# MV Phenot

Where MM makes some changes he comment the original MV line starting with a “;MM”. If the changes are extended to a part of code “MM+” and “MM-“ are used.

### Detecting more than a growing season

The parameter “pow\_tr” (line 175, phenot\_p2\_stat2.pro), the desired statistical significance level to detect peaks, is a critical one: for example, if set to 0.0001 it does not detect two GS in “Math-Double-Sine\_ZZ-zzz\_c1\_19961997\_f10\_d20090311.xdat”; when set to 0.005 it does.

Could it be interesting to make a feedback between this parameter and the model fitting? For example, if the model cannot be fitted (maybe because there are two growing seasons as in “Math-Double-Sine\_ZZ-zzz\_c1\_19961997\_f10\_d20090311.xdat”) , the process is restarted using a lower threshold.

### Other files tested

See phenot.job

## Checks after 25 feb discussion

We are focusing on the results we get with the well-behaved fAPAR profile of:

SeaWiFS\_IT-PT1\_c03\_19972006\_f10\_d20090108.xdat

### Is there any difference with the IDL Lomb routine?

There is no macroscopic differences between the two power spectra.

### Is Lomb Norm using the incomplete seasons at the beginning and end of the time series?

Yes, it is using the whole set of data. By manually removing these uncompleted tails, numeric results are slightly different but 2 growing seasons are detected in both cases.

### Can we think a feedback between model fit and Lob (if Lomb suggests two GS and than the model is able to fit only one..)?

### Check the IDL implementation of Lomb comparing with the original MATLAB code. Run the two over some datasets.

### Try the Jan version of Lomb

### Use autocorrelation instead of Lomb? Use both?

The autocorrelation works fine. I am using it now, ngspy is now set to the number found by autocorrelation.

### Lomb: the significance level is not based on noise present in the data.. study the effect of noise and including it?

## CHANGES

### Use autocorrelation instead of Lomb? Use both?

The autocorrelation works fine. I am using it now (line 136 of phenot\_p2\_stat2), ngspy is now set to the number found by autocorrelation.

### ITMAX in CURVEFIT

The algorithm was not able to fit some of the GS referring to my Niger profiles. I Increased the number of iteration ITMAX to 100 (it is 20 by default) in phenot\_p3\_pdhtf (line 364) and the problem seems to be solved.

### Minimum variability, faprangeminthresh

The variable “faprangeminthresh” (to be compared with “(fap95pctl - fap05pctl)”) was set to 0.15 preventing a significant number of Niger profile (with indeed very low fAPAR oscillation, i.e. max peak of about 0.20) to be analysed. I reduced this threshold to 0.10 to extend the analysis to all pixels.

### Iniend

It appears that sometimes the iniper procedure selects a subperiod where to search for a growing season that is too short (no growing season at all is present in the subperiod). This is because the timper parameter is set too short in order to avoid missing a short season. I adjusted the timper as 1.5\*timper. The idea now is to adjust timper, doubling it, if the optimization process fails. Now , if optimization fails, we go back to iniper with a timper=alpha\*timper, where alpha = 2. This solution seem to work for the moment.

But it is too sensitive to this method…

### Abrupt end of the season

Phenot\_p3\_phdf contains a check on the possible abrupt end of GS (line 577). As a result, if fapar [stopgs] GE fap50pctl the stop of GS was moved forward until it was below fap50pctl.

In Niger, fap50pctl is very high compared to the time series. As a result, the EOS was move forward artificially. I used fap75pctl instead.

### End of the record

When using the autocorrelation it performs quite well when he has data but keep on searching for growing season after the last complete one. That is, for example in the case of our IT file, it finds two seasons after the last complete one, and it is not able to fit the model.

The only check is done on in pdhtf line 287 as a diagnostic [; Make sure the first and last good values within that new period are at least as low as the median (50th) percentile of the entire distribution, and that there is a local maximum; above that value, to avoid trying to fit a growing season through a partial record where only the initial rise (or only the final decline) is available (e.g., near the start; or end of the record) ]. For now, this is only a diagnostic.

Now I check that in phenot\_p3\_iniper if the iniend is reduced because the end of file was found. If iniend exceed the end of the record and the above conditions are met, the optimization is not performed. In NRT we’ll have to face this problem.

### 1 or 2 seasons

When we have two distinct seasons it works well (see VGs\_R282C238\_c1\_20002010\_f10\_d20110304.xdat, which is smoothed, see below).

