The fitting of radial velocity curves using Adaptive Simulated Annealing

Ramón Iglesias-Marzoa[1,2]**, Mercedes López-Morales**[3]**, and María Jesús Arévalo Morales**[1,4]

[1] Astrophysics Department, Universidad de La Laguna, E-38205 La Laguna, Tenerife, Spain
[2] Centro de Estudios de Física del Cosmos de Aragón, Plaza San Juan 1, E-44001 Teruel, Spain
[3] Harvard-Smithsonian Center for Astrophysics, 60 Garden Street, Cambridge, MA 02138, USA
[4] Instituto de Astrofísica de Canarias, E-38200 La Laguna, Tenerife, Spain

Abstract

We present a new code for fitting radial velocities of stellar binaries and exoplanets using an Adaptive Simulated Annealing (ASA) global minimisation method. ASA belongs to the family of Monte Carlo methods and its main advantages are that it only needs evaluations of the objective function, it does not rely on derivatives, and the parameters space can be periodically redefined and rescaled for individual parameters. ASA is easily scalable since the physics is concentrated in only one function and can be modified to account for more complex models. Our ASA code minimises the χ^2 function in the multidimensional parameters space to obtain the full set of parameters (P, T_p, e, ω, γ, K_1, K_2) of the keplerian radial velocity curves which best represent the observations. As a comparison we checked our results with the published solutions for several binary stars and exoplanets with available radial velocities data. We achieve good agreement within the limits imposed by the uncertainties.

1   Introduction

Precise, absolute masses of stars and planets can be only measured if they are part of binary (star-star or star-planet) systems. The masses can be derived by fitting the radial velocity (RV) curves of those systems to the keplerian orbital equations. Those equations need to be solved numerically using multi-parameter minimisation techniques but this process gets complicated by the need to explore a wide parameter space and by the existence of many

potential local minima. To deal with this problem the most frecuently used software packages include methods for local and global minimisation.

Simulated Annealing (SA) is a multi-parameter minimisation technique that draws on metallurgic optimum cooling processing methods [12, 14, 9, 1] and has been implemented and tested in some of the most popular exoplanet and binary star modeling codes, but this technique has been somewhat demoted on claims of being notoriously slow and less efficient than other methods (e.g. [16, 8, 19]).

A variation of the original SA algorithm, called Adaptive Simulated Annealing (ASA) [5, 6, 7], overcomes the speed problem of standard SA methods and ensures fast converging to a global minimum solution for stellar binaries and exoplanet RV curves. ASA was applied by [15] to the computation of orbits for binary systems with RV and visual measurements. But the cases where the only observations available are the RV are many more, for instance, exoplanets or single/double line eclipsing binaries. In this proceedings we present our implementation of an ASA algorithm to fit those observations.

2   The Adaptive Simulated Annealing algorithm

An excellent and precise description of how the ASA algorithm can be implemented is the work of [2] which is based in the original ASA papers. Our implemented ASA algorithm follows closely his work. Another very good description of the ASA algorithm and SA in general is [10].

The ASA algorithm minimises an objective function in a way that resembles the behaviour of the internal energy of a body heated and then slowly cooled (anneal): an independent parameter called *temperature* is reduced following an annealing law; for each temperature, a number N_{gen} of possible *test states* are generated from a previous state, and the objective function is computed for each one; a test state is accepted or rejected based in the acceptance rules, which depends on the temperature; if a test state is accepted, it becomes the actual state and, if it is the one with the minimum absolute energy, it is saved; the temperature is reduced to begin the process from the last accepted state.

The key feature in all the SA algorithms, what prevent them from getting stuck in a local minimum, is that any state with worst energy configuration, i.e. an 'uphill movement', can be accepted with a certain probability. Thus, to work properly, any SA method consists on the next three functions: a *probability density function* for a N-parameter space, where N are the parameters to minimise, which generates new test states; a *probability of acceptance function*, which determines whether a given step solution is accepted or discarded, and the *annealing schedule*, which defines how the parameters change with each iteration step. The functional form of these functions for the ASA algorithm with their convergence properties can be found in [5, 6, 7, 2]. We refer the reader to these references to obtain more information.

