


RESEARCH

Open Access



# GMA: Gap Imputing Algorithm for time series missing values

Abd Alhamid Rabia Khattab<sup>1\*</sup> , Nada Mohamed Elshennawy<sup>1</sup> and Mahmoud Fahmy<sup>1</sup>

\*Correspondence:  
PG\_38835@f-eng.tanta.edu.eg

<sup>1</sup> Computer and Automatic  
Control Department, Faculty  
of Engineering, Tanta University,  
Tanta, Egypt

## Abstract

Data collected from the environment in computer engineering may include missing values due to various factors, such as lost readings from sensors caused by communication errors or power outages. Missing data can result in inaccurate analysis or even false alarms. It is therefore essential to identify missing values and correct them as accurately as possible to ensure the integrity of the analysis and the effectiveness of any decision-making based on the data. This paper presents a new approach, the Gap Imputing Algorithm (GMA), for imputing missing values in time series data. The Gap Imputing Algorithm (GMA) identifies sequences of missing values and determines the periodic time of the time series. Then, it searches for the most similar subsequence from historical data. Unlike previous work, GMA supports any type of time series and is resilient to consecutively missing values with different gaps distances. The experimental findings, which were based on both real-world and benchmark datasets, demonstrate that the GMA framework proposed in this study outperforms other methods in terms of accuracy. Specifically, our proposed method achieves an accuracy score that is 5 to 20% higher than that of other methods. Furthermore, the GMA framework is well suited to handling missing gaps with larger distances, and it produces more accurate imputations, particularly for datasets with strong periodic patterns.

**Keywords:** Time series, Incomplete subsequence, Missing data imputation

## Introduction

Sensing technologies are now widely employed in computer engineering for persistent and collaborative monitoring of the physical environment. These sensors generate large geo-tagged time series data, which can be used to improve human understanding of various ambient conditions including water level, stream flow observation, and meteorological conditions [1]. Missing values can occur for a variety of reasons, including sensor failures, transmission errors, power outages, and other technical issues. When dealing with time series data, missing readings can be especially problematic because they can disrupt real-time monitoring and compromise the accuracy of further data analysis, such as prediction and inference [2]. Living missing values untreated can cause incorrect or ill-defined results [3]. In order to address missing values in time series data, there are a number of techniques that can be used. One approach is to use imputation methods, which involve estimating missing values based on other data points in the time series. Another approach is to use interpolation methods, which involve estimating

missing values by interpolating between nearby data points. Both of these techniques can be effective, but it is important to choose the right approach based on the specific characteristics of the time series data and the goals of the analysis [4]. It is important to understand the pattern of missing values, as this can have an impact on the analysis of the data. Two common types of missing data patterns are [5]:

- (1) Missing at Random (MAR): In this type of missing data, the probability of a value being missing is dependent on other variables in the dataset, but not on the missing value itself. That means, the missing value is related to the observed values in the dataset.
- (2) Missing Completely at Random (MCAR): In this type of missing data, the probability of a value being missing is unrelated to any other variables in the dataset, including the observed values. That means, the missing values are completely random and not related to the values in the dataset.

There is also a third type of missing data pattern, called Missing Not at Random (MNAR), where the probability of a value being missing is dependent on the missing value itself, but this is less commonly encountered in practice. Understanding the pattern of missing values is important because it can impact the analysis of the data, and different techniques are used to handle different types of missing data patterns. There are many methods that can be used to handle missing data. Missing values can be handled in a dataset through either single imputation or multiple imputation methods. Single imputation methods involve replacing missing data points with a single value, and the most common techniques include mean, average, or median imputation. On the other hand, multiple imputation methods create multiple values for the missing data points [6]. Each of these methods has its advantages and disadvantages, and the choice of technique depends on the specific characteristics of the data and the research question being investigated.

Effective approaches for predicting missing values from accessible data are needed. Algorithms for recovering missing data blocks can utilize various techniques such as matrix completion principles or pattern matching. Matrix completion-based algorithms treat a set of series as a matrix and apply methods that aim to complete the missing entries. On the other hand, pattern-matching algorithms utilize the observed values from the sensors to replace the missing data blocks. By using these methods, algorithms can reconstruct missing data blocks from the available information, allowing for more complete and accurate data analysis [7].

A commonly used approach for replacing missing data gaps involves utilizing the values from the most similar subsequence. This technique falls under the category of pattern-matching algorithms; the Dynamic Time Warping (DTW) algorithm is a highly effective pattern-matching technique that is extensively employed across numerous problem domains. Nonetheless, a drawback of employing DTW is its tendency to be computationally expensive and time-consuming, which can impact the algorithm's overall efficiency and performance. Some researchers have proposed a solution to the computational expense of Dynamic Time Warping (DTW) by suggesting the use of shape-feature extraction algorithms that extract sequence features in sliding windows.

This approach involves calculating DDTW only if the correlation between the shape features of the window and the subsequences before the missing gap is sufficiently high [8]. Results from this method have demonstrated better outcomes when dealing with time series that have strong seasonality and high correlation. It is worth noting that while DTW can identify the most similar patterns with similar dynamics, it can warp the shape by expanding or compressing, which can lead to the position of the missing gaps not aligning with the original pattern's position, to overcome this limitation we decided to employ Inverse Fourier Transform to predict the length of the seasonal period beforehand. This allows us to understand the dataset's characteristics and identify the length of the missing gap, particularly if it represents one or more seasonal periods, so we can apply a suitable algorithm to deal with it.

The objective of this study is to create novel methods for accurately and efficiently imputing missing or anomalous data in computer systems. Our focus is on developing techniques to impute missing values in seasonal time series. We observe that seasonal patterns tend to exhibit similarities over time. Therefore, we propose to leverage this phenomenon by replicating the pattern from the most comparable subsequence in the historical data. Through this approach, we have developed an effective method for imputing missing data, which involves utilizing simple operations for pattern searching and matching. The paper makes the following technical contributions:

- We present and formalize GMA: Gap Imputing Algorithm to impute missing values in time series, which covers stationary, nonlinear relationships, and seasonal time series.
- In computer engineering, we use the inverse Fourier transform to determine the periodic length of each time series. This helps us gain a better understanding of the dataset's characteristics and identify suitable algorithms for handling the missing gaps. We then apply appropriate algorithms to address these gaps.
- To identify similar historical situations, we rely on techniques such as the Euclidean distance, Spearman's Rank-Order Correlation, and Kendall's tau, which enable us to measure the similarity between patterns accurately. By leveraging these techniques, we can improve the accuracy and efficiency of our data analysis and make more informed decisions based on the insights gained.
- We empirically show on real-world and synthetic datasets that GMA outperforms state-of-the-art solutions, and it is capable of effectively imputing values in time series with extended blocks of consecutively missing values.

