Granger Causality Based Hierarchical Time Series Clustering for State Estimation [⋆]

Sin Yong Tan [*] **Homagni Saha** [*] **Margarite Jacoby** [**]
Anthony R. Florita [***] **Gregor P. Henze** [**,***] **Soumik Sarkar** [*]

[*] *Iowa State University, Ames, IA 50010 USA*
(e-mail: tsyong98, hsaha, soumiks@iastate.edu).
[**] *University of Colorado Boulder, Boulder, CO 80309 USA*
(e-mail: margarite.jacoby, gregor.henze@colorado.edu).
[***] *National Renewable Energy Laboratory, Golden, CO 80401 USA*
(e-mail: anthony.florita@nrel.gov).

Abstract:
Clustering is an unsupervised learning technique that is useful when working with a large volume of unlabeled data. Complex dynamical systems in real life often entail data streaming from a large number of sources. Although it is desirable to use all source variables to form accurate state estimates, it is often impractical due to large computational power requirements, and sufficiently robust algorithms to handle these cases are not common. We propose a hierarchical time series clustering technique based on symbolic dynamic filtering and Granger causality, which serves as a dimensionality reduction and noise-rejection tool. Our process forms a hierarchy of variables in the multivariate time series with clustering of relevant variables at each level, thus separating out noise and less relevant variables. A new distance metric based on Granger causality is proposed and used for the time series clustering, as well as validated on empirical data sets. Experimental results from occupancy detection and building temperature estimation tasks shows fidelity to the empirical data sets while maintaining state-prediction accuracy with substantially reduced data dimensionality.

Keywords: Time series clustering, time series state estimation, occupancy detection, dimensionality reduction

1. INTRODUCTION

Complex cyber-physical systems (CPS) are abundant in engineering applications. Examples include modern buildings (Liu et al. (2018); Tan et al. (2019)), transportation networks (Liu et al. (2016)), robotics (Dunbabin and Marques (2012)), smart home Internet-of-Things (IoT) (Darianian and Michael (2008)), and wind farms (Jiang et al. (2017)). Such systems feature a large number of sensors collecting data, which form vast multivariate time series, while containing different types of interactions among variables. Interactions can be both spatial and temporal, and for the purposes of control and decision making it is crucial to understand such interactions. It is possible to use physics-based models for understanding these multivariate time series, but it becomes infeasible with an increasing number of subsystems. When considering the total number of states that may be used for estimation, the problem can become intractably large without dimensionality reduction or model simplification. Data-driven techniques (Qin (2012); Yin et al. (2012)) have started receiving attention from industry and academia alike because they tend to be scalable and accurate. These techniques rely on vast quantities of data to learn efficient representations. Forming efficient representations of the system for state estimation can benefit from understanding spatiotemporal (causal) interactions across the system. Information theoretic techniques can help in this regard; e.g., Granger causality can provide relevant insights when considering the effectiveness of control mechanisms (Granger (1988)) or for identifying key features in large-scale CPSs for anomaly detection (Saha et al. (2018)). Comparable research exists in finance (Dimpfl and Peter (2013)), neuroscience (Vicente et al. (2011)), and social sciences (Ver Steeg and Galstyan (2012)); however, with focus on identifying causal interactions among large-scale CPSs, many time series aspects have not been explored sufficiently.

Time series clustering techniques can be extremely useful in managing "state explosions" by reducing the number of variables in multivariate observations. Current research can be divided into three thrusts:

(1) *Representation methods* focus on learning efficient representations of multivariate time series for dimen-

[⋆] This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AR0000938. Funding provided by U.S. Department of Energy, Advanced Research Projects Agency-Energy (ARPA-E). The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

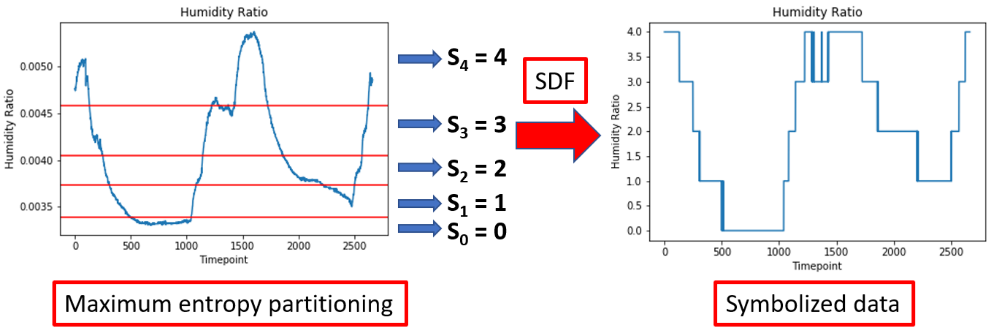

Fig. 1. Illustrating the process of symbolic dynamic filtering, where each partition, or bin, on the left plot is expressed as an integer value ranging from 0 to 4, from which we obtain the symbolized time series on the right.

sionality reduction, which is achieved by transforming the time series into a lower dimensional feature space or by the extraction of relevant features (Lin et al. (2003); Keogh et al. (2005); Duan et al. (2006)).

(2) *Similarity measures* focus on developing metrics and finding distances among time series. Commonly used distance metrics are Hausdorff Basalto et al. (2007), Modified Hausdorff, HMM-based Oates et al. (2000a), Dynamic Time Warping (DTW) Berndt and Clifford (1994), Euclidean distance, and Longest Common Sub-Sequence (LCSS) Górecki (2014). Herein, we propose a novel metric based on the extent of causality to a target time series variable in multivariate series.

