Tools for Transit and Radial Velocity Modelling and Analysis

Hans J. Deeg

Abstract An overview over analysis or modelling tools related to the radial velocity and transit methods for exoplanet detection and characterization is given, with over 40 software tools presented. Tools that cover related subjects, such as physical exoplanet modelling, stellar activity, limb darkening, or stellar parameter estimates are included as well. All presented tools have been verified to be publicly available, although inoperative software repositories were not uncommon. This issue needs to be addressed, with some of the existing solutions for long-term storage to be considered.

Introduction

Tools for the modelling or analysis of data play an important role in exoplanet research. In this chapter we give an overview over such tools currently available to the research community. Given that an overwhelming majority of current planet detection and characterisation work is based on radial velocity or transit observations, which furthermore are often used in combination, only tools involving either or both of these methods are presented here. This was further limited to tools for the interpretation of reduced data, meaning that tools for the extraction of photometry or spectroscopy from raw data were not considered, and neither were tools for transit searches, nor generic fitting or analysis tools (see also the chapter 'Bayesian methods for exoplanet science'). It was furthermore required that the tools to be listed are publicly available (but not necessarily for free) in the form of concrete computer codes – that is, models or algorithms without an associated code are not included, with some exceptions of fundamental algorithms. Some reasonable level of docu-

Hans J. Deeg
Instituto de Astrofísica de Canarias, C. Via Lactea S/N, E-38205 La Laguna, Tenerife, Spain; Universidad de La Laguna, Dept. de Astrofísica, E-38206 La Laguna, Tenerife, Spain; e-mail: hdeeg@iac.es

mentation about the tool's usage, be it within the software distribution or in external documents (articles, websites) has also been called for. It was also required that the tools are reasonably generic - that is, that they are not limited to a very specific application or research project. In the overview that follows, tools for the analysis of Radial velocities (RV) and for photometric transits are followed by tools that combine both methods and by tools concerned with further issues, like orbital dynamics, stellar activity, exoplanet atmospheres, etc. The software tools listed in this chapter have been compiled from several sources:

- The *Astrophysics Source Code Library* at `http://ascl.net`. At present, ASCL is the most systematic compilation of astronomy software. It does not contain the source codes, but short descriptions with links to their sources and to publications describing the tool. ASCL entries are listed in the Astrophysics Data System (ADS) and authors of astronomy related codes are encouraged to contact ASCL for their inclusion.

- The *IDL Astronomy User's Library* at the NASA Goddard Space Flight Center, at `http://idlastro.gsfc.nasa.gov/`. It contains for one codes in the IDL[1] language that have been deposited in this library, and for another, contains a list with links to other astronomy IDL software sites.

- The list of *Kepler/K2 software tools* at `https://keplerscience.arc.nasa.gov/software.html`. Whereas most of them are specific to the Kepler and K2 missions, some tools with a wider scope have been included in this chapter.

- Numerous tools were also included through feedback from the Exoplanets Mailing List at Grenoble University[2], following an enquiry by the author in February 2017.

A few more considerations for the use of this work:
Unless otherwise noted, the listed tools are expected to function under UNIX-based operating systems such as Linux or MacOS. Given that most tools are distributed as source codes without complex system interactions, porting to other systems should usually be straightforward if compilers and libraries can be found.
In the following listing, we begin with sections dedicated to the modelling of radial velocity and transit data, and continue with tools that combine such data or that treat more specific aspects. For all entries, a link to the code's website (further referred to as 'code site') is provided and if possible, the code's language and the year of the last update is indicated, revised between April and May 2017. During the course of this work, issues with the availability of the codes were not uncommon, however. This topic is treated in the epilogue.

[1] IDL is a registered trademark of Harris Corporation.

[2] `https://listes.obs.ujf-grenoble.fr/sympa/info/exoplanets`

Analysis of radial velocity data

In this section, tools for the modelling or analysis of radial velocity (RV) data obtained in exoplanet research are given. We also note that tools for the analysis of RV data are available within the Data Analysis Center for Exoplanets (DACE) of Geneva Observatory, see entry.

Systemic `http://www.stefanom.org/systemic/`
One of the earliest tools for the fitting RV's with exoplanet signals, Systemic (Meschiari et al. 2009, 2012) exists in two distinct versions. The principal tool is the Systemic Console 2, with an interactive GUI that permits RV fitting not only of Keplerian curves but also orbital integration by various methods and contains dedicated algorithms (Bootstrap, MCMC) for error estimation. The code is written in C and R and is also available as an executable for MacOS. Systemic Life provides a reduced set of Systemic's features in an online web-interface. It contains a selection of preloaded RV data, permitting their visualisation and fitting and is primarily intended for educational use. Last update: 2015.

RVLIN `http://exoplanets.org/old/code/`
RVlin (Wright and Howard 2012) is a set of IDL routines to fit an arbitrary number of Keplerian curves to radial velocity data. It can handle data from multiple telescope, solving for the RV-offset, constraints on parameters (e.g. period, eccentricity), and incorporate transit timing data. It considers only strictly periodic Keplerian orbits and uses a Levenberg-Marquard (LM) fitter. An associated parameter uncertainty estimator using the bootstrapping method, BOOTTRAN, (Wang and Wright 2012), also written in IDL, is available as on the same site. Last updates: 2015(RVLIN) and 2012 (BOOTTRAN).

RVFIT `http://www.cefca.es/people/~riglesias/rvfit.html`
Iglesias-Marzoa et al. (2015a,b) present RVFIT, which fits non-precessing Keplerian RV curves for double-line and single-line binary stars or exoplanets. It uses an optimised adaptive simulated annealing algorithm, which is a heuristic algorithm to minimise functions in multiparametric spaces. RVFIT is written in IDL and includes also an MCMC sampler to determine errors around the best solution. Last update: 2015.

RadVel
`https://github.com/California-Planet-Search/radvel`
This package is labelled as a 'General Toolkit for Modeling Radial Velocities' (Fulton and Petigura, in prep.). It is intended for the RV fitting of Keplerian orbits and written in Python, with an MCMC sampler based on the emcee code and provides multiple graphical outputs. Last update 2017.

PlanetPack `http://sourceforge.net/projects/planetpack`
This package (Baluev 2013a,b) facilitates the analysis of RV data with unknown jitter, employing multiple Keplerian orbits or dynamical N-body simulations. PlanetPack is a command-line interpreter written in c++ that can run either in an inter-

active mode or in a batch mode for script interpretation. Last update 2015. Further updates have been announced (Baluev 2015)including the Rossiter effect, transit fitting and the analysis of transit time variations, but are not released yet.