ASA differs from classical SA in two aspects: each parameter to fit has their own generating temperature $T_{i,gen}$, and every N_{accept} acceptances, these temperatures are rescaled using the sensitivities, which depends on the local topology of the parameters space and must be computed numerically.

3 The cost function

Assuming gaussian uncertainties, the function to minimise is the χ^2, which, for a double line binary system, is given by:

$$\chi^2 = \sum_{i=1}^{N_i} \left(\frac{v_{calc,i} - v_i}{\sigma_i} \right)^2 + \sum_{j=1}^{N_j} \left(\frac{v_{calc,j} - v_j}{\sigma_j} \right)^2, \tag{1}$$

where N_i and N_j are the number of RV for the primary and secondary components, v_i and v_j, and σ_i and σ_j are the measured velocities and their associated errors. In addition, $v_{calc,i}$ and $v_{calc,j}$ corresponds to the expected RV of the components calculated using the keplerian orbit equation:

$$v_{calc,i} = \gamma + K_1[\cos(\theta + \omega) + e\cos\omega], \tag{2}$$

where γ is the center of mass velocity of the system measured from the Sun, θ is the true anomaly, e is the eccentricity of the orbit, and K_1 is the amplitude of the RV. A similar equation is applied to $v_{calc,j}$ using K_2 and $\omega' = \omega + \pi$ for the secondary component. The true anomaly θ is calculated from the eccentric anomaly E in the usual way, and this one from the mean anomaly M using the Kepler equation:

$$E - e\sin E = M = \frac{2\pi}{P}(t - T_p) \tag{3}$$

Resolving numerically this equation is a key step in the algorithm since the generating function scans the parameters space thousands of times in an ergodic fashion, and every possible combination of parameters can be generated. Thus, the numerical computation of the Kepler equation must be robust and fast. To achieve this goals, we implemented a two step algorithm: first, an initial value for the eccentric anomaly E is computed using a third order Maclaurin series expansion; and second, a Newton classical iterative method was used to refine the computed initial value to the desired accuracy.

4 Computation of the uncertainties

For a guess of the parameter uncertainties, the code computes the Fisher matrix $[F]$ at the global minimum. The covariance matrix is obtained by inversion $[C] = [F]^{-1}$ and the parameter variances extracted from the main diagonal. Since the uncertainties obtained in this way cannot represent non-linear correlations among parameters, we programed the Markov Chain Monte Carlo (MCMC) method as a separate code.

Our MCMC code makes use of the Metropolis-Hastings (MH) algorithm to sample the distribution $P(x)$ of the cost function in the vicinity of the global optimum. This code works in two basic steps: first, a new state x' is generated from a candidate distribution $q(x)$; second, the decision to accept or to reject the new state is made using the MH acceptance

probability A which, for a symmetric candidate distribution $q(x)$, is given by:

$$A(x, x') = \min \left(1, \exp \left[\frac{1}{2} (\chi^2(x) - \chi^2(x')) \right] \right) \tag{4}$$

The code uses a gaussian $q(x)$ distribution centered in the actual state x with a σ, which need to be optimized for a good mixing of the Markov chain: too small values of σ can lead to a slow scanning of the whole range of parameters and for too large values the Markov chain gets stuck in particular values for long periods of time. We developed a simple algorithm to dynamically adjust σ before the MCMC was started. In this algorithm we assume a desired acceptance probability of $p \simeq 0.25$ to obtain a good mixing of the Markov chain, since some studies claim that this is the best value for spaces with dimensions 5 to 100 (see [3, 11]). The typical chain length is 10^5 acceptances, enough to provide a reliable measurement of the uncertainties through the marginalized histogram for each parameter. In many cases the uncertainties can be calculated by fitting a gaussian to this histogram but, whether it is asymmetric, a 68% shortest confidence interval can be computed.

