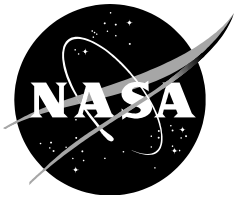
NASA/TP—2004–220501



Short Data Gap Filling Algorithm Prototype

Chandrasekaran, Hema
Ames Research Center, Moffett Field

December 2004

NASA STI Program ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

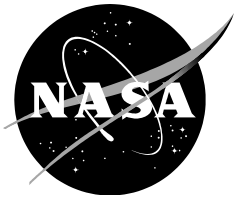
- Access the NASA STI program home page at http://www.sti.nasa.gov

- E-mail your question to help@sti.nasa.gov

- Phone the NASA STI Information Desk at 757-864-9658

- Write to:
  NASA STI Information Desk
  Mail Stop 148
  NASA Langley Research Center
  Hampton, VA 23681-2199

NASA/TP—2004–220501



Short Data Gap Filling Algorithm Prototype

Chandrasekaran, Hema
Ames Research Center, Moffett Field CA

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field CA

December 2004

Acknowledgments

This report is available in electronic form at
http://

KPO @ AMES DESIGN NOTE

Design Note No.: KADN-26067

Title: Short Data Gap Filling Algorithm Prototype

Author: H. Chandrasekaran Signature: _____

GS SE Approval: C. Middour Signature: _____

Science Approval: J. Jenkins Signature: _____

Distribution: C. Allen, S. Bryson, D. Caldwell, H. Chandrasekaran, J. Jenkins, C. Middour, D. Pletcher, K. Topka, R. Thompson, J. Voss

Revision History:

Rev. Letter	Revision Description	Date	Author/Initials
-	Original Release	12/29/2004	HC

Design Note No.:	KADN-26067	Rev.:	-	Date:	12/29/2004
Title:	Short Data Gap Filling Algorithm Prototype				
Author:	H. Chandrasekaran				

Overview:

This design note describes an algorithm to fill short data gaps encountered in pixel or flux time series using auto-regressive modeling techniques.

Recommendations:

This algorithm should be used in filling short data gaps encountered in pixel or flux time series whenever a complete time series is required as in the case of transiting planet search algorithm.

Reference Documents

1. B. Porat and B. Friedlander, "ARMA Spectral Estimation of Time Series with Missing Observations," *IEEE Transactions on Information Theory*, Vol. IT-30, No. 6, November 1984.
2. J. Jenkins, *Detecting Transits: NASA Ames Planetary Science Seminar*, September 6, 2000.
3. C. W. Therrien, Discrete Random Signals and Statistical Signal Processing, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1992.
4. http://www.star.le.ac.uk/~sav2/timing/define_time.html
5. K. Fukuda, H. E. Stanley; L. A. Amaral, *Heuristic Segmentation of a Nonstationary Time Series*, http://citebase.eprints.org/cgi-bin/citations?id=oai:arXiv.org:cond-mat/0308068

Applicable Documents

KSOC-21075	Transiting Planet Search Software Detailed Design
KSOC-21084	Pre-Search Data Conditioning Software Detailed Design
KSOC-21076	Reflected Light Planet Search Software Detailed Design
KSOC-21073	Photometric Analysis Software Detailed Design

Open Items/Action Required

1. Need a criterion to decide on the AR model order. Need to find whether AIC (Akaike's Information Criterion), MDL (Rissanen's Minimum Descriptor Length) are of any use for time series AR model order selection [3].
2. Need a method to decide on the correlation window length.

TBDs/TBRs

Paragraph	TBD Item

1 Brief Description of the Algorithm

This algorithm deals with the problem of missing observations from evenly sampled signals. Missing observations are caused by a variety of reasons, such as fading phenomena in propagation channels, intermittent sensor failures, periodic interferences, and removal of outliers-measurements with obvious gross errors [1]. The data gaps caused by such missing samples can occur randomly or in a regular pattern. The matched filter transit detection algorithm and the periodogram based detection algorithm for reflected light search require a complete time series. The missing data samples are estimated using an auto regressive (AR) model for a stationary time series [2]. This is represented in equation form as

$$\begin{aligned}\hat{x}[n] &= \left\{c_1 x[n-L-1] + c_2 x[n-L-2] + \ldots + c_p x[n-L-p]\right\} + \\ &\quad \left\{b_1 x[n+M+1] + b_2 x[n+M+2] + \ldots + b_q x[n+M+q]\right\} \\ &= \sum_{i=1}^{p} c_i x[n-L-i] + \sum_{i=1}^{q} b_i x[n+M+i]\end{aligned} \tag{1}$$

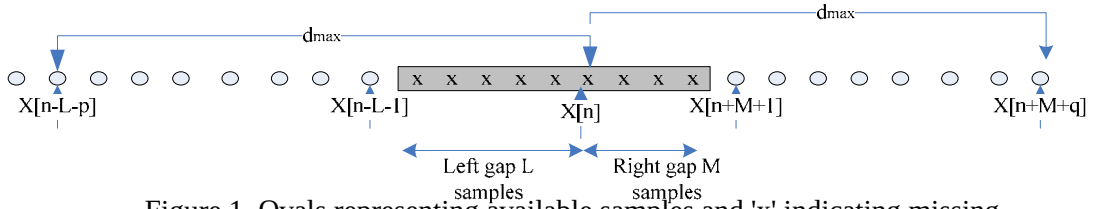


Figure 1. Ovals representing available samples and 'x' indicating missing samples in a time series.

