phot2lc

Release 1.7.3

Jul 20, 2023

Contents

phot2lc is a pure-Python interactive tool for extracting light curves from time-series photometric data.

phot2lc is largely inspired by WQED (Thompson & Mullally 2009, 2013), and is currently designed to work with the photometric outputs from MAESTRO (Dalessio 2010, 2013), ccd_hsp (Kanaan et al. 2002), ULTRACAM (Dhillon et al. 2007), and HiPERCAM (Dhillon et al. 2021).

CHAPTER 1

Installation

Current version:

You can install phot2lc with pip:

```
pip install phot2lc
```

The phot2lc project can also be found at GitHub and at PyPI.

CHAPTER 2

Requirements

- Python >= 3.6
- Matplotlib >= 3.1.3
- Astropy >= 4.0
- LMFIT >= 1.0.1

CHAPTER 3

Package Contents

- **phot2lc** – The main light curve extraction program
- **photconfig** – Configures some of phot2lc's defaults
- **weldlc** – Welds together multiple light curve files into one
- **quicklook** – Generates quicklook plots of light curves and their periodograms

3.1 Guide

3.1.1 Introduction

3.1.1.1 What is phot2lc?

First of all, **phot2lc** is *not* a photometry pipeline. Rather, **phot2lc** is a program that ingests the output from various photometry pipelines and provides users with a set of tools to extract and manipulate divided light curves. It is largely inspired by WQED, and provides functionality such as comparison star selection, aperture size selection, removing poor-quality data, polynomial fitting, and barycentric time corrections.

3.1.1.2 Currently Supported Photometry Pipelines

phot2lc will not automatically work with photometric data from all pipelines. Each pipeline will produce output with different content and formats, and **phot2lc** must be configured to properly ingest the data.

Output from the following photometry pipelines is currently supported:

- ccd_hsp (Kanaan et al. 2002, source code = hsp)
- MAESTRO (Dalessio 2010, 2013, source code = mae)
- ULTRACAM (Dhillon et al. 2007, source code = ucm)
- HiPERCAM (Dhillon et al. 2021, source code = hcm)

If your preferred photometry pipeline is not listed here, please *contact Zach Vanderbosch* about adding support for it in **phot2lc**.

3.1.1.3 Currently Supported Telescopes

In addition to loading in outputs from photometric pipelines, **phot2lc** also loads in one or more of the actual images (FITS, ucm, or hcm format), both for display purposes and to grab some key header information such as observer name, filter name, exposure time, observation timestamps, etc. For different telescopes and instruments, the header keywords may have different names or the timestamps may have different formats, so **phot2lc** needs to be properly configured to read the data.

Data from the following telescopes/instruments are currently supported:

- McDonald 2.1m with ProEM EMCCD (telcode = mcd2)
- McDonald 2.7m with Coude Guide Camera (telcode = coud)
- Perkins 1.8m with PRISM (telcode = perk)
- Paul and Jane Meyer Observatory 0.6m with ProEM EMCCD (telcode = pjmo)
- Las Cumbres Observatory 1.0m with Sinistro (telcode = lco1)
- Kitt Peak 2.1m with KPED (telcode = kped)
- Palomar 200-in with CHIMERA (telcode = p200)
- Pico dos Dias Observatory 1.6m with Ixon Camera (telcode = opd)

If your preferred telescope+instrument is not listed here, please *contact Zach Vanderbosch* about adding support for it in **phot2lc**. In this case, however, users may find it relatively easy to add support for a new instrument themselves by adding a new entry into the “teledat.py” script that is installed with **phot2lc**.

3.1.2 Configuration

- *Running photconfig*
- *The stars.dat File*

3.1.2.1 Running photconfig

When you first install phot2lc, you will need to change some of the default settings before it works properly on your machine and with your data. A script named **photconfig** is provided for this purpose. Run photconfig from the command line by simply typing

```
photconfig
```

at which a point you will be lead through a series of prompts to edit the configuration file “config.dat.” The parameters that need to be set are:

- **author**

- **image_list_name**
- **pixloc_name**
- **photbase_name**
- **stardat_location**
- **default_telescope**
- **default_source**
- **default_image**
- **default_object**

After initially installing phot2lc, the default values for these parameters should be as follows:

```
author            = Zach Vanderbosch
image_list_name   = olist
pixloc_name       = phot_coords.orig
photbase_name     = runbase
stardat_location  = /home/zachvanderbosch/data/stars.dat
default_telescope = mcd2
default_source    = hsp
default_image     = None
default_object    = None
```

Below are detailed descriptions for each parameter's meaning and use:

- **author**: phot2lc User name (e.g. your name).
- **image_list_name**: Name of the file which contains the list of image filenames corresponding to each photometric data point. The images are used for a few purposes, such as grabbing some key header information, and using the first filename in the list to display an image for target and comparison star identification. Depending on which photometry pipeline was used, the image headers may also be used to generate time stamps for each data point in the light curve.
  - If using MAESTRO, ULTRACAM, or HiPERCAM output, the time stamps will automatically come from the photometric output files instead of the images, so only the first image listed within image_list_name is used for display purposes and for grabbing some header information. This parameter can be set to **None** as long as a single image name is provided using the **phot2lc** command line option -i (–image).
  - If using ccd_hsp output, this file is only needed if you want to generate time stamps for your light curve by grabbing times from each individual FITS header. If this parameter is set to **None**, then timestamps will be generated using the exposure time multiplied by the number of exposures since the first image. In this case, the name of the first image must be provided using the **phot2lc** command line option -i (–image).
- **pixloc_name**: Name of the file containing initial guesses for the pixel coordinates of each star.
  - This is optional, used only by phot2lc for marking stars when displaying the first image, and can be set to **None** if unavailable. When using ULTRACAM or HiPERCAM pipelines, it is assumed that this file will be structured like a ".ape" file created by the setaper routine of the respective pipelines. For ccd_hsp and MAESTRO, a simple ascii file with space-delimited columns is expected, where the first and second columns provide the x and y coordinates, respectively. In all cases, the target is assumed to be the first set of coordinates in the file.
- **photbase_name**: Base filename for the output files generated by each photometric pipeline.
  - **phot2lc** uses this parameter to search for all files in the current working directory containing photbase_name. These are the files generated by a given photometry pipeline that contain the raw photometric counts for the target, comparison stars, and sky, and in some cases also contain the timestamps for each