(3) A *cluster prototype* is an element in the data space that serves to characterize a cluster. The aim is to find appropriate cluster prototype (Rabiner et al. (1979); Bagnall and Janacek (2005); Ratanamahatana and Keogh (2005)), in which the quality of clustering is often dependent on the quality of the prototype.

Existing time series clustering algorithms can be divided into six main types: *partitioning-based* as in (Hautamaki et al. (2008); Guo et al. (2008)) , *hierarchical* as in (Oates et al. (2000b); Hirano and Tsumoto (2005)), *grid-based* as in (Wang et al. (1997); Sheikholeslami et al. (1998)), *model-based* as in (Kohonen (1990); Corduas and Piccolo (2008); Ramoni et al. (2002)) *density-based* as in (Ankerst et al. (1999); Ester et al. (1996)), and *multi-step clustering* as in (Lin et al. (2005); Zhang et al. (2011)). We chose to combine an agglomerative hierarchical clustering (Al-Dabooni and Wunsch (2018)) approach with state estimation at each level, giving rise to a multi-level state estimation problem. While hierarchical methods generally provide better visualization capabilities on what features are relatively important, they do not require specifications of the required number of clusters and are thus flexible. Furthermore, we make the technique scalable by using a repartitioning technique (discussed in Section 2.1) using symbolic dynamic filtering.

Contributions:

(1) A hierarchical time series clustering technique based on symbolic dynamic filtering and a novel Granger causality based similarity metric is proposed.

(2) The algorithm's performance for state estimation problems using real data sets demonstrates its robustness.

(3) An efficient dimensionality reduction technique for time series is provided that can maintain the prediction accuracy for a target variable when performing state estimation.

2. BACKGROUND

In this section, we discuss concepts that will become a foundation for our hierarchical clustering framework.

2.1 Symbolic Dynamic Filtering (SDF) based Encoding

Symbolic dynamic filtering is a tool used to describe the behavior of nonlinear dynamical systems. The concept of formal languages is used to describe transitions from smooth dynamics to a symbolic domain that is discrete (Badii and Politi (1999)). The core idea is to partition the phase space of the dynamical system so that a coordinate grid for the space is obtained in the form of a finite number of cells. The cells, arranged in order of occurrence, form the symbol sequence S, and the unique identifiers that are used to denote each symbol form the alphabet set Σ. Appropriately partitioning a time series data can filter out noise and make the representation more robust (Gupta and Ray (2007)). An example of partitioning a continuous time series, $X(t)$, is presented in Fig. 1. We assume there is a partitioning function $\mathcal{X} : X(t) \to D$, mapping the continuous elements of $X(t)$ to the discrete elements of D. There are numerous partitioning method that have been used in the literature, such as uniform partitioning, maximum entropy partitioning, maximally bijective discretization, and statistically similar discretization (Sarkar and Srivastav (2016)). In our work, we use maximum entropy partitioning (MEP) to discretize.

2.2 Symbol Sequence to State Sequence

Time embedding is a technique followed in the dynamical systems literature to identify key features in time series as a collection of states. In our work, a simple embedding is performed to convert a symbol sequence to a state sequence. Consider a *symbolic* sequence denoted as $X = \{x_1, x_2, \ldots, x_n\}$, each *state* can be interpreted as a collection of symbols preceding a time step t. To quantify the embedding dimension, we denote k as a parameter to indicate the length of history embedded in a *state* (also known as "depth" of embedding). Using this parameter, we can denote a state at time t with an embedding dimension of k as, $\bar{x}_t^k := \{x_{t-k}, \ldots, x_{t-1}, x_t\}$ (i.e., k historical observations from X at time t). Hence, a *state sequence* can be expressed as $\bar{X}^k = \{\bar{x}_1^k, \bar{x}_2^k, \ldots, \bar{x}_n^k\}$.

Figure 2 shows an example of state sequence generation from a symbol sequence. In this example, the embedding dimension is $k = 2$, thus each state in the state sequence contains a collection of three symbols, including one current timestep symbol and two historic symbols from previous timesteps.

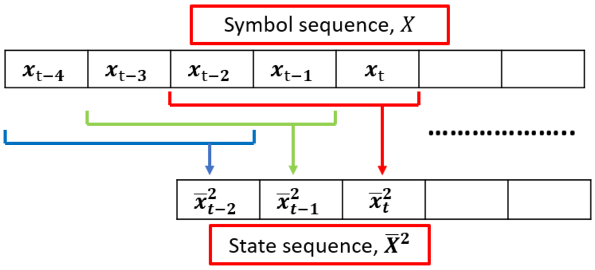

Fig. 2. Demonstrating the process of embedding a symbol sequence X into a state sequence $\bar{X}^2$, with each *state* having an embedding dimension $k = 2$.