Figure 1 represents equation (1) in pictorial form with a sliding window positioned over the missing sample $x[n]$ and extending d_{max} samples on either side of $x[n]$. Here $x[n]$ is estimated as a weighted sum of p previous available samples falling within the window and q later samples with the data gap being $(L+M)$ samples long. It is easily seen (by forming the sample correlation function from the time series **x**) that the correlation function (also the covariance function) of an AR process satisfies the difference equation (1). The AR model is unique in that among all possible random processes that could match the (p+q+1) given values of the correlation function and extend it to the gap, the AR process is the one that has maximum entropy. In other words, the AR process is the *most random* process that can still match the given correlation values. This property is the most appropriate for filling in missing data since it is presumptuous to force the correlation function to assume specific values (such as zero) in the region where it is not known and the AR model provides an extension which is maximally noncommittal [3].

1.1 Stationary Time Series

A random process is said to be *stationary* (in the *strict* sense) if its statistical descriptions do not depend on time. For example, the mean value should be independent of time, satisfying $E\left(x[t]\right) = E\left(x[t+T]\right)$ where $E(\cdot)$ represents an ensemble average operation. The above definition used the ensemble average to define the stationarity of a process. In many cases it is also possible to describe the statistical moments of a process using time averages. If the (time averaged) first and second moments (mean and auto-correlation function) do not change from realization to realization (i.e. between different members of the ensemble) the process is said to be ergodic. For ergodic processes, therefore, the time

averages (for example: the mean value) are equivalent to ensemble averages. Ergodic processes are useful because it means that the statistical properties can be measured using time averages over a single realization. In practice many stationary processes are ergodic [4].

For an ergodic, stationary random process, the joint probability density function $f_{x[n_0],x[n_1],\ldots,x[n_K]}$ for any set of K+1 samples must be the same for any other set of K+1 samples with the same spacing. This condition implies that all the moments (for example: mean, variance) for any value of K are the same with the same inter sample spacing. If this requirement is relaxed so that mean of the random process is a constant and the autocorrelation function is a function of the spacing between the samples, then the random process is said to be *wide-sense stationary* or *weakly stationary*. In common usage, the term stationary is usually taken to mean wide-sense stationary. Suppose that the time series **x** of length N is a discrete time random process and **x** is subjected to random skipping or deleting of some samples. The missing sample $x[n]$ (dependent variable) can be represented as a linear combination of available samples (independent variables). This equation is in the form of a statistical regression and since both dependent and independent variables belong to the same process, this random process **x** is called an *autoregressive* or AR process.

Repeating equation (1) for the sake of continuity,

$$\begin{aligned}\hat{x}[n] &= \left\{c_1 x[n-L-1] + c_2 x[n-L-2] + \ldots + c_p x[n-L-p]\right\} + \\ &\quad \left\{b_1 x[n+M+1] + b_2 x[n+M+2] + \ldots + b_p x[n+M+q]\right\} \\ &= \sum_{i=1}^{p} c_i x[n-L-i] + \sum_{i=1}^{q} b_i x[n+M+i] \\ &= \sum_{\substack{i=-(L+p) \\ i\neq\{-L,\ldots,M\}}}^{(M+q)} a_i\, x[n+i]\end{aligned} \tag{1}$$

Here the missing sample $x[n]$ is estimated as a weighted sum of p previous available samples (starting at time index (n-L-1)) and q later samples (starting at time index (n + M+1)) with the data gap being (L+M) samples long. The AR model order is $P = p+q$ [2].

Define the error function as $E = \left\langle \left(x[n] - \hat{x}[n]\right)^2 \right\rangle = \left\langle \left(x[n] - \sum_{\substack{i=-(L+p) \\ i\neq\{-L,\ldots,M\}}}^{(M+q)} a_i\, x[n+i] \right)^2 \right\rangle$ (2)

where the operator $\langle\ \rangle$ stands for time average over N samples of the time series.

The AR model parameters values of a_k that minimize the error function are found by taking the partial derivative of E with respect to a_k resulting in

$$\frac{\partial E}{\partial a_k} = \left\langle \left(x[n] - \sum_{\substack{i=-(L+p) \\ i\neq\{-L,\ldots,M\}}}^{(M+q)} a_i\, x[n+i] \right) \left(-x[n+k]\right) \right\rangle \tag{3}$$

Setting the partial derivative with respect to a_k to zero results in

$$\left\langle x[n]x[n+k]\right\rangle - \left\langle \sum_{\substack{i=-(L+p)\\ i\neq\{-L,...,M\}}}^{(M+q)} a_i\, x[n+i]x[n+k] \right\rangle = 0,\ k\begin{cases} = -(L+p), -(L+p+1), ...(M+q) \\ \neq -L, -L+1, ..1,2, ...M \text{ (missing samples index)} \end{cases} \tag{4}$$

Noticing that $\langle\ \rangle$ operates over time index *n*, the above equation can be rewritten as

$$\left\langle x[n]x[n+k]\right\rangle - \sum_{\substack{i=-(L+p)\\ i\neq\{-L,...,M\}}}^{(M+q)} a_i \left\langle x[n+i]x[n+k]\right\rangle = 0,\ k\begin{cases} = -(L+p), -(L+p+1), ...(M+q) \\ \neq -L, -L+1, ..1,2, ...M \end{cases} \tag{5}$$

Since the estimate of the autocorrelation function for lag k using time average is $r[k] = \left\langle x[n]x[n+k]\right\rangle$ and for lag (k+i) the estimate is $r[i-k] = \left\langle x[n+i]x[n+k]\right\rangle$, the above equation can be simplified as

$$r[k] - \sum_{\substack{i=-(L+p)\\ i\neq\{-L,...,M\}}}^{(M+q)} a_i\, r[i-k] = 0,\quad k\begin{cases} = -(L+p), -(L+p+1), ...(M+q) \\ \neq -L, -L+1, ..1,2, ...M \end{cases} \tag{6}$$

Writing equation (6) in matrix form,

$$\mathbf{r}_n = \mathbf{a} * \mathbf{R}_n \text{ and } \mathbf{a} = \mathbf{R}_n^{-1} * \mathbf{r}_n \tag{7}$$

Here it is important to remember that the autocorrelation function vector $\mathbf{r_n}$ does not contain the elements corresponding to the missing samples. By setting the missing samples to zero and including them in the autocorrelation function vector **r** computation, allows us to use the existing Matlab function calls. Then $\mathbf{r_n}$ can be extracted as a sub vector with elements whose indices correspond to existing samples only. The autocorrelation matrix $\mathbf{R_n}$ does not contain the rows or columns corresponding to indices of missing samples. This matrix too can be extracted as a sub block matrix from the autocorrelation matrix **R** containing all the indices, including the missing samples set to zero.

