Fitting Rectangular Signals to Time Series Data by Metaheuristic Algorithms

Andreas M. Chwatal and Günther R. Raidl

Vienna University of Technology, Vienna, Austria
{chwatal|raidl}@ads.tuwien.ac.at

Abstract. In this work we consider the application of metaheuristic algorithms to the problem of fitting rectangular signals to time-data series. The application background is to search for transit signals of exoplanets in stellar photometric observation data. The presented algorithms include an Evolution Strategy and Differential Evolution; both algorithms use an efficient reduction of the search space by exactly solving a subproblem. The presented results affirm the presented methods to be promising and effective tools for the discovery of the first multi-transit planetary system.

1 Introduction

Fitting parametrized models to data series is a frequently performed task in scientific computing. Nevertheless, finding (near-)optimal fits of superposed periodical signals to time-series data becomes a non-trivial problem when non-sinusoidal models are considered. In this case it is not always possible to derive good model parameters from the Fourier spectrum. Noisy data may further complicate this task. Finding good fits, which is in fact a continuous parameter optimization problem, is a computationally challenging task under these circumstances. In this work, we consider the problem of fitting rectangular signals to time-data series, and present metaheuristic algorithms to solve the problem.

2 Problem Description

The particular application background comes from the field of astronomy, in particular the problem of finding signals from transiting exoplanets in stellar photometric light-curves. For a comprehensive overview on exoplanets and detection methods see [1]. A transiting planet periodically shadows some of the light from its host star for a short time when it moves into our line of sight to the star. During the transit the luminosity of the star is marginally reduced. By neglecting the in- and egress phases, the transit-lightcurve can be well approximated by a periodic rectangular signal. The corresponding parameters are the period p the transit occurs with, a phase offset τ, the length l of the transit, and finally the transit depth d. The latter parameter corresponds to the percentage of light from the star being shadowed by the transiting planet. Figure 1 depicts

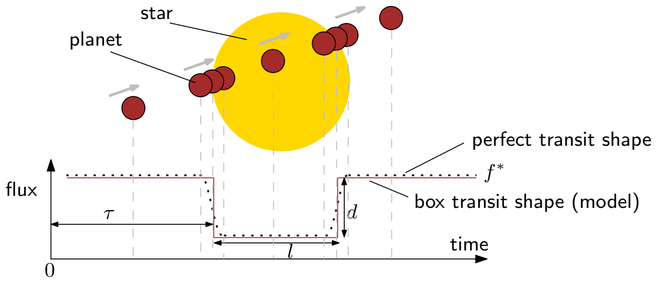

Fig. 1. Transiting planet and corresponding lightcurve

the situation for a single planet. Assuming M planets, the signal of the model at time t is given by

$$\phi(t) = f^* - \sum_{j=1}^{M} \chi_j^t d_j, \tag{1}$$

where f^* denotes a further parameter describing the regular flux (luminosity) of the host star; χ_j^t indicates if planet j is transiting at time t and is given by

$$\chi_j^t = \begin{cases} 1 & \text{if } \tau_j < t \bmod p_j \leq \tau_j + l_j \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The observed data series is given by a list of $\{(t_i, f_i)\}, 1 \leq i \leq N$, where t_i denotes a particular observation time and f_i the observed photon flux (i.e. luminosity) at that given time. Let further $m_j = (p_j, l_j, d_j, \tau_j)$ and hence $\boldsymbol{m}$ be the vector of all model parameters (except f^*). The overall quality of the fit can be characterized by the root mean square error

$$f(\boldsymbol{m}, f^*, \boldsymbol{t}, \boldsymbol{f}) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (f_i - \phi(t_i))^2}. \tag{3}$$

The overall objective is to find a parameter setup for $\boldsymbol{m}$ and f^* minimizing Eq. (3), i.e. to find a model with minimal deviation from the observations. Due to stellar fluctuations and measurement errors real-world instances contain noisy signals. The signal-to-noise ratios can be expected to be very low, i.e. the respective values of d_j will be in the same order of magnitude as the standard deviation σ_f of the input values.