When they are too close (similar to none) or as in VGs\_R182C237\_c1\_20002010\_f10\_d20110304.xdat, where we don’t find always two seasons (some years have 1 season), the autocorrelation sees only 1 season (instead the lomb sgarg would see always 2). In any case, if I manually set 2 seasons, it is not always able to detect them.

TRY to see what happen if I always set two season even if there is one?

### Smoothing and gapfilling

When data are not smoothed (VGs\_R282C238\_c1\_20002010\_f10\_d20110304.xdat and its non smoothed version, VGT) it may fail to the detect some seasons that otherways, using smoothed data, it would detect.

## Modification to allow resume

The objective is to avoid re-running on the entire time series when new data are available. The limit is that that basic observations will not be changed (ngspy) and only pixels with globstat retcode = 0 will be updated. For a complete revision the algorithm has to be re-run from scratch.

After a run the situation is:

* A fAPAR time series of 0 to n-1 dekads has been investigated (n is the number of bands in the bil fapar file)
* It is known if the pixel has variability (globstat retcode =0) and the ngspy.
* Xoffset has to be saved, will saved in xoffset
* The number of years evaluated (it’s the number of bands of phenol products). n\_full\_solar\_years=floor((ndec-(xoffst[0]+1))/36.0) The last two can be incomplete..
* Pheno File must be open and copied to updated files containing more bands (eventually, if the new years is GT than the old number of bands)
* Computations has to start at y=ystart (phenot\_p3\_pdhtf\_mm line 144)

# To do:

## FAILURE IN ONE SEASON

Re-optimize the season with mean values as first guess

## NRT

The objective is to find the SOS when it happens.

Use some statistic to the detect how it grows?

## Detection of the windows for model inversion

Given a time series of fAPAR defined over the decadal time interval [0, .., d]

**1.** Compute the autocorrelation or Lomb Scargle to determine the relevant frequencies. One growing season: 1 frequency of period 36 decades. Two growing season: two major frequencies (period 36, and 18), or three major frequencies if the second one is not 18 (period 36, period T, period 36-T).

|  |
| --- |
| For example consider the following cycles:  One season, T=36  Two seasons: T=36, 11, 36-11. NOTE THAT HERE AUTOCORR DOES NOT WORK WELL!! |

**2.** Compute the average year on the data series starting form the first available decade

|  |
| --- |
| We may get something like this (one season)  Or like this (two seasons) |

**3.** locate the overall maximum

|  |
| --- |
| Position of the overall max = Xmax = 35 |

*If there is on growing season:*

**4(1).** locate the minimum (or average position of fapar below a threshold/pecentile)

|  |
| --- |
| Position of the overall min Xmin= 17 |

**5(1)**. The model optimization window is [i\*36+ Xmin, (i+1)\*36+ Xmin], where i is the progressive number of years to be investigated (i=[0, ..])

*If there are two growing seasons:*

**4(2).** now we know that we can have a second max at Xmax +11 or Xmax -11, check which of the two is the second maximum

|  |
| --- |
| Position of the second max = Xmax2 = 9 |

**5(2).** locate the two minima between the two maxima

|  |
| --- |
| Position of the first min (between 35 and 9) Xmin1 = 5 and second one (between 9 and 35) Xmin2= 20 |

**6(2).** The series starts at 5, with a first window [i\*36+ Xmin1, i\*36+ Xmin2], and a second window [i\*36+ Xmin2, (i+1)\*36+ Xmin1]

Dear Michele,

The interpretation of the Growing Season (GS) simulation results you sent me yesterday is made complicated by at least two problems:

- The first and most important is that a large fraction of the simulated calendar year (about 23 decades out of 36 in the case of a single GS and 16 decades out of 36 in the case of a double GS) is assigned null FAPAR values. This feature artificially increases the autocorrelation at small lags and reduces the autocorrelation at longer lags, until you reach the annual cycle of course. In fact, you can see that the autocorrelogram (as exhibited in the JPEG file) is not even capable of detecting a secondary peak beyond the annual frequency.

- The other issue is that your second simulated GS is so short (about 7 decades) that it does indeed look like an appendix or an extension of the previous GS, at least in statistical terms: the separation between the two successive GS could almost be generated by a passing cloud during a single decade... Is it actually possible to grow and harvest a crop in 70 days (barely over two months), especially starting from bare ground (since the starting value for the 'second' GS is 0)?