Once the AR model parameter vector **a** is determined by solving equation (7), the missing sample is estimated as the weighted sum of the previous *p* samples and later *q* samples according to equation (1). It is useful to remember that the correlation matrix **R** of a stationary discrete time random process is Hermitian and almost always positive definite (when there are no linear dependencies between samples) and thus the inverse of **R** exists for almost all practical cases.

1.1.1 Pseudo Code for Missing Data Prediction Algorithm (Stationary Time Series)

1. Replace the missing data values in the time series $\mathbf{x} = \{x[0], x[1], x[2], ..., x[N-1]\}$ with 0. Form the autocorrelation function vector **r** as

   $$r[k] = \frac{1}{N}\sum_{n=0}^{N-1} x[n]x[n+k],\quad 0 \le k \le N-1$$

   (The vector **r** is used to form the autocorrelation matrix **R** discussed in step 3. The entries in **R** corresponding to the indices of missing samples are not used. Yet, the missing samples need to be filled with 0 to account for the time lag).

2. Define the data fill window length *dmax* (AR model order) specifying the maximum distance the available points can be used in filling in each missing point.
3. Form the autocorrelation matrix **R** as a Toeplitz matrix formed from the elements of the vector **r**(1:2∗*dmax* +1). This is correct, since for a stationary random process, the correlation and covariance matrices do not change if (2∗*dmax*+1) samples are taken anywhere in the time series. Here the initial (2∗*dmax*+1) samples are used to form the autocorrelation matrix.
4. Get the distance of the *i*th non-available point from all available points. Collect all the available points that fall within the fill window. Let their indices be $\{(n-L-1, n-L-2, ..., n-L-p), (n+M+1, n+M+2, ..., n+M+q)\}$ where the window extends *dmax* on each direction from the missing data sample at time index *n*. Here it assumed that the data gap around the missing sample is *L*+*M* samples wide.
5. Set up equation (7) as follows: Collect all elements of **r** with indices $\{(L+1, L+2, ..., L+p), (M+1, M+2, ..., M+q)\}$ where *M*+*q* is the last greatest sample index inside the fill window. Now form $\mathbf{r}_n$ as follows:

$$r_n = \left\{ r[L+p], ..., r[L+1], \frac{\downarrow \text{at lag L}}{L} ..... \frac{\downarrow \text{at lag 0}}{\text{missing}} ..... \frac{\downarrow \text{at lag M}}{M} ....r[M+1], ..., r[M+q] \right\}$$

6. Extract a sub block matrix $\mathbf{R}_n$ from **R** by removing row and column entries corresponding to the missing data samples indices. This sub block matrix $\mathbf{R_n}$ is not Toeplitz any more but is still symmetric and positive semi definite and thus invertible.
7. Solve for the vector **a** in equation (7).
8. Estimate the missing data sample as a weighted sum of available data samples within the window where the weights are given by the vector **a**.
9. Repeat steps 4 through 8 for all missing samples.

1.2 Nonstationary Time Series

For a nonstationary time series, estimating the missing sample value is similar to the stationary time series except that the autocorrelation function vector **r** is computed adaptively since the correlation function (a second order moment) is now a function of time index. A correlation window length is defined (much greater than the AR model order, about ten times or so) within which the time series is considered to be *stationary*. The auto covariance matrix **R** is formed as a Toeplitz matrix from the vector **r**. Each missing sample is estimated as a weighted sum of previous *p* samples and *q* later samples around the missing sample, where the weight vector **a** is obtained by solving equation (7).

1.2.1 Pseudo Code for Missing Data Prediction Algorithm (Nonstationary Time Series)

1. Replace the missing data values in the time series $\mathbf{x} = \{x[0], x[1], x[2], ..., x[N-1]\}$ with 0.
2. Define the data fill window length *dmax* (AR model order) specifying the maximum distance the available points can be used in filling in each missing point. Define also the correlation window length *rdmax* which is much greater (for example: ten times) than *dmax*.