3 Previous Work

Several applications of genetic algorithms in astronomy are outlined in [2], and have since then been successfully applied for many purposes. In particular for the detection of exoplanets, evolutionary algorithms have been used with some success. For instance, an evolution strategy for fitting Keplerian models to radial velocity data is described in [3].

The development of efficient transit detection algorithms has recently gained more interest in the scientific community, as space-based missions like CoRoT[1] provide a great amount of observational data. One of the most popular approaches is the *box fitting least-square algorithm* [4]. This approach, as well as *phase dispersion minimization* [5] have the main drawback, that they are only directly applicable for finding single planet transits.

So far, no multi-planet system could be discovered by the transit method, which is possibly due to the difficulty of detecting their signals in (existing) observational data. More efficient techniques to tackle this numeric optimization problem would thus be a valuable contribution to exoplanet research.

4 Improvement and Evaluation of Candidate Solutions

The overall search process becomes more efficient when optimal values of depths d_j are derived from p_j, l_j, τ_j for each planet j. For this purpose we introduce binary flags $(b_1, \ldots, b_M)$ for each observation point $o_i = (t_i, f_i), i = 1, \ldots, N$, indicating which planet is transiting at the given time. These flags can be interpreted as integer number with binary representation $b_1 b_2 \ldots b_M \in [0, 2^M - 1]$, implying a partitioning of the set $O = \{o_1, \ldots, o_N\}$ of all observation points $O = O_0 \cup O_1 \cup \ldots \cup O_{2^M-1}$. Assuming two planets $M = 2$ we obtain the set of out-of-transit observations O_0, the sets O_1, O_2 of transit events of planets one and two respectively, and the set O_3 where planets one and two are transiting simultaneously. Optimal transit depths can be derived by minimizing

$$f(\boldsymbol{d}) = \sum_{i=1}^{N} (f_i - (f^* - \sum_{j=1}^{M} \chi_j^i d_j))^2, \tag{4}$$

which can be achieved by solving the system of linear equations resulting from $\frac{\partial f(\boldsymbol{d})}{\partial d_k} = 2 \sum_{i=1}^{N} \left(f_i - f^* + \sum_{j=1}^{M} \chi_j^i d_j \right) \cdot \chi_k^i = 0$ for all $k = 1, \ldots, M$. Let $\hat{f}^K = \sum_{i \in \bigcup_{k \in K} O_k} f_i, K \subseteq \{0, \ldots, 2^M - 1\}$ denote the sum of the observed photon fluxes from groups $\bigcup_{k \in K} O_k$, and $\hat{f} = \sum_{i=1}^{N} f_i$ analogously. Let further $n_K = |\bigcup_{k \in K} O_k|$ and $\tilde{\chi}_j^i, j = 1, \ldots, 2^M - 1,\ i = 1, \ldots, N$ indicate if observation i belongs to group j. For the case $M = 2$ we can now derive a direct expression by the partial derivative $\frac{\partial f(\boldsymbol{d})}{\partial d_1} = 2 \cdot \sum_{i=1}^{N} \left(f_i - 2f^* + 2 \sum_{j=1}^{2^M-1} \tilde{\chi}_j^i d_j \right) \cdot \left(\tilde{\chi}_1^i + \tilde{\chi}_3^i \right) = 0$ from which we obtain

$$d_1 = f^* - \frac{\hat{f}^{1,3}}{n_{1,3}} - \frac{n_3}{n_{1,3}} \cdot d_2, \quad \text{and} \quad d_2 = f^* - \frac{\hat{f}^{2,3}}{n_{2,3}} - \frac{n_3}{n_{2,3}} \cdot d_1, \tag{5}$$

where $f^* = \hat{f}^0 / n_0$. By inserting d_2 into the equation for d_1 we obtain

$$d_1 = \left((1 - \frac{n_3}{n_{1,3}}) \hat{f}^0 - \frac{1}{n_{2,3}} \hat{f}^{2,3} + \frac{n_3}{n_{1,3} \cdot n_{2,3}} \hat{f}^{2,3} \right) \cdot \left(1 - \frac{n_3^2}{n_{1,3} \cdot n_{2,3}} \right)^{-1}, \tag{6}$$

and a corresponding equation for d_2 by inserting d_1 into the equation for d_2.