2.3 Formulation of Granger causality

In this section we review the concept of Granger causality. The idea behind Granger causality is that a time series Y, Granger causes another time series X, if the predictive power of a model for X is increased by including the histories of both X and Y, over a model that includes the past history of X alone. After applying SDF on the time series variables X and Y, we consider them to be discrete. Let a *state* $\bar{x}_{t-1}^k = \{x_{t-k-1}, ..., x_{t-2}, x_{t-1}\}$, where t is the discrete time index. Let $F(x_t|\bar{x}_{t-1}^k, \bar{y}_{t-1}^k)$ denote the distribution function of the target variable X, conditioned on the joint (k-lag) history $\bar{x}_{t-1}^k, \bar{y}_{t-1}^k$ of both itself and a source variable Y. Also, let $F(x_t|\bar{x}_{t-1}^k)$ denote the distribution function of X_t conditioned on just its own k-history. Then variable Y is said to Granger cause variable X (with k lags) if and only if:

$$F(x_t|\bar{x}_{t-1}^k, \bar{y}_{t-1}^k) \neq F(x_t|\bar{x}_{t-1}^k) \tag{1}$$

Granger's formulation was based on vector autoregressive modeling (VAR), which often restricts application to general nonlinear processes. Recently, however, a nonlinear data-driven causality metric, transfer entropy, has been introduced (Vicente et al. (2011)). To describe transfer entropy, we begin by stating Shannon entropy for the variable X, which is given by the following expression:

$$H_X = -\sum_n p(x) \log(p(x)) \tag{2}$$

where $x = 1, 2, \ldots, n$, for all states (total of n) the variable X can assume, and $p(x)$ is the associated probability of the state x occurring. Now, let us assume we have another variable Y, with associated states obtained after discretization denoted by y. Conditional entropy is given by:

$$H_{X|Y} = \sum_n p(x,y) \log(p(x|y)) = H_{XY} - H_Y \tag{3}$$

where H_{XY} is the entropy of the equivalent time series representing X and Y occurring together

$$H_{XY} = -\sum_{n_X} \sum_{n_Y} p(x,y) \log(p(x,y)). \tag{4}$$

Consider two such symbolic time series X and Y. Let the observation at the $(t+1)^{th}$ instant of sequence X be x_{t+1}, which depends on its previous state, $\bar{x}_t^k := \{x_{t-k}, \ldots, x_{t-1}, x_t\}$ and the state of Y, $\bar{y}_t^k := \{y_{t-k}, \ldots, y_{t-1}, y_t\}$. With this setup, transfer entropy for the two systems can be defined as the difference of conditional entropies as follows (Barnett et al. (2009)):

$$\mathcal{T}_{X \to Y} = H_{Y|\bar{Y}^k} - H_{Y|\bar{Y}^k \bar{X}^k} \tag{5}$$

There exist efficient methods in literature to calculate both the values of conditional entropies as given by the following equations (using a time lag of 1, symbols at $t+1$ rely on states till t):

$$H_{Y|\bar{X}^k,\bar{Y}^k} = -\sum_n p(y_{t+1}) \log(p(y_{t+1}|\bar{x}_t^k, \bar{x}_t^k)) \tag{6}$$

$$H_{Y|\bar{Y}^k} = -\sum_n p(y_{t+1}) \log(p(y_{t+1}|\bar{y}_t^k)) \tag{7}$$

Using Equations 3 through 7, and applying Bayes rule to evaluate conditional probabilities, we obtain (Martini et al. (2011)):

$$\mathcal{T}_{X \to Y} = \sum p(y_{t+1}, \bar{y}_t^k, \bar{x}_t^k) \frac{\log(p(y_{t+1}|\bar{y}_t^k, \bar{x}_t^k)}{\log(p(y_{t+1}|\bar{y}_t^k)}) \tag{8}$$

We consider this in a symbolic domain, which makes it similar to symbolic transfer entropy, elaborated in detail in Staniek and Lehnertz (2008). In Schindlerova (2011) the authors proved that transfer entropy and Granger causality are equivalent for Gaussian variables. We use transfer entropy as the metric for causality. However, justifying why transfer entropy is an appropriate replacement for the original formulation of Granger causality is beyond the scope of the paper and readers are referred to (Saha et al. (2018)) for further clarification.

3. METHODOLOGY AND FRAMEWORK

3.1 Hierarchical Clustering

Our hierarchical clustering approach is an agglomerative, or bottom-up, clustering approach, meaning we start with a collection of distinct variables and gradually reduce the number of variables by forming joint representations (or clusters) of variables. Such a joint representation is often referred to as a "supernode" in CPSs (Alcaraz et al. (2017)). The key technique that we use for forming efficient abstractions of continuous time series is through SDF, discussed in Section 2.1. After converting the multivariate time series into individual symbol sequences, we initiate our hierarchical clustering algorithm. During the clustering process, our algorithm selectively fuses a pair of time series together at each level of the tree. It does so by comparing state estimation capabilities for all pairwise combinations of time series using our Granger causality based similarity metric at the current level in the tree, similar to Ward's method in hierarchical clustering by Murtagh and Legendre (2014). Figure 3 shows an illustration of how the cluster was developed at different levels as the clustering algorithm progresses. Starting with n nodes at the root, the algorithm develops $n-1$ levels in the clustering tree, with each horizontal line representing one layer. At each level, the input data dimension reduces by one due to the merging.

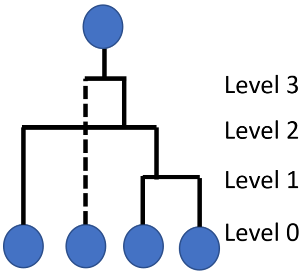

Fig. 3. An example of a 3-level hierarchical clustering tree result from our clustering algorithm.

3.2 Granger causality based clustering similarity metric

In Section 2.3, we denoted two symbol sequences X and Y, to be the source and target variables respectively, and Eq. 8 provided the directed flow of information from X to Y. In our case of having multiple time series, we designate the variable that we want to estimate the state of as the *target variable* and designate all the other time series as the *source variables* that are eventually (hierarchically) clustered. For the following equations in this subsection, we denote X and Y as individual source variable, XY as the fused source variables, and Z as the target-variable that we are predicting.

In order to maintain the state estimation after fusion, we seek a similarity metric that determines the fusion pairs that possess the minimum difference to the estimation power of individual X and Y source variables from the fused XY variable. In other words, the chosen fused representation XY should retain as much possible information about predicting Z as individual representations of X and Y. We capture this notion by using a metric as follows. Let $\mathcal{M}$ be an ordered list of all possible pairwise combinations of variable indices at a particular level in the hierarchy, where $\mathcal{M} = \binom{n}{2} = \{(1,2),(1,3),\ldots,(n-1,n)\}$, and n is the number of source variables at that hierarchy. At each level of the hierarchy we choose one best fusion combination, indexed as c from all combinations in $\mathcal{M}$ by using the following rule:

$$c = \underset{XY\in\mathcal{M}}{\arg\min}\{(T_{X\to Z} - T_{XY\to Z}) + (T_{Y\to Z} - T_{XY\to Z})\} \quad (9)$$

After we obtain index c, the corresponding variable combination associated with that index are merged together into a "cluster" or fused variable XY. This concept has been often sought in literature as a generalization of the concept of transfer entropy, and used in a slightly different context to improve the transfer entropy metric, when information flows from one variable to another though a third variable in the causal chain. In Sun and Bollt (2014), the authors define "causation entropy" as follows (aligning their notation with ours):

$$\mathcal{C}_{Y\to Z|(Z,X)} = H_{Z|\bar{Z}^k\bar{X}^k} - H_{Z|\bar{Z}^k\bar{X}^k\bar{Y}^k} \quad (10)$$
$$\mathcal{C}_{X\to Z|(Z,Y)} = H_{Z|\bar{Z}^k\bar{Y}^k} - H_{Z|\bar{Z}^k\bar{X}^k\bar{Y}^k} \quad (11)$$

Equation 10 denotes the extra information provided to Z by Y in addition to the information already provided to Z by other sources. Accordingly, Equation 11 denotes the extra information provided to Z by X in addition to the information already provided to Z by other sources. By comparing Equations 3 through 7 with Equations 9 through 11, our similarity metric in Equation 9 can be reinterpreted as the argmin of the list containing the negative of the sum of causation entropies of each pair of source-variables in consideration to the target-variable as shown in Equation 12.

$$\begin{aligned}&(T_{X\to Z} - T_{XY\to Z}) + (T_{Y\to Z} - T_{XY\to Z})\\ &= (H_{Z|\bar{Z}^k} - H_{Z|\bar{Z}^k\bar{X}^k}) - (H_{Z|\bar{Z}^k} - H_{Z|\bar{Z}^k\bar{X}^k\bar{Y}^k})\\ &+(H_{Z|\bar{Z}^k} - H_{Z|\bar{Z}^k\bar{Y}^k}) - (H_{Z|\bar{Z}^k} - H_{Z|\bar{Z}^k\bar{X}^k\bar{Y}^k})\\ &= -(\mathcal{C}_{Y\to Z|(Z,X)} + \mathcal{C}_{X\to Z|(Z,Y)})\end{aligned} \quad (12)$$

In other words, if X and Y are selected for clustering at a certain level, it implies that out of all possible pairs of variable combinations, X and Y together contribute the highest causation entropy. This also means that X provides more information to Z when Y is the extra variable, and Y provides more information to Z when X is the extra variable. Thus, in this scenario, X and Y are selected to be clustered together.

3.3 Symbol sequence fusion using repartitioning

After having selected the pair of variables X and Y to be clustered together, we seek to form a joint representation of the variables for consideration in clustering at the next higher level. Let the symbol sequence of X be given by $\{x_1, x_2, \ldots, x_n\}$ and the symbol sequence of Y be given by $\{y_1, y_2, \ldots, y_n\}$, where n is the length of the symbol sequence, we first form a merged symbol sequence denoted by $\{x_1y_1, x_2y_2, \ldots, x_ny_n\}$. Then, we assign values to the merged symbol sequence by letting x_iy_i be a number in the $b_x \times b_y$-ary numbering system, where b_x is the number of unique symbols in X and b_y the number of unique symbols in Y. After having assigned values to the merged symbol sequence, we again repartition the merged symbol sequence based on the desired number of unique symbols and obtain a joint representation. A detailed flowchart of the process is illustrated in Fig. 4.

