Indian Academy of Sciences, Bengaluru
Indian National Science Academy, New Delhi
The National Academy of Sciences India, Prayagraj

SUMMER RESEARCH FELLOWSHIPS — 2021

Format for the Final-week Report*

Name of the candidate : Vikash K

Application Registration no. : PHYS411

Date of Commencement of work : 15/06/2021

Mode of work : From Home: ✓ Guide's Laboratory:

Date of completion : 25/08/2021

Total no. of days worked : 62

Name of the guide : Dr Shashi Bhushan Pandey

Guide's institution : Aryabhatta Research Institute of Observational Sciences

Project title : Microphysics Parameters Estimation - GRB 130603B

Address with pin code to which the certificate could be sent:

No 60, 2nd main road, 1st part, V.G.P Layout, Palavakkam,
Chennai - 600041, TamilNadu.

E-mail ID: vikkykotty@gmail.com

Phone No: 7904099475

TA Form attached with final report (not applicable for those **working from home**) : YES _______ NO _______

If, NO, Please specify reason

Signature of the candidate

Date: 04/09/2021

Signature of the guide

Date: ______________

*The final report could be anywhere between 20 and 25 pages including tables, figures etc.
This format should be the first page of the report and should be stapled with the main report.

(For office use only; do not fill/tear)

Candidate's name:	Fellowship amount:
Student: Teacher:	Deduction:
Guide's name:	TA fare:
KVPY Fellow: INSPIRE Fellow:	Amount to be paid:
PFMS Unique Code:	A/c holder's name:
Others	

Microphysics Parameters Estimation - GRB 130603B

Vikash Kotteeswaran

August 29, 2021

Abstract

This work primarily focuses on finding the External forward shock wave parameters of GRB 130603B considering a constant ambient density. GRB 130603B is a short gamma ray burst with burst duration of 0.18s (T_{90}) and is supplemented with Radio, Optical and X-rays observation data. Simultaneous multi-wavelength observation starts from 0.37 days after the GRB trigger. The observed light curves and the Spectral Energy Distribution (SED) that can be obtained using the observed data points are sufficient enough to estimate the spectral breaks. The spectral breaks are then inverted to estimate the external forward shock parameters. The half opening angle of GRB jets and the jet equivalent kinetic energy of the blast wave are also estimated along with them.

1 Introduction to the theory

GRBs are the most fascinating fireworks in the universe and, there is still a lot to be learnt about them. These are believed to release their energies through jets and later in their times, these resemble an expanding isotropic fireball ($i.e$) Expanding fireball model could explain the external forward shock at later times. These apparent fireballs are the shocks produced by the central engine. These jets form forward and reverse shocks resulting from the collision with its ambient medium. There can also be energy injection to the shocks from shocks colliding to themselves due to the jets of long active inner engine but this work is not going to consider external shocks with energy injection and we are interested in forward shock waves and these shocks are characterized with some physical parameters p (electron distribution index), ϵ_e, ϵ_B (fractional energy in the accelerated electrons and magnetic fields immediately behind the shock wave respectively), their ambient density profile (n and A_*; n is the number density of protons in constant density medium, A_* is density parameter for a wind like Circumburst Medium defined as $n(R) = 3\times10^{35}A_*R^{-2}$, $A_* = \dot{M}_w/(4\pi V_w \times 5\times10^{11})\ g/cm$, $\dot{M}_w$ is the mass-loss rate, and V_w is the wind velocity) and E^K_{iso} (the isotropic equivalent kinetic energy of the blast wave). The half opening angle of GRB jets (θ_j) and the jet equivalent kinetic energy of the blast wave (E^K_{jet}) is also estimated using the aforementioned quantities.

1.1 Synchrotron process

The process that takes place inside the forward external shock is the Synchrotron process, the spectrum is a result of a distribution of electrons following the Synchrotron process due to the amplified magnetic field behind the forward shock waves. The electrons injected into the system follow a power-law distribution with a minimum Lorentz factor of γ_m (This is the most probable LF of the electrons too and ν_m is the associated frequency with γ_m). The Synchrotron cooling takes place as a result of the decrease in energy of the electrons to Synchrotron radiation and cooling is significant if Lorentz factors of the electrons are above a certain Lorentz factor γ_c (and ν_c is the associated frequency with γ_c). Based on this, they are two types of cooling, fast and slow cooling: when γ_m is higher than γ_c then there occurs fast cooling since LF (Lorentz factor) of most of the electrons are higher than γ_c, and when γ_m is lower than γ_c then there occurs slow cooling since there isn't significant cooling. It is observed that most of the GRBs follow slow cooling and so γ_c is higher than γ_m in most cases ($\nu_c > \nu_m$). Also, as a result of a distribution of electrons, several additional phenomena take places, such as Synchrotron Self Absorption and Inverse Compton radiation. Synchrotron self-absorption is where electrons absorb the emitted Synchrotron radiation. This becomes significant below a characteristic LF γ_a (below ν_a). The shape of the Synchrotron radiation spectrum depends mainly on the Synchrotron cooling effect as it affects the electron distribution. In the case of slow cooling, the distributions are in such a way that the spectrum between $\nu_m < \nu < \nu_c$ is proportional to $\nu^{-\frac{(p-1)}{2}}$, between $\nu_a < \nu < \nu_m$ is proportional to $\nu^{\frac{1}{3}}$ and beyond $\nu > \nu_c$ is proportional to $\nu^{-\frac{p}{2}}$.