[1] CoRoT: **Co**nvection **Ro**tation and planetary **T**ransits; European space telescope

5 Fitness-Landscape Analysis

In order to evaluate the applicability of metaheuristics for solving this problem, we performed a comprehensive fitness-landscape analysis. For this purpose we created numerous test instances containing signals from two planets. For each configuration we created multiple instances with different signal-to-noise ratios. Figure 2 shows the fitness-distance correlation for one typical instance. For the measure of the distances to the global optimum we used simple Euclidean distances. The left plot shows the view of the whole parameter space. One can see that there is almost no correlation of fitness values to the distances to the global optimum. All points have roughly the same value, which corresponds to the level of noise of the input instance. This effect is due to adjustment of the model transit-depths and the out-of-transit stellar flux according to the other (randomly created) parameter values, as described in Section 4. As a consequence the depths are set to zero for most configurations, and the out-of-transit stellar flux is set to the average value of all data points. Values higher than this average seldomly occur when the out-of-transit average (due to the model) is lower than the in-transit average value.

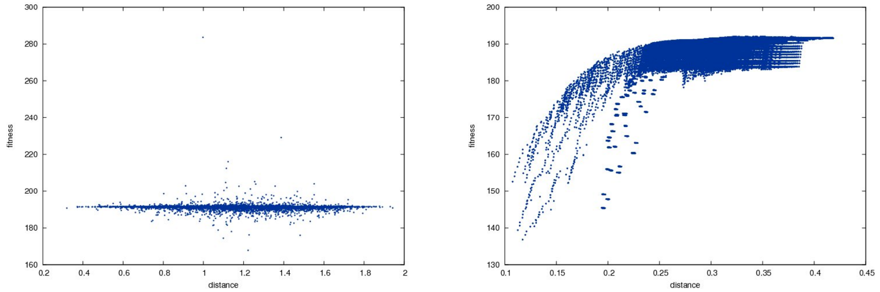

Fig. 2. Fitness-distance correlation diagrams, for the whole parameter-space (left), and a restricted parameter space close to the global optimum solution (right)

The right plot of Fig. 2 shows a closer view to the global optimum. Here, parameter values have been enumerated in a discretized way such that all distances are smaller than 0.42. This plot clearly shows that a strong correlation of distances to fitness values appears when coming close to the global optimum. These results indicate that it is very hard to find the region of the global optimum, but if that region has been found, it is relatively easy to find the global optimum itself. For problems with these properties metaheuristic algorithms are known to be a good choice. For the particular case, they must facilitate effective mechanisms for self-adaptation, i.e. to facilitate an explorative search process until the region of the global optimum is found, and then change their behavior to a fine grained exploitative search.

6 Metaheuristic Algorithms

There exists a variety of algorithms for heuristically solving difficult continuous parameter optimization problems like *Evolution Strategies* (ES), *Differential Evolution* (DE) and *Continuous Scatter Search* (CSS) [6–8]. These algorithms are population-based approaches, iteratively modifying and evaluating a set of candidate solutions. In order to find strengths and drawbacks of these methods w.r.t. this particular problem we implemented them without major modifications. Preceding experiments showed that concerning our problem ES and DE are clearly superior to CSS, as the latter one suffers from relatively time consuming subset generation. Hence, we now focus on ES and DE, which are briefly described in the following.

