Visual Exploration of Air Quality Data with a Time-correlation-partitioning Tree Based on Information Theory

FANGZHOU GUO, Zhejiang University, P.R. China
TIANLONG GU, Guilin University of Electronic Technology, P.R. China
WEI CHEN, FEIRAN WU, and QI WANG, Zhejiang University, P.R. China
LEI SHI, Institute of Software Chinese Academy of Sciences, P.R. China
HUAMIN QU, Hong Kong University of Science and Technology, P.R. China

Discovering the correlations among variables of air quality data is challenging, because the correlation time series are long-lasting, multi-faceted, and information-sparse. In this article, we propose a novel visual representation, called Time-correlation-partitioning (TCP) tree, that compactly characterizes correlations of multiple air quality variables and their evolutions. A TCP tree is generated by partitioning the information-theoretic correlation time series into pieces with respect to the variable hierarchy and temporal variations, and reorganizing these pieces into a hierarchically nested structure. The visual exploration of a TCP tree provides a sparse data traversal of the correlation variations and a situation-aware analysis of correlations among variables. This can help meteorologists understand the correlations among air quality variables better. We demonstrate the efficiency of our approach in a real-world air quality investigation scenario.

CCS Concepts: • **Human-centered computing → Visual analytics**;

Additional Key Words and Phrases: Sensor, multivariate time series, information theory, transfer entropy

ACM Reference format:
Fangzhou Guo, Tianlong Gu, Wei Chen, Feiran Wu, Qi Wang, Lei Shi, and Huamin Qu. 2019. Visual Exploration of Air Quality Data with a Time-correlation-partitioning Tree Based on Information Theory. *ACM Trans. Interact. Intell. Syst.* 9, 1, Article 4 (February 2019), 23 pages.
https://doi.org/10.1145/3182187

This work is supported by National 973 Program of China (Grant No. 2015CB352503) and National Natural Science Foundation of China (Grants No. 61772456, No. U1609217, and No. 617611360202).

Authors' addresses: F. Guo, State Key Lab of CAD&CG, Zhejiang University, 866 Yuhangtang Road, Hangzhou, Zhejiang, P.R. China; email: guofangzhou@zju.edu.cn; T. Gu (Corresponding Author), Guilin University of Electronic Technology, P.R. China; email: cctlgu@guet.edu.cn; W. Chen (Corresponding Author), State Key Lab of CAD&CG, Zhejiang University, 866 Yuhangtang Road, Hangzhou, Zhejiang, P.R. China; email: chenwei@cad.zju.edu.cn; F. Wu, Huawei Technologies Co., Ltd., Hangzhou, Zhejiang, P.R. China; email: wfr007@163.com; Q. Wang, State Key Lab of CAD&CG, Zhejiang University, 866 Yuhangtang Rd, Hangzhou, Zhejiang, P.R. China; email: qiwang_cad@zju.edu.cn; L. Shi, Institute of Software Chinese Academy of Sciences, Beijing, P.R. China; email: shil@ios.ac.cn; H. Qu, Hong Kong University of Science and Technology, Hong Kong, P.R. China; email: huamin@cse.ust.hk.

2160-6455/2019/02-ART4 $15.00
https://doi.org/10.1145/3182187

1 INTRODUCTION

The rapid growth of industrial economy and oil-fueled vehicles has dramatically increased the global air pollution all over the world. According to WHO, ambient air pollution contributes to 6.7% of all deaths.[1] Due to this strong tie between air quality and health [29], the air quality problem has attracted growing attentions. For many years, meteorologists have been analyzing the air pollutants (such as oxynitride and particulate) together with weather variables (such as temperature and relative humidity), which are monitored in modern cities, to understand the dynamics of air pollutants.

At the heart of fighting global air pollutions, analyzing air quality data requires interdisciplinary knowledge and techniques to exploit of the time-oriented, multivariate nature of this data, and to enhance situation awareness for domain users. In this practice, data visualization techniques incorporated with clustering, dimension reduction, and data simplification analysis can be important to provide a clear view of multiple air quality variables and their evolutions [10, 19]. Previous studies have made significant progress on monitoring, analyzing, and forecasting the air quality and weather conditions. In visualization, the weather data and air pollutants are often displayed on a map monitoring the air quality in certain area [36, 48]. Regression analysis [11], statistical analysis [29], and correlation analysis [24] are often used to analyze the patterns in air quality data. As for forecasting, most studies concentrated on the visualization of predictive models and the ensembling of data for more precise forecast result [25]. The correlation analysis between air pollutant and weather has been focusing on static times, e.g., year by year, without considering the evolution of the correlations along time [24]. In particular, little attention has been paid to the mutual and dynamic influences of multi-faceted variables.

This article aims at the temporal correlation analysis of air pollutants and weather variables collected from multiple sensors. Importantly, we introduce both symmetric and asymmetric information-theoretic measures to capture the correlations among variables. While most correlation visualization and analysis techniques [5, 43] can be applied to our scenario to interpret multiple sensor data, the resulting correlations displayed are temporally long-lasting, dynamically changing, multi-faceted, and information-sparse, making the task of interactive exploration time-consuming. We note that the temporal coherence is frequently used in the correlation analysis of time series. Using this coherence appropriately, we can effectively abstract the time series to support efficient sparse data traversal. Meanwhile, physically meaningful correlations only exist in a limited set of variable tuples. Exploiting this sparsity can greatly reduce the analysis overhead. Conventional time-partition [14] or variable-based graph structures [2, 24] have been proven to be effective in characterizing the coherence, trend, and similarity in terms of time or variable. However, treating the temporal and multi-faceted variations equally may prevent the possibility of detecting interesting correlations of a specific variable pair in a small time interval.

Based on the above observations, we proposed a novel hierarchical data structure named time-correlation-partition tree (TCP tree) and embedded this novel structure in a visualization system. The TCP tree is capable of capturing the sparsity in both the temporal domain and the variable domain. Integrating two domains into a single tree structure enables users to explore the correlations in different level of details and analyze the temporal patterns of the correlations within a consistent visualization. Furthermore, we develop an interactive prototype visualization system (see in Figure 1), which allows dynamic tree construction and on-demand visual exploration of local correlation time series based on the TCP tree structure and empowers users with a capability of locating and identifying interesting correlation patterns in a context-aware fashion. In summary, the contribution of this article can be summarized as:

[1] http://www.who.int/gho/phe/outdoor_air_pollution/burden/en/.

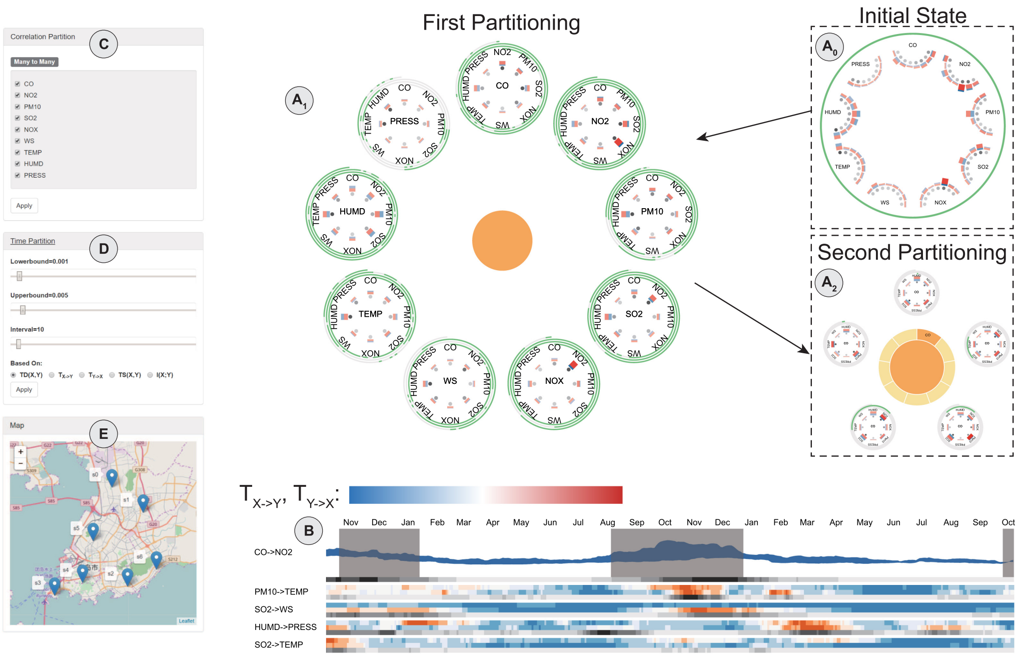


Fig. 1. Information-theoretic visualization of the air quality data with our TCP tree structure. A0, A1, and A2 are three sequential states of a time-correlation-partition (TCP) tree view. A0 is the initial state of the TCP tree and represents the aggregated correlations among all variables along the entire time axis. A1 is the state after applying a correlation partition on A0, and A2 is the state after applying a time partition on the node CO in A1. B denotes a hybrid visualization of pixel map and line chart and shows the details of variable correlations. C and D are the parameter panels for variable-oriented and temporal partitions, respectively. E is a map that shows the spatial distribution of the sensors.

- A novel hierarchical data structure, TCP tree, which organizes the correlations among air quality variables in both temporal and variable domain. The TCP tree intuitively enables users to explore and analyze the evolution of correlations among sets of air quality variables.
- An interactive visualization system illustrating the TCP tree and a set of novel visual designs enable users to interactively construct TCP tree and explore the correlations.

The rest of this article is structured as follows. Section 2 summarizes the related work. We describe the analytical tasks and design goals in Section 3. Section 4 explains how the information-theoretic correlations are calculated. Section 5 introduces the structure and construction of the TCP tree. The visual design is elaborated in Section 6. Section 7 summarizes the interactions in the system, and the case study is introduced in Section 8. Finally, we conclude this article in Section 9.

2 RELATED WORK

2.1 Air Quality Analysis

Analysis of multivariate air quality data turns out to be a prolonged scientific battle involving analysts from diverse academic domains. Qu et al. [24] integrated a suite of novel visualizations into their comprehensive system, including circular bar charts and weighted complete graphs, in support of the analysis of the air pollution problem in Hong Kong. Although they take into account the key role played by wind direction and speed in weather data visualization, the lack of corresponding geographical information maintains a fatal shortcoming of their research work. Zheng et al. [49], however, employed a co-training-based semi-supervised learning approach to improve the air quality inference accuracy. Both spatially and temporally related features are identified in their approach.