  image and count uncertainty estimates. This parameter can be overridden with the **phot2lc** command line option -p (–photname).

  - For ULTRA/HiPERCAM, it is assumed that the pipelines were used in "variable aperture size" mode, and so only one output photometry file exists. If **phot2lc** finds multiple files containing the given base name, an error will occur if either of these pipelines are defined as the source.

- **stardat_location**: Path and filename for file containing the RA and Dec corresponding to a given object name (e.g. /user/zvander/stars.dat). This parameter is required, as is *the stars.dat file*.
- **default_telescope**: Default telescope code.
  - Telescope codes correspond to an entry within teledat.py and describe both the telescope, instrument, and FITS header keywords needed to set some phot2lc parameters. For a list of currently supported telescope codes, see the *Introduction*, or type **phot2lc -c** on the command line. This default setting can be overridden with the **phot2lc** command line option -t (–telescope).
- **default_source**: Default photometry pipeline whose output is being loaded in to **phot2lc**. Must be one of the currently supported pipelines listed in the *Introduction*. This default setting can be overridden with the **phot2lc** command line option -s (–source).
- **default_image**: Default image name.
  - When using MAESTRO output, or ccd_hsp output with only the first image for time stamp generation, you may anticipate all of your first images to have the same name (e.g. firstimage.fits). In such a case, you can use this parameter to automatically set the image name instead of having to define it with the **phot2lc** -i command line option. If set, you can still use the -i option to override the default in the event a different image name is used. If unused, set this parameter to **None**.
- **default_object**: Default object name.
  - A useful parameter to set if you intend to reduce a lot of light curves at once for a single object whose name cannot be obtained directly from the image header. This object name needs to correspond to an object name within your stars.dat file, since this is how the object's coordinates are acquired for barycentric time corrections. If unused, set to **None**. This default setting can be overridden with the **phot2lc** command line option -o (–object).

3.1.2.2 The stars.dat File

stars.dat is a seven-column, whitespace-delimited text file used to store object names and their corresponding RA and Dec coordinates. **The RA and Dec must be ICRS J2000 coordinates for proper** *barycentric corrections*. You can actually name the stars.dat file whatever you want, as long as it matches the filename you provide in your **config.dat** file, but from here on out this documentation will refer to the file as stars.dat. Below are three example lines within a stars.dat file:

```
GD358           16 47 18        +32 28 32
ZTFJ0139+5245   01 39 06.17     +52 45 36.89
V386ser         16 10 33.62889 -01 02 23.20995
```

As you can see, each entry needs an object name (*no spaces allowed!*), and an RA and Dec in HMSDMS format with only spaces as delimiters. The decimal values can be to any precision you want, and you can put as many spaces between each column as you want. When you run phot2lc, the program will look for an entry in stars.dat that matches the object name retrieved from the image header or given as a command line argument.

3.1.3 Quickstart

This quickstart guide assumes you have already performed photometry using one of the supported pipelines (see *Introduction*), and have already *configured phot2lc* to run properly on your machine.

3.1.3.1 Basic Usage

phot2lc is a command line tool, and can be executed from the command line by simply typing

```
phot2lc
```

phot2lc also has several command line arguments available:

```
-h --help      Show command line options
-c --codes     Print a list of available telescope codes
-t --telescope Code name for telescope used
-s --source    Code name for photometry program used
-p --photname  Base filename for photometry output files
-i --image     Name of specific image instead of list
-o --object    Name of object matching stars.dat entry
-b --barycorr  If invoked, do NOT perform barycentric corrections
```

If these command line arguments are used, they will *override the defaults* that are set within the config.dat file by the photconfig program.

3.1.3.2 Input Files

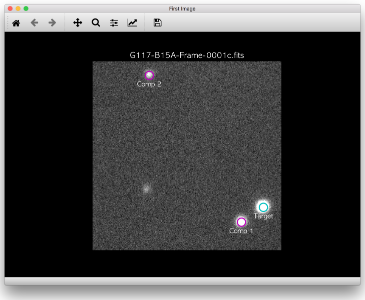
For ccd_hsp output, phot2lc will assume the photometry files have base names of *runbase* followed by the aperture size (e.g. runbase2.5), while for MAESTRO output the photometry files are assumed to have base names of *counts*, again followed by the aperture size (e.g. counts_02.5).