Individuals are directly encoded as vectors of real values in both approaches. Both algorithms do not use any local optimization method, except the techniques described in Section 4. General purpose local optimizers as for instance the Nelder-Mead method [9] turned out to be too time-consuming, and are, as indicated by the Figure 2, only beneficial if already very close to the global optimum. They are therefore not used in our evaluation.

6.1 Evolution Strategy

The ES can be classified as a (μ, λ)-ES with self-adaptation of strategy parameters [10], where μ denotes the size of the population and λ the number of offsprings created in each generation. It turned out to be advantageous to use a variant of elitist-selection which creates the new population by deterministically taking the best μ individuals from the μ parents and λ offsprings, but taking at most $\hat{\mu}$ individuals from the parents. Hence, our selection is in fact in-between $(\mu + \lambda)$-selection and (μ, λ)-selection.

Mutation is considered to be the primary operator and is performed by adding Gaussian random values to the parameters x_k (see Eq. (9)), where the standard deviation is given by a strategy parameter σ_k, associated with each parameter.

$$x'_k = x_k + N_k(0, \sigma'_k) \tag{7}$$

These strategy parameters are also modified by the evolutionary operators, which facilitates self-adaption of the search process.

$$\sigma'_k = \sigma_k \cdot e^{N(0,\tau_0)+N_k(0,\tau)} \tag{8}$$

After the application of the evolutionary operators, the optimal transit-depths $d_j, j = 1, \ldots, M$ are calculated before fitness function evaluation. If some depth is set to 0.0 – implying that this particular planet-model does not improve the quality of the fit at all – a new random planet is created on this position, which might increase diversity among the population. Prior to mutation recombination operators might be applied with some probability. We use the intermediate recombination, given by

$$x'_k = \alpha_k \cdot x^1_k + (1 - \alpha_k)x^2_k, \tag{9}$$

where x_k^1 and x_k^2 denote the parameters of the parents and α_k is a uniform random number from the interval $[-\beta, 1+\beta]$ for each parameter k, where $\beta = \frac{1}{2}$ turned out to be most successful.

6.2 Differential Evolution

Differential Evolution (DE) is a particular variant of an evolutionary algorithm, operating on a population of individuals which are encoded by a vector of real parameter values. Mutation is performed by combining three randomly selected individuals with indices (r_1, r_2 and r_3) to a new individual $v_{i,t+1}$ by

$$v_{i,t+1}^j = x_{r_1,t}^j + F \cdot (x_{r_2,t}^j - x_{r_3,t}^j), \tag{10}$$

where $F \in [0,2]$. Using the notation $u_{i,t+1} = (u_{i,t+1}^1, u_{i,t+1}^2, \ldots, u_{i,t+1}^{3 \cdot M})$ for a particular individual, crossover is given by

$$u_{i,t+1}^j = \begin{cases} v_{i,t+1}^j & \text{if } r_j \leq C_R \vee j = r_i \\ x_{i,t}^j & \text{if } r_j > C_R \wedge j \neq r_i \end{cases} \tag{11}$$

where $C_R \in [0,1]$ denotes the crossoverrate and $r_j, r_i \in [0,1]$ random numbers. The new individual $x_{i,t+1}$ is obtained by

$$x_{i,t+1} = \begin{cases} u_{i,t} & \text{if } f(u_{i,t}) < f(x_{i,t}) \\ x_{i,t} & \text{otherwise.} \end{cases} \tag{12}$$

7 Results

For an extensive evaluation of our algorithms we created artificial test-instances. Real stellar signals typically do not only contain the rough (nearly) rectangular signals from the transiting planet, but also portions of stellar jitter and measurement errors. We take this into account by adding Gaussian random variables to each data point in the artificial signal. We thus create three instances for each configuration: one strictly rectangular signal and two noisy ones with different standard deviations.

Table 1 shows the results of 50 independent runs for various test instances. The first part shows the results for single signals, whereas the second part contains two-planet signals. For each algorithm we report the percentage of times where optimal solutions have been obtained and the average running times. Each column contains three values corresponding to signals without noise and with noise of $\sigma = 100$ and $\sigma = 300$ for the particular instances respectively. For some instances no results are available (indicated by “n/a”), as the algorithm stopped prematuerly because of many solutions having lower fitness values than the solution of the artificial signal.