3.4 Hierarchical time series clustering algorithm

To summarize our hierarchical clustering algorithm, in this subsection, we formally present our algorithm below. In the algorithm, we denote the target variable as Z and a list of source variables as $S = \{s_1, s_2, ..., s_n\}$ where n is the total number of source variables. On top of that, we define

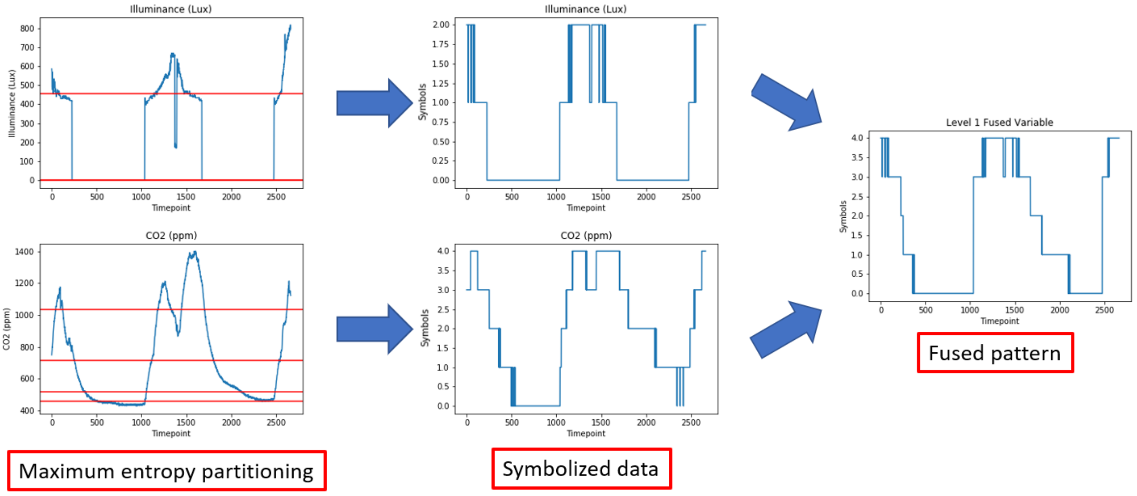

Fig. 4. Flowchart showing the fusion and repartitioning process between illuminance and CO_2 time series data.

another variable n_h as the number of source variables in every hierarchy/level.

Algorithm 1 Hierarchical Time Series Clustering

```
Require: Symbolized source and target variables.
Initialize $n_h = n$.
while $n_h > 1$ do
    Compute $T_{s_1 \to Z}, T_{s_2 \to Z}, \ldots, T_{s_{n_h} \to Z}$
    Combinations $\mathcal{M} = \binom{n_h}{2} = \{m_1, m_2, \ldots\}$
    Compute $T_M = \{T_{m_1 \to Z}, T_{m_2 \to Z}, \ldots\}$
    Determine best fusion pair index $c$ [Eq. 9].
    Fuse selected pair $s_X, s_Y$ and repartition it to desired number of symbols [Section 3.3]
    $S \leftarrow S - \{s_X, s_Y\} + \{s_{XY}\}$
    $n_h \leftarrow n_h - 1$
end while
```

4. EXPERIMENTS

To demonstrate the performance of our algorithm, two open source data sets are used and summarized below.

4.1 Occupancy (OCC) Data Set

The first data set used is the University of California, Irvine's building occupancy detection data set by Candanedo and Feldheim (2016) (OCC data set). This is a multivariate data set comprises of five time series data that describes the indoor condition of an office room. This time series data includes temperature (°C), relative humidity (%), illuminance level (lux), CO_2 (ppm) and humidity ratio (kg-water-vapor/kg-air), each sampled at a 1-minute time interval. The ground truth or the target variable of this data set is the room occupancy, denoted with nominal labels of zeros (unoccupied) and ones (occupied).

4.2 Air Handling Units (AHU) Data Set

The second data set that we used is the OpenEI "Long-term data on 3 office Air Handling Units" (AHU data set). This data set consist of multiple variables that describes the state of the air handling units (AHU) in an office building located in Richland, Washington. For this paper, we used eight variables, including outside air temperature (OAT, $^\circ F$), return air temperature (RAT, $^\circ F$), outside air damper command (OA Damper CMD), cooling valve command (Cool Valve CMD), discharge air temperature (DAT, $^\circ F$), supply fan speed command (Su Fan speed CMD), discharge air static pressure (DA Static P), and return fan speed command (Re Fan Speed CMD), each sampled at a 1-minute time interval. The target variable for this data set is the average zone temperature ($^\circ F$).

Target Variable Discretization. Unlike the OCC data set that already have discreet (0/1) labels, for this AHU data set, we discretized the target variable (average zone temperature) by performing SDF with 10 symbols. Each of the symbols is a small partition bin that represents a small range of temperature. The symbolization is required for transfer entropy computation and symbolic data fusion.

4.3 Classifiers

The classifier that we used for state estimation in the experiment is the random forest classifier. Random forests fall under the umbrella of ensemble learning, where classification or regression is performed by constructing and collecting state estimates from multiple decision trees. For our random forest, we used 500 estimators (trees), each with maximum depth of 100 and using the entropy split criterion.

5. RESULTS AND DISCUSSION

In this section, we present the performance of our algorithm on the above-mentioned data set. We will show that by performing the proposed hierarchical clustering, we effectively reduce the data dimension, while preserving the information and prediction accuracy.

5.1 Hierarchical Clustering Tree

To visualize the hierarchical clustering results of the two above mentioned data set, the clustering tree for both OCC and AHU data set are plotted as shown in Fig. 5.

