Spatio-Temporal Change Detection from Multidimensional Arrays: detecting deforestation from MODIS time series

Meng Lu[a,*], Edzer Pebesma[a], Alber Sanchez[a,b], Jan Verbesselt[c]

[a]*Institute for Geoinformatics, Westfälische Wilhelms-Universitt Münster (WWU), Heisenbergstraße 2, 48149 Münster, Germany*
[b]*National Institute for Space Research (INPE), Av. dos Astronautas, 1758, 12227-007, São José dos Campos, Brazil*
[c]*Laboratory of Geo-Information Science and Remote Sensing, Wageningen University, Droevendaalsesteeg 3, Wageningen 6708 PB, The Netherlands*

Abstract

Growing availability of long-term satellite imagery enables change modeling with advanced spatio-temporal statistical methods. Multidimensional arrays naturally match the structure of spatio-temporal satellite data and can provide a clean modeling process for complex spatio-temporal analysis over large datasets. Our study case illustrates the detection of breakpoints in MODIS imagery time series for land cover change in the Brazilian Amazon using the BFAST (Breaks For Additive Season and Trend) change detection framework. BFAST includes an Empirical Fluctuation Process (EFP) to alarm the change and a change point time locating process. We extend the EFP to account for the spatial autocorrelation between spatial neighbors and assess the effects of spatial correlation when applying BFAST on satellite image time series. In addition, we evaluate how sensitive EFP is to the assumption that its time series residuals are temporally uncorrelated, by modelling it as an autoregressive process. We use arrays as a unified data structure for the modeling process, R to execute the analysis, and an array database management system to scale computation. Our results point to BFAST as a robust approach against mild temporal and spatial correlation, to the use of arrays to ease the modeling process of spatio-temporal change, and towards communicable and scalable analysis.

Keywords: BFAST, time series analysis, spatial correlation, temporal correlation, array database, spatio-temporal change modeling

1. Introduction

Advanced earth observation satellite sensors provide remote sensing products that are rich in spatial, temporal, and spectral information. Open access policies of space agencies

[*] corresponding author
Email address: meng.lu@uni-muenster.de (Meng Lu)

Preprint submitted to ISPRS Journal of Photogrammetry and Remote Sensing *March 23, 2016*

and the progress of remote sensing technologies make these products more accessible, which enables a wide range of novel applications, such as near real-time global change monitoring. This, however, calls for efficient handling and scalable processing of the massive amounts of available data. Major challenges include big data management, multidimensional data information extraction, and complex large-scale spatio-temporal change modeling procedures implementation and result visualization. These challenges call for novel data management and analytics tools and advanced spatio-temporal statistical algorithms.

Typical remote sensing satellite images are regularly discretised in space and time, and can naturally be represented as multidimensional arrays. The array data structure facilitates change modeling in many ways. Firstly, the array data structure allows a clean data processing procedure which simplifies data preparation, and avoid data structure conversions during the analysis. Wickham (2014) calls the unified data preparing process to "tidy data", and suggests restructuring all datasets into single, long tables. Since most earth observation data (i.e. earth information collected by remote sensing technologies) come as time series of multispectral images, and structuring such datasets into arrays is the more natural approach for data storage, analysis and visualisation. In addition, the array data structure allows flexible application of spatio-temporal statistical algorithms (Zscheischler et al., 2013) and other information extraction methodologies (Mello et al., 2013), which was already exploited in the on-line analytical processing (OLAP) approach to analyse business data (Chaudhuri and Dayal, 1997; Viswanathan and Schneider, 2011). Finally, the array data structure facilitates parallelizing of the modeling process (Stonebraker et al., 2013). Array Data Management and Analytics Software (DMAS), which stores and operates on data as multidimensional arrays, can thus be used to scale the process and resolve the difficulties of large memory consumption and computational bottlenecks usually found in non-parallelized systems. Examples of array DMAS include SciDB (Cudre-Mauroux et al., 2009) and rasdaman (Baumann, 1994).

Remotely sensed image time series analysis (Verbesselt et al., 2010; Broich et al., 2011) has been drawing more attention in pixel-based change detection in recent years (Jianya et al., 2008; Banskota et al., 2014) due to the increased availability of long-term satellite image time series and improved computational power. Statistically, these methods can be classified as detecting change in mean (Kuan and Hornik, 1995), (e.g. by tests based on OLS (Ordinary Least Squares) residuals such as CUSUM (Cumulative Sum) test (Brown et al., 1975)), or change in regression parameters, (e.g. by tests that assess all regression coefficients such as supLM (supremum Lagrange Multiplier) test (Andrews, 1993; Zeileis and Hothorn, 2013)). Change detection with time series imagery solves many problems that are infeasible with bi-temporal analysis (Coppin et al., 2004; Jianya et al., 2008). There are several examples: 1) image time series analysis enables detection of unknown historical changes retrospectively, and monitoring of changes in near real-time (Verbesselt et al., 2012); 2) image time series analysis is able to classify land cover types that are of subtle differences in reflection. For example, one difficulty in analysing tropical forest conservation from remotely sensed imagery *pairs* is to discriminate plantations from secondary forests (Lucas et al., 1993); 3) the regression model is flexible, and can integrate variables that will affect the process. For example, it is hard to distinguish between climate-induced forest drought and anthropogenic deforestation. Integration of climate variables, such as precipitation and temperature, can assist differentiating between these changes (Dutrieux et al., 2015); 4) in terms of reliability, satellite image time series analysis has the advantage of being more resistant to noise (Coppin et al., 2004).

One popular time series change detection tool that raised attention in image time series analysis is BFAST (Breaks For Additive Season and Trend) (Verbesselt et al., 2010, 2012). BFAST constitutes a change detection procedure on top of a comprehensive set of serial structural change detection tools. BFAST has been applied in various cases, such as detection of shifts in vegetation trends (Jong et al., 2012; Forkel et al., 2013). BFAST detects the structural change in trend and seasonality of a time series, which has many applications. For example, the seasonality between agriculture products (e.g. soybean) and rainforest are different, which enables the discrimination of different kinds of forest disturbances (e.g. changes from forest to agriculture vs. forest fire). BFAST treats observations as serially uncorrelated. Since it models pixel time series independently, possible spatial correlation around the area is ignored. Simple extensions to BFAST could model the residuals as an autoregressive (AR) process, and/or adopt a simultaneous autoregressive (SAR) model for the spatial residual process.

In this paper, we apply BFAST to our study region, and evaluate the effect of extending BFAST with temporal and spatial correlations. We want to do this in such a way that 1) it can be extended to global-scale data and 2) it is reproducible by other scientists within a reasonable effort. This means that we need to use a high-level data analysis language, such as R; that we need to use an open source Array Data Management and Analytics Software (DMAS) that allows parallel execution of the R code; and finally that we publish all the scripts to recreate the database and carry out the computational experiments on the data.

The study case concerns historical forest cover change detection with long-term MODIS image time series. We show how pixel-based time series analysis are extended to region-based joint spatio-temporal analysis, how the whole change modeling process and spatio-temporal information exploitation are simplified by multidimensional arrays, and how Array DMAS implement and scale the process. The study case is extensible and the methodologies are generic and can form the basis for further remote sensing data experiments.

The paper is organized as follows. Section 2 introduces and discusses multidimensional arrays. Section 3 describes how we model spatio-temporal change. Section 4 introduces the study case. Section 5 presents results, and sections 6 and 7 finish with discussion and conclusions, respectively.

2. Multidimensional arrays

Most natural phenomena can be represented in multidimensional arrays once they are sampled and quantized in a computer system. The dimensionality of an array can be flexibly set for efficient information extraction and modeling. Examples of practical array abstraction include: 1-D ordered tables or time series (t); 2-D satellite images (x/y); 3-D satellite image time series (x/y/t); 4-D multi-spectral spatio-temporal data (band/x/y/t); subsurface hydrological data (x/y/z/t); and 5-D multi-sensor, multi-spectral spatio-temporal data (sensor/band/x/y/t).