3. Get the distance of the *i*th non-available point from all available points. Collect all the available points that fall within the fill window. Let their indices be $\{(n-L-1, n-L-2, ..., n-L-p), (n+M+1, n+M+2, ..., n+M+q)\}$ where the window extends *dmax* on each direction from the missing data sample at time index *n*. Here it assumed that the data gap around the missing sample is *L*+*M* samples wide.
4. Collect all the available points that are within the correlation window of length *rdmax* around the missing sample. Compute the sample mean using the available samples only. Subtract the sample mean from the available samples in the correlation window. Form the autocovariance function vector **r** as $r[k] = \frac{1}{N_s}\sum_{n=0}^{N_s-1} x'[n]x'[n+k]$, $0 \le k \le N_s - 1$ where N_s is the number of points inside the correlation window (x' indicates mean removed available samples; missing samples are still 0 and the mean is not removed from the missing samples). The correlation window is much greater than the data fill window length *dmax*. (The vector **r** is used to form the autocorrelation matrix **R** discussed in step 5. The entries in **R** corresponding to the indices of missing samples are not used. Yet, the missing samples need to be filled with 0 to account for the time lag).
5. Form the auto covariance matrix **R** as a Toeplitz matrix formed from the 2∗*dmax* elements of the vector **r**. This is correct since for a stationary segment of a random process, the correlation and covariance matrices do not change if (2∗*dmax*+1) samples are taken anywhere in the stationary segment.
6. Get the distance of the *i*th non-available point from all available points. Collect all the available points that fall within the fill window. Let their indices be $\{(n-L-1, n-L-2, ..., n-L-p), (n+M+1, n+M+2, ..., n+M+q)\}$ where the window extends *dmax* on each direction from the missing data sample at time index *n*. Here it assumed that the data gap around the missing sample is *L*+*M* samples wide.
7. Set up equation (7) as follows: Form the sub vector $\mathbf{r_n}$ from **r** by collecting all elements of **r** with indices $\{(L+1, L+2, ..., L+p), (M+1, M+2, ..., M+q)\}$ where *M*+*q* is the last greatest sample index inside the fill window. Now form $\mathbf{r}_n$ as follows:
$$r_n = \left\{ r[L+p], ..., r[L+1], \underset{L}{\downarrow \text{at lag L}} ..... \underset{\text{missing}}{\downarrow \text{at lag 0}} ..... \underset{M}{\downarrow \text{at lag M}} ....r[M+1], ..., r[M+q] \right\}$$
8. Extract a sub block matrix $\mathbf{R}_n$ from **R** by removing row and column entries corresponding to the missing data samples indices. This sub block matrix $\mathbf{R_n}$ is not Toeplitz any more but still symmetric and positive semi definite and thus invertible.
9. Solve for the vector **a** in equation (7).
10. Estimate the missing data sample as a weighted sum of available data samples within the data window where the weights are given by the vector **a**. Add the sample mean of the stationary segment to the estimate.
11. Include the estimated sample into the available sample pool.
12. Repeat steps 3 through 10 for all the missing samples.

1.2.2 Data Gap Filling: Stationary versus Nonstationary Time Series

Stationary Time Series	Nonstationary Time Series
Autocorrelation function vector **r** is computed only once for the time series.	Auto covariance function vector **r** is computed as many times as there are missing points in the time series.
Newly estimated missing sample is not included in the autocorrelation vector for recalculation.	Newly estimated sample is added to the available sample pool and is included in the auto covariance vector computation for other missing samples.
Autocorrelation matrix is used in equation (7) to compute vector **a**.	Auto covariance matrix is used in equation (7) to compute vector **a**.

1.2.3 Points to Ponder (Issues to Investigate, Parameters to Optimize)

Need a criterion to decide on the AR model order. Need to find whether AIC (Akaike's Information Criterion), MDL (Rissanen's Minimum Descriptor Length) are of any use for time series AR model order selection [3].

Need a method to decide on the correlation window length.

Need to see if the approach to quantify a nonstationary time series as consisting of a number of time segments that are stationary is useful. (In general, it is impossible to obtain the exact segmentation of a nonstationary time series because of the complexity of the calculation. An exact segmentation algorithm requires a computation time that scales as $O(N^N)$, where N is the number of points in the time series. Several heuristic methods are reported in the literature, which deal with the problem of meaningfully segmenting a Nonstationary time series [5]).

2 Description of input data

2.1 Stationary Time Series

Two stationary time series of length 2048 samples are generated as a combination of two sinusoids (one with a frequency of 50 Hz, 1000 samples long and another with a frequency of 2 Hz and 400 samples long) in gaussian noise with standard deviation of 0.1, 0.5. Data gaps ranging from 8 to 20 samples are introduced every 200 samples interval. Thus control data time series with all samples intact and missing data time series are available for evaluating short data gap filling algorithm.

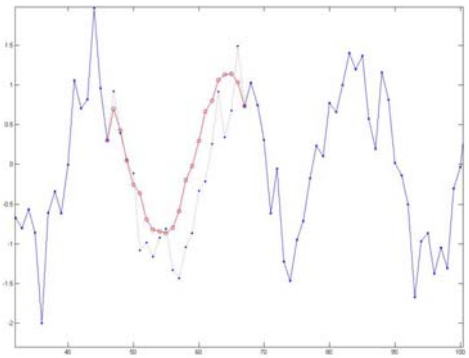
2.2 Nonstationary Time Series

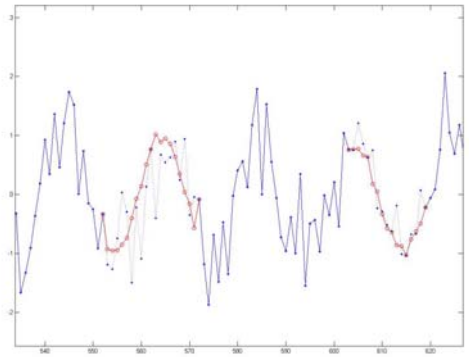
Measurements of solar irradiance values made by DIARAD (Dual Irradiance Absolute Radiometer) instrument aboard the SOHO aircraft (Solar and Heliospheric Observatory) is used as input data. The input data is truncated to 5000 samples before the first data gap in encountered. This is to make sure that the time series is complete for evaluating the data gap filling algorithm. As for the case of stationary time series, data gaps ranging from 8 to 20 samples are introduced every 200 samples interval.

