# Storage formats of SH-coefficients in MATLAB

Nico Sneeuw

Institut für Astronomische und Physikalische Geodäsie

Technische Universität München

14 May 1999

## SC-Format: working storage

When using MATLAB in problems related to spherical harmonic (SH) coefficients, one should store the $\bar{C}_{lm}$'s and $\bar{S}_{lm}$'s in some matrix format. In my MATLAB `*.m` files I'm applying the following conventions, called *SC-triangle format*, or *SC-format* to be short. Suppose we have a set of real-valued SH-coefficients, complete to maximum degree $L$. Then the SC-format stores the coefficients in the following $(L+1) \times (2L+1)$ matrix:

$$\begin{bmatrix} & & & & C_{00} & & & & \\ & \epsilon & & S_{11} & C_{10} & C_{11} & & \epsilon & \\ & & S_{22} & S_{21} & C_{20} & C_{21} & C_{22} & & \\ & & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \\ S_{LL} & \cdots & S_{L2} & S_{L1} & C_{L0} & C_{L1} & C_{L2} & \cdots & C_{LL} \end{bmatrix}$$

Remarks:

*i).* Complex-valued coefficients $\bar{K}_{lm}$ could be stored in SC-format as well. In that case replace $\bar{C}_{lm}$ by $\bar{K}_{lm}$ for positive order $m$, and $\bar{S}_{lm}$ by $\bar{K}_{l,-m}$.

*ii).* The upper left and right corners should be filled with zeros. For some numerical purposes, it could be handy though, to use a small number $\epsilon$, e.g. $\epsilon = 10^{-20}$. For instance computing SNR per coefficient will cause trouble if $\epsilon = 0$.

*iii).* For geodesy the first two rows, corresponding to $l = 0$ and $l = 1$, are not necessary and the SC-format could have been defined as a $(L-1) \times (2L+1)$ matrix. However, for the sake of generality, $l = 0$ and $l = 1$ are included in the SC-format convention.

*iv).* In general, if the SC-matrix would be called `A`, a single $\bar{C}_{lm}$ must be referred to as `A(l+1,lmax+1+m)`, and the corresponding $\bar{S}_{lm}$ as `A(l+1,lmax+1-m)`. Zonal coefficients

are in column $L+1$, so referring to the even zonals starting with $C_{20}$ e.g., would look like `A(3:2:lmax+1,lmax+1)`. The low pass filter, which sets a new maximum degree L2, would be represented by the SC-matrix `A(1:l2+1,lmax+1-l2:lmax+1+l2)`. And so on.

*v).* The name SC-format is chosen since the $\bar{S}_{lm}$-coefficients are at the LHS and the $\bar{C}_{lm}$'s at the RHS of the matrix.

## CS-Format: storage storage

Working with the SC-format under MATLAB is convenient. For actual storage of the coefficients on disk it is less efficient, since all the $\epsilon$'s have to be stored as well. In principle an $(L+1)^2$ matrix should do. For *storage* storage therefore I define the *CS-square format* or *CS-format* in short, according to:

$$\begin{bmatrix} C_{00} & S_{11} & S_{21} & \cdots & S_{L1} \\ C_{10} & C_{11} & S_{22} & \cdots & S_{L2} \\ C_{20} & C_{21} & C_{22} & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & S_{LL} \\ C_{L0} & C_{L1} & C_{L2} & \cdots & C_{LL} \end{bmatrix}$$

Remarks:

*i).* Compared to the SC-format, the CS-one is twice as economical in storage, at the expense of being less accessible. But maybe this is only a matter of taste.

*ii).* A number of coefficient sets are stored in SC-format (and in MATLAB's binary `*.mat`-format). Momentarily under `/users/sneeuw/matlab/matdata`. If this directory is on the MATLAB searchpath, a simple `load jgm1s` would return two $61 \times 61$ matrices: one containing the coefficients, the other containing the corresponding standard deviations.

*iii).* The name CS-format is chosen because the $\bar{C}_{lm}$'s are at the lower left corner and the $\bar{S}_{lm}$'s at the upper right one.

## Conversions

Of course it is quite easy to convert both formats into eachother. I wrote 2 small MATLAB routines: `sc2cs.m` and `cs2sc.m`, to be found (momentarily) in directory `matlab/physgeo/shtools`. Note that the CS $\rightarrow$ SC conversion allows to define a background $\epsilon$ value. Here they are:

```
function cs = sc2cs(field)

% SC2CS(FIELD) converts the rectangular (L+1)x(2L+1) matrix FIELD, containing
%       spherical harmonics coefficients in /S|C\ storage format into a
%       square (L+1)x(L+1) matrix in |C\S| format.
%
% Nico Sneeuw
% Munich, 22/07/94

[rows,cols] = size(field);
lmax = rows -1;
if cols ~= 2*lmax+1, error('Matrix dimensions must be (L+1)x(2L+1).'), end

c  = field(:,lmax+1:2*lmax+1);
s  = [zeros(lmax+1,1) field(:,1:lmax)];
cs = tril(c) + triu(rot90(s),1);



function sc = cs2sc(field,backval)

% CS2SC(FIELD,backval) converts the square (L+1)x(L+1) matrix FIELD, containing
%       spherical harmonics coefficients in |C\S| storage format into a
%       rectangular (L+1)x(2L+1) matrix in  /S|C\format.
%       The argument backval is optional and describes the matrix entries,
%       where m > l. Default is 1e-20!
%
% Nico Sneeuw
% Munich, 22/07/94

if nargin == 1, backval = 1e-20; end
[rows,cols] = size(field);
lmax = rows -1;
if cols ~= rows, error('I expect a square matrix.'), end

c  = tril(field);
s  = rot90(triu(field,1),-1);
mask = backval*ones(lmax+1,2*lmax+1);
mask = mask - trapstrip(mask);
sc = mask + [s(:,2:lmax+1) c];
```