The Inverse Compton process takes place because of the high energy electrons present behind the shock and their high number density, the electrons uplift the energy of the photons from the shock to higher energies by trans-

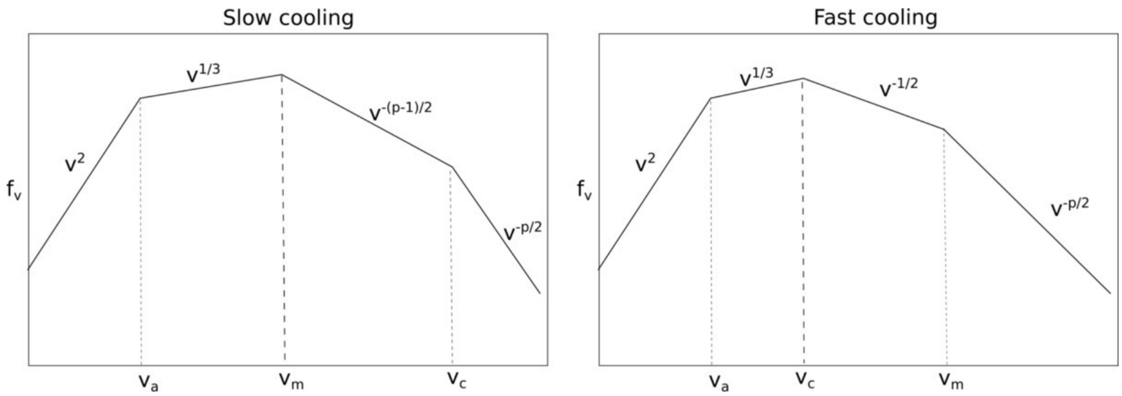

Figure 1: Shape of the Synchrotron spectrum in the case of slow and fast cooling.

ferring their part of the energy to the photons. The Inverse Compton process also changes the shape of the spectrum *i.e* its temporal and spectral slopes. The strength of the Inverse Compton effect depends on the Compton factor $Y = P_{IC}/P_{Syn}$. In our case, it will be assumed that $Y < 1$. The above spectrum is the one that is fitted with the observed data to model the SED (Spectral Energy Distribution).

1.2 Synchrotron spectrum and light curve

The flux from the external forward shock which undergoes the Synchrotron process is given by [SPN98],

In the case of Slow cooling,

$$f_\nu = \begin{cases} f_{\nu,\max}\left(\frac{\nu_a}{\nu_m}\right)^{1/3}\left(\frac{\nu}{\nu_a}\right)^2, & \nu < \nu_a \\ f_{\nu,\max}\left(\frac{\nu}{\nu_m}\right)^{1/3}, & \nu_a < \nu < \nu_m \\ f_{\nu,\max}\left(\frac{\nu}{\nu_m}\right)^{-(p-1)/2}, & \nu_m < \nu < \nu_c \\ f_{\nu,\max}\left(\frac{\nu_c}{\nu_m}\right)^{-(p-1)/2}\left(\frac{\nu}{\nu_c}\right)^{-p/2} & \nu > \nu_c \end{cases} \tag{1}$$

In the case of Fast cooling,

$$f_\nu = \begin{cases} f_{\nu,\max}\left(\frac{\nu_a}{\nu_c}\right)^{1/3}\left(\frac{\nu}{\nu_a}\right)^2, & \nu < \nu_a \\ f_{\nu,\max}\left(\frac{\nu}{\nu_c}\right)^{1/3}, & \nu_a < \nu < \nu_c \\ f_{\nu,\max}\left(\frac{\nu}{\nu_c}\right)^{-1/2}, & \nu_c < \nu < \nu_m \\ f_{\nu,\max}\left(\frac{\nu_m}{\nu_c}\right)^{-1/2}\left(\frac{\nu}{\nu_m}\right)^{-p/2}. & \nu > \nu_m \end{cases} \tag{2}$$

$f_{\nu,max}$ is the maximum flux of the spectrum, It is supposedly the flux at ν_c in the case of fast cooling and ν_m in the case of slow cooling. The expressions for the spectral breaks and $f_{\nu,max}$ have been derived for both the constant medium and wind medium cases [GS02], they are,

For a constant medium,

$$\nu_m = 3.73 \times 10^{15}\ \text{Hz} \times (p-0.67)(1+z)^{1/2}\epsilon_B^{1/2}\left[\epsilon_e g(p)\right]^2 E_{52}^{1/2} t_{obs,d}^{-3/2} \tag{3}$$

$$\nu_c = 6.37 \times 10^{13}\ \text{Hz} \times (p-0.46) \times e^{-1.16p}(1+z)^{-1/2}\epsilon_B^{-3/2}E_{52}^{-1/2}n_p^{-1}t_{obs,d}^{-1/2} \tag{4}$$

$$\nu_a = 1.24 \times 10^{9}\ \text{Hz} \times (1+z)^{-1}\left(\frac{p-1}{3p+2}\right)^{3/5}\left[\epsilon_e g(p)\right]^{-1}\epsilon_B^{1/5}E_{52}^{1/5}n_p^{3/5} \tag{5}$$

$$f_{\nu,\max} = 9.93\text{mJy} \times (p+0.14)(1+z)\epsilon_B^{1/2}E_{52}n_p^{1/2}D_{L,28}^{-2}, \quad n_p \propto R^0 \tag{6}$$

For a wind-like medium,

$$\nu_m = 4.02 \times 10^{15}\ \text{Hz} \times (p-0.69)(1+z)^{1/2}\epsilon_B^{1/2}\left[\epsilon_e g(p)\right]^2 E_{52}^{1/2} t_{obs,d}^{-3/2} \tag{7}$$

$$\nu_c = 4.4 \times 10^{10}\ \text{Hz} \times (3.45-p)e^{0.45p}(1+z)^{-3/2}\epsilon_B^{-3/2}E_{52}^{1/2}A_*^{-2}t_{obs,d}^{1/2} \tag{8}$$

$$\nu_a = 8.31 \times 10^{9}\ \text{Hz} \times (1+z)^{-2/5}\left(\frac{p-1}{3p+2}\right)^{3/5}\left[\epsilon_e g(p)\right]^{-1}\epsilon_B^{1/5}E_{52}^{-2/5}A_*^{6/5}t_{obs,d}^{-3/5} \tag{9}$$

$$f_{\nu,\max} = 76.9\text{mJy} \times (p+0.12)(1+z)^{3/2}\epsilon_B^{1/2}E_{52}A_*D_{L,28}^{-2}t_{\text{obs,d}}^{-1/2}, n_p \propto R^{-2} \tag{10}$$

Here $D_{L,28}$ is the luminosity distance of the Synchrotron source in the units of 10^{28} cm and

$$g(p) \simeq \begin{cases} \frac{p-2}{p-1}, & p > 2 \\ \ln^{-1}\left(\gamma_M/\gamma_m\right), & p = 2 \end{cases}$$

γ_M is the maximum Lorentz factor of the electrons. In our case p will be greater than 2. So these are the required tools to continue the search further.

