

Poszukiwanie mikrosoczewek grawitacyjnych

Adam Gonstal, Kamil Kolasa, Rafał Kornel, Konrad Maliszewski, Anna Olechowska

19 lutego 2020

Abstrakt

Poniżej opisany projekt studencki polegał na analizie fragmentu danych z projektu OGLE III w celu znalezienia zjawisk mikrosoczewkowania grawitacyjnego. Autorom zostały udostępnione dane z teleskopu w Las Campanas w Chile, dotyczące m.in. pomiarów jasności dla ok. 260 tysięcy gwiazd, zbieranych na przestrzeni ok. 6 lat. W ramach projektu utworzony został algorytm analizujący dane dla każdej gwiazdy i zwracający wykresy zależności jasności od czasu dla tych gwiazd (krzywe blasku), które według algorytmu mogły dawać efekt soczewki. Około 27% (24 na 88) zwróconych krzywych blasku zostało przez autorów pracy zakwalifikowanych jako potencjalne rzeczywiste przypadki mikrosoczewkowania grawitacyjnego.

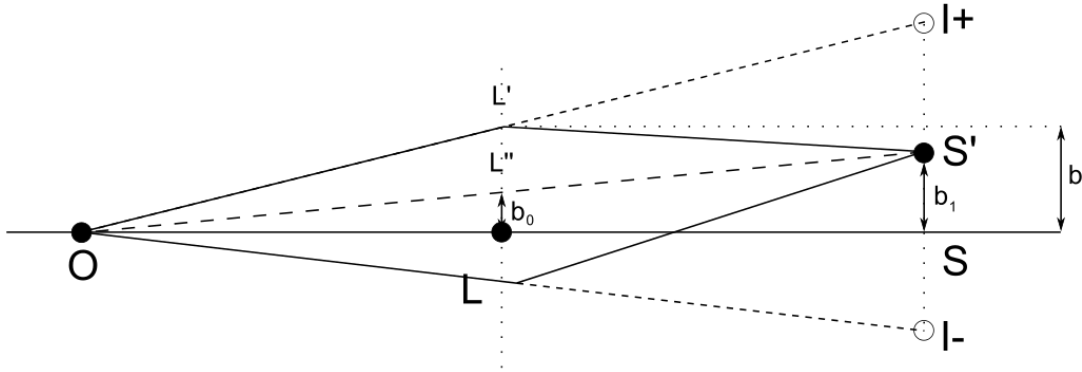
Spis treści

1	Wstęp	3
1.1	Wstęp teoretyczny	3
1.2	Projekt OGLE	4
2	Analiza danych	4
2.1	Dane	4
2.2	Struktura programu	5
2.3	Odsiew szumu	5
2.4	Format danych - klasa 'Curve'	5
2.5	Opis algorytmu	6
2.6	Napotkane trudności	6
3	Rezultaty	7
3.1	Znalezione soczewki	7
3.2	Wizualizacje	7
4	Bibliografia	9

1 Wstęp

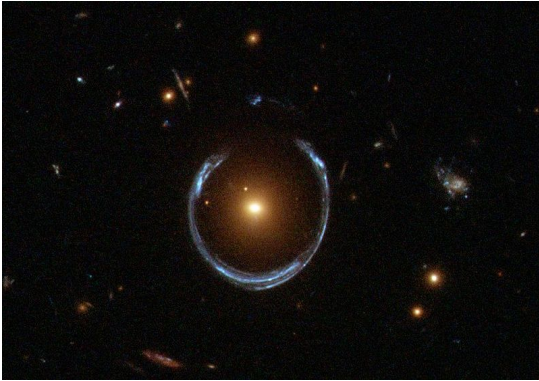
1.1 Wstęp teoretyczny

Zjawisko soczewkowania grawitacyjnego wynika z zakrzywienia czasoprzestrzeni przez masy się w niej znajdujące. Konsekwencją tego jest poruszanie się promieni świetlnych po zakrzywionych torach, tj. najkrótszych możliwych, w przestrzeni Mińkowskiego. W związku z tym, w sytuacji gdy w okolicach linii łączącej źródło światła (np. galaktykę) z obserwatorem znajdzie się odpowiednio duża masa, światło będzie omijać taką masę, co przedstawia Rysunek 1.

