Developer's guide

for reference implementation of the H,G_1,G_2 and H,G_{12} photometric systems

Antti Penttilä
Department of Physics, University of Helsinki
version 2012-11-01

1 Introduction

This document will give guidelines on how to create a 'reference implementation' of the H,G_1,G_2 tools into new language/environment. We encourage to create and distribute also other tools for H,G_1,G_2 system, but a reference implementation must follow some rules. First, the H,G_1,G_2 tools should be programmed as a library or similar structure that can be called from other programs. In that way the functionality of the H,G_1,G_2 system is created, but data processing, I/O etc. are left to the program that calls this library. The functions that the library should provide are listed in Sec. 2. You should also provide a simple example program that will use the library and fit one target, the asteroid (44) Nysa for example (see Sec. 4). Please document your code. See Sec. 3 for notes on the mathematical background of the procedures needed in the library.

The H,G_1,G_2 system is introduced in Muinonen et al. (2010) [1] and we hope that you will cite that when publishing results using this system. Furhtermore, if one uses these official H,G_1,G_2 tools, also the corresponding code should be cited. Suitable citation style could be, e.g., "Fortran implementation of H,G_1,G_2 tools [2]".

2 Public procedures in reference implementation

Next, I will list procedures (i.e. functions/subroutines) that *must be implemented* in reference implementation of H,G_1,G_2 system. The details of the implementation are left to the developer and can vary from language/environment to another. For example, how optional arguments are handled in particular environment, how vectors and matrices are returned as a result, how errors are handled etc. can vary. However, the following functionality must be implemented in one way or another:

Procedure `form_base`

This is called once before other procedures can be called. It will initialize the basis functions for the system. Current basis functions are presented in [1] and in Sec. 3. Note that one of the ideas behind the new photometric system defined with splines is that it can be revised if needed. The Fortran reference implementation has already optional version of `form_base` with filename as input argument. The basis function definitions would be read from that file, but the implementation of that procedure

is currently empty and will return error when called. So, initialization with default system is fine for now.

Procedure `fit_HG1G2`

Will find best-fit parameters a_1, a_2, a_3 for the data and return them, converted to H,G_1,G_2 parameters.

Inputs:

`data` The data to fit the system. List of (phase angle, observed magnitude)-pairs.

`weights` Optional argument. The fit must be implemented in such a way that it is possible for observed magnitudes to *i*) have no error estimate, *ii*) have constant default standard error of 0.03 mag, *iii*) have constant user-supplied error in magnitude scale, or *iv*) have individual error estimates in magnitude scale.

Outputs:

`parameters` Result of (weighted) linear fit, converted to H,G_1,G_2values.

`rms-value` The residual-mean-square value of the fit in magnitude scale (see Sec. 3.4).

`covariance_matrix` Covariance matrix of the best-fit linear parameters (a_1, a_2, a_3).

Procedure `fit_HG12`

Will find best-fit parameters c_1, c_2 for the data and return them, converted to H,G_{12} parameters.

Inputs:

`data` The data to fit the system. List of (phase angle, observed magnitude)-pairs.

`weights` Optional argument. The fit must be implemented in such a way that it is possible for observed magnitudes to *i*) have no error estimate, *ii*) have constant default standard error of 0.03 mag, *iii*) have constant user-supplied error in magnitude scale, or *iv*) have individual error estimates in magnitude scale.

Outputs:

`parameters` Result of (weighted) linear fit, converted to H, G_{12}-values.

`rms-value` The residual-mean-square value of the fit in magnitude scale (see Sec. 3.4).

`covariance_matrix` Covariance matrix of the best-fit linear parameters (c_1, c_2).

Procedure `simulate_errors`

Will simulate the error estimates for fitted H,G_1,G_2 or H,G_{12} parameters. Must be implemented for both three and two-parameter systems.

Inputs:

`parameters` Best-fit parameters, as returned from fitting.

`covariance_matrix` Covariance matrix of the best-fit linear parameters, as returned from fitting.

`simulation_n` Optional argument. Number of simulated values for error estimation. Default value in current Fortran code is 100 000.

Outputs:

`error_matrix` Simulated errors for (H,G_1,G_2) or (H,G_{12}), and for the slope parameter k. For each parameter, the simulated mean and median values must be given, and also the following percentiles: 0.0013499, 0.0227501, 0.158655, 0.841345, 0.97725, and 0.99865. These percentiles correspond to limits of 3-, 2-, and 1-sigma or 99.73%, 95.45% and 68.3% confidence intervals for the value.