PSOAP `https://github.com/iancze/PSOAP`
A new technique for the disentangling of time series spectra with Gaussian processes is presented by Czekala et al. (2017), where the posterior distributions of the orbital parameters and intrinsic, rest-frame stellar spectra of single or double-lined binaries are explored simultaneously without the invocation of cross-correlation templates. This technique is implemented in PSOAP (Precision Spectroscopic Orbits A-Parametrically), which is available as an open source Python package. Last update 2017.

PyORBIT `https://github.com/LucaMalavolta/PyORBIT`
This is a package for the analysis of RV time series and ancillary data (photometry, activity indicators, etc) to simultaneously characterize the orbital parameters of exoplanets and noise from the activity of the host star (Malavolta 2016). For dynamical orbit integration, it requires the installation of either the TRADES or TTVFast codes (see corresponding entries). Stellar activity can be modelled in various ways and systematics from several instruments may be considered. The code, written in Python, is also prepared for the incorporation of new methods for stellar activity modelling or for parameter estimations. Last update 2017.

Analysis of transiting planets

In this section we include tools that are useful for the analysis of light curves involving transiting planets. Some more tools involving the analysis of transiting planets are found in the sections entitled 'Tools centred on stellar activity and star spots' and 'Combined analysis of RV and transit data'. We do not include tools to prepare data for modelling or for the search for transits.

Modelling and fitting of exoplanet transit light curves

The tools in this section are primarily used to fit light curves of transit-events. For an introduction to the solutions and parameters that can be obtained from transit fits, we refer to the seminal paper by Seager and Mallén-Ornelas (2003). For analytical solutions to the modelling of transits including limb darkening we refer to the transit models by Mandel and Agol (2002, see below) and by Giménez (2006, implemented by PyTransit -see entry- and others).

Mandel and Agol transit model
`http://faculty.washington.edu/agol/transit.html`

Mandel and Agol (2002) presented an analytical solution to model transit light curves with quadratic limb darkening coefficients. This algorithm provides only the modelling of the light curve of a single transit with a minimal parameter set. This model is straightforward to integrate into more complex approaches and is likely the most widely used transit modeller. Implementations in several languages (IDL, Fortran, Python, Matlab) are available though the given website. This model has also been implemented in several other packages, such as TAP, ktransit, EXOFAST, the latter one with a much faster IDL implementation (see corresponding entries). Last update of the authors' original version in 2002.

ktransit `https://github.com/mrtommyb/ktransit`
This is a basic exoplanet transit modeling tool that permits a full orbital model with varying eccentricities. It contains also model fitter using the LM algorithm. ktransit is written in Python while the underlying model is a Fortran implementation of the Mandel and Agol (2002) limb darkened transit model. Last update 2017.

TAP `https://github.com/zgazak/TAP`
Gazak et al. (2011, 2012) created the Transit Analysis Package. TAP is designed to input transit light curves and fit them using the Mandel and Agol (2002) model. The software is able to simultaneously analyse multiple transits observed in different conditions (instrument, filter, weather, etc) and perform an interpretation using a Bayesian MCMC analysis. The graphical interface and its option for interactive model fitting makes TAP also very suitable as a tool for classroom activities. TAP is written in IDL; a compiled version 'tap.sav' can also be found on the web, which can be executed with the IDL Virtual Machine for which free licences are available. We note that a manual of TAP has been published as a preprint of Gazak et al. (2012). Last update 2015.

TAC-maker `http://astro.shu-bg.net/software/TAC-maker`
The 'Transit Analytical Curve maker' (Kjurkchieva et al. 2013a,b) is a tool to generate transit models either with darkening profiles that may either be the commonly used 'laws' or be defined by the user. TAC-maker uses a semi-analytical approach with a numerical computation of the integrals defining the overlapping zones in transits, which can be made to arbitrary precision. The code is accessible through a graphical user interface and written in Python; last update 2014.

batman `https://github.com/lkreidberg/batman`
Kreidberg (2015) presents batman (BAsic Transit Model cAlculatioN in Python), a Python package for the modelling of exoplanet transit light curves. The batman package supports calculation of light curves for any radially symmetric stellar limb darkening law, using an integration algorithm for models that cannot be quickly calculated analytically. The code uses C extension modules to speed up model calculation and is parallelized with OpenMP. This code can be set to calculate with accuracies better than 0.001 ppm, with the aim to analyse very precise transit light curves as may be anticipated from upcoming facilities. The batman package is open source and written in Python with some C extension modules. Last update 2017.

PyTransit https://github.com/hpparvi/PyTransit
PyTransit (Parviainen 2015a,b; Parviainen and Borsato 2016) is a package providing optimized versions of the Mandel & Agol (see corresponding entry) and Giménez (2006) transit models. The models are implemented natively in Fortran with OpenMP parallelization, and are accessed by an easy-to-use object-oriented Python interface. The code implements optimizations that separate the effects from stellar limb darkening from the geometric effects, which allows for efficient model evaluation for multicolor observations and transmission spectroscopy; and the models implement a built-in supersampling to account for extended exposure times for analysis of Kepler-like light curves. The code is written in Python, last update 2017.

Modelling of multi-body occultations, moons, planetary rings

Here we present some tools for more complex transit scenarios that may include other bodies besides planets. For 3-body occultations involving a planet with a moon, the analytical LUNA algorithm by Kipping (2011) is also of note, although no mention of its availability as a code is made.

mttr `http://szofi.elte.hu/~apal/utils/astro/mttr/`
Pál (2012) describes an algorithm and deduce the related mathematical formulae for the computation of observed fluxes in stellar and planetary systems with an arbitrary number of bodies that are part of a transit or occultation event. The algorithm divides multi-body eclipsing configurations into several arcs whose emission properties are calculated separately; it does not have any limits or constraints on the geometry and can be applied for almost all of the available limb darkening models as well. The tool is written in C as a plugin for the fitter-program 'lfit.c' described in Pál (2009). Last update 2011

Exorings `https://github.com/mkenworthy/exorings`
Kenworthy and Mamajek (2015a,b) provide a set of tools for the detailed modelling, displaying, and fitting of eclipses generated by giant exoplanetary rings. These rings are assumed to be much bigger than the eclipsed star, and fill the Hill sphere of an orbiting planet. The code is written in Python, last update 2015.