Air quality monitoring is of great assistance to analysts during the air quality analysis process. Völgyesi et al. [36] proposed SensorMap, an overall air quality monitoring system based on car-mounted sensor data, to gain a detailed picture of the air quality in a large area at a low cost. Unfortunately, their work leans more toward hardware platform development rather than visualization and analysis. Another essential task of air quality analysis lies in prediction. Different from works that only focus on measuring temporal correlations of weather variables and air pollution [3, 17], Demuzere et al. [11] took it a step further and extended their method to an alternative air quality prediction tool on the basis of similar correlation investigation. WeaVER [25] presents a series of practical encoding choices to interpret multiple weather features as well as their interactions, which benefits weather forecasts in an intuitive way. However, no formal evaluations are provided to prove the validity of their visualization designs. In this article, we construct a TCP Tree structure tailored for air quality analysis that characterizes correlations of multiple variables and their evolutions based on information entropy measurements.

2.2 Information Theory in Visualization

Information theory has recently attracted much attention in the visualization field [6–8, 41]. Using information theory in data analysis and visualization can help build connections between data communications and data analysis and visualization [41]. Theoretically, the stages of a visualization process can be interpreted using the taxonomy of information theory [8].

Generally, the information entropy is employed to measure information quantitatively. It is quite useful for locating important regions and improving the analysis and visualization efficiency, e.g., placing seeds for streamline generation [46]. One example is the view selection that can be optimized by measuring the information entropy associated with different views [4, 18, 32]. The quality of the LOD view [40] can also be evaluated by using the information entropy. Similarly, the importance-driven focus of attention [39] can be captured by building an information channel between objects and viewpoints.

An important usage of the information theory is to measure the correlation between two variables. The symmetrical mutual information can be used to evaluate the similarity between isosurfaces [5]. Likewise, the relative information between multi-modal [15] or time-varying datasets [43] is essential to achieve importance-driven visualization. The recently developed transfer entropy [28] can be used to characterize the asymmetrical correlations between two time series and has proven to be effective for volume visualization [42], neuroscience [38], and social media analysis [37]. More recently, an information-aware framework was introduced to explore multivariate datasets [2]. Our approach advances the scheme by exploiting the temporal variations of a large-scale multi-variate time series.

2.3 Time-series Data Structuring and Visualization

Visualizing and structuring time series has been a classical research topic. The standard visualization for linear time dimension would be a two-dimensional plot: one axis for time, the other for data value. Weber et al. [45] designed a spiral-shaped time axis where careful selection of cycle length could reveal the cyclic pattern of the data. If the time dimension refers to date, then a calendar view [35] can be adopted to visualize the value changes in different days. Tominski et al. [33] employed parallel coordinates to represent time series.

Structuring time is an important scheme to capture the semantic evolution along the timeline. For instance, the time line structure [23] is commonly used to represent events, activities, or even status. Storyline [22] and ThemeRiver [16] can be used to represent the evolution of multiple-variables. We employ a ThemeRiver-like structure to display the computed correlation time series. Other data structures, like trees and graphs, can also be used to depict the time-oriented evolution structure. For instance, a tree structure is automatically generated to incorporate animations into time-varying data for illustrative narration. The event graph [26] is widely applied to capture the connections among different time pieces. Likewise, a TransGraph [13] was designed to organize a time-varying volume data set into a hierarchy of states and visualize the resulting transition relationships. A pioneering work similar to ours is the time-space-partitioning (TSP) tree that reformulates a time-varying volume dataset to a nested tree structure for the purpose of sparse data traversal and rendering acceleration. To the best of our knowledge, the proposed data structure is the first to characterize the space of time, variable, and correlation in an information-theoretic way.

3 ANALYTICAL TASKS & DESIGN GOALS

In this section, we first describe the features of the air quality data, thereafter, we introduce the analytical tasks of analyzing the evolution of correlations among air quality variables, and then we summarize the design goals of the system to fulfill these tasks.

3.1 The Time-correlation-variable Space

Generally, air quality data is a set of time series of weather variables and air pollutants monitored by multiple sensors. The complete set of symmetrical and asymmetrical time-varying correlations among these time series spans a triple space, called the *time-correlation-variable* (TCV) space, whose three dimensions are time, variable, and correlation, respectively (see Figure 2(c)).

In particular, variables are the basic units, and can be grouped from multiple perspectives, namely, locations, sensor types, and variable types. We denote the hierarchical organization of variables in terms of locations, sensor types, and categories as the *variable space*. This actually organizes the TCV space along the variable dimension, as illustrated in Figure 2(c). Meanwhile, along the time dimension a time series can be recursively subdivided into a hierarchical *time space* by either exploiting their temporal coherence or using a uniform subdivision scheme.

3.2 Analytical Tasks

We first summarize the analytical tasks of analyzing the correlations among different air quality variables and time steps.

- **Identify the time slices when the correlations among air quality variables are significant.** The correlations of air quality variables are time-varying; therefore, there are time slices when the correlations are weak and strong. Filtering the time slices according to the correlation strength and identifying when variables have significant correlation is the basic task when analyzing temporal correlations among variables.
- **Discover the periodic patterns of the correlations of air quality variables.** The correlations of air quality variables may change periodically along time. Discovering these periodic patterns helps meteorologists understand the evolution pattern of the air quality.
- **Discover the transition patterns of the asymmetrical correlations among air quality variables.** For asymmetrical correlations, transition patterns (i.e., X affects Y and Y affects Z) may exists among the variables. Discovering these patterns helps meteorologists explore the order of importance of air variables.

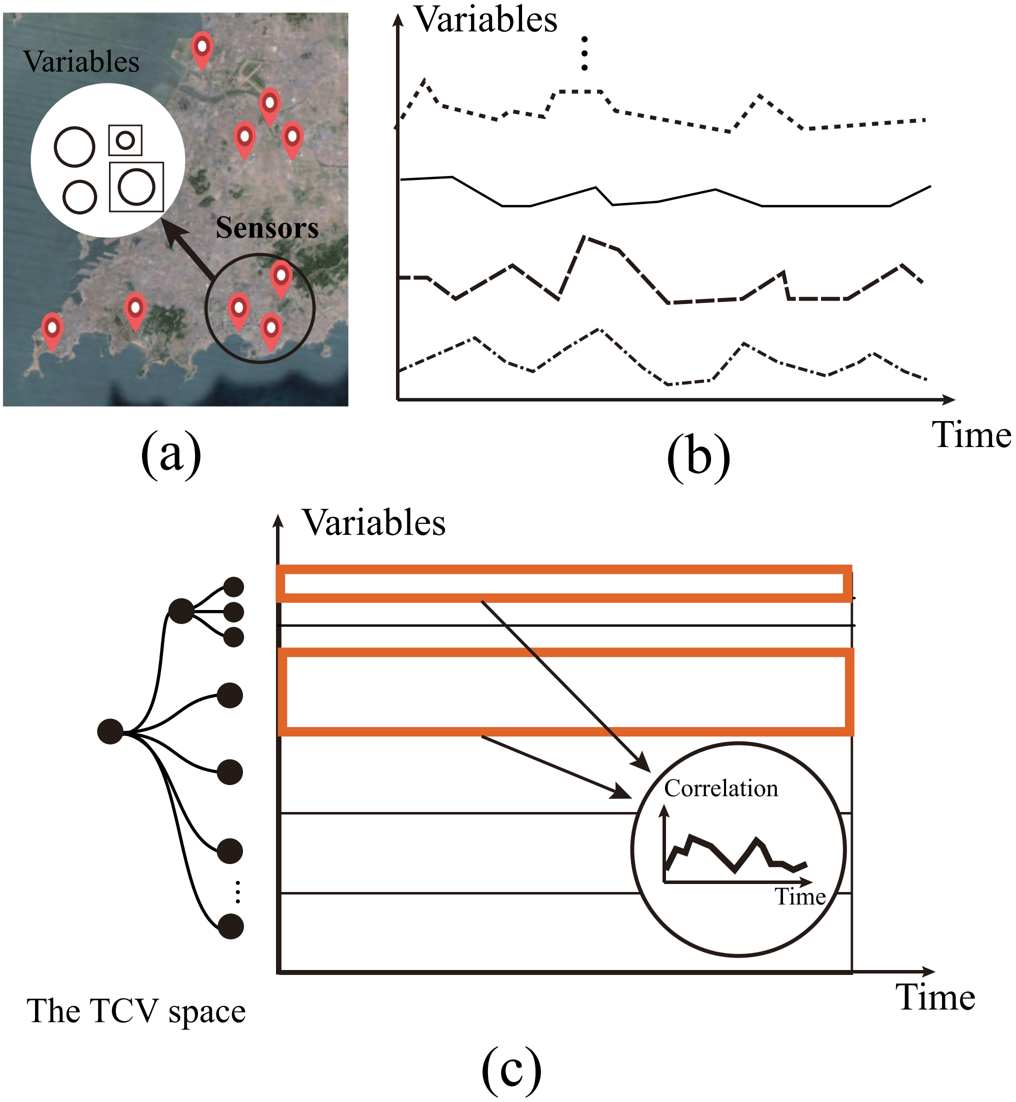


Fig. 2. (a) Sensors and variables; (b) the input dataset, e.g., time series of sensor readings; (c) the time-correlation-variable space converted from the input dataset.

3.3 Design Goals

Due to the complexity of the data space of the correlation among air quality variables, directly visualize the correlations is not capable for fulfilling the analytical tasks. Therefore, we designed a novel data structure, named time-correlation-partitioning (TCP) tree, to organize the data. To assist users to accomplish all the analytical tasks, we summarized the design goals of the visualization of TCP tree, which includes three aspects:

- **Hierarchy.** While it is easy to present the TCV space as a large pixel-based map or stream-graph, depicting the entire dataset with a hierarchically abstracted bundle of informative pieces makes the understanding and exploration of the dataset more efficient.

- **Informativeness.** The key of the visualization is to provide a clear view of the evolution and variations between correlations of multiple variables. Two representative entropy measures (symmetrical or asymmetrical) are encoded.
- **Completeness.** The visualization should be self-contained, i.e., present all relevant information, including the overall structure of the TCV space, the detailed correlation time series of all variable pairs and their aggregations, as well as the temporal coherence and variations.

4 INFORMATION-THEORETIC CORRELATIONS

We employ the concept of information entropy to represent the correlations between two time-varying sequences within a specific time interval. Below, we first briefly describe two types of information entropy measures.