The root of the clustering trees (Level 0) in Fig. 5 are annotated with all the source variables and two nodes

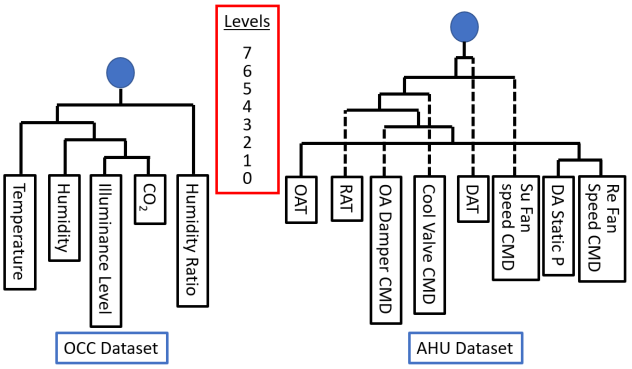

Fig. 5. Clustering process of OCC and AHU data set at different levels from initial n nodes to one supernode.

are merged in each of the levels. Some parameters that was used in the symbolic dynamic filtering and state sequence generation is as follows. For OCC data set, each source variables are partitioned into 5 symbols and with an embedding dimension (depth) of 3 during state sequence generation. On the other hand, 10 symbols are used to partition the variables in the AHU data set, and 5 historical observations (symbols) are embedded in each state.

5.2 Performance Evaluation

To evaluate the quality of the merged variables, we perform a classification in each stage of the tree to determine how the merged variables affect the overall prediction performance. Table 1 tabulates all the prediction results in each level of the clustering tree for both OCC and AHU data set.

Table 1. OCC data set and AHU data set prediction performance at each level of the clustering tree.

Level	OCC Data Set Accuracy	AHU Data Set RMSE
0	93.24 %	1.7774
1	93.00 %	1.8228
2	94.37 %	1.7401
3	90.86 %	1.7263
4	97.12 %	1.6914
5	-	1.8245
6	-	1.9947
7	-	2.0129

To clarify, the level zero in Table 1 represents the prediction performance from using the root variables and could be viewed as the baseline performance for each data set. Due to the different nature of the target variables on both data set, two different metrics, accuracy and root-mean-square error (RMSE) were used to evaluate the prediction performance for OCC and AHU data set respectively.

Although both data sets are building related, the OCC data set has binary nominal labels or class labels, which makes accuracy metric a more suitable choice for performance evaluation. On the other hand, the AHU data set target variable is a regression problem, where RMSE would be a better choice as a measure of difference between the prediction and the true target.

Based on the overview offered in Table 1, the prediction performance was well maintained around the baseline (multivariate classification accuracy without any clustering) for both the OCC and AHU data sets despite natural information loss due to dimensionality reduction. In fact, the OCC data set has a prediction accuracy above 90% throughout all levels. However, it was observed that there is a slight increase in RMSE for the AHU data set on level 5 and so on.

Our deduction for this increase in RMSE is, as the algorithm progresses, it will start to merge variables that are relatively uninformative for the prediction, and in some cases, could negatively impact the fused pattern and overall performance. There are several methods that could be implemented to tackle this issue. One potential solution to this issue is to set a dynamic threshold that limits the fusion with only informational source variables, and stops the algorithm once the remaining potential pairs are below the threshold. Another solution is, instead of limiting the similarity metric threshold, the number of clusters or "supernodes" formed by this hierarchical algorithm can also be a user defined variable, where the algorithm will stops as the supernodes formed had reached the desired number similar to k-means (top-down) clustering.

Fig. 6 shows the plot of the prediction for OCC data set at the level 4 of the clustering tree. The plot shows an almost perfect prediction, also the transition from occupied to unoccupied was captured accurately. The small 2.88% error in the prediction is due to the false positive around the 200-minute timestamp. On the other hand, Fig. 7 presents the plot of the best prediction for AHU data set at level 4 of the clustering tree. Although the prediction has a slight variance, it can capture the transitions in the average zone temperatures accurately.

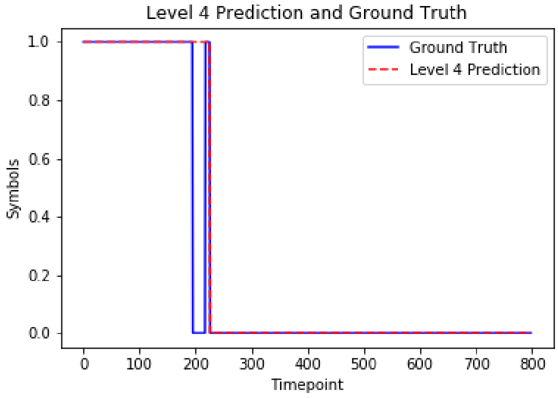

Fig. 6. OCC data set prediction at level 4 (full clustering) of the clustering tree.

In order to verify that the similarity metric can select informational and relevant time series for fusion, we performed an experiment where we introduce random noise (standard normal) along with the original time series as input to observe the resulting clustering pattern. Interestingly, we observed that the noise did not merge with the main cluster early in the early levels, but instead, they formed a cluster of their own, and it merged only with the first (main) cluster in the end when there are no other merging options. This observation is illustrated in Fig. 8, where

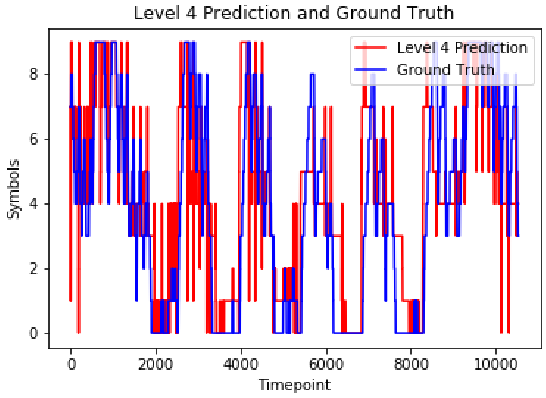

Fig. 7. AHU data set best prediction (at level 4).

the noises are represented with red circles, and they are merged in the second last level to form a “supernode” as shown in the red box, before merging with the main cluster on the left of the tree on the last level. Note that the arrangement or the order of source variables at the root does not affect the clustering sequence in any way, as the fusion pair selection is deterministic, with no randomness involved in the process.