UTM `https://tiny.cc/utm-ufit`
The Universal Transit Modeller (Deeg 2009, 2014) is a highly flexible light-curve simulator for all kinds of mutual transits or eclipses between arbitrary numbers or stars, planets, moons or planetary rings, on Keplerian orbits that may be hierarchical and include osculating elements. All common limb darkening laws are provided, with the option to model starspots as well. Resampling for the analysis of slowly sampled light curves (Kipping 2010) and the modelling of various types of out-of-transit brightness variations is also possible. UTM provides configurable textual and graphical output, with optional animations of transit events. It uses scaled pixelised object representations that permit precise calculations for occulters of any

size-ratio (e.g. Earth-Sun transits). The associated UFIT (Universal Fitter) code enables the fitting of any of UTM's input parameters, either with the amoeba algorithm or MCMC routines based on EXOFAST (see entry). Tools to generate MCMC parameter distributions or covariances are also included. Both UTM and UFIT are driven by setup-files with a simple syntax; UFIT also provides them as output for further fitting attempts. Pre-processor macros enable both programs to ingest arbitrary sets of input parameters for modelling or fitting. UTM/UFIT is written in IDL; last update 2016.

Tools for eclipsing binary fitting

Here we list several codes that were mainly developed for the fitting of eclipsing binary light curves but which are also used for the fitting of transiting planets, replacing one of the stellar components by a planet.

ellc
`http://cdsarc.u-strasbg.fr/viz-bin/qcat?J/A+A/591/A111`
Maxted (2016) presents ELLC - a fast, flexible light curve model for detached eclipsing binary stars and transiting exoplanets. It may include the effects of star spots, Doppler boosting and light-travel time within binaries with eccentric orbits. It models stars as triaxial ellipsoids; the model can also be used to calculate the Rossiter-McLaughlin effect on radial velocities during eclipses. *ellc* is claimed to be fast enough to enable the use of Monte Carlo methods for data analysis and model testing. The light curve model is implemented as Fortran subroutines that are called from a user interface written in Python. Last update 2016.

JKTEBOP
`http://www.astro.keele.ac.uk/~jkt/codes/jktebop.html`
This code (Southworth et al. 2004; Southworth 2013, further references in code site) is primarily intended to fit detached eclipsing binary stars in order to derive radii and other parameters, but has also been used widely for the fitting of exoplanet transits. Both light curves and radial velocities may be fitted and all common limb darkening laws are provided while times of minimum light, spectroscopic light ratios, third light values, and orbital eccentricity information can be included as constraints to its solutions. JKTEBOP models the two components as spheres for the eclipse shapes and as biaxial spheroids for the calculation of the reflection and ellipsoidal effects. It uses the Levenberg-Marquardt optimisation and Monte Carlo or bootstrapping error analysis algorithms are available. All input and output is through text files but an interactive interface, ALCFIT written by P. Maxted, is available though the code's site. Both codes are written in Fortran; JKTEBOP's last update was in 2014.

PHOEBE 2.0 `http://phoebe-project.org/`
PHysics Of Eclipsing BinariEs (Prša et al. 2016) permits the computation of light and RV curves of all types of eclipsing systems, including phase curves, light time

effects etc. PHOEBE 2.0 is a complete rewrite of prior versions and uses triangulation as a discretization algorithm of stellar surfaces. Other geometric bodies (i.e., planets, accretion disks, artificial structures) may also be modelled with the same formalism. It is stated as having a steep learning curve but an ample set of documentation is available on the code's site. PHOEBE's backend is written in C while Python is used for the frontend; last update in 2017.

Binary Maker 3 `http://www.binarymaker.com`
This is a commercial program by Contact Software for the modelling of binary systems in much detail, including star spots, eccentric orbits, asynchronous rotation etc, permiting also comparison between models and data of light curves and radial velocities (Bradstreet 2005). Its outstanding features are an interactive interface and graphical outputs that make it suitable for educational activities. Binary Maker is written in java and available for all common desktop systems.

Tools to analyse transmission spectra

Besides the following entry for Ex-Transmit, Line et al. (2013) present the CaltecH Inverse ModEling and Retrieval Algorithm (CHIMERA), however without mentioning a concrete implementation.

Exo-Transmit `https://github.com/elizakempton/Exo_Transmit`
Kempton et al. (2017) present Exo-Transmit as "A software package to calculate exoplanet transmission spectra for planets of varied composition. The code is designed to generate spectra of planets with a wide range of atmospheric composition, temperature, surface gravity, and size, and is therefore applicable to exoplanets ranging in mass and size from hot Jupiters down to rocky super-Earths. Spectra can be generated with or without clouds or hazes with options to (1) include an optically thick cloud deck at a user-specified atmospheric pressure or (2) to augment the nominal Rayleigh scattering by a user-specified factor." Exo-Transmit is written in C; last update 2017.

Combined analysis of RV and transit data

The tools listed here combine the analysis of RVs with a detailed analysis of transit lightcurves. For tools focussed on the analysis of RVs and transit *times*, see the next section.

EXOFAST
`http://astroutils.astronomy.ohio-state.edu/exofast`
Eastman et al. (2012, 2013) created a widely used suite of over 30 IDL routines designed to fit exoplanetary transits and radial velocity variations simul-

taneously or separately and characterize the parameter uncertainties and covariances with a differential evolution MCMC method. EXOFAST uses the Mandel and Agol (2002) algorithm as basic transit modeller, providing a fast implementation in IDL (EXOFAST_OCCULTQUAD). In the code site, several further routines related to precise Heliocentric and Barycentric Julian Date calculation (Eastman et al. 2010) are available, as well as an online applet of a subset of EXOFAST. The main code's last update was in 2014. A complimentary version, EXOFASTv2 (`https://github.com/jdeast/EXOFASTv2` is being advanced in 2017. It is aimed at being more powerful and flexible, with fits for multiple planets, multiple transits that may be in different bands, treatment of TTVs, inclusion of evolutionary models and SEDs and more. Another related project, ExofastGPU (by E. Ford at `https://github.com/AstroGPU/ExofastGpu`) is a collection of parallelized functions for transit light curve modelling in C++ , which take advantage of both multi-core architectures and GPU accelerators.

exonailer `https://github.com/nespinoza/exonailer`
This tool is annouced as an easy-to-use code that allows the fitting of exoplanet transit lightcurves, radial velocities or both (Espinoza et al. 2016). It implements the batman code (see entry) which permits the use of different limb darkening laws efficiently. The joint RV and light curve analysis is performed using the emcee MCMC ensemble sampler (Foreman-Mackey et al. 2013). exonailer is written in Python; last update 2016

TLCM
The Transit Light Curve Modeller (Csizmadia in prep., with a short description given in Smith et al. 2017) is a code for the analysis of photometric time series of transits with detailed modelling of out-of-transit light variations, including beaming-, reflection- and ellipsoidal-effects, as well as user-defined baselines and features. TLCM permits joint light curve and radial velocity fits, that may include the Rossiter-MacLaughlin effect as well as RV drifts or offsets by different instruments. TLCM may be used to analyse eclipses of both exoplanet and well-detached binary systems. The code is based on a combination of a genetic algorithm and simulated annealing; it is also able to manage red-noise effects via wavelet analysis. TLCM is written in IDL, last update 2017.

Tools centered on orbital dynamics

Here we collect tools with a focus on orbital dynamics, mainly with the aim to evaluate orbital period variations that express themselves as transit time variations (TTV) and perturbations in radial velocities. Further related models and algorithms, but without public codes available, are formulae that compute perturbations to orbital elements to high order (Nesvorný 2009); analytic expressions for the TTVs including effects of first- and second- order resonances (Hadden and Lithwick 2016); solutions to eclipse time variations of stellar triple systems (Carter et al. 2011; Borkovits

et al. 2015); photo-dynamical models combining orbital integration with an analysis of transit curves and radial velocities (e.g. Almenara et al. 2015; Barros et al. 2015). In the framework of circumbinary planets, photo-dynamical models are described in e.g. Kostov et al. (2013, 2014); Welsh et al. (2015). We also mention Swarm-NG (Dindar et al. 2012, 2013), a library for parallel n-body integrations harnessing the power of GPUs with a focus on simulations of planetary systems.

TRADES `https://github.com/lucaborsato/trades`
'TRAnsits and Dynamics of Exoplanetary Systems' (Borsato et al. 2014; Borsato 2016) simultaneously fits observed radial velocities and sets of transit times to determine the orbital parameters of exoplanetary systems from observational data. It is implemented as a dynamical simulator for N-body systems that also fits the available data during orbital integration and determines the best combination of the orbital parameters using grid search, χ^2 minimization, genetic algorithms, particle swarm optimization, and bootstrap analysis. TRADES is implemented in Fortran 90 with OpenMP for parallel computation; last update 2017.

TTVFast `http://github.com/kdeck/TTVFast`
This code (Deck et al. 2014b,a) efficiently calculates transit times for n-planet systems and the corresponding radial velocities. The code uses a symplectic integrator with a Keplerian interpolator for the calculation of transit times of weakly perturbed Keplerian orbits around single stars. The code has been tuned for speed, using $\approx$15 times less steps than Bulirsch-Stoer based integrators, and not for accuracy of the calculated times; the desired time-steps can however be tuned for any system. For data fitting, this code has to be called from separate fitting routines. Both Fortran and C versions are available, as well as a Python wrapper (ttvfast-Python) by S. Walker at `https://github.com/mindriot101/ttvfast-Python`; last update in 2017.

TTVFaster `https://github.com/ericagol/TTVFaster`
Agol and Deck (2016a,b) derive an analytic formula for TTVs accurate to first order in eccentricity and planet – star mass ratio, valid for nearly plane-parallel transiting planets. Their formula is based on perturbation theory, given that the eccentricities of many planets exhibiting TTVs are small. With its higher computational efficiency over numerical solutions, quoted as 400 times faster than TTVFast, it is is expected to be used in combination with MCMC methods for fits to suitable datasets, both from transits and from radial velocities, while its analytical nature helps to pinpoint TTVs' physical origins. TTVFaster has been implemented in IDL, Julia, Python and C; last update 2016.

Tools centred on stellar activity and starspots

In this section we collect tools that model effects of stellar activity in data from exoplanets.

SOAP 2.0 `http://www.astro.up.pt/resources/soap2`
Dumusque et al. (2014) present an updated version of the 'Spot Oscillation And Planet' code (Boisse et al. 2012). For a version of SOAP adapted to the modelling of planetary transits, see the entry on SOAP-T. In SOAP 2.0, activity from star spots and plagues and their impact on the photometric and RV measurement of a star is modelled, including the bisector span. It can also be used to correct RV variations when a they are dominated by a major active region. SOAP 2.0 can be invoked remotely in its website through the upload of a parameter file, but may also be downloaded as a Python code; last updated in 2015.

SOAP-T `http://www.astro.up.pt/resources/soap-t`
Oshagh et al. (2013) have created SOAP-T for the modelling of photometric and RV signals of a system consisting of a rotating spotted star and a transiting planet, based on the 'Spot Oscillation And Planet' code (Boisse et al. 2012, see also entry for SOAP 2.0). SOAP-T can be used to study activity-induced anomalies inside transit light curves and the Rossiter-McLaughlin effect, to better constrain orbital configurations and properties of planetary systems and the active zones of their host stars. The tool is implemented as a web-interface that requires the upload of a parameter file.

KSint `http://eduscisoft.com/KSINT/`
This program employs Kelvin-Stokes integration (Montalto et al. 2014, 2015) to solve for mutual photometric effects produced by planets and spots or spot regions, permitting the construction of complex spot shapes. It identifies integrable and non integrable arcs on the objects profiles and analytically calculates the solution. KSint provides the solution up to the second degree of the limb darkening law. It is distributed as a Fortran 95 code, last updated in 2014.

SPOTROD `https://github.com/bencebeky/spotrod`
Béky (2014); Béky et al. (2014) present SPOTROD, a semi-analytic model for transits of spotted stars, related to the earlier code *macula* by Kipping (2012) for the modelling of out of transit brightness variations of spotted stars. SPOTROD generates models for planetary transits of stars with an arbitrary limb darkening law and a number of homogeneous, circular spots on their surface. The core code is written in C, with an API in Python. Last update 2016.

PRISM
Tregloan-Reed et al. (2013, 2015) present PRISM, for the modelling of transits affected by star-starspots, together with GEMC, a hybrid optimization code combining MCMC and a genetic algorithm based on the Differential Evolution Markov Chain (DE-MC) by Ter Braak (2006). PRISM uses pixelated representions of star and planet for the modelling of transit, quadratic limb darkening and starspot. Both programs are written in IDL and available on request from `j.j.tregloan.reed@gmail.com` (the link given in Tregloan-Reed et al. (2015) is no longer maintained); last update 2016.

Tools for the evaluation of other aspects of exoplanet light curves

The two tools presented are for the evaluation of noises in light curves and for the modelling of stellar ellipsoidal variations and planetary phase functions.

MC^3 `https://github.com/pcubillos/MCcubed`
'Multi-Core Markov-Chain Monte Carlo', also denoted as MCCubed, is a general model-fitting tool (Cubillos et al. 2017). MC^3 provides Levenberg-Marquardt optimization and differential-evolution MCMC routines for user-defined models to estimate best-fitting model parameters and their credible regions. It provides also a time-averaging routine for correlated-noise analysis, which Cubillos et al. (2017) apply to exoplanet light curves. MC^3 is implemented as a Python/C code; last update 2017.

EVIL-MC `http://www.lpl.arizona.edu/~bjackson/idl_code`
Jackson et al. (2012) have developed developed the 'Ellipsoidal Variations Induced by a Low-Mass Companion' (EVIL-MC) code to calculate stellar ellipsoidal variations and planetary phase functions, with application to Kepler and CoRoT planets. The code is in IDL, last update 2012.

Transit probabilities and validation

Besides the entry for VESPA below, for exoplanet validation we also note the codes BLENDER (Torres et al. 2011) and PASTIS (Díaz et al. 2014). Both are for the detailed evaluation of individual planet candidates and require rather complex input; neither of them is publicly distributed. For the search and vetting of Kepler/K2 candidates, there is also the publicly available K2PS pipeline (Pope et al. 2016; Parviainen et al. 2016).

CORBITS `https://github.com/jbrakensiek/CORBITS`
Brakensiek and Ragozzine (2016b,a) have developed the 'Computed Occurrence of Revolving Bodies for the Investigation of Transiting Systems', which analytically calculates the transit probability of any combination of planets in multi-planet systems. They claim this algorithm to be more accurate and orders of magnitude faster than previous ones, based on comparisons with Monte Carlo simulations. CORBITS is written in C++ with associated routines in Python and R; last update 2016.

VESPA `https://github.com/timothydmorton/VESPA`
Morton (2012, 2015) developed their code 'Validation of Exoplanet Signals using a Probabilistic Algorithm (VESPA)' for the calculation of false positive probabilities for transit signals. In contrary to the more complex validation codes mentioned above, VESPA uses as input only some parameters of transit lightcurves and host stars. Together with informed assumptions about field star and multiple-star properties, this makes it suitable for the bulk-processing of lightcurves. VESPA is written in Python, last update 2016.

Tools for physical exoplanet modelling

Currently, we are only aware of some tools that model exoplanet atmospheres.

BART https://github.com/exosports/BART
The 'Bayesian Atmospheric Radiative Transfer code' is project led by J. Harrington (Cubillos 2016; Blecic 2016; Cubillos et al. 2016) that implements a Bayesian radiative-transfer scheme for retrieving parameters from spectra of planetary atmospheres. A thermochemical-equilibrium code, a one-dimensional line-by-line radiative-transfer code, and an MCMC statistical module are combined to constrain the atmospheric temperature and chemical-abundance profiles of exoplanets. BART is written in C and Python, last update 2016. Cubillos and collaborators are also working on a new version 'Pyrat Bay', see `http://pcubillos.github.io/pyratbay/`

transit `https://github.com/exosports/transit`
This radiative-transfer code calculates the transmission or emission spectrum of a planetary atmosphere with application to extrasolar-planet transit and eclipse observations, respectively (Rojo 2006; Cubillos 2016; Blecic 2016). *Transit* computes the spectra by solving the one-dimensional line-by-line radiative-transfer equation for an atmospheric model. The code is written in C; last update 2017.

TEA https://github.com/dzesmin/TEA
The 'Thermochemical Equilibrium Abundances' code is used to calculate gaseous molecular abundances under thermochemical equilibrium conditions, for a particular temperature and pressure or a list of temperature-pressure pairs (Blecic et al. 2015, 2016). TEA uses 84 elemental species and thermodynamical data for more then 600 gaseous molecular species, and may adopt any initial elemental abundances. The code was tested against CEA (Chemical Equilibrium with Applications, `http://www.grc.nasa.gov/WWW/CEAWeb`) and has been applied to models of several hot Jupiters, producing expected results. TEA is written in Python; last update 2017.

Tools to analyse or model central stars

For the characterisation of extrasolar planets or planet systems, a detailed knowledge about the central stars is necessary. Below, we first some tools that are commonly used in the exoplanet community for the precise determination of stellar parameters from spectral data. This is followed by several tools that permit the modelling or evaluation of limb darkening effects.

SME `http://www.stsci.edu/~valenti/sme.html`
The Spectroscopy Made Easy package fits an observed high-resolution stellar spectrum against synthetic spectra from the Vienna Atomic Line Database (VALD) to determine stellar parameters (Piskunov and Valenti 2017; Valenti and Piskunov

1996, 2012). A nonlinear least squares algorithm is used to solve for any subset of allowed parameters, which include atomic data (log *gf* and van der Waals damping constants), model atmosphere specifications (T_{eff}, log *g*), elemental abundances, and radial, turbulent, and rotational velocities. LTE synthesis software handles discontiguous spectral intervals and complex atomic blends. The synthetic spectra from VALD are available through the code's site. SME is written in IDL, last update 2016.

VWA `https://sites.google.com/site/vikingpowersoftware/`
The Versatile Wavelength Analysis package was designed to analyse stellar spectra of main sequence and giant FGK-type stars by iterative fitting of calculated synthetic spectra. It automatically selects the least blended lines from the Vienna Atomic Line Database (VALD), and consequently adjusts the abundance in order to find the best match between the calculated and observed spectra. It is originally described in Bruntt et al. (2002). VWA was used to characterise many of the principal targets of the COROT satellite mission, e.g. Bruntt (2009); Bruntt et al. (2010). The code's site contains also the needed spectral libraries. VWA is written in IDL; last update 2015.

FASMA `http://www.iastro.pt/fasma`
'Fast Analysis of Spectra Made Automatically' (Andreasen et al. 2017) employes the method of equivalent width (EW) measures to derive stellar parameters as well as elemental abundances. High-resolution spectra with high S/N are recommended as input data. FASMA is implemented as a web-tool, although conversion to a standalone application that performs remote requests to the code's site is possible.

Tools to evaluate limb darkening

LDTk `https://github.com/hpparvi/ldtk`
Parviainen and Aigrain (2015) present the 'Limb Darkening Toolkit' that automates the calculation of custom stellar limb darkening (LD) profiles and model-specific limb darkening coefficients using the library of PHOENIX-generated specific intensity spectra by Husser et al 2013. The aim of the package is to facilitate exoplanet transit light curve modelling, especially transmission spectroscopy, where the modelling is carried out for custom narrow passbands. LDTk can be used to compute custom LD profiles with uncertainties propagated from the uncertainties in the stellar parameter estimates. It can also can be used to estimate the LD-model-specific coefficients with uncertainties for the most common LD models. LDTk may also be directly integrated into the log posterior computation of any pre-existing modelling code with minimal modifications. LDTk is written in Python; last update 2017.

limb-darkening `https://github.com/nespinoza/limb-darkening`
Espinoza and Jordán (2015) provide a guide about the use of limb darkening coefficients in transit fits. The associated codes may generate limb darkening coefficients

using arbitrary response functions, based on ATLAS and PHOENIX model atmospheres. These codes are written in Python, last update 2015.

ld-exosim `https://github.com/nespinoza/ld-exosim/`
Espinoza and Jordán (2016) investigate also the selection of the best limb darkening law for the modelling of a transit light curves. The associated code ld-exosim may select the optimal limb darkening relation and is able to calculate the limb darkening induced biases on various exoplanet parameters. ld-exosim is written in Python; last update 2016.