2.1. Potential application of multidimensional arrays in remote sensing

As a multidimensional data structure, arrays have the potential to bring many advanced information extraction into practical use. For example, instead of using a single spectral layer (e.g. vegetation index), multi-spectral multi-temporal approaches (e.g. spectral-temporal surface (Mello et al., 2013)) use more information and thus are able to better represent the earth surface (Mello et al., 2013). This multi-spectral multi-temporal approach can be integrated with spatial information. Data can be organized as 4-D arrays with space, time and bands as four dimensions, and algorithms can be applied to them. Similar examples can be found in data fusion (Castanedo, 2013), where data from different sensors can be organized on two dimensions, and in spatio-temporal statistical modeling. In addition, the developed spatio-temporal statistical algorithms can be flexibly applied within array partitions that span the relevant array dimensions. This study especially demonstrates how array data can be used in spatio-temporal change modeling, and how an Array Data Management and Analytics Software System (DMAS) can be used for parallelization and scaling.

2.2. Tidy data with Array data structure

The open source data analysis programming language R (R Core Team, 2015) provides rich data analysis tools. All entities R works on are objects. A special type of object is the array. For instance, the following code segment creates a 100 × 100 × 10 × 5 array, requests its dimensions, prints the length of the data vector (the product of the dimensions), and shows the length of a one-dimensional sub-array (vector) in the third dimension:

```
> a = array(NA, c(100,100,10,5))
> dim(a)
[1] 100 100  10   5
> length(a)
[1] 500000
> length(a[10,10,,1])
[1] 10
```

Such arrays are held in main memory, are dense, and hence do not scale to massive data or for sparse arrays. They allow to efficiently carry out functions over single dimensions (or sets of dimensions), such as is done in remote sensing time series analysis. Also, arrays keep no information on how dimensions or indexes relate to time, space, or other data properties, so they require additional book-keeping.

Certain types of object are developed to facilitate spatial, temporal or spatio-temporal analysis. For example, time series objects are used for time series analysis; the space-time objects in R (Pebesma, 2012) are developed for spatio-temporal data processing, analysis, and visualisation; the R packages raster (Hijmans, 2015) and spatial.tools (Greenberg, 2014) are used to process and analyze large-scale raster data. Functions are then developed for specific types of objects. Spatio-temporal Kriging functions work with space-time objects, and map algebra operations (Tomlin, 1990) can be applied to a raster. As an example, a space-time array with air quality data aq can be subsetted for a particular state, time period, and air quality parameter by

```
aq[California, "2008-03::2014-09", "BlackSmoke"]
```

which indicates that reference to spatial, temporal, and measured quality are directly expressed in the syntax, and do not require the use of integer indexes.

We use arrays as a flexible data structure for change modeling with multiple dimensionality. Those objects (space-time, raster, etc.) that can be viewed as simple arrays with attributes, can conveniently be converted to multidimensional arrays and between each other. In addition, array as an object can be used for the analysis of spatially and temporally regularly sampled spatio-temporal data. Thus we "tidy data" (Wickham, 2014) by organizing data as multidimensional arrays, and use array data to provide a clean, communicable, and scalable way for spatio-temporal data analysis.

2.3. Array Data Management and Analytics Software (DMAS)

R and MATLAB (MATLAB, 2015) are powerful data analysis tools that support array data. However, for these data analytics systems storage and computation are relatively difficult to scale when dealing with massive data. In this study we use SciDB (Cudre-Mauroux et al., 2009) array DMAS, which scales by design. SciDB splits large arrays into equally sized and potentially overlapping chunks of data, which are assigned to different worker nodes. These worker nodes may be processed on the same machine, or distributed over a cluster composed of independent servers (Figure 1). Each node controls its own storage and memory. One of the nodes is the coordinator, which is responsible for client-server communications and for query execution coordination. The remaining worker nodes participate in the distributed query processing. As a result, SciDB automatically considers both parallelization and out-of-core processing. The earth observation community is exploring the spatial analysis applications of array databases. Planthaber et al. (2012) use SciDB as an analysis platform for processing low level MODIS products. As SciDB does not provide specific spatio-temporal interfaces, Câmara et al. (2014) propose to extend SciDB with a spatio-temporal query interpreter, and create a platform to analyze the spatio-temporal field data type.

2.4. Scaling the process with SciDB and R

Ongoing research attempts to integrate SciDB and R, to access both the scalability and the data management capabilities of SciDB, as well as the data analysis power of R (Leyshock et al., 2013). SciDB provides two ways to interact with R: by SCIDBR and by r_exec. The R package SCIDBR (Lewis, 2015a) acts as a SciDB client. In this way SciDB operations are wrapped into R syntax, and are performed on SciDB arrays through the R interface. It uses a reference in R to point to SciDB array, and data can be materialized in R through indexing. For example, we can extract data from the array that we stored in SciDB called Juara_array, which is a 849 × 945 × 636 array with "evi2" as one of the attributes. We extract the evi2 value of pixels at location (1,1) from time 1 to time 4:

```
> library("SCIDBR")
> scidbconnect("http://the.server.org/", port = 49971)
> Juara<-scidb("Juara_array")
> Juara
A reference to a  849x945x636 SciDB array
> Juara[1,1,1:4][,"evi2"][]
[1] 0.5837997 0.4504056 0.1979948 0.3451148
```

While this way greatly facilitates data exchange between SciDB and R, it restricts applications to the use of built-in SciDB operations only, or requires moving all data in and out of R.

The alternative way of integration is to invoke R scripts inside database queries through a SciDB plugin `r_exec` (Lewis, 2015b): The `r_exec` function (Lewis, 2015b) works in R as:

```
load_library("r_exec")
r_exec(Input_Array, "expr= R_expression")
```

The R code is written after `expr=`. The results are returned as a list and can be restored into SciDB arrays. This allows for including functionality of R and its extension packages. Each SciDB instance runs its own R processes (Figure 1) and scripts are called independently over all chunks.

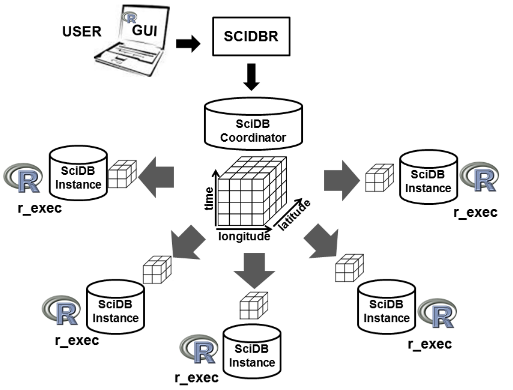


Figure 1: Distribution of the computation to multiple SciDB nodes

3. Spatio-temporal change modeling

Most of spatio-temporal analysis approaches are developed in two separate stages – spatial analysis after time series analysis, or time series analysis after spatial analysis (Schabenberger and Gotway, 2004). Two-stage approaches miss information spanning both space and time Schabenberger and Gotway (2004). Joint spatio-temporal analysis, which accounts for the spatio-temporal dependency and jointly model the spatio-temporal process, is preferable over two-stage methods. Some modern statistics develop joint spatio-temporal change detection methods by using random fields (Bolin et al., 2009), or hierarchical modeling (Cressie and Wikle, 2011), which could integrate spatial information in time series analysis. However, most methods only detect gradual change (i.e. if the trend of the time series regression is significant). There are few works that attempt joint spatio-temporal change detection for abrupt changes (Zscheischler et al., 2013). In addition, the added complexity of spatio-temporal analysis may make the process impractical. Indeed, even pure time series analysis algorithms are difficult to apply on a large scale (Ban et al., 2015).