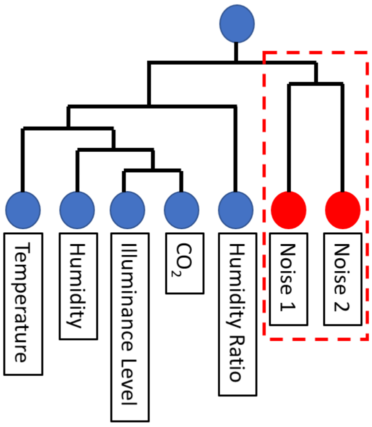

Fig. 8. OCC data set clustering tree with two gaussion white noises (red circles).

6. CONCLUSION

We developed a Granger causality based hierarchical time series clustering algorithm that combines dimensionality reduction with robustification against noise. We also propose a Granger causality based similarity metric that incrementally clusters pairs of most relevant time series that could preserve the predictive power at each clustering level. We show experimentally on real data sets that by the injection of noise sensors in the source variables, the algorithm only merges the noise sensors towards the end of the algorithm when there are no other fusion choices. Results on real data sets also suggest that the algorithm is applicable to both discrete and continuous state estimation. As mentioned in Section 5.2, merging an irrelevant or uninformative time series might negatively impact the state estimation power of the merged variables. In order to tackle this problem, in future work, a dynamic threshold of similarity metric can be set or a desired number of clusters / supernodes formed can be predefined, to ensure that the algorithm will only merge informative time series, or stops the algorithm as it reaches the defined number of clusters. Our algorithm is fairly general to be used in conjunction with a variety of classification and regression tasks. Given our novel approach in combining state estimation with clustering, comparison baselines with other clustering algorithms could not yet be established and this is left for future work. For faster run-time performance, parallelization can be used for the computation of the transfer entropies in Algorithm 1, which will increase performance significantly.

REFERENCES

Al-Dabooni, S. and Wunsch, D. (2018). Model order reduction based on agglomerative hierarchical clustering. *IEEE transactions on neural networks and learning systems*, 30(6), 1881–1895.

Alcaraz, C., Lopez, J., and Choo, K.K.R. (2017). Resilient interconnection in cyber-physical control systems. *Computers & Security*, 71, 2–14.

Ankerst, M., Breunig, M.M., Kriegel, H.P., and Sander, J. (1999). Optics: ordering points to identify the clustering structure. In *ACM Sigmod record*, volume 28, 49–60. ACM.

Badii, R. and Politi, A. (1999). *Complexity: Hierarchical structures and scaling in physics*, volume 6. Cambridge University Press.

Bagnall, A. and Janacek, G. (2005). Clustering time series with clipped data. *Machine Learning*, 58(2-3), 151–178.

Barnett, L., Barrett, A.B., and Seth, A.K. (2009). Granger causality and transfer entropy are equivalent for gaussian variables. *Physical review letters*, 103(23), 238701.

Basalto, N., Bellotti, R., De Carlo, F., Facchi, P., Pantaleo, E., and Pascazio, S. (2007). Hausdorff clustering of financial time series. *Physica A: Statistical Mechanics and its Applications*, 379(2), 635–644.

Berndt, D.J. and Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, 359–370. Seattle, WA.

Candanedo, L.M. and Feldheim, V. (2016). Accurate occupancy detection of an office room from light, temperature, humidity and co2 measurements using statistical learning models. *Energy and Buildings*, 112, 28–39.

Corduas, M. and Piccolo, D. (2008). Time series clustering and classification by the autoregressive metric. *Computational statistics & data analysis*, 52(4), 1860–1872.

Darianian, M. and Michael, M.P. (2008). Smart home mobile rfid-based internet-of-things systems and services. In *Advanced Computer Theory and Engineering, 2008. ICACTE'08. International Conference on*, 116–120. IEEE.

Dimpfl, T. and Peter, F.J. (2013). Using transfer entropy to measure information flows between financial markets. *Studies in Nonlinear Dynamics and Econometrics*, 17(1), 85–102.

Duan, G., Suzuki, Y., and Kawagoe, K. (2006). Grid representation of time series data for similarity search. *The institute of Electronic, Information, and Communication Engineer.*

Dunbabin, M. and Marques, L. (2012). Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics & Automation Magazine*, 19(1), 24–39.

Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, 226–231.

Górecki, T. (2014). Using derivatives in a longest common subsequence dissimilarity measure for time series classification. *Pattern Recognition Letters*, 45, 99–105.

Granger, C.W. (1988). Causality, cointegration, and control. *Journal of Economic Dynamics and Control*, 12(2-3), 551–559.

Guo, C., Jia, H., and Zhang, N. (2008). Time series clustering based on ica for stock data analysis. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, 1–4. IEEE.