Multi-purpose tools

DACE `https://dace.unige.ch`
The University of Geneva is developing the Data & Analysis Center for Exoplanets (DACE), which is a web platform dedicated to extrasolar planets data visualisation, exchange and analysis. There are several online tools to analyse RV data as well as transit light curves, either on public data accessible through the tool or on own data that may be imported. Other sections concern access to catalogues, observation planning, stellar models, as well as planet and planet-system formation and evolution. The platform serves as a common interface for codes written in different languages and offers the possibility to run long computations as background jobs.

Epilogue: Ephemeral software availability and solutions

During the preparation of this contribution, the author experienced that the availability of codes presented in the literature is far from assured. For one, some articles are centred on the presentation of a code without mentioning its availability. Such articles were excluded in this contribution and in the author's opinion, refereed journals should accept these only for well-argued exceptions.

More frequent were however cases where links to codes were given but were found to be inoperative, sometimes even in refereed publications less than 3 years old. In all of them, the inoperative links pertained to institute websites whose authors had moved on. In some cases, alternative access methods could be provided after contacting the authors. It is however evident that the long-term availability of many codes is uncertain. The loss of access to a code is not only a loss of a tool for the analysis of data, but hinders also the reproducibility and falsifiability of published results[3].

[3] We note here also the initiative for the Reproducible Research, with a license (RRSL, Harrington 2017, as used in BART and TEA) that stipulates the publication of the computer codes behind published results.

Besides institute websites, many codes listed in this chapter are being hosted at GitHub[4] (`www.github.com`), which provides free user accounts for codes that are public and open source. This site's raison d'être is however the collaborative development of codes, not their long-term storage, and neither could a clear statement on GitHub's site about the long-term availability of codes be found, which is a particular concern for accounts that become inactive.

A valuable initiative that resolves both the issues of long-term storage and the unambigous identification of a given digital resource is Zenodo (`http://zenodo.org`), hosted at CERN. This site provides both long-term storage and assigns digital object identifiers (DOI numbers) to stored resources, permitting to cite them directly. We note that several versions of a code can be stored at Zenodo, all receiving distinct DOIs, while code from Github can be deposited at Zenodo in a streamlined process. An example of the citation of a Zenodo resource is 'Parviainen and Borsato (2016)' in the entry of PyTransit. Another option for long-term storage of codes is the depositing of source codes at CDS, an example of this is *ellc* (see entry). Developers of scientific codes are encouraged to take advantage of these solutions!

Cross-References

- Bayesian methods for exoplanet science (Parviainen)
- Stellar limb darkening's effects on exoplanet characterization (Csizmadia)
- The Impact of Stellar Activity on the Detection and Characterization of Exoplanets (Cameron)
- Characterization of exoplanets: Observations and modelling of orbital phase curves (Parmentier & Crossfield)
- Radial velocities as an exoplanet discovery method (Wright)
- Transit photometry as an exoplanet discovery method (Bakos)
- Transit Timing and Duration Variations as an Exoplanet Discovery Method (Agol & Fabrycky)

Acknowledgements The author thanks all those who provided feedback to his call for nominations to this listing, made through the Exoplanets Mailing List at Grenoble University, and those who commented on its first draft. This work has been supported by the Spanish Spanish Secretary of State for R&D&i (MINECO) under the grant ESP2015-65712-C5-4-R.

References

Agol E Deck K (2016a) Transit Timing to First Order in Eccentricity. ApJ818:177

Agol E Deck K (2016b) TTVFaster: First order eccentricity transit timing variations (TTVs). Astrophysics Source Code Library, ascl:1604.012

[4] GitHub is a trademark registered in the United States by GitHub, Inc.

Almenara JM, Díaz RF, Mardling R et al. (2015) Absolute masses and radii determination in multiplanetary systems without stellar models. MNRAS453:2644–2652
Andreasen DT, Sousa SG, Tsantaki M et al. (2017) SWEET-Cat update and FASMA. A new minimization procedure for stellar parameters using high-quality spectra. A&A600:A69
Baluev RV (2013a) PlanetPack: A radial-velocity time-series analysis tool facilitating exoplanets detection, characterization, and dynamical simulations. Astronomy and Computing 2:18–26
Baluev RV (2013b) PlanetPack: Radial-velocity time-series analysis tool. Astrophysics Source Code Library, ascl:1311.004
Baluev RV (2015) PlanetPack3: a software tool for exoplanets characterization from radial velocity and transit data. IAU General Assembly 22:2249592
Barros SCC, Almenara JM, Demangeon O et al. (2015) Photodynamical mass determination of the multiplanetary system K2-19. MNRAS454:4267–4276
Béky B (2014) SPOTROD: Semi-analytic model for transits of spotted stars. Astrophysics Source Code Library, ascl:1411.015
Béky B, Kipping DM Holman MJ (2014) SPOTROD: a semi-analytic model for transits of spotted stars. MNRAS442:3686–3699
Blecic J (2016) Observations, Thermochemical Calculations, and Modeling of Exoplanetary Atmospheres. ArXiv e-prints, arXiv:160402692
Blecic J, Harrington J Bowman MO (2015) TEA: Thermal Equilibrium Abundances. Astrophysics Source Code Library, ascl:1505.031
Blecic J, Harrington J Bowman MO (2016) TEA: A Code Calculating Thermochemical Equilibrium Abundances. ApJS225:4
Boisse I, Bonfils X Santos NC (2012) SOAP. A tool for the fast computation of photometry and radial velocity induced by stellar spots. A&A545:A109
Borkovits T, Rappaport S, Hajdu T Sztakovics J (2015) Eclipse timing variation analyses of eccentric binaries with close tertiaries in the Kepler field. MNRAS448:946–993
Borsato L (2016) TRADES: TRAnsits and Dynamics of Exoplanetary Systems. Astrophysics Source Code Library, ascl:1601.001
Borsato L, Marzari F, Nascimbeni V et al. (2014) TRADES: A new software to derive orbital parameters from observed transit times and radial velocities. Revisiting Kepler-11 and Kepler-9. A&A571:A38
Bradstreet DH (2005) Fundamentals of Solving Eclipsing Binary Light Curves Using Binary Maker 3. Society for Astronomical Sciences Annual Symposium 24:23
Brakensiek J Ragozzine D (2016a) CORBITS: Efficient Geometric Probabilities of Multi-Transiting Exoplanetary Systems. Astrophysics Source Code Library, ascl:1603.002
Brakensiek J Ragozzine D (2016b) Efficient Geometric Probabilities of Multi-Transiting Exoplanetary Systems from CORBITS. ApJ821:47
Bruntt H (2009) Accurate fundamental parameters of CoRoT asteroseismic targets. The solar-like stars HD 49933, HD 175726, HD 181420, and HD 181906. A&A506:235–244
Bruntt H, Catala C, Garrido R et al. (2002) Abundance analysis of targets for the COROT/MONS asteroseismology missions. I. Semi-automatic abundance analysis of the gamma Dor star HD 49434. A&A389:345–354
Bruntt H, Deleuil M, Fridlund M et al. (2010) Improved stellar parameters of CoRoT-7. A star hosting two super Earths. A&A519:A51
Carter JA, Fabrycky DC, Ragozzine D et al. (2011) KOI-126: A Triply Eclipsing Hierarchical Triple with Two Low-Mass Stars. Science 331:562
Cubillos P, Blecic J, Harrington J et al. (2016) BART: Bayesian Atmospheric Radiative Transfer fitting code. Astrophysics Source Code Library, ascl:1608.004
Cubillos P, Harrington J, Loredo TJ et al. (2017) On Correlated-noise Analyses Applied to Exoplanet Light Curves. AJ153:3
Cubillos PE (2016) Characterizing Exoplanet Atmospheres: From Light-curve Observations to Radiative-transfer Modeling. ArXiv e-prints, arXiv:160401320
Czekala I, Mandel KS, Andrews SM et al. (2017) Disentangling Time Series Spectra with Gaussian Processes: Applications to Radial Velocity Analysis. ArXiv e-prints, arXiv:170205652

