APPENDIX. Seasonal-trend decomposition of a univariate time series using Local Linear Regression (LLR)

All the environmental and phytoplankton abundance time series analyzed in the present study have been decomposed in the additive form $y(t) = T(t) + S(t) + R(t)$, where $T(t)$, $S(t)$ and $R(t)$ denote the long-term trend, the seasonal and the remainder components, respectively. As pointed out in Section 2.3, nonparametric approximations for each of these components were obtained by using local linear regression –LLR– techniques (Fan and Gijbels, 1996). In order to control different aspects of the estimation process, such as examining the impact of a range of specific bandwidths, evaluating whether a constant seasonal component is present or providing suitable plots in an automatic way, the LLR approximations were self-programmed in the R language (R Core Team, 2014). This Appendix is devoted to describe the functions REG and SEASONAL.TS.12, which provide LLR approximations to the long-term trend and the seasonal components, respectively, and whose source code is available in the supplementary file STdecompositionLLR.txt. Specifically, the different steps of the estimation procedure are detailed and graphically illustrated by applying both functions to one of the phytoplankton abundance time series.

A.1. The functions.

The function REG provides local linear fits of a time series using different bandwidths selected by the user, but it also allows to estimate directly this parameter by an automatic plug-in procedure. This procedure, proposed by Ruppert et al. (1995), is relatively simple, straightforward to implement and presents good theoretical properties and an excellent performance in simulations. This automatic bandwidth selector is available in R throughout the function dpill in the R-package KernSmooth (Wand
From a unidimensional ts object series.in, which can have missing values, the function REG is invoked in the form:

```
REG(series.in, bands=c(2.5, 5, 10, 15), imputation=0)
```

Argument bands provides the sequence of smoothing parameters considered to compute the LLR estimates, while imputation is a single indicator index moving on \{0,1,...,length(bands)\} and specifying the bandwidth used to fill the gaps. If imputation=0, the LLR estimate based on the bandwidth generated from dpill is used. If imputation=k, the LLR estimate to impute the missing data is calculated using the k-th element of bands. The output of REG includes: (i) a multiple ts object (series.out) with the original series, the imputed series and the smoothed series for each of the bandwidths in bands, (ii) the plug-in selector computed by dpill (b.optimal), (iii) the bandwidth used to impute the series (b.imputation), and (iv) the explained and residual sums of squares obtained for each estimator. A graphical output showing the original and the estimated series is also supplied.

The function SEASONAL.TS.12 provides the estimated seasonal component for a time series by using LLR smoothing. This function always considers a frequency equal to 12, i.e. it is assumed that the series have been monthly sampled, from January to December. From a unidimensional ts object series.in, which can have missing values, the function SEASONAL.TS.12 is invoked in the form:

```
SEASONAL.TS.12(series.in, est.trends, band)
```

Argument est.trends contains the estimated long-term trend for series.in (obtained for example from the REG function) and band determines the bandwidth considered, namely the plug-in selector obtained with dpill, if band=0, or the value directly indicated by band, otherwise. SEASONAL.TS.12 produces a data matrix (dat12) where each row contains the centered time series for each year, the estimated seasonal component (season) and the bandwidth used (b).

An illustrative example using the abundances series Chaetoceros spp. is given below.

**A.2. Decomposition of the abundances series Chaetoceros spp. (CHASPP).**

Before performing the seasonal-trend decomposition, the series is log-transformed to reduce effects of skewness and extreme observations, and then a ts object is created from the transformed data.

```
R> ldata.CHASPP <- log(data.CHASPP + 1)
R> ts.ldata.CHASPP <- ts(ldata.CHASPP, frequency=12, start=c(1989,5))
```
First, local linear smoothers of the log-transformed time series `ts.ldata.CHASPP` are computed for a range of smoothing levels ($b_N = 2.5, 5, 10$ and $15$) in order to shed light on the underlying structure. This is carried out using the function `REG` as follows, which produces the plot depicted in Figure A.1.

```r
R> Smooth.CHASPP <- REG(ts.ldata.CHASPP, bands=c(2.5, 5, 10, 15), imputation=0)
```

![Original, imputed and estimated series](image)

Figure A.1. Original, imputed and smoothed series with bandwidths $b_N = 20$ (plug-in selector generated using the function `dpill`), $2.5$, $5$, $10$ and $15$.

The list `Smooth.CHASPP` stores the output of the function `REG`. In particular, `Smooth.CHASPP$sc` provides the explained and residual squared sums for each estimator, such as shown below.

```r
R> Smooth.CHASPP$sc
          Total_ss Residual_ss Explained_ss     % Exp.
  b.opt=13  476.4608     416.3498     60.11101 0.1261615
```

The list `Smooth.CHASPP` stores the output of the function `REG`. In particular, `Smooth.CHASPP$sc` provides the explained and residual squared sums for each estimator, such as shown below.
A bandwidth sufficiently large canceling short cycle effects is necessary to obtain a proper estimate of the long-term trend. Figure A.1 suggests that $b_N = 10$ months, i.e. taking averages on two years approximately, produces a reasonable fit. Note that larger bandwidths do not lead to a substantial increase of the explained variability. The smoother with $b_N = 10$ is stored in Smooth.CHASPP$series[,6].$

Next step consists on estimating the seasonal component using SEASONAL.TS.12, which consists of the steps outlined below. The series is detrended by doing ts.ldata.CHASPP - Smooth.CHASPP$series[,6], and the resulting series is split into subseries of length 12 covering annual periods on the observation interval. Based on all the annual curves, the series of monthly averages is obtained and then smoothed using a LLR estimate with the plug-in selector introduced by Ruppert et al. (1995). All of these steps are automatically performed by running SEASONAL.TS.12.

R> sc.CHASPP <- SEASONAL.TS.12(series.in = Smooth.CHASPP$series[,2], est.trends = Smooth.CHASPP$series[,6], band=0)

The list sc.CHASPP is formed by three components: sc.CHASPP$dat12, matrix storing the subseries of length 12 for the years in study, sc.CHASPP$season, vector storing the estimated seasonal component, and sc.CHASPP$b, the dpill bandwidth. The plot of the estimated seasonal component shown in Figure A.2 has been obtained by executing the following code line.

R> plot(1:12, sc.CHASPP$season, type="l", ylab="Seasonal Component", xlab="Month", ylim=range(sc.CHASPP$season), col=2, main = " Chaetoceros spp. (CHASPP)", font.main=3)
Figure A.2. Estimated seasonal component.

Supplementary references:

