Least Squared Analysis of Eclipsing Main Sequence Binary Systems

Noah Schapera, Mingxuan Liu

Emory University, PHYS311

Abstract:

In this project, we identify basic properties of an eclipsing binary system using period folding and manual analytic techniques. Using these parameters as constraints, we preform least squared regression analysis of the system to find additional parameters including individual star mass, temperature, and radius.

I modified the TwoStars C++ code provided in *An Introduction to Modern Astrophysics* by Carol and Ostile to generate a large number of model datasets for combinations of main sequence stars in approximately circular orbits. In addition, I developed a Python based least squared algorithm which compares observed transit depth of the system to those of models in order to determine which parameter set best matches the system in question.

Ultimately, we were able to successfully derive a likely parameter set for stars in the SV Cam. system which generally align with those calculated by spectroscopic analysis. However, our technique could still be improved by modelling a larger number of non-main sequence stars.

Introduction:

Eclipsing binary systems are systems of two or more stars which, from the perspective of earth, appear to eclipse one another during an orbital period. Both stars in the system are gravitationally bound to one another and orbit each other with a period that depends on the mass of each star and their orbital separation.

Because one star passes in front of each other with respect to earth, the apparent magnitude of the system varies with time. We observe two dips in brightness per period – a large drop when the hotter object (star A) passes behind the cooler object (star B), and a smaller secondary drop when star B passes behind star A. The maximum brightness of the system occurs when both stars are clearly visible.

Analyzing this light curve can tell us crucial information about the masses of each star, the temperature of each star, and their orbital separation. Therefore, developing a method to analyze these light curves quickly and objectively can help us better understand the distribution of stars and binary systems in the universe.

Methods:

Generating Models:

The TwoStars code provided in *An Introduction to Modern Astrophysics* [1] allows the user to generate a model dataset for an eclipsing binary system given a number of parameters for each star and the systems overall orbital parameters. For each star, the user can input a radius, mass, and temperature. In addition, the user inputs an orbital period, eccentricity, inclination, argument of the perihelion, and information about the systems center-of-mass velocity.

Because all of these parameters can be independently varied, a naïve approach of simply running a for loop for each parameter results in a time complexity of approximately O(N^13). This is clearly sub-optimal.

In order to simplify model generation, we assume the center of mass velocity of the system, eccentricity, and argument of the perihelion is zero. In addition, we know this system is an eclipsing binary so the inclination must be approximately 90 degrees.

Using the Lightkurve python package, Mingxuan developed a program to extract the period of the system from our observational data.

Main sequence stars have a known temperature, mass, and radius relationship. If we assume that both stars in our system are main sequence, we can simply vary the mass of each star independently and the temperature and radius will follow. For this purpose, we used a table of main sequence masses, radii, and temperatures for main sequence stars from the Australian National Telescope Facility (ANTF). [2]

Mass/M_{Sun}	**Luminosity/L_{Sun}**	**Effective Temperature (K)**	**Radius/R_{Sun}**	**Main sequence lifespan (yrs)**
0.10	3×10^{-3}	2,900	0.16	2×10^{12}
0.50	0.03	3,800	0.6	2×10^{11}
0.75	0.3	5,000	0.8	3×10^{10}
1.0	1	6,000	1.0	1×10^{10}
1.5	5	7,000	1.4	2×10^{9}
3	60	11,000	2.5	2×10^{8}
5	600	17,000	3.8	7×10^{7}
10	10,000	22,000	5.6	2×10^{7}
15	17,000	28,000	6.8	1×10^{7}
25	80,000	35,000	8.7	7×10^{6}
60	790,000	44,500	15	3.4×10^{6}

Implementing these assumptions to our model generation code we reduce the time complexity to O(N^2).

We modify the TwoStars program so that it runs in two nested for loops, one which varies star A's mass/temperature/radius, one which varies star B's mass/temperature/radius.

The program calculates the bolometric magnitude of the system given these parameters over a single period in time steps of ~T/1000. It then saves these data points in a text file and continues to the next parameter set.

<u>Least Squared Regression Algorithm:</u>

We are interested in calculating the difference in magnitude between the primary and secondary eclipse depth. This information will tell us about the temperature, and therefore masses, of the main sequence stars in our system.

We define a fit metric represent this goal

$$E = (\Delta M_{observed} - \Delta M_{model})^2 + (\Delta m_{observed} - \Delta m_{model})^2$$

Where ΔM represents the primary eclipse depth, and Δm represents the secondary eclipse depth.