5 Fitting test with synthetic datasets

To characterize the code's performance we carry out intensive tests with synthetic datasets with the following goals: to asses the stability of the code, to characterize the behaviour of the code when fitting RV curves with different signal to noise ratios, and to measure the run time for different number of RV observations. Because of its interest we show the results of this last tests in Table 1. In that Table the t_{68}, t_{95}, and t_{99} values are the times in which the code fitted all the parameters for the 68%, 95%, and 99% of the runs, respectively, for synthetic datasets with 15, 50, 100 and 1000 observations.

Table 1: Execution times measured over 1000 consecutive runs for several datasets. Each time was measured computing the cumulative distribution function (CDF) and interpolating the desired percentage. Note that doubling the t_{68} execution times is a good guess for t_{99}.

Number of points	Execution times (s)		
	t_{68}	t_{95}	t_{99}
15	4.1	6.8	8.9
50	7.6	12.7	16.1
100	20.2	30.4	38.5
1000	81.1	132.7	175.1

6 Fitting tests on real systems

To compare the results with other published solutions we selected a small group of objects with published RV and orbital solution and with a variety of physical configurations. In

Table 2: Results for the fits to RV data for the selected systems. For short, only a subset of the fitted parameters are displayed. The uncertainties were computed using a MCMC chain (see text). The symmetric uncertainties were calculated by fitting a gaussian to the marginalized histogram. The asymmetric uncertainties were computed using the 68.3% shortest confidence interval over the histogram. The uncertainties are expressed as the two last digits of the parameter, with the decimal point present where necessary for guidance. ^{f} means fixed and adopted from the original paper.

Object Name	P (d)	e −	ω (deg)	K_1 (km/s)	K_2 (km/s)	Remarks	Reference
LV Her	18.4359535^f	0.61273(73)	352.20(.24)	67.24(.19)	68.59(.27)	−	[18]
	18.43600(14)	0.6137(19)	352.22(.27)	67.31(.24)	68.67(.33)	all free	[this work]
	18.4359535^f	0.6138(20)	352.23(.28)	67.32(.26)	68.68(.35)	P fixed	[this work]
Kepler 78b	0.3550(4)	0.0^f	−	0.00196(32)	−	−	[13, 17, 4]
	0.34951(15)	0.0^f	90^f	$0.00173(^{+24}_{-38})$	−	T_p, e, ω fixed	[this work]
	$0.35500744(6)^f$	0.0^f	90^f	0.00186(28)	−	P, e, ω fixed	[this work]

Table 2 we show the results of our fitting tests for two of the selected systems.

LV Her is a double-line eclipsing binary star with a highly eccentric orbit, a high quality RV curve and a long observing time interval relative to its period [18]. Two fits were done: the first one leaving all the parameters free and the second one fixing P to the value from the light curve to mimic the fit of the reference [18]. In both cases all the parameters are recovered (see Fig. 1).

Kepler 78b is the first Earth-like transiting exoplanet discovered in the Kepler mission data ([13, 17, 4]). It has a nearly circular orbit and the RV data set has a very low signal to noise ratio (see Fig. 1). Two fits were done: one in which T_p, e, and ω were fixed to reproduce the analysis of [13], resulting in lower values of P and K_1, and a second one fixing P, e, and ω with the values obtained from the Kepler light curve. In this second fit the value for K_1 has a better agreement with the value published in [13].