Deck K, Agol E, Holman M Nesvorny D (2014a) TTVFast: Transit timing inversion. Astrophysics Source Code Library, ascl:1404.015

Deck KM, Agol E, Holman MJ Nesvorný D (2014b) TTVFast: An Efficient and Accurate Code for Transit Timing Inversion Problems. ApJ787:132

Deeg H (2009) UTM, a universal simulator for lightcurves of transiting systems. In: Pont F, Sasselov D Holman MJ (eds) Transiting Planets, IAU Symposium, vol 253, pp 388–391, DOI 10.1017/S1743921308026720

Deeg HJ (2014) UTM: Universal Transit Modeller. Astrophysics Source Code Library, ascl:1412.003

Díaz RF, Almenara JM, Santerne A et al. (2014) PASTIS: Bayesian extrasolar planet validation - I. General framework, models, and performance. MNRAS441:983–1004

Dindar S, Ford EB, Juric M et al. (2012) Swarm-NG: Parallel n-body Integrations. Astrophysics Source Code Library, ascl:1208.012

Dindar S, Ford EB, Juric M et al. (2013) Swarm-NG: A CUDA library for Parallel n-body Integrations with focus on simulations of planetary systems. New Astronomy 23:6–18

Dumusque X, Boisse I Santos NC (2014) SOAP 2.0: A Tool to Estimate the Photometric and Radial Velocity Variations Induced by Stellar Spots and Plages. ApJ796:132

Eastman J, Siverd R Gaudi BS (2010) Achieving Better Than 1 Minute Accuracy in the Heliocentric and Barycentric Julian Dates. PASP122:935

Eastman J, Gaudi BS Agol E (2012) EXOFAST: Fast transit and/or RV fitter for single exoplanet. Astrophysics Source Code Library, ascl:1207.001

Eastman J, Gaudi BS Agol E (2013) EXOFAST: A Fast Exoplanetary Fitting Suite in IDL. PASP125:83

Espinoza N Jordán A (2015) Limb darkening and exoplanets: testing stellar model atmospheres and identifying biases in transit parameters. MNRAS450:1879–1899

Espinoza N Jordán A (2016) Limb darkening and exoplanets - II. Choosing the best law for optimal retrieval of transit parameters. MNRAS457:3573–3581

Espinoza N, Brahm R, Jordán A et al. (2016) Discovery and Validation of a High-Density sub-Neptune from the K2 Mission. ApJ830:43

Foreman-Mackey D, Conley A, Meierjurgen Farr W et al. (2013) emcee: The MCMC Hammer. Astrophysics Source Code Library, ascl:1303.002

Gazak JZ, Johnson JA, Tonry J et al. (2011) Transit Analysis Package (TAP and autoKep): IDL Graphical User Interfaces for Extrasolar Planet Transit Photometry. Astrophysics Source Code Library, ascl:1106.014

Gazak JZ, Johnson JA, Tonry J et al. (2012) Transit Analysis Package: An IDL Graphical User Interface for Exoplanet Transit Photometry. Advances in Astronomy 2012:697967

Giménez A (2006) Equations for the analysis of the light curves of extra-solar planetary transits. A&A450:1231–1237

Hadden S Lithwick Y (2016) Numerical and Analytical Modeling of Transit Timing Variations. ApJ828:44

Harrington J (2017) Reproducible research. URL `https://planets.ucf.edu/resources/reproducible-research/`

Iglesias-Marzoa R, López-Morales M Jesús Arévalo Morales M (2015a) rvfit: Radial velocity curves fitting for binary stars or exoplanets. Astrophysics Source Code Library, ascl:1505.020

Iglesias-Marzoa R, López-Morales M Jesús Arévalo Morales M (2015b) The rvfit Code: A Detailed Adaptive Simulated Annealing Code for Fitting Binaries and Exoplanets Radial Velocities. PASP127:567

Jackson BK, Lewis NK, Barnes JW et al. (2012) The EVIL-MC Model for Ellipsoidal Variations of Planet-hosting Stars and Applications to the HAT-P-7 System. ApJ751:112

Kempton EMR, Lupu R, Owusu-Asare A, Slough P Cale B (2017) Exo-Transmit: An Open-Source Code for Calculating Transmission Spectra for Exoplanet Atmospheres of Varied Composition. PASP129(4):044,402

Kenworthy MA Mamajek EE (2015a) Exorings: Exoring modelling software. Astrophysics Source Code Library, ascl:1501.012

