Submitted to *INFORMS Journal on Computing*

Supplemental Materials to Optimal Frameworks for Detecting Anomalies in Sensor-Intensive Heterogeneous Networks

Ramin Moghaddass[a,b], Yongtao Guan[b]
[a]Department of Industrial Engineering, University of Miami, Coral Gables, USA
[b]Department of Management Science, Miami Herbert Business School, University of Miami, Coral Gables, USA

Appendix A: Proof of Remark 2.

From the definition of the conditional probability and the constant prior terms ($\Pr(x_{nm}=1)$), we have

$$\Pr(x_{nm}=1|y_{n'j};n'\in R_n)\propto\frac{\Pr(y_{n'j};n'\in R_n|x_{nm}=1)}{\Pr(y_{n'j};n'\in R_n)}. \tag{1}$$

Given the binary nature of each node's anomaly status and each sensor attribute's output, we can show that the value of binary sensor attribute j at node n follows a Bernoulli distribution as follows:

$$y_{nj}\sim \text{Ber}\big(\sum_{m=1}^{M}(\alpha_{jm}-\beta_j)\,x_{nm}(t)+\beta_j\big),$$

where α_{jm} can be interpreted as the true positive rate (TPR) for attribute j and the mth anomaly and β_j as the false positive rate (FPR) for attribute j, that is

$$\alpha_{jm}=\Pr(y_{nj}=1|x_{nm}=1),\quad \beta_j=\Pr(y_{nj}=1|\sum_{m=1}^{M}x_{nm}=0),\quad \forall n\in\{1,...,N\},j\in\{1,...,J\},m\in\{1,...,M\}.$$

Based on the propagation of anomalies in subgraph R_n, the nominator of Eq. (25) can be re-written as

$$\Pr(y_{n'j};n'\in R_n|x_{n'm}=1;n'\in R_n)=\prod_{n'\in R_n}\Pr(y_{n'j}|x_{n'm}=1)$$
$$=\prod_{n'\in R_n}\alpha_{jm}^{y_{n'j}}\,(1-\alpha_{jm})^{1-y_{n'j}}=\frac{\alpha_{jm}^{\sum_{n'\in R_n}y_{n'j}}(1-\alpha_{jm})^{\sum_{n'\in R_n}\mathbb{1}\{s_{n'j}=1\}}}{(1-\alpha_{jm})^{\sum_{n'\in R_n}y_{n'j}}}.$$

The right hand side of the above equation depends only on $\sum_{n'\in R_n}y_{n'j}$ and $\sum_{n'\in R_n}\mathbb{1}\{s_{n'j}=1\}$ and constant parameter α_{jm}. We now show that the denominator of Eq. (25) also depends on these elements. For notational convenience, we show the proof for $M=1$, that is when only one type of anomaly can occur in the network. The proof can be extended to a general M. By denoting P_n as the random variable representing the fraction of anomalous nodes in the subgraph R_n as

$$P_n = \frac{\sum\limits_{n';n'\in R_n, j\in S_{n'}} x_{n'1}}{\sum\limits_{n'\in R_n} \mathbb{1}\{s_{n'j}=1\}},$$

and using the law of total probability, we have

$$\Pr(y_{n'j}; n'\in R_n) = \sum_{p\in\{\frac{i}{\sum_{n'\in R_n}\mathbb{1}\{s_{n'j}=1\}}; i=0,1,\ldots,\sum_{n'\in R_n}\mathbb{1}\{s_{n'j}=1\}\}} \Pr(y_{n'j}; n'\in R_n|P_n=p)\Pr(P_n=p).$$

The first element of the RHS of the above equation can be computed by marginalization as follows:

$$\sum_{x_{n'1};n'\in R_n} \Pr(y_{n'j}, x_{n'1}; n'\in R_n|P_n=p) = \sum_{x_{n'1};n'\in R_n}\prod_{n'\in R_n} \Pr(y_{n'j}|x_{n'1})\Pr(x_{n'1}|P_n=p).$$

Now by assuming that the prior $\Pr(x_{n'1}=1|P_n=p)$ equals p for all nodes in the subgraphs and prior $\Pr(P_n=p)$ can be estimated from historical data (or theoretically calculated), the right hand side of the above equation can be simplified as