The remainder of the paper is organized as follows: "[Related work](#)" section describes the associate works. The model overview in "[Overview](#)" section. "[Proposed methods](#)" section shows the detail of proposed method. "[Experimental set-up](#)" section shows the details datasets, comparative methods, experiment setting, and evaluation, and results are displayed in "[Results and discussion](#)" section. Finally, the paper is concluded in "[Conclusion](#)" section.

## Related work

Missing data imputing as shown in Table 1 involves two primary implementation techniques: univariate and multivariate. The univariate approach makes use of a single variable to estimate the missing values, whereas multivariate techniques analyze the relationship between multiple variables to estimate missing data [9]. Multivariate methods can be further classified into three categories: matrix completion principles, pattern matching [7], and machine learning imputing. To provide a more comprehensive overview of the related work in missing data imputation, we have classified the existing algorithms into four categories, which are as follows:

### Univariate methods

Hong [10] proposed a method called "MLBUI" for filling consecutive missing values in univariate time series using machine learning methods. The data before and after the gap are transformed into multivariate time series, followed by forward and backward forecasting using ML methods to estimate the missing values. The imputation of the gap is then done by taking the average values of both forecast sets. Paternoster [11] explained that the "most frequent value" imputation technique consists of substituting missing data with the value that appears most frequently for the given variable. This imputation strategy is typically applied to categorical variables or numerical variables with a finite set of possible values. Kulanuwat [12] studied missing data imputation in electronic health records (EHR) using three methods: mean imputation, regression imputation, and multiple imputation. Mean imputation involves replacing missing values with the mean of available data, while regression imputation uses a regression model to predict missing values. Multiple imputation generates multiple plausible imputed datasets using a statistical model and combines them for a single estimate. The study found that multiple imputation was the most effective method for imputing missing data in EHR, producing estimates closer to true values and with less bias compared to the other methods.

### Pattern matching

Yi [13] proposed a method called spatio-temporal multi-view-based learning (ST-MVL) to fill missing readings in geo-sensory time series data. The method takes into account the temporal correlation between readings at different timestamps in the same series and the spatial correlation between different time series. The method combines empirical statistic models (Inverse Distance Weighting and Simple Exponential Smoothing) with data-driven algorithms (User-based and Item-based Collaborative Filtering) to handle different types of missing data cases. The method is evaluated using Beijing air quality and meteorological data. Wellenzohn [14] proposed a method for continuously imputing missing values in streams of time series data. The proposed method, called CIViC (Continuous Imputation of Values in time series with Clustering), uses a clustering algorithm to group similar time series and then uses the grouped time series to impute missing values. The authors evaluated their method on several real-world datasets and compared it to other imputation methods. Zhang and Thorburn [15] proposed a dual-head sequence-to-sequence (Seq2Seq) model for imputing missing values in time series data. Seq2Seq models are a type of recurrent neural network (RNN) that can be

**Table 1** Related work summary

	References	Datasets	Description	
Univariate methods	[10]	4 time series used: CO2 concentrations, Phu Lien air temperature, NNGC1 F1 V1 003 (NNGC), and Ba Tri temperature	This approach involves transforming data into a multivariate time series, using machine learning for forward and backward forecasting to estimate missing values, and imputing gaps with the average of both forecast sets. It adheres to academic standards of syntax and grammar	
	[11]	Scientific research data on factors causing crime for males and females	The "most frequent value" imputation method replaces missing data with the mode of the variable. It is typically used for categorical variables or numerical variables that have a limited range of values	
	[12]	Water-level data from telemetry stations across Thailand	This study compared three methods for imputing missing data: mean imputation, regression imputation, and multiple imputation. The results showed that multiple imputation was the most effective method and produced less biased estimates compared to the other two methods	
Multivariate methods	Pattern matching	[13]	Air quality and meteorological data in Beijing, China	The proposed ST-MVL method fills missing readings in geo-sensory time series data by considering temporal and spatial correlations. It uses empirical statistic models and data-driven algorithms to handle different types of missing data cases
		[14]	2 datasets: SBR meteorological time series in South Tyrol and Flights dataset	Clustering algorithm was employed to group similar time series, and the resulting groups were used to impute missing values. This approach is specifically designed for continuous streams of time series data

**Table 1** (continued)

	References	Datasets	Description
	[15]	Data collected from in-situ monitoring station in Mulgrave-Russell catchment, Australia	The proposed method involves using a Seq2Seq model to impute missing values in time series data. This model utilizes a dual-head architecture that includes an encoder and two decoders, each corresponding to one direction of the time series data. Seq2Seq models are a type of recurrent neural network (RNN) that can be applied for sequence prediction and generation
Matrix completion principles	[16]	Face images under varying illuminations: 168 × 192 resolution, 55 frames	The proposed technique for robust low-rank matrix recovery is capable of handling data corruption and utilizes orthonormal subspace learning to estimate a low-rank matrix from incomplete or corrupted data. This method has shown promising results in experiments and outperformed existing methods, and can be applied in various applications such as image processing, signal processing, and recommendation systems
	[17]	Netflix data: 17,770 movies rated by 480,189 customers	The proposed method utilizes spectral regularization to promote low-rank solutions and impose structural constraints on the estimated matrix. This approach has proved to be effective in dealing with ill-posed problems and improving the performance of matrix completion. The method also allows for incorporating additional side information, such as similarity between items or users, to further enhance the estimation

