Towards an automated processing of Gaia eclipsing binaries

Christos Siopis, Gilles Sadowski
Institut d'Astronomie et d'Astrophysique, Université Libre de Bruxelles, 1050 Bruxelles, Belgium.
Christos.Siopis@ulb.ac.be, gilles.sadowski@ulb.ac.be

Introduction

Eclipsing variable stars are interesting objects, not only in their own right, but also because their fortuitous geometry sometimes allows for the simultaneous extraction of several astrophysical parameters (masses, radii, etc.) from their light and radial velocity curves. These valuable constraints for stellar theory cannot be obtained in any other way for such a wide variety of stellar types and configurations. Eclipsing binaries can also be harnessed as distance indicators [1, 2, 3].

The International Variable Star Index (VSX) contains $\approx$ 40,000 eclipsing binaries. Despite their astrophysical significance, only $\approx$ 100 of them have been carefully studied to this day. In fact, as a result of several microlensing and extrasolar planet surveys over the past two decades, many more eclipsing variables have been observed than there are in the VSX. However, eclipsing binaries often remain unidentified in these surveys. The future promises an ever-growing avalanche of data. Gaia is expected to discover $\approx 10^6$ eclipsing variables whereas the Large Synoptic Survey Telescope (LSST) is expected to boost that number to $\approx 10^7$.

The aim of this paper is to briefly present the current state of development of the segment of Gaia's data reduction pipeline that pertains to eclipsing binary processing. It should be emphasized that this project is still under development, and many aspects presented below are subject to change.

Analysis of large datasets, such as expected from Gaia, requires special considerations. Most obvious among them is the need for a fully automated processing. Indeed, when eclipsing binary light curves (and, possibly, radial velocity curves) are "solved" in the usual, semi-manual way (e.g., using the Wilson-Devinney code [4, 5, 6]), the human expert is responsible for (i) acquiring the data and assessing their quality, (ii) judging the nature of the eclipsing system in order to come up with initial estimates for the values of the physical parameters to be fit, and finally (iii) fitting the data and interpreting the solution. These steps need to be automated when working with large datasets. In the case of the Gaia pipeline, the procedure is broken down into the following way:

1. Use photometric data to determine whether the object is a variable star.
2. Use spectroscopic data, when available, to determine whether the object is a spectroscopic binary (SB1 or SB2).
3. If it is a variable star, find its period and determine the probability that it be an eclipsing binary.
4. If it is a spectroscopic binary, calculate an orbital solution.

At this point, evaluate whether the object should be processed as an eclipsing binary. If so, then:

5. Evaluate the quality of the photometric (and spectroscopic, when available) data.
6. Perform a global optimization to (hopefully) land somewhere inside the convex region of the global minimum.
7. Perform a local optimization to home in more accurately on the minimum.
8. Evaluate the quality of the solution.

Steps 1 and 2 are the responsibility of Coordination Units 5 ("Photometric Processing") and 6 ("Spectroscopic Processing"), respectively, in Gaia's data reduction pipeline. These CUs convert CCD pixel data to time series of calibrated photometry and spectroscopy for each object detected by Gaia.

Step 3 is triggered when CU5 detects a photometric variability that exceeds a certain threshold. The object is then processed by CU7 ("Variability Processing") in order to determine its period and to characterize the variability (see [7] for a presentation of current CU7 software as applied to the analysis of Hipparcos variables).

Step 4 is triggered when an object is identified by CU6 as a spectroscopic binary (SB1 or SB2). In that case, the working group "Spectroscopic Binaries", that has been set up under CU4 ("Object Processing"), produces an orbital solution. This happens as part of the general pipeline flow, irrespective of whether the object is a photometric variable or not.

The processing of an object as an eclipsing binary (steps 5-8) is the focus of this presentation. It is implemented by the "Eclipsing Binaries" and "Solution Combiner" working groups that have been set up to that end under CU4. They are triggered by either of the following two conditions: (i) The probability that the object be an eclipsing variable (step 3) is greater than some threshold value; or (ii) the object is a photometric variable as well as a spectroscopic binary, with an orbital (spectroscopic) period consistent with its photometric period (step 4).

Step 5 consists primarily of a number of sanity checks to ascertain that the input data make sense. For the time being, the Eclipsing Binaries working group relies on the checks that are made upstream in the pipeline (steps 1-4). Indeed, most data quality issues should make it impossible for an object to be identified as an eclipsing binary in steps 3 or 4. However, as development of steps 1-4 is still in progress, the possibility remains that certain tests should have to be implemented in step 5.