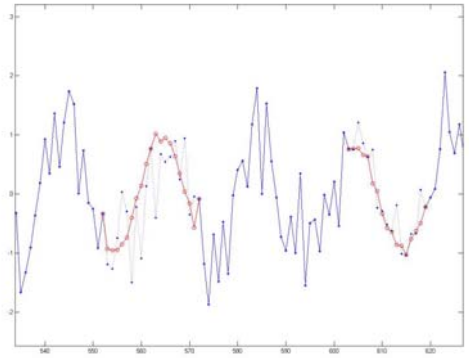
3 Expected Results

3.1 Filling Stationary Time Series Data Gap

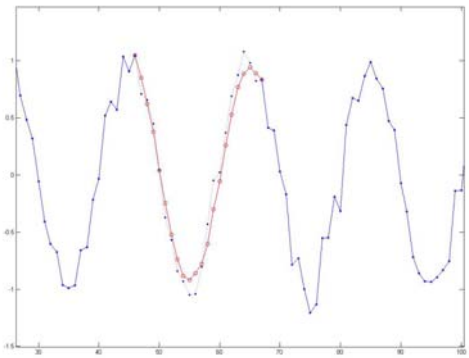
Time series with gaussian noise $\sigma = 0.5$

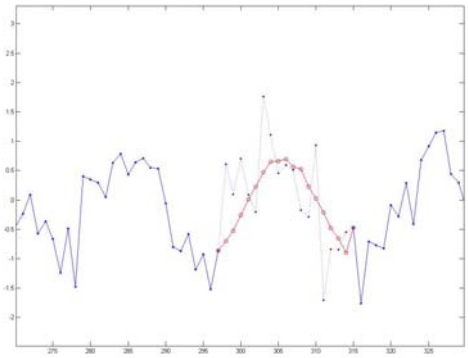


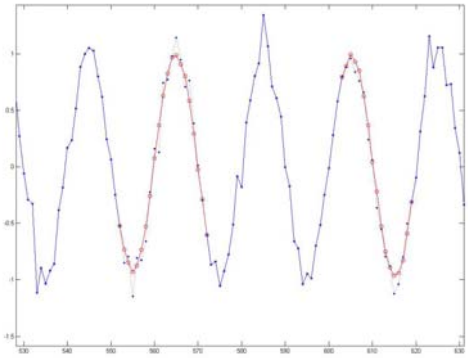




Time series with gaussian noise $\sigma = 0.1$







Design Note No.:	KADN-26067	Rev.:	-	Date:	12/29/2004
Title:	Short Data Gap Filling Algorithm Prototype				
Author:	H. Chandrasekaran				

Time series with gaussian noise $\sigma = 0.5$

Time series with gaussian noise $\sigma = 0.1$

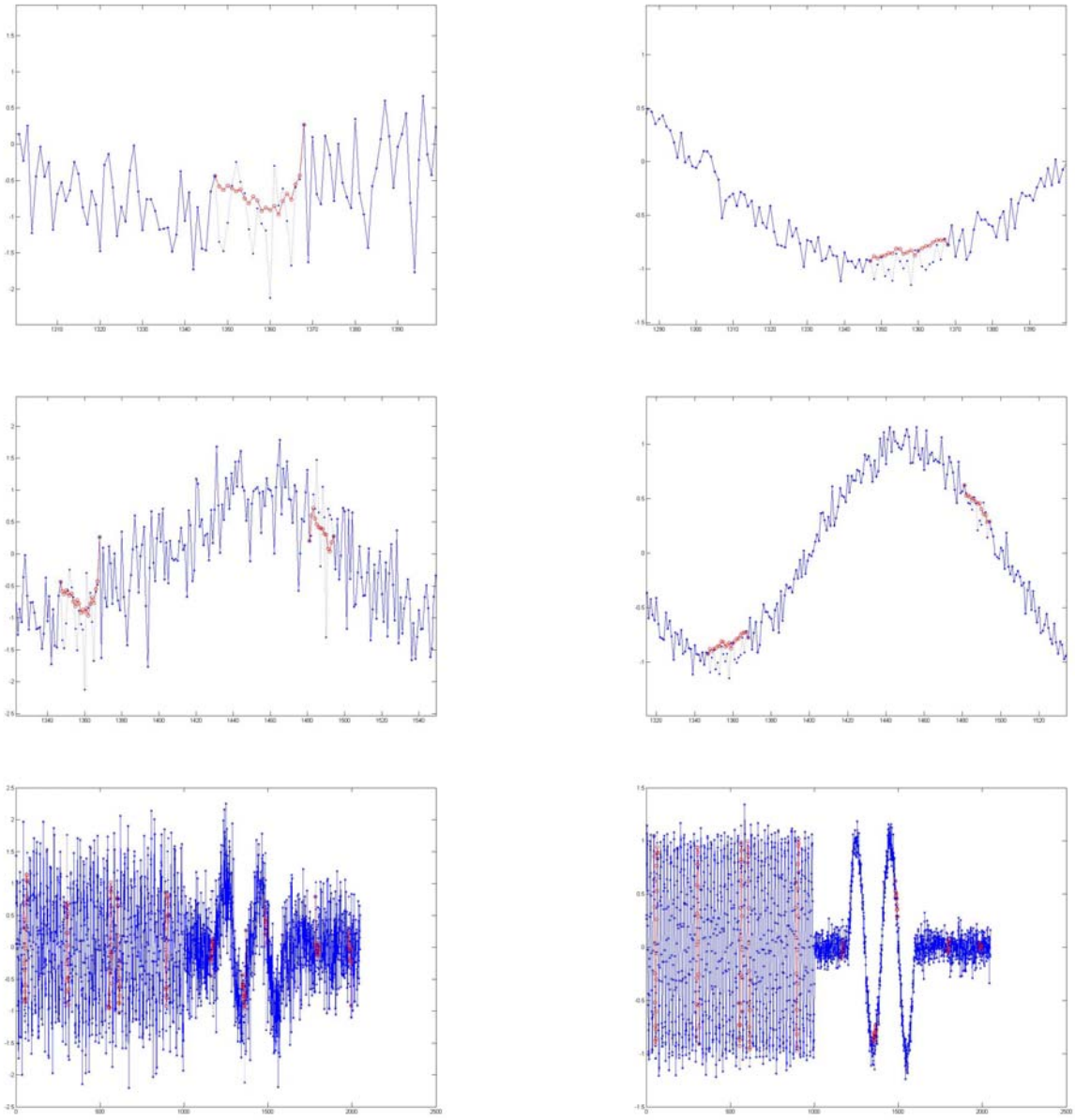
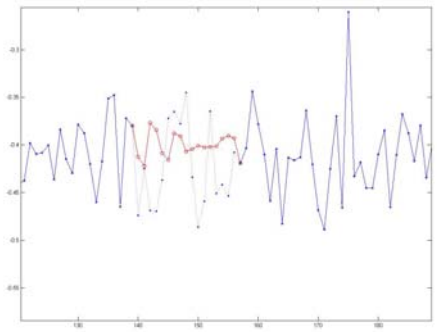
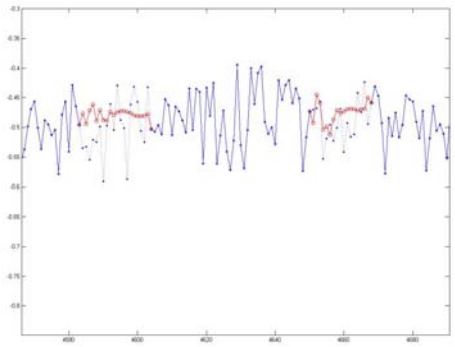
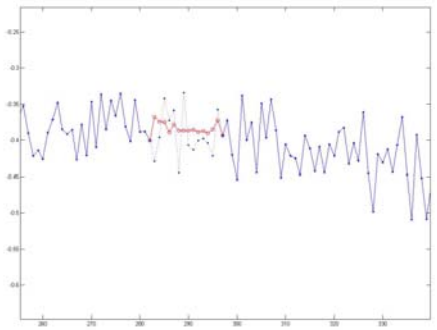