For two time series $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_n)$, $1 \leq n \leq m, n, m \in \mathbb{N}$, the *mutual information* I(X;Y) is defined as

$$I(X;Y) = I(Y;X) = H(X) + H(Y) - H(X,Y). \tag{1}$$

Here, $H(\cdot)$ denotes the entropy of a time series. Equation (1) explains the reduction in the uncertainty of X due to the knowledge of Y [9] and vice versa. Note that the mutual information between two time series is measured along an identical timeline.

If the time delay of the information transfer is taken into account, or say, consider the information transfer from the time series X to the time series Y in a time interval, then the *transfer entropy* [28] from X to Y can be defined as

$$\begin{aligned} T_{X\to Y} &= \sum_{1\leq n\leq m} p\left(y_{n+1}, y_n^{(l)}, x_n^{(k)}\right) \log \frac{p\left(y_{n+1}|y_n^{(l)}, x_n^{(k)}\right)}{p\left(y_{n+1}|y_n^{(l)}\right)}, \\ p\left(y_{n+1}|y_n^{(l)}, x_n^{(k)}\right) &= \frac{p\left(y_{n+1}, y_n^{(l)}, x_n^{(k)}\right)}{p\left(y_n^{(l)}, x_n^{(k)}\right)}, \\ p\left(y_{n+1}|y_n^{(l)}\right) &= \frac{p\left(y_{n+1}, y_n^{(l)}\right)}{p\left(y_n^{(l)}\right)}, \end{aligned} \tag{2}$$

where $x_n^{(k)} = (x_n, \ldots x_{n-k+1})$ and $y_n^{(l)} = (y_n, \ldots y_{n-l+1})$ denote the past states of X and Y with two Markov processes of order k and order l, and $p(\cdot)$ denotes the proportion of a specific sequence in X and Y. Note that k and l are two adjustable constants. Please refer to Reference [28] for more details.

In principle, the transfer entropy explains the reduction of uncertainty in Y due to the past states of X. $(T_{X\to Y} - T_{Y\to X})$ indicates the dominant strength of influence from X to Y. Thus, we can judge that X influences Y if it is larger than zero and vice versa. We denote $TD(X,Y) = (T_{X\to Y} - T_{Y\to X})$ as the *transfer entropy difference* and $TS(X,Y) = (T_{X\to Y} + T_{Y\to X})$ as the *transfer entropy summation*. When $TD(X,Y) > 0$, we say X affects Y, and when $TD(X,Y) < 0$, we say Y affects X.

Essentially, $I(X;Y)$ represents a symmetrical correlation, while $TD(X,Y)$ encodes an asymmetrical correlation. Each measure computes a numerical value for two time series whose time ranges are supposed to be limited. For long-term time series, we sample the entire time range with a sequence of shifted windows where the shift and width are denoted by Δt and w, respectively. We compute the *correlation time series* with respect to the window sequence.

Temporal aggregation of correlations calculated by aggregation operations, such as sum, average, median, peak, and valley, has three forms, including **one-to-one**, **one-to-many**, and **many-to-many** correlations. A one-to-one correlation is the correlation time series between a

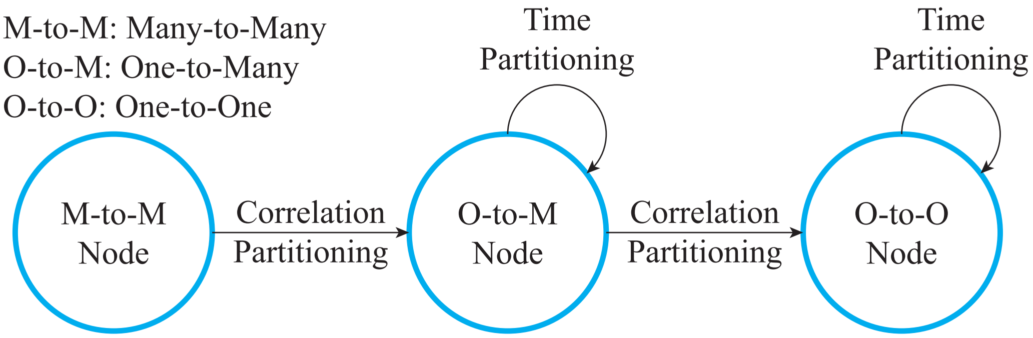


Fig. 3. The state diagram of node types and partitioning operations.

pair of variables, which can be summarized into a single value by applying an aggregation operation. A one-to-many correlation is a vector formed by the one-to-one correlations between a variable and a set of variables, and shows the summarized correlation of one specific variable. A many-to-many correlation is a matrix formed by the one-to-one correlations between two sets of variables and overviews the correlations among all variables.

5 TIME-CORRELATION-PARTITIONING TREE

In this section, we first introduce the structure of TCP tree and how it organizes the correlation data, and then we introduce the two partition operations for constructing the TCP tree.

The input air quality data can be denoted as $A = \{A_1, A_2, \ldots, A_k\}$, where A_i is a variable in the data and $A_i = \{A_i^0, A_i^1, \ldots, A_i^t\}$, where t is the number of timestamps. Then the correlation among variables in the data can be denoted as a three-dimensional matrix C, whose entry is a 3-tuple $C_{i,j,k} = \{T_{A_i^{(k,k+w)} \rightarrow A_j^{(k,k+w)}}, T_{A_j^{(k,k+w)} \rightarrow A_i^{(k,k+w)}}, I(A_i^{(k,k+w)}, A_j^{(k,k+w)})\}$, where w is the window length, which is 28 days in this article. The TCP tree organizes this matrix in a hierarchical structure to help users explore correlations of multivariate time series.

5.1 The Tree Structure

The TCP tree is designed to characterize the time-varying correlations hierarchically in both the variable domain and the temporal domain. In each tree node, its associated correlation time series and temporally aggregated correlations are recorded. In particular, there are three types of tree nodes in the tree, including many-to-many nodes, one-to-many nodes, and one-to-one nodes, corresponding to the three forms of the temporally aggregated correlations. A many-to-many node corresponds to the correlation matrix C. A one-to-many nodes corresponds to a row of the correlation matrix C. A one-to-one node corresponds to an entry of the correlation matrix C. Each node in the tree stores correlations in a specific period of time. Initially, the root node stores correlations in the complete time period. Time-partitioning operation divides the temporal domain of one-to-many nodes and one-to-one nodes and generates child nodes each of which stores correlations in a segment of the parent temporal domain. The state diagram of the node type and partitioning operations is shown in Figure 3.

The initial state of a TCP tree, which is also the root of a TCP tree, represents the aggregated correlations among all variables along the whole time axis and thus it is a many-to-many node, showing an overview of the correlations. The root node is partitioned into multiple sub-nodes iteratively by a sequence of time-partitioning and correlation-partitioning operations. After each partitioning operation is applied, new nodes are generated and appended to the partitioned node as

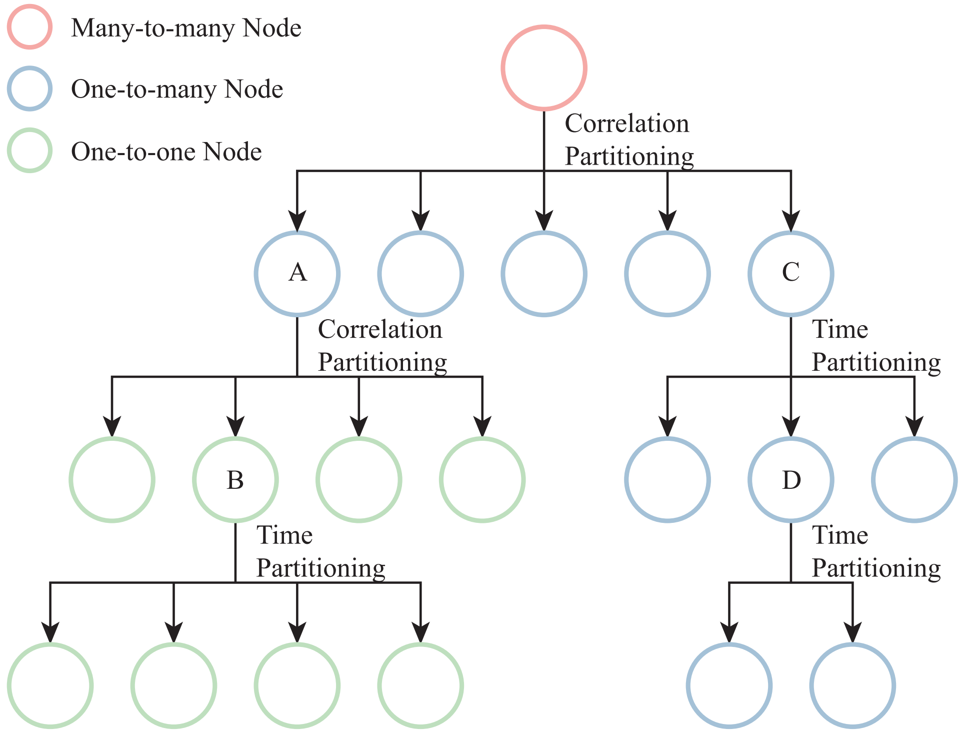


Fig. 4. A TCP tree after five partitioning operations.

its children. Figure 4 shows a TCP tree after several partitioning operations. Initially, a correlation-partitioning operation is applied on the root node and five one-to-many nodes are generated. Then a correlation-partitioning operation is applied on the node A, and four one-to-one nodes are generated. By applying a time-partitioning operation on node B, four one-to-one nodes are generated, each of which indicates the correlation is significant in a specific time period. However, a time-partitioning operation is applied on node C, and three one-to-many nodes are generated. After adjusting the parameters of time-partitioning operation, another time-partitioning is applied on node D, and two one-to-many nodes are generated.

The order of the correlation-partitioning follows the variable hierarchy. The variable hierarchy can be built upon natural hierarchy of the underlying dataset, including:

- The variable types that are relevant to the monitored objects (e.g., PM10 and PM2.5 belong to Particle Matter series) can be used to group variables or sensors;
- The relations among sensors (e.g., the sensor network) and the dependency of variables to sensors, can be used to group variables;
- Environment-related factors and the spatial locations of sensors can be used to categorize variables associated with sensors;
- A group of sensors, or a group of variables can be hierarchically organized based on the domain experience or analysis tasks.

In the following sections, we elaborate the correlation-partitioning and the time-partitioning operations.