Procedure `system_version`

Will return the version number (date) of the basis function system in use. This is mainly for future, since we have only one system (version number 20101000) in use.

Outputs:

`version` System version of the basis functions, integer code. The original system in [1] is 20101000.

`low_limit` Optional. The smallest phase angle where the basis functions are defined. In v.20101000 this is 0°.

`high_limit` Optional. The largest phase angle where the basis functions are defined. In v.20101000 this is 150°.

Procedure `give_base_values`

This procedure is mainly for debugging. It will return the values of the basis functions Φ_1, Φ_2, Φ_3 at given phase angle.

Inputs:

`phase_angle` Phase angle where basis functions are to be evaluated

Outputs:

`function_values` Values of the basis functions Φ_1, Φ_2, Φ_3 at given phase angle.

3 Implementation notes

3.1 Basis functions

3.1.1 H,G_1,G_2 system

In H,G_1,G_2 system the observed (visual) magnitudes V are modeled with

$$V(\alpha) = H - 2.5\log_{10}(G_1\Phi_1(\alpha) + G_2\Phi_2(\alpha) + (1 - G_1 - G_2)\Phi_3(\alpha)), \quad (1)$$

where α is phase angle, Φ_i's are basis functions, and H,G_1,G_2 are parameters. This system can be converted to following linear system in flux density space:

$$F(\alpha) = a_1\Phi_1(\alpha) + a_2\Phi_2(\alpha) + a_3\Phi_3(\alpha) \quad (2)$$

with transformations

$$\begin{aligned} F(\alpha) &= 10^{-0.4V(\alpha)} & a_1 &= 10^{-0.4H}G_1 \\ a_2 &= 10^{-0.4H}G_2 & a_3 &= 10^{-0.4H}(1 - G_1 - G_2). \end{aligned} \quad (3)$$

The inverse transformations from linear parameters are

$$H = -2.5\log_{10}(a_1 + a_2 + a_3), \quad G_1 = \frac{a_1}{a_1 + a_2 + a_3}, \quad G_2 = \frac{a_2}{a_1 + a_2 + a_3}. \quad (4)$$

The basis functions Φ_i are the key factors in this system. They are defined piecewise, and can consist of *i*) a constant value, *ii*) a first degree linear function, or *iii*) a cubic spline function. You should prepare for the fact that the basis functions may change in future releases of the system. You should only assume that there are three basis functions and that they can have piecewise defined parts *i*)–*iii*). The current, original and, at the moment, only version of H,G_1,G_2 basis functions is:

$$\Phi_1(\alpha) = \begin{cases} 1 - \dfrac{6\alpha}{\pi}, 0^\circ \leq \alpha \leq 7.5^\circ \\ \xi_1(\alpha), 7.5^\circ < \alpha \leq 150^\circ \end{cases} \quad (5)$$

$$\Phi_2(\alpha) = \begin{cases} 1 - \dfrac{9\alpha}{5\pi}, 0^\circ \leq \alpha \leq 7.5^\circ \\ \xi_2(\alpha), 7.5^\circ < \alpha \leq 150^\circ \end{cases} \quad (6)$$

$$\Phi_3(\alpha) = \begin{cases} \xi_3(\alpha), 0^\circ \leq \alpha \leq 30^\circ \\ 0, 30^\circ < \alpha \leq 150^\circ \end{cases}, \quad (7)$$

where ξ_i's are cubic splines, see Sec. 3.2.

3.1.2 H,G_{12} system

In H,G_{12} system the observed (visual) magnitudes V are modeled as in H,G_1,G_2 system in Eq. (1), but with extra condition that

$$G_1 = \beta_1 G_{12} + \beta_0, \quad G_2 = \gamma_1 G_{12} + \gamma_0. \quad (8)$$

There are two possibilities (low- or high-albedo object) for the actual values for coefficients β and γ, namely either

$$\beta_1 = 0.7527, \quad \beta_0 = 0.06164, \quad \gamma_1 = -0.9612, \quad \gamma_0 = 0.6270, \text{ or} \quad (9)$$

$$\beta_1 = 0.9529, \quad \beta_0 = 0.02162, \quad \gamma_1 = -0.6125, \quad \gamma_0 = 0.5572. \quad (10)$$