1.3 Inversion of Spectral breaks

The spectral breaks are found in observations by modelling the spectrum of the observed data and their light curves and the equations 3 - 10 are inverted to get the shock wave parameters. I performed the inversion of the equations 3 - 10 myself and the Inverted equations are, (Keeping $p = 2.5$ for an example)

For a constant medium,

$$\epsilon_e = 0.0347\ (1+z) \left(\frac{\nu_a}{10^9}\right)^{5/6} \left(\frac{\nu_c}{10^{14}}\right)^{1/4} \left(\frac{\nu_m}{10^{13}}\right)^{11/12} D_{L,28}^{-1}\ f_{\nu,max}^{-1/2}\ t_d^{3/2} \tag{11}$$

$$\epsilon_B = 252.753\ (1+z)^{-3} \left(\frac{\nu_a}{10^9}\right)^{-5/2} \left(\frac{\nu_c}{10^{14}}\right)^{-5/4} \left(\frac{\nu_m}{10^{13}}\right)^{-5/4} D_{L,28}\ f_{\nu,max}^{1/2}\ t_d^{-5/2} \tag{12}$$

$$n = 2.591 \times 10^{-5}\ (1+z)^5 \left(\frac{\nu_a}{10^9}\right)^{25/6} \left(\frac{\nu_c}{10^{14}}\right)^{3/4} \left(\frac{\nu_m}{10^{13}}\right)^{25/12} D_{L,28}^{-3}\ f_{\nu,max}^{-3/2}\ t_d^{7/2} \tag{13}$$

$$E_{52} = 0.471\ (1+z)^{-2} \left(\frac{\nu_a}{10^9}\right)^{-5/6} \left(\frac{\nu_c}{10^{14}}\right)^{1/4} \left(\frac{\nu_m}{10^{13}}\right)^{-5/12} D_{L,28}^{3}\ f_{\nu,max}^{3/2}\ t_d^{-1/2} \tag{14}$$

For a wind-like medium,

$$\epsilon_e = 0.0517\ (1+z) \left(\frac{\nu_a}{10^9}\right)^{5/6} \left(\frac{\nu_c}{10^{14}}\right)^{1/4} \left(\frac{\nu_m}{10^{13}}\right)^{11/12} D_{L,28}^{-1}\ f_{\nu,max}^{-1/2}\ t_d^{3/2} \tag{15}$$

$$\epsilon_B = 74.438\ (1+z)^{-3} \left(\frac{\nu_a}{10^9}\right)^{-5/2} \left(\frac{\nu_c}{10^{14}}\right)^{-5/4} \left(\frac{\nu_m}{10^{13}}\right)^{-5/4} D_{L,28}\ f_{\nu,max}^{1/2}\ t_d^{-5/2} \tag{16}$$

$$A_* = 10.36 \times 10^{-4}\ (1+z) \left(\frac{\nu_a}{10^9}\right)^{5/3} \left(\frac{\nu_c}{10^{14}}\right)^{1/2} \left(\frac{\nu_m}{10^{13}}\right)^{5/6} t_d^{2} \tag{17}$$

$$E_{52} = 0.287\ (1+z)^{-2} \left(\frac{\nu_a}{10^9}\right)^{-5/6} \left(\frac{\nu_c}{10^{14}}\right)^{1/4} \left(\frac{\nu_m}{10^{13}}\right)^{-5/12} D_{L,28}^{3}\ f_{\nu,max}^{3/2}\ t_d^{-1/2} \tag{18}$$

These are the equations that will be employed to estimate External forward shock microphysics parameters given the data is sufficient. Theoretically it is possible to find all the parameters but in reality, the data is not always perfect and so the parameters are estimated with considerably high uncertainties.

1.4 Jet break

GRB afterglow light curve suffer a break after usually a day past the burst where the light curve steepens and decay index will become $2 \sim 3$ after the break. This is called the Jet break. The steepening of the light curve is due to two effects. First is time dilation, light from each point on the shock is beamed and due to that, only a fraction of the jet will be visible to the observer and initially, it would appear to be an expanding isotropic fireball but as time increases, the Lorentz factor decreases and the beaming angle of photon radiation increases with itself from which the opening angle of the jet increases and light curve steepens due to this. Secondly steepening occurs also due to the jet's sideways expansion, this adds to the relativistic beaming effect and produces a jet break after a particular time usually ~ 1 day. The spectrum and spectral breaks after the jet break has the following characteristics in the case of slow cooling, [Rho99] [KZ15]

$$\nu_m \propto t_{obs}^{-2} \tag{19}$$

$$\nu_c \propto t_{obs}^{0} \tag{20}$$

$$F_{\nu,\max} \propto t_{obs}^{-1} \tag{21}$$

$$f_\nu \propto \begin{cases} \nu^{1/3} t_{obs}^{-1/3}, & \nu_a < \nu < \nu_m \\ \nu^{-(p-1)/2} t_{obs}^{-p}, & \nu_m < \nu < \nu_c \\ \nu^{-p/2} t_{obs}^{-p}, & \nu > \nu_c \end{cases} \tag{22}$$

Both the wind and constant medium system follow the same broken power law and this also helps in determining p as well. The jet break time lets us estimate the jet equivalent kinetic energy of the blast wave and they can be estimated from the half opening angle determined from,

For constant medium,

$$\theta_j = 0.07\ rad\ \left(\frac{t_b}{1\ day}\right)^{3/8} \left(\frac{1+z}{2}\right)^{-3/8} \left(\frac{E_{iso}^K}{10^{53} ergs}\right)^{-1/8} \left(\frac{n}{0.1\ cm^{-3}}\right)^{1/8} \tag{23}$$