Kenworthy MA Mamajek EE (2015b) Modeling Giant Extrasolar Ring Systems in Eclipse and the Case of J1407b: Sculpting by Exomoons? ApJ800:126
Kipping DM (2010) Binning is sinning: morphological light-curve distortions due to finite integration time. MNRAS408:1758–1769
Kipping DM (2011) LUNA: an algorithm for generating dynamic planet-moon transits. MNRAS416:689–709
Kipping DM (2012) An analytic model for rotational modulations in the photometry of spotted stars. MNRAS427:2487–2511
Kjurkchieva D, Dimitrov D, Vladev A Yotov V (2013a) New approach for modelling of transiting exoplanets for arbitrary limb-darkening law. MNRAS431:3654–3662
Kjurkchieva D, Dimitrov D, Vladev A Yotov V (2013b) TAC-maker: Transit Analytical Curve maker. Astrophysics Source Code Library, ascl:1303.010
Kostov VB, McCullough PR, Hinse TC et al. (2013) A Gas Giant Circumbinary Planet Transiting the F Star Primary of the Eclipsing Binary Star KIC 4862625 and the Independent Discovery and Characterization of the Two Transiting Planets in the Kepler-47 System. ApJ770:52
Kostov VB, McCullough PR, Carter JA et al. (2014) Kepler-413b: A Slightly Misaligned, Neptune-size Transiting Circumbinary Planet. ApJ784:14
Kreidberg L (2015) batman: BAsic Transit Model cAlculatioN in Python. PASP127:1161
Line MR, Wolf AS, Zhang X et al. (2013) A Systematic Retrieval Analysis of Secondary Eclipse Spectra. I. A Comparison of Atmospheric Retrieval Techniques. ApJ775:137
Malavolta L (2016) PyORBIT: Exoplanet orbital parameters and stellar activity. Astrophysics Source Code Library, ascl:1612.008
Mandel K Agol E (2002) Analytic Light Curves for Planetary Transit Searches. ApJ580:L171–L175
Maxted PFL (2016) ellc: A fast, flexible light curve model for detached eclipsing binary stars and transiting exoplanets. A&A591:A111
Meschiari S, Wolf AS, Rivera E et al. (2009) Systemic: A Testbed for Characterizing the Detection of Extrasolar Planets. I. The Systemic Console Package. PASP121:1016
Meschiari S, Wolf AS, Rivera E et al. (2012) Systemic Console: Advanced analysis of exoplanetary data. Astrophysics Source Code Library, ascl:1210.018
Montalto M, Boué G, Oshagh M et al. (2014) Improvements on analytic modelling of stellar spots. MNRAS444:1721–1728
Montalto M, Boué G, Oshagh M et al. (2015) KS Integration: Kelvin-Stokes integration. Astrophysics Source Code Library, ascl:1505.004
Morton TD (2012) An Efficient Automated Validation Procedure for Exoplanet Transit Candidates. ApJ761:6
Morton TD (2015) VESPA: False positive probabilities calculator. Astrophysics Source Code Library, ascl:1503.011
Nesvorný D (2009) Transit Timing Variations for Eccentric and Inclined Exoplanets. ApJ701:1116–1122
Oshagh M, Boisse I, Boué G et al. (2013) SOAP-T: a tool to study the light curve and radial velocity of a system with a transiting planet and a rotating spotted star. A&A549:A35
Pál A (2009) Tools for discovering and characterizing extrasolar planets. PhD thesis, Department of Astronomy, Eötvös Loránd University
Pál A (2012) Light-curve modelling for mutual transits. MNRAS420:1630–1635
Parviainen H (2015a) PYTRANSIT: fast and easy exoplanet transit modelling in PYTHON. MNRAS450:3233–3238
Parviainen H (2015b) PyTransit: Transit light curve modeling. Astrophysics Source Code Library, ascl:1505.024
Parviainen H Aigrain S (2015) LDTK: Limb Darkening Toolkit. MNRAS453:3821–3826
Parviainen H Borsato L (2016) hpparvi/pytransit: v1.0-beta.4. DOI 10.5281/zenodo.157363, URL `https://doi.org/10.5281/zenodo.157363`
Parviainen H, Pope B Aigrain S (2016) K2PS: K2 Planet search. Astrophysics Source Code Library, ascl:1607.010

Piskunov N Valenti JA (2017) Spectroscopy Made Easy: Evolution. A&A597:A16
Pope BJS, Parviainen H Aigrain S (2016) Transiting exoplanet candidates from K2 Campaigns 5 and 6. MNRAS461:3399–3409
Prša A, Conroy KE, Horvat M et al. (2016) Physics Of Eclipsing Binaries. II. Toward the Increased Model Fidelity. ApJS227:29
Rojo PM (2006) Transit spectroscopy of the extrasolar planet HD 209458B: The search for water. PhD thesis, Cornell University
Seager S Mallén-Ornelas G (2003) A Unique Solution of Planet and Star Parameters from an Extrasolar Planet Transit Light Curve. ApJ585:1038–1055
Smith AMS, Gandolfi D, Barragán O et al. (2017) K2-99: a subgiant hosting a transiting warm Jupiter in an eccentric orbit and a long-period companion. MNRAS464:2708–2716
Southworth J (2013) The solar-type eclipsing binary system LL Aquarii. A&A557:A119
Southworth J, Maxted PFL Smalley B (2004) Eclipsing binaries in open clusters - II. V453 Cyg in NGC 6871. MNRAS351:1277–1289
Torres G, Fressin F, Batalha NM et al. (2011) Modeling Kepler Transit Light Curves as False Positives: Rejection of Blend Scenarios for Kepler-9, and Validation of Kepler-9 d, A Super-earth-size Planet in a Multiple System. ApJ727:24
Tregloan-Reed J, Southworth J Tappert C (2013) Transits and starspots in the WASP-19 planetary system. MNRAS428:3671–3679
Tregloan-Reed J, Southworth J, Burgdorf M et al. (2015) Transits and starspots in the WASP-6 planetary system. MNRAS450:1760–1769
Valenti JA Piskunov N (1996) Spectroscopy made easy: A new tool for fitting observations with synthetic spectra. A&AS118:595–603
Valenti JA Piskunov N (2012) SME: Spectroscopy Made Easy. Astrophysics Source Code Library, ascl:1202.013
Wang SX Wright JT (2012) BOOTTRAN: Error Bars for Keplerian Orbital Parameters. Astrophysics Source Code Library, ascl:1210.030
Welsh WF, Orosz JA, Short DR et al. (2015) Kepler 453 b - The 10th Kepler Transiting Circumbinary Planet. ApJ809:26
Wright J Howard A (2012) RVLIN: Fitting Keplerian curves to radial velocity data. Astrophysics Source Code Library, ascl:1210.031