For both algorithms we set the number of maximum iterations to 1000. We did not impose a time limit, but runs have been stopped when the global optimum was found. With “global optimum” we refer to a solution which is close

Table 1. Test-instances and corresponding success ratios of evolution strategy and differential evolution and average running times

Instance-	Parameters				(50/100,500)-ES		DE ($\|P\| = 200$)	
name	p	l	d	τ	(% opt.)	$t_{\text{avg}}[s]$	(% opt.)	$t_{\text{avg}}[s]$
art-100	1.0	0.10	100.0	0.5	100,100, 62	34, 34,311	100,100, 86	181,196,207
art-101	1.0	0.10	500.0	0.5	100,100,100	26, 19, 31	100,100,100	182,192,217
art-102	2.0	0.10	100.0	0.5	100,100, 28	12, 13,313	100,100, 28	129,135, 65
art-103	2.0	0.10	500.0	0.5	100,100,100	15, 12, 13	100,100,100	127,159,132
art-104	2.0	0.10	100.0	0.5	100,100, 94	12, 14, 48	100,100,100	108,109,140
art-105	2.0	0.05	500.0	0.5	100, 90,n/a	13, 49,n/a	100,100,n/a	103,115,137
art-106	1.0	0.05	500.0	0.5	98, 96, 70	22, 37,134	100,100,100	264,164,161
art-107	1.0	0.05	300.0	0.5	100,100, 46	23, 24,296	100,100, 86	159,155,180
art-108	1.0	0.05	100.0	0.5	94, 96,n/a	37, 27,n/a	100,100,n/a	162,164, 52
art-109	1.0	0.02	500.0	0.5	70, 72, 4	46, 60,576	100,100, 0	136,141,192
art-110	1.0	0.02	300.0	0.5	86, 76, 4	30, 61,498	100,100, 8	137,149,291
art-111	1.0	0.02	100.0	0.5	82, 26,n/a	38,208,n/a	100, 44,n/a	149,147,163
art-210	1.0/2.2	0.10/0.10	500.0/500.0	0.5/1.0	92, 90, 94	205,322,252	12, 12,n/a	526,531,602
art-211	1.0/2.2	0.10/0.10	500.0/300.0	0.5/1.0	98, 92, 80	314,318,422	56, 36, 8	518,527,616
art-212	1.0/2.2	0.10/0.05	300.0/500.0	0.5/1.0	78, 78, 48	322,440,658	20, 12, 28	479,453,475
art-213	1.0/2.2	0.05/0.05	500.0/500.0	0.5/1.0	88, 88, 46	374,601,663	0, 0, 0	481,500,474
art-214	1.0/7.5	0.05/0.20	400.0/500.0	0.5/1.0	54, 52, 40	316,306,396	32, 28, 14	217,328,340
art-215	1.0/7.5	0.10/0.20	400.0/500.0	0.5/1.0	72, 78, 72	348,339,312	100,100, 98	607,597,446
art-216	1.0/3.1	0.05/0.10	400.0/500.0	0.5/1.0	56, 60, 4	316,483,735	12, 6, 2	351,355,426
art-217	1.0/3.1	0.05/0.10	500.0/400.0	0.5/1.0	66, 76, 18	382,525,720	0, 0, 4	447,466,418
art-218	1.0/3.1	0.05/0.10	500.0/300.0	0.5/1.0	82, 72, 22	594,770,831	0, 0, 6	451,477,472
art-219	1.0/3.1	0.05/0.10	500.0/200.0	0.5/1.0	74, 76, 4	744,647,807	0, 2, 0	481,479,484

to the artificial signal and has the same (or lower) objective function value. Altough unlikely, better solutions might exist, i.e. solutions where arbitrary fitting of the noise yields lower deviations to the observations than the original imposed signal. Such situations are indicated by "n/a" in Table 1, as the algorithm is prematurely stopped in these cases.