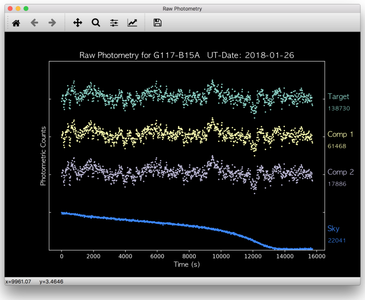
In addition, the following files/info should be present:

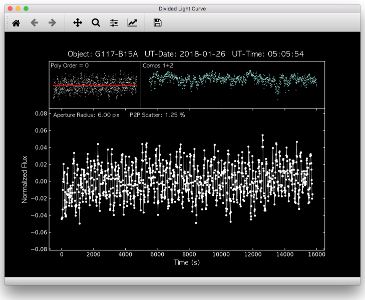
- *stars.dat* (for RA and Dec coordinates)
- The first FITS image (or all images within **image_list_name**)
- If available, the file pointed to by **pixloc_name**
- An entry within teledat.py corresponding to your telescope/instrument
  - If you need to add a new entry into teledat.py for your telescope/instrument, you can do this yourself by manually editing the teledat.py file. The format for teledat.py entries is explained within the script itself. Alternatively, you can *send a request to Zach Vanderbosch* to add your telescope configuration into the official phot2lc package.

3.1.3.3 Startup

Once you have started phot2lc and it successfully loads all the necessary files, the following three windows will appear (*click to enlarge*):







From left to right, these windows are named:

- **First Image**: Displays the first image. The target and comparison stars are marked if a *pixloc_name* file was provided, otherwise this will just be an unmarked image. This image will be saved as a PNG file when the program is run.
- **Raw Photometry**: Displays the raw photometry for target and comparison stars, along with the amount of sky subtracted from the target. The mean values are displayed to the right of each light curve.
- **Divided Light Curve**: Primarily displays the divided light curve, but also displays the summed comparison star flux (top right panel) and the polynomial being used to de-trend the light curve (top left panel). This is where most of the interactive tools are used.

3.1.3.4 Command Options

Similar to WQED, commands in phot2lc are executed via keyboard inputs. After startup and once you have clicked within one of the windows, you can print out a full list of command options in your terminal by pressing “?”. The output will look like the following:

```
COMMAND LIST - Divided Light Curve:
  - Type '?' to re-print this list of commands.
  - Type 'd' to delete the point nearest the cursor.
  - Type 'a' to add back a deleted point.
  - Type 'A' to add back ALL deleted points.
  - Type 'g' to activate/deactivate garbage rectangle.
  - Type 'r' to activate/deactivate reverse garbage rectangle.
  - Type 'z' to activate/deactivate zoom rectangle.
  - Type 'Z' to restore zoom to original.
  - Type 'x' to perform sigma clipping.
  - Type 's' to toggle the display of deleted points.
  - Type 'f' to perform a polynomial fit, WITHOUT sigma rejections.
  - Type 'F' to perform a polynomial fit, WITH sigma rejections."
  - Type 'c' to choose comparison stars for division.
  - Type 'v' to move to previous aperture size.
  - Type 'w' to move to next aperture size.
  - Type 'Q' to close plots and exit the program.
  - Type 'W' to close plots and continue without grid search.
  - Type 'G' to close plots and continue with grid search.

COMMAND LIST - Aperture Selection:
```

(continues on next page)

(continued from previous page)

```
  - Type '?' to re-print this list of commands.
  - Type 'd' to delete the point nearest the cursor.
  - Type 'A' to add back all deleted points.
  - Type 'Q' to close plots and exit the program.
  - Type 'W' to save lightcurve with *USER* Selection.
  - Type 'G' to save lightcurve with *GRID* Selection.
```

The "Divided Light Curve" command list provides your options when you are working within the *Divided Light Curve* window. Within the *First Image* and *Raw Photometry* windows, the only commands available are "?", "Q", "W", and "G". The other command list for aperture selection is for the next window that appears if you decide to continue light curve extraction by typing the "W" or "G" keys when working in the *Divided Light Curve* window.

3.1.3.5 Aperture Selection

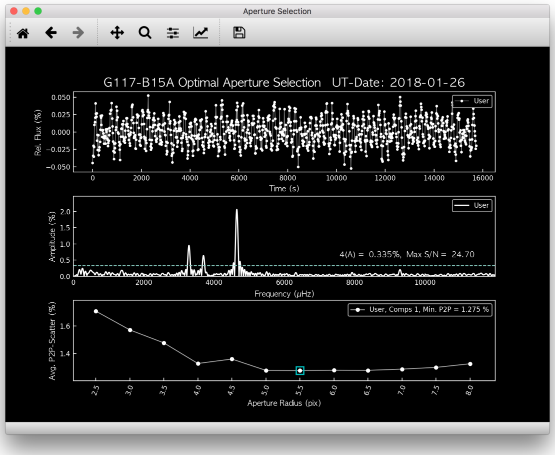
phot2lc selects the optimal aperture size by identifying the light curve with the lowest average point-to-point (P2P) scatter. The average P2P scatter is defined as:

$$\langle \mathrm{P2P} \rangle = \sqrt{\frac{\sum_{i=0}^{N-1} (y_{i+1} - y_i)^2}{N-1}}$$

