside”. The pixels are distributed along evenly spaced equatorial lines of iso-latitude, where the number of these lines are given by (4·Nside - 1). Figure 2.4 shows four spheres split into differeing numbers of pixels.

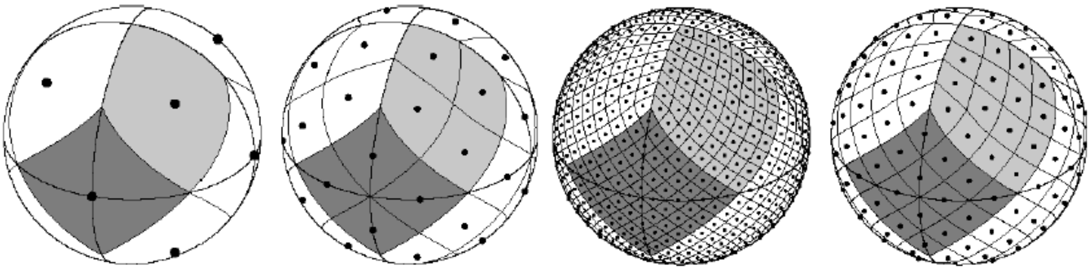


Figure 2.4: Pixelised Spheres - Pixelised spheres representing Nside = 1, 2, 4, 8.(23)

A direct consequence of the symmetry involved in the lines of isolatitude mean that only particular numbers of Nside are acceptable; Nside must be equal to $2^n \ \forall \ n \in Z$. The number of pixels as seen in 2.4 is given by $12 \cdot Nside^2$. Each pixel is given a particular identification number, which can be used to track that pixel's position when the flat representation is made. There are two numbering schemes possible, nested and ring, and their differences are demonstrated in 2.5. The ring numbering scheme is what has been used in this study as it's the most simple (22).

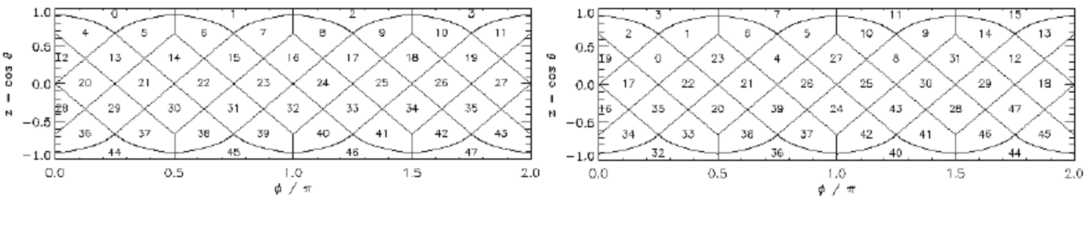


Figure 2.5: Cylindrical HEALPix Projection - Both images are projections of a cylinder but in two different numbering schemes; left - ring, right - nested.(23)

2.3.3.2 Software Package

The HEALPix software package consists of a number of subroutines that can perform various tasks within other programming languages e.g. C, Java. Any spherically distributed data can be processed using HEALPix to generate a more simplistic representation, this includes theoretical projections and actual data.

The software utilises FITS inputs/outputs, as outlined in (24), and some Fortran libraries(23). There are various tasks that the software package can execute, however only a few have been used in this study. Angular co-ordinates were converted into pixel identification numbers using the subroutine “ang2pix”, similarly routines such as “write healpix map” used appropriate input arrays to create HEALPix maps of the allsky. A few FITS routines were also used and a more detailed description can be found in (24). Healpy is a specific utilisation of HEALPix in Python programs which was previously unavailable, however this became possible during this study and was therefore used where appropriate, see (25). The healpy package requires matplotlib, numpy and pyfits libraries in order to operate fully.

2.3.3.3 Allsky Map

2.6 is a HEALPix projection of the allsky as seen by the LAT over the last 3years. The colour of each pixel is related to a number allocated to that pixel, for this map the number is that of the total photons recorded from that region. There are mostly regions of purely background photons which are blue in this map, along with more intense photon sources of varying colours. Within the map one can see the galactic plane, which is the large horizontal line across the centre. The map is centered on the galactic centre within the galactic plane. Other well known sources can be observed all over the map e.g. SN1054 (Crab Nebula) can be seen on the far right slightly underneath the galactic plane, the Geminga Pulsar is located almost directly above the Crab nebula. The Vela Pulsar is situated at the same latitude as the galactic plane but to the right of the central part, whilst the 3C454.3 Blazar is in the centre of the bottom left quadrant.

The galactic co-ordinate system is used to reference the locations of events and sources within the HEALPix maps. Two co-ordinates are given to each position, latitude and longitude. The longitude values range from 90^o at the top of the map to

-90^o at the bottom. The latitude ranges from 0^o to 360^o; 0^o located centrally and increasing, from right to left, to 179^o at the far left edge whilst the far right edge starts at 180^0 and continues, from right to left, to 360^o back at the centre of the map.

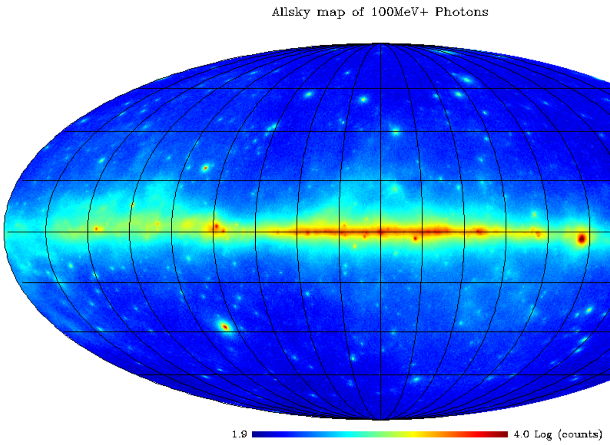


Figure 2.6: HEALPix map of photon counts - 196,608 pixel map showing the number of photons above 100MeV as recorded by the LAT since its launch. The scale is limited to 10,000 and is logarithmic

Maps like 2.6 were created for various variables throughout this study to observe different statistics, both for uniformity and anomalies. The position of every pixel remains the same in each map, and only the scale changes depending on the associated statistic being represented.

3

Data Selection

3.1 Event Classes

The data collected from the LAT is classified into four event classes, outlined in (18, 26).

1. Transient
2. Source
3. Diffuse
4. Dataclean

The transient event class tolerates the highest level of background in order to increase its statistics. This class accepts a higher level of false detections from other charged particles e.g. background electrons, muons, that could be mistaken as true photon events. The transient class contains every recorded event, from which all other event classes are a subset. This event class maximises effective search area but includes lower quality events; it is best suited for short, burst events that do not require high significance levels.

The dataclean event class is the strictest of all the event classes, with a very low background rate and less statistics as a result. This event class is suited to studying large regions for spectral features that are sensitive to background contamination. Source and diffuse are subsets of transient (as is dataclean), but with varying levels of background acceptance and effective area. Most analyses will use source and diffuse classes whilst the transient class is not used frequently as it is difficult to attach a significance to any findings due to the nature of the low quality events.

For this study it was decided that using all of the data from event classes 2, 3 and 4 was optimal to maximise the amount of statistics whilst not compromising the integrity of any results. The transient class was observed to have approximately fifty times more statistics than the sum of the remaining event classes, however it also includes false positives of events that simulated an actual photon event(26).

3.2 Fermi Science Tool Cuts

There were a number of initial decisions that needed to be made regarding the Fermi Science Tools parameters. The timeframe that was being considered was made as long as possible to give as much opportunity to record a photon burst event from an evaporating PBH as possible; the mission start time (MST) was used as the events start time, and data from all subsequent times was used. The maximum photon energy cut was left as the LAT maximum of 300GeV, the minimum was chosen to be 100MeV. As previously mentioned in the first paper by Hawking, and in a subsequent paper by Hawking and Page, photons were expected only above 100MeV(17**?**). Data between 1GeV and 300GeV was considered however there was a significantly smaller proportion of data. The zenith angle cut was left as the recommended 105^o.

The gtmktime tool used all of the recommended filters; the rocking angle compensation, only the highest data quality and only at times when the LAT was functioning perfectly. These were used in order to maximise the quality of the data.

3.3 Pixel Number

The number of pixels to break the allsky into was chosen by viewing HEALPix maps with differing numbers of pixels and by considering the amount of data being analysed. It became clear on observation of HEALPix maps that choosing an Nside ≤ 16 created too few pixels for any kind of useful image representation or source localisation.