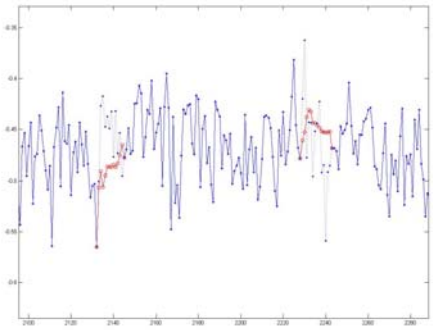
Figure 1. Filling data gaps in a stationary time series consisting of two sinusoids in gaussian noise with $\sigma = 0.1$ and $\sigma = 0.5$. (The two plots on the last row just show the two complete noisy time series)

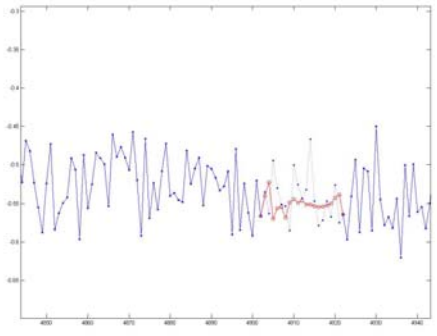
3.2 Filling Nonstationary Time Series Data Gap

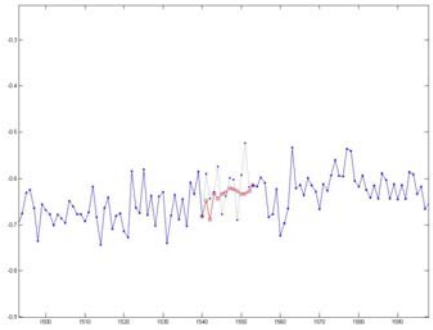












Design Note No.:	KADN-26067	Rev.:	-	Date:	12/29/2004
Title:	Short Data Gap Filling Algorithm Prototype				
Author:	H. Chandrasekaran				

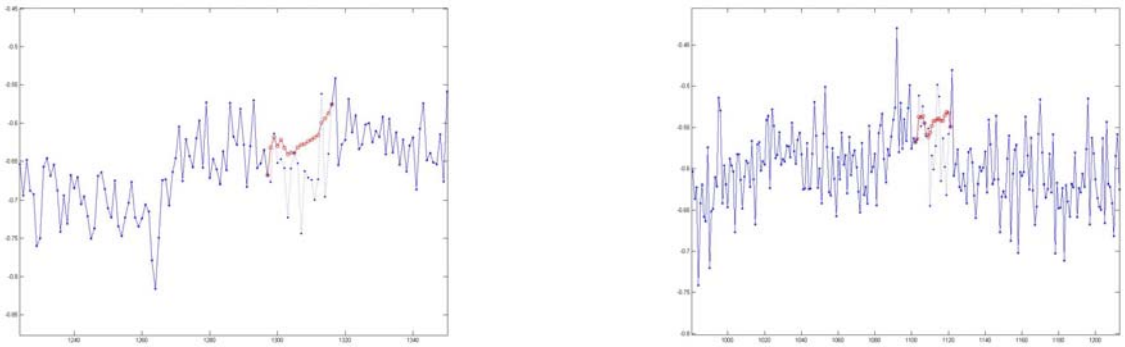


Figure 2. Filling data gaps in a nonstationary time series consisting solar irradiance values measured by DIARAD instrument aboard the SOHO spacecraft.