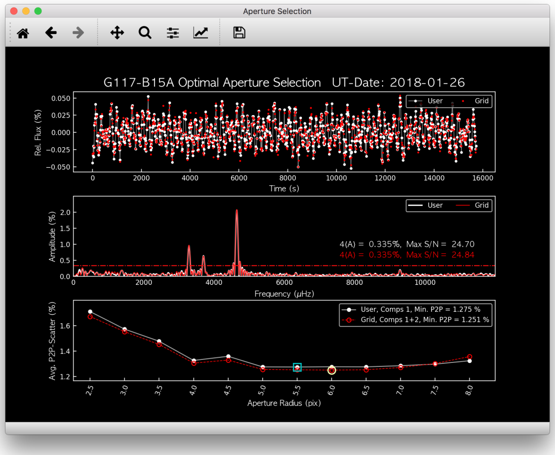
The $\langle \mathrm{P2P} \rangle$ is determined for all light curves using the same settings defined in the *Divided Light Curve* window (i.e. same deleted points, same polynomial division, same comparison stars used). Currently, this selection is automated and phot2lc does not allow manual selection of a different aperture size. *A future version of phot2lc will likely provide this capability.*

After pressing the "W" or "G" key from any of the previous windows (i.e. *Divided Light Curve*, *Raw Photometry*, or *First Image*), these windows will be closed and a new window will appear named *Aperture Selection*. This window displays the optimal light curve along with its periodogram, and a plot of $\langle \mathrm{P2P} \rangle$ versus aperture size for all light curves.

If you chose the "W" option to continue, only one light curve, periodogram, and $\langle \mathrm{P2P} \rangle$ curve will be shown (left image below). If you chose the "G" key, however, a grid search will have been performed considering all possible comparison star and aperture size combinations. In this case, two light curves, periodograms, and $\langle \mathrm{P2P} \rangle$ curves will be shown (right image below). One set (colored white) corresponds to the standard optimization procedure performed with the User-selected comparison stars, the same as if you had pressed "W". The second set (colored red), corresponds to the optimal comparison star combination found by the grid search.



4(A) = 0.335%, Max S/N = 24.70


Aperture Selection
G117-B15A Optimal Aperture Selection UT-Date: 2018-01-26
Rel. Flux (%)
User
Grid
Time (s)
Amplitude (%)
User
Grid
Frequency (μHz)
Avg. P2P-Scatter (%)
User, Comps 1, Min. P2P = 1.275 %
Grid, Comps 1+2, Min. P2P = 1.251 %
Aperture Radius (pix)

4(A) = 0.335%, Max S/N = 24.70
4(A) = 0.335%, Max S/N = 24.84

If you did not perform a grid search, you can now save the optimal light curve by again pressing “W”. If you did perform a grid search, you have the option save either the optimal user-selected light curve with “W” or the optimal grid-search-selected light curve with “G”.

3.1.3.6 Output Files

- *The .lc File*
- *The .phot File*
- *The phot2lc_log File*

Upon choosing to save your light curve with “W” or “G”, phot2lc will perform the *barycentric time corrections* to each point and then provide you with three output files. The *.lc* file contains the divided light curve while the *.phot* file contains the raw photometry corresponding to the optimal aperture size. These files are modeled very closely after the *.lc1* and *.wq* files generated by WQED. The third output file is called, *phot2lc_log.txt*, and stores information about your saved light curve which allows phot2lc to essentially restore your session for a particular object (*more details below*).

The .lc and .phot files start with two time columns, the first corresponding to the original times and the second corresponding to the barycentric corrected times. Both are provided in the event that you consider the barycentric corrections for your particular object to be unreliable for any reason.

The naming convention for the files are **<object>_<obs_date>.lc** and **<object>_<obs_date>.phot**, where <object> is replaced by the object name and <obs_date> is replaced by the observation date in YYYYMMDD format.

3.1.3.6.1 The .lc File

The .lc file consists of a header followed by three columns of data. The header provides information with regards to both the observations and the light curve extraction. The three columns are (1) the mid-exposure times in seconds relative to the first exposure, (2) the relative flux values, and (3) the error on the relative flux. Below is an example .lc1 file showing the header and the first and last three rows of data:

```
# Object     = G117-B15A                     # Name of Object
# RA         = 09 24 15.27                   # Object Right Ascension
# Dec        = +35 16 51.3                   # Object Declination
# Telescope  = McDonald 2.1m                 # Name of Telescope
# Instrument = ProEM                         # Name of Instrument
# TeleCode   = mcd2                          # Teledat Code Name
# Date       = 2018-01-26                    # Mid-Exposure UTC Date at T0
# Time       = 05:06:01.500                  # Mid-Exposure UTC Time at T0
# MJD        = 58144.212517361               # Mid-Exposure UTC MJD at T0
# Exptime    = 15.000000                     # Exposure Time (s)
# Filter     = BG40                          # Filter Name
# BJED       = 2458144.718681479             # Mid Exp. Barycentric Julian Date
# Barycorr   = True                          # Barycentric Corrections Applied?
# ApPhot     = ccd_hsp                       # Photometry Program
# OrigFile   = runbase6.                     # Source Photometry Filename
# ApRadius   = 6.00                          # Aperture Radius (pixels)
# AvgScatter = 1.25                          # Avg. Point-to-Point Scatter (%)
# Comps      = 2                             # Comparison stars used
# PolyOrder  = 3                             # Degree of Polynomial Division
# Nkeep      = 1049                          # Number of points in light curve
# Ndelete    = 0                             # Number of points removed
# Author     = Zach Vanderbosch              # Author of this light curve
# CreatedOn  = 2021-02-10 08:57:51.459       # Date created
```

(continues on next page)

(continued from previous page)

```
# Columns: Time (s), Relative Flux, Relative Flux Error
        0.000  -0.031284   0.002743
       15.000  -0.030957   0.002611
       30.000  -0.029607   0.002626
       ...        ...         ...
    15690.173  -0.011376   0.002271
    15705.173  -0.029081   0.002532
    15720.174  -0.024124   0.002615
```

3.1.3.6.2 The .phot File

The .phot file also starts with a header providing some of the same information as the .lc file, and is then followed by several columns of data. The first column is the same as for the .lc file, providing the mid-exposure times with respect to the first exposure.