Considering the amount of data available, an Nside above 64 would mean that there would be too little data distributed per pixel. The limited angular resolution of the LAT lead to the realisation that only an Nside of 32 would be useful for this study. An

Nside of 32 would create 12,288 pixels. The angular resolution of photons of 100MeV is 3.5^o as outlined in 2.3, a pixel number higher than 12,288 would be unphysical as the pixel sizes would smaller than the resolution of the detector. The solid angle of each pixel can be calculated by dividing the total solid angle of a sphere by the number of pixels.

$$\frac{4 \cdot \pi}{Numberofpixels} = \Omega \tag{3.1}$$

Ω = solid angle of one pixel

For 12,288 pixels, each pixel represents a solid angle of 0.001023 sternradians. This equates to approximately 5^o per pixel, slightly larger than the resolution of the LAT at 100MeV.

4

Experimental Procedure

The experimental procedure consisted almost wholly of computational analysis, utilising the clean set of data obtained from the two previously mentioned Fermi Science Tools. The procedure was broken up into several programs which acted as natural breaks in the different analysis processes, those natural breaks will also be used here as the sections within this chapter. The description of the analysis, as reported here, is in the same order as it was enacted. All of the programs, including comments, can be viewed in the appendix.

4.1 C Language Program

The first analytical procedure was conducted using a program written in C language. The main function of the program was very simple; it operated upon the .fits file created by the Fermi Science Tools and printed a list of particular data as its output. Routines included in the fitsio header were used to read from and write into .fits files (24), in combination with more regular functions. The program takes three arguments, one input file, one output file and an Nside number, it is also programmed with a failsafe message if the incorrect number of arguments are given.

4.1.1 Analytical Process 1

The program's first process is to allocate data types to the necessary variables utilised in the program, this also allocates memory to the variables depending on their length and type. The first fitisio routines in the program enact the following functions; open the input file ready for reading, move to the neccessary header/data unit (HDU) within the file, create a table pointing to the data from the file. The program then allocates memory to a number of arrays which are used later to record several variables from each event. A different set of fitsio routines then read chosen columns from the input file and record the data into several arrays, the three recorded variables are the theta (θ) and phi (ϕ) angles and event times. The theta and phi angles relate to the spatial position of events in galactic co-ordinates: longitude and latitude repsetively. These angles can be derived from equatorial co-ordinates, right ascension and declination, using the following tool from NASA (27). The event times are outputted at a later stage with no manipulation, however the theta and phi angles are converted from degrees into radians as follows.

$$\theta = \frac{\pi}{180^o} \cdot \theta \tag{4.1}$$

$$\phi = \frac{\pi}{180^o} \cdot -(\phi - 90^o) \tag{4.2}$$

The phi angles are translated from a range of $90^o to - 90^o$, into $0^o to 180^o$ due to the numbering scheme used within the HEALPix routines.

The arrays containing these new angles are subsequently processed using another fitsio routine, “ang2pix-ring”. The routine takes as its arguments the two angle arrays, an output array and the user specified argument Nside. The Nside argument informs the routine the number of pixels to dissect the allsky into, and then allocates to each pair of theta and phi angles a pixel number. Each pixel number is unique and refers to a spatial position in the 3D allsky, it is equivalent to binning photon events by source location. This list of pixel numbers is then used to make an array of the number of events within each pixel; an array with the same number of elements as total the number of pixels (12,288 for this study). The array of counts within each pixel is used within the final fitsio routine, which constructs an observable 2D image representing the 3D allsky, a HEALPix map. A HEALPix map of photon counts is shown in 2.6. The final procedure for the C language program is to print a two columned list of each event’s pixel number and event time, along with the number of pixels and total number of events.

4.1.2 Program Notes

This program was based on one originally written by Stefanie Januschek (28), it was heavily edited for this study; for all intents and purposes the code was basically rewritten. The original program included routines regarding a separate HEALPix function, however these were removed as they were unnecessary for this study. The creation of the counts array was originally programmed in a different way, however a much faster method was formulated and it is this new method that is used to save running time. The original method has been left in the program but only as a comment. The new method is estimated to be the (number of events)x faster than the method outlined in the original program. The creation of the HEALPix counts map within this program was optional once healpy became available. The HEALPix map was written into a .fits file, this file was viewed using the program Mollview in IDL.

The printed list of pixel numbers and times was created using a small Linux command when running the program. The current program prints the list onto the monitor for instant viewing, however by using a small Linux function the list was printed into a plain text file. The photon event times were recorded in terms of "mission elapsed time (MET)", in seconds. The MET is the time since the FGRT launched added to the MST. The text file containing the event times did not contain METs, but rather (METs - MST), this allowed the printed strings to be shorter thus making the next program's reading of the values faster. The MST is a fixed time so can always be added again later if necessary, however since the main aim of the study was to observe time intervals between photons the inclusion of the MST was unnecessary.

4.2 Python Language Program (Histogram Data)

The text file containing the pixel number and time of each event was used as the input for the first Python language program, its main objective was to process the events data into histogram data for each pixel. Numerical and scientific functions were used heavily in the program, namely from the scipy library, to create and manipulate numerical arrays. The program took three arguments, a text file containing the parameter data, a text file containing the event data, and a text file to write the histogram data into.

4.2.1 Analytical Process 2

The initial routines the program enacts are simply to read the data from each column into two separate arrays, this allows for direct manipulation of the data within the program. The next process to occur is the grouping of event times according to their pixel numbers. A 2D array is created with a number of elements equal to the number of pixels, and each element being an array capable of storing multiple values. The pixel number of each event is read, and the time of that event is copied into the corresponding element in the 2D array. The resulting 2D array consists of elements which contain the times of every photon event in that pixel. The next process to occur is calculating the time intervals between subsequent photons, for every pixel. It was not necessary to sort the event times in each pixel in ascending or descending order as they were already sorted, from earliest to latest. The original event data was already recorded chronologically; the processes involved in the analysis retain that order hence the omission of routines that order the times in each pixel. The next phase in the analysis is the creation of histogram data, achieved using a numpy function which requires input data and desired bin sizes; this is necessary for each element in the 2D array to create histogram data for every pixel.

The histogram binning consisted of eight bins of logarithmically increasing width, the first spanning from the LAT dead time to 10s, the second from 10s until 100s etc. This histogram data is recorded in a similar 2D array as the last, but with each element corresponding to eight values; the frequency of photon events in each bin. The histogram data is printed into a simple text file.

4.2.2 Program Notes

The program originally only took one input, a single text file containing all the event data, however the final code was altered to accomodate several smaller text files. The server the analysis took place on limited the amount of memory allocated to any one user which limited how much memory the program was allowed to use. The initial text file containing all of the event data required more memory than the server would allow, so a simple Linux command was used to split the single file into several smaller files. The histogram data created from each smaller file could be simply added together at the end as it was only recording frequency. Each file, of which there were six, was approximately 200Megabytes (MB) in size; the histogram data was much smaller in size so was easy to store.

For every pixel, the time interval between subsequent photons across two different files was calculated by temporarily storing the final events within each file, and where necessary calculating the interval between the first event of the next file and the temporarily stored time.

The time bins were chosen logarithmically to include all of the data available, but to isolate the time interval in which a PBH photon burst was expected; the shortest time bin.

4.3 Python Language Program (Likelihood Function)

The histogram data from the first python language program was used as the input file for the second python language program. The modular approach of having separate programs allowed for easier parameter changes to be made, without having to repeatedly run time-consuming routines. The program only took one input as an argument, but created a number of HEALPix maps according to user defined specifications.

4.3.1 Analytical Process 3

The program first reads the histogram data from the input file and then stores it in a 2D array. The histogram for each pixel is fitted with an expected function, based on the data from seven bins; excluding the first. The expected function for each pixel is calculated independently to account for fluctuations within each region. Each function is created by maximising the poisson likelihood for two variables. In order to fit a function to the data, the midpoints of the logarithms (log) of each time bin were taken to be x-axis data, converting the histograms into scatter plots.