**Table 1** (continued)

	References	Datasets	Description
	[18]	Hydrological time series with tuples of timestamp and observation value	SVD and CD are two widely used methods for imputing missing values in time series data. SVD decomposes the dataset into a subset of singular values, while CD calculates the distance between the missing value and its neighboring points based on correlation. CD has been found to be more accurate and computationally for time series datasets with low correlation
	[19]	National water quality reference index data monitored by Haimen Bay station	A proposed approach for imputing missing values in data involves combining low-rank matrix completion and sparse representation. The approach first uses low-rank matrix completion to impute missing values based on a low-rank structure assumption. Then, sparse representation is employed to refine the imputed data by assuming it can be represented as a linear combination of a few basis elements
	[20]	Rainfall data from 4 stations in Malaysia	The method combines PCA and Bayesian modeling to estimate missing values in a dataset
Machine learning imputing	[21]	Measured water levels in 7 monitoring wells in the USA	RF algorithm used for imputing missing values in a dataset with continuous variables
	[22]	Three field-based time series were used, including traffic speed data, water flow rate data, and the Nottem dataset	A hybrid approach was used to impute missing data, where regression imputation predicted missing input variables, and data augmentation created synthetic data points for missing output variables. The approach was applied to a dataset with missing values in both input and output variables

**Table 1** (continued)

References	Datasets	Description
[23]	Seven datasets were used from the UCI and KEEL repositories	A genetic algorithm is proposed to impute missing values in datasets with multiple missing observations and different data types. The algorithm minimizes a multi-objective fitness function based on Minkowski distance of statistical measures between available and completed data
[24]	Two untargeted metabolomics datasets from the COPDGene cohort were used	A two-step approach was used for imputing missing values, involving a random forest classifier to classify the missing mechanism and mechanism-specific algorithms for imputation. The approach improved imputations by reducing bias and producing values closer to the original data
[25]	Letter and SPAM datasets <a href="https://archive.ics.uci.edu/">https://archive.ics.uci.edu/</a>	A method that estimates missing values in datasets using a generative adversarial network (GAN) model

used for sequence prediction and generation tasks. In their study, Zhang and Thorburn used a dual-head architecture, which includes an encoder and two decoders, to predict the missing values in a time series dataset. The two decoders correspond to the two directions of the time series data (forward and backward).

### Matrix completion principles

Shu et al. [16] proposed a method for robust low-rank matrix recovery that can handle data corruption. Low-rank matrix recovery is a fundamental problem in computer vision and machine learning, and it involves estimating a low-rank matrix from corrupted or incomplete data. The proposed method is based on orthonormal subspace learning, which is a technique for finding the principal subspace of a given set of data. Mazumder, Hastie, and Tibshirani [17] proposed a method for matrix completion, which involves recovering missing entries in a matrix. The authors noted that matrix completion has important applications in various fields, including recommender systems and collaborative filtering. The proposed method is based on spectral regularization and involves solving a convex optimization problem. Khayati, Böhlen, and Cudré-Mauroux [18] compared two methods, Singular Value Decomposition (SVD) and Correlation Distance (CD), for recovering missing values in time series datasets. The authors noted that missing data are a common problem in time series datasets and can have a significant impact on subsequent analysis. SVD and CD are two methods that can be used for



imputing missing values in time series data, and they differ in how they select a subset of the data to use for imputation. The authors evaluated the two methods on several datasets and found that CD performed better than SVD in terms of accuracy and computational efficiency, especially when the time series data had low correlation. Jianlong Xu [19] proposed a method for imputing missing data in high-dimensional datasets by combining low-rank matrix completion and sparse representation. The authors argued that the high dimensionality of the data and the sparsely of the missing values require an approach that can effectively capture the underlying structure of the data. The proposed method first utilized low-rank matrix completion to impute the missing values in the data matrix, leveraging the assumption that the data have a low-rank structure. The method then employed sparse representation to refine the imputed data, utilizing the assumption that the data can be represented as a linear combination of a few basis elements. Lai and Kuok [20] suggested employing a statistical method known as Bayesian Principal Component Analysis (BPCA) to perform imputation of missing values in rainfall data. BPCA is a technique that merges Principal Component Analysis (PCA) with Bayesian modeling to estimate missing data points in a dataset.

#### **Machine learning imputing**

Dwivedi [21] used Random Forest (RF) to impute continuous missing values in a dataset. The RF algorithm was used to impute missing values in a dataset containing continuous variables. The performance of RF imputation was compared with other imputation methods, such as k-nearest neighbors (KNN) and mean imputation. Bokde [22] proposed a method for imputing missing values in a dataset using a hybrid approach that combines regression imputation and data augmentation. The study deals with a dataset that had missing values in both the input and output variables. They used regression imputation to impute the missing values in the input variables by predicting them based on the available data. For the missing values in the output variables, they used data augmentation, which involves creating synthetic data points to fill in the missing values. A genetic algorithm approach to estimate missing data in multivariate databases is proposed [23]. Genetic algorithms are effective at handling multiple missing observations and different types of data, unlike traditional methods that only deal with univariate continuous data. The proposed algorithm minimizes a new multi-objective fitness function based on Minkowski distance of means, variances, covariances, and skewness between available and completed data. The approach is compared to EM algorithm and auxiliary regressions using a continuous/discrete dataset, and benchmarked against seven datasets. Jonathan [24] designed imputation algorithm to handle missing values in metabolomics datasets, which are often caused by various mechanisms such as instrument detection limits, data collection and processing conditions, and random factors. The algorithm takes a mechanism-aware approach and consists of two steps. In the first step, a random forest classifier is used to classify the missing mechanism for each missing value in the dataset. In the second step, missing values are imputed using mechanism-specific imputation algorithms, namely MAR/MCAR or MNAR. Simulations were conducted using complete data and different missing patterns to test the performance of the proposed algorithm. Results showed that the two-step approach reduced bias and provided imputations that were closer to the original data compared to using a single

imputation algorithm for all missing values. Overall, this mechanism-aware imputation algorithm offers a promising solution for handling missing values in metabolomics datasets and improving downstream analyses. Trubitsyna and Irina [25] developed a method for estimating missing values in datasets through the use of a generative adversarial network (GAN)-based model named DEGAIN. The performance of DEGAIN is evaluated on two publicly available datasets, namely Letter Recognition and SPAM, and compared against existing methods.