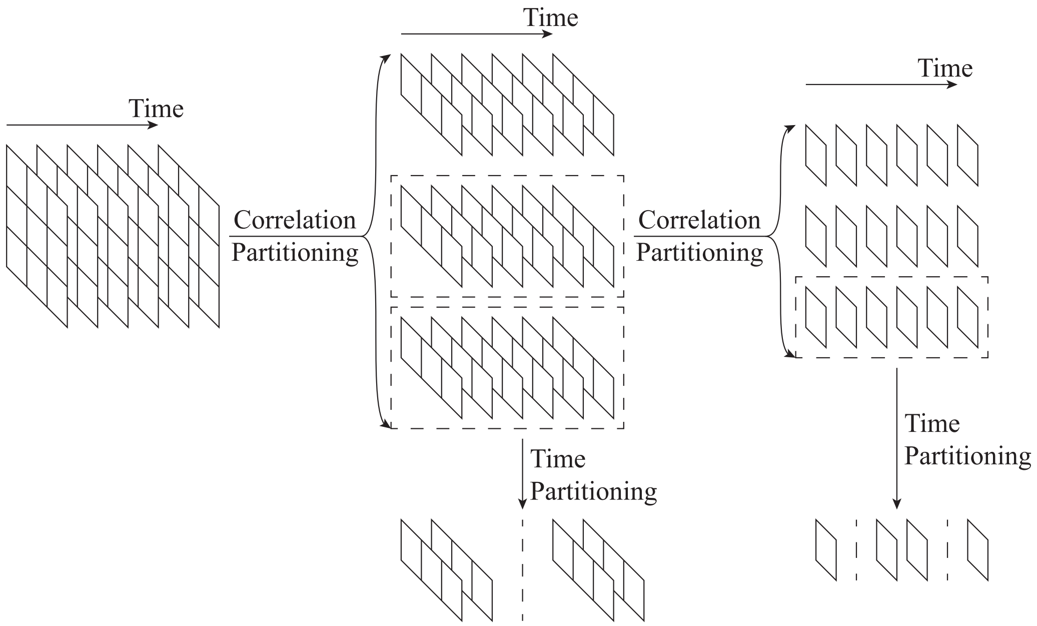


Fig. 5. A schematic diagram shows how correlation data is partitioned by correlation-partitioning and time-partitioning operations.

5.2 Correlation Partition and Time Partition

The construction of the TCP tree is a dynamic procedure. Two different partition operations are applied to tree nodes iteratively, including correlation partition and time partition. The correlation partition is capable on many-to-many nodes and one-to-many nodes, and the time partition is capable on one-to-many nodes and one-to-one nodes. Figure 5 shows how data is partitioned by the two operations.

Correlation partition is based on the hierarchical structure of the data. Without time-partition operations, a many-to-many node is partitioned to a set of one-to-many nodes and then a one-to-many nodes is partitioned to a set of one-to-one nodes. In this way, users are enabled to concentrate on different correlation sets with correlation partitions. The reason why we support users to partition and explore the TCP tree hierarchically is that the results of time partitions on one-to-many nodes and one-to-one nodes are different.

Time partition is applied on one-to-many nodes and one-to-one nodes in variable tree. In a TCP tree, each tree node is attached with a correlation time series. By time partition, time pieces with significant relevance are extracted, each of which is appended to a new tree node. A threshold-based method is designed to obtain the user-interested partitions, e.g., the partitions have high $TD(X, Y)$ and so on, as shown in Figure 6. For one-to-one nodes, the method is directed applied. For one-to-many nodes, one critical problem is the conflict process of different partitions of all involved time series, because each one-to-one correlation time series can have an independent partition scheme. We solve this problem with a two-stage process. In the first stage, the threshold-based method is applied to all correlation time series in the node. In the second stage, we construct a joint partition by leveraging the obtained partitions in the first stage, and then we build a time tree based on the partition. The detail of the time partition method is shown in Algorithm 1.

The TCP tree organizes correlations among variables in multivariate time series in a hierarchical structure. In the analysis of such correlations, users might be interested in the correlations

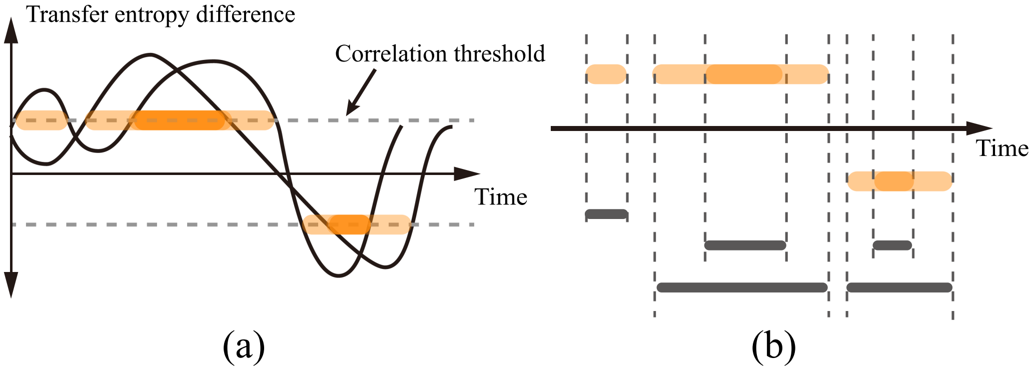


Fig. 6. Adaptive partition demonstrated with two asymmetrical correlation time series. (a) Filtering the time series with a high correlation threshold. (b) Constructing a joint partition by different operations.

among different combinations of variables. Besides, the correlation between two variables is dynamically changing along time. The TCP tree structure helps users to find desired correlations between variables in a level-of-detail scheme and find time periods when correlations are significant by providing an interactive construction and exploration process with two partitioning operations.

6 VISUALIZATION OF A TCP TREE

A TCP tree is an information-theoretic, compact, and hierarchical characterization of the correlations among air quality variables.

6.1 The Tree Structure

The tree structure is dynamically built and modified by a series of user-defined-partitioning operations. The initial design of the visualization of the tree structure is a single-node-link diagram with the layout shown in Figure 7. However, there are two major problems of this design. First, it lacks space efficiency. Second, it cannot offer flexible navigation after multiple partition operations.

Thus, we use a more compact design that combines a sunburst diagram and a node-link diagram to solve the two problems (see in Figure 8). Initially, a single node is used to represent the entire set of the correlations, which contains many-to-many correlations. After partition is applied, the partition result is represented by a group of nodes surrounding a sunburst diagram (see in Figure 8(A)), which represents the tree structure before partitioning. Nodes generated by the partition surround the sunburst diagram and segments on the helix in a node are preview of the time partition result on the node, as shown in Figure 8(B).

Because each tree node corresponds to a set of or a single correlation time series, it is necessary to label the time range. A variable tree node extends the same time range as its parent node, however, a time tree node only contains one of the segments partitioned from the its parent node. A helix outside each node is used to show the time range of the time series. Segments are added to the helix to give a preview of potential result of time partition on the node, as shown in Figure 8.

In each tree node, no matter it is a variable tree node or time tree node, it supports users to freely modify the variables in it. For example, in a many-to-many variable tree node, users can remove PRESS and WS to explore the correlations among the remaining variables. The modification of variables will effect the result of following partition operations as the set of variables is consistent between parent node and leaf nodes.

ALGORITHM 1: Time-partitioning Method

```
Input: Correlation time series $Cor$, threshold $\theta$, partition mode $mode$
Output: The partition result $partition$
$timeSegmentArray = [], i = 0;$
for each time series $c$ in $Cor$ do
    $timeSegments = [];$
    for each correlation triple $v$ at time stamp $t$ in $c$ do
        $T_{a\to b}, T_{b\to a}, I = v;$
        $TD = T_{a\to b} - T_{b\to a}, TS = T_{a\to b} + T_{b\to a};$
        $baseValue = base(T_{a\to b}, T_{b\to a}, I, TD, TS, mode);$
        if $|baseValue| > \theta$ then
            $timeSegments[t] = True;$
        else
            $timeSegments[t] = False;$
        end
    end
    $timeSegmentArray[i] = timeSegments, i++;$
end
$partition = [], t = 0, segment = [], index = 0, state = -1, sIndex = 0;$
repeat
    if $state == -1$ then
        for each $ele$ in $timeSegmentArray$ do
            if $ele[t]$ then
                $state = 0, segment[sIndex] = t, sIndex++;$
                $break;$
            end
        end
    end
    if $state == 0$ then
        $flag = False;$
        for each $ele$ in $timeSegmentArray$ do
            $flag| = ele[t];$
        end
        if $flag$ then
            $segment[sIndex] = t, sIndex++;$
        else
            $partition[index] = segment, index++;$
            $segment = [], sIndex = 0;$
            $state = -1;$
        end
    end
    $t+ = 1;$
until $t > timestamps;$
```

6.2 Aggregated Correlations

To increase the readability of the tree nodes of TCPTree, aggregated correlations are visualized inside each node. The first design of the aggregated correlations inside the tree node is shown in Figure 9. However, we identify two major limitations in this design. First, the length of the links

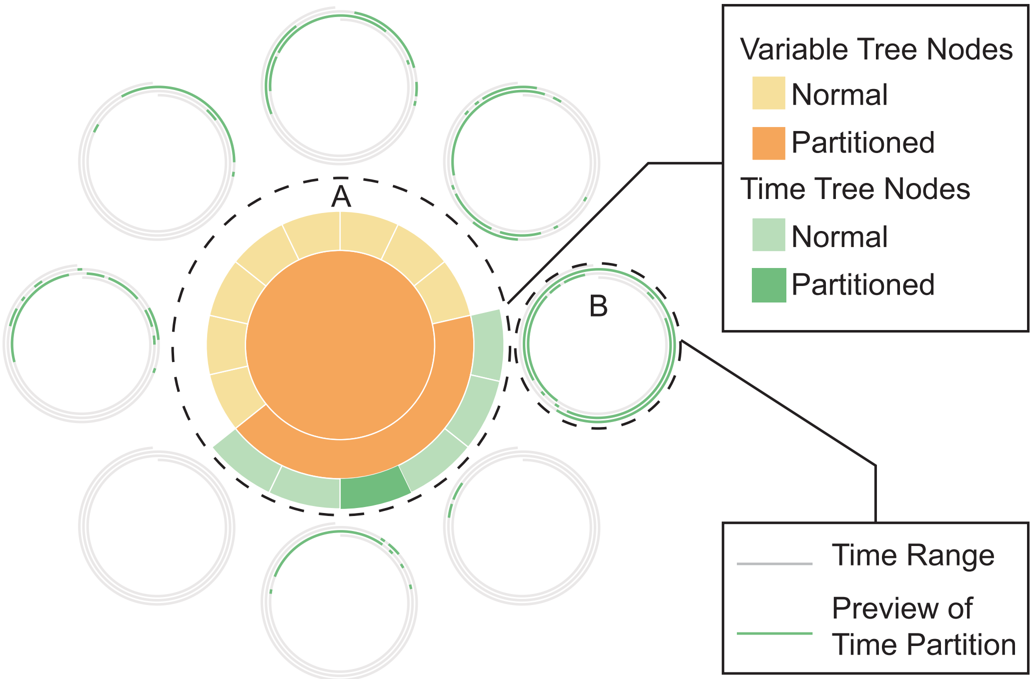


Fig. 8. A TCP tree consists of two parts. Part A is the structure of the tree, which is formed by the tree nodes generated by partitioning before the latest partitioning (green nodes are time tree nodes and yellow nodes are variable tree nodes). Part B is the tree nodes generated by the latest partitioning; the green segments on the helix are the preview of the time-partitioning.