The likelihood function of counts within each bin is poisson distributed, due to the nature of counting photons (29). So the combined distribution of all bins is the product of the functions, P.

$$P = \prod \frac{u^{y_i} \cdot e^{-u}}{y_i!} \tag{4.3}$$

P = likelihood function, u = expected function, y_i = actual data

The distribution of the expectation function (u) is exponential when divided by bin width, as described earlier. The fitted function is given in equation (4.4).

$$u = e^{a \cdot t_i + b} \cdot \Delta t_i \tag{4.4}$$

a = unknown parameter, b = unknown parameter, t_i =midpoint of logarithm of each bin

The actual data and expected function can be shown to be linear on a plot of the natural logarithm (natural log) of counts per bin width against the bin midpoints.

$$ln\left(\frac{u}{\Delta t_i}\right) = a \cdot x_i + b \tag{4.5}$$

In order to maximise the likelihood function, it is easier to deal with the natural log instead. The log function is monotonically increasing for all values hence the position of a maximum of the natural log is equivalent the position of a maximum in the ordinary function.

$$ln(P) = \sum -u + y_i \cdot ln(u) - ln(y_i!) \tag{4.6}$$

The maximum of equation (4.6) is found when the differentials with respect to both unknown parameters are equal to zero. Simultaneously solving both equations is necessary due to the two variables being correlated and not independent. The two differentials are those in equations (4.7) and (4.8).

$$\frac{\delta(ln(P))}{\delta a} = \sum t_i \cdot (-\Delta t_i \cdot u + y_i) \tag{4.7}$$

$$\frac{\delta(ln(P))}{\delta b} = \sum (-\Delta t_i \cdot u + y_i) \tag{4.8}$$

Equations (4.7) and (4.8) are unsolvable analytically, they must simultaneously be solved numerically, in this study by using the Newton-Raphson iteration. Due to the presence of two variables, the Newton-Raphson iteration requires a 2x2 Jacobian matrix of differentials in order to solve for correction terms. Due to equations (4.7) and (4.8) already being first order differentials the Jacobian is made up of second order differentials.

$$J = \begin{pmatrix} \frac{\delta^2(ln(P))}{\delta a^2} & \frac{\delta^2(ln(P))}{\delta a \delta b} \\ \frac{\delta^2(ln(P))}{\delta b \delta a} & \frac{\delta^2(ln(P))}{\delta b^2} \end{pmatrix} \tag{4.9}$$

J = Jacobian matrix

An initial guess for each unknown parameter is required and then a correction to each guess is calculated by solving equation (4.10), here written in matrix form.

$$\begin{pmatrix} \Delta a \\ \Delta b \end{pmatrix} = J^{-1} \begin{pmatrix} -\frac{\delta(ln(P))}{\delta a} \\ -\frac{\delta(ln(P))}{\delta b} \end{pmatrix} \tag{4.10}$$

Δa = correction term to parameter a, Δb = correction term to parameter b, J^{-1} = inverted Jacobian

The correction terms in equation (4.10) are added to the previous guesses of the unknown variables and the iteration is repeated. This process allows for repeated fitting of the function until a value within a set tolerance of zero is achieved in the first differentials. The tolerance was set at 0.0002.

In this way an expected function is calculated for the data in each pixel. A chi-squared minimisation fit was considered, however due to the nature of poisson errors and the possible occurance of zero values this was deemed unsuitable. The program continues by plotting HEALPix maps of a number of different statistics.

4.3.2 Program Notes

A pre-existing Newton-Raphson iteration solver exists in the scipy library however because the differentials were known, it was more accurate to program the iteration manually. Instead of using an approximation (e.g. Sterling's approximation) to calculate the natural logarithm of large number factorials, used in calculating the poisson log likelihood, the exact values were calculated by looping over a summation. The HEALPix maps that were created utilised the healpy library, this only became a possibility during the study so was used for convenience. The initial photon counts HEALPix map created earlier in the analysis could also be created using healpy, but like all of the maps they were optional functions that could be used or ignored where necessary.

The midpoints of the logarithm of the time bins was taken, not the logarithm of the midpoints; these two quantities are not equal and it was considered that the former was the correct quantity.

4.4 Distribution Problem

The initial observation that a problem with the fitting function existed was made by viewing the HEALPix map of the poisson log likelihood for all bins. Deviations from the expected function were observed in pixels that had no extra photon sources other than purely from background radiation, these pixels should have followed the expected function very closely. From this observation further investigation into the nature of the problem was conducted. The problem was observed when viewing the graph of the natural log of counts per bin width against the time midpoints.

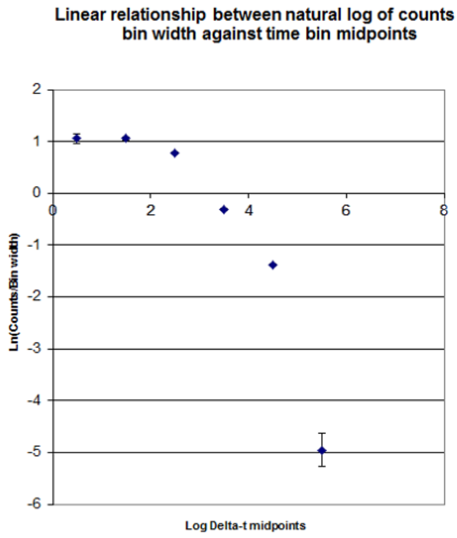


Figure 4.1: Non-Linear Photon Distribution - A non linear distribution is observed due to inherent exposure problems

The distribution is expected to be linear, however it can be seen that there is a flattening of the shape over the first three bins. This unexpected problem was prevalent enough to be a systematic problem rather than a random one. It appeared in the distribution of all of the sample pixels viewed, greatly distorting the likelihood of an exponential function fitting the data.

4.4.1 Exposure Profile

During an orbit the FGRT sweeps its field of view across the allsky to observe all points and after 2 orbits the exposure at every point is approximately equal (18); 30mins of viewing time every 2 orbits. It was realised that this exposure caused a natural biased to longer time intervals between photons. Although the exposure time may be equal for every pixel, the exposure profile could vary, and at present this information is not available. Whichever assumption of the profile is taken, the nature of intermittent exposure creates intervals between photons that are longer due to the lack of exposure between them. This causes an increase in the number of longer intervals, and decreases the number of events with shorter time intervals, causing the slight plateauing noticed in the shortest time bins. This difference is large enough to distort the data distribution sufficiently to make the likelihood function very small.

The solution to this problem was found by changing the histogram bin sizes; a linear time binning between the dead time and 10mins. It was assumed that the exposure profile for each pixel was uniform for at least 10minutes at a time hence with enough data in these bins the expected distribution should be observed at this timescale. The exposure profile was found to be uniform over 10minute intervals as expected. By using linear bins the expected function (u) will be an exponential as stated in equation (4.4), aswell as being an exponential when divided by bin width, as the width of each bin is now uniform.

It was also decided that gtmktime must not be used in order to avoid the possibility of a similar problem being caused by the "good time intervals". The intervals are very similar to those mentioned in the exposure profile but vary in length and position randomly. It was deemed that not using gtmktime was necessary to preserve the expectation distribution of the data. The exclusion of the tool meant that the overall quality of the data used in the study was lower, but this was a necessary precaution to avoid distribution distortion in one, several, or all of the pixels. The number of pixels used was kept the same as it was the smallest solid angle possible giving the best source localisation, whilst remaining within the LAT resolution capabilities.

5

Results

Due to the nature of the research, the results of the analysis can be summarised in two HEALPix maps of the allsky. 5.2 is the allsky map of all pixels with a natural log likelihood (P) less than -50. These are all of the pixels which meet the criteria of being significantly different from the expected distribution in each instance. The criteria was chosen so as to be high enough to exclude pixels of only slight variation but low enough to include all anomalous pixels. The expected signal from an evaporating PBH should be significant enough to meet this criteria. 5.2 retains a very similar structure to that of 2.6. Most of the pixels of interest are along the galactic plane or include very well known sources. In order to find a possible PBH burst this study searches for sources of distortion to the expected background distribution over short time intervals. All high energy sources which could cause significant distortion must be eliminated from consideration to prevent false positives being identified. The possibility of false negatives is discussed at the end of this report.

The pixels containing well known high energy photon sources were of no interest as the photons from evaporating PBHs would be indistinguishable from those of the sources; these pixels were not considered to be particularly significant. The sources around which pixels were not considered for further analysis are highlighted in 2.3.3.3; the Vela Pulsar, Crab Nebula, galactic plane, Geminga Pulsar and Blazar 3C454.3. These sources emit high energy photons ($\geq$ 100MeV) with a variability in time that can only be described by the light curves of each object. Various papers outlining the light curves of each source have been written with increasing accuracy over the last several decades over several energy ranges.