If you distribute your simulated GS more evenly during the year, or at least allow each GS to extend long enough in time to occupy most (not necessarily all) of the calendar year, you'll see that the autocorrelation will be able to detect the two GS, but the time series must be such that the autocorrelation between the two GS peaks is not completely cancelled by the lack of correlation for the same lags but in other periods of the year.

- Another issue to consider is whether null FAPAR values are reasonable outside the GS. As noted above, such constant, repetitive values strongly affect the results because the contributions to the autocorrelation at those small lags amounts to a 'perfect' correlation.

- My experience with looking at FAPAR signals for areas subject to double cropping is that the FAPAR does drop between the two GS, but not quite to the same low level as during the winter/dry season. This may be a minor point, but it may also impact the shape of the autocorrelation function.

Let me know what you think of these points. Cheers, Michel.

After the discussion with MV.

For the moment let’s consider that autocor failure is not a big problem because it is related to the peculiarity of the example. Try FFT to see what we get (use n=1024 or some however 2^n).

Season length. Considering crops in Africa and according to the experts here, a growing season length is normally >90 days. A situation with a GS of 70 days may occasionally occur and it would denote crop failure. So it is not very common but it is interesting to be detected for FOODSEC. In addition, If we consider also forage crops (for example alphalpa, “erba medica”) we can have GS as short as one month.

**Controls**

The pixel is not considered if:

* more than 40% of the values in the input record are invalid
* the oscillation is low ([95th - 5th] percentile < faprangeminthresh)
* unrecognized periodicity (autocorrelation could not find maxima)
* GSPY<1 or GSPY>2

**Summary of modification**

The objective is to find out and isolate the breakpoints of the elementary time windows suitable for the optimization of the mathematical model. These windows are hereafter referred to as optimization window (OW).

The “solar year” is here defined as the time window of 36 decades that can include one or two growing seasons. This time window is repeated all over the time series and it represents the first level of the time series segmentation. If there is only one GS, the solar year (opportunely adjusted locally) is the OW. If more than one GS is present in the solar year (max 2 seasons for the moment), the solar year is subset in two OWs (opportunely adjusted locally)

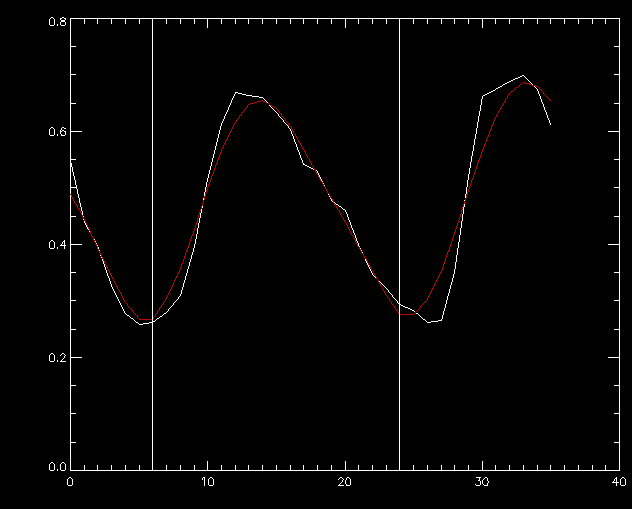
**1. Detection of the number of seasons during the solar year**

Based on the number of relative maxima in the autocorrelation

**2. Detection of: start and stop of the solar year, start and stop of the OW(s)**

It’s based on the analysis of the median year because we consider that annual cycle cannot extend beyond 36 decades (this might not be the case for irrigated fields within the tropics but it is not a concern for FOODSEC or drought, focusing on rain fed).

The median year is built without any concern about temporal location of the actual solar year, i.e. the first decade of the available time series is 1.



Example form VGs\_R282C238\_c1\_20002010\_f10\_d20110304.xdat showing 2 GS. Median year (white curve) and smoothed median year (red curve). White vertical lines refer to smoothed curve minima.

In order to set the breakpoints (vertical white lines in the figure above) we use a moving average (to avoid excessive bias I used a Savitzky-Golay smoothing kernel) with a window of 18 and 12 decades in the case of 1 or 2 GS, respectively. Afterward we detect the dates of the minima. The number of minima is equal to the number of GS detected by the autocorrelation.

The solar year is defined to start at the decade of the first minimum (occurring at XOFFSET) regardless any other consideration (Michel, for the time being I could not find any good reason to prefer one GS to the other).

At this point we know where to place the breakpoints defining the solar year (i\*36+xoffset, where i is the index of the year).