Thus, the H,G_{12} system has only two parameters. This system can also be transformed to linear in flux density space:

$$F(\alpha) = c_1\Gamma_1(\alpha) + c_2\Gamma_2(\alpha), \text{ where} \quad (11)$$

$$\Gamma_1(\alpha) = (\beta_0\Phi_1(\alpha) + \gamma_0\Phi_2(\alpha) + \Phi_3(\alpha) - \beta_0\Phi_3(\alpha) - \gamma_0\Phi_3(\alpha)), \text{ and} \quad (12)$$

$$\Gamma_2(\alpha) = (\beta_1\Phi_1(\alpha) + \gamma_1\Phi_2(\alpha) - (\beta_1 + \gamma_1)\Phi_3(\alpha)). \quad (13)$$

The inverse transformation to original parameters in this case is

$$H = -2.5\log_{10}(c_1), \quad G_{12} = \frac{c_2}{c_1}. \quad (14)$$

In practice, one fits the linear system in Eq. (11) two times using values in Eqs. (9) and (10), and chooses the one with smaller rms-value.

3.2 Splines

The basis functions are defined with the help of cubic splines. Cubic spline is piecewise third-order polynomial function that will pass all the given knot points k^i, which are pairs of (k_x^i, k_y^i) coordinates. Furthermore, the first and the second derivatives of the spline are continuous through the whole support of the spline, i.e. from k_x^1 to k_x^n. The spline between knots k^i and k^{i+1} can be presented in the form

$$\xi_i(x) = (1-t_i)k_y^i + t_i k_y^{i+1} + t_i(1-t_i)(a_i(1-t_i) + b_i t_i), \tag{15}$$

where

$$t_i = \frac{x - k_x^i}{k_x^{i+1} - k_x^i},\ a_i = d_i(k_x^{i+1} - k_x^i) - (k_y^{i+1} - k_y^i),\ b_i = -d_{i+1}(k_x^{i+1} - k_x^i) + (k_y^{i+1} - k_y^i) \tag{16}$$

and where d_i is the first derivative of ξ at k_x^i.

From the above one can see that if the derivatives d_i can be solved the spline is completely defined. There are three possibilities for conditions that will uniquely define the spline system: require zero second derivatives at k^1 and k^n (natural spline), give definite values for first derivatives at k^1 and k^n (clamped spline), or require that the first and second derivatives continue cyclically between k^n and k^1 (periodic spline). We use the second option of so-called clamped spline.

The solution to the spline system (i.e. vector of derivatives $\boldsymbol{d} = (d_1, \ldots, d_n)$) can be written as a linear system $\mathbf{A}\,\boldsymbol{d} = \boldsymbol{y}$, where the coefficient matrix $\mathbf{A}$ is tridiagonal. The entries in the matrix are:

$$A_{i,i} = \frac{2}{k_x^i - k_x^{i-1}} + \frac{2}{k_x^{i+1} - k_x^i}, \quad A_{i-1,i} = \frac{1}{k_x^i - k_x^{i-1}}, \text{ and} \tag{17}$$

$$A_{i,i+1} = \frac{1}{k_x^{i+1} - k_x^i} \quad \text{for } i = 2, n-1, \text{ and} \tag{18}$$

$$A_{1,1} = 1, \quad A_{n,n} = 1. \tag{19}$$

All the other elements in $\mathbf{A}$ are zero. Vector $\boldsymbol{y}$ is defined as

$$y_i = 3\left(\frac{k_y^i - k_y^{i-1}}{(k_x^i - k_x^{i-1})^2} + \frac{k_y^{i+1} - k_y^i}{(k_x^{i+1} - k_x^i)^2}\right) \quad \text{for } i = 2, n-1, \text{ and} \tag{20}$$

$$y_1 = d_1, \quad y_n = d_n, \tag{21}$$

where d_1 and d_n are known. Once the derivatives have been solved they can be inserted into Eqs. (15) and (16).

The current system of splines has the following knots and derivatives at endpoints:

α	ξ_1	ξ_2
7.5	0.75	0.925
30.0	0.33486	0.628842
60.0	0.134106	0.317555
90.0	0.0511048	0.127164
120.0	0.0214657	0.0223739
150.0	0.0036397	0.000165057

and

α	ξ_3
0.0	1.0
0.3	0.833812
1.0	0.577354
2.0	0.421448
4.0	0.231742
8.0	0.103482
12.0	0.0617335
20.0	0.016107
30.0	0.0

The first derivatives at first and last knots are $d_1^1 = -1.90986, d_n^1 = -0.0913286, d_1^2 = -0.572958, d_n^2 = -8.6573138 \times 10^{-8}, d_1^3 = -0.106301, d_n^3 = 0.0$.