3.1. BFAST

BFAST decomposes the time series into seasonal, trend, and remainder components with STL (Seasonal-Trend decomposition procedure based on Loess) (Cleveland et al., 1990). Then BFAST sequentially detects breakpoints in trend and seasonal components in an iterative way. This breakpoints detection process consists of two steps: notifying the structural change of the time series, and estimating breakpoints. In the first step, a fluctuation test on regression residuals can be chosen from a group of Empirical Fluctuation Process (EFP) methods implemented by Zeileis et al. (2003). If the EFP test suggests the structural change in time series, the number and location of the breakpoints will be estimated using the method of Bai and Perron (1998, 2003) (referred to as BP method). This method dynamically locates the optimal position of the breakpoints by minimizing the SSR (Sum of Squared Residual), and determines the optimal number of the breakpoints by minimizing the BIC (Bayesian Information Criteria). An example of a graphical output of BFAST is shown in Figure 2.

3.2. Extended BFAST

In an extended version of BFAST, we mainly improve on the EFP by 1) Fitting a trend-seasonal model instead of decomposing time series into trend and seasonal components, 2) integrating AR(1) to account for the serial autocorrelation, and 3) integrating SAR model to account for the spatial autocorrelation. From the EFP we can get the p-value for the time series of each pixel. If the p-value is less than the significance level (e.g. 0.05), the BP method (Bai and Perron, 1998, 2003) in BFAST is applied to identify the number and time of the breakpoints. Different significance levels were experimented with to observe if the results were improved.

3.2.1. Empirical fluctuation process

Given the linear regression model:

$$y_i = x_i^{\intercal} \beta_i + u_i, \qquad (i = 1, \dots, t) \tag{1}$$

where x_i contains m independent variables, and β_i is an $m \times 1$ coefficient vector. The test is concerned with testing the constancy of the regression coefficients β_i (Kuan and Hornik, 1995). The null hypothesis (H_0) of the fluctuation test is that the $\beta_i = \beta_0$ for all i. Under H_0, the fluctuation of u_i is characterized by the Functional Central Limit

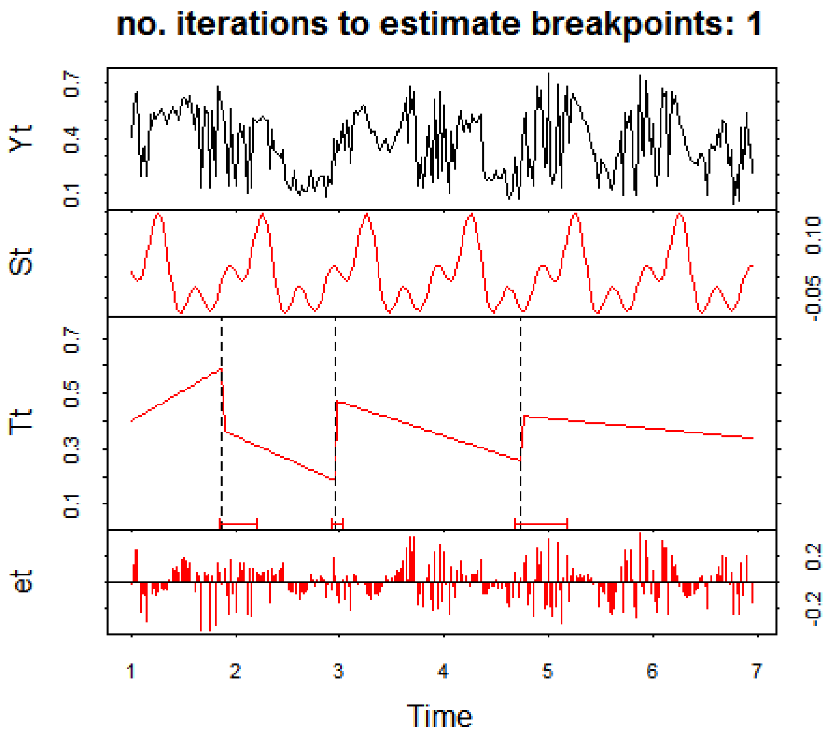


Figure 2: An example of graphical output of BFAST. Y_t is the graph of raw time series of the EVI2 (Jiang et al., 2008) vegetation index, S_t is the graph of the seasonal component, T_t is the graph of the trend component, and e_t is the graph of the residuals. In this example, three abrupt changes are detected in the trend component and the corresponding confidence interval of these breakpoints are indicated.

Theorem (FCLT) (Basseville and Nikiforov, 1993), of which the CUSUM (Cumulative Sum) of regression residuals converge to a Brownian Motion. Thus, a change in β_i can be detected if the CUSUM of regression residuals do not satisfy the FCLT (Basseville and Nikiforov, 1993). In this study we specifically look at OLS-CUSUM (Ploberger and Krämer, 1992) and OLS-MOSUM (Moving Sum) tests (Chu et al., 1995). Compared to the OLS-CUSUM test, which uses all the residuals, the OLS-MOSUM test applies a predefined moving window that accumulates a fixed number of residuals within the window. The limiting process of OLS-CUSUM is a standard Brownian bridge (Ploberger and Krämer, 1992). The limiting process of OLS-MOSUM is increments of a Brownian bridge (Chu et al., 1995).

We assume the model has a linear trend and a harmonic season, and adopt the model described in Verbesselt et al. (2012). The x and β in this model are:

$$x = (1, t, \sin(2\pi 1t/f), \cos(2\pi 1t/f), \ldots, \sin(2\pi kt/f)), \cos(2\pi kt/f)))^\intercal$$

$$\beta = (\alpha_1, \alpha_2, \lambda_1 cos(\delta_1), \lambda_1 sin(\delta_1), \ldots, \lambda_k cos(\delta_k), \lambda_k sin(\delta_k))^\intercal,$$

where α_1 is the intercept, α_2 is the slope of the trend model, λ is the amplitude of the seasonal model, δ is the phase of the seasonal model, f is the frequency and k is the number of harmonic terms. In this study we use third order harmonics ($k = 3$).

3.2.2. Serial autocorrelation correction with AR(1) model

We correct the autocorrelated error of each time series with the first order AR (AR(1)) model:

$$y_i = x_i^{\intercal}\beta_i + y_{i-1}\phi + w, \qquad (i = 2, \dots, t) \tag{2}$$

where the temporal dependence is described by ϕ, and w is white noise. To examine structural change, the OLS-CUSUM and OLS-MOSUM are applied to w. We will call these methods AR(1) OLS-CUSUM and AR(1) OLS-MOSUM, respectively.

3.2.3. Spatio-temporal statistical model

When applying the linear regression model on each pixel time series (t time steps) of a spatial neighborhood (with n pixels) of a pixel, we have:

$$\begin{aligned}\mathbf{Y} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{u} \\ \mathbf{Y} &= (\vec{y_1}, \dots, \vec{y_t})^{\intercal} \\ \mathbf{u} &= (\vec{u_1}, \dots, \vec{u_t})^{\intercal},\end{aligned}$$

where $\vec{y_i}$ contains the observations of n neighboring pixels. $\mathbf{Y}$ is the matrix of the observations in space and time. $\vec{u_i}$ contains the corresponding errors of the neighborhood.