For wind medium,

$$\theta_j = 0.1\ rad\ \left(\frac{t_b}{1\ day}\right)^{1/4} \left(\frac{1+z}{2}\right)^{-1/4} \left(\frac{E_{iso}^K}{10^{53}\ ergs}\right)^{-1/4} \left(\frac{A_*}{1\ g/cm}\right)^{1/4} \tag{24}$$

2 Observation Data

X-ray observations were made by the X-Ray Telescope (XRT) onboard Swift, the X-ray light curve data were collected from the Swift light curve repository. Optical observations consists of Optical bands, Near Infrared and Ultraviolet bands and they were primarily observed by The Ultra-Violet and Optical Telescope (UVOT) on Swift, Magellan (Baade & Clay) and Gemini telescopes (North & South). Radio observation data were obtained from Karl G. Jansky

Very Large Array (VLA). These observation data (other than the X-ray light curves) were already compiled by [Fon+13] , So I made use of that to proceed with. The Galactic extinction was found to be $E(B-V)=0.02$ and Optical Extinction was found to be $A_V \approx 1\ mag$ [Fon+13].

3 Model Fitting

3.1 Light curves Fitting

Firstly, the parameter p that constructs the entire characteristics of a shock is estimated from fitting Radio, Optical and X-ray light curves to a broken power-law with a single break, then their temporal indices are used to find p and whether the data points are between (ν_m, ν_c) or (ν_a, ν_m) or beyond ν_c.

The fit was performed with χ^2 minimization and the best fit broken power law is found by looping over broken power laws with different temporal breaks and following χ^2 minimization. The optical light curve fit is performed with the r band optical data even though after $\sim$ 2 days old the data points behave dissimilarly to an External forward shock with no energy injection but the fluxes after 2 days come with a huge uncertainty such that some of the values range from any lower value to the recorded flux value. The X-ray light curve fit is performed using data which are after 9160s since the data before 9120s has already been fitted and the data after 9120s seems to require a break in the middle on itself. The optical light curve has been fitted with a broken power-law which has powers -1.084 ± 0.068 before the break and -2.76 ± 0.241 after the break and break is situated at 0.468 days with a $\chi^2/dof = 15.62/9$. The X-ray light curve has been fitted with a broken power law of powers -1.87 ± 0.309 and -2.44 ± 0.198 before and after the break respectively and the break situated at 0.494 days with a $\chi^2/dof = 12.65/9$. The radio light curve has been fitted with a single power-law since it lacked enough data to try fit models with breaks. The radio light curve has been fitted with a power-law of power -0.46 ± 0.137 with a $\chi^2/dof = 1.79/7$.

The temporal breaks found from fitting the optical and X-ray are close enough and so, it is a jet break since a jet break must be observed simultaneously in every band. The power of the fit after the jet break is the value of p from comparing to eq 22 and so from Optical fit, p is 2.76 ± 0.241 and from X-rays, p is 2.44 ± 0.198. The value of p is estimated to be between 2.44 and 2.76. We are not sure about where optical and X-ray frequencies are between as of now, but from the Radio light curve fit, observed radio bands lie between (ν_a, ν_m) since the temporal index is closer to 0.33 [eq 22]. The Light curves are shown in fig 2.

The Optical spectral index calculated considering a constant medium using the relation between temporal index and [eq 1] p is 0.95 when p = 2.91 and 0.62 when p = 2.24 if optical band frequencies were between (ν_m, ν_c) and if they are beyond ν_c, Optical spectral indices were 1.45 when p = 2.91 and 1.12 when p = 2.24. On the other side, the value of the optical spectral index found from the optical temporal index considering a constant medium is 0.72 ± 0.04 which is when compared with the above values tells us that the optical r band lies between (ν_m, ν_c). There was a disparity when the ambient density was considered to be wind-like and I wasn't able to support the data by considering a wind-like medium so a constant medium case is considered from now on. The X-ray spectral index calculated in a constant medium case from the first approach returns the same values as above and when compared with the value (1.58 ± 0.2) got from the X-ray temporal index, it tells that X-ray bands are beyond ν_c.

3.2 Obtaining SED

We are now certain of the placements of the observed bands in the spectrum, next is to obtain a SED from the observed data at a particular time after the burst trigger. The time chosen is 0.37 days, because it is the time that has observations from radio, optical and x-rays, and it is below the jet break time. Optical data at 0.37 days consists of data points (at t = 0.33, 0.34, 0.37) that are closer to 0.37 days, Radio data consists of data points at 0.37 days from burst trigger and x-ray data consist of data points which are near 0.37 (there was only one around 0.4 days). Since we have assumed that this system follows slow cooling, we fit the observed data using eq 1. The observed data is used to find the coefficients in eq 1 and obtain the SED from them. The coefficients are a function of $f_{\nu,max},\ \nu_a,\ \nu_m\ and\ \nu_c$ since they are all constants at a particular time. For example, the flux between (ν_a, ν_m) has the coefficient

