# 1 Introduction

TISMO (TIme series Spot MOdeling program) attempts to solve an astronomical inverse problem : fitting a light curve of a spotted variable star. The model assumption is a spotted star : the stellar rotation causes light variation. The program outputs the geometrical parameters of the spots as a function of time. The program can also be used for modeling spotted stars in binary systems where the luminosities of the two active components are fairly similar.

The applied analytical model of Budding (1977) interprets the active regions as circular shape intensity anomalies (i.e. spots having higher or lower brightness compared to their environment) on the surface of the star. The location and the size of a spot can be unambiguously given by three parameters (longitude, latitude and angular radius).

It is easy to see that the same light curve can be obtained from different spot configurations using different number of spots. Consequently there are correlations among the parameters. Those correlations cause problems in fitting real light curves of a given spaced, finite amount of data with some magnitudes of error. On the basis of the above considerations and according to the test results (Kővári & Bartus, 1996) it is no use to apply more than $2-3$ spots. Essential but not geometrical parameters (spot temperature, uspotted brightness, inclination, limb darkening etc.) can also have strong effects on the geometrical parameters (for instance the spot temperature on the size of the spot). Thus, those quantities should be determined before the actual modeling as precisely as it is possible.

# 2 Modeling the data

For the fitting the Levenberg-Marquardt method (LMM) was applied since it works very well in practice and has become the standard of nonlinear least-squares routines. More detailed description of LMM can be found in Data Reduction and Error Analysis for the Physical Sciences (Bevington, 1969). Briefly, the best-fit parameters are determined iteratively by the minimalization of $\chi^2$. During the iteration first we are presumably far from the minimum of $\chi^2$, so it is worth taking a step down the gradient in the parameter space because it gives a faster, however, a less accurate effort. Sufficently close to the minimum of the $\chi^2$ the function can be locally characterized and approximated by a quadratic form, from which the smaller but more certain steps in the direction of the minimum can easily be computed. The transition between the two submethods is carried out automatically by changing a control parameter.

Since Budding's model is valid for phased light curves an extension is necessary to introduce for modeling the time series light curves. It is quite a simple operation. The program takes a given part of the light curve, phases it together and finds a fit as for the phased curves, then the algorythm proceeds on the light curve by an amount of time, and repeats the process again.

During the iterations a so-called curvature matrix is inverted. The curvature matrix is built up by the second derivatives of the $\chi^2$ with respect to the parameters. This inversion is a crucial part of the process because in case of strongly correlated parameters the matrix gets close to singularity especially near the minimum. Therefore, as a consequence of inaccurate matrix inversion, unreliable or bad solutions can be obtained. A development of the original Levenberg-Marquardt method is needed that diagnoses the problem and tends to solve it, too. The very powerful set of techniques for dealing with matrices that are either singular or numerically very close to singular, is known as *SVD = Singular Value Decomposition*. A matrix to be inverted can be rewritten as the product of an orthogonal matrix $\mathbf{U}$, a diagonal matrix $\mathbf{W}$ and the transpose of an orthogonal matrix $\mathbf{V}$. The inverse of the original matrix can easily be obtained as $\mathbf{V} \cdot diag[\frac{1}{w_j}] \cdot \mathbf{U^T}$. The so-called *condition number* (defined as the ratio of the largest $w_j$ to the smallest one) reports on the status of singularity. A matrix is *ill-conditioned* if its condition number is too large (or even infinite), that is, its reciprocal approaches the machine's floating point precision (Press et al., 1988). Such matrices cannot be treated by normal inversion methods. In those cases SVD offer a solution by setting $\frac{1}{w_j}$ to 0 for the too small $w_j's$.

As a result the correlation matrix is also given by the LMM method, which consists of standard deviations of the parameters as the diagonal and the correlation among them as the off diagonal elements.

The necessary compter routines come from Numerical Recipes In C (Press et al., 1988).