If the errors in $\mathbf{u}$ are spatially correlated, the spatial dependence can be used to improve the time series analysis. We use a SAR model to characterize the spatial correlations between the neighboring pixels, as in

$$\mathbf{u} = \mathbf{B}\mathbf{u} + \mathbf{v} \qquad \mathbf{B} = \rho\mathbf{W}, \tag{3}$$

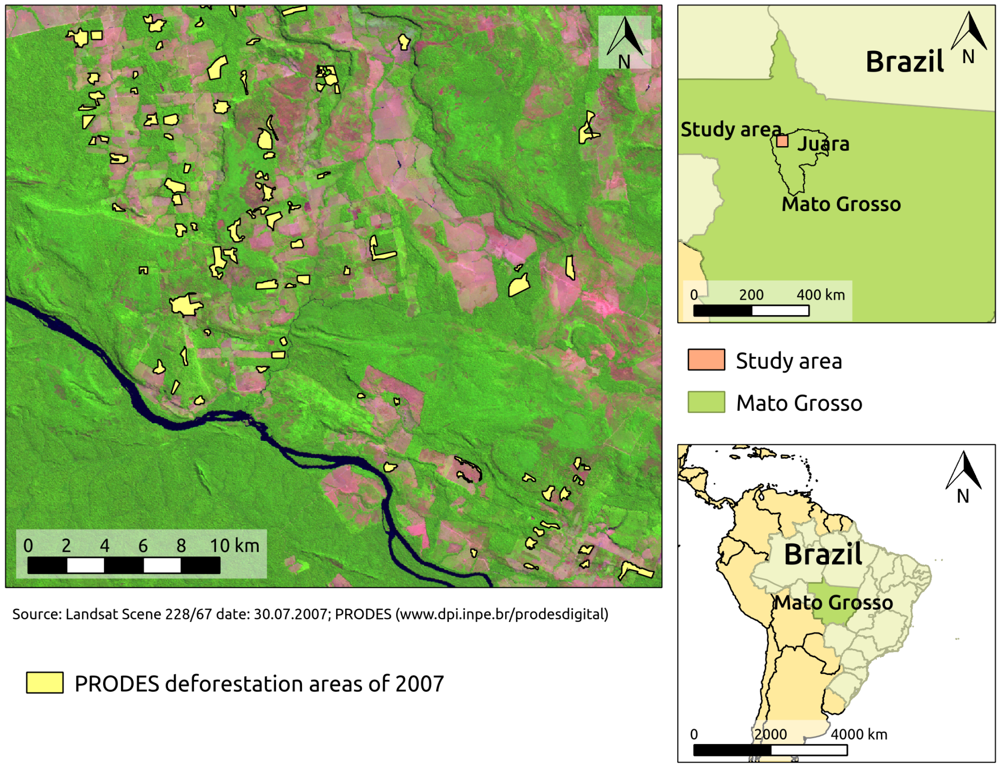
where $\mathbf{B}$ is the matrix of parameters describing spatial correlations between residuals. It is the spatial dependence parameter ρ multiplied by a weight matrix $\mathbf{W}$. The spatial dependence parameter ρ can be estimated with maximum likelihood. We used a 3×3 window. The $\mathbf{W}$ is filled with 1 for neighboring pixels and identical time, and 0 otherwise. $\mathbf{v}$ is assumed to contain independent residuals from the auto-regression. The model can be expressed as:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + (\mathbf{I} - \rho\mathbf{W})^{-1}\mathbf{v}, \tag{4}$$

of which the regression residuals $\mathbf{v}$ between time series are spatially uncorrelated. To examine structural change, the OLS-CUSUM and OLS-MOSUM are applied to $\mathbf{v}$. We will call these methods SAR OLS-CUSUM, and SAR OLS-MOSUM methods, respectively. SAR and AR models were implemented using the R package spdep (Bivand and Piras, 2015).

4. Study case

A 3-D spatio-temporal modeling case is developed in land cover change with MODIS image time series. The study site is in Juara (21,387 km^2; Figure 3, right), Mato Grosso, Brazil, where land cover changes have been observed during the past decades. In order to compare the speed between paralleled and unparalleled computation, a smaller subset of the Juara site (Figure 3, left) was sampled for the experiments, which consists of 150 (longitude)× 150 (latitude)× 636 (time steps) pixels. From the false color Landsat image, the urban area (brown), vegetation area (green), and a river (dark blue) can be observed. The study uses 12 years of the first two bands of MODIS 09Q1 product, which are of 8-day temporal resolution, 250m spatial resolution, and are atmospherically corrected (Vermote et al., 2002). The study period is from the year 2000 to the year 2012. We use vegetation index, EVI2 (Jiang et al., 2008) to represent the land cover. The study only detects changes in the forest to guarantee that result can be validated with products from Amazonian forest monitoring system from Brazils National Institute for Space Research (INPE). The Amazonian forest monitoring systems are developed by visually interpreting the satellite images at different times. We compare the detected change with PRODES (INPE, 2015c; Shimabukuro et al., 2012; Hansen et al., 2008) – a yearly cumulative clear-cut deforestation inventory with 30m Landsat data, DETER (INPE, 2015b; Shimabukuro et al., 2012; Hansen et al., 2008) – a half-monthly near real-time forest degradation and deforestation monitoring system with 250m MODIS data, and DEGRAD (INPE, 2015a) – a yearly cumulative forest degradation and deforestation monitoring system with 30m Landsat. In addition, Landsat5 imagery are visually interpreted to assist the validation process.



Source: Landsat Scene 228/67 date: 30.07.2007; PRODES (www.dpi.inpe.br/prodesdigital)

Figure 3: Study area and example of validation data. Right: location of the Juara study area in Mato Grosso. Left: PRODES reported deforestation area plotted on Landsat5 image in 2007 of the sample area.

4.1. Change modeling process

4.1.1. Data storage and pre-processing

The data were stored as a 3 dimensional array in SciDB, with longitude, latitude, and time as dimensions. Digital numbers of Band1 (red band), Band2 (near infrared band) and pixel QA (Quality Assurance) (USGS, 2014) are the attributes of the array. Then we selected the study area and calculated the EVI2 (Jiang et al., 2008) using the red and near infrared bands. At last, a basic filtering function is applied to remove the low quality pixels (i.e. one or more bands that are of faulty data) based on the QA. The removed pixels are interpolated with the median EVI2 of a $3 \times 3 \times 3$ spatio-temporal neighborhood. This simple interpolation method was chosen because 1) more than 80% (3704 out of 22500 pixels) of data are of good quality, 2) these low quality pixels are regularly spread over space and time, 3) there are strong spatio-temporal correlations within $3 \times 3 \times 3$ neighborhood, and 4) the changes remain over a period much longer than the temporal resolution. More sophisticated interpolation methods (e.g. Kriging) can be used, but are beyond the scope of this study.

4.1.2. Change detection

The next step is to apply the original and extended BFAST model to the spatio-temporal region to detect change. We set the moving window size (h) of the MOSUM process to 0.15. The analysis was carried out in R and parallelised with SciDB using `r_exec`.

4.1.3. Post-processing and result analysis

The results are returned as a list and is reconstructed to a 3-D array in SciDB, with the longitude, latitude, and time as dimensions to indicate the spatio-temporal location of breakpoints. The changes can be classified based on the magnitude of the breakpoints and the slopes of the trend before and after the breakpoints. These properties of the changes (magnitude, the classes of changes) are stored as attributes. The result array can be materialized in R for visualization and other analysis.

4.2. Results assessment

A forest mask was created from the PRODES forest-deforestation map of the year 2000. The non-forested area before the year 2000 were masked out for results validation. A validation dataset was generated from the deforestation maps from PRODES, and forest degradation maps from DEGRAD and DETER. The deforestation events that are indicated in PRODES, DETER and DEGRAD are combined to validate the detected changes by EFP of BFAST and the extended BFAST. The BFAST results are compared with the validation dataset in space, and the detected changes are classified into error matrix:

- True Positive (TP): the **changed** area that are indicated by BFAST as **changed** area indicated by the validation dataset.

- True Negative (TN): the **unchanged** area that are indicated by BFAST as **unchanged** area indicated by the validation dataset.