In this paper, we propose a novel approach for computing the missing values in incomplete subsequences, called Gap Imputing Algorithm (GMA). We divide the time series into two subsequences: one that contains the complete data and another that contains the missing gaps. To fill in the missing data, we use a pattern-matching approach by analyzing the similarity between the complete and incomplete subsequences. Specifically, we imitate the pattern of the complete subsequence to recreate the missing data.

## Overview

We have developed a method to recover missing values in an incomplete time series  $S$ , where readings are taken at equal intervals. Our approach involves identifying gaps in  $S$  that have null values, and dividing  $S$  into two parts:  $S_f$  which has no missing values, and  $S_m$ , which contains the missing gaps  $W$ . Then, we utilize the Fourier transform [26] on the  $S_f$  in order to obtain the number of readings in the periodic sequence  $P$  for each time series.  $P$  represents the number of readings required for the time series to complete one cycle. Each missing gap  $w$  is then analyzed to determine its surrounding right pattern  $R$ , which comprises  $P$  readings, and its left pattern  $L$ , which also comprises  $P$  readings.  $W$  contains  $N$  missing values. To identify the two patterns in the  $S_f$  set that are most similar to  $R$  and  $L$ , we utilize the Kendall tau correlation measure [27]. Subsequently, we employ the algorithm outlined in the following section to complete the missing values in  $W$ . Table 2 explains the symbols and annotations.

## Proposed methods

We have developed a novel method for imputing missing values in a time series using Fourier transform and a new filling algorithm. Our approach involves using Fourier transform to determine the wavelength of each time series, followed by identifying the sequence period for each series. This enables us to use an optimal imputation method to fill in the gaps in the time series. Our proposed GMA method consists of four main steps:

### Identifying the missing gaps

We identify a missing gaps in a time series denoted by  $W = \{w_1, w_2, w_3, \dots\}$ . For each gap  $w$ , we determine the preceding and succeeding data points, as well as the number of missing points in the gap between them. By performing this analysis, we generate an array  $G$  that records the number of missing data points for each gap, as well as the preceding and succeeding data points.

**Table 2** List of abbreviations

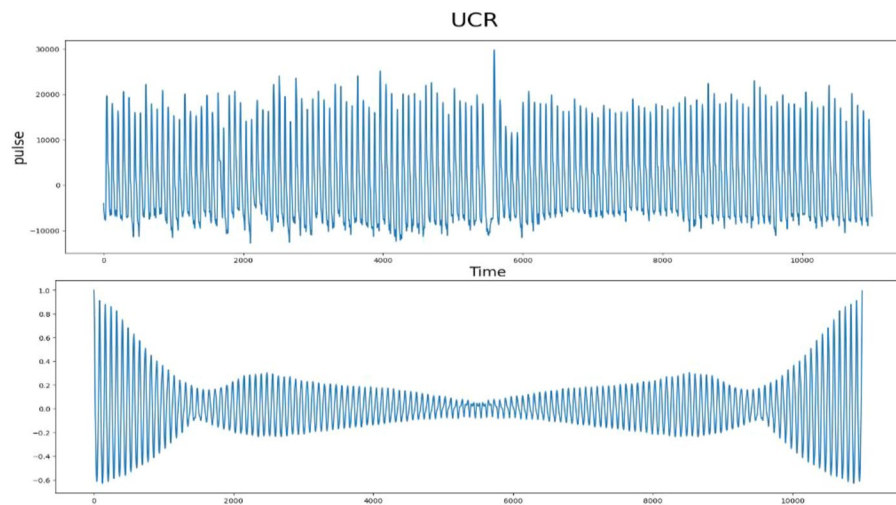
Symbol	Definition
$GMA$	The Gap Imputing Algorithm
$MAR$	Missing at random
$MCAR$	Missing completely at random
$MNAR$	Missing not at random
$S$	The total time series
$S_f$	The subsequence with no missing values
$S_m$	The subsequence with missing values
$P$	The number of readings in the periodic sequence for the longest component within the time series
$W$	Missing gaps
$R$	The right pattern for the missing gap
$L$	The left pattern for the missing gap
$N$	The number of readings in the missing gap
$w_l$	the most similarity pattern to $L$ in $S_f$
$w_r$	the most similarity pattern to $R$ in $S_f$
$b$	the start index for $w_l$
$e$	the end index for $w_r$
$s_r$	Correlation value between $R$ and $w_r$
$s_l$	Correlation value between $L$ and $w_l$
RMSE	mean squared error
MAE	mean absolute error
FSM	Full Subsequence Matching algorithm

### Time series analysis

After identifying the gaps in the time series  $S$  that have null values, and dividing  $S$  into two parts, it is typically advisable to focus on the subset of the time series that contains complete data, which we denote as  $S_f$ . This is because incomplete or missing data can introduce biases and inaccuracies into the analysis. A fundamental characteristic that needs to be identified is whether the data are stationary or (seasonal) periodically repeated, and the number of readings  $P$  in the periodic cycle. The discrete Fourier Transform (DFT) is a mathematical technique that analyzes the time series in the frequency domain. By performing the DFT on the time series, it decomposes it into its constituent frequencies and obtain information about the spectral content of the time series [26]. We use the DFT output as input for the inverse DFT Python function [28]; this function calculates the peak frequency of the signal using the *argmax* function and then determines the period of the signal by taking the reciprocal of the frequency. Figure 1 shows the URC and its inverse DFT.