Steps 6 and 7 constitute the core of the eclipsing binary processing. We followed the usual practice in the field, namely finding the synthetic light/radial velocity curves that best fit (in a maximum-likelihood/least-squares sense) the observed light/radial velocity curves. The eclipsing binary simulator that produces the synthetic curves is outlined in section 1 below, while the solution finder is outlined in section 2, along with some remarks about evaluating the quality of the solution.

There are three aspects to eclipsing binary processing in Gaia that set it apart from other large-survey projects. First, Gaia is a scanning mission. As a consequence of the "scanning law" (which prescribes how Gaia's spin axis evolves with time) and depending on the position of the object in the sky, there will be between about 30 and 200 multicolor photometric observations (approximating the V, R and B bands) per object during the 5-year mission, with a mean of ≈ 70 triplets of photometric data per object. Similarly, there will be between about 20 and 120 spectroscopic observations (mean: ≈ 30). Furthermore, these observations will follow a non-uniform (albeit quasi-regular) sampling in time. Therefore, the solution process will have to cope with a much smaller number of datapoints than in other surveys, and those datapoints may not even sample adequately the crucial segments of the light curve that give maximum leverage for finding a solution, such as the points of ingress, egress and minimum light. This will probably be the greatest constraint and challenge in the processing of Gaia eclipsing binaries.

Second, the mission duration, in combination with Gaia's scanning law, implies a large number of short-period eclipsing binary detections. Therefore, the data processing software cannot be written assuming a detached geometry or spherical stars. Rather, the more general Roche model has to be assumed.

Finally, there is a mission requirement that the software pipeline be written in the Java programming language, and follow stringent coding guidelines. There is no publicly available eclipsing binary software with the requisite feature set written in Java, so everything had to be written from scratch. When completed, the software will be released under a free software license. We hope that the availability of this clearly written and well documented code will be of benefit to the entire eclipsing binary community.