For ES we used the parameter setting $\mu = 100, \lambda = 500$, and $\hat{\mu} = 50$. Prior to mutation we performed intermediate recombination for the strategy parameters and parameters with a probability of 0.8. For the DE algorithm we used $F = 1, C_R = 0.5$ and a population size of 200.

For both algorithms we used the parameter-space reduction as described in Section 4 and an advanced method to speed up the fitness-function evaluation which is beyond the scope of this paper. The optimal calculation of the depths significantly improves the ability of the algorithm to improve existing solutions quickly. All tests have been performed on a heterogenous cluster mostly consisting of recent hardware like Intel Core2 Quad, Intel Xeon and Dual-Core AMD Opteron processors.

The results show that optimal solutions can be obtained with high probability and acceptable running times for these data instances. Although not part of this work, we want to emphasize that the algorithms have comparable performance on real data-instances obtained from the CoRoT space telescope, which are known to contain planetary signals. For this purpose we added additional artificial signals to selected data instances, as so far no multi-planet signals have been found in this data.

8 Conclusions and Future Work

Both algorithms, ES and DE, exhibit a good performance and robustness on the test-instances presented in Section 7. Generally the ES converges faster which is mainly due to the effectiveness of the self-adaptation mechanism regarding the particular structure of the solution space. Although the DE algorithm generally requires longer running times, for some instances higher success ratios are obtained. Hence, both approaches have a justification to be used in practice.

An important part of transit detection algorithms, not considered in this work, is to compute a value indicating the statistic significance of the resulting fit. Such a measure enables to distinguish real signals from signals containing just noise and non-periodic signals and obviously should keep the false-alarm probability to a reasonably small rate. Techniques, currently used for single-planet signals (e.g. see [4]) are not directly applicable to multi-planet fits obtained by this approach. Hence we currently simply use the ratio between the standard deviation of the obtained fit to the standard deviation of the raw data, or alternatively Student's t-test in order to test if the in-transit levels have significantly different values in comparison to the out-of-transit levels. More extensive (blind) testing needs to be performed to asses the reliability of these approaches, but also more elaborate techniques might be necessary. Nevertheless, it is likely that current indicators are already able to select a reasonable subset of candidates from the huge amount of real-world input-data being worth analyzed in more detail subsequently. The application of the presented algorithms to yet publicly available CoRoT data is ongoing.

References

1. Deeg, H., Belmonte, J.A., Aparicio, A., eds.: Extrasolar Planets. Cambridge University Press (2008)
2. Charbonneau, P.: Genetic Algorithms in Astronomy and Astrophysics. Astrophysical Journal Supplement **101** (1995) 309–334
3. Chwatal, A.M., Raidl, G.R.: Determining orbital elements of extrasolar planets by evolution strategies. In Moreno-Díaz, R., et al., eds.: Computer Aided Systems Theory – EUROCAST 2007. Volume LNCS 4739 of LNCS. (2007) 870–877
4. Kovács, G., Zucker, S., Mazeh, T.: A box-fitting algorithm in the search for periodic transits. Astronomy and Astrophysics **391** (2002) 369–377
5. Stellingwerf, R.F.: Period determination using phase dispersion minimization. Astrophysical Journal **224** (1978) 953–960
6. Bäck, T.: Evolutionary Algorithms in Theory and Practice. Oxford University Press, New York (1996)
7. Storn, R., Price, K.: Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization **11** (1997) 341–359
8. Glover, F., Laguna, M., Marti, R.: Fundamentals of scatter search and path relinking. Control and Cybernetics **29**(3) (2000) 653–684
9. Nelder, J., Mead, R.: A simplex method for function minimization. The Computer Journal (7) (1964) 308–313
10. Schwefel, H.P.: Numerical Optimization of Computer Models. Wiley, Chichester (1981)