4 Method of evaluation

4.1 Stationary Time Series

Run the script [*MSE*] = *Batch_Fill_DataGap_Stationary*(*NRUNS*, *print_plots*, *sigma*) where *NRUNS* specifies the number of AR models of increasing order the stationary time series is fitted with to estimate the missing samples. The starting AR model order is 3. The *print_plots* flag is set to 1 when *NRUNS* < 10 and when it is set, ten plots are captured for each of the runs. The plots are saved as .jpg files in newly created directories under the current directory. The script *Generate_Stationary_TimeSeries.m* called by *Batch_Fill_DataGap_Stationary* generates the stationary time series as the sum of two sinusoids corrupted by gaussian noise with standard deviation *sigma*. In this time series, short gaps ranging in size from 5 samples to 20 samples are introduced every 200 samples of the time series randomly.

To get the plots in figure 1, run the script *Batch_Fill_DataGap_Stationary* twice; for the first iteration, *NRUNS* = 1 and *print_plots* = 1, and *sigma* = 0.5 and for the next iteration *NRUNS* = 1 and *print_plots* = 1, and *sigma* = 0.1. To decide what AR model estimates the missing data well, run the script *Batch_Fill_DataGap_Stationary* with *NRUNS* = 25 and *print_plots* = 0, and *sigma* = 0.5 or *sigma* = 0.1. The results are saved in a text file called *Stationary_Time_Series_25_Runs.txt* and as a plot called *Stationary_TimeSeries_ModelOrder_Selection.jpg*

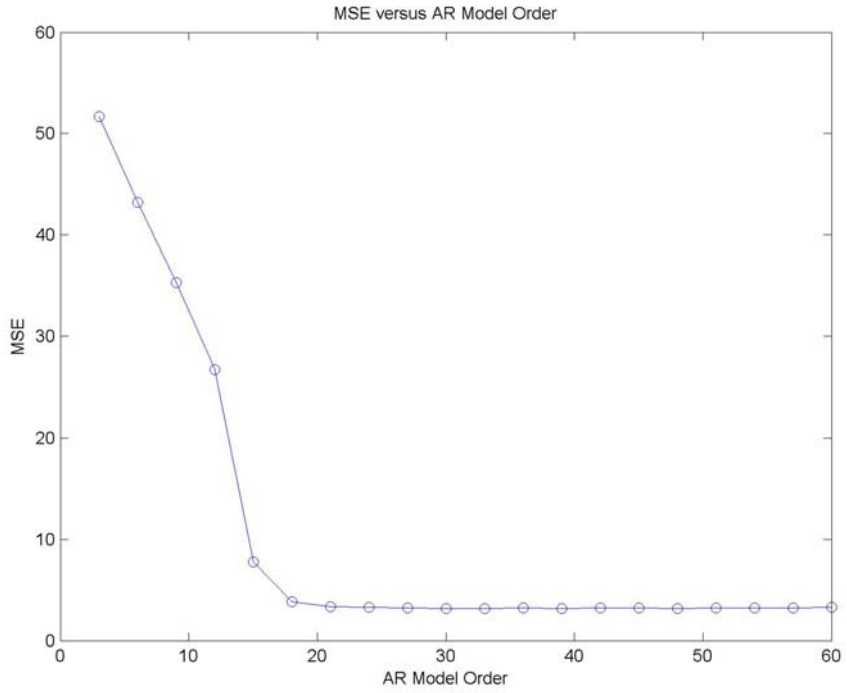


Figure 3. AR Model Order versus Mean Square Error (MSE) of fit in a stationary time series (two sinusoids with additive gaussian noise with $\sigma = 0.1$)

4.2 Nonstationary Time Series

Run the script [*Runs*] = *Batch_Fill_DataGap_NonStationary*(*NRUNS1, NRUNS2, print_plots*) where *NRUNS1* specifies the number of AR models of increasing order the nonstationary time series is fitted with to estimate the missing samples. The starting AR model order is 3. *NRUNS2* specifies the number of correlation windows of increasing length to be used for each AR model order. The starting correlation window length is 15*AR model order. The *print_plots* flag is set to 1 when *NRUNS1* NRUNS2* < 20 and when it is set, ten plots are captured for each of the runs. The plots are saved as .jpg files in newly created directories under the current directory. The script *Generate_NonStationary_TimeSeries.m* called by *Batch_Fill_DataGap_NonStationary* generates the nonstationary time series from the *solnew.mat* (a Matlab workspace containing the solar flux measurements made by DIARAD instrument aboard SOHO spacecraft). This time series has very long data gaps. For prototyping purposes, 5000 initial samples before any data gap is encountered, are used. In this time series, short gaps ranging in size from 5 samples to 20 samples are introduced every 200 samples of the time series randomly.

To get the plots in figure 2, run the script *Batch_Fill_DataGap_NonStationary* with *NRUNS1* = 1, *NRUNS2* = 1, and *print_plots* = 1. To decide what AR model and correlation window length estimates the missing data well, run the script *Batch_Fill_DataGap_NonStationary* with *NRUNS1* = 25, *NRUNS2* = 15 and *print_plots* = 0 The results are saved in a text file called *NonStationary_Time_Series_25_15_Runs.txt* and as plots called *NonStationary_TimeSeries_ModelOrder_Selection1.jpg* (a 3D line plot, as shown in figure 4)and *NonStationary_TimeSeries_ModelOrder_Selection2.jpg* (a mesh plot, as shown in figure 5)