3.3 Weighted linear fit for H,G_1,G_2 or H,G_{12}

The linear model in Eq. (1) or (11) can be solved using standard linear regression tools. If we mark the parameter vector by $\boldsymbol{\beta}$, the vector of observed magnitudes by $\boldsymbol{y}$, and the design matrix, i.e. matrix containing the values of the linear basis functions (Φ_1, Φ_2, Φ_3) in H,G_1,G_2 or (Γ_1, Γ_2) in H,G_{12} at observed phase angles, by $\mathbf{A}$, our model can be written as

$$\boldsymbol{y} = \mathbf{A}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{22}$$

where $\boldsymbol{\epsilon}$ are the errors of the fit. The solution $\hat{\boldsymbol{\beta}}$ is given by the normal equations, $\hat{\boldsymbol{\beta}} = (\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\boldsymbol{y}$. The covariance matrix of the estimate is proportional to $(\mathbf{A}'\mathbf{A})^{-1}$.

If we know the errors (standard errors σ, to be exact) of the observations, they can be included in the fit, and the solution is called weighted. It can be easily shown that the abovementioned formula apply to weighted system as well, as long as we replace $\boldsymbol{y}$ with $\boldsymbol{y}^*$, where $y_i^* = y_i/\sigma_i$, and $\mathbf{A}$ with $\mathbf{A}^*$, where $[A]_{ij}^* = [A]_{ij}/\sigma_i$. In case of known errors the covariance matrix is exactly $(\mathbf{A^{*\prime}A}^*)^{-1}$.

Please note that the fit is done in flux density scale. Therefore, if we know or assume the error σ_{mag} in magnitude scale for observation (α, y_{mag}), the corresponding error in flux density scale is

$$\sigma = 10^{-\frac{2y_{mag}}{5}} \left(10^{\frac{2\sigma_{mag}}{5}} - 1\right). \tag{23}$$

Furthermore, noticing that $y = 10^{-\frac{2y_{mag}}{5}}$ shows us that in the weighted analysis

$$y^* = \left(10^{\frac{2\sigma_{mag}}{5}} - 1\right)^{-1} \tag{24}$$

and the original observed magnitudes y_i will go into the Eq. (22) only through the weights σ_i in the matrix $\mathbf{A}^*$.

If the unweighted version of the fit is used, the covariance matrix must include the estimate of the residual variance (see next Sec.) σ^2. In that case the covariance matrix is $\sigma^2(\mathbf{A}'\mathbf{A})^{-1}$.

3.4 Residual variance

The residual variance of the fit is computed from the fit residuals e_i. With predicted values $\hat{\boldsymbol{y}} = \mathbf{A}\hat{\boldsymbol{\beta}}$, the residuals in flux density space are $e_i = y_i - \hat{y}_i$ and the residual variance is $\sigma^2 = \boldsymbol{e}' \cdot \boldsymbol{e}/(n-k)$, where $k = 2$ for H,G_{12} and $k = 3$ for H,G_1,G_2.

In magnitude space, however, the residual variance can be computes as

$$\sigma^2_{mag} = \frac{25}{4(n-k)} \sum_i [\log_{10}(y_i/\hat{y}_i)]^2. \tag{25}$$

3.5 Error simulation

The linear fit (Sec. 3.3) will produce best, most probable values for linear parameters $\hat{\boldsymbol{\beta}} = (a_1, a_2, a_3)$ (for H,G_1,G_2) or $\hat{\boldsymbol{\beta}} = (c_1, c_2)$ (for H,G_{12}). The conversion to magnitude space, however, is not linear. Therefore converting directly to H,G_1,G_2 or H,G_{12} parameters will not give best, unbiased estimates. Better estimates can be computed with straightforward simulation. The linear fit will produce both $\hat{\boldsymbol{\beta}}$ and its covariance matrix $\mathbf{C} = (\mathbf{A}'\mathbf{A})^{-1}$ (without weights the residual variance σ^2 is also needed in the covariance matrix). It is quite safe to assume that $\hat{\boldsymbol{\beta}}$ obeys multinormal distribution. When a large number of samples $\boldsymbol{\beta}_i$ is drawn from normal distribution, and converted to H^i, G_1^i, G_2^i, the median or mean value of those gives the best estimate for H,G_1,G_2. Furthermore, the percentiles of H^i, G_1^i, G_2^i can be used as confidence intervals for H,G_1,G_2.