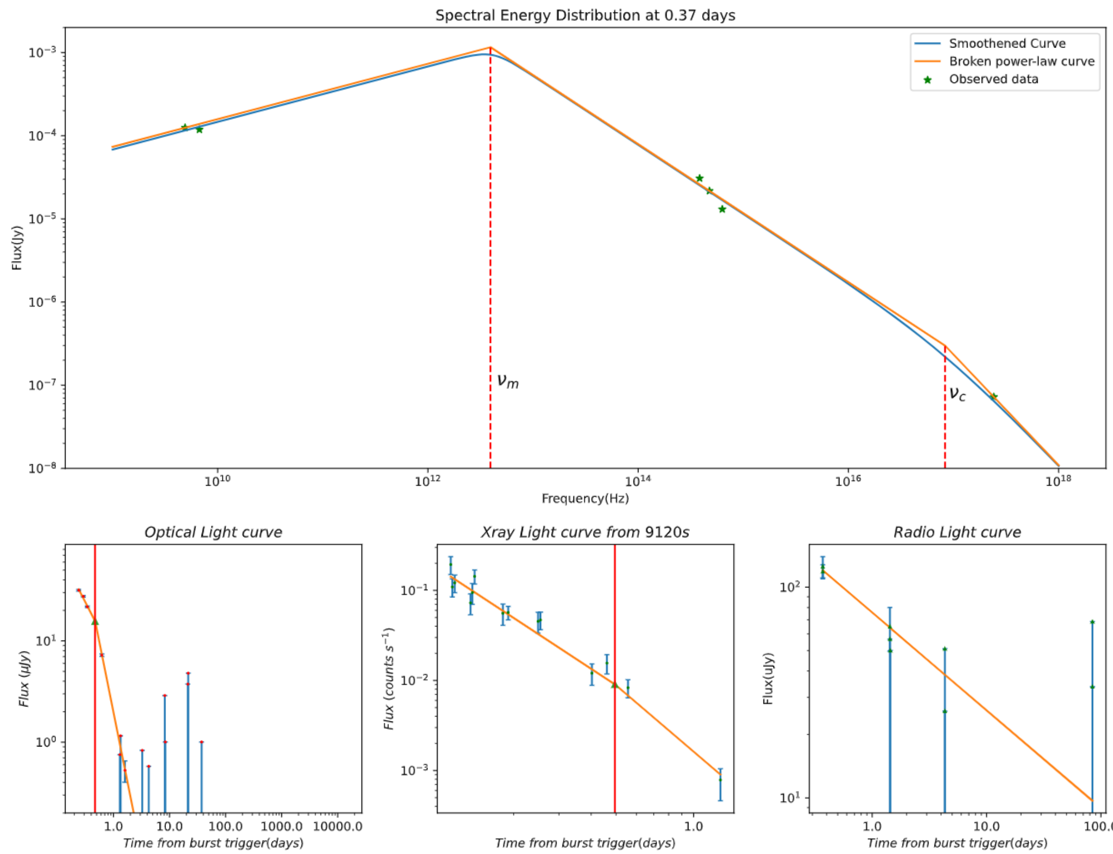

Figure 2: The figure above is the Spectral Energy Distribution at t = 0.37 days after the burst trigger for p = 2.66. The vertical lines represent the spectral breaks in the first figure. The figures below are the light curves of optical, x-rays and radio (from left). The vertical lines represent the jet break in the sub-figures. The stars represent the observed data points.

$f_{\nu,max}$ $\nu_m^{-1/3}$ and this is found from using the data points that lie between that respective interval of frequency. These are then used to fit broken power laws to the data points and they are smoothened using the Beuermann function [Beu+99]. Since it was not possible to find ν_a in this work, ν_a was obtained to be $6.8^{+1.13}_{-0.89} \times 10^8$ Hz at 0.37 days from [Fon+13].

4 Results

all the quantities that have been found so far will be put to use to estimate the External forward shock parameters by looping over different models that have different values of p, and the error bounds on the spectral breaks and $f_{\nu,max}$ are also considered in place. Estimation is performed using 11-14 considering a constant ambient medium. The model follows equipartition of ϵ_e, ϵ_B

i.e $\epsilon_e, \epsilon_B < 1/3$. The constraint on p is from 2.24 to 2.91 and on t_j (jet break time) is 0.468 to 0.494.

After running through different possible values of p and t_j, the shock parameter values and other quantities have been recorded for each of those combinations. From the iteration of different model, the possible values and the median value of the parameters and the quantities is calculated. The estimated possible values are given by

$$9.15 \times 10^{-2} < \epsilon_e < 0.33 \tag{25}$$

$$1.38 \times 10^{-3} < \epsilon_B < 0.33 \tag{26}$$

$$4.2 \times 10^{51} \ ergs < E^K_{iso} < 14.4 \times 10^{51} \ ergs \tag{27}$$

$$2.79 \times 10^{-2} \ cm^{-3} < n < 2.24 \ cm^{-3} \tag{28}$$

$$4.21^\circ < \theta_j < 6.89^\circ \tag{29}$$

$$1.4 \times 10^{49} \ ergs < E^K_{jet} < 8.19 \times 10^{49} \ ergs \tag{30}$$

The median value of $\epsilon_e,\ \epsilon_B,\ E^K_{iso},\ n,\ \theta_j,\ E^K_{jet}$ are $0.148,\ 0.024,\ 8.07 \times 10^{51}$ ergs, 0.22 cm^{-3}, $5.33^\circ,\ 3.4 \times 10^{49}\ ergs$.

5 Acknowledgement

The main reference to this paper is [KZ15]. All of the parts that have not been referenced are from this paper. This was supplemented by my mentor Dr. Shashi Bhushan Pandey and a special thanks to my mentor for introducing me to some of these aspects and supplementing with necessary equipments. I have made use of the article [Fon+13] for the data sets. This work has been carried out independently and with my own efforts and not in any way trying to replicate the results on the above paper.

6 Code snippets

All the codes have been written in python and the imports that will be required are mentioned as well. The snippets are described using python comments in each of them.

6.1 Obtaining SED

Imports required

```
from tools import *
from math import atan, pi
import numpy as np
```