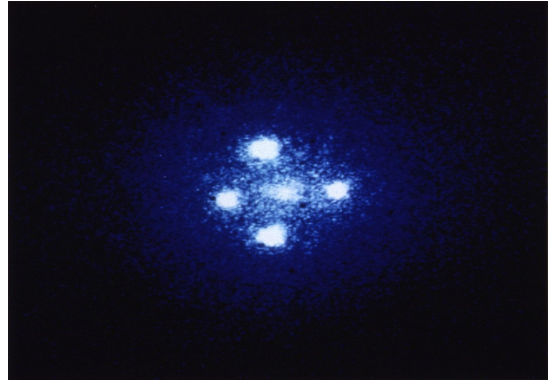


Rysunek 1: Przykład zjawiska soczewkowania, gdzie źródło S' jest obserwowane jako dwa obrazy $I+$ i $I-$ MesserWoland (2006).

Obraz źródła widziany przez obserwatora może ulec różnym deformacjom, tj. rozciągnięciu, przemieszczeniu, kilkukrotnemu odbiciu, a także wzmocnieniu, co w przypadku tego projektu jest najistotniejszym aspektem soczewkowania. Przykłady takiego zjawiska przedstawiają Rysunki 2a i 2b.



(a) Galaktyka LRG 3-757 soczewkująca obraz galaktyki znajdującej się za nią ESA/Hubble and NASA (2011)



(b) Kwazar Q2237+030 soczewkowany przez galaktykę ZW 2237+030, tzw. krzyż Einsteina NASA and STScI (1990)

Rysunek 2: Przykłady deformacji obrazu spowodowane soczewkowaniem grawitacyjnym

W szczególnych przypadkach, gdy masa soczewkująca jest stosunkowo nieduża, a jej tor ruchu przecina bądź jest bardzo bliski torowi promieni świetlnych od źródła do obserwatora, efekty deformacji obrazu mają zbyt małe rozmiary kątowe, by udało się je zaobserwować z

Ziemi. W takich sytuacjach jedyną obserwowalną konsekwencją zajścia soczewki jest wzmocnienie jasności. Ten specyficzny rodzaj soczewkowania nazywany jest mikrosoczewkowaniem grawitacyjnym. Przykładem jego może być obiekt z pobliskiej galaktyki wysyłający ku Ziemi promieniowanie elektromagnetyczne, na którego drodze znajduje się masywna planeta. Wzmocnienie można wyrazić jako wielkość μ , będącą ilorazem strumienia światła bez wzmocnienia oraz z wzmocnieniem. Ponieważ natężenie światła I jest stałe w czasie, będzie to wyłącznie iloraz kątów brylowych, z których światło dociera do obserwatora.

$$\mu = \frac{Id\Omega}{Id\Omega_0} = \frac{d\Omega}{d\Omega_0} \quad (1)$$

Wzmocnienie można również dobrze opisać za pomocą odległości u źródła światła od soczewki, którą można przedstawić za pomocą kilku parametrów, które dla danej soczewki można przyjąć jako stałe w trakcie trwania zjawiska. Wspomniane parametry geometryczne mają wpływ na t_E , tj. czas Einsteina. Wielkość b jest wielkością analogiczną do parametru zderzenia i także jest stała. Czas t_0 jest momentem największego wzmocnienia, z kolei jedyną zmienną we wzorze 2 jest czas t .

$$u(t) = \sqrt{\left(\frac{t - t_0}{t_E}\right)^2 + b^2} \quad (2)$$

Znając już zależność $u(t)$ można powiązać ją ze wspomnianym wcześniej wzmocnieniem μ , tj. wyprowadzić wzór 3, zwany także krzywą Paczyńskiego. Przykładową krzywą Paczyńskiego przedstawia Rysunek 3. W skali sześciu lat mikrosoczewka trwająca ok. 70 dni widocznie wyróżnia się skokiem jasności w trakcie trwania zjawiska, co było punktem wyjściowym przy konstrukcji algorytmu i zostanie opisane dokładniej w kolejnych rozdziałach.

$$\mu(u) = \frac{u^2 + 2}{u\sqrt{u^2 + 4}} \quad (3)$$

1.2 Projekt OGLE

Projekt OGLE tj. „Optical Gravitational Lensing Experiment” jest projektem naukowym prowadzonym w obserwatorium Las Campanas w Chile, przez Obserwatorium Astronomiczne Uniwersytetu Warszawskiego, w ramach którego m.in. wykonywane są pomiary jasności gwiazd, mające na celu poszukiwanie zjawisk mikrosoczewkowania.