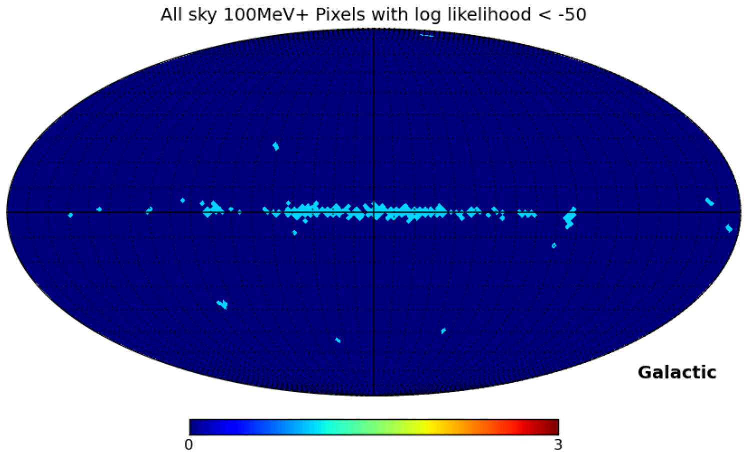


Figure 5.1: Pixels of interest - All of the pixels in light blue are those with a log likelihood less than -50, most are clustered around known sources and the galactic plane. The scale is set for contrast purposes

5.1 Candidates for further analysis

The pixels chosen for further investigation are shown in 5.1. These were those that could not immediately be attributed to the aforementioned well known high energy photon sources. The refinement process of excluding pixels was somewhat subjective and involved observing 5.2 and 2.6. By comparing the two maps and only in tandem with contextual knowledge of photon sources, most of the pixels were ruled out for further study. Only five pixels remained after eliminating all others. The pixels are as follows in order from top to bottom, left to right.

1. Pixel 54
2. Pixel 3282
3. Pixel 7584
4. Pixel 10870
5. Pixel 11094

The process by which the candidates for further analysis were selected was subjective in the sense that it relied on prior knowledge of photon sources. It was therefore

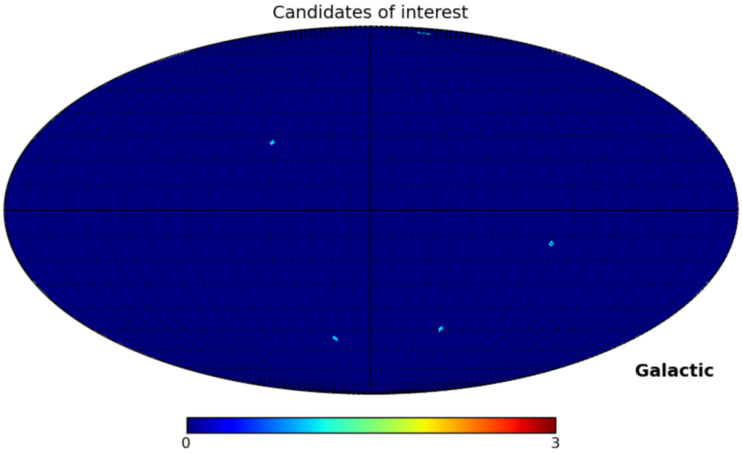


Figure 5.2: Five candidates for further analysis - Allsky map of the five pixels with log likelihood less than -50 and no well known high energy sources within them. The scale is set for contrast purposes

important to check the co-ordinates of the remaining five pixels to see if they could be accounted for by GRBs or other less well known sources. The unique identification number of each pixel allowed for the localisation of the five pixels in galactic co-ordinates by using a HEALPix routine, “pix2ang”. The GRBs recorded by the LAT (30) and all of the high energy photon sources recorded by the LAT (31) were found on the fermi website. There were only 20 GRBs with known source co-ordinates, and 1451 high energy photon sources given by the 1FGL catalog. GRBs emit a burst of high energy photons similar to those expected from a PBH, but they are distinguishable as they are over a much wider energy range and for a much longer duration. The high energy photon sources listed in the 1FGL catalog are less well known than those listed earlier, and due to their proximity and size are less noticable in 2.6. The resolution of the LAT and the size of each pixel was taken into account when deciding whether a GRB or other source was included in any of the five pixels.

Given the limited number of GRBs it was possible to directly plot their positions within 5.1 as shown in 5.3; the allsky map of the pixels of interest and GRBs. One can see immediately that four of the five pixels of interest (3282, 7584, 10870 and 11094) contain GRBs.

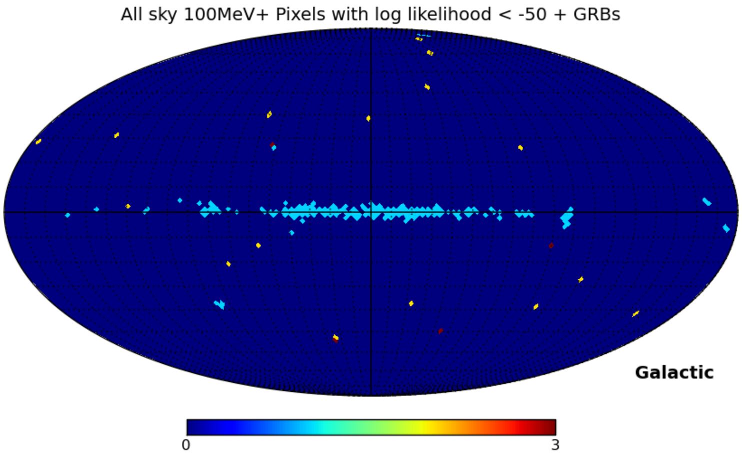


Figure 5.3: Pixels of interest and GRBS - Light blue represents pixels of interest, yellow represents GRBS and red denotes the presence of a GRB in a pixel of interest

Pixel 3282 contained the burst 090902B. The burst was a particularly bright one and is described in more detail in (32).

GRB 080916C was found within pixel 7584. This was one of the most luminous bursts ever measured and a video representation of the burst can even be found in (33). Several papers have been written about the burst pertaining to different instruments, one regarding the LAT is found here (34).

Pixel 10870 included the GRB 090926A. (35) describes the afterglow of this burst as measured by the LAT.

Pixel 11094 included 090510. The LAT observations of the burst are outlined in (36).

The number of high energy photons emitted from these bursts masks any possible PBH signals within this analysis. In a future analysis the burst could be localised to a specific time and there may exist a different signal within the same pixel, however that analysis has not been conducted within this study. The elimination of the aforementioned four pixels as possible PBH evaporation signals left just pixel 54. The 1FGL LAT catalog was used in order to check whether the pixel contained any high energy sources.

The catalog highlighted a known source within pixel 54, reference number 1FGL J1224.7+2121. This source is a high energy blazar and is well documented in (37).

There were no further pixels with a disproportionately high number of photons within a small time interval which could not be attributed to well known photon sources. No PBH evaporation signals were observed.

6

Discussion

The results of the analysis did not show any evidence of evaporating PBHs in the last 3years, as measured by the LAT. One must be careful however and make the distinction between 'no evidence' and 'evidence of nothing'. It must be stated that there was 'no evidence of evaporating PBHs', and not 'evidence of no evaporating PBHs'. The field of view and exposure profile of the LAT means that only 20% of the allsky was being observed at any time, along with the analysis itself placing some restrictions on what would be considered as evidence. The analysis would not have highlighted any PBH bursts that occured within the same pixel as a GRB and/or high energy photon source. This makes it definitively untrue to say that there is evidence of no evaporating PBHs in the last 3years.

6.1 Density Limit

The analysis highlighted no PBH evaporating signals, hence it is possible to place a probabilistic limit on the frequency of such events. The way in which this limit is created is firstly by calculating the maximum distance at which an evaporation signal could be recorded. This is done my examining the lowest fluence the LAT is capable of detecting and equating that to the energy released by a PBH. By then inverting the arrow of causation it is possible to state that because nothing was measured within this distance we have a volume over which one can calculate a density limit.

An existing limit is given in (38), based on the specifications of the EGRET. The distance over which it is predicted the EGRET can detect such a burst is between 20 and 100 pc. The density limit is stated as $5 \cdot 10^{-2} pc^{-3} yr^{-1}$, based over 2.5 years and on a total viewing time of approximately one third of that time (18). The specifications of the LAT are far superior to that of the EGRET hence a larger range of detection should be found. The EGRET field of view is only approximately 0.5 sternradians, whilst the LAT effectively views one fifth of the allsky at any one time. This should create a better estimate of the density of such events than previously.

6.1.1 Limit Calculation

According to Page and Hawking (4) the energy expected to be released in the final burst of a PBH is 10^{34}ergs. The range over which the LAT can detect PBH bursts should eclipse the minimum and maximum range of the EGRET by a significant amount. A lower and upper detection limit of PBHs by the LAT and this analysis should be calculable by looking at the maximum and minimum fluences capable of being detected.

The GRB 090902B was detected by the LAT and by this analysis, so is used within these limit calculations as it acts as a good indicator of what kinds of fluences can be measured within this study. The fluence of the burst is documented in (32) as $4.36 \cdot 10^{-4} ergcm^{-2}$, however this is the fluence over the entire energy range of the burst, starting well below what is measurable by the LAT. It is therefore important to try and calculate the fluence from 100MeV upwards.