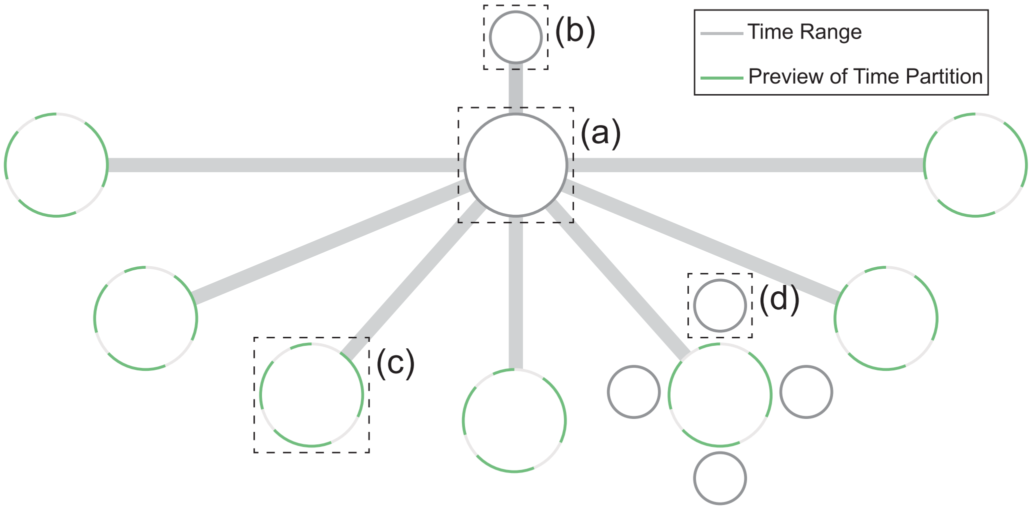


Fig. 7. The first visual design of the tree structure. (a) The latest partitioned node; (b) the ancestor of the latest partitioned node; (c) newly generated node by partitioning with a preview of time-partitioning operation; and (d) the nodes generated by time-partitioning operation.

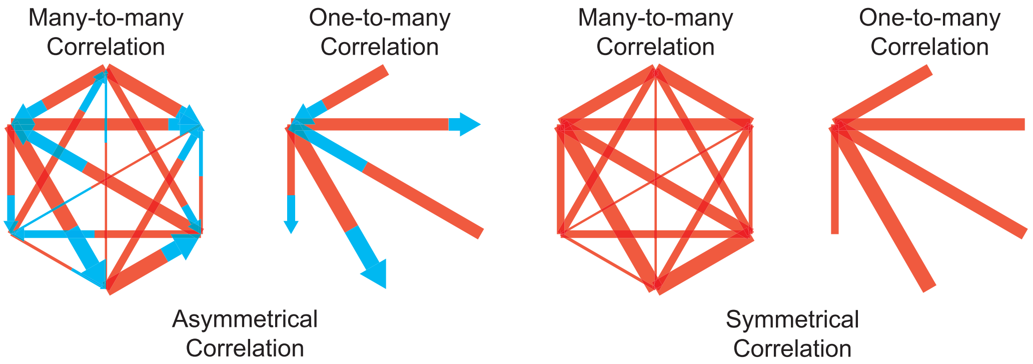


Fig. 9. The original design of aggregated correlations. Asymmetrical correlations are represented by directed links and symmetrical correlations are represented by undirected links.

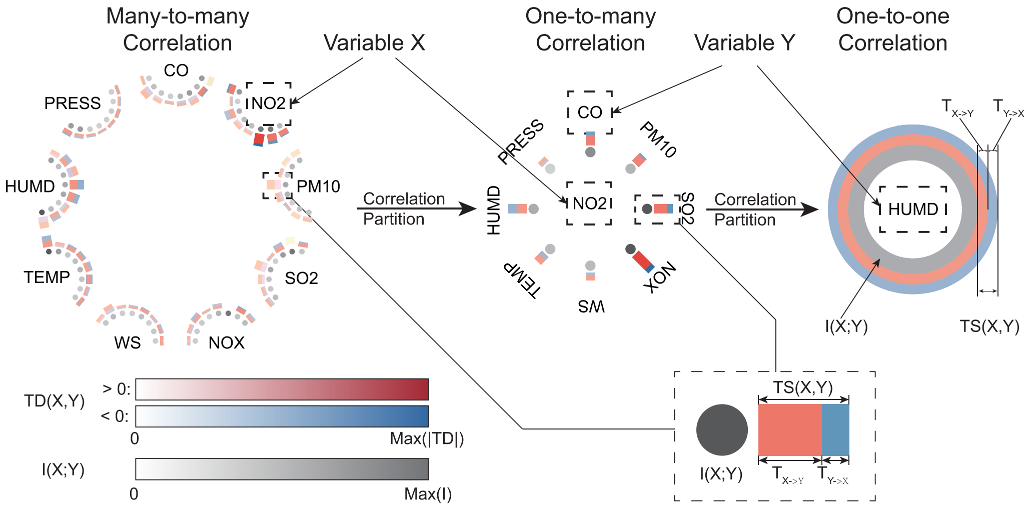


Fig. 10. The visual design of three types of correlations. For many-to-many nodes and one-to-many nodes, we use a glyph design to encode $TD(X,Y)$, $TS(X,Y)$, $T_{X\to Y}$, and $T_{Y\to X}$. For one-to-one nodes, we use three circles to encode these values.

interfere with users' cognition of the strength of the relations, because of the varying lengths of lines without any information encoded. Second, users' have to switch between the asymmetrical correlation and symmetrical correlation repeatedly for comparison.

Therefore, we improve the design by using different layouts with the same glyph design and color encodings to visualize the three types of nodes, as shown in Figure 10. The time partition results of one-to-many node and one-to-one node are different, it is the reason why we still remain the one-to-one nodes in the TCP tree node.

- **Many-to-many correlations.** The basic visual scheme of the temporally aggregated many-to-many correlations is a radial node-link graph. It is an overview of correlations among variables in the TCP tree. Each attribute is a node and all the nodes are uniformly distributed on a circle. To avoid the visual clutter, the correlations among attributes are represented by small rectangles distributed around the node, as shown in Figure 10. For example, the aggregated correlations among CO and other attributes are encoded by a group of small glyphs (see in Figure 10). Each glyph is formed by a circle and a rectangle. The color of the circle encodes $I(X;Y)$; and the length of the rectangle encodes $TS(X,Y)$. The rectangle is divided into two sub-rectangles, "X-to-Y" sub-rectangle, which is closer to the circle, and "Y-to-X" sub-rectangle. The length of the "X-to-Y" sub-rectangle encodes $T_{X\to Y}$ and the length of the "Y-to-X" sub-rectangle encodes $T_{Y\to X}$. The color of two sub-rectangles encodes TD. Specifically, the color of the "X-to-Y" sub-rectangle encodes $TD(X,Y)$ and the color of the "Y-to-X" sub-rectangle encodes $TD(Y,X)$. If $TD \geq 0$, then the color mapping is from $[0, max(|TD|)]$ to $[white, red]$. If $TD < 0$, then the color mapping is from $[0, max(|TD|)]$ to $[white, blue]$. The sequential color schemes of "blues" and "reds" in COLORBREWER [1] are used. Note that the maximum values of I and TD are calculated based on the data contained in the tree node instead of the global data.

- **One-to-many correlations.** For one-to-many correlations, say the correlations among variable X and a set of variables S, a radial layout is used: X is placed on the center and variables in S are placed around X. The aggregated correlations are represented by rectangles the same as the many-to-many correlations, as shown in Figure 10.
- **One-to-one correlations.** The visual encoding of one-to-one correlations is very simple, as shown in Figure 10. For a pair of variables (X,Y), the name of variable Y is placed in the center. Three concentric circles from the inside out represent $I(X;Y)$, $T_{X\to Y}$, $T_{Y\to X}$, respectively, with the color mapping shown in Figure 10.

As the data organized by TCP tree structure is a three-dimension matrix, it is natural to use a matrix representation to visualize the correlations. However, there are two major disadvantages of the matrix representation for visualizing correlation data in this article. First, symmetric and asymmetric correlations are both considered, thus each entry of the matrix is a triple, which cannot be visualized by a simple matrix. Second, current visual encodings can help users directly observe $T_{x\to y}, T_{y\to x}, TD(x,y), TS(x,y)$, and $I(x,y)$ without interactions, while it is not easy to encode this information in a simple matrix representation.

6.3 Details of Correlation Time Series

In tree view, the hierarchy and aggregated correlations are visualized; however, the details of the correlation time series are still missing. For completeness, the details are visualized by colored 2D *pixel maps* and modified line charts, as shown in Figure 1(b). When using 2D pixel maps, each asymmetrical correlation time series ($T_{X\to Y}$ and $T_{Y\to X}$) is represented by two rows of pixelbars, and each symmetrical correlation time series is represented by a single line of grey pixelbars. The modified line chart is used to show the $TD(X,Y)$. After the pixelmap of the asymmetrical correlations between X and Y is expanded, the modified line chart is shown.

When using modified line charts, $T_{X\to Y}$ and $T_{Y\to X}$ are represented by two polylines and $TD(X,Y)$ is emphasized by the colored region (the region between the two polylines). Users can freely switch between the two visualization forms by clicking.

Initially, the order of the pixel maps and line charts is decided by the attribute order in the data. While users are exploring the correlation in the tree view, associated pixel maps and line charts will be highlighted and reordered to the top of the list. When time partition operations are performed in the tree view, associated pixel maps are first expanded to line charts and all associated line charts are reordered to the top. Grey rectangles are added on line charts to label the partitioned time regions.