Snippet

```
def Create_model(obs_freqs, obs_fluxs, nums, p):
    ## Reference : THE SHAPE OF SPECTRAL BREAKS IN GRB AFTERGLOWS,
    ## JONATHAN GRANOT AND RE'EM SARI, 2008 [GS02]
    ## Beuermann sharpness parameters (around vm and vc respectively)
    ## vm, vc - synchrotron injection frequency and synchrotron ccoling frequency
    n1 = 3*1.76-0.38*p
    n2 = 3*0.80-0.03*p
    freq = 10**np.linspace(9, 18, 10**4-len(obs_freqs))
    freq = np.sort(list(set(np.append(freq, obs_freqs))))

    ## Fluxes constructed using the observed data points between va and vm, vm and vc,
    ## and beyond vc respectively
    leftcoeff = (obs_fluxs[0]*obs_freqs[0]**(-1/3) + obs_fluxs[1]
                *obs_freqs[1]**(-1/3))/2
    midcoeff = (obs_fluxs[2]*obs_freqs[2]**((p-1)/2) + obs_fluxs[3]
               *obs_freqs[3]**((p-1)/2) + obs_fluxs[4]*obs_freqs[4]**((p-1)/2))/3

    leftvm = freq**(1/3)*leftcoeff
    midvm = freq**(-(p-1)/2)*midcoeff
    rightvm = freq**(-p/2)*obs_fluxs[-1]*obs_freqs[-1]**(p/2)

    ## this is to find the intersection of the fluxes between va and vm, vm and vc,
    ## and beyond vc respectively, they correspond to vm and vc
    l1 = line(np.log10([freq[0], leftvm[0]]),  np.log10([freq[-1], leftvm[-1]]))
    l2 = line(np.log10([freq[0], midvm[0]]),   np.log10([freq[-1], midvm[-1]]))
    l3 = line(np.log10([freq[0], rightvm[0]]), np.log10([freq[-1], rightvm[-1]]))
    vm, fm = intersection(l1, l2)
    vc, fc = intersection(l2, l3)
    vm, vc, fm, fc = 10**vm, 10**vc, 10**fm, 10**fc

    ## Smoothens the curve using Beuermann function
    F = fm*( (freq/vm)**(-n1/3) + (freq/vm)**(-n1*(1-p)/2) )**(-1/n1)
    Ftilda = ( 1 + (freq/vc)**(n2*((1-p)/2-(-p/2))) )**(-1/n2)
```

```
    ## Spectral breaks
    spectral_breaks = np.array([[np.array([vm]), np.array([fm])],[np.array([vc]),
                  np.array([fc])]], dtype=object)

    return freq, F*Ftilda, spectral_breaks
```

6.2 LC curve fitting

Imports required

```
import numpy as np
from scipy.optimize import curve_fit
from Model_Fitting import *
from tools import *
```

Snippet

```
## Function to be fitted with
def func(x, a, b):
    return a*(x**b)

## Fits the above function to the datapoints
def fit_curve(obs_times, obs_fluxes, obs_err, break_est, print_err = False):
    fitted_chisq = np.inf

    for br in break_est:
        ## this try and except is for occurence of any errors since I got some errors
        ## while making this
        try:
            ## bef_br and aft_br are the parameters describing the curve constructed
            bef_br, pcov1 = curve_fit(func, obs_times[obs_times<=br],
                            obs_fluxes[obs_times<=br], sigma = obs_err[obs_times<=br],
                            absolute_sigma=True , maxfev = 20000)
            aft_br, pcov2 = curve_fit(func, obs_times[obs_times>=br],
                            obs_fluxes[obs_times>=br], sigma = obs_err[obs_times>=br],
                            absolute_sigma=True , maxfev = 20000)

            times = np.sort(np.append(np.linspace(obs_times[0], obs_times[-1], 10**4),
                    obs_times))

            ## fluxes before and after the cornered break
            p1 = func(times, *bef_br)
            p2 = func(times, *aft_br)

            ## Intersects the two fluxes to find the break point
            l1 = line(np.log10([times[0], p1[0]]), np.log10([times[-1], p1[-1]]))
            l2 = line(np.log10([times[0], p2[0]]), np.log10([times[-1], p2[-1]]))
            int_point = intersection(l1, l2)

            ## Constructs the lightcurve with a break includes the break point in them
            pred_fluxes = np.array(list(p1[times <= (10**int_point[0])]) +
```

```python
                    [10**int_point[1]] + list(p2[times >= (10**int_point[0])]))
            times1 = np.array(list(times[times <= (10**int_point[0])]) +
                     [10**int_point[0]] + list(times[times >= (10**int_point[0])]))

            chisq = np.sum(((np.array(list(func(obs_times[obs_times<=br], *bef_br))+
                    list(func(obs_times[obs_times>=br], *aft_br))) - obs_fluxes)
                    /(obs_err))**2)
            pval = 1-chi2.cdf(chisq, len(obs_times)-5)

            ## stores the required values to compare and return the best fit values
            if (len(obs_times[obs_times<=10**int_point[0]]) > 1 and
                  len(obs_times[obs_times>=10**int_point[0]]) > 1):
                if chisq < fitted_chisq:
                    fitted_coeff = np.array([bef_br, aft_br])
                    fitted_br = [10**int_point[0], 10**int_point[1]]
                    fitted_fluxes = pred_fluxes
                    fitted_chisq = chisq
                    fitted_coeff_err = [np.sqrt(np.diag(pcov1)),
                                        np.sqrt(np.diag(pcov2))]
                    fitted_pval = pval

        except Exception as e:
            if print_err:
                print(e)
            else:
                break

    return (times1, fitted_fluxes ,fitted_coeff, fitted_br, fitted_chisq,
            fitted_pval, fitted_coeff_err)
```

6.3 Parameter Estimation

Imports required

```python
import numpy as np
from tools import *
from Inverted_equations import *
from tqdm import tqdm
from  Model_Fitting import Create_model
```