An estimate of this photon fluence can be made to reasonable accuracy by integrating 6.1 over time. The simplest calculation would be to take the flux as constant over the first 30 seconds, and then follow the stated power law until 1000 seconds. The sum of both parts sums to a photon fluence of $0.151 countscm^{-2}$. To achieve an energy fluence however, one must either assume a flat energy range over which the photons were distributed, or to use the energy spectrum of the burst as outlined in 090902B. The complication involved with the latter is that one must also account for the time evolution of the spectrum which isn't readily available. If one assumes a flat energy range at the lower threshold of 100MeV the fluence is calculated at $8.27 \cdot 10^{-5} ergcm^{-2}$. A better calculation however is given by (39) which shows that the LAT fluence is $3 \cdot 10^{-5} ergcm^{-2}$, of the same order as our initial calculation. This is the highest LAT fluence detected from a GRB by this study whose data was readily available as seen in (39), this should allow the calculation of a lower range limit above which the LAT

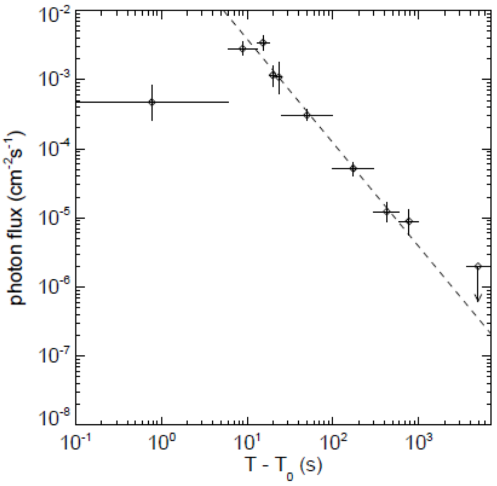


Fig. 2.— Light curve of GRB 090902B for energies 0.1–300 GeV from unbinned likelihood fits to the LAT data. After the prompt phase, extended or afterglow emission consistent with a temporal profile $\propto t^{-1.5}$ (dashed line) lasts until $\sim T_0 + 1000$ s. The upper limit at times $> T_0 + 3600$ s was derived from the data collected after the source emerged from occultation by the Earth.

Figure 6.1: Photon Flux over time from 090902B - Integrating over time gives the fluence of the burst from 100MeV upwards(32)

would not be overloaded by a PBH burst. The calculation now equates an evaporating PBH with that fluence, which the LAT is capable of detecting.

$$\frac{10^{34} erg}{4\pi R^2} = 3 \cdot 10^{-5} ergcm^{-2} \tag{6.1}$$

R = radius in parsecs

(6.1) equates to a radius of 16.69pc. This distance is relatively low, less than the EGRET lower detection distance. To calculate the upper range limit one must consider the minimum fluence the LAT can detect, one indicator of this is given in (18). It implies that a fluence as low as $10^{-7} ergcm^{-2}$ can be detected by the LAT. Repeating the same calculation creates an upper limit of 289.16pc. These results imply that an evaporation sgnal anywhere between 16.69 and 289.16pc should be measured by the LAT, provided its field of view is located on that region at the time of the burst. Previously the EGRET would have been overloaded at any range less than 20pc by a PBH burst, however the LAT should be able to detect such a burst at a closer range, and at a larger range.

The density limit can be calculated by firstly defining the volume over which the

LAT can detect such burst events. The LAT detection volume is spherical in all 3 dimensions with the detection range given by the upper range limit found earlier. One must remember not to simply calculate the spherical volume of the upper range limit, but to subtract the volume of the lower range limit from this to give the volume the LAT is capable of detecting within. One must also account for the one fifth field of view of the LAT and volume cuts made from the data. A region of approximately 20^o ($\frac{\pi}{9}$ in radians) around the galactic plane is ignored, as no signals from this region could be observed. This estimate was made by viewing the graticule in 5.1 which is set at 10^o intervals. The simplest way to calculate the observational volume, is to use spherical triple integration and set the limits accordingly.

$$V = \int_{16.69}^{289.16} r^2 dr \int_0^{\pi-\frac{\pi}{9}} sin\theta d\theta \int^{2\pi} d\phi \tag{6.2}$$

V = volume

This volume is calculated to be equal to $9.82 \cdot 10^7 pc^3$. Multiplying by one fifth, due to the field of view, gives an effective search volume of $1.96 \cdot 10^7 pc^3$. Factoring in the observation time of 3years gives a time-volume of $5.88 \cdot 10^7 pc^3 yr$.

The local distribution (within this galaxy) of PBHs can only be assumed to be isotropic, and due to their nature of being discrete and independent, poissonian. We use a poisson distribution to model the frequency of PBH's within our galaxy (and the scope of this analysis), based on asumptions of them being isotropic, independent and discrete within the region. Confidence levels can be used to determine the probability of finding an event within a given density. A confidence interval of 95% corresponds to a value of 2 sigma, hence the experimental upper density limit from this study can be calculated as below.

$$UL = \frac{2}{5.88 \cdot 10^7} \tag{6.3}$$

UL = upper density limit

UL equates to an upper limit density of $3.40 \cdot 10^{-8} pc^{-3} yr{-}1$ within a 95% confidence level.

7

Conclusion

This research was aimed at trying to find signals of evaporating PBHs due to Hawking radiation. The analysis utilised all of the data recorded by the LAT since its launch in June 2008. The analysis was mostly statistical and computational, involving the grouping of events by location and then calculating the time difference between subsequent events within each pixel. These time intervals were recorded in a histogram for every pixel, and binned into linearly increasing time bins. In each pixel a function was fitted to all bins except for the one representing the shortest time interval between photons. The number of photons in the smallest time interval bins were then compared with the expected values given by the fitted functions. The few pixels that contained extra photon events in the smallest bins were cross-referenced with known high energy sources and GRBs. No pixels remained that did not contain known sources, hence no evidence of evaporating PBHs was found in this study. A maximum distance within which a future signal should be recorded by the LAT was calculated as 289.16pc.

The lack of a signal was in some ways expected. The density limit calculated in this study exceeds that of the former EGRET limit by a significant amount which in itself is the only real result of the analysis.

7.1 Analysis Evaluation

The computational analysis in this study was created specifically for this study whilst almost all of the code was written from scratch, not an adaptation from other work. The accuracy and speed of the programs were made the priority, along with flexibility.

Some of the recorded time values included over14 significant figures, but even then the accuracy was kept in a minimum of 12 significant figures. The speed of every program was kept in mind but a modular approach was used to process data into different forms. The modular approach allowed for other programs (specifically the last) to work using processed data and avoid re-running time-consuming routines. The time taken by the first and last programs was less than 5minutes each whilst the middle program varied depending on the number of pieces of data to be read, the final data selection took approximately 90minutes to process. Each program is very flexible with variables, input files and output files given by arguments at the start of the programs. With the exception of stating the number of input files in the second Python language program, re-rerunning the analysis for any amount of data would not require any other code changes.

The subjective nature of the refinement process, in which pixels were rejected based on the existence of known sources within them, was something that could cause problems in the future. The process worked effectively in this study due to the extensive knowledge of sources provided by the research group; most of the sources were identified early and kept in mind until the refinement stage of the analysis. It is possible that for a different individual in a different department, not having such readily available knowledge would force them to analyse far more pixels individually. This process is the only one which is not run by the programs, and is therefore dependent on the individual and group the analysis is being run within.

8

Future Analysis

Although no evaporation signals were found within this study, it does not preclude the possibility of detecting a signal in the near future, as the FGRT has a remaining 7years on its expected lifespan. Any future searches for PBHs could predicate their analysis on this study by making improvements, or attempt a completely different approach.

8.1 Improvements

As mentioned earlier, there is little that could be changed within the programs to increase speed or accuracy. The nature of the analysis could be changed, however this would be an extension or different type of analysis rather than an improvement. An obvious improvement involves the location of the GRBs detected by the LAT. The fermi data outlining the details of the GRBs is outdated and in some cases contradictory. The location of the GRBs is given in equatorial co-ordinates, in hours:minutes:seconds/degrees:arc-minutes:arc-seconds aswell as in degrees. The former co-ordinate system was initially used to convert the GRB locations into galactic co-ordinates (27) and then plot them in HEALPix maps, however it became obvious that there was a problem with the locations, see 8.1.

The locations of the GRBs is independent and should be distributed evenly; further investigations highlighted inconsistencies in the GRB data (30). The locations given in degrees were used instead and the correct distribution was found, as shown in 8.2. The error in locations was due to the inconsistent data provided by the LAT team, and not due to this analysis.