- False Negative (FN): the **unchanged** area that are indicated by BFAST as **changed** area indicated by the validation dataset.
- False Positive (FP): the **changed** area that are indicated by BFAST as **unchanged** area indicated by the validation dataset.

We use the producers accuracy from Pontius et al. (2008) (referred to as Pontius producer's accuracy) as a measure to assess the results. The Pontius producer's accuracy is defined as: $TP(TP + FN + FP)^{-1}$.

Pontius et al. (2008) defined this measure based on the observation that in land cover change analysis, the majority of the pixels remain unchanged, which also applies for this study case. In this case the large number of TN makes the results less assessable.This measure thus takes out the TN, and gives a higher accuracy to the statistical models. Eight models are evaluated here i.e. SAR-MOSUM, SAR-CUSUM, MOSUM, CUSUM, AR SAR-CUSUM, AR SAR-MOSUM, AR MOSUM, AR CUSUM.

4.3. Efficiency and scalability

The efficiency of using `for loop`, R array function `apply()`, and `r_exec` are assessed by comparing the computation time between three implementations:

1. implement BFAST with `for loop` in R (implementation 1)
2. apply BFAST on array with R function `apply()` in R (implementation 2)
3. scale implementation 2 with SciDB using `r_exec` (implementation 3).

5. Results

5.1. SAR integrated EFP

The highest producer's accuracy is achieved with OLS-MOSUM method and AR(1) OLS-MOSUM method (Figure 4). OLS-MOSUM has detected more TP than the other three methods. OLS-CUSUM method detected less changes, as well as the least FP. Both the SAR OLS-MOSUM and SAR OLS-CUSUM method have detected less FP and less TP compared to their original pure time series methods. The producer's accuracy are higher for the original BFAST methods, which are more sensitive to changes. The higher significance level (p-values) for declaring the structural change of a time series resulted in higher producer's accuracy. Integration of AR(1) model yields to similar results as original BFAST.

Figure (5) shows the map of agreement (TP,TN,FP and TP) detected by different BFAST methods. The yellow and blue regions (TP,FP) are the areas where BFAST methods indicated structural change of a time series. It can be observed that the changes detected by SAR integrated methods are more discrete and more spread out than the original pixel based analysis. The green regions represent the largest agreement class (TN), which are the constant forest areas from 2000 to 2012, supported both by the validation dataset as well as BFAST. The OLS-CUSUM test, SAR OLS-CUSUM test and the OLS-MOSUM test all show the lower left region as being slightly disturbed, but the SAR OLS-MOSUM test indicates many small region of change (FP). The maps of AR(1) model are not shown since the differences are not obvious.

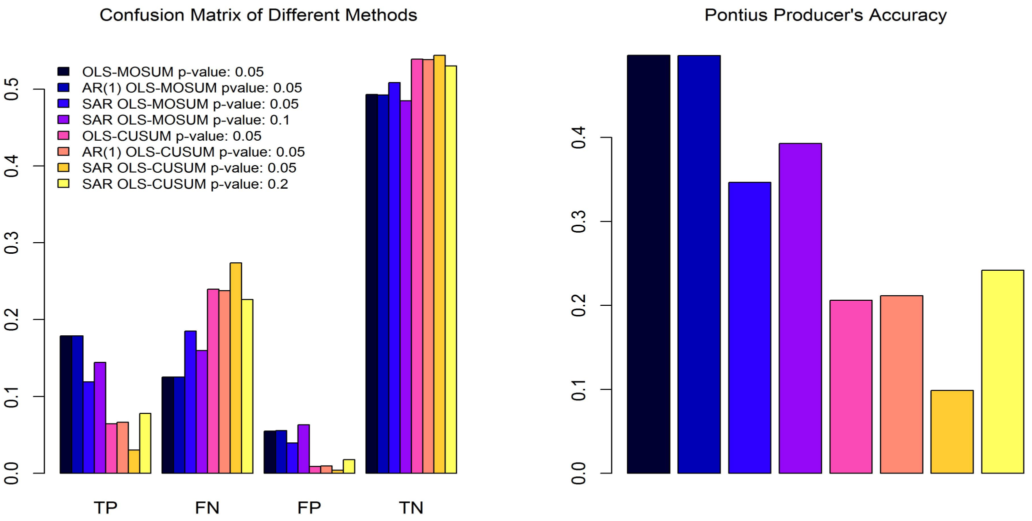


Figure 4: Comparison between different methods. Left: confusion matrix of detected changes in each validation category for each method. Right: Pontius producer's accuracy for each method

5.2. Efficiency and scalability

The result shows that the implementation 2 is much more efficient than implementation 1, especially when the time series is longer. It is about $2 \times n$ times faster than implementation 1, when the length of one time series is n times longer than the other time series. With 8 SciDB instances, implementation 3 turned out to be about 4 times faster than implementation 2.

5.3. Reproducibility

The modeling process can be reproduced by executing the scripts available at Github: scalable spatio-temporal BFAST[1]. These scripts cover the workflow from software installation to result validation. The spatio-temporal modeling approaches are written as R scripts. The scalability is achieved through the combination of SciDB and R using the `r_exec` plugin (see section 2.4. Scaling the process with SciDB and R). The main R scripts include:

- SAR integrated EFP process with R (`R_SAR_efp`),
- SAR integrated EFP process with SciDBR (`rexec_SAR_efp`),
- Validation result reproduction (`repro_SAR_efp`) ,

We separate the reproduction process with "`R_SAR_efp`" and "`rexec_SAR_efp`", respectively, for reproducing the change detection result with SciDB or only R. This separation – besides allowing to compare the scalable and non-scalable processes – enables processing small areas when R is sufficient. The scripts will reproduce the results in Figure (4) as long as the same data are used.