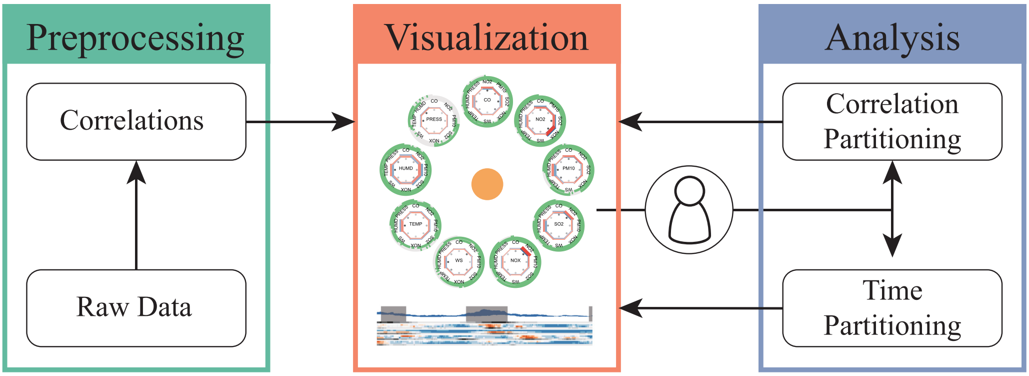


Fig. 11. The pipeline of the prototype system.

7 VISUAL EXPLORATION WITH THE TCPTREE

We develop a prototype system to support users to interactively explore the correlations among air quality variables based on the TCPTree structure (as shown in Figure 1). The system pipeline is shown in Figure 11. The system consists of three major modules, including a preprocessing module, a visualization module, and a analysis module. In the preprocessing module, the symmetric and asymmetric correlations among variables in raw time-series data are calculated and stored. In the visualization module, the correlations are presented to users in a integrated user interface. It consists of three major views: a TCPTree view, which visualizes the TCPTree structure; a correlation time-series view, which shows the details of correlation time series among variables; and a control panel, which enables users to select partitioning operations and adjust parameters in partitioning algorithm. In analysis module, a TCPTree structure is maintained and updated based on user interactions. The system is based on the Browser/Server technique. The server is implemented by JAVA and the user-interface is implemented by Javascript with D3.js and angularjs.

The prototype system provides rich interactions to support the exploration of time-varying correlations with the dynamic constructions of the TCPTree structure.

- **Construction of the TCPTree.** The TCPTree structure is dynamically built by a series of user interactions. After clicking on a tree node, available partitioning operations will be shown in the control panel. In the correlation-partitioning panel, users can filter variables by checking and unchecking variables. In the time-partitioning panel, users can adjust the upper bound and the lower bound of the time-partitioning threshold and select different partition basis. After a partitioning operation is applied to the selected node, a group of new nodes will be generated and appended to the selected node. The visualization of the TCPTree will also be updated: The selected node and its sibling will be collapsed into the sunburst diagram (see in Figure 8(A)) and new nodes will be shown around the sunburst diagram (see in Figure 8(B)). In this way, users can iteratively select tree nodes, apply partitioning operations on the nodes, and dynamically build the TCPTree.
- **Traversal in the TCPTree.** The system supports users to freely traverse in the TCPTree. The sunburst diagram help users navigate their exploration in the TCPTree. After clicking on nodes in the sunburst diagram, currently displayed nodes are first collapsed into the sunburst diagram and the clicked node and its sibling will be expanded and shown. The visualization of TCPTree and the details of correlation time series are linked. Once a certain time node is specified, the related correlation time series will be highlighted and moved to the top of the correlation time-series view.

- **Hierarchical Analysis of Aggregated Correlations.** Starting from analyzing many-to-many correlations depicted in the root node, nodes indicating one-to-many and one-to-one correlations can be gradually obtained by continual partition operations whose parameters determine the examine levels of correlations. The temporally aggregated correlations attached in time tree can be either one-to-many or one-to-one correlations that, respectively, summarizes the correlations among variable sets and variable pairs in a time interval. In addition, comparing the aggregated correlations of a sequence of time intervals is made easy, because they are visually aligned around the central node.
- **Temporal Analysis of Correlation Time Series.** On one hand, the correlation time-series map is supposed to respond to interactions in the TCP tree view, such as node selection, in support of correlation analysis between various variables; on the other hand, the correlation time-series map provides guidance for correlation analysis by indicating not only the temporal trend but also the temporal variations in terms of symmetrical and asymmetrical correlations. Users are allowed to traverse along the time line and investigate the aggregated correlations around a time point with the help of the correlation map.

8 CASE STUDY

We applied our approach to a real-world Air Quality dataset. This dataset was collected in seven observation stations of a modern city (8 million citizens) in 3 years (2009–2011). Each station contains several sensors, recording nine variables every hour: five air pollutants (CO, NO2, PM10, SO2, NOX), and four weather variables (Speed of Wind (WS), Temperature (TEMP), Humidity (HUMD), Pressure (PRESS)). A pre-processing stage is adopted in our research to filter anomalous cases, such as silent samples, faults, and outliers. After that, a TCP tree of either air quality variables or sensors are generated. All the experiments were conducted on a PC with 3.2GHz dual core, 8G memory.

8.1 Parameters

The integrated system requires tuning a sequence of user-adjustable parameters, which can be classified into two categories.

In terms of the variable tree, the variable hierarchy can be determined by users before a correlation-partitioning is applied to a tree node by checking or unchecking variables in the parameter panel, as shown in Figure 1(C).

For time tree, the time-based partition of a correlation time series is performed in an adaptive fashion. Four parameters of time-partitioning are user-adjustable, including the two correlation thresholds, time-interval and time-partitioning basis (see in Figure 1(D)).

8.2 Case 1: Correlations among Variables

Using the TCP tree, the analyst started by checking the aggregated correlations among variables. To analyze the correlations between pollutants, he hid the weather variable nodes. By inspecting both asymmetrical and symmetrical correlations with the tree view, he quickly found that NO2 dominantly influences NOX and SO2 while the other correlations are relatively weak (Figure 12(A)). This is reasonable, because NO2 is the dominant part of NOX and is released mainly by combustions, such as vehicles or power plants. Further investigation into the NO2 variable suggests that there exists interesting correlations depicting an influence chain from NO2 to CO, and finally to PM10. Interestingly, correlations associated with NO2 and PM10 seem to be different (Figure 12). In contrast to NO2, which affects other variables, PM10 is affected by other variables. After examining the one-to-many correlations, the analyst discovered that NO2 is a strong influence factor on other pollutants, while PM10 barely receives significant influences from others

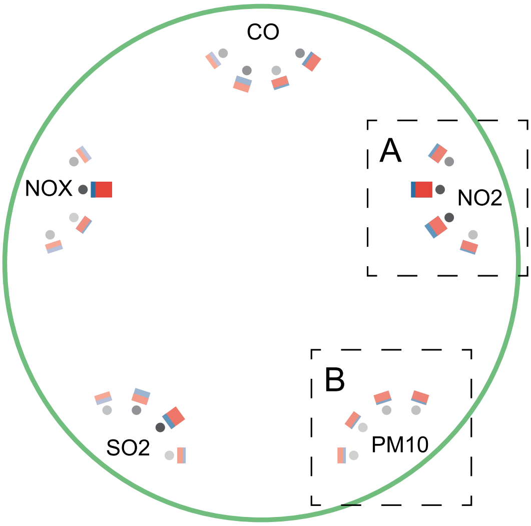


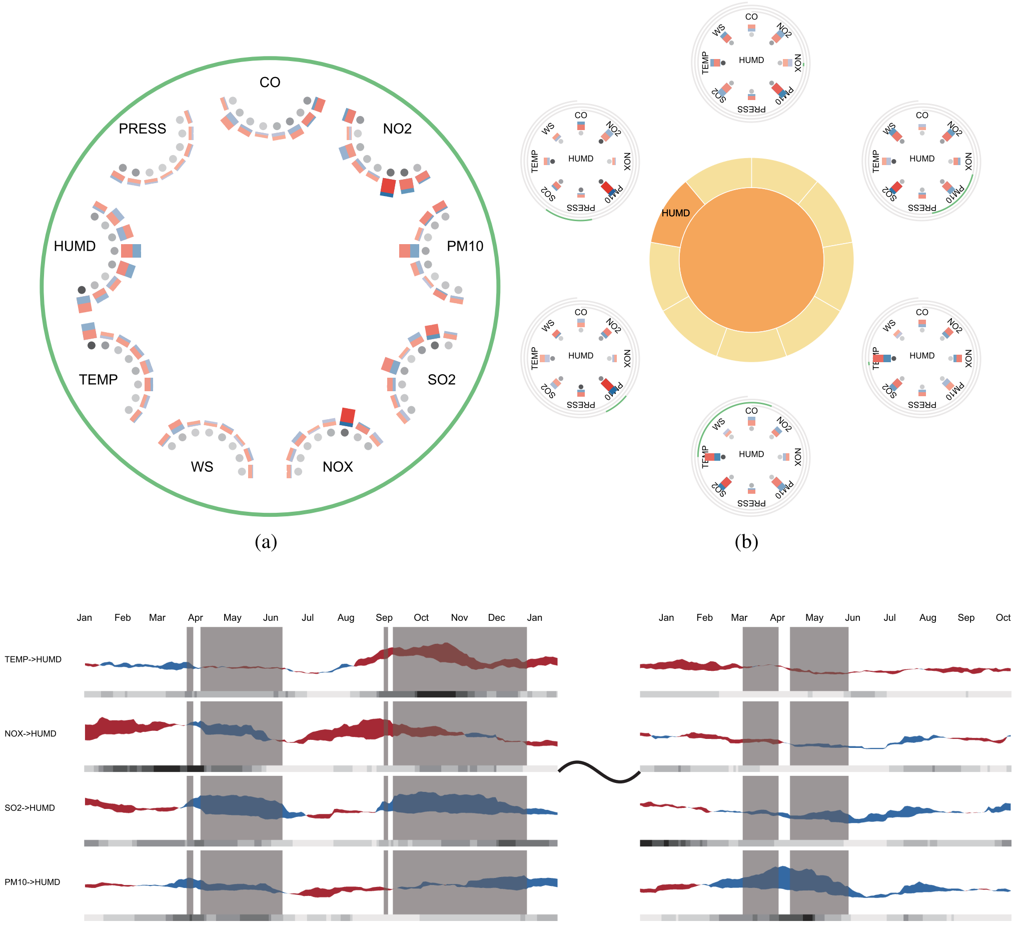
Fig. 12. Correlation summarizations among five pollutants shown by a many-to-many node. (A) The aggregated asymmetrical and symmetrical one-to-many correlations of NO2. (B) The aggregated asymmetrical and symmetrical one-to-many correlations of PM10.

because of the weak linking edges. According to the analysis process, the analyst ultimately drew the conclusion that CO and NO2 play a predominant role in contributing to the release of PM10. This conclusion makes sense, because PM10 is mainly caused by coal-based combustion.