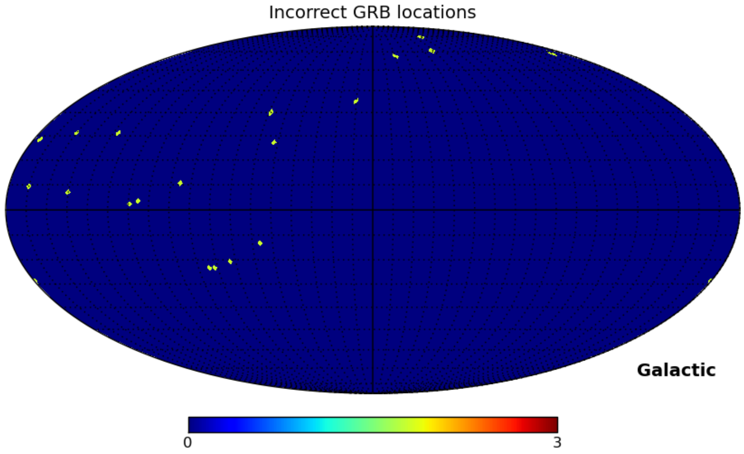


Figure 8.1: HEALPix map of false GRB locations - Several of the GRB locations are in incorrect locations, and create the observed distriubtion, with only one event in the bottom right quadrant

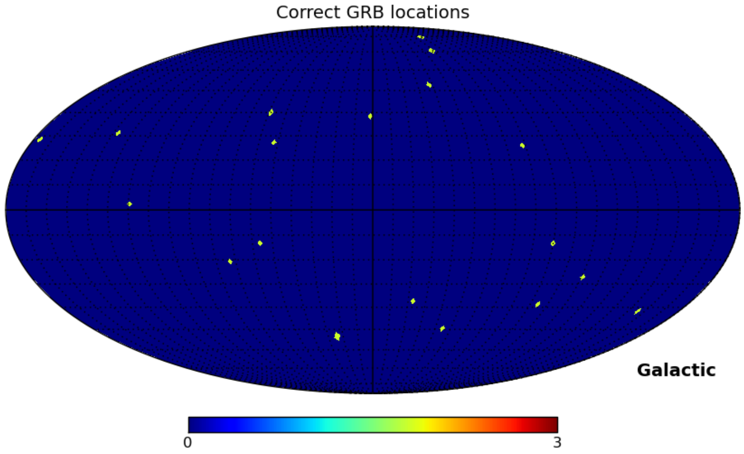


Figure 8.2: HEALPix map of true GRB locations - The location of every GRB is correct and shows an even distribution, as expected

The problem was found after extensive searching to be only a small typo in a few instances where a minus sign was omitted. The length of the co-ordinates given in the form of hours:minutes:seconds/degrees:arc-minutes:arc-seconds were exactly equal to the width of the columns in the data tables. The minus sign on the necessary GRB locations was therefore a character too long and not shown in the table. The co-ordinates in degrees was much shorter and the minus sign conserved for all GRBs. The most likely explanation of such an error is a miscommunication between the data team and the website designers over the width of the columns.

The recent fermi symposium highlighted the more recent data with an extra three GRBs, this data has still not been updated on the fermi website. If this analysis were repeated it would be vital that the fermi team be contacted directly in order to request more up to date data and to correct the pre-existing typos.

Future analysis would be over a larger time, which would provide more data. Possibly, using the transient event class could also increase the probablity of finding an evaporation signal, however methods to account for non-photonic events and factor in the lower significance of any findings would have to be devised. Not finding any signals in the future would still allow for a bettering of the current density limit outlined in this study, or even to calculate it using a more precise method.

8.2 Extensions

The analysis in this study is based on creating time interval histograms and expected functions based on these histograms. A simpler yet possibly more effective method would be that of a "sliding time window". This type of analysis would give a better idea of the time distribution of photons in each pixel and is equivalent to calculating a moving average. The analysis would be based on creating a timeframe of approximately one minute, and moving it across the entire time since the LAT launch in one second increments. The 'sliding window' would count the number of photon events within that minute and plot a continuous distribution of the number of photon events against time. This method would be computationally very simple to write, however would require a dedicated server to ensure it does not take an inordinate amount of time. The number of photons as a function of time could be observed for each pixel and regions of consistently high photon numbers could be disregarded, whereas transient burst events would be seen as single peaks in the distribution. This type of analysis would also mean that the exposure profile of each pixel need not be factored out of the analysis, as these times would simply represent zero counts for the duration of non exposure. The analysis would not only highlight pixels that included burst events, but without any further investigation, the exact time of such events. Given only a few weeks this analysis could take place by utilising the existing C language program to read the event data, and then adapting the first Python language program to create time distributions instead of histograms.

In future studies, continued failure to locate PBHs using larger datasets could cast doubt over the theory predicting their formation or Hawking radiation. This study and that of the EGRET did not detect any evaporation signals, however only further studies could corroborate these results or confirm the theory.

References

[1] Hawking. **Black hole explosions?** *Nature*, 1974. 2, 7, 11

[2] Ukwatta. **Sensitivity of the FERMI Detectors to Gamma-Ray Bursts from Evaporating Primordial Black Holes**. *arXiv1003.4515U*, 2010. 2, 11

[3] http://fermi.gsfc.nasa.gov/public/. 2, 8

[4] Page and Hawking. **Particle emission rates from a black hole**. 1976. 2, 35

[5] **http://mathworld.wolfram.com/ExponentialDistribution.html**. 4

[6] http://casa.colorado.edu/ ajsh/singularity.html. 4

[7] http://library.thinkquest.org/C0110369/Defn.htm. 5

[8] Psaltis Johannse. **Testing the No-Hair Theorem with Observations in the Electromagnetic Spectrum: I. Properties of a Quasi-Kerr Spacetime**. *arXiv:1003.3415*, 2010. 5, 6

[9] http://en.wikipedia.org/wiki/Schwarzschild radius. 5

[10] Schwarzschild. **Uber das Gravitationsfeld einer Kugel aus inkompressibler Flussigkeit nach der Einsteinschen Theorie**. *Sitzungsberichte der Deutschen Akademie der Wissenschaften zu Berlin, Klasse fur Mathematik, Physik, und Technik*, 1916. 5

[11] http://en.wikipedia.org/wiki/Black hole. 6

[12] Ott O'Connor. **Black Hole Formation in Failing Core-Collapse Supernovae**. *arXiv:1010.5550*, 2010. 6

[13] Musco. **Primordial black hole formation**. *Black Holes in General Relativity and String Theory*, 2008. 7

[14] Jedamzik Niemeyer. **Dynamics of Primordial Black Hole Formation**. *arXiv:astro-ph/9901292*, 1999. 7

[15] Novikov Zeldovich. **Astron.Zh. 43 758**. 1966. 7

[16] Halzen et al. **Gamma rays and energetic particles from primordial black holes**. *Nature*, 1991. 7, 8

[17] Carr. **Primordial Black Holes: Do They Exist and Are They Useful?** *arXiv:astro-ph/0511743v1*, 2005. 7, 8, 9, 18

[18] Atwood et al. **The Large Area Telescope on the Fermi Gamma-ray Space Telescope Mission**. *arXiv:0902.1089*, 2009. 9, 10, 11, 12, 17, 28, 35, 36

[19] http://heasarc.nasa.gov/docs/cgro/egret/egret doc.html. 9

[20] Moiseev. **Astropart. Phys., 27, 339**. 2007. 11

[21] http://fermi.gsfc.nasa.gov/ssc/data/analysis/documentation/Cicerone/. 12, 13

[22] http://healpix.jpl.nasa.gov/. 13, 14

[23] HEALPix Primer. 14, 15

[24] http://heasarc.gsfc.nasa.gov/docs/software/fitsio. 15, 20

[25] http://code.google.com/p/healpy/. 15

[26] http://fermi.gsfc.nasa.gov/ssc/data/analysis/documentation/Cicerone/Cicerone Data/LAT-DP.html. 17, 18

[27] http://heasarc.nasa.gov/cgi bin/Tools/convcoord/convcoord.pl. 21, 40

[28] Stefanie Januschek. **Analyse zur Detektierbarkeit von Strukturen Dunkler Materie-Halos mit dem Fermi-LAT-Experiment**. 2011. 22

[29] http://plasma.physics.wisc.edu/uploadedfiles/journal/Stoneking300.pdf. 24

[30] http://fermi.gsfc.nasa.gov/ssc/observations/types/grbs/grb table/. 31, 40

[31] http://fermi.gsfc.nasa.gov/ssc/data/access/lat/1yr catalog/1FGL catalog submitted.pdf. 31