If only one season is present, the first guess OW0 is equal the solar year. If two seasons are present OW1=[start of the solar year : occurrence of the second minimum] and OW2=[occurrence of the second minimum: stop of the solar year].

**3. Optimization**

The analysis of the series than is performed year by year. In the case of 1 GS we are targeting OW0 while in the case of two seasons we are targeting first OW1 and then OW2 (same approach, just repeated twice). The OW is defined by its breakpoints (start and stop OWs, OWe).

3.1 Check that there is enough variability in OW, if not skip the OW and go to the next (recording a flag for missed GS)

3.2 Define the locally adjusted breakpoints for the OW by setting them to the occurrence of min(smooth(fAPAR)) around the OWs and OWe (search window: 6 and 4 for 1 and 2 GS, respectively)

Note: the part broadening the window if >1/4 of data are missing and concentrated in one of the two tails of the curve has been removed since not applying to this new scheme. A flag to indicate this event is maintained.

3.3 Fit the model

Range parameters (a1 and a4) are set to double value in the first guess (“The factor 2 is included because the model divides this amplitude by 2”). Indeed the (tanh()+1)/2 varies from 0 to 1, so there parameters should not be multiplied.

I considered to optimize the fitting using a constrained minimization (constrained Levenberg-Marquardt least-squares fit).

Reasons to constrain the function parameters:

* In principle, with the unconstrained optimization, parameters can assume any value regardless of their physical meaning;
* The model parameter first guess (initialization) is assigned on physical basis, it may be contradictory to accept any unphysical outcome;

Reasons for not constraing:

* growing season is deemed to start/stop at the decades where the modelled time series exceeds (drops below) a fraction of the amplitudes that are computed on the fitted values. So, if the fitting is fine, there should not be a problem;
* it si very difficult to constrain the parameter efficiently, for example a1, a4 and a0. They can all be constrained in 0-1, and we can have as result 0.6, 0.6, 0.5 (just an example). So the boundaries are ok but the result is not physical again.

So, for the time being:

* keep unconstrained optimization
* use MPFIT instead of curve fit, giving more diagnostic

NRT detection

Objective: to retrieve NRT a start of the growing season which is consistent with the one that would be estimated by the full algorithm optimized under full data availability (all solar year available).

Why?

1. For NRT phenology detection which is meaningful by itself
2. In order to compute relevant statistics involving summation from start of GS (e.g. ) of the current growing season and to compare them with previous years

How?

By learning from the past (statistics from previous years).

The rule for determining SOS is now: on the fitted data, SOS is the decade where fit\_fAPAR > threshold. Threshold = 0.05 [max(fit\_fAPAR) – fit\_fAPAR(0)].

\*\*\* obs: maybe 0.05 is a little bit too low

For every available year in the dataset I compute the threshold. Such threshold will be small for seasons presenting low amplitude (bad start) and big for high amplitude (good start).

By choosing the overall minimum as a NRT threshold I may:

* Anticipate SOS of a current good season;
* Delay SOS of a current bad season.

Both scenarios are conservatives with respect to FOODSEC monitoring, and therefore acceptable.

NRT Method

A mowing backward window (length winlen = 10/(number of GS per year), centred on it last element), is successively displaced – one decade at a time – from the last EOS forward, until the current observation.

A three parameters exponential function is optimized over this window. The SOS is detected when the fitted fAPAR rises of more than threshold. ~~The exact decade when the fitted fAPAR rises of more than threshold form the min(fit) is taken as str.~~

Michel suggest to use the same model (double hyper tangent) leaving free only the parameters of the first part

This procedure is very sensitive to:

* Winlen
* Fraction of rise/drop to detect str stop in the pdhtf (now declared in phenot, fract\_thresh), now is set to 0.10 in phenot.pro
* Threshold, computed now as the mean

**NRT detection using the same model used for monitoring**

We have n full solar years + the current one. The idea is that at the beginning of the current year we have of course no information but the expectation that it will follow the “average year”. So we have an expectation that sets the prior knowledge of model parameters (set by taking the average of the model parameters over n years so that I get also a measure of variances and covariances).

Once observations are coming we optimize the pdhtf model (M) over the available observations and taking into account our prior knowledge with cost function described in Pinty et al., 2007.

The first term of the equation will increase its weight as we get more data for the current year. At the beginning of the year the problem is ill posed and it is solved thanks to the prior knowledge.

Cost = Observation component + A priori component

Michel suggests to look at the fraction of the two contributions to the toatal cost function to understand what is driving.