To further investigate how weather variables influence the pollutants, the analyst studied all weather variables. Almost all the strongest asymmetrical correlations come from the HUMD (Figure 13(a)). The one-to-many aggregated correlations indicate that though the mutual information between each variable pair is strong, it is hard to determine the influence direction, because the difference of the asymmetrical correlation is too small. Thus, he selected the HUMD to check the one-to-many correlation time series. The adaptive partition with a threshold yields several interesting findings that show different patterns (Figure 13(b)). He studied the consecutive time spans and realized the the unsteady mutual influence from HUMD to the other variables, among which TEMP always has a high symmetrical correlation. He checked the details of the correlations and corresponding time slices in the detailed view (Figure 13(c)). According to the time tree node, most pollutants are influenced by HUMD as the color of the corresponding rectangles are relatively deeper.

The analyst continued his exploration by selecting other variables. Surprisingly, the one-to-many correlation time series of all pollutants appear to have a similar trend from September to November every year. For example, according to the details of the correlations among CO and other variables, almost all correlations increase among these months in 2009 and 2010, especially for NO2 and PM10 (Figure 14(a)). By time-partitioning, two time slices are obtained (see in Figure 14(b)). From the aggregated correlations, it is clear that CO significantly influences PM10 and has strong correlation with SO2 (for large $I(CO; SO2)$ and small $TD(CO, SO2)$), but it is influenced by NO2.

One possible reason might be the coal-based centralized heating service provided by the government in the city, which starts at every September, and is turned on or off depending on the actual temperature. If the service is on, then it consumes a vast amount of carbon energy and causes a dramatic increase of pollutants. The truth that the centralized heating service is not stable in these months may be related to the variations of correlations among air pollutants.



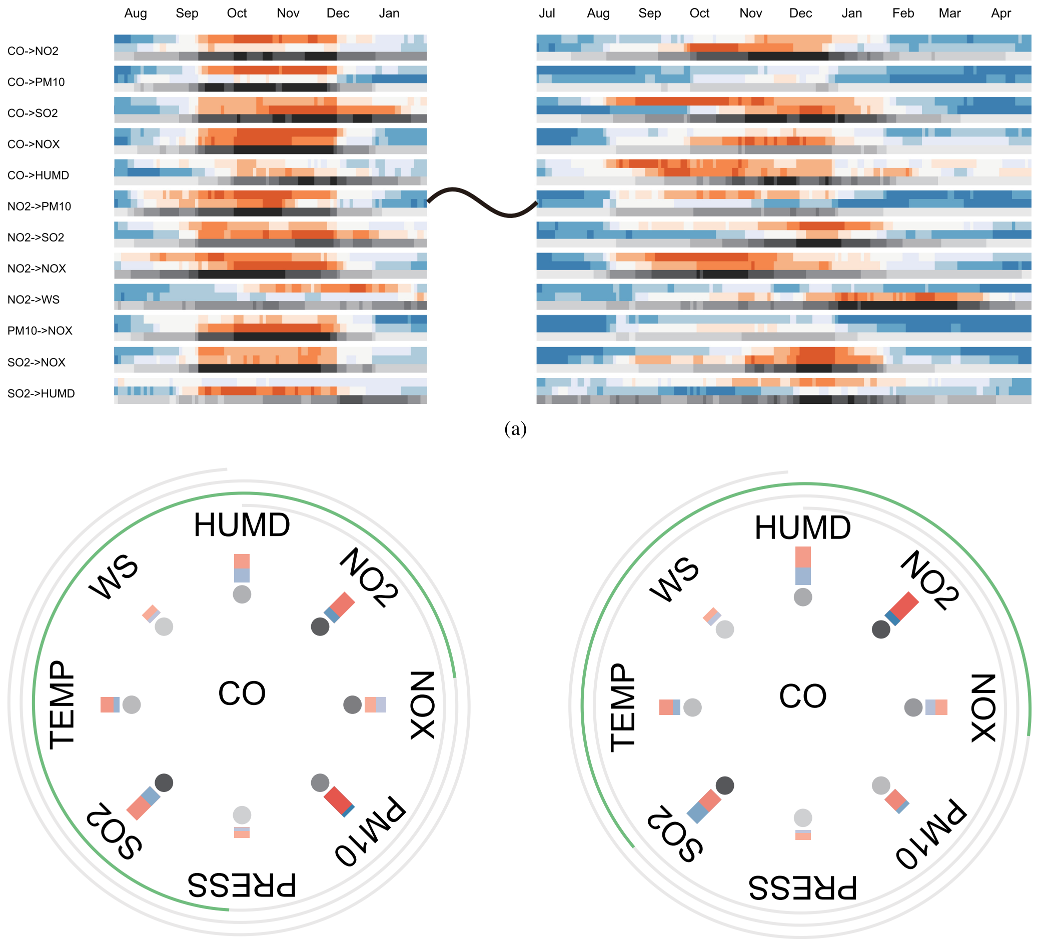
(a) (b)

(c)

Fig. 13. Navigating the time tree node relating to HUMD. (a) The aggregated many-to-many correlations of all the pollutants and weather variables. (b) The time-partitioning result of the aggregated one-to-many correlations of HUMD. Several consecutive partitions are exploited. (c) The correlation time series that correspond to consecutive partitions in (b).

8.3 Case 2: The Correlations among Observation Stations

The analyst decided to navigate the second level of the variable tree to discover the correlations among observation stations. He expanded the tree node associated with one variable (e.g., CO). The tree view shows the correlations among observation stations. Two dominant influences (from s3 to s5 and from s4 to s5) attracted his attention (Figure 15(a)). By referring to the map, he found that these three stations are quite near (right of Figure 15(a)). When further studying other correlations, the analyst discovered the correlation pattern among observation stations based on CO. It seems that s3 always influenced the others. The streamgraph verified this observation except for the months from October 2010 to February 2011. In these months, s4 influenced s3 significantly (Figure 15(b)). To find more evidences, the analyst studied SO2. A similar pattern appeared (Figure 15(a)). He concluded that s3, s4, and s5 are likely to be three interrelated pollution areas. The



(a)

(b)

Fig. 14. (a) The correlation time series for CO. (b) The partitions and their correlation maps. A clear periodical pattern can be seen: CO and other variables have strong correlations from September to December.

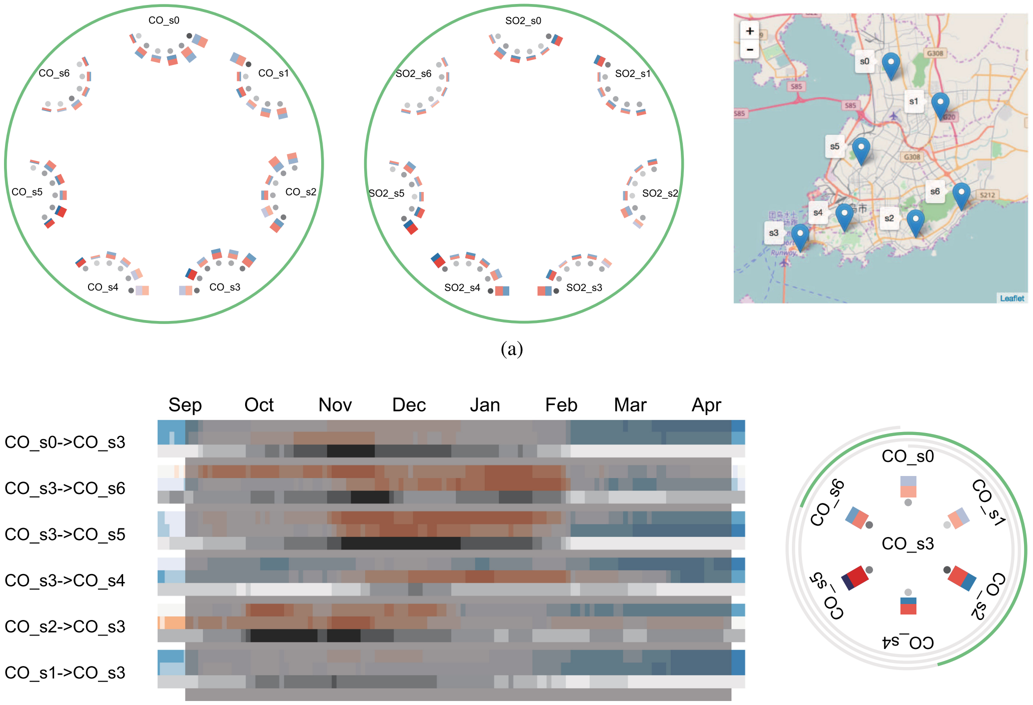
pollutants CO and SO2 probably spread from s3 and s4 to s5 while the influence between s3 and s4 changed in a specific time period.

Furthermore, the analyst found that s6 may be a special station that is seldom influenced by others. He then checked the sensor positions in the map and found s6 to be quite near a mountain in the southeast. The unique geographical location could be the reason of its low dependence.

9 CONCLUSION

This article presents a novel data structure called TCP tree that captures both the variable hierarchy and the temporal variation of correlations hidden in the air quality data. The case study on a real-life dataset verifies that such a hierarchical structure can help exploit the sparsity of a large-scale air quality time series in an information-theoretic way.

As future work is concerned, we believe that the proposed hierarchical structure can be extended for characterizing dynamic network structures and applied in analyzing sensor network data. A hybrid network-tree structure can be suitable for this scenario. We also plan to extend the proposed



(a)

(b)

Fig. 15. (a) Aggregated correlations of s6 (sensor 6) under variable SO2 and CO and the position of sensors. (b) The partition based on s3 highlights time intervals in which s4 influenced s3, while s3 influenced other sensors.

method to other time-varying dataset, e.g., time-varying volume or flow dataset. In addition, we plan to conduct evaluation on the TCP tree and the system for the efficiency of the data structure and the visual designs for supporting users analyzing the correlations of multivariate time series.