Acknowledgments

References

[1] Černý, V. 1985, Optimization Theory and Applications, 45, 41

[2] Chen, S., & Luk, B. L. 1999, Signal Processing, 79, 117

[3] Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. 2003, *Bayesian Data Analysis, 2nd Ed.*, Chapman & Hall/CRC
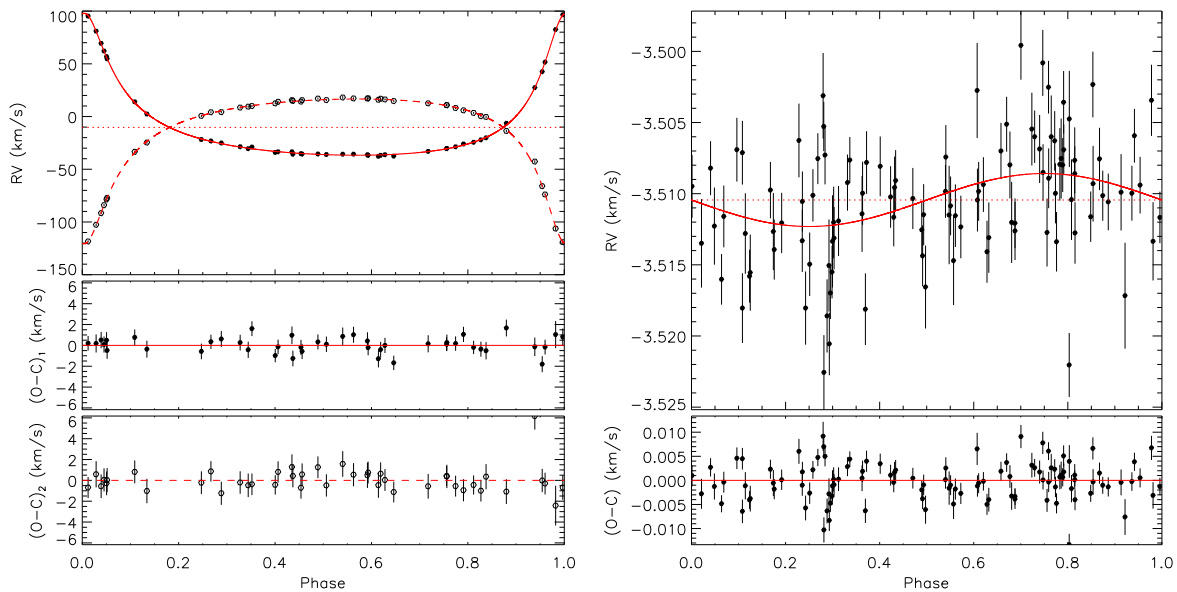
Figure 1: Fits obtained with our ASA implementation for the eccentric double-lined eclipsing binary LV Her (left panel) and the exoplanet system Kepler 78b (right panel). In this case the fit was done fixing P, e, and ω.

[4] Howard, A., et al. 2013, Nature, 503, 381

[5] Ingber, L. 1989, Mathl. Comput. Modelling, 12, 967

[6] Ingber, L. 1993, Mathl. Comput. Modelling, 18, N11, 29

[7] Ingber, L. 1996, Control and Cybernetics, 25, N1, 33

[8] Kallrath, J., Milone, E. F. 2006, Eclipsing Binary Stars. Modelling and Analysis (Springer)

[9] Kirkpatrick S., Gelatt, C. D., & Vecchi M. P. 1983, Sci, 220, 671

[10] Locatelli, M. 2000, Simulated annealing algorithms for continuous global optimization, in the Handbook of Global Optimization II, Kluwer Academic Publishers, p179

[11] MacKay, D. J. C. 2006, Information Theory, Inference, and Learning Algorithms (Cambridge University Press)

[12] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. 1953, J. Chem. Phys. 21(6), 1087

[13] Pepe, F., et al. 2013, Nature, 503, 377

[14] Pincus, M. 1970, Operations Research, 18, 1225

[15] Pourbaix, D. 1998, A&AS, 131, 377

[16] Prša, A., & Zwitter, T. 2005, ApJ, 628, 426

[17] Sanchis-Ojeda, R., et al. 2013, ApJ, 774, 54

[18] Torres, G., Sandberg Lacy, C. H., & Claret, A. 2009, AJ, 138, 1622

[19] Wichmann, R. 1999, Nightfall Users Manual