### Extract the similar subsequences to surrounding right and left pattern R,L

Once we have generated the array  $G$  in the first step that records the missing gaps, we can use this array to extract subsequences from the left and right sides of each gap  $w$ . Specifically, we extract a subsequence of length  $P$  points from the left side of the gap (denoted by  $L$ ) and another subsequence of the same length from the right side of the gap (denoted by  $R$ ). Then, we use these subsequences to search for the most similar

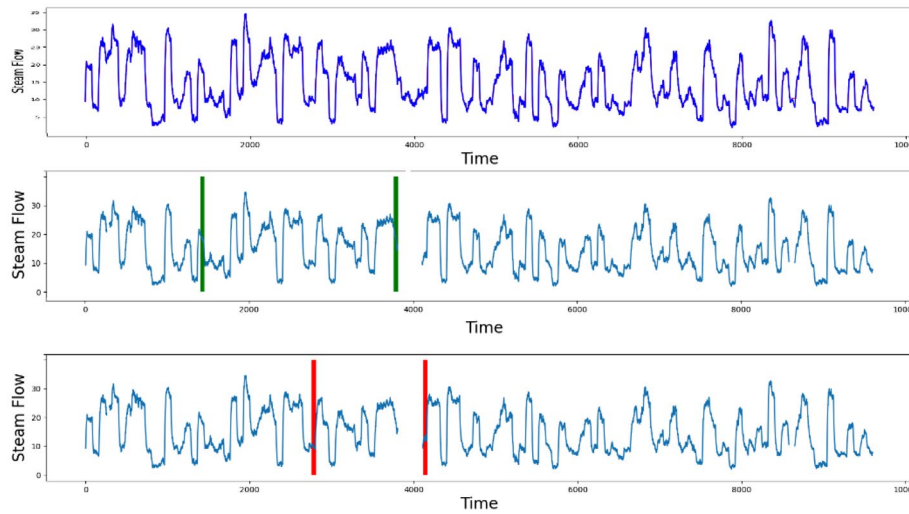


**Fig. 1** URC dataset [33] and its inverse DFT

subsequences to  $L$  and  $R$ , denoted by  $w_l$  and  $w_r$ , respectively. This similarity search can be performed using various techniques, such as dynamic time warping [29], Pearson correlation [30], or Euclidean distance [31]. Experiments have shown that the most suitable for our algorithm is Kendall tau technique [32] as it compares the direction of the points, up or down, in value.

In general, if the data are normally distributed and the relationship between the variables is expected to be linear, Pearson correlation may be the most appropriate technique to use. If the data are not normally distributed or the relationship between the variables is not expected to be linear, Euclidean distance or Kendall tau may be more appropriate. However, the specific technique used should be selected based on the characteristics of the data and the research question being investigated.

The Kendall rank correlation coefficient is a statistical measure that is used to determine the degree of similarity between two sets of variables. It is a nonparametric measure that is used to quantify the strength of the relationship between two sets based on the ranks of their values. The coefficient ranges from  $-1$  to  $1$ , with  $-1$  indicating a perfect negative correlation,  $0$  indicating no correlation, and  $1$  indicating a perfect positive correlation. Figure 2 shows the Kendall results. The similarity pattern to the left,  $w_l$ , is represented by a green rectangle. Its corresponding similarity measure,  $s_l$ , is  $0.735$ . The similarity pattern to the right,  $w_r$ , is represented by a red rectangle. Its corresponding similarity measure,  $s_r$ , is  $0.540$ .



**Fig. 2** Steam flow dataset [34] and the Kendall results:  $w_l$  is the most similar for L pattern (green rectangle) and  $w_r$  is the most similar for R pattern (red rectangle)

**GMA algorithm**

```

GMA Algorithm
Input:  $S_f$  the complete sequence
            $b$  the start index for ( $w_r$  the similarity to R)
            $s_r$  correlation value between R and  $w_r$ 
            $e$  the end index for ( $w_l$  the similarity to L)
            $N$  number of missing points
            $s_l$  correlation value between L and  $w_l$ 
FUNCTION GMA( $w, b, s_r, e, s_l, N$ )
// Calculate the tuas retio
 $T_1 = s_r / (s_r + s_l)$ 
 $T_2 = s_l / (s_r + s_l)$ 
// Method 1: Filling the gap with mutation
imp1 = []
FOR  $i = 0$  TO ( $\text{FLOOR}(T_1 * N) - 1$ ) DO
imp1[ $i$ ] =  $S_f[e+i]$ 
END FOR
FOR  $i = \text{FLOOR}(T_1 * N)$  TO  $N$  DO
imp1[ $i$ ] =  $S_f[b-(T_1 * N)+ i]$ 
END FOR
// Method 2: Filling the gap with combination
imp2 = []
FOR  $i = 0$  TO  $N$  DO
imp2[ $i$ ] =  $(T_1 / N * S_f[e+i]) + (T_2 / N * S_f[b-N+i])$ 
END FOR
// Return the results of both methods
RETURN (imp1, imp2)
END FUNCTION
    
```

### Imputation the gaps

We developed two different techniques to impute missing values: muting imputation and ratio imputation, as shown in Algorithm 1. The algorithm takes the missing gap  $w$ , the similarity pattern to the right  $w_r$  and its corresponding similarity measure  $s_r$ , the similarity pattern to the left  $w_l$  and its corresponding similarity measure  $s_l$ , and the number of missing points  $N$ . The algorithm uses two methods to fill in the missing gap. Method 1 of the algorithm employs the two similar patterns,  $w_r$  and  $w_l$ , to fill the missing gap. Specifically, the gap is filled by  $w_r$  from 1 to  $s_r * N / (s_r + s_l)$  and filled by  $w_l$  from  $s_r * N / (s_r + s_l)$  to  $N$ . The time complexity of Method 1 is approximately  $O(N)$ . Method 2 combines the two similar patterns  $w_r$  and  $w_l$  based on their correlation values  $s_r$  and  $s_l$ , to fill the gap. The time complexity of Method 2 is approximately  $O(N)$ . The computational complexity of the GMA function is linear with respect to the length of the missing gap  $N$ .

## Experimental setup

### Dataset

- (1) The UCR\_BIDMC1\_2500 benchmark is a time series dataset that is part of the UCR Time Series Anomaly Archive [33]. It contains 2500 instances, each consisting of 128 observations. The dataset was collected from intensive care unit (ICU) patients, where each instance represents the continuous physiological signals of a patient over a 6-h period. The anomalies in this dataset correspond to changes in the patients' physiological conditions that require medical attention, such as cardiac arrest or shock. This benchmark dataset is specifically designed to address common flaws present in other anomaly detection benchmarks, including trivial and unrealistic anomaly intensity, misleading ground truth, and running to failure bias. Using the inverse Fourier transformer technique, we were able to extract the periodic cycle of the UCR\_BIDMC1\_2500 dataset, which was determined to be 50 readings in length. Figure 1 shows the UCR\_BIDMC1\_2500 dataset and its inverse DFT.
- (2) The Steamgen dataset is a commonly used benchmark dataset in the field of process control and system identification the dataset consists of 6000 samples, each containing 19 features that describe the operating conditions and performance of the steam generator [34]. These features include variables such as steam flow rate, water level, and temperature, as well as indicators of system faults and disturbances. As shown in Fig. 2, we have chosen to focus on the steam flow feature. Using the inverse Fourier transformer technique, we were able to extract the periodic cycle of the steam flow dataset, which was determined to be 497 readings in length.