[32] Abdo et al. **Fermi Observations of GRB 090902B A Distinct Spectral Component in the Prompt and Delayed Emission**. 2009. 32, 35, 36

[33] Omodei and Ritz. **http://www.archive.org/details/GMM-10344**. 2009. 32

[34] LAT and GBM Collaboration. **Fermi Observations of High-Energy Gamma Ray Emission from GRB 080916C**. 2009. 32

[35] Swenson et al. **GRB 090926A and Bright Late-time Fermi LAT GRB Afterglows**. 2010. 32

[36] Lat and GBM Collaboration. **Fermi Observations of GRB 090510**. 2010. 32

[37] Francesco Verrecchia. **Swift follow-up of the blazar 4C+21.35 after a bright gamma-ray flare**. 2010. 33

[38] Fichtel et al. **Search of the energetic gamma-ray experiemnt telescope data for high energy gamma-ray microsecond bursts**. 1994. 34

[39] James Chiang. **Fermi observations of GRB 090902B, fermi symposium**. 2009. 35

```
//C language program (1)

#include <stdio.h>
#include <string.h>
#include "fitsio.h"
#include <chealpix.h>
#include <stdlib.h>
#include <math.h>

#define PI 3.141592654

int main(int argc, char *argv[])
{

if (argc!= 4)
      {
       printf("\n");
       printf("\tThis program converts an events(.fits) file into file
pixelised in HEALPix.\n");
       printf("\tA counts table is created that lists which HEALPix are
used.\n");
       printf("\tTherefore the HEALPix that best fits certain l and b
values of the events file is calculated.\n\n\t");


       printf("\t\tUsage: ang2HEALPix-print <Infile Name> <Outfile Name>
<nside>\n:");
       return(0);
      }
/* nside MUST BE 2, 4, 8, 16, 32, 64 etc...*/

fitsfile *infptr; /*pointer to input and output .fits files*/
fitsfile *outptr;
int status, colnum, anynulls, i, j, hdutype, nfound, nkeys, morekeys = 0,
nullanz = 0;
long naxes[2], frow, felem, nside=atof(argv[3]), ipring, naxis2,
LATstart=239557417; /*atof converts nside string into double*/
float nullval; /*ipring, pixel identification number in ring*/
char infilename[] = "argv[1]"; /*name for existing .fits file*/
char healpixmap[] = "argv[2]"; /*name for new HEALPix map*/
char coordsys[15] = "Galactic";

status = 0;


/*open the existing .fits files and move to look at 2nd HDU in fits
file*/
fits_open_file(&infptr, argv[1], READONLY, &status);

fits_movabs_hdu(infptr, 2, &hdutype, &status); /*move to 2nd HDU (can see
2nd HDU in fv), specific data for each event*/

fits_read_keys_lng(infptr, "NAXIS", 1, 2, naxes, &nfound, &status);
/*NAXIS 1 and 2 gives table size*/

naxis2 = naxes[1]; /*naxis2 = number of rows(events)*/
```

```
long healpix = 12*nside*nside; /*from number of sides to number of
pixels*

/*allocate memory to arrays. size of each element x number of elements
(nside). cant directly create array of*/
/*nside elements, as cant use dynamic input*/
float *counts = malloc(healpix*sizeof(*counts));
float *arrayl = malloc(naxis2*sizeof(*arrayl));
float *arrayb = malloc(naxis2*sizeof(*arrayb)); /*Use sizeof(*name)
instead of type as then can just change type at start*/
double *arrayt = malloc(naxis2*sizeof(*arrayt)); /*and dont have to
change each type in brackets too, if change required*/
long *ipringctrl;
ipringctrl = calloc(naxis2,sizeof(*ipringctrl)); /*changed from original
code from healpix to naxis2, otherwise not*/
                                    /*enough memory allocated to for loop
of iprings*/

if (hdutype!= BINARY_TBL)
      {
       printf("Error: a binary table was expected in this HDU\n"); /*2nd
HDU is binary table format*/
       return;
      }

frow = 1; /*first row in table to write*/
felem = 1; /*first element in table to write*/
nullval = -400.;

/*read the 1 and b values out of the .fits file*/
fits_read_col(infptr, TFLOAT, 4, frow, felem, naxes[1], &nullval, arrayl,
&anynulls, &status); /*read column 4 into arrayl*/
fits_read_col(infptr, TFLOAT, 5, frow, felem, naxes[1], &nullval, arrayb,
&anynulls, &status);
fits_read_col(infptr, TDOUBLE, 10, frow, felem, naxes[1], &nullval,
arrayt, &anynulls, &status);


/*calculate the ipringctrl array for certain values of 1 and b*/
for(i=0; i<naxis2; i++)
      {
/*Convert to radians, for arrayb need to convert from -90->90 scale to 0-
>180, with 0 at north due to ang2pix reference settings, see healpix c
routines*/
       arrayl[i] = (PI/180.)*arrayl[i];
       arrayb[i] = (PI/180.)*-(arrayb[i]-90.);

       ang2pix_ring(nside, arrayb[i], arrayl[i], &ipring); /*renders the
pixel number ipring for a pixel which*/
                             /*resolution parameter nside, contains the
point on the sphere*/
       ipringctrl[i] = ipring;                    /*at angular
coordinates from arrayl and arrayb.*/

      }

/*Print out details of number of events, healpix, and array data for
python inputs*/
```

```
      printf("%li\n", naxis2);
      printf("%li\n", healpix);
/*Note. Printed time has the LAT start time minused from it, makes
smaller numbers for faster computing in Python*/
for(i=0; i<naxis2; i++)
      {
       printf("%li ", ipringctrl[i]);
       printf("%f\n", arrayt[i] - LATstart);
      }

/*Create counts map, much faster way of reading the events. Only read
naxis2 events*/
for(i=0; i<naxis2; i++)
      {
       counts[ipringctrl[i]]++;
      }


/*Original way of creating counts array, much slower than new method
though. Read naxis2*healpix events*/
/*for(i=0; i<healpix; i++)
      {
       float anzahl=0.;

       for(j=0; j<naxis2; j++)
            {
             if(i==ipringctrl[j])
             anzahl++;
            }

       counts[i] = anzahl;

       if(counts[i]==0)
       nullanz++;
      }*/

/*write the arrays' into a healpix map*/
int control = -1;

control = write_healpix_map(counts, nside, argv[2], 0, &coordsys[0]); /*0
pixel ordering gives ring and not nested*/

free(arrayl);
free(arrayb);
free(arrayt);
free(counts);
free(ipringctrl);

return 0;

}
```

```
#Python Language Program (2)

#Import text file with values from C code, then manipulate everything
easier in Python
#Had to use unix function 'split' to split text files into smallers files
approximately 200MB each, avoid memory error
#Used split with line numbers (9,200,000 lines per part) so no one time
value or pixel value left behind

import sys
import numpy as np

#Start time of LAT for reference. All imported times have this reference
time minused from them to make them shorter
LATstart=239557417

"""Open files and extract data"""

#sys allows the file to be read to be specified as an argument instead of
changing code each time want to use new file
Param = open(sys.argv[1])
Pixels = open(sys.argv[2], 'w')

#Reads the first line of text to find number of events and the second
line for number of pixels
#Don't need to change line, done automatically because of cursor movement
when reading using Python
naxis2=long(Param.readline())
healpix=long(Param.readline())

#Create array of histogram data outside of for loop, so it will be saved
for each file.
n =[]
temparray = []

###!! CHANGE Z VALUE, STARTING FROM 3 UPWARDS TO SPECIFY NUMBER OF FILES
!!###
for z in range(3,9):
#Had to do it this way with multiple infiles, otherwise get memory error.
#Create a variable that can refer to each infile for each loop. Each file
is closed at the end of the loop, memory saving.
    Arrays = open(sys.argv[z])

#Create arrays within the loop, so that for each cycle they are emptied,
and dont hold the previous infiles data.
    pta = []
    arraydelta = []

#Clever numpy routine genfromtxt
#Reads two columns into two arrays whilst converting them at the same
time, gives two arrays
    print 'Opening ipring file{0}'.format(z)
    ipringctrl = np.genfromtxt(Arrays, delimiter=' ', dtype=long,
usecols=(0))
#Reset reader to start of script, then skip the first two rows of naxis2
and healpix
    Arrays.seek(0, 0)
```

```
#Had to make variable type floats even though after subtraction but still