Projekt kierowany jest przez prof. Andrzeja Udalskiego, a jego zasadniczym obszarem obserwacji są rejony w pobliżu centrum naszej galaktyki. Od początku istnienia projektu (kwiecień 1992 roku) zaobserwowano: 20 nowych planet pozasłonecznych, ponad 4000 zjawisk mikrosoczewkowania oraz kilkaset tysięcy nowych gwiazd zmiennych¹. Projekt jest aktualnie (od marca 2010 roku) w czwartej fazie realizacji. Poprzednie fazy miały miejsce w latach: 1992-1995 (OGLE-I), 1997-2001 (OGLE-II), 2001-2009 (OGLE-III) i wiązały się ze stosowaniem innych detektorów.

2 Analiza danych

2.1 Dane

Dane na których oparty jest nasz projekt studencki pochodzą z projektu OGLE-III i zawierają 280.000 plików tekstowych, w których znajduje się około 2400 pomiarów jasności

¹Dane ze strony internetowej projektu: <http://www.astrouw.edu.pl/index.php/ogle-artykul>

zebranych na przestrzeni 6-ciu lat. Każdy pomiar był reprezentowany przez linijkę tekstu, zawierał datę pomiaru, jasność gwiazdy oraz błąd pomiarowy. Warto wspomnieć, że jakość pomiarów jest daleka do ideału i musieliśmy przefiltrować dane, najpierw odrzucając błędy grube. Wśród nich były pomiary o jasności 99 mag, co nie jest fizyczne i było spowodowane najprawdopodobniej sposobem obróbki zdjęć zebranych przez sensor ccd.

2.2 Struktura programu

Do analizy danych napisaliśmy program w języku Python. Dokładny kod znajduje się na platformie github <https://github.com/wolf3213/Projekt-Astrolensing>. Składa się on z kilku części:

1. „Main”, czyli część główna programu, w której podawane były nazwy katalogów, służył do uruchamiania programu.
2. „Predictor”, czyli plik zawierający klasę odpowiedzialną za działanie algorytmu.
3. „Parser”, czyli plik zawierający klasę służącą do parsowania plików wejściowy do obiektów na których potem operował algorytm.
4. „Curve”, czyli plik zawierający klasę odpowiedzialną za przechowywanie danych, tworzenie wykresów i wstępne odrzucanie szumów.

W programie wykorzystaliśmy następujące biblioteki:

1. Numpy - do przechowywania pomiarów,
2. Matplotlib - modułu pyplot do rysowania pomiarów oraz dopasowanych krzywych,
3. Scipy - funkcji curve_fit z modułu optimize, do dopasowania parametrów krzywej Paczyńskiego.

2.3 Odsiew szumu

Właściwa część programu zaczynała się dopiero po wczytaniu danych. Wstępnie odfiltrowywał on wszystkie potencjalnie nienadające się do obróbki dane. Obejmowało to dane, które:

1. Zawierały pomiary jasności o wartości większej/równej 99 magnitudo
2. Zostały wykonane między 1. a 2137., oraz 2450. i 2500. dniem (właściwie indeksem daty) trwania projektu OGLE. Musieliśmy je usunąć, ponieważ zawierały zbyt dużą ilość błędnych pomiarów, za bardzo odstających od średniej jasności. Usunięcie kilkudziesięciu pomiarów nie wpłynęło negatywnie na analizę danych, oraz nie zmniejszyło informacji zawartych w pomiarach.

2.4 Format danych - klasa 'Curve'

Wczytane i przefiltrowane dane były przez nas przechowywane jako obiekty klasy 'Curve'. Zawierała ona pola, takie jak:

1. times - listę z biblioteki numpy (np.array) zawierającą daty pomiarów, uporządkowaną rosnąco,

2. mags - listę z biblioteki numpy zawierającą wartości jasności gwiazdy,
3. errors - listę z biblioteki numpy zawierającą błędy pomiarowe,

oraz kilka innych, mniej znaczących pól (np. mag_mean - średnia jasność gwiazdy, mag_std - odchylenie standardowe jasności). Obiekty posiadały również kilka przydatnych funkcji, ułatwiających obróbkę danych (i nie tylko), m.in:

1. update_data - metoda służąca do przypisania obiektowi nowych danych (np. pozbawionych szumu) oraz obliczenia jeszcze raz wartości średnich, odchyleń standardowych itp.
2. cut_points - metoda usuwająca pomiary z zadanych przedziałów czasowych,
3. plot - metoda odpowiedzialna za rysowanie wykresów krzywych,
4. fit - metoda obliczająca parametry najlepszego dopasowania krzywej Paczyńskiego.