[1] https://github.com/ifgi/scalable-spatial-temporal-BFAST

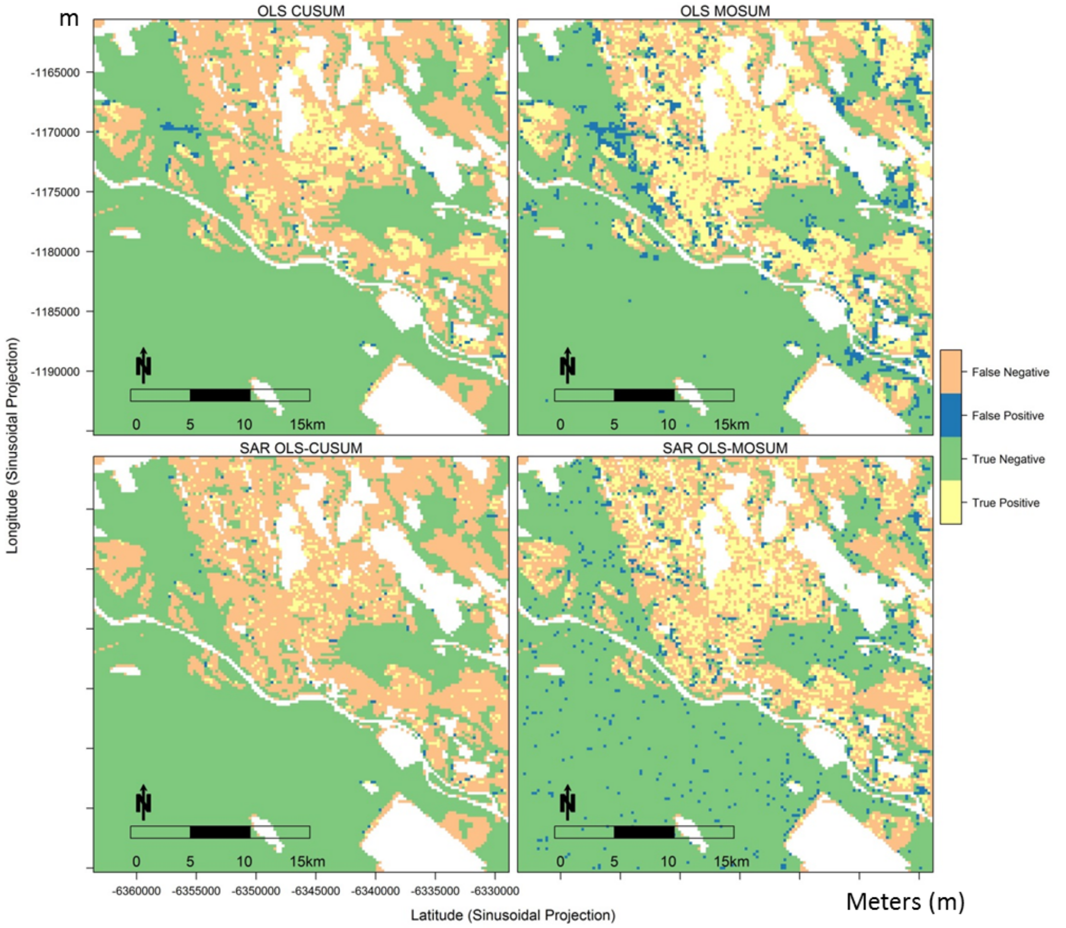


Figure 5: Spatial distribution of the different validation categories of detected breakpoints. Orange: FN, Blue: FP, Green: TN, Yellow: TP. The white space are the non-forestry area before the year 2000.

6. Discussion

6.1. Results of spatio-temporal modeling

6.1.1. Comparison between CUSUM and MOSUM

In this study, the OLS-MOSUM method outperformed the OLS-CUSUM method. One possible reason is that when time series is long, the OLS-CUSUM method neutralizes the fluctuations of residuals and becomes less sensitive to change. The OLS-MOSUM method uses only recent information and is not affected by the length of the time series. Other statistical reasons that OLS-CUSUM are not sensitive to some kinds of parameter changes can be found in Chu et al. (1995). Chu et al. (1995) indicated that while the performance of the OLS-MOSUM method is comparable with the OLS-CUSUM method when there is a single structural change, the OLS-MOSUM method performs better than the OLS-CUSUM method when the parameter first changes into a new level and then returns to the original level. This difference between the OLS-CUSUM and the OLS-MOSUM methods can be evaluated with the BP method, and may be used to distinguish between changes that are resilient (e.g. forest fire), and changes that are non-reversible (e.g. urbanization). On the other hand, with less observations, the OLS-MOSUM method is less resistant to noise. The determination of the moving window size is thus important. This window size can be calibrated. Note that the calibration is computationally expensive because the boundary-crossing probability for each window size is different.

6.1.2. Extended BFAST

The spatial dependence parameter ρ of the SAR model (Equation 3) has a mean of 0.83 (0 indicates no correlation and 1 indicates perfect correlation) (standard deviation (SD) 0.06). The distribution of ρ (Figure 6, left) suggests strong spatial correlation within almost all the 3×3 spatial regions. The serial dependence parameter ϕ of the AR(1) model (Equation 2) has a mean of 0.06 (SD 0.1) (a median of 0.038). The distribution of ϕ (Figure 6) shows that most time series have weak serial correlation. Only 10% of ϕ are higher than 0.2, and suggests mild AR(1) serial correlation. This might be due to the time series of the land cover in this study case, such as tropical forest, urban area, and agricultural area, are noisy over a long time period after removing the trend and seasonality. The BFAST and extended BFAST can be applied on more strongly autocorrelated time series for further evaluation.

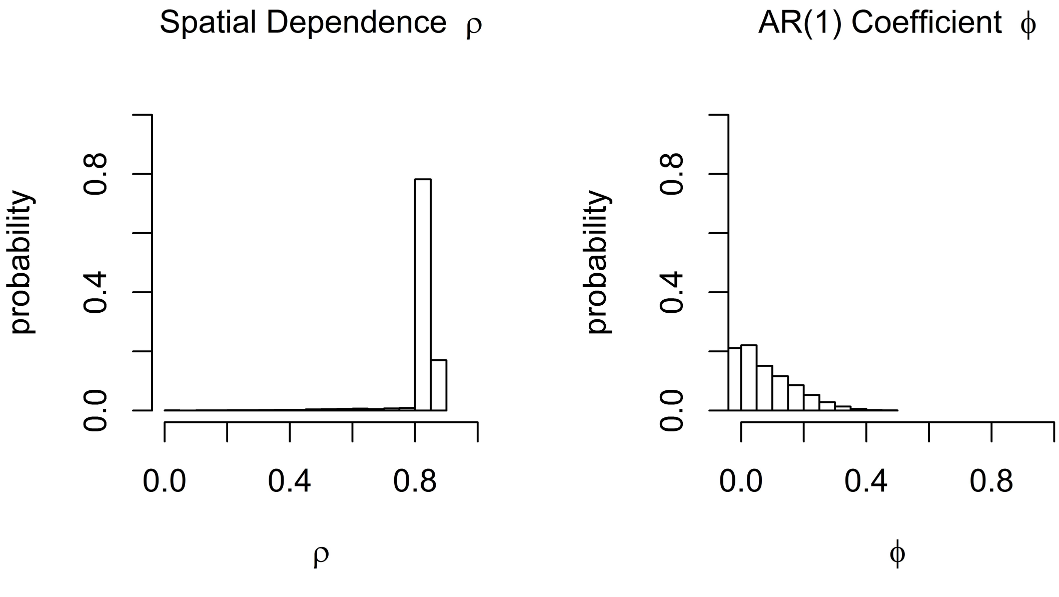


Figure 6: Spatial and temporal dependence parameters. Left: histogram of the spatial dependence parameter ρ in the SAR model (Equation 3), which has been applied on each of the 3×3 spatial region of time series. Right: histogram of the serial correlation parameter (ϕ) in the AR(1) model (Equation 2) of each time series.

Integration of the spatial model into the temporal process (Equation 4) smooths the regression residuals over a region. The effect of spatial neighbors in the change modeling process depends on the amount of spatial autocorrelation. In this study case, where strong spatial autocorrelations are present, the change is mainly the change in the correlation of the spatial neighborhood. Thus, the SAR integrated model may be more resistant to temporal noise. Changes may however go undetected when the spatial correlation remains unchanged, e.g. a patch of forest cells are clear cut and recovered simultaneously over a region. In this case, the border between deforestation and forest may be detected. This explains the results that the changes detected by SAR integrated model are more scattered. The residuals of the pure time series analysis and the residuals of the SAR integrated method can be added to include both the long-term temporal effects (e.g. drought), and the spatial effects for change detection.

The best result is achieved with both both OLS-MOSUM and OLS-MOSUM with AR(1) correction. This result suggests that in this forest degradation and deforestation case, BFAST is robust against mild serial correlation and spatial correlation. However, the potential of integrating spatial information to improve the pure time series analysis will need to be further investigated (Cressie and Wikle, 2011).

In the extended BFAST, the EFP process models fit a season-trend model at once. This extension is very similar as the near-real time monitoring approach presented by Verbesselt et al. (2012). The differ in the aspect the monitoring approach has been designed to detect change at the end of a time series (i.e. monitoring) while the BFAST and the extended version are able to detect changes (i.e. breakpoints) within the time series. The season-trend model is necessary for SAR integrated time series analysis in that the separate detrending and deseasonalization makes the time series less consistent.