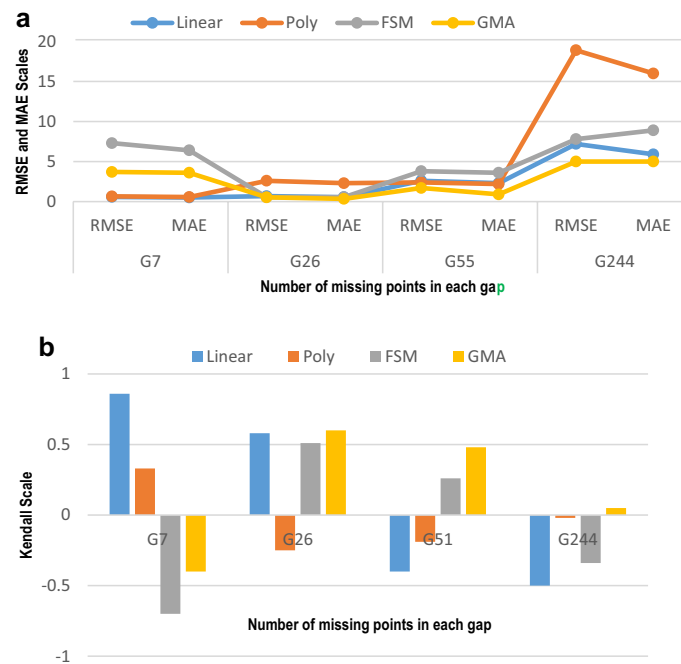
### Missing data generation

We simulated missing data in order to enable us to evaluate the efficacy of different imputation techniques. To generate datasets with missing data, we systematically removed consecutive values from the dataset, assuming that the deletions occurred randomly.

In the case of the URC dataset, we created gaps of sizes 6, 21, 26, 40, and 101 points, as the periodic cycle of this dataset is 50 points. For the steam flow feature, we created gaps of sizes 7, 21, 51 and 244 points, as the periodic cycle for steam flow is 507 points. These various missing data scenarios were simulated in order to test the performance of different imputation approaches. Experiments have shown that if the size of the missing gap is greater than the periodic cycle value; then, the error in imputation is significantly higher using any imputation method.

**Comparative imputation methods**

We have selected several widely used imputation techniques for evaluating the effectiveness of our proposed methods. These techniques include linear interpolation [35] and polynomial interpolation [36], and Full Subsequence Matching (FSM) [9]. Linear interpolation is a common method for filling in Fig. 3. The missing values in the stream flow dataset are estimated by utilizing the linear relationship between adjacent data points. Polynomial interpolation is a more complex variant of this technique, where a polynomial function is used to approximate the missing values based on the surrounding data points. FSM is a pattern-matching approach to imputation, where the missing value is estimated by identifying similar subsequences of data within the dataset and using them to make a prediction. This technique is useful for datasets with repeated patterns or cyclical trends. By comparing the performance of our methods against these established techniques, we aim to demonstrate the efficacy of our approach and provide valuable insights into the most effective methods for imputing missing data.



**Fig. 3** Performance indexes of 4 methods on steam flow dataset

### Experimental setting

Our experiments were conducted on a server equipped with Core i7 Intel processors running at 2.60 GHz, 8 GB RAM, and a 250 GB SATA hard drive. We implemented our proposed framework using the open source Python package *missval*, which offers a range of missing value imputation methods, as well as visualization and performance evaluation tools. The package is publicly available on Github at <https://github.com/Eng-Khattab/missval>. In addition, we have made the two datasets used in this study available for public access. To perform interpolation, we utilized the "interpolate" class from the pandas DataFrame [35] Python library, which offers a convenient method for filling in missing values using interpolation techniques. Specifically, we employed a linear approach for linear interpolation and a polynomial method with a second-order polynomial for polynomial interpolation. To perform the subsequence matching (FSM) methods, we used the matrix profile python library called STUMPY [34].

### Evaluation metrics

The performance of an imputation method is commonly evaluated by measuring its accuracy using three widely-used metrics: root mean square error (RMSE), mean absolute error (MAE), Kendall's tau measure between the actual pattern and the imputed pattern. These metrics are defined as follows:

- (1) The root mean square error (RMSE) [37] is a measure of the differences between the actual and imputed values, calculated as the square root of the average of the squared differences:

$$RMSE = \sqrt{\frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(y^{\wedge} - yi)^2}$$

Where  $y^{\wedge}$  is the predicted value and  $y$  is actual value.

- (2) The mean absolute error (MAE) [37] is another measure of the differences between the actual and imputed values, calculated as the average of the absolute differences:

$$MAE = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(y^{\wedge} - yi)$$

- (3) *Kendall's tau* [27] is a measure of the correlation between the actual and imputed patterns, which takes into account the order or rank of the values rather than their actual magnitudes. It ranges between  $- 1$  (perfect negative correlation) to  $1$  (perfect positive correlation), with  $0$  indicating no correlation:

$$Kendall's\ tau = (\text{number of concordant pairs} - \text{number of discordant pairs}) / (\text{number of pairs})$$

where a pair is concordant if the relative order of the values in the actual pattern is the same as in the imputed pattern, and discordant if the order is different. The number of pairs is equal to  $n(n - 1)/2$  for a dataset with  $n$  samples.



### Results and discussion

We employed four distinct algorithms for filling the missing data points in two datasets. For the steam flow dataset, there are gaps of sizes 7, 21, 51, and 244 points, as the periodic cycle for steam flow is 507 points. Our results, presented in Tables 3 and 4 and Figs. 3a and b, demonstrate the superiority of our algorithm for larger gaps that exceed a quarter of the identified time series period length, as outlined in "Overview" section. Conversely, the linear algorithm performs exceptionally well for smaller gaps, particularly at lower gap levels. These findings emphasize the importance of accurately determining the time period for dataset before imputing the missing gaps.