The remaining columns provide the sky-subtracted photometric counts for the target followed by each comparison star. The last column is always the sky column and represents the amount of background counts subtracted from the target's aperture. Sky counts are not provided for the comparison stars. The number of comparison star columns will reflect the original number of comparison stars loaded in the *ccd_hsp* or *MAESTRO* photometry files, even if they don't all get used to generate the divided light curve. However, if specific points were deleted from the divided light curve, they will also be removed from this file.

Below is an example .phot file, again showing the header followed by the first and last three rows of data:

```
# Object     = G117-B15A                    # Name of Object
# RA         = 09 24 15.27                  # Object Right Ascension
# Dec        = +35 16 51.3                  # Object Declination
# Telescope  = McDonald 2.1m                # Name of Telescope
# Instrument = ProEM                        # Name of Instrument
# TeleCode   = mcd2                         # Teledat Code Name
# Date       = 2018-01-26                   # Mid-Exp. UTC Start Date
# Time       = 05:06:01.500                 # Mid-Exp. UTC Start Time
# MJD        = 58144.212517361              # Mid Exposure UTC MJD Start
# Exptime    = 15.000000                    # Exposure Time (s)
# Filter     = BG40                         # Filter Name
# BJED       = 2458144.718681479            # Mid Exp. Barycentric Julian Date
# Barycorr   = True                         # Barycentric Corrections Applied?
# ApPhot     = ccd_hsp                      # Photometry Program
# OrigFile   = runbase6.                    # Source Photometry Filename
# ApRadius   = 6.00                         # Aperture Radius (pixels)
# Nkeep      = 1049                         # Number of points in light curve
# Ndelete    = 0                            # Number of points removed
# Columns: Time (s), Target, Comparisons, Sky
       0.000     192620    88577    25195    78351
          15.000     206569    94587    27371    76162
      30.000     204690    92457    28212    75914
      ...         ...       ...      ...      ...
   15690.173     214079    93761    27382    13946
   15705.173     175979    78464    22928    14297
   15720.174     166926    73647    22034    15337
```

3.1.3.6.3 The phot2lc_log File

The *phot2lc_log.txt* file saves information about the light curve extraction so that if you want to rerun phot2lc for an object, it will automatically apply the same changes that you had already made. This file is only generated if you have previously saved a light curve and includes information about which points had been deleted, what polynomial was used for detrending, and which comparison stars had been used for division. This can save large amounts of time for users when small modifications need to be made after the original light curve extraction.

In addition, this file stores each data point's raw timestamp, prior to barycentric corrections, so that the timing information is preserved. This is useful if phot2lc originally generated timestamps via the header keywords within a list of FITS files, but you would like to remove most of the FITS files post-extraction to save computer space. If a *phot2lc_log* file is present, phot2lc will preferentially use it to generate the time stamps upon execution.

An example *phot2lc_log* file is shown below, which shows the header and the first and last three rows of data. The first data column provides the raw mid-exposure times relative to the first exposure, and the second column is just a boolean value where 0 = Deleted and 1 = Kept.

```
#     OBJECT = G117-B15A
# POLYNOMIAL = 3,0,3.00,3.00
#      DTMID = 2018-01-26T05:06:01.500
#      COMPS = 1+2
#       TEXP = 15.000000
     0.000  1
    15.000  1
    30.000  1
    ...    ...
 15690.000  1
 15705.000  1
 15720.000  1
```

3.1.4 Barycentric Corrections

While most of what phot2lc does is relatively straightforward, the barycentric time corrections are perhaps the most important and easily mishandled part of this program. phot2lc uses the Astropy Time package to perform point-by-point barycentric time corrections which account for the observed object's sky coordinates, the Earth location of the observatory from which the observations were made, the date and time at which the observations were made, and the number of leap seconds that have occurred. If any of these four pieces of information are incorrect, the barycentric time corrections will be unreliable. The sources for each piece of information are listed below:

- **Object Coordinates** come from the stars.dat file
- **Observatory Location** is identified using the telescope code (e.g. mcd2, lco1, pjmo, etc.)
- **Observation Date-Time** comes from the image time stamps
- **Leap Seconds** are now automatically checked and updated by Astropy (v4.0 or later)

Common sources of timing errors include typos in the stars.dat file, using the wrong coordinate epoch (must be J2000!), using the wrong object name which would load the wrong coordinates from stars.dat, using the wrong telescope code for your reductions, using an outdated leap seconds file (hopefully not since Astropy auto-updates it now), or perhaps errors in the raw time stamps for your images. It is always worth double checking these pieces of information throughout your reduction process!

3.1.4.1 Timing Verifications

To verify that phot2lc is providing accurate barycentric corrections, a comparison with the corrections from other programs (WQED and UTC2BJD) has been performed for 73 different objects with a total of 138 light curves. Distributions of Barycentric Julian Date (BJD) differences between these programs are shown below, which are calculated using the mid-exposure time of the first data point for each light curve.