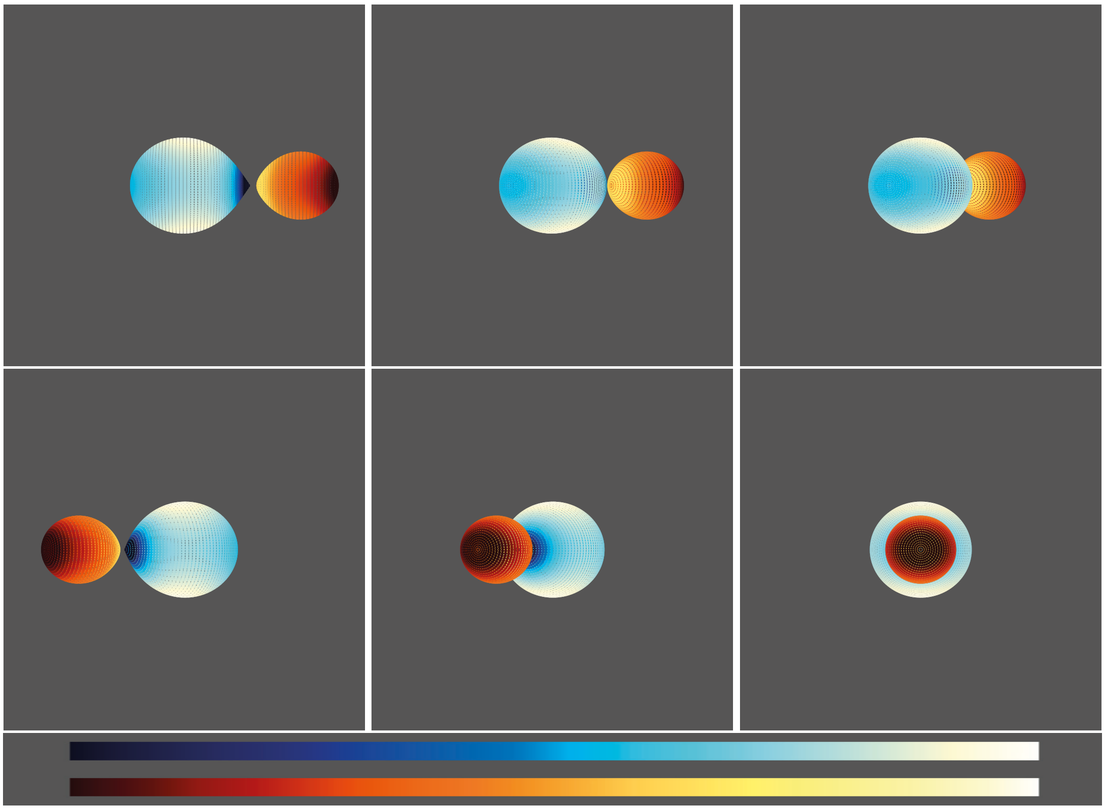

Figure 1: The eclipsing binary simulator represents the flux-emitting surfaces of the two stellar components by a two-dimensional mesh, corresponding to equipotential surfaces of the Roche model. In the example shown here, the mesh of the bigger and of the smaller component are made up of 4031 and 2065 surface elements, respectively. Surface temperatures are color-coded as shown in the two palettes at the bottom, separately for each component. They range from 6600 K to 7150 K (left to right) in the top palette, and from 4900 K to 5500 K in the bottom palette. From left to right, and from top to bottom, the snapshots correspond to phases $0.0, 0.10, 0.15, 0.57, 0.67$ and 0.75.

1. The eclipsing binary simulator

The purpose of the eclipsing binary simulator is to produce synthetic multicolor light and radial velocity curves, given a set of physical parameters that completely describe the system. For the reason presented in the Introduction, we implemented a fully general Roche model simulator where the surfaces of the two stars are represented by meshes that cover the equipotential surfaces of the Roche potential (Fig. 1). All important binary star geometries can thus be modeled: Detached, semi-detached and overcontact. Orbits can be circular or (for detached and semi-detached systems) elliptical. The emitted fluxes are currently calculated from Phoenix and Atlas stellar atmosphere models. The simulator can take into account the effects of limb darkening, gravity brightening, mutual irradiation, asynchronous rotation of the two components, third light, and spots. These are probably already more effects than we will be able to fit given the expected quality of the data. Other effects that we do not expect to be able to fit, and which are not part of the simulator, include the presence of an accretion disk and/or of clouds of material in the binary system.

In order to expedite the early phases of simulator code development, we studied the algorithms used in NIGHTFALL, an eclipsing binary software written in C by Rainer Wichmann (Hamburg Observatory) and available as free software. We adopted many of Nightfall's algorithms, sometimes with modifications, and rewrote them in Java in a way consistent with Java coding patterns and Gaia project requirements.