$$\begin{aligned}\prod_{n'\in R_n}\Big(\sum_{x_{n'1};n'\in R_n}\Pr(y_{n'j}|x_{n'1})\Pr(x_{n'1}|P_n=p)\Big) &= \prod_{n'\in R_n}\Big[p(\alpha_{j1}-\beta_j)+\beta_j\Big]^{y_{n'j}}\Big[1-\big(p(\alpha_{j1}-\beta_j)+\beta_j\big)\Big]^{1-y_{n'j}}\\ &= \frac{\Big[p(\alpha_{j1}-\beta_j)+\beta_j\Big]^{\sum_{n'\in R_n} y_{n'j}}\Big[1-\big(p(\alpha_{j1}-\beta_j)+\beta_j\big)\Big]^{\sum_{n'\in R_n}\mathbb{1}\{s_{n'j}=1\}}}{\Big[1-\big(p(\alpha_{j1}-\beta_j)+\beta_j\big)\Big]^{\sum_{n'\in R_n} y_{n'j}}},\end{aligned}$$

which also depends on the two elements $\sum\limits_{n'\in R_n} y_{n'j}$ and $\sum\limits_{n'\in R_n}\mathbb{1}\{s_{n'j}=1\}$ and constants α_{j1}, β_j, and p. This completes the proof.

Appendix B: Training the Network Anomaly Detection Framework

Algorithm 1 Model Training for the Anomaly Detection Framework

Input: Network Topology Parameters, Network Sensor Data ($\boldsymbol{y}_n(t) = [y_{n1}(t), ..., y_{nJ}(t)]$), and Network Nodes Anomaly Labels ($\boldsymbol{x}_n(t) = [x_{n1}(t), ..., x_{nM}(t)]$) for $n \in \{1, ..., N\}$ and $t \in \{1, ..., T_1\}$, where T_1 is the number of time instances in the training data, N is the number of network samples, and M is the number of anomaly types

- Network Structure and Sensor Data Aggregation

for $n = 1$ to N **do**

- Extract sensor attributes in subgraph R_n based on the anomaly propagation set of node n.

for $j = 1$ to J **do**

for $t = 1$ to T_1 **do**

- Use the aggregation functions in Eqs. (2)-(3) in the article to obtain $\bar{y}_{nj}(t)$ and $\bar{s}_{nj}(t)$.

end for

end for

end for

- Autoencoder Training

- Define the set of hyperparameters for the grid search.
- Define other parameters of the Autoencoder. Define the validation set and the evaluation metric.
- Conduct the grid search to find the optimal hyperparameters. Use $[\bar{\boldsymbol{y}}_n(t), \bar{\boldsymbol{s}}_n(t)]$ for all $n \in \{n, ..., N\}$ and $t \in \{1, ..., T_1\}$ at both input and output layers where $\bar{\boldsymbol{y}}_n(t) = [\bar{y}_{n1}(t), ..., \bar{y}_{nJ}(t)]$ and $\bar{\boldsymbol{s}}_n(t) = [\bar{s}_{n1}(t), ..., \bar{s}_{nJ}(t)]$.
- Utilize the optimal encoder to transform the entire data set to a D-dimensional embedding space where D is the number of neurons in the bottleneck layer.
- The transformed data in the embedding space are recorded as $\boldsymbol{h}_n(t) = [h_{n1}(t), ...h_{nD}(t)]$ for $n \in \{1, ..., N\}$ and $t \in \{1, ..., T_1\}$.

- MLP Training

- Define the set of hyperparameters for the grid search.
- Define other parameters of the MLP. Define the validation set and the evaluation metric.
- Conduct the grid search to find the optimal set of hyperparameters for the MLP. Use $\boldsymbol{h}_n(t) = [h_{n1}(t), ...h_{nD}(t)]$ and $\boldsymbol{x}_n(t) = [x_{n1}(t), ..., x_{nM}(t)]$ for $n \in \{1, ..., N\}$ and $t \in \{1, ..., T_1\}$ in the input and output layers, respectively.

Output:

- Trained Autoencoder and Trained MLP.
- By repeating all of the above steps and considering the entire network as one unit as discussed in Section 5.5 of the article, we can train MLP_0 as well.

Appendix C: Proof of Remark 5.

It is clear that if an anomaly originates at node n, then all nodes inside its anomaly propagation set (i.e., R_n) are impacted by the anomaly. If we define a binary variable q_n to refer to whether node n and all nodes in set R_n are under an anomaly condition, then $\sum_{n=1}^{N} q_n$ represents the number of subsets

with an anomaly. Since we want to maximize the number of such subsets, then the objective function becomes Eq. (15) in the article. To avoid overlapping subsets and disjointed subsets of anomalies, we need to make sure that no node is in more than 1 activated subset, where activated subset means the corresponding q is 1. To do so, we define an indicator function $\mathbb{1}\{s \in R_n\}$ to denote whether node s is inside the anomaly propagation set of node n. To make sure there is no overlap between activated subsets, we need to make sure the sum of $\sum_{n=1}^{N} \mathbb{1}\{s \in R_n\} q_n$ over n is less than or equal to 1 for node s, which gives the constraint given in Eq. (16) in the article. This completes the proof. In the extreme case where $R_n = \{n\}$ (that is when the anomaly cannot propagate to other nodes), then q_n=1 for all n and $Q^* = N$. Thus Q^* cannot be larger than N.