2.5 Opis algorytmu

Program, operując już na nieodrzuconych danych, oblicza dla pomiarów jasności danej gwiazdy: odchylenie standardowe σ_{mag} i średnią m_0 . Następnie sprawdza, które pomiary jasności m spełniają nierówność

$$m < \sigma_{mag} \cdot A + m_0 \quad (4)$$

to znaczy są jaśniejsze niż średnia plus pewna wielokrotność σ_{mag} i zlicza je, przy czym A jest arbitralnie ustaloną dodatnią liczbą. Następnie dla punktów spełniających nierówność, liczy odchylenie standardowe po czasie σ_t , oraz średni czas t tych punktów.

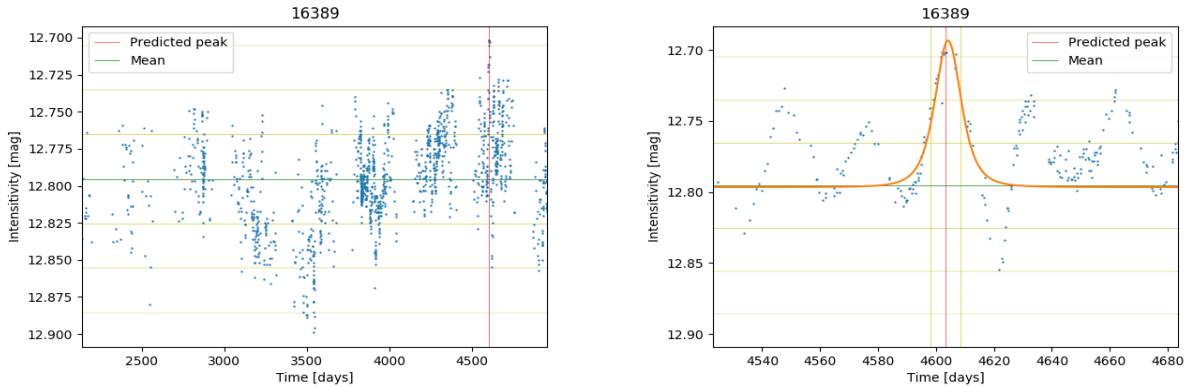
Wartość A jest skorelowana z głębokością soczewek, których szukamy. Im większa wartość A , tym głębsze soczewki program będzie znajdował, w drugą stronę, dla małych wartości A program powinien wykryć soczewki płytkie. Niestety w rzeczywistości płytkie soczewki często nie były wykrywane, ponieważ były głębokości szumu. W rezultacie udało nam się znaleźć jedynie głębsze soczewki.

Program determinował czy krzywa jest soczewką na podstawie wielkości σ_t . Jest to szerokość rozkładu punktów odrzuconych - spełniających nierówność (4) wokół wyliczonej średniej, która rozumiana może być jako moment największego wyjaśnienia soczewki. Ustalając górną granicę szerokości rozkładu odrzuconych punktów kierowaliśmy się założeniem, że soczewki które nas interesują, tzn. te głębokie, trwają zazwyczaj bardzo krótko, często nie dłużej niż kilkadziesiąt dni. Jeśli więc algorytm wykrył, że punkty pomiarowe spełniające nierówność (4) są położone blisko średniej ich czasu, zwracał krzywą jako podejrzaną o soczewkowanie. Aby zrozumieć dlaczego algorytm działa, trzeba uzmysłowić sobie fakt, że dla krzywej bez soczewki, punkty spełniające (4) to pomiary zaszumione, a te są statystycznie równo rozłożone po całej krzywej. Zatem dla krzywej bez soczewki wartość σ_t powinna być duża.

2.6 Napotkane trudności

Algorytm przedstawiony przez nasz zespół nie był pozbawiony mankamentów. Jego głównym celem było jak najsprawniejsze przesianie danych, które z pewnością nie są soczewkami. Głównym problemem, z którym musieliśmy się skonfrontować była konieczność ręcznego przejrzenia plików, które program zakwalifikował jako potencjalne soczewki. Przykładową

krzywą blasku, którą program zakwalifikował jako soczewkę, a która w rzeczywistości jest gwiazdą zmienną przedstawia rysunek poniżej.



Rysunek 3: Gwiazda zmienna w pliku 16389 zakwalifikowana przez algorytm jako soczewka

Wśród podejrzanych krzywych znajdowało się też dużo takich, których pomiary w określonych przedziałach czasowych były bardzo niedokładne i odbiegające od średniej. Zniwelowaliśmy ten problem wycinając część pomiarów.

