

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

Sin Yong Tan^{*} Homagni Saha^{*} Margarite Jacoby^{**}
Gregor 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).

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 (Saha et al. (2019b,a)), 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 becomes intractably large without dimensionality reduction or model simplification. Data-driven techniques have received attention from industry and academia alike due to their accuracy and scalability. 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 also exists in other fields such as finance (Dimpfl and Peter (2013)) and neuroscience (Vicente et al. (2011)); however, with focus on identifying causal interactions among large-scale CPSs, many time series aspects have not been explored sufficiently.

Time series clustering techniques are extremely useful in managing “state explosions” by reducing the number of variables in multivariate observations. Existing time series clustering algorithms can be divided into six main types: *partitioning-based* as in (Hautamaki et al. (2008)), *hierarchical* as in (Hirano and Tsumoto (2005)), *grid-based* as in (Wang et al. (1997)), *model-based* as in (Corduas and Piccolo (2008)), *density-based* as in (Ankerst et al. (1999)), and *multi-step clustering* as in (Zhang et al. (2011)). We chose to combine an agglomerative hierarchical clustering approach with state estimation at each level, giving rise to a multi-level state estimation problem. While hierarchical methods generally provide better visualization on relatively important features, they do not require specifications of the required number of clusters and are thus flexible. Furthermore, we enhance

[★] This work was authored in part by Anthony R. Florita from 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.

our algorithm's scalability using a repartitioning technique known as symbolic dynamic filtering (SDF).

Contributions:

- (1) Proposed a hierarchical time series clustering technique based on SDF and a novel Granger causality based similarity metric.
- (2) Demonstrated algorithm's performance and robustness for state estimation problems on real data sets.
- (3) Introduced an efficient dimensionality reduction technique for time series which maintains the target variable state estimation accuracy.

2. BACKGROUND

2.1 Symbolic Dynamic Filtering (SDF) based Encoding

SDF 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 discrete symbolic domain (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 . We assume there is a partitioning function $\mathcal{X} : X(t) \rightarrow D$, mapping the continuous elements of $X(t)$ to the discrete elements of D . There are numerous partitioning method used in different literature, such as uniform partitioning, maximum entropy partitioning (MEP), maximally bijective discretization, and statistically similar discretization (Sarkar and Srivastav (2016)). In our work, we mainly used MEP for discretization.

2.2 Symbol Sequence to State Sequence

Time embedding is a technique in the dynamical systems literature to identify key features in time series as a collection of states. In our work, embedding is performed by converting symbol sequence to state sequence. Consider a *symbolic* sequence $X = \{x_1, x_2, \dots, x_n\}$, 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 (also known as "depth" of embedding) in a *state*. Using this parameter, we denote a state at time t with an embedding dimension of k as, $\bar{x}_t^k := \{x_{t-k}, \dots, x_{t-1}, x_t\}$ (i.e., k historical observations from X at time t). Hence, a *state sequence* can be expressed as $\bar{X}^k = \{\bar{x}_1^k, \bar{x}_2^k, \dots, \bar{x}_n^k\}$. Fig. 1 shows an example of *state sequence* generation from a *symbol* sequence with embedding dimension $k = 2$, where each state in the state sequence contains a collection of three symbols, including one current timestep symbol and two historic symbols from previous timesteps.

2.3 Formulation 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. Let *state* $\bar{x}_{t-1}^k = \{x_{t-k-1}, \dots, x_{t-2}, x_{t-1}\}$, where t is the discrete time index.

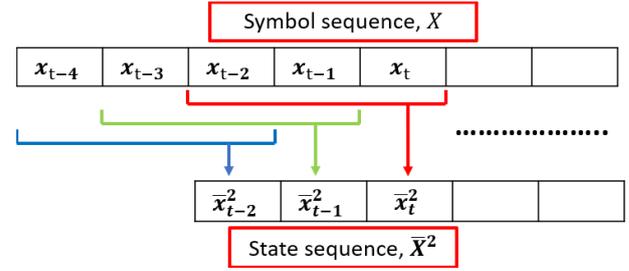


Fig. 1. Demonstrating the process of embedding a symbol sequence X into a state sequence \bar{X}^2 , with each *state* having an embedding dimension $k = 2$.