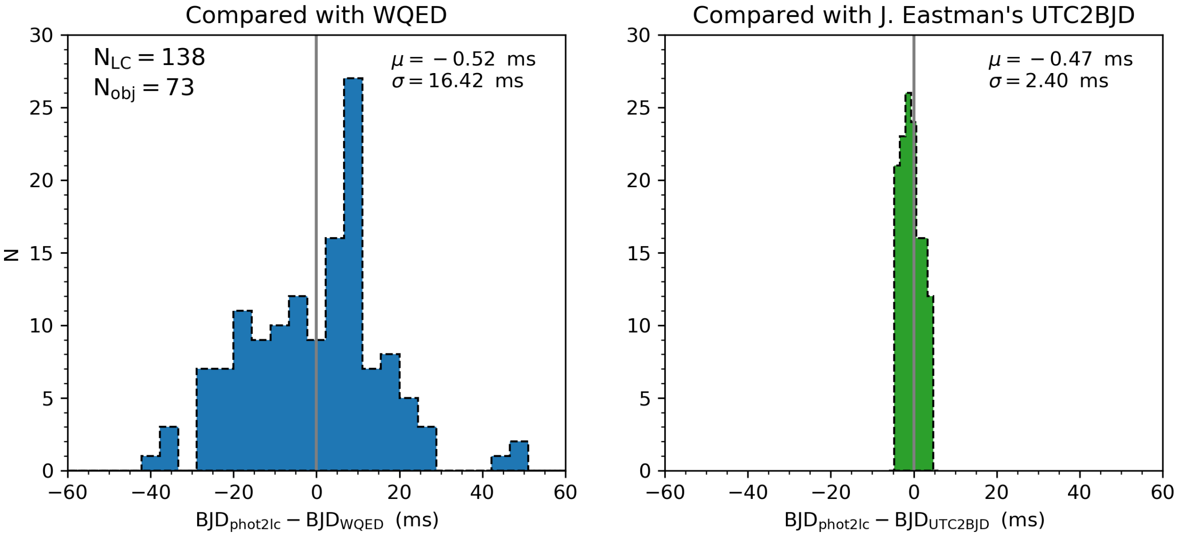


Fig. 1: Distributions of BJD differences between phot2lc, WQED, and UTC2BJD.

The distribution of BJD differences between phot2lc and WQED is much broader than with UTC2BJD, but WQED does not take into account the Earth location of the observatory used, whereas both phot2lc and UTC2BJD do. The excellent agreement with UTC2BJD suggests that taking the Earth location into account corrects most of the larger disagreements between phot2lc and WQED, which max out at around 50 milliseconds for this sample. While this is often much less than the timing uncertainties associated with image acquisition, the improved accuracy certainly doesn't hurt and with Astropy is a very easy correction to perform.

Another difference between phot2lc and WQED is the point-by-point correction method. WQED performs barycentric corrections by calculating a linear interpolation between the first data point's time stamp and a time stamp 0.5-days later. phot2lc, on the other hand, performs a barycentric correction for each point individually. This difference is expected to result in only minor changes in the timing corrections, often different by a few milliseconds or less, but can occasionally reach higher. In our sample, only one out of 73 objects exceeded a 10ms correction difference compared with WQED due to phot2lc's point-by-point method (see figure below). Again, this is much smaller than the typical timing uncertainties, but the additional timing accuracy certainly doesn't hurt.

3.1.5 Additional Tools

Besides the main light curve extraction program (phot2lc) and the configuration program (photconfig), the phot2lc package also comes with a program for welding individual light curve files together (weldlc), and a program for generating light curve + periodogram plots using the .lc output files. This section provides information on how to use these tools.

- *weldlc*
- *quicklook*

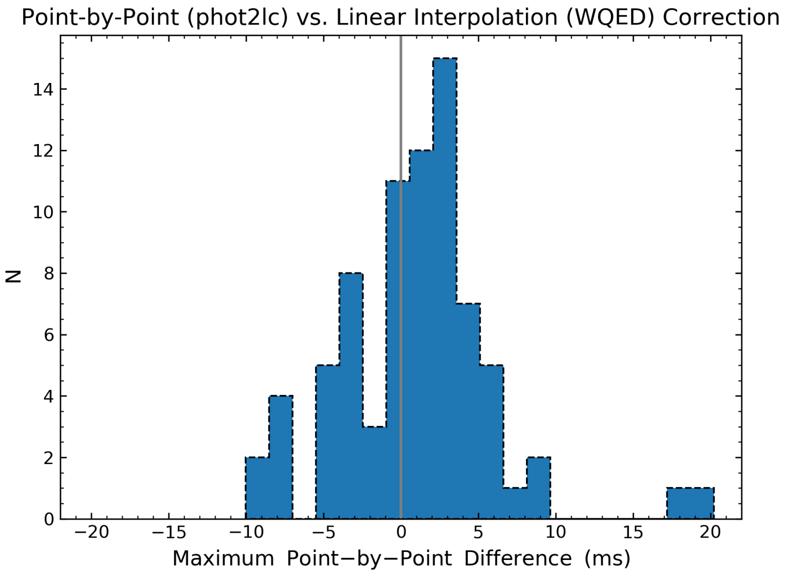


Fig. 2: Distribution of the largest difference in corrections per light curve when performing a point-by-point (phot2lc) versus linear interpolation (WQED) correction method.