3 Rezultaty

W celu oszacowania jaki błąd pierwszego rodzaju jest generowany przez nasz program, wzięliśmy potwierdzone soczewki udostępnione na stronie OGLE, i sprawdziliśmy ile z nich zostanie wykryte przez program. Nasz program, dla wartości $A = 3$ oraz górnej granicy σ_t punktów spełniających (4) wykrył 26 soczewek na 36 plików. Pliki na których przeprowadzone zostały testy zostały zapisane w folderze code/lens/ na naszym repozytorium na githubie.

3.1 Znalezione soczewki

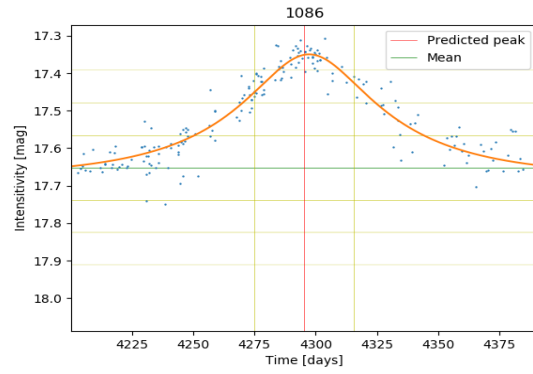
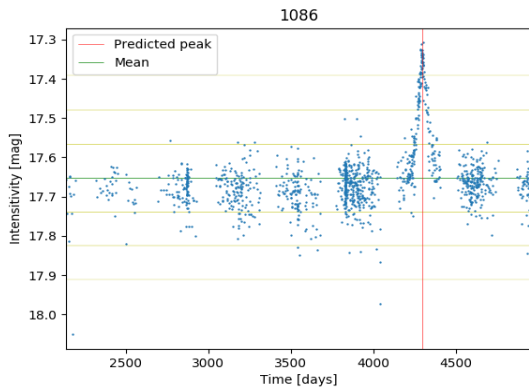
Uruchomiliśmy program na pełnym zbiorze dostarczonych danych przy parametrach:

$$A = 3, T = 30$$

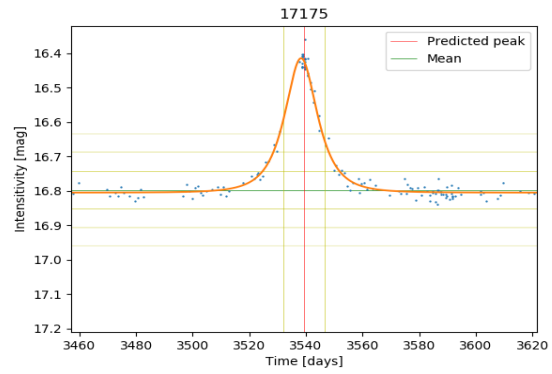
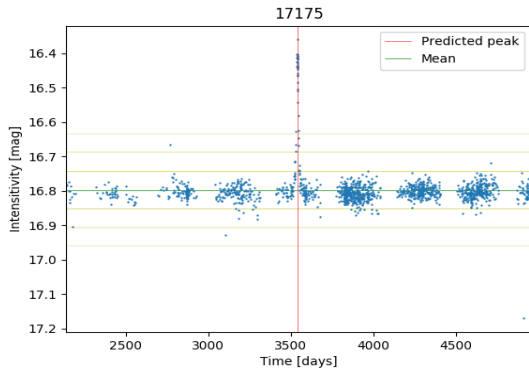
Według nas znaleźliśmy, przy tych parametrach 24 soczewki, 31 wykresów jasności jest do dalszej dyskusji, oraz przypadkowo 14 gwiazd zmiennych. Przykładowe wykresy jasności soczewek załączone są w podsekcji „Wizualizacje”.