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 its own k -history. Then variable Y is said to Granger cause variable X (with k lags) if and only if:

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

Granger's formulation was based on vector autoregressive modeling, 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 first express the Shannon entropy for variable X as:

$$H_X = - \sum_n p(x) \log(p(x)) \quad (2)$$

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

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

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

$$H_{XY} = - \sum_{n_x} \sum_{n_y} p(x, y) \log(p(x, y)). \quad (4)$$

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, $\bar{x}_t^k := \{x_{t-k}, \dots, x_{t-1}, x_t\}$ and the state of Y , $\bar{y}_t^k := \{y_{t-k}, \dots, y_{t-1}, y_t\}$. With this setup, transfer entropy for the two systems can be defined as the difference of conditional entropies as follows (Barnett et al. (2009)):

$$\mathcal{T}_{X \rightarrow Y} = H_{Y|\bar{y}^k} - H_{Y|\bar{y}^k \bar{x}^k} \quad (5)$$

There exist efficient methods in literature to calculate both conditional entropies in Eq. 5 using the following equations (using time lag of 1, symbols at $t+1$ rely on states through t):

$$H_{Y|\bar{y}^k} = - \sum_n p(y_{t+1}) \log(p(y_{t+1} | \bar{y}_t^k)) \quad (6)$$

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

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

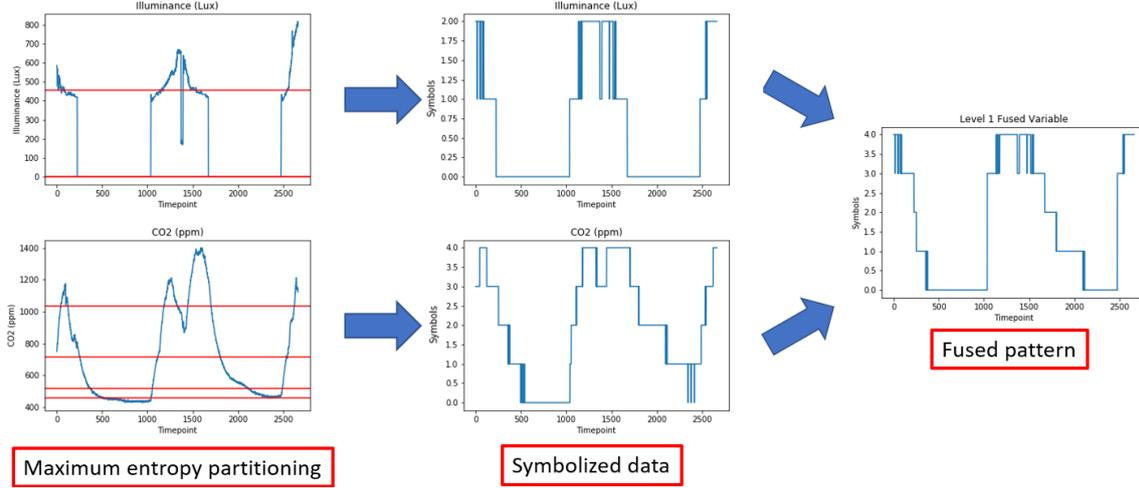


Fig. 2. Flowchart showing the fusion and repartitioning process between illuminance and CO_2 time series data.

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

This equation aligns with the symbolic transfer entropy elaborated in Staniek and Lehnertz (2008), and the equivalence between transfer entropy and Granger causality for Gaussian variables are also proven in Schindler (2011).

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 of variables. Such a joint representation is often known as “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 state 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).

3.2 Granger causality based clustering similarity metric

In our multivariate time series setting, 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.

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 $\mathcal{M} = \binom{n}{2} = \{(1, 2), (1, 3), \dots, (n-1, n)\}$, and n is the number of source variables at that hierarchy. The best fusion combination, indexed as c , was selected from all combinations in \mathcal{M} by using the following rule:

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

After obtaining index c , the 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 through a third variable in the causal chain. In Sun and Bollt (2014), *causation entropy* was defined as follows (aligning their notation with ours):

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

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

Eq. 10 denotes the extra information provided to Z by Y in addition to the information already provided to Z by other sources. Accordingly, Eq. 11 denotes the extra information provided to Z by X in addition to the information already provided to Z by other sources. By comparing Eqs. 3 - 7 with Eqs. 9 - 11, our similarity metric in Eq. 9 can be reinterpreted as the argmin of:

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

In other words, if X and Y are selected for clustering, it implies that X and Y together contribute the highest causation entropy out of all variable combinations in \mathcal{M} .

3.3 Symbol sequence fusion using repartitioning

After selecting the pair of variables X and Y to be clustered, we seek to form a joint representation of these variables. Let the symbol sequence of X be given by $\{x_1, x_2, \dots, x_n\}$ and the symbol sequence of Y be given

by $\{y_1, y_2, \dots, y_n\}$, where n is the length of the symbol sequence. We first form a merged symbol sequence denoted by $\{x_1y_1, x_2y_2, \dots, x_ny_n\}$. 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 and b_y are the number of unique symbols in X and Y respectively. After assigning values to the merged symbol sequence, we 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. 2.

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 $S = \{s_1, s_2, \dots, s_n\}$ where n is the total number of source variables, and n_h is the number of source variables in every hierarchy/level.