Gupta, S. and Ray, A. (2007). Symbolic dynamic filtering for data-driven pattern recognition. *Pattern recognition: theory and application*, 17–71.

Hautamaki, V., Nykanen, P., and Franti, P. (2008). Time-series clustering by approximate prototypes. In *2008 19th International Conference on Pattern Recognition*, 1–4. IEEE.

Hirano, S. and Tsumoto, S. (2005). Empirical comparison of clustering methods for long time-series databases. In *Active Mining*, 268–286. Springer.

Jiang, Z., Liu, C., Akintayo, A., Henze, G.P., and Sarkar, S. (2017). Energy prediction using spatiotemporal pattern networks. *Applied Energy*, 206, 1022–1039.

Keogh, E., Lin, J., and Fu, A. (2005). Hot sax: Efficiently finding the most unusual time series subsequence. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, 8–pp. Ieee.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9), 1464–1480.

Lin, J., Keogh, E., Lonardi, S., and Chiu, B. (2003). A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, 2–11. ACM.

Lin, J., Vlachos, M., Keogh, E., Gunopulos, D., Liu, J., Yu, S., and Le, J. (2005). A mpaa-based iterative clustering algorithm augmented by nearest neighbors search for time-series data streams. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 333–342. Springer.

Liu, C., Akintayo, A., Jiang, Z., Henze, G.P., and Sarkar, S. (2018). Multivariate exploration of non-intrusive load monitoring via spatiotemporal pattern network. *Applied Energy*, 211, 1106–1122.

Liu, C., Ghosal, S., Jiang, Z., and Sarkar, S. (2016). An unsupervised spatiotemporal graphical modeling approach to anomaly detection in distributed cps. In *Proceedings of the 7th International Conference on Cyber-Physical Systems*, 1. IEEE Press.

Martini, M., Kranz, T.A., Wagner, T., and Lehnertz, K. (2011). Inferring directional interactions from transient signals with symbolic transfer entropy. *Physical Review E*, 83(1), 011919.

Murtagh, F. and Legendre, P. (2014). Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? *Journal of classification*, 31(3), 274–295.

Oates, T., Firoiu, L., and Cohen, P.R. (2000a). Using dynamic time warping to bootstrap hmm-based clustering of time series. In *Sequence Learning*, 35–52. Springer.

Oates, T., Schmill, M.D., and Cohen, P.R. (2000b). A method for clustering the experiences of a mobile robot that accords with human judgments. In *AAAI/IAAI*, 846–851.

Qin, S.J. (2012). Survey on data-driven industrial process monitoring and diagnosis. *Annual reviews in control*, 36(2), 220–234.

Rabiner, L., Levinson, S., Rosenberg, A., and Wilpon, J. (1979). Speaker-independent recognition of isolated words using clustering techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(4), 336–349.

Ramoni, M., Sebastiani, P., and Cohen, P. (2002). Bayesian clustering by dynamics. *Machine learning*, 47(1), 91–121.

Ratanamahatana, C.A. and Keogh, E. (2005). Multimedia retrieval using time series representation and relevance feedback. In *International Conference on Asian Digital Libraries*, 400–405. Springer.

Saha, H., Liu, C., Jiang, Z., and Sarkar, S. (2018). Exploring granger causality in dynamical systems modeling and performance monitoring. In *2018 IEEE Conference on Decision and Control (CDC)*, 2537–2542. IEEE.

Sarkar, S. and Srivastav, A. (2016). A composite discretization scheme for symbolic identification of complex systems. *Signal Processing*, 125, 156–170.

Schindlerova, K. (2011). Equivalence of granger causality and transfer entropy: A generalization.

Sheikholeslami, G., Chatterjee, S., and Zhang, A. (1998). Wavecluster: A multi-resolution clustering approach for very large spatial databases. In *VLDB*, volume 98, 428–439.

Staniek, M. and Lehnertz, K. (2008). Symbolic transfer entropy. *Physical Review Letters*, 100(15), 158101.

Sun, J. and Bollt, E.M. (2014). Causation entropy identifies indirect influences, dominance of neighbors and anticipatory couplings. *Physica D: Nonlinear Phenomena*, 267, 49–57.

Tan, S.Y., Saha, H., Florita, A.R., Henze, G.P., and Sarkar, S. (2019). A flexible framework for building occupancy detection using spatiotemporal pattern networks. In *2019 American Control Conference (ACC)*, 5884–5889. IEEE.

Ver Steeg, G. and Galstyan, A. (2012). Information transfer in social media. In *Proceedings of the 21st international conference on World Wide Web*, 509–518. ACM.

Vicente, R., Wibral, M., Lindner, M., and Pipa, G. (2011). Transfer entropy—a model-free measure of effective connectivity for the neurosciences. *Journal of computational neuroscience*, 30(1), 45–67.

Wang, W., Yang, J., Muntz, R., et al. (1997). Sting: A statistical information grid approach to spatial data mining. In *VLDB*, volume 97, 186–195.

Yin, S., Ding, S.X., Haghani, A., Hao, H., and Zhang, P. (2012). A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark tennessee eastman process. *Journal of Process Control*, 22(9), 1567–1581.

Zhang, X., Liu, J., Du, Y., and Lv, T. (2011). A novel clustering method on time series data. *Expert Systems with Applications*, 38(9), 11891–11900.