The results were presented in two tables and two figures. The first table and figure show the percentage of root mean squared error (RMSE) and mean absolute error (MAE), where lower values indicate better performance. The Kendall correlation analysis results are presented in both the second table and accompanying figure. A higher correlation value indicates a stronger relationship between the compensated data and the original data, and then, the compensated data closely align with the original data in terms of their characteristics. Our findings indicate that achieving better results with the Full Subsequence Matching (FSM) algorithm may require a significant number of repetitions. This is because the algorithm relies on random selection of the length of the right and left patterns, which can lead to variability in the outcomes.

Our algorithm employs two distinct methods to fill in missing data gaps. The first method involves alternating between two patterns based on their *tuas* and is utilized when the gap size exceeds the periodic cycle *p*. Conversely, the second method involves combining the two patterns based on their *tuas* to fill the missing gap less than the periodic cycle.

The results which include in Tables 5 and 6 and Fig. 4a and b show the effectiveness of our model when applied to URC data set, with the exception of gap 26. It is worth noting that gap 26 is an anomaly in the original data. The values presented in Tables 5 and 6 may appear to be large due to the wide domain of the dataset, which ranges from -10,000 to 30,000 as shown in Fig. 1a. As a result, the errors may also be expressed in large values.

**Table 3** Performance indexes of 4 methods on steam flow dataset

Gap points		G7		G26		G51		G244	
Error metrics		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Algorithm	Linear	0.58	0.51	0.68	0.55	2.6	2.3	7.2	5.9
	Poly	0.65	0.57	2.6	2.3	2.4	2.19	18.9	16
	FSM	7.3	6.4	0.55	0.55	3.8	3.6	7.8	8.9
	GMA	3.7	3.6	0.53	0.34	1.7	0.9	5	5

**Table 4** Kendall *tau* results between imputing gaps and actual data for steam flow dataset

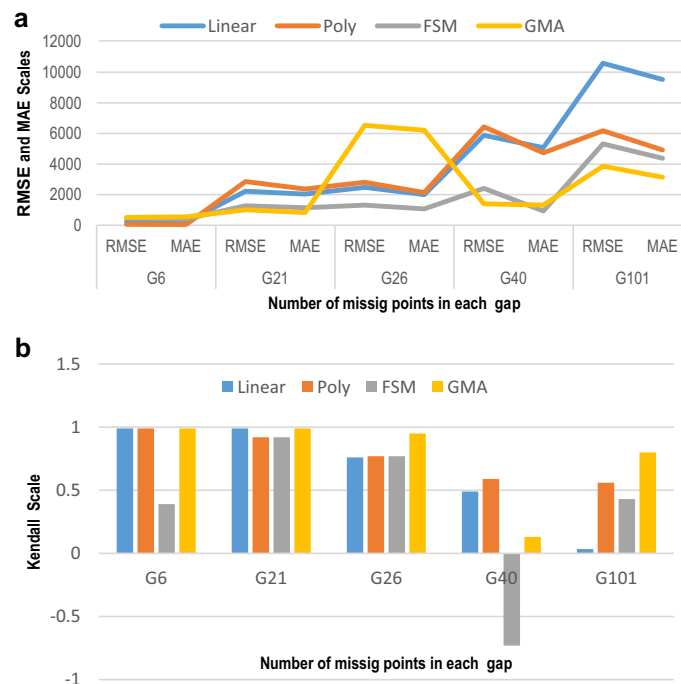
Gap points	G7	G26	G51	G244
Linear	0.86	0.58	-0.4	-0.5
Poly	0.33	-0.25	-0.19	-0.02
FSM	-0.7	0.51	0.26	-0.34
GMA	-0.4	0.6	0.48	0.05

**Table 5** Performance indexes of 4 methods on URC dataset

GP	G6		G26		G33		G40		G101	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Linear	229	209	2222	2034	2476	2007	5877	5070	10,591	9527
Poly	57	50	2857	2372	2811	2139	6433	4738	6175	4921
FSM	474	416	1271	1144	1316	1074	2408	934	5318	4374
GMA	512	555	1015	830	6530	6213	1398	1319	3859	3144

**Table 6** Kendell tua results between imputing gaps and actual data for the URC data set

GP	G6	G26	G33	G40	G101
Linear	0.99	0.99	0.76	0.49	0.035
Poly	0.99	0.92	0.77	0.59	0.56
FSM	0.39	0.92	0.77	-0.73	0.43
GMA	0.99	0.99	0.95	0.13	0.8



**Fig. 4** Performance indexes of 4 methods on URC dataset

**Conclusion**

This paper presents the Gap Imputing Algorithm (GMA), a novel method for imputing missing values in time series data. GMA is specifically designed to address the challenging problem of consecutively missing values with varying gap distances in time series analysis. Initially, GMA identifies sequences of missing values and determines the periodicity of the time series. It then searches for the most similar subsequences in the historical data to fill in the missing gap. GMA employs two methods to impute the missing data gaps, depending on the gap size. If the gap size exceeds the periodic cycle  $p$ , GMA

utilizes the first method, which involves alternating between the two most similar patterns to the missing gap terminals based on their correlation scale. On the other hand, if the missing gap size is less than the periodic cycle, the second method is used. This involves combining the two similar patterns based on their correlation scale with the most similar patterns to fill in the missing data. Experimental results demonstrate that GMA outperforms existing methods in terms of accuracy, particularly for datasets with long periodic patterns and larger missing gaps. Using the periodic cycle to determine the pattern length leads to a more precise and accurate result. In contrast, other algorithms require multiple runs because they rely on random selection of the length of the right and left patterns, which can result in variability in the outcomes.

Overall, this research contributes to the development of more effective and efficient missing value imputation techniques in time series data analysis. The practical implications of these findings are significant, as accurate imputation of missing data is crucial for a wide range of applications.

#### Acknowledgements

The Department of Computer and Control Engineering at Tanta University deserves thanks for providing us with their expertise and valuable advice, which we greatly appreciate.