Algorithm 1 Hierarchical Time Series Clustering

Require: Symbolized source and target variables.

- 1: Initialize $n_h = n$.
- 2: **while** $n_h > 1$ **do**
- 3: Compute $T_{s_1 \rightarrow Z}, T_{s_2 \rightarrow Z}, \dots, T_{s_{n_h} \rightarrow Z}$
- 4: Combinations $\mathcal{M} = \binom{n_h}{2} = \{m_1, m_2, \dots\}$
- 5: Compute $T_M = \{T_{m_1 \rightarrow Z}, T_{m_2 \rightarrow Z}, \dots\}$
- 6: Determine best fusion pair index c [Eq. 9].
- 7: Fuse selected pair s_X, s_Y and repartition it to desired number of symbols [Section 3.3]
- 8: $S \leftarrow S - \{s_X, s_Y\} + \{s_{XY}\}$
- 9: $n_h \leftarrow n_h - 1$
- 10: **end while**

In term of computational complexity of the algorithm, the bottleneck would be the computation of transfer entropy for *all* possible fusion pairs in line 5. The number of pairs of transfer entropy to be computed is determined in line 4, where the computation complexity could be express as $\mathcal{O}(n \text{ choose } 2) = \mathcal{O}(\frac{n!}{2(n-2)!}) = \mathcal{O}(n^2)$.

4. EXPERIMENTS

4.1 Occupancy (OCC) Data Set

To demonstrate the performance of our algorithm, two open source data sets are used in experiments. The first data set used is the University of California, Irvine's building occupancy detection data set by Candanedo and Feldheim (2016). This data set is comprised of five time series data streams that describe the indoor condition of an office space, and include temperature ($^{\circ}\text{C}$), relative humidity (%), illuminance level (lux), CO_2 (ppm) and humidity ratio (kg-water-vapor/kg-air). 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 used is the OpenEI "Long-term data on 3 office Air Handling Units" (AHU data set). This data set consists of multiple variables that describe 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, $^{\circ}\text{F}$), return air temperature (RAT, $^{\circ}\text{F}$), outside air damper command (OA Damper CMD), cooling valve command (Cool Valve CMD), discharge air temperature (DAT, $^{\circ}\text{F}$), supply fan speed command (Su Fan speed CMD), discharge air static pressure (DA Static P), and return fan speed command (Re Fan Speed CMD).

Target Variable Discretization. Unlike the OCC data set that already have discrete (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 entropy split criterion.

5. RESULTS AND DISCUSSION

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. 3.

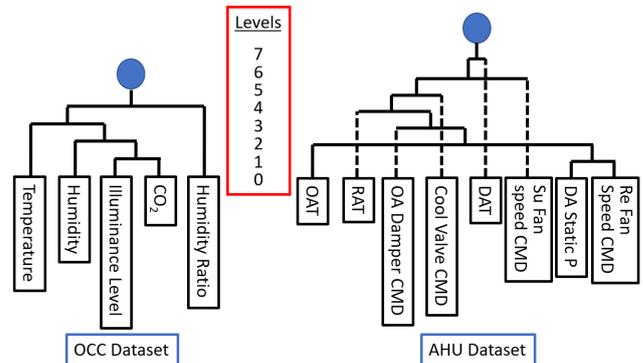


Fig. 3. Clustering process of OCC and AHU data set at different levels from initial n nodes to one supernode.

The root of the clustering trees (Level 0) in Fig. 3 are annotated with all the source variables and two nodes are merged in each of the levels. Some parameters that were used in the symbolic dynamic filtering and state sequence generation are as follows. For the OCC data set, each source variable is partitioned into five symbols with an embedding dimension (depth) of three during state sequence generation. On the other hand, ten symbols are used to partition the variables in the AHU data set, and five historical observations (symbols) are embedded in each state.

5.2 Performance Evaluation

To evaluate the quality of the merged variables, we perform 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. Prediction performance of OCC data and AHU data at each level of 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. Although both data sets are building related, 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, thus root-mean-square error (RMSE) would be a better choice as a measure of difference between the prediction and the true target.

From Table 1, the prediction performance was well maintained around the baseline for both the OCC and AHU data sets despite natural information loss due to dimensionality reduction. In fact, the OCC data set has prediction accuracies 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 starts 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. Another solution is, instead of limiting the similarity metric threshold, the desired number of clusters or “supernodes” formed by this hierarchical algorithm can be pre-defined, where the algorithm will stop as the supernodes formed reach the desired number similar to *k*-means (top-down) clustering.

Fig. 4 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 and 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. 5 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.