## 3 Working with the program

The program runs under UNIX by the command : tismo <inputfile>. The structure of the `inputfile` is as listed below. Of course the data can change according to the application. This example file and the result files for this inputfile can be found in `/d1/bartus/progs/tismo/sample/`.

```
Input_data_file:            inp.dat
Output_parameter_file:      out.par
Output_info_file:           out.inf
Output_fitted_curve_file:   out.ill
U:                          1.0
L1:                         1.0
USI:                        -0.740
kw:                         0.165
u:                          0.743
Epoch:                      40017.17
Period:                     24.4284
Number_of_spots:            2
Inclination:                68.0
Initial_parameters:         360.0     80.0    40.0
                            120.0     -80.0   50.0
Iterated_period:            1.2
Step:                       0.5
```

A short description of the records:

The `Input_data_file` should be a two-column file with *mean Julian dates of the modeled interval* in the first column and *magnitudes* in the second one. The program writes the resulting parameters into the `Output_parameter_file`, where the columns are ordered as : *Julian date*, $\lambda_1$, $\beta_1$, $\gamma_1$, $\lambda_2$, $\beta_2$, $\gamma_2$, ..., *standard error of* $\lambda_1$, *standard error of* $\beta_1$, *standard error of* $\gamma_1$, *standard error of* $\lambda_2$, *standard error of* $\beta_2$, *standard error of* $\gamma_2$,..., *0 or −1, 0 or −1*,. where $\lambda_j$, $\beta_j$ and $\gamma_j$ are the longitude, latitude and angular extension of the j*th* spot respectively. The numbers of the last two columns indicate wheter or not the curvature matrix was close to the singularity during the iteration *(−1 (yes) or 0 (no))* and wheter or not that occured during the last inversion when the correlation matrix was computed *(−1 (yes) or 0 (no))*. The least-square values and all the error messages displayed at running time are written into the `Output_info_file` and the data of the fitted light curve into the `Output_fitted_curve_file`. `U` is the fiducial intensity, `L1` is the fractional luminosity of the star on which the spot is supposed to exist. `L1` is equal to 1 for single stars. `USI` is the unspotted magnitude, `kw` is the flux ratio, `u` is the limb darkening. `Epoch` and `Period` should be given in *days*. The program fit n spots given by the `Number_of_spots`. The `Inclination` is in *degrees*. `Initial_parameters` ($\lambda_j$, $\beta_j$, $\gamma_j$) are also in *degrees* and are the starting values of the iteration. They should be written *row by row*. As the process was explained in Section 2 the program reads `Iterated_period` amount of data from the light curve, finds a fit, and steps further by `Step` (both of them should be given in *phase units*).

When the entire light curve is modeled, the opportunity is offered to display the results graphically by an X11 window, and to make a hardcopy of the plots as well. Those values should be treated carefully. In the panels of the parameters open circles will indicate if spot overlap occured during the iteration. Errorbars are also marked. An example is shown on the coversheet of this manual.

## 4 Troubleshooting

Running the program different error messages may appear. User is warned if `Initial_parameters` are overlapping or if overlap occurs during the iteration. In the latter case the either `Initial_parameters` or the `Iterated_period` or the `Step` should be modified. The modification of the last two also may help when there are large errors or many singularities, or the fit looks quite ugly etc. The probability of such problems grow in cases of poorly covered time intervals. Altogether, the operation request some experience.

In case of any problem, improving idea, or opinion concerning the program, please send an e-mail to `bartus@buda.konkoly.hu`.

## 5 References


- Bevington, P. R., 1969, Data Reduction and Error Analysis for the Physical Sciences (New York: McGraw-Hill), Chapter 11
- Budding, E., 1977, Ap&SS, 48, 207
- Kővari, Zs. & Bartus, J, 1996, A&A submitted
- Press, W.H., Brian, P.F., Teukolsky, S.A., Vetterling, W.T., 1988, Numerical Recipes in C, Chapter 2, 14