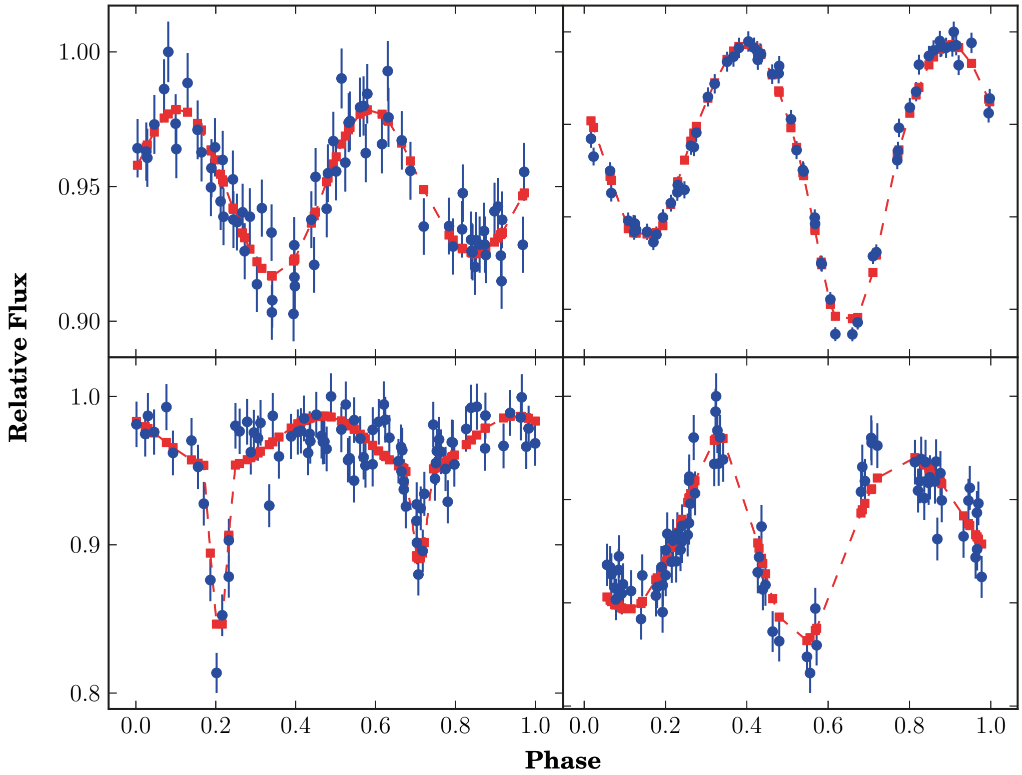

Figure 2: A preliminary test of the solution finder. The circles with error bars are simulated Gaia eclipsing binary photometric observations. The squares connected with a dashed line indicate the best fit. The exact period was assumed to be known; period search is normally performed by CU7 but here the period was taken from the simulator. The fitted quantities are the fill factors for both components, the time of primary eclipse, the flux scaling, the surface temperature of one component, the eccentricity, the inclination, and the argument of periastron.

The simulator is in an advanced stage of completion. The two principal remaining tasks are the calculation of limb-darkening coefficients for Gaia's photometric filters, and a detailed comparison of the output light curves with those produced by the Wilson-Devinney code, the standard benchmark code in the field.

2. The solution finder

The core of this project is the solution finder. It estimates, via a least-squares fitting procedure, the physical parameters (masses, radii, orbital elements etc.) that best fit the observations. The least-squares optimization problem is nonlinear in the general case, therefore a two-step approach is adopted: First, a global optimization method is used to (hopefully) place the search vector inside the convex region of the global minimum; then, a local optimization method takes over to home in more efficiently on that minimum.

The global optimizer does, in fact, what is arguably the most important task of the human expert in the case of quasi-manual processing, namely to provide a good initial estimate for the local optimization problem. Our code currently only implements a very rough "global optimizer": It compares the observed light curve to a pre-calculated library of 10,000 reference light curves, which sample parameter space in a representative way, in order to find the reference curve that most resembles the observations in a least-squares sense.

The local optimization is at a more mature stage of development. We are testing three local optimizers for performance and accuracy: Powell's conjugate gradient descent, downhill Simplex (Nelder-Mead), and Levenberg-Marquardt. The first two require no derivatives whereas the third one does. Derivatives may be analytically computed for a few parameters, but most require a numerical evaluation which is a CPU-intensive and sometimes treacherous process. Unfortunately, algorithmic differentiation is not currently an option, even though it would be ideally suited for this task, because it is not yet sufficiently developed for Java.

Figure 2 shows some preliminary results from using the solution finder to fit simulated eclipsing binary light curves. The solution finder is still work in progress. The two major tasks under way are (i) improvements to the global optimizer, and (ii) error estimation, in a way that will be both informative (especially in the case of underconstrained problems due to bad phase coverage of data) and not prohibitive in terms of computing resources.

Conclusion

Gaia is expected to discover a large number ($\approx 10^6$) of eclipsing binary systems. However, most of these systems will be observed with relatively few datapoints compared to the number required for obtaining well-constrained physical parameter values. As we are making progress towards writing a fully automated eclipsing binary solver for Gaia, it is important to define the expectations for the software. We believe that it should be able to (i) provide good solutions when data of adequate quality and quantity are available; (ii) identify, whenever possible, "interesting" objects for further study and possibly follow-up observations (e.g., systems suitable for determining spectrophotometric or purely photometric mass ratios, systems that might be good distance indicators, etc.); and (iii) enable meaningful statistical studies for the majority of objects with poorly constrained solutions.

References

[1] Russel, H. N. 1948. *Harvard Obs. Monograph* No. 7, 181.

[2] Paczynski, B. 1997. In: Livio, M. (Ed.), The Extragalactic Distance Scale. *Cambridge University Press*, CA, London, p. 273.

[3] Paczynski, B. 2000. *A&AS*, 196, 1001.

[4] Wilson, R. E., Devinney E. J. 1971.*ApJ*, 166, 605.

[5] Wilson R. E. 1979. *ApJ*, 234, 1054.

[6] Wilson R. E. 1990. *ApJ*, 356, 613.

[7] Dubath, P. et al. 2011. *MNRAS*, 414, 2602.