get 5d.p. of accuracy.
#Decimal not allowed in matplotlib so must work in floats, faster too.
#Even working from string->decimal->float, still get float errors on
accuracy. Unavoidable
    print 'Opening time file{0}'.format(z)
    arrayt = np.genfromtxt(Arrays, delimiter=' ', dtype=float,
usecols=(1))
    print 'Closing file{0}'.format(z)
    Arrays.close()


    """Create arraydelta array, where each row corresponds to a pixel;
each element is the time of an event in that pixel"""

#Create array with healpix number of arrays as the elements
    for i in range(healpix):
        pta.append([])
        arraydelta.append([])

#Created this method which takes thousands of times less time to run than
the original C method
#Moves each value in one read to correct pta row, than having to loop
over for every value filling from 0, 1, 2...etc
    for i in range(len(ipringctrl)):
        pta[ipringctrl[i]].append(arrayt[i])

#Create array of delta-t's. Have to use different case depending on which
infile in use, need to refer to last entries.
    if (z==3):
        for i in range(len(pta)):
            for j in range(len(pta[i])-1):
                arraydelta[i].append(pta[i][j+1] - pta[i][j])
            temparray.append(pta[i][-1])
    else:
        for i in range(len(pta)):
            arraydelta[i].append(pta[i][0] - temparray[i])
        temparray = []
        for i in range(len(pta)):
            for j in range(len(pta[i])-1):
                arraydelta[i].append(pta[i][j+1] - pta[i][j])
            temparray.append(pta[i][-1])

    """Create histogram data"""

#For every pixel, do the calculations
    for x in range(len(pta)):

#Create frequency and bin values.
        (f, bins) = np.histogram(arraydelta[x],
bins=(0.00001,60,120,180,240,300,360,420,480,540,600))
        if (z==3):
            n.append(f)
        else:
            n[x] = np.add(n[x],f)

#Print frequency values. Do it in this way to get formatting right
(without brackets around each row)
```

```
for x in range(len(n)):
   print>>Pixels
,n[x][0],n[x][1],n[x][2],n[x][3],n[x][4],n[x][5],n[x][6],n[x][7],n[x][8],
n[x][9]

Param.close()
Pixels.close()
```

```
#Python Language Program (3)

#Import histogram data from first Python code. Better to keep different
codes separate.
#Take histogram data into a 2D array.

import sys
import numpy as np
import scipy
import os
import math
import matplotlib
matplotlib.use('TkAgg')
import matplotlib.mlab as mlab
import matplotlib.pyplot as plt
sys.path.append('/d4/alexgw/SOFTWARE/python')
import healpy
import pylab

#Can only use code on crf server due to healpy location not being on this
computer

#Start time of LAT for reference. All imported times have this reference
time minused from them to make them shorter
LATstart=239557417

#Value for exp^1
E = 2.7182818284590451

#'Midpoints of the log delta-time bins'; same for all pixels. NOTE not
same as 'log of midpoints of delta-time bins'.
lm = np.array([30.0, 90.0, 150.0, 210.0, 270.0, 330.0, 390.0, 450.0,
510.0, 570.0])

binwidth = np.array([60.0, 60.0, 60.0, 60.0, 60.0, 60.0, 60.0, 60.0,
60.0, 60.0])

"""Open files and extract data"""

#sys allows the file to be read to be specified as an argument instead of
changing code each time want to use new file
HistData = open(sys.argv[1])
#Open file containing coordinates of all GRBs
GRBData = open(sys.argv[2])

#Read histogram arrays into 2D array called Counts. Each element is the
histogram data for one pixel.
#Length of Counts is same as number of healpix. All numbers addeded
together give number of events, naxis2.
Counts = np.genfromtxt(HistData, dtype=int)

#Read GRB angles in degrees into two arays
l = np.genfromtxt(GRBData, dtype=float, usecols=(0))
GRBData.seek(0, 0)
b = np.genfromtxt(GRBData, dtype=float, usecols=(1))
GRBData.close()
```

```
#Convert into radians. Translation required to change co-ordinate scale
to match healpix. See HEALPix Notes.
for i in range(len(b)):
    l[i] = l[i]*math.pi/180.0
    b[i] = -(b[i]-90.0)*math.pi/180.0

#Allocate pixel number
GRB = healpy.ang2pix(32, b, l, 0)

#Create arrays needed for healpix maps
LnLike = np.zeros(len(Counts))
Interest = np.zeros(len(Counts))

"""Try and find prediction values by maximising log likelihood of poisson
distribution"""
#Starting guesses for a and b. If data well behaved like 100MeV+ data,
easier to leave starting guesses as last pixel solutions.
#If data badly behaved, need to put guesses inside for loop to reset each
time, each rate may not be similar to last.
a = -0.001
b = 1.0

#Fit function u=binwidth*exp(ax + b). Maximise poisson likelihood and use
2function Newton Raphson iteration.
#Plotting ln(u/binwidth) against log(delta-t midpoints) gives linear best
line on data.
for x in range(len(Counts)):

    for i in range(100):
#Calculate derivative values and then see if they are close enough to
zero to be considered good fits.
        dlnPda = sum(lm[1:]*(-binwidth[1:]*(E**(a*lm[1:] + b)) +
Counts[x][1:]))

        dlnPdb = sum(-binwidth[1:]*(E**(a*lm[1:] + b)) + Counts[x][1:])

        if(abs(dlnPda) < 0.0002 and abs(dlnPdb) < 0.0002):
            break

        elif(i==99):
            print 'Iteration failed for pixel {0}'.format(x), Counts[x]
            exit()

        else:
#Need to compute deltaa and deltab to add to the starting values
#Computer second derivates to make Jacobian. Clairaut's theorem means
d2lnPdadb = d2lnPdbda
            d2lnPda2 = sum((-lm[1:]**2)*(binwidth[1:]*(E**(a*lm[1:] +
b))))

            d2lnPdb2 = sum(-binwidth[1:]*(E**(a*lm[1:] + b)))

            d2lnPdadb = sum(-lm[1:]*(binwidth[1:]*(E**(a*lm[1:] + b))))

#Invert Jacobian and solve to find delta-a and delta-b. Add these to a
and b and restart iteration.
            Deltaa = (-d2lnPdb2*dlnPda +
d2lnPdadb*dlnPdb)/(d2lnPdb2*d2lnPda2 - (d2lnPdadb**2))
```

```
            Deltab = (d2lnPdadb*dlnPda -
d2lnPda2*dlnPdb)/(d2lnPdb2*d2lnPda2 - (d2lnPdadb**2))

        a = a + Deltaa
        b = b + Deltab

#Create temporary array storing values of natural log of counts for each
bin.
    LnCountsFact = np.zeros(len(Counts[x]))
    for i in range(len(Counts[x])):
        Temp = 0.0
#Using this additive approach as it is completely accurate for large
factorials isntead of an approximation.
        for j in range(Counts[x][i]):
            Temp = np.log(j+1) + Temp
        LnCountsFact[i] = Temp

#Add this pixels values of LogLikelihood for first and all bins to
arrays.
    LnLike[x] = sum(-binwidth*(E**(a*lm + b)) + Counts[x]*(a*lm + b +
np.log(binwidth)) - LnCountsFact)

#Create array representing interesting pixels
    if(LnLike[x] < -50):
        Interest[x] = 1
    else:
        Interest[x] = 0

#Use healpy to plot allsky map of whichever array is required.

#Poisson Log Likelihood for all bins
healpy.mollview(LnLike, coord='g',fig=0, min=-500.0, title='All sky
100MeV+ Poisson Log Likelihood for all bins')
healpy.graticule(dmer = 10.0, dpar = 10.0, coord = 'G', color = 'black')
plt.savefig("/d4/jbains/AllskyPoissonLogLikelihood-32N.png")
plt.clf()

#Interesting pixels, 0 = nothing, 1 = candidate, 2 = GRB, 3 = candidate +
GRB
for i in (GRB):
    if (Interest[i] == 1):
        Interest[i] = 3
    else:
        Interest[i] = 2

healpy.mollview(Interest, coord='g',fig=0, title='All sky 100MeV+ Pixels
with log likelihood < -50 and GRBs')
healpy.graticule(dmer = 10.0, dpar = 10.0, coord = 'G', color = 'black')
plt.savefig("/d4/jbains/AllskyInterestPixels-32N.png")
plt.clf()

#Photon Counts map
PixCounts = np.zeros(len(Counts))
for i in range(len(Counts)):
    PixCounts[i] = sum(Counts[i])
```

```
healpy.mollview(PixCounts, coord='g',fig=0, max = 5000.0, title='All sky
100MeV+ Photon Counts')
healpy.graticule(dmer = 10.0, dpar = 10.0, coord = 'G', color = 'black')
plt.savefig("/d4/jbains/AllskyCounts-32N.png")
plt.clf()

HistData.close()
```