3.1.5.1 weldlc

A program designed for combining individual phot2lc light curve (.lc) files into a single light curve file. This program behaves much like WQED's *weld* tool. Two command line options are available for this tool:

```
-f --infiles     Input files.
-b --bypass      Bypass the barycentric correction check and weld LC's anyways.
-o --outfile     Output filename.
```

wildlc will check the *barycorr* entries within each .lc file to make sure every light curve being welded has had barycentric time corrections applied. If one or more of the .lc files has barycorr = False, wildly will throw an error. If you know that your light curves are not all barycentric corrected and want to weld anyways, you can use the –bypass (-b) command to skip the check. An example usage would look like:

```
weldlc -f *.lc -o combined.lc
```

or

```
weldlc -f *.lc -o combined.lc --bypass
```

where **.lc* would provide as input every .lc file within the current folder, and *combined.lc* would be the name of the resulting file with the combined light curve. A welded light curve file will have the same three data columns as an original .lc file, but different header information. An example header is given below:

```
# Object      = G117-B15A                             # Name of Object
# RA          = 09 24 15.27                           # Object Right Ascension
# Dec         = +35 16 51.3                           # Object Declination
# Date        = 2018-01-26                            # Mid-Exposure TDB Date at T0
# Time        = 05:14:54.080                          # Mid-Exposure TDB Time at T0
# BJED        = 2458144.718681479                     # Mid-Exposure TDB JD at T0
# BaryCorr    = True                                  # Whether the times are barycentric␣
↪corrected
```

(continues on next page)

(continued from previous page)

```
# WeldNum    = 2                          # Number of files welded
# Npoints    = 3905                       # Number of data points
# Tspan      = 1.14824075                 # Time spanned by data (days)
# WeldDate   = 2021-02-10 12:19:45.245    # File creation date
# Columns    = T-mid (s), Rel. Flux, Rel. Flux Error
```

3.1.5.2 quicklook

A program designed for quick analysis of phot2lc light curves and their periodograms. quicklook is also a command line tool with the following command line options:

```
-f --files      Input file(s) to perform quicklook on.
-s --save       Whether to save the quicklook plot.
-p --prewhiten  Whether to perform a pre-whitening sequence.
-l --lower      Lower frequency limit for pre-whitening search (micro-Hertz,␣
↪default=500).
-u --upper      Upper frequency limit for pre-whitening search (micro-Hertz,␣
↪default=100000).
-n --num        Maximum number of pre-whitening iterations (default=10).
-w --wqedlc     Whether the input file(s) are from WQED.
```

The *-s*, *-p*, and *-w* options don't require an actual input. When they are called, they automatically pass a boolean value of TRUE to the quicklook program.

The only required command line input is *-f*, which you use to specify the file, or files, you wish to perform a quicklook for. Quicklook will always display the resulting plot, but will only save the plot if the *-s* option is specified. An example usage would be

```
quicklook -f *.lc
```

and the displayed plot would look like the following:

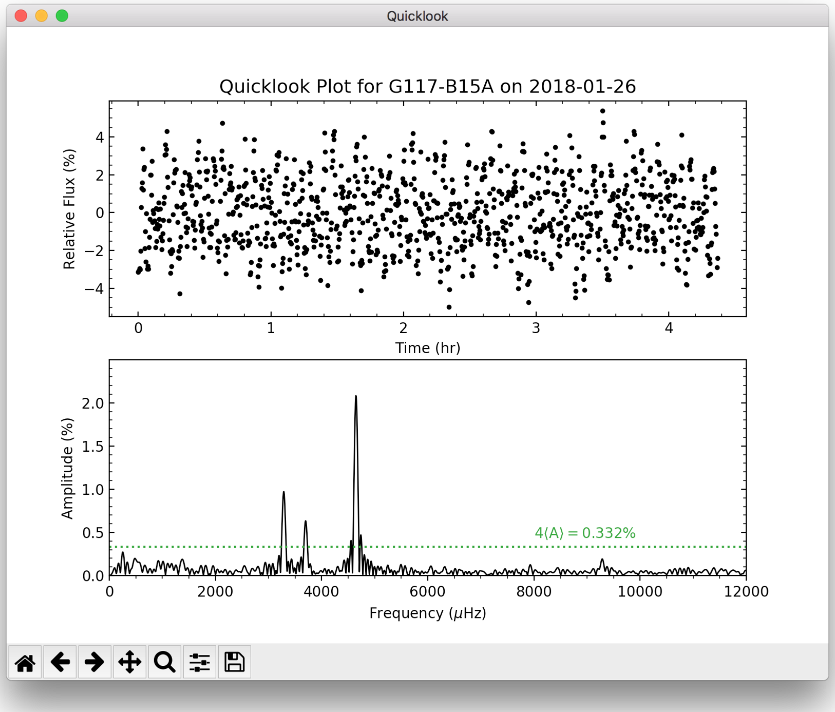
If you would like to perform a pre-whitening sequence to identify significant peaks in the periodogram, you need only specify the *-p* option. Pre-whitening is a procedure by which significant peaks in the periodogram are iteratively identified and then removed by subtracting a corresponding best-fit sinusoidal model from the light curve. Peaks are identified as significant if they rise above a significance threshold, which is defined in quicklook as four times the average periodogram amplitude, $\langle 4\text{A} \rangle$, between 500 and 12,000 μHz. If a significant peak is found, its height and center are used as the initial guesses for a sinusoidal model's amplitude and frequency, and a least squares fit (via LMFIT) is then performed to optimize those parameters. The resulting model is then subtracted from the light curve and a new periodogram of the residuals is calculated. This process repeats itself until no more significant peaks are found or until the maximum number of pre-whitening iterations set by *-n* is reached. Peaks are identified in order of decreasing amplitude.