3.2 Wizualizacje

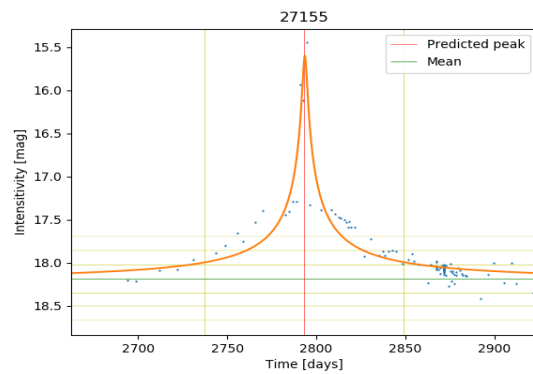
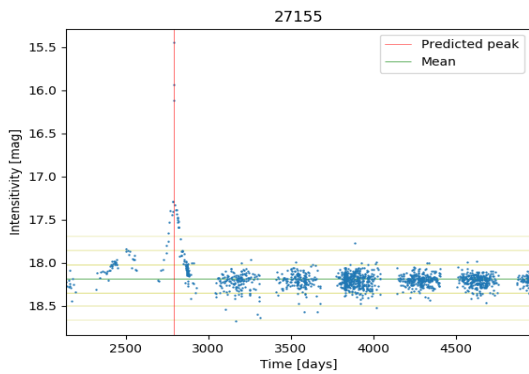
Wynikiem działania programu są pliki graficzne w formacie .png, które przedstawiają dopasowaną krzywą Paczyńskiego do danych w momencie wystąpienia potencjalnej soczewki. Poziome linie koloru żółtego to kolejne wielokrotności wartości σ_{mag} , zaś pionowa czerwona to predykcja momentu wystąpienia największego pojaśnienia, dokonana przez algorytm. Pliki zostały przez nas przejrane i wybrane zostały z nich te, które wizualnie najbardziej odpowiadały faktycznemu wystąpieniu soczewki. Część z nich przedstawiliśmy na rysunkach poniżej.



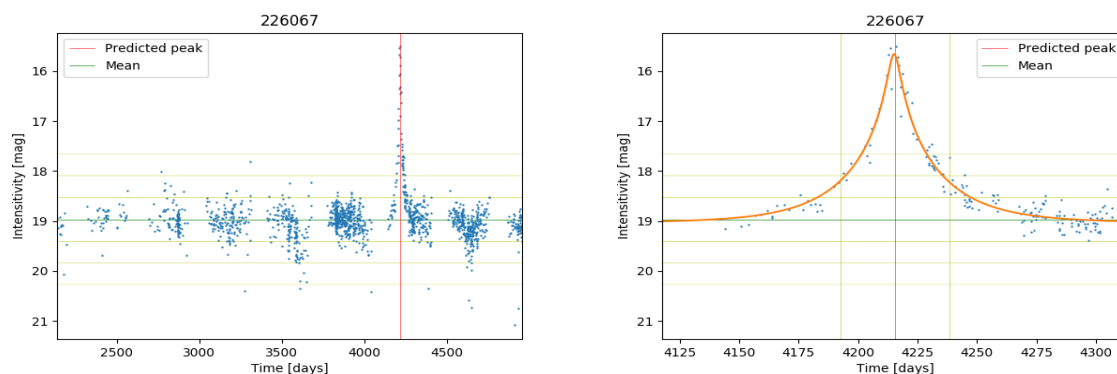
Rysunek 4: Peak znaleziony wśród danych dla gwiazdy w pliku o numerze 1086 wraz z przybliżeniem i dopasowaną krzywą Paczyńskiego



Rysunek 5: Peak znaleziony wśród danych dla gwiazdy w pliku o numerze 17175 wraz z przybliżeniem i dopasowaną krzywą Paczyńskiego



Rysunek 6: Peak znaleziony wśród danych dla gwiazdy w pliku o numerze 27155 wraz z przybliżeniem i dopasowaną krzywą Paczyńskiego



Rysunek 7: Peak znaleziony wśród danych dla gwiazdy w pliku o numerze 226067 wraz z przybliżeniem i dopasowaną krzywą Paczyńskiego

4 Bibliografia

ESA/Hubble and NASA. A Horseshoe Einstein Ring from Hubble.

https://en.wikipedia.org/wiki/File:A_Horseshoe_Einstein_Ring_from_Hubble.JPG, 2011. [Online; accessed 21-December-2011].

MesserWoland. Dwa promienie.

https://pl.wikipedia.org/wiki/Soczewkowanie_grawitacyjne#/media/%Plik:Dwa_promienie.svg, 2006. [Online; accessed 20-September-2006].

ESA NASA and STScI. Einstein Cross. https://pl.wikipedia.org/wiki/Krzy%C5%BC_Einsteina#/media/Plik:Einstein_cross.jpg,

1990. [Online; accessed 13-September-1990].

Jan Skowron. Analiza niestandardowych zjawisk mikrosoczewkowania grawitacyjnego gwiazd galaktyki, 2009. Rozprawa doktorska napisana w Obserwatorium Astronomicznym Uniwersytetu Warszawskiego pod kierunkiem prof. dr. hab. Michała Jaroszyńskiego.