#### Author contributions

AAK conducted the experiments and written the Python code. NME conceived the study and written the algorithm. MMF participated in the study's design and coordination and provided assistance in drafting the manuscript. All authors read and approved the final manuscript.

#### Funding

This research did not receive any type of grant from funding agencies, either public or private sectors, commercial, or not profit sectors.

#### Availability of data and materials

The developed package and datasets analyzed during the current study are available in the Github repository <https://github.com/Eng-Khattab/missval>.

#### Declarations

##### Competing interests

The authors declare that they have no competing interests.

Received: 1 March 2023 Accepted: 24 April 2023

Published online: 30 August 2023

#### References

1. Zheng Y, Capra L, Wolfson O, Yang H (2014) Urban computing: concepts, methodologies, and applications. *ACM Trans Intell Syst Technol* 38
2. Yi X, Zheng Y, Zhang J, Li T (2015) ST-MVL: filling missing values in geo-sensory time series data. In: *Conference on artificial intelligence*
3. José Cambroneró JK (2017) Query optimization for dynamic imputation. *The VLDB Endowment*, 10
4. Liao W, Bak-Jensen B, Pillai JR, Yang D, Wang Y (2021) Data-driven missing data imputation for wind farms using context encoder. *J Mod Power Syst Clean Energy* 10(4):964–976
5. Little RJ (1992) Regression with missing X's: a review. *J Am Stat Assoc* 87(420):1227–1237
6. Enders CK (2010) *Applied missing data analysis*. Guilford Press, New York
7. Mourad Khayati AL (2020) Mind the gap: an experimental evaluation of imputation of missing values techniques in time series. *VLDB Endowment*, 13
8. Irfan Pratama AE (2016) A review of missing values handling methods on time-series data. In: *International conference on information technology systems and innovation (ICITSI)*
9. Khampuangson T, Wang W (2022) Novel methods for imputing missing values in water level. *Water Resour Manag*. <https://doi.org/10.1007/s11269-022-03408-6>
10. Thi-Thu-Hong PH (2020) Machine learning for univariate time series imputation. Preprint MAPR
11. Paternoster RB (1998) Using the correct statistical test for the equality of regression coefficients. *Criminology* 859–866:36

12. Kulanuwat L et al (2021) Anomaly detection using a sliding window technique and data imputation. *Water* 13(13):1862
13. Yi XZ (2016) ST-MVL: Filling Missing Values in Geo-sensory Time Series Data. In: The 25th International Joint Conference on Artificial Intelligence.
14. Wellenzohn KB (2017) Continuous imputation of missing values in streams of pattern-determining time series. In: The 20th international conference on extending database technology, EDBT
15. Zhang Y (2021) Dual-head sequence-to-sequence model for imputing missing data in multivariate time series. *IEEE J Biomed Health Inform* 25:1692–1702
16. Shu XP (2014) Robust orthonormal subspace learning: efficient recovery of corrupted low-rank matrices. In: IEEE conference on computer vision and pattern recognition, CVPR. Columbus, OH, USA
17. Mazumder RH (2010) Spectral regularization algorithms for learning large incomplete matrices. *J Mach Learn Res* 11
18. Khayati M, Böhlen MH, Mauroux PC (2015) Using lowly correlated time series to recover missing values in time series: a comparison between SVD and CD. In: Advances in spatial and temporal databases. 14th international symposium, SSTD
19. Xu J (2021) FM-GRU: a time series prediction method for water quality based on seq2seq framework. *Water* 13(8):1031
20. Lai WY, Kuok KK (2019) A study on bayesian principal component analysis for addressing missing rainfall water. *Water Resour Manage* 33:2615–2628
21. Dwivedi D (2022) Imputation of contiguous gaps and extremes of subhourly groundwater time series using random forests. *J Mach Learn Model Comput* 3(2)
22. Bokde N (2018) A novel imputation methodology for time series based on pattern sequence forecasting. *Pattern Recogn Lett* 116:88–96
23. Figueroa-García JCP (2022) A genetic algorithm for multivariate missing data imputation. *Inf Sci*
24. Dekermanjian JP, Shaddox E, Nandy D, Ghosh D, Kechris K (2022) Mechanism-aware imputation: a two-step approach in handling missing values in metabolomics. *BMC Bioinf* 23(1):1–17
25. Trubitsyna RS (2022) DEGAIn: generative-adversarial-network-based missing data imputation. *Information* 13(12):575
26. Oppenheim AV (2010) Discrete-time signal processing (3rd ed.). Upper Saddle River, NJ: Pearson Prentice Hall
27. Abdi H (2007) The kendall rank correlation coefficient. *encyclopedia of measurement and statistics*
28. Community TS (2023) Scipy.fft.rfft. (The SciPy community) Retrieved 2023, from <https://docs.scipy.org/doc/scipy/reference/generated/scipy.fft.rfft.html>
29. Rakthanmanon TK (2012) Searching and mining trillions of time series subsequences under dynamic time warping. In: The 18th ACM
30. Mapreduce A, Gu J, Zhang (2016) *J Parallel Distrib Comput* 95: 54–62
31. Park H (2009) A simple and fast algorithm for K-medoids clustering. *Expert Syst Appl* 36:3336–3341
32. Gibbons JD (2011) Nonparametric statistical inference. CRC Press, 14
33. Keogh RW (2021) Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. In: *IEEE transactions on knowledge and data engineering*
34. Stumpy (2023) Steamgen example. (STUMPY) Retrieved 2023, from STUMPY: [https://stumpy.readthedocs.io/en/latest/Tutorial\\_The\\_Matrix\\_Profile.html](https://stumpy.readthedocs.io/en/latest/Tutorial_The_Matrix_Profile.html)
35. Pandas (2023) pandas.DataFrame.interpolate. (pandas) Retrieved from pandas: <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.interpolate.html>
36. Qingkai Kong TS (2020) Python programming and numerical methods—a guide for engineers and scientists. Elsevier
37. Bennett N, Croke B, Guariso G, Guillaume JH, Jakeman A, Marsili-Libelli S, Norton J (2013) Characterising performance of environmental models. *Environ Modell Softw*. <https://doi.org/10.1016/j.envsoft.2012.09.011>

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)

---