6.1.3. Uncertainty in validation dataset

The PRODES, DETER, and DEGRAD systems from INPE are developed with manual image interpretation. The identification of forest degradation or deforestation is done by comparing a land cover classification map with Landsat or MODIS images. The results are randomly validated in field. However, the processes are unreproducible, and the precision of the interpretation is hard to assess. Simulated datasets could be used for a more rigorous validation process. In addition, the resolution of the data that is used is coarser than the PRODES and DEGRAD products, and is likely to miss information about changes. The use of higher resolution data can further contribute to the validation process.

6.2. Array abstraction for data analysis and scalability

Current studies apply BFAST on a RasterStack (e.g. the BfastSpatial package published on GitHub (Dutrieux et al., 2014)), which firstly constructs a time series from each raster pixel and then computes BFAST on each time series. This method enables parallel computation and visualization of results as raster layers. One limitation is that the raster is of 2-D, and the 3-D RasterStack is not flexible. For example, it is difficult to perform joined spatio-temporal analysis on 3-D partitions, only two-stage analysis can be performed. Multidimensional arrays facilitate implementing of higher dimensionality analysis (i.e. 3-D and beyond). In addition, the BFAST outputs can only be stored as values of each raster layer, which is inconvenient to have time as an additional dimension, since only one output variable (e.g. magnitude) can be stored; however, it is often useful to have a 3-D spatio-temporal array with all the necessary variables (e.g. significance of the trend, change magnitude, types of changes), so that the change coefficients at a certain time can be selected, analyzed and visualized. We store the desired variables as attributes of a 3-D result array (SciDB array) with space and time as dimensions, from where different variables can be stored and accessed. Lastly, SciDB array can be sparse, which is a more efficient way of storing sparse data (e.g. when only information at pixels with breakpoints is of interest).

In this study, the data analysis was entirely written in R, and scales in a simple way when using SciDB. The consistent data structure in SciDB and R make the process clear and relatively simple. From data preparation to result visualization, the data structure remains constant. The array abstraction facilitates the selection of particular spatio-temporal regions to apply statistical algorithms. Filtering and other sophisticated spatio-temporal data analysis methods can conveniently be applied to array partitions. The relationships between the spatio-temporal neighbors are naturally described by a weight matrix, and can be applied directly on selected regions with spatio-temporal statistical algorithms that are represented as linear models.

Instead of loading files and reconstructing the data structure (e.g. forming a time series from satellite imagery), the arrays are stored in the DMAS and can be used directly for data processing and analysis. To achieve the most efficient computation distribution with SciDB, the chunk size can be optimized. There are several computationally intensive processes in this modeling process:

1. The SAR model (Equation 4) estimates 72 regression parameters at once for each pixel neighbourhood (intercept, trend, and six harmonic parameter for each pixel, with 9 pixels in one region ($8 \times 9 = 72$)),
2. The BP method for timing and identifying the number of the breakpoints in time series include dynamic insertion of breakpoints to minimize the BIC and SSR.
3. The simulation of the boundary-crossing probability of OLS-MOSUM for different moving window sizes when choosing the most suitable window size.

These processes all become feasible with the scalability of DMAS. This study used 8 worker node instances of SciDB. Previous experiments (Câmara et al., 2014) suggest the speed of computation improve linearly with additional instances. A more powerful DMAS setup may allow for global scale computation, and realize near real-time large-scale change monitoring (Verbesselt et al., 2012).

Apart from using arrays in conjunction with an array database, the array itself is advantageous for parallel computation, which is supported in R. For example, the raster package supports parallel computing with the focal function, where the analysis are performed in each focal zone; function RasterEngine in package spatial.tools further extends the focal function of the raster package to 3-D and implements focal function in chunks (as an array of raster brick instead of a raster layer). The chunk of an array is suitable to be a processing unit. For example, to model a regional spatio-temporal phenomenon, each region can be put into a chunk and the process in each region can be parallelized.

This study is a first step towards a more long-term goal, which is to publish and communicate scientific large-scale earth observation studies in a completely transparent and reproducible fashion. One requirement for this is the use of open source software and scalable DMAS, which led to the choices for R and SciDB in this study. In addition to the R scripts provided here, this would require simple ways to install and operate array databases by third parties.

7. Conclusions

This paper discusses how arrays as an abstract data type can help modeling change. The array structure stores and analyzes spatio-temporal data as their natural form, which provides a clean and communicable process for data preparing, spatio-temporal change modeling and analyzing results. This study illustrates this capability of array structure by evaluating BFAST functions and extending BFAST from pixel-based time series analysis to region-based spatio-temporal analysis where the data are spatially or temporally correlated, and by storing BFAST outputs as a 3-D spatio-temporal array for post-processing and further analysis.

The study case was developed in forest degradation and deforestation modeling, which contribute directly to forest management and environmental conservation. These methods are subject to further tests with other datasets of higher spatial resolution or simulated datasets, to better control the validation process. The study is extensible and the methods can be the basis of a wide range of domains such as ecological, hydrological, or climate change modeling. In the next step the study will be extended to a higher dimensionality, e.g. by modeling change directly from multiple spectral bands.

References

Andrews, D. W., 1993. Tests for parameter instability and structural change with unknown change point. Econometrica 61 (4), 821–856.

Bai, J., Perron, P., 1998. Estimating and testing linear models with multiple structural changes. Econometrica 66 (1), 47–78.

Bai, J., Perron, P., 2003. Computation and analysis of multiple structural change models. Journal of Applied Econometrics 18 (1), 1–22.

Ban, Y., Gong, P., Giri, C., 2015. Global land cover mapping using earth observation satellite data: Recent progresses and challenges. ISPRS Journal of Photogrammetry and Remote Sensing 103, 1–6.

Banskota, A., Kayastha, N., Falkowski, M. J., Wulder, M. A., Froese, R. E., White, J. C., 2014. Forest monitoring using landsat time series data: A review. Canadian Journal of Remote Sensing 40 (5), 362–384.

Basseville, M., Nikiforov, I. V., 1993. Detection of abrupt changes: Theory and application. Prentice Hall Englewood Cliffs.

Baumann, P., 1994. Management of multidimensional discrete data. The VLDB Journal 3 (4), 401–444.

Bivand, R., Piras, G., 2015. Comparing implementations of estimation methods for spatial econometrics. Journal of Statistical Software 63 (1), 1–36.

Bolin, D., Lindström, J., Eklundh, L., Lindgren, F., 2009. Fast estimation of spatially dependent temporal vegetation trends using gaussian markov random fields. Computational Statistics & Data Analysis 53 (8), 2885–2896.

Broich, M., Hansen, M. C., Potapov, P., Adusei, B., Lindquist, E., Stehman, S. V., 2011. Time-series analysis of multi-resolution optical imagery for quantifying forest cover loss in Sumatra and Kalimantan, Indonesia. International Journal of Applied Earth Observation and Geoinformation 13 (2), 277–291.

Brown, R. L., Durbin, J., Evans, J. M., 1975. Techniques for testing the constancy of regression relationships over time. Journal of the Royal Statistical Society: Series B (Methodological) 37 (2), 149–192.

Câmara, G., Egenhofer, M., Ferreira, K., Andrade, P., Queiroz, G., Sanchez, A., Jones, J., Vinhas, L., 2014. Fields as a generic data type for big spatial data. In: Geographic Information Science. Vol. 8728 of Lecture Notes in Computer Science. pp. 159–172.

Castanedo, F., 2013. A review of data fusion techniques. The Scientific World Journal 2013, 1–19.

Chaudhuri, S., Dayal, U., 1997. An overview of data warehousing and OLAP technology. ACM Sigmod Record 26 (1), 65–74.

Chu, C.-S. J., Hornik, K., Kuan, C.-M., 1995. MOSUM tests for parameter constancy. Biometrika 82 (3), 603–617.

Cleveland, R. B., Cleveland, W. S., McRae, J. E., Terpenning, I., 1990. STL: A seasonal-trend decomposition procedure based on loess. Journal of Official Statistics 6 (1), 3–73.