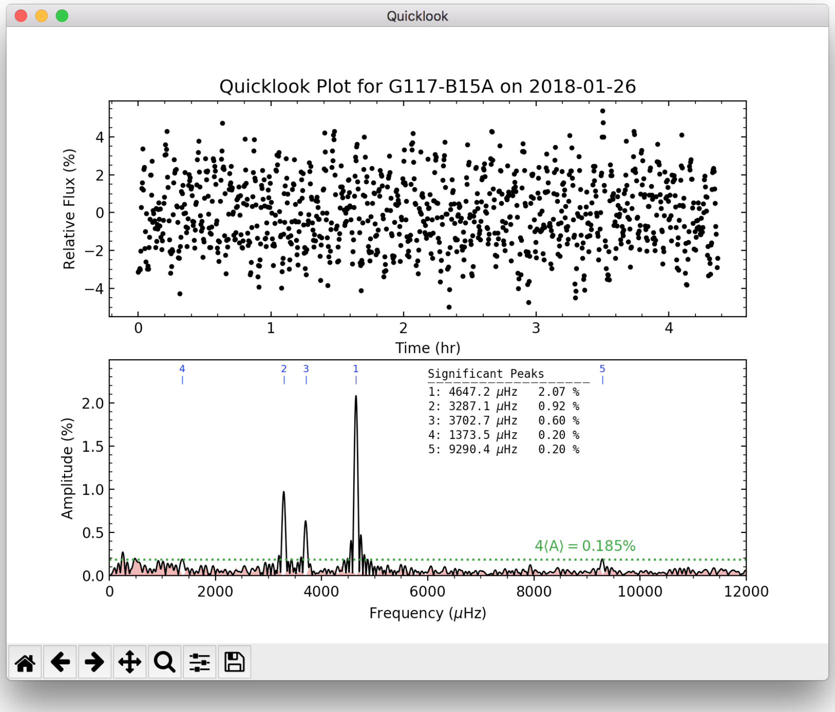
An example of using the pre-whitening option would look like:

```
quicklook -f *.lc -p
```

The resulting display (shown below) now has numbered markers labeling the locations of significant peaks, along with a table showing the best-fit frequency and amplitude of each peak from LMFIT. The black periodogram is before pre-whitening, while the red-shaded periodogram is after pre-whitening. The $\langle 4\text{A} \rangle$ threshold is representative of the red pre-whitened periodogram.

When the *-p* option is specified, the *-l*, *-u*, and *-n* options also become available. In the plot above, if for some reason you wanted to exclude peaks 4 and 5 from the pre-whitening procedure, you could do this by either limiting the frequency search range or by limiting the number of pre-whitening iterations. The commands would look like





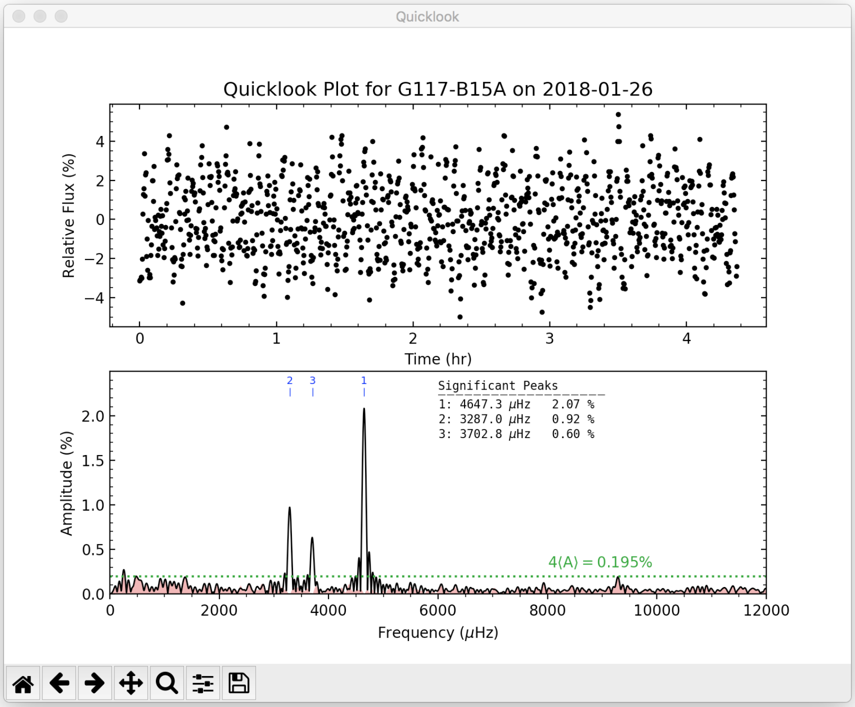
```
Significant Peaks
-----------------
1: 4647.2 μHz   2.07 %
2: 3287.1 μHz   0.92 %
3: 3702.7 μHz   0.60 %
4: 1373.5 μHz   0.20 %
5: 9290.4 μHz   0.20 %
```

```
quicklook -f *.lc -p -l 2000 -u 6000
```

or

```
quicklook -f *.lc -p -n 3
```

Either way, the resulting plot would look like:



```
Significant Peaks
--------------------
1: 4647.3 μHz   2.07 %
2: 3287.0 μHz   0.92 %
3: 3702.8 μHz   0.60 %
```

Lastly, if you wish to use the quicklook function to analyze WQED light curve files, this is possible if you just specify the *-w* option:

```
quicklook -f *.lc1 -w
```

3.1.6 License

3.1.7 Contact

Questions? Please contact zvanderbosch@gmail.com

3.1.8 Need Help

If you're having trouble, please email zvanderbosch@gmail.com