This fit metric represents the degree to which the model's and the observational data's transit depths align with each other. If the difference between primary/secondary transit depths of the model and observational data is large, the fit metric will also be large, indicating a poor fit. If the difference is small, the fit metric will also be small, indicating a good fit.

Mingxuan used analytical techniques to measure the primary and secondary eclipse depth of each systems light curve. The Python script I developed to perform the least squared analysis takes in these measured eclipse depth values from the user.

Using for loops, the program iterates through all combinations (without repetition) of main sequence stars in the ANTF table.

For each combination, we access the text file containing that model dataset, and parse its contents using the CSV library. We calculate the models primary and secondary eclipse depth, compare it to our measured values, and calculate the fit metric.

We store this fit metric in an array along with information about the model used to generate it.

After iterating through all combinations of main sequence stars, we identify the minimum fit metric calculated and the parameters of the model used to calculate it. We return these parameters and end the program.

Results and Discussion:

Because of the assumptions we made in order to reduce model generation runtime, our algorithm is very limited in what types of systems can be analyzed. We assume the stars must be in a circular orbit, and they must both be on the main sequence. These restrictions prevent us from accurately analyzing any system with post-main sequence components like red giants or white dwarfs.

However, even with this limitation we were able to analyze some objects successfully. In particular, we were able to generate (relatively) accurate parameters for the SV Cam system.

Our algorithm identified that the closet model to the observed data consisted of a 0.5M_{sun}/3800K/0.6R_{sun} star with a 1.0M_{sun}/6000K/1R_{sun} star, which generated a primary transit depth of 0.61 and a secondary transit depth of 0.06.

The primary transit depth of the actual SV Cam system is 0.65, which aligns with our calculated value. However, the secondary transit depth is slightly larger than expected at 0.15. Subsequently, the fit parameter calculated for this model is still relatively high at 0.0094. Regardless, the model light curve shares significant visual similarities with the observed light curve.

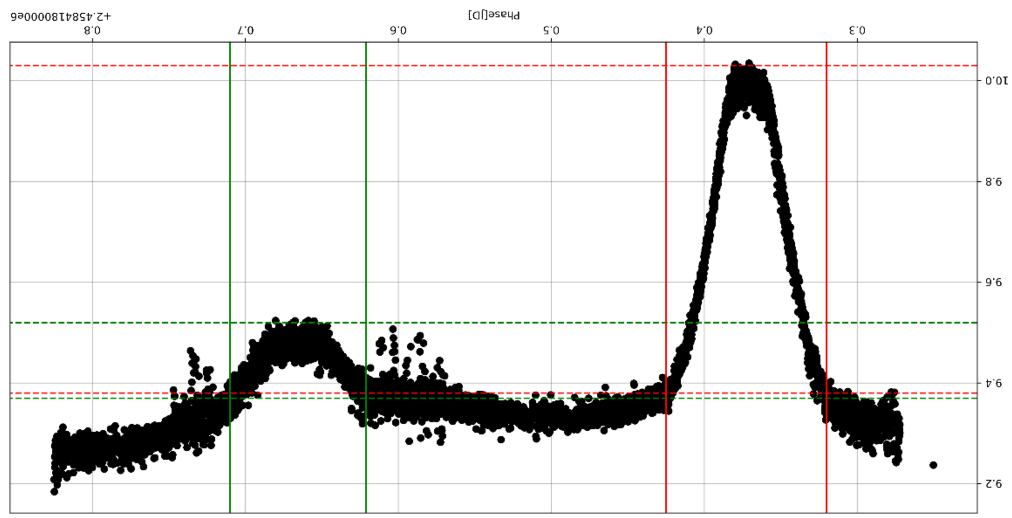

Figure 1: Observed light curve of SV Cam (apparent magnitude). Light curve flipped for ease of visual comparison.