Drawing samples from multinormal distribution $\mathcal{N}(\hat{\boldsymbol{\beta}}, \mathbf{C})$ is quite simple, but if there are no tools for that in the environment where you are implementing the code, you will need $\mathbf{V}$ such that $\mathbf{C} = \mathbf{V}\mathbf{V}'$, either via eigenvalue decomposition ($\mathbf{C} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}' = (\mathbf{U}\boldsymbol{\Lambda}^{1/2})(\mathbf{U}\boldsymbol{\Lambda}^{1/2})'$) or directly from Cholesky decomposition. Once you have $\mathbf{V}$ and a vector $\boldsymbol{z}^i$ of independent normal random deviates from $\mathcal{N}(0,1)$, the dependent random sample can be computed as

$$(H^i, G_1^i, G_2^i) = \hat{\boldsymbol{\beta}} + \mathbf{V}\boldsymbol{z}^i. \tag{26}$$

4 Example data

With the example data you can test your implementation of the H,G_1,G_2 tools. We use observations of asteroid (44) Nysa, they should be available on the website of the H,G_1,G_2 tools. The 23 observations are also given here as (phase angle [in degrees], visual magnitude)-pairs: (0.17, 6.911), (0.36, 6.972), (0.63, 7.014), (0.75, 7.033), (0.98, 7.052), (1.23, 7.08), (1.62, 7.105), (2.02, 7.126), (4.95, 7.235), (8.27, 7.304), (9.78, 7.341), (11.59, 7.385), (12.94, 7.425), (13.2, 7.426), (13.27, 7.427), (13.58, 7.433), (13.81, 7.437), (13.89, 7.434), (17.16, 7.511), (18.52, 7.524), (19.0, 7.551), (19.4, 7.545), (21.47, 7.599).

With these observations you should receive H,G_1,G_2 linear parameters $(a_1, a_2, a_3) = (8.591298 \times 10^{-5}, 1.162542 \times 10^{-3}, 4.823764 \times 10^{-4})$, which are $(H, G_1, G_2) = (6.904$

$36326, 0.04963684, 0.67166677)$ when converted to magnitude scale. Residual variance is 0.00928867 mag and covariance matrix is

$$\mathbf{C} = 10^{-8} \begin{pmatrix} 1.838463 & -1.192184 & -0.954880 \\ -1.192184 & 0.783019 & 0.600461 \\ -0.954880 & 0.600461 & 0.587111 \end{pmatrix}$$

With H,G_{12} system the fit should give $(c_1, c_2) = (1.744196 \times 10^{-3}, -1.144900 \times 10^{-4})$, which are $(H, G_{12}) = (6.89601187, -0.06564058)$ when converted to magnitude scale. Residual variance is 0.00965089 mag and covariance matrix is

$$\mathbf{C} = 10^{-9} \begin{pmatrix} 0.405839 & 0.668774 \\ 0.668774 & 1.511718 \end{pmatrix}$$

When simulating errors for these estimates, you should receive approximately (some variation is expected) for H,G_1,G_2 fit:

	H	G_1	G_2	k
mean	6.90464149	0.05069915	0.67134989	-0.65956207
median	6.90445042	0.04988134	0.67162814	-0.66536858
3-σ low	6.84076255	-0.17786513	0.53563757	-1.04724612
2-σ low	6.86251247	-0.10443993	0.58266152	-0.92723517
σ low	6.88336178	-0.02843426	0.62750429	-0.80111904
σ high	6.92599531	0.12955407	0.71542704	-0.51759176
2-σ high	6.94770094	0.21127966	0.75783444	-0.35782130
3-σ high	6.96974101	0.29658309	0.79906285	-0.18799414

and for H,G_{12} fit (G_{12} is converted to G_1 and G_2):

	H	G_1	G_2	k
mean	6.89605285	0.06155389	0.62710997	-0.69245221
median	6.89597405	0.06155385	0.62711001	-0.69245214
3-σ low	6.85863026	0.06146565	0.62699700	-0.69262821
2-σ low	6.87110729	0.06149520	0.62703504	-0.69256893
σ low	6.88349590	0.06152470	0.62707250	-0.69251058
σ high	6.90857687	0.06158322	0.62714723	-0.69239416
2-σ high	6.92106342	0.06161256	0.62718489	-0.69233549
3-σ high	6.93420209	0.06164231	0.62722259	-0.69227676

References

[1] K. Muinonen, I.N. Belskaya, A. Cellino, M. Delbò, A.-C. Levasseur-Regourd, A. Penttilä, and E.F. Tedesco. A three-parameter magnitude phase function for asteroids. *Icarus*, 209(2):542–555, 2010.

[2] A. Penttilä. Fortran 2003 implementation of H,G_1,G_2 tools, 11 2012. Computer algorithm, retrieved from `http://wiki.helsinki.fi/display/PSR/HG1G2+tools`.