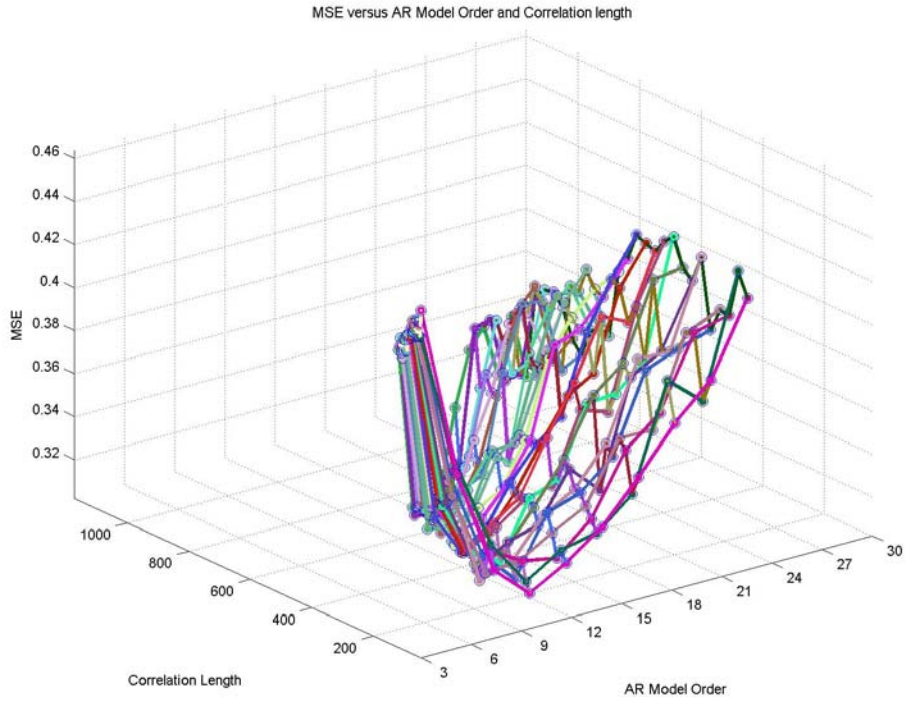


Figure 4. Nonstationary time series: AR model order and correlation window length versus MSE of fit (*NonStationary_TimeSeries_ModelOrder_Selection1.jpg)*

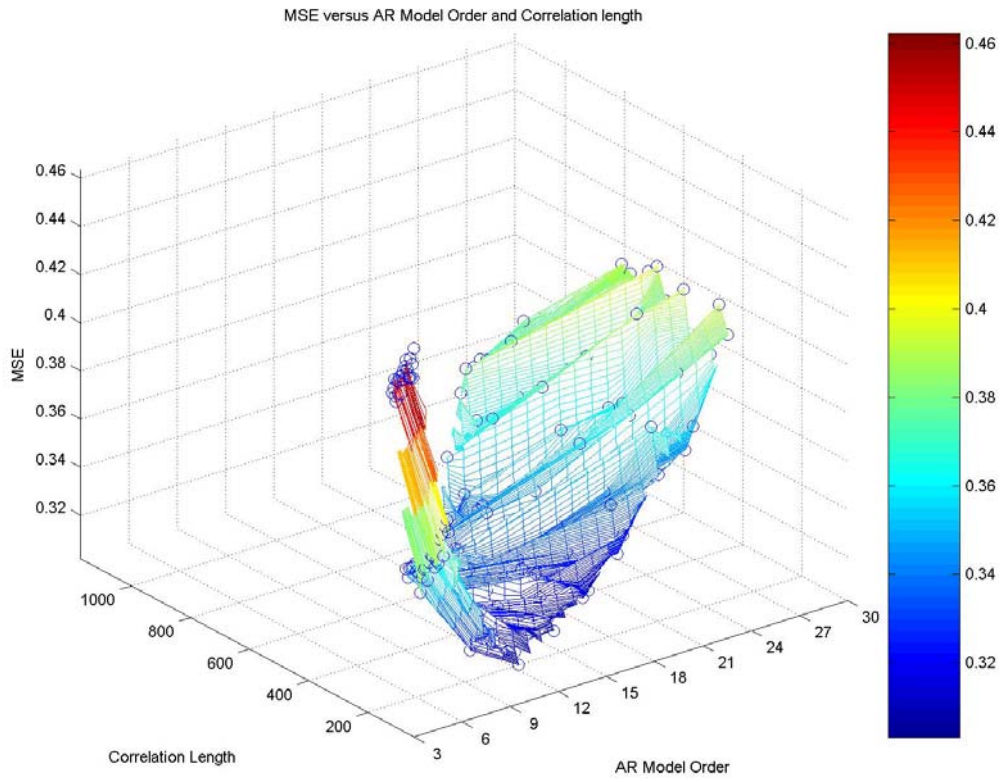


Figure 5. Nonstationary time series: AR model order and correlation window length versus MSE of fit (*NonStationary_TimeSeries_ModelOrder_Selection2.jpg)*

5 Location of source code

The following scripts have been checked into the source control system CVS residing on the machine flux.arc.nasa.gov under the directory \home\cvs\so\algorithms\prototype\short_data_gap:

1. Batch_Fill_DataGap_NonStationary.m
2. Batch_Fill_DataGap_Stationary.m
3. Generate_Stationary_Time_Series.m
4. Generate_NonStationary_Time_Series.m
5. Fill_Short_DataGap_1.m
6. Fill_Short_DataGap_2.m
7. Plot_TimeSeries_With_Gap.m
8. Get_Correlation.m

6 Location of test data

The scripts *Generate_Stationary_Time_Series.m* and *Generate_NonStationary_Time_Series.m* generate test data as well. These scripts can be modified (in the case of stationary time series) easily to yield a time series with different lengths, different frequency sinusoids, and gaussian noise with different variance.

7 Performance Measurements

As the data gaps in the time series are filled, plots of original samples, missing samples, and estimated samples are shown on the screen for each data gap. These plots are also captured as .jpg files (if *print_plots* option is set). This should serve as a visual check. The quality of the filled in samples is measured by the MSE of fit which in turn depends on the AR model and the correlation window chosen. These parameters can be optimized as described in sections 5.1 and 5.2.

8 Flowchart, Class diagrams