REFERENCES

[1] COLORBREWER 2.0. Retrieved from http://colorbrewer2.org/#type=sequential&scheme=Reds&n=3.

[2] Ayan Biswas, Soumya Dutta, Han-Wei Shen, and Jonathan Woodring. 2013. An information-aware framework for exploring multivariate data sets. *Trans. Visual. Comput. Graph.* 19, 12 (2013), 2683–2692.

[3] Anna Lisa Bondi and Antonella Plaia. 2005. Weather variables and air pollution via hierarchical linear models. *Stat. Environ.* (2005), 237–240.

[4] U. D. Bordoloi and Han Wei Shen. 2005. View selection for volume rendering. In *IEEE Visualization*. 62.

[5] Stefan Bruckner and Torsten Möller. 2010. Isosurface similarity maps. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 773–782.

[6] Min Chen, Miquel Feixas, Ivan Viola, Anton Bardera, Han-Wei Shen, and Mateu Sbert. 2016. *Information Theory Tools for Visualization*. CRC Press.

[7] Min Chen and Amos Golan. 2016. What may visualization processes optimize? *Trans. Visual. Comput. Graph.* 22, 12 (2016), 2619–2632.

[8] Min Chen and Heike Jaenicke. 2010. An information-theoretic framework for visualization. *Trans. Visual. Comput. Graph.* 16, 6 (2010), 1206–1215.

[9] Thomas M. Cover and Joy A. Thomas. 2012. *Elements of Information Theory*. John Wiley & Sons.

[10] Tuan Nhon Dang, Anushka Anand, and Leland Wilkinson. 2013. Timeseer: Scagnostics for high-dimensional time series. *Trans. Visual. Comput. Graph.* 19, 3 (2013), 470–483.

[11] M. Demuzere, R. M. Trigo, Vila Guerau De Arellano, and N. P. M. Van Lipzig. 2009. The impact of weather and atmospheric circulation on O3 and PM10 levels at a rural mid-latitude site. *Atmos. Chem. Phys.* 9, 2009 (2009), 2695–2714.

[12] Miquel Feixas, Esteve Del Acebo, Philippe Bekaert, and Mateu Sbert. 1999. An information theory framework for the analysis of scene complexity. In *Computer Graphics Forum*, Vol. 18. Wiley Online Library, 95–106.

[13] Yi Gu and Chaoli Wang. 2011. Transgraph: Hierarchical exploration of transition relationships in time-varying volumetric data. *Trans. Visual. Comput. Graph.* 17, 12 (2011), 2015–2024.

[14] Steffen Hadlak, Heidrun Schumann, Clemens H. Cap, and Till Wollenberg. 2013. Supporting the visual analysis of dynamic networks by clustering associated temporal attributes. *IEEE Trans. Visual. Comput. Graph.* 19, 12 (2013), 2267–2276.

[15] Martin Haidacher, Stefan Bruckner, Armin Kanitsar, and M. Eduard Gröller. 2008. Information-based transfer functions for multimodal visualization. In *Proceedings of the 1st Eurographics conference on Visual Computing for Biomedicine*. Eurographics Association, 101–108.

[16] Susan Havre, Beth Hetzler, and Lucy Nowell. 2000. ThemeRiver: Visualizing theme changes over time. In *Proceedings of the IEEE Symposium on Information Visualization.* 115–123.

[17] K. Ito, G. D. Thurston, A. Nadas, and M. Lippmann. 2001. Monitor-to-monitor temporal correlation of air pollution and weather variables in the North-Central U.S. *J. Exposure Anal. Environ. Epidemiol.* 11, 1 (2001), 21–32.

[18] Guangfeng Ji and Han-Wei Shen. 2006. Dynamic view selection for time-varying volumes. *Trans. Visual. Comput. Graph.* 12, 5 (2006), 1109–1116.

[19] Teng-Yok Lee and Han-Wei Shen. 2009. Visualization and exploration of temporal trend relationships in multivariate time-varying data. *Trans. Visual. Comput. Graph.* 15, 6 (2009), 1359–1366.

[20] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. 2003. TOSSIM: Accurate and scalable simulation of entire TinyOS applications. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems.*

[21] Jessica Lin, Eamonn Keogh, Stefano Lonardi, Jeffrey P. Lankford, and Daonna M. Nystrom. 2004. VizTree: A tool for visually mining and monitoring massive time-series databases. In *Proceedings of the 30th International Conference on Very Large Data Bases-Volume 30*. VLDB Endowment, 1269–1272.

[22] Michael Ogawa and Kwan-Liu Ma. 2010. Software evolution storylines. In *Proceedings of the 5th International Symposium on Software Visualization.* ACM, 35–42.

[23] Catherine Plaisant, Brett Milash, Anne Rose, Seth Widoff, and Ben Shneiderman. 1996. LifeLines: Visualizing personal histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, 221–227.

[24] Huamin Qu, Wing Yi Chan, Anbang Xu, Kai Lun Chung, Kai Hon Lau, and Ping Guo. 2007. Visual analysis of the air pollution problem in Hong Kong. *Trans. Visual. Comput. Graph.* 13, 6 (2007), 1408–1415.

[25] P. S. Quinan and M. Meyer. 2016. Visually comparing weather features in forecasts. *Trans. Visual. Comput. Graph.* 22, 1 (2016), 389–398.

[26] William Ribarsky, Zachary Wartell, and Wenwen Dou. 2012. Event structuring as a general approach to building knowledge in time-based collections. In *Expanding the Frontiers of Visual Analytics and Visualization.* Springer, London, 149–162. DOI: https://doi.org/10.1007/978-1-4471-2804-5_9

[27] Jaume Rigau, Miquel Feixas, and Mateu Sbert. 2005. Shape complexity based on mutual information. In *Proceedings of the International Conference Shape Modeling and Applications.* IEEE, 355–360.

[28] Thomas Schreiber. 2000. Measuring information transfer. *Phys. Rev. Lett.* 85, 2 (2000), 461.

[29] R. Sharovsky, L. A. M. César, and J. A. F. Ramires. 2004. Temperature, air pollution, and mortality from myocardial infarction in Sao Paulo, Brazil. *Brazil. J. Med. Biol. Res.* 37, 11 (2004), 1651–1657.

[30] Han-Wei Shen, Ling-Jen Chiang, and Kwan-Liu Ma. 1999. A fast volume rendering algorithm for time-varying fields using a time-space-partitioning (TSP) tree. In *Proceedings of the Conference on Visualization: Celebrating Ten Years.* IEEE Computer Society Press, 371–377.

[31] Lei Shi, Qi Liao, Yuan He, Rui Li, Aaron Striegel, and Zhong Su. 2011. SAVE: Sensor anomaly visualization engine. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology.* 201–210.

[32] Shigeo Takahashi, Issei Fujishiro, Yuriko Takeshima, and Tomoyuki Nishita. 2005. A feature-driven approach to locating optimal viewpoints for volume visualization. In *IEEE Visualization.* 495–502.

[33] Christian Tominski, James Abello, and Heidrun Schumann. 2004. Axes-based visualizations with radial layouts. In *Proceedings of the ACM Symposium on Applied Computing.* ACM, 1242–1247.

[34] Martin Turon. 2005. MOTE-VIEW: A sensor network monitoring and management tool. In *Proceedings of the 2nd IEEE Workshop on Embedded Networked Sensors.* 11–17.

[35] Jarke J. Van Wijk and Edward R. Van Selow. 1999. Cluster and calendar based visualization of time-series data. In *Proceedings of the IEEE Symposium on Information Visualization.* 4–9.

[36] Péter Völgyesi, András Nádas, Xenofon Koutsoukos, and Ákos Lédecz. 2008. Air quality monitoring with SensorMap. In *International Conference on Information Processing in Sensor Networks.* 529–530.

[37] Greg Ver Steeg and Aram Galstyan. 2012. Information transfer in social media. In *Proceedings of the 21st International Conference on World Wide Web*. ACM, 509–518.

[38] Raul Vicente, Michael Wibral, Michael Lindner, and Gordon Pipa. 2011. Transfer entropy—A model-free measure of effective connectivity for the neurosciences. *J. Comput. Neurosci.* 30, 1 (2011), 45–67.

[39] Ivan Viola, Miquel Feixas, Mateu Sbert, and Meister Eduard Groller. 2006. Importance-driven focus of attention. *Trans. Visual. Comput. Graph.* 12, 5 (2006), 933–940.

[40] Chaoli Wang and Han-Wei Shen. 2006. LOD map-a visual interface for navigating multiresolution volume visualization. *Trans. Visual. Comput. Graph.* 12, 5 (2006), 1029–1036.

[41] Chaoli Wang and Han-Wei Shen. 2011. Information theory in scientific visualization. *Entropy* 13, 1 (2011), 254–273.

[42] Chaoli Wang, Hongfeng Yu, Ray W. Grout, Kwan-Liu Ma, and Jacqueline H. Chen. 2011. Analyzing information transfer in time-varying multivariate data. In *Proceedings of the IEEE Pacific Visualization Symposium*. IEEE, 99–106.

[43] Chaoli Wang, Hongfeng Yu, and Kwan-Liu Ma. 2008. Importance-driven time-varying data visualization. *Trans. Visual. Comput. Graph.* 14, 6 (2008), 1547–1554.

[44] Jingyuan Wang, Robert Sisneros, and Jian Huang. 2013. Interactive selection of multivariate features in large spatiotemporal data. In *Proceedings of the IEEE Pacific Visualization Symposium*. 145–152.

[45] Marc Weber, Marc Alexa, and Wolfgang Muller. 2001. Visualizing time-series on spirals. In *Proceedings of the IEEE Symposium on Information Visualization*. 7–14.

[46] Lijie Xu, Teng-Yok Lee, and Han-Wei Shen. 2010. An information-theoretic framework for flow visualization. *Trans. Visual. Comput. Graph.* 16, 6 (2010), 1216–1224.

[47] Li Yu, Aidong Lu, William Ribarsky, and Wei Chen. 2010. Automatic animation for time-varying data visualization. In *Proceedings of the Computer Graphics Forum*, Vol. 29. Wiley Online Library, 2271–2280.

[48] Yu Zheng, Furui Liu, and Hsun-Ping Hsieh. 2013. U-Air: When urban air quality inference meets big data. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1436–1444.

[49] Yu Zheng, Furui Liu, and Hsun-Ping Hsieh. 2013. U-Air: When urban air quality inference meets big data. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1436–1444.

Received August 2017; revised April 2018; accepted June 2018