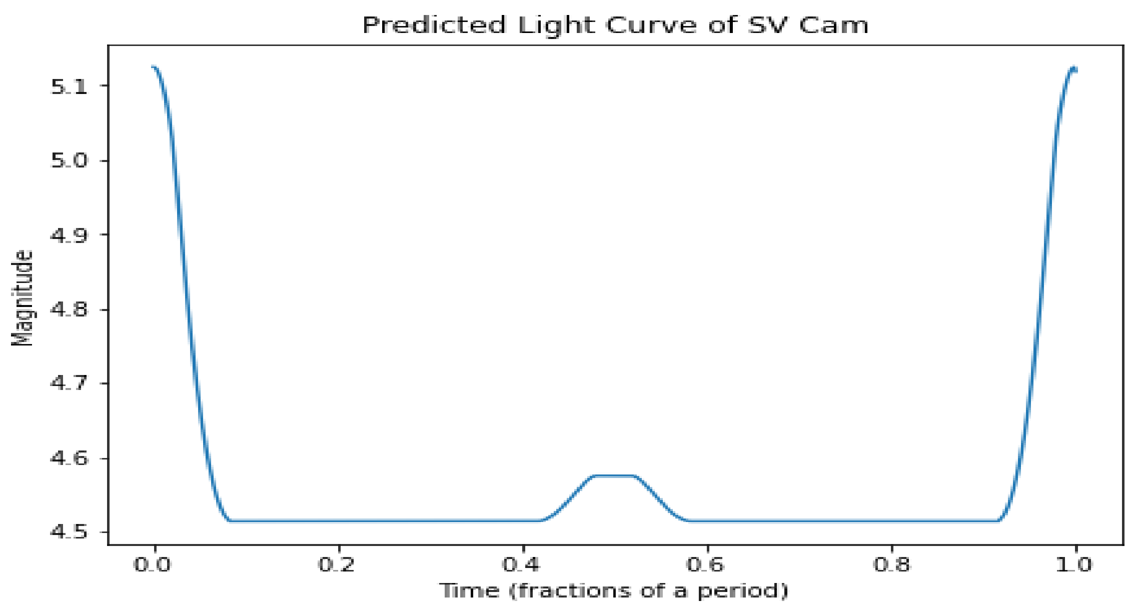

Figure 2: Predicted light curve of 0.5 - 1.0 solar mass star system, in bolometric magnitude

Note, while each of these light curves use a different magnitude measurement, they only differ from each other by a scale factor. Therefore, we can still calculate and directly compare the difference between magnitudes for the primary and secondary eclipse depth. In addition, the phase of the predicted model is shifted. The maximum primary eclipse depth occurs at T=0, while the secondary eclipse depth occurs at T=0.5.

According to H. Lehmann et. al. [3] the SV Cam system actually consists of a $0.70M_{sun}/0.76R_{sun}$ star and a $1.09M_{sun}/1.18R_{sun}$ star.

Given the relatively large step sizes between star masses in the ANTF table, it is not entirely surprising that our algorithm was not able to determine the precise mass of each star. However, the results are still useful in limiting the search parameters of future studies.

Using our predicted mass/temperature/radius values as starting points, it would now be much more efficient to generate a new set of models around these values to fine tune the analysis. For example, we could generate models for systems with stars between 0.9-1.1M_{sun} and 0.5-0.7M_{sun}. Applying our algorithm to these new model sets, we would likely find new system parameters even closer to the expected values. We could repeat this process an arbitrary number of times, asymptotically approaching SV Cam's true parameters.

In the future, one of the first modifications I would make to our analysis technique would be to extend our model generation code to include non-main-sequence stars. However, because these objects do not follow a predictable relationship between mass, temperature, and radius we may be forced to use the slow O(N^6) method of iterating through each star's mass, temperature, and radius independently. This would add a significant amount of time to our model generation, but it may be possible to overcome this limitation by using dedicated computational servers or implementing multithreading. This addition would significantly increase the number of systems we would be able to analyze using our algorithm, possibly including HW Vir.

However, even after implementing this change, we would still be unable to model contact binary systems. The TwoStars framework is simply not advanced enough to model the complicated interactions between stars that overlap one another, and we would likely need to transition to a new, much more sophisticated modelling software to deal with these cases.

Ultimately, I would classify this project as a partial success. We were able to develop a framework that correctly identifies the approximate parameters of main sequence eclipsing binary systems and would easily be able to extend this program to fine tune our analysis. However, this technique only works for a very small number of systems. With further development, I believe that this technique could be generalized to much wider variety of systems but doing so may require significant computational power.

Citations:

[2]Australian National Telescope Foundation. (2022, February 15). *Main sequence stars*. Main Sequence Stars. Retrieved April 26, 2022, from https://www.atnf.csiro.au/outreach/education/senior/astrophysics/stellarevolution_mainsequence.html

[1]Carroll, B. W., & Ostlie, D. A. (2018). TwoStars (C++) Source Code. In *An introduction to modern astrophysics*. story, Cambridge University Press.

[3]Lehmann, H., Hempelmann, A., & Wolter, U. (2002). High-resolution spectroscopic monitoring of SV camelopardalis. *Astronomy & Astrophysics*, *392*(3), 963–970. https://doi.org/10.1051/0004-6361:20020972