In order to verify that the similarity metric can select informational and relevant time series for fusion, we performed

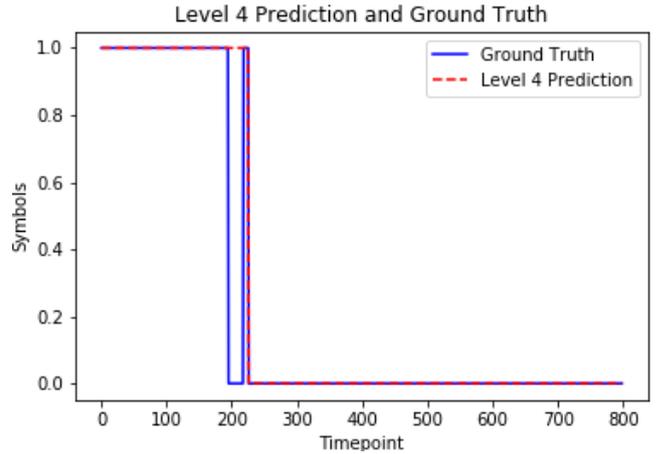


Fig. 4. OCC data set prediction at level 4 (full clustering) of the clustering tree.

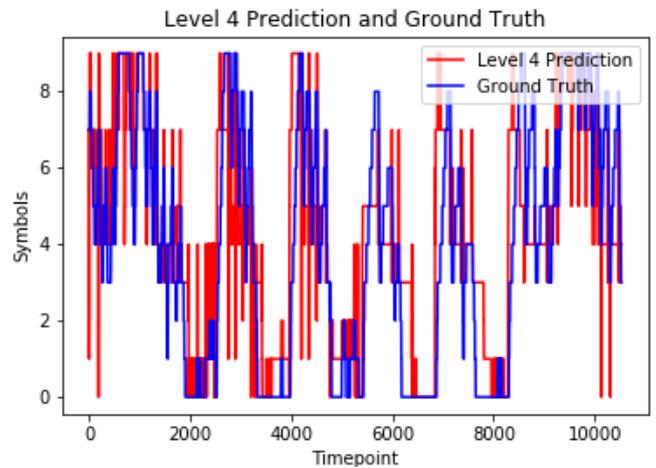


Fig. 5. AHU data set best prediction (at level 4).

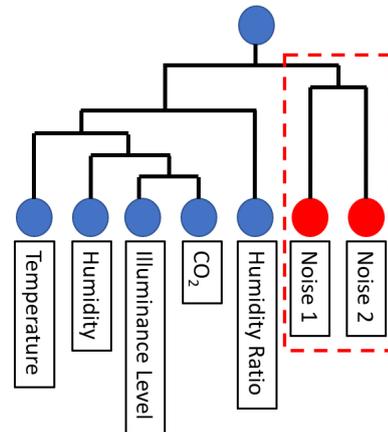


Fig. 6. OCC data set clustering tree with two gaussian white noises (red circles).

an experiment where we introduced multiple random noise terms (standard normal) along with the original time series as input to observe the resulting clustering pattern. Interestingly, the noise variables did not merge with the main cluster in the early levels, but instead formed a cluster of their own, which merged only with the main cluster in the end when there were no other merging options available. This observation is illustrated in Fig. 6, where the noise variables (red circles) are merged in the second-to-last level to form a “supernode” in the red box, before merging with

the main cluster on the left of the tree in the last level. Note the arrangement or the order of source variables at the root does not affect the clustering sequence, as the fusion pair selection is deterministic, with no randomness involved in the process.

6. CONCLUSION

We developed a Granger causality based hierarchical time series clustering algorithm that combines dimensionality reduction with robustification against noise. We also proposed a Granger causality based similarity metric that incrementally clusters pairs of most relevant time series at each clustering level while preserving the predictive power. We showed 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 uninformative time series might negatively impact the state estimation power of the merged variables. To tackle this problem, a dynamic threshold for 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. For faster run-time performance, parallelization can be done on the computation of the transfer entropies in Algorithm 1, which will reduce the run-time significantly.

REFERENCES

- 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.
- 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.
- 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.
- 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.
- Granger, C.W. (1988). Causality, cointegration, and control. *Journal of Economic Dynamics and Control*, 12(2-3), 551–559.
- 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.
- 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.
- 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.
- Saha, H., Tan, S.Y., Jiang, Z., and Sarkar, S. (2019a). Learning state switching for multi sensor integration. In *2019 Indian Control Conference (ICC)*. IEEE.
- Saha, H., Venkataraman, V., Speranzon, A., and Sarkar, S. (2019b). A perspective on multi-agent communication for information fusion. *arXiv preprint arXiv:1911.03743*.
- Sarkar, S. and Srivastav, A. (2016). A composite discretization scheme for symbolic identification of complex systems. *Signal Processing*, 125, 156–170.
- Schindler, K. (2011). Equivalence of granger causality and transfer entropy: A generalization. *Applied Mathematical Sciences*, 5, 3637 – 3648.
- 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.
- 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.
- 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.