Coppin, P., Jonckheere, I., Nackaerts, K., Muys, B., Lambin, E., 2004. Digital change detection methods in ecosystem monitoring: A review. International Journal of Remote Sensing 25 (9), 1565–1596.

Cressie, N., Wikle, C. K., 2011. Statistics for spatio-temporal data. John Wiley & Sons.

Cudre-Mauroux, P., Kimura, H., Lim, K.-T., Rogers, J., Simakov, R., Soroush, E., Velikhov, P., Wang, D. L., Balazinska, M., Becla, J., DeWitt, D., Heath, B., Maier, D., Madden, S., Patel, J., Stonebraker, M., Zdonik, S., 2009. A demonstration of SciDB: A science-oriented DBMS. In: The Proceedings of the VLDB Endowment. Vol. 2. pp. 1534–1537.

Dutrieux, L., DeVries, B., Verbesselt, J., 2014. Utilities to monitor for change on satellite image time-series. URL https://github.com/dutri001/bfastSpatial

Dutrieux, L. P., Verbesselt, J., Kooistra, L., Herold, M., 2015. Monitoring forest cover loss using multiple data streams, a case study of a tropical dry forest in Bolivia. ISPRS Journal of Photogrammetry and Remote Sensing 07, 112–125.

Forkel, M., Carvalhais, N., Verbesselt, J., Mahecha, M. D., Neigh, C. S., Reichstein, M., 2013. Trend change detection in NDVI time series: Effects of inter-annual variability and methodology. Remote Sensing 5 (5), 2113–2144.

Greenberg, J. A., 2014. spatial.tools: R functions for working with spatial data. R package version 1.4.8. URL http://CRAN.R-project.org/package=spatial.tools

Hansen, M. C., Shimabukuro, Y. E., Potapov, P., Pittman, K., 2008. Comparing annual MODIS and PRODES forest cover change data for advancing monitoring of Brazilian forest cover. Remote Sensing of Environment 112 (10), 3784–3793.

Hijmans, R. J., 2015. raster: Geographic data analysis and modeling. R package version 2.3-40. URL http://CRAN.R-project.org/package=raster

INPE, 2015a. DEGRAD: Mapping of forest degradation in the Brazilian Amazon. Last accessed: Jan 2015. URL http://www.obt.inpe.br/degrad/

INPE, 2015b. DETER: A near real-time forest degradation monitoring system of Brazilian Amazon. Last accessed: Jan 2015. URL http://www.obt.inpe.br/deter/

INPE, 2015c. PRODES: Deforestation estimates in the Brazilian Amazon. Last accessed: Jan 2015. URL http://www.obt.inpe.br/prodes/

Jiang, Z., Huete, A. R., Didan, K., Miura, T., 2008. Development of a two-band enhanced vegetation index without a blue band. Remote Sensing of Environment 112 (10), 3833–3845.

Jianya, G., Haigang, S., Guorui, M., Qiming, Z., 2008. A review of multi-temporal remote sensing data change detection algorithms. The International Archives of the Photogrammetry, Remote Sensing and

Spatial Information Sciences 37 (B7), 757–762.

Jong, R., Verbesselt, J., Schaepman, M. E., Bruin, S., 2012. Trend changes in global greening and browning: Contribution of short-term trends to longer-term change. Global Change Biology 18 (2), 642–655.

Kuan, C.-M., Hornik, K., 1995. The generalized fluctuation test: A unifying view. Econometric Reviews 14 (2), 135–161.

Lewis, B. W., 2015a. An R interface to SciDB.
URL https://github.com/Paradigm4/SciDBR.git

Lewis, B. W., 2015b. Run R programs within SciDB queries.
URL https://github.com/Paradigm4/r_exec.git

Leyshock, P., Maier, D., Tufte, K., 2013. Agrios: A hybrid approach to scalable data analysis systems. In: IEEE International Conference on Big Data. pp. 85–93.

Lucas, R., Honzak, M., Foody, G. M., Curran, P., Corves, C., 1993. Characterizing tropical secondary forests using multi-temporal landsat sensor imagery. International Journal of Remote Sensing 14 (16), 3061–3067.

MATLAB, 2015. MATLAB version 8.5.0.197613 (R2015a). The MathWorks Inc.

Mello, M. P., Vieira, C. A. O., Rudorff, B. F. T., Aplin, P., Santos, R. D. C., Aguiar, D. A., 2013. STARS: A new method for multitemporal remote sensing. IEEE Transactions on Geoscience and Remote Sensing 51 (4), 1897–1913.

Pebesma, E., 2012. spacetime: Spatio-Temporal data in R. Journal of Statistical Software 51 (7), 1–30.

Planthaber, G., Stonebraker, M., Frew, J., 2012. Earthdb: Scalable analysis of MODIS data using SciDB. In: ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data. pp. 11–19.

Ploberger, W., Krämer, W., 1992. The CUSUM test with OLS residuals. Econometrica 60 (2), 271–285.

Pontius, R. J., Boersma, W., Castella, J.-C., Clarke, K., de Nijs, T., Dietzel, C., Duan, Z., Fotsing, E., Goldstein, N., Kok, K., Koomen, E., Lippitt, C., McConnell, W., Mohd Sood, A., Pijanowski, B., Pithadia, S., Sweeney, S., Trung, T., Veldkamp, A., Verburg, P., 2008. Comparing the input, output, and validation maps for several models of land change. The Annals of Regional Science 42 (1), 11–37.

R Core Team, 2015. R: A Language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.
URL http://www.R-project.org/

Schabenberger, O., Gotway, C. A., 2004. Statistical methods for spatial data analysis. CRC Press.

Shimabukuro, Y. E., Santos, J. R., Formaggion, A. R., Duarte, V., Rudorff, B. F. T., 2012. The Brazilian Amazon monitoring program: PRODES and DETER projects. Global forest monitoring from earth observation, 153–169.

Stonebraker, M., Brown, P., Zhang, D., Becla, J., 2013. SciDB: A database management system for applications with complex analytics. Computing in Science & Engineering 15 (3), 54–62.

Tomlin, D. C., 1990. A map algebra. Harvard Graduate School of Design.

USGS, 2014. Surface reflectance 8-day L3 global 250m. Last accessed: May 2015.
URL https://lpdaac.usgs.gov/products/modis_products_table/mod09q1

Verbesselt, J., Hyndman, R., Newnham, G., Culvenor, D., 2010. Detecting trend and seasonal changes in satellite image time series. Remote Sensing of Environment 114 (1), 106–115.

Verbesselt, J., Zeileis, A., Herold, M., 2012. Near real-time disturbance detection using satellite image time series. Remote Sensing of Environment 123, 98–108.

Vermote, E. F., El Saleous, N. Z., Justice, C. O., 2002. Atmospheric correction of MODIS data in the visible to middle infrared: First results. Remote Sensing of Environment 83 (1), 97–111.

Viswanathan, G., Schneider, M., 2011. On the requirements for user-centric spatial data warehousing and SOLAP. In: Database Systems for Adanced Applications. Vol. 6637 of Lecture Notes in Computer Science. pp. 144–155.

Wickham, H., 2014. Tidy data. Journal of Statistical Software 59 (10), 1–23.

Zeileis, A., Hothorn, T., 2013. A toolbox of permutation tests for structural change. Statistical Papers 54 (4), 931–954.

Zeileis, A., Kleiber, C., Krämer, W., Hornik, K., 2003. Testing and dating of structural changes in practice. Computational Statistics & Data Analysis 44 (12), 109–123.

Zscheischler, J., Mahecha, M. D., Harmeling, S., Reichstein, M., 2013. Detection and attribution of large spatiotemporal extreme events in earth observation data. Ecological Informatics 15, 66–73.