Snippet

```python
## ee, eB, E52, theta, E52_jet represents fraction of energy in the electron and
## the magnetic field in the shock wave, the isotropic kinetic energy of the burst,
## the half opening angle and the jet equivalent kinetic energy of the burst.
## 0 near variables represents computations in constant medium.
## ps, tbds, vas are the ranges of electron distribution power-law, break time and
## synchrotron self absorption frequency.
## DL28 is the luminosity distance in terms of 10^28 cm
## td is the time of observation in days
ps = np.linspace(2.444-0.198, 2.766+0.245, 50)
```

```python
tbds = np.linspace(0.468, 0.494, 50)
vas = [5.91e8, 6.8e8, 7.93e8]
DL28 = 0.58
z = 0.356
td = 0.37

vms, vcs = {}, {}

ee_0 = {}
eB_0 = {}
n_0 = {}
E52_0 = {}
theta_0 = {}
E52jet_0 = {}
va_range0 = {}

## For printing a progress bar
pbar = tqdm(range(len(ps)*len(vas)*3), desc = 'Loop Progress', position = 0)

## First loop loops the error bounded on the vm and vc
for i in range(3):
    ## pol_err is to add the error to fluxes and err_type is for saving the values
    ## accordingly
    if i == 0:
        pol_err = 0
        err_type = 'zero_err'
    elif i == 1:
        pol_err = -errs[0]
        err_type = 'neg_err'
    elif i == 2:
        pol_err = errs[1]
        err_type = 'pos_err'

    ee_0[err_type] = {}
    eB_0[err_type] = {}
    n_0[err_type] = {}
    E52_0[err_type] = {}
    theta_0[err_type] = {}
    E52jet_0[err_type] = {}
    va_range0[err_type] = {}

    vms[err_type] = {}
    vcs[err_type] = {}

    ## this loop is to run through different values of p
    for p in ps:
        _, __, spec_breaks = Create_model(freqs, fluxs + pol_err, 10**5, p)
        vms[err_type][f'{p}'] = list(spec_breaks[0])
        vcs[err_type][f'{p}'] = list(spec_breaks[1])

        vm = spec_breaks[0][0]
        vc = spec_breaks[1][0]
        fmmax = spec_breaks[0][1]*1000
```

```
        ee_0[err_type][f'{p}'] = {}
        eB_0[err_type][f'{p}'] = {}
        n_0[err_type][f'{p}'] = {}
        E52_0[err_type][f'{p}'] = {}
        theta_0[err_type][f'{p}'] = {}
        E52jet_0[err_type][f'{p}'] = {}
        va_range0[err_type][f'{p}'] = {}

        ## this loop is to run through different values of va
        for va in vas:
            ee0temp = ee0(p, z, td, DL28, va, vm, vc, fmmax)[0]
            eB0temp = eB0(p, z, td, DL28, va, vm, vc, fmmax)[0]

            ## the consition is given to include the equipartition of epsilon_e
            ## and epsilon_B
            if ee0temp < 0.33 and eB0temp < 0.33:
                ee_0[err_type][f'{p}'][f'{va}'] = ee0temp
                eB_0[err_type][f'{p}'][f'{va}'] = eB0temp
                n_0[err_type][f'{p}'][f'{va}'] = n0(p, z, td, DL28, va, vm, vc,
                                                    fmmax)[0]
                E52_0[err_type][f'{p}'][f'{va}'] = E520(p, z, td, DL28, va, vm, vc,
                                                        fmmax)[0]
                theta_0[err_type][f'{p}'][f'{va}'] = {}
                E52jet_0[err_type][f'{p}'][f'{va}'] = {}
                for tbd in tbds:
                    theta_0[err_type][f'{p}'][f'{va}'][f'{tbd}'] = theta0(z, tbd,
                                E52_0[err_type][f'{p}'][f'{va}'],
                                n_0[err_type][f'{p}'][f'{va}'])
                    E52jet_0[err_type][f'{p}'][f'{va}'][f'{tbd}'] = EK_to_Ejet(
                                theta_0[err_type][f'{p}'][f'{va}'][f'{tbd}'],
                                E52_0[err_type][f'{p}'][f'{va}'])
                va_range0[err_type][f'{p}'][f'{va}'] = va

            pbar.update(1)
```

6.4 Inverted Equations

Imports required

```
from math import pi, sqrt, exp, cos
```

Snippet

```
def E520(p, z, td, DL28, va, vm, vc, fvmmax):
    return (7.60403416292678e8*DL28**3*fvmmax**1.5*(((-2+p)**2*(-0.67+p))
           /(-1+p)**2)**0.4166666666666667*(((-1+p)*(6-p+p**2)**0.6)
           /(-2+p))**0.8333333333333334*vc**0.25)/(((-0.46+p)/exp(1.16*p))**0.25*
           va**0.8333333333333334*vm**0.4166666666666667*((0.14+p)*(1+z))**1.5
           *sqrt(td*(1+z)))

def ee0(p, z, td, DL28, va, vm, vc, fvmmax):
```

```
    return (1.5686351147608723e-25*sqrt(0.14+p)*td**1.5*va**0.8333333333333334
            *vc**0.25*vm**0.9166666666666666*(1+z))/(DL28*sqrt(fvmmax)*(((-2+p)**2*
            (-0.67+p))/(-1+p)**2)**0.9166666666666666*((-0.46+p)/exp(1.16*p))**0.25
            *(((-1+p)*((-1+p)/(2+3*p))**0.6)/(-2+p))**0.8333333333333334)

def eB0(p, z, td, DL28, va, vm, vc, fvmmax):
    return (9.013426442087402e58*sqrt(fvmmax)*(((-2+p)**2*(-0.67+p))/(-1+p)**2)**1.25*
            ((-0.46+p)/exp(1.16*p))**1.25*(((-1+p)*((-1+p)/(2+3*p))**0.6)/((-2+p)*td*
             (1+z)))**2.5)/(va**2.5*vc**1.25*vm**1.25*sqrt(((0.14+p)*(1+z))/DL28**2))

def n0(p, z, td, DL28, va, vm, vc, fvmmax):
    return (6.486377353377558e-80*va**4.166666666666667*vc**0.75*vm**2.0833333333333335*
            ((0.14+p)*(1+z))**1.5*(td*(1+z))**3.5)/(DL28**3*fvmmax**1.5*
            (((-2+p)**2*(-0.67+p))/(-1+p)**2)**2.0833333333333335*
            ((-0.46+p)/exp(1.16*p))**0.75*(((-1+p)*((-1+p)/(2+3*p))**0.6)/
            (-2+p))**4.166666666666667)

def theta0(z, tbd, E52, n): ## tbd in days, E52 in ergs, n in cm^-3
    return 180/pi*0.07*(tbd)**(3/8)*((1+z)/2)**(-3/8)*(E52/10)**(-1/8)*(n/0.1)**(1/8)

def theta2(z, tbd, E52, As): ## tbd in days, E52 in ergs, As in g cm^-1
    return 180/pi*0.16*(tbd)**(1/4)*((1+z)/2)**(-1/4)*(E52/10)**(-1/4)*(As/1)**(1/4)

def EK_to_Ejet(theta, E52): ## theta in degrees
    return (1-cos(theta*pi/180))*E52
```

These are the equations used above in the first four functions,

$$\mathrm{ee} \to \frac{1.56864\times10^{-25}\,\sqrt{0.14+p}\;\mathrm{td}^{3/2}\,\mathrm{va}^{5/6}\,\mathrm{vc}^{1/4}\,\mathrm{vm}^{11/12}\,(1.+z)}{\mathrm{DL28}\,\sqrt{\mathrm{fmax}}\left(\frac{(-2.+p)^2\,(-0.67+p)}{(-1.+p)^2}\right)^{11/12}\left(e^{-1.16\,p}\,(-0.46+p)\right)^{1/4}\left(\frac{(-1.+p)\left(\frac{-1.+p}{2.+3.\,p}\right)^{3/5}}{-2.+p}\right)^{5/6}}$$

$$\mathrm{eB} \to \frac{9.01343\times10^{58}\,\sqrt{\mathrm{fmax}}\left(\frac{(-2.+p)^2\,(-0.67+p)}{(-1.+p)^2}\right)^{5/4}\left(e^{-1.16\,p}\,(-0.46+p)\right)^{5/4}\left(\frac{(-1.+p)\left(\frac{-1.+p}{2.+3.\,p}\right)^{3/5}}{(-2.+p)\,\mathrm{td}\,(1.+z)}\right)^{5/2}}{\mathrm{va}^{5/2}\,\mathrm{vc}^{5/4}\,\mathrm{vm}^{5/4}\,\sqrt{\frac{(0.14+p)\,(1.+z)}{\mathrm{DL28}^2}}}$$

$$\mathrm{n} \to \frac{6.48638\times10^{-80}\,\mathrm{va}^{25/6}\,\mathrm{vc}^{3/4}\,\mathrm{vm}^{25/12}\,\left((0.14+p)\,(1.+z)\right)^{3/2}\,\left(\mathrm{td}\,(1.+z)\right)^{7/2}}{\mathrm{DL28}^3\,\mathrm{fmax}^{3/2}\left(\frac{(-2.+p)^2\,(-0.67+p)}{(-1.+p)^2}\right)^{25/12}\left(e^{-1.16\,p}\,(-0.46+p)\right)^{3/4}\left(\frac{(-1.+p)\left(\frac{-1.+p}{2.+3.\,p}\right)^{3/5}}{-2.+p}\right)^{25/6}}$$

$$\mathrm{E52} \to \frac{1.31706\times10^{9}\,\mathrm{DL28}^3\,\mathrm{fmax}^{3/2}\left(\frac{(-2.+p)^2\,(-0.67+p)}{(-1.+p)^2}\right)^{5/12}\left(\frac{(-1.+p)\left(\frac{-1.+p}{2.+3.\,p}\right)^{3/5}}{-2.+p}\right)^{5/6}\mathrm{vc}^{1/4}}{\left(e^{-1.16\,p}\,(-0.46+p)\right)^{1/4}\,\mathrm{va}^{5/6}\,\mathrm{vm}^{5/12}\,\left((0.14+p)\,(1.+z)\right)^{3/2}\,\sqrt{\mathrm{td}\,(1.+z)}}$$

6.5 tools used

Imports required

```
from math import cos, sin, radians
import numpy as np
```

Snippet

```
def line(p1, p2):
    ## Computes the three coefficients of a line equation in the form Ax + By = C
    ## the inputs must be two 2d points
    A = (p1[1] - p2[1])
    B = (p2[0] - p1[0])
    C = (p2[0]*p1[1] - p1[0]*p2[1])
    return A, B, C

def intersection(L1, L2):
    ## Finds the intersection between two lines
    ## the inputs must be two line coefficients
    D  = L1[0] * L2[1] - L1[1] * L2[0]
    Dx = L1[2] * L2[1] - L1[1] * L2[2]
    Dy = L1[0] * L2[2] - L1[2] * L2[0]
    if D != 0:
        px = Dx / D
        py = Dy / D
        return px,py
    else:
        print('No Intersection')
        return False

def rotate(point, origin, angle):
    ## Rotates a point counterclockwise by a given angle around a given origin.
    ## The angle should be given in degrees
    ox, oy = origin
    px, py = point
    angle = radians(angle)

    qx = ox + cos(angle) * (px - ox) - sin(angle) * (py - oy)
    qy = oy + sin(angle) * (px - ox) + cos(angle) * (py - oy)
    return np.array(qx), np.array(qy)

def rotate_a_curve(x, y, angle):
    ## Rotates a curve by a given angle around a given origin
    ## The angle should be given in degrees.
    rot_x = []
    rot_y = []
    for i,j in zip(x, y):
        frei, fluxi = rotate([i, j], [x[-1], y[-1]], angle)
        rot_x += [frei]
        rot_y += [fluxi]

    return np.array(rot_x), np.array(rot_y)
```

References

[SPN98] Re'em Sari, Tsvi Piran, and Ramesh Narayan. "Spectra and Light Curves of Gamma-Ray Burst Afterglows". In: *The Astrophysical Journal* 497.1 (1998), pp. L17–L20. DOI: 10.1086/311269.

[Beu+99] K. Beuermann et al. "VLT observations of GRB 990510 and its environment". In: *Astronomy & Astrophysics* 352 (1999), pp. L26–L30.

[Rho99] James E. Rhoads. "The Dynamics and Light Curves of Beamed Gamma-Ray Burst Afterglows". In: *The Astrophysical Journal* 525.2 (1999), pp. 737–749. DOI: 10.1086/307907.

[GS02] Jonathan Granot and Re'em Sari. "The Shape of Spectral Breaks in Gamma-Ray Burst Afterglows". In: *The Astrophysical Journal* 568.2 (2002), pp. 820–829. DOI: 10.1086/338966.

[Fon+13] W. Fong et al. "SHORT GRB 130603B: DISCOVERY OF A JET BREAK IN THE OPTICAL AND RADIO AFTERGLOWS, AND A MYSTERIOUS LATE-TIME X-RAY EXCESS". In: *The Astrophysical Journal* 780.2 (2013), p. 118. DOI: 10.1088/0004-637x/780/2/118.

[KZ15] Pawan Kumar and Bing Zhang. "The physics of gamma-ray bursts & relativistic jets". In: *Physics Reports* 561 (2015), pp. 1–109. DOI: 10.1016/j.physrep.2014.09.008.