Appendix D: Algorithm for Real-Time Anomaly Detection

Algorithm 2 Real-Time Anomaly Detection at Time t using $\mathcal{M}_0$, $\mathcal{M}_1$, and $\mathcal{M}_2$

Input:

- Trained MLP and Trained Autoencoder from **Algorithm 1**.
- Network Topology Parameters and Network Sensor Data ($\boldsymbol{y}_n(t) = [y_{n1}(t), ..., y_{nJ}(t)]$) for $n \in \{1, ..., N\}$.

Model 0 ($\mathcal{M}_0$) - Node-level Anomaly Detection

for $n = 1$ to N **do**

- Apply network aggregation with the aggregation functions, dimensionality reduction with the trained autoencoder, and anomaly classification with the trained MLP.
- Output the probability distribution of anomalies ($\gamma_{nm}(t)$) from the MLP output layer and the most likely anomaly label of all nodes ($v_n(t)$) for $n \in \{1, ..., N\}$ and $m \in \{1, ..., M\}$.

end for

Models 1 and 2 ($\mathcal{M}_1$-$\mathcal{M}_2$) - Network-Level Anomaly Detection

- Apply the Prescreening step using the trained binary classifier MLP_0 (**Optional**).

if The network is found to be anomalous from MLP_0 **then**

- Apply the Optimization Initialization steps using Remarks 3-6.
- Calculate anomaly scores for $n \in \{1, ..., N\}$ and $m \in \{1, ..., M\}$.
- Compute the probabilistic confusion matrix $\mathbf{P} = [p_{i,j}]$ to be used in $\mathcal{M}_2$.
- Setup optimization Models $\mathcal{M}_1$ and $\mathcal{M}_2$.
- Find the optimal solutions of $o_{nm}(t)$ for all $n \in \{1, ..., N\}$ and $m \in \{1, .., M\}$.
- Use the anomaly propagation sets to find the predicted status of each node ($v_n(t), n \in \{1, ..., N\}$).

else

- The network has no anomalous nodes. That is

$o_{nm}(t) = 0, v_n(t) = 0, \quad n \in \{1, ..., N\}, m \in \{1, .., M\}$.

end if

Output:

- The optimal values of $o_{nm}(t)$ and $v_n(t)$ for all $n \in \{1, ..., N\}$ and $m \in \{1, .., M\}$ for $\mathcal{M}_0$, $\mathcal{M}_1$, and $\mathcal{M}_2$.
- The optimal values of $x_{nm}(t)$ can be found from Eq. (1).

Appendix E: Network Datasets

This section summarizes some of the data used in the numerical experiments. A total of 5 network datasets are included in the experiments. Below, we provide a brief description of each dataset. The datasets (recorded as R files) can be downloaded from https://figshare.com/s/5700482fb14d29b616f5.

E.1. Network Dataset 1 (used in Section 6.1)

The topology of the network from this dataset is from a 1,138 bus power network provided in a Network Repository (Rossi and Ahmed 2015). We generated random sensors and anomalies as discussed in the paper. The dataset includes random sensor data and the true locations of anomalies and their type.

E.2. Network Dataset 2 (used in Section 6.1)

This network dataset is from the the U.S. Reference Network Model (NREL 2016). We selected a part of this topology with 10,089 nodes (only selected device nodes under node HVMVSubstation-P12U-nSSEE1 69) and then randomly selected some nodes to be the hosts of sensors. For this dataset, the sensor data and true locations of anomalies and their type are reported. Nodes are renamed from 1-10,089.

E.3. Network Dataset 3 (used in Section 6.5)

The network topology is obtained from (Georgescu 2012), which is a small network with 43 nodes and 78 edges. The adjacency matrix is used to build the graph topology and anomaly propagation paths. There are two datasets: one for full propagation (g1) and one for partial propagation (g2).

E.4. Network Dataset 4 (used in Section 6.5)

This dataset has 130 nodes and 180 edges and is related to the well-known Battle of the Water Sensor Networks (BWSN) dataset (Ostfeld et al. 2008). The adjacency matrix is used to build the graph topology and define the anomaly propagation paths. Sensor data for two scenarios of full and partial propagation are simulated separately. There are two datasets: one for full propagation (g1) and one for partial propagation (g2).

E.5. Network Dataset 5 (used in Section 6.6)

This dataset included R data files for randomly generated graphs of 6 types. Each file's name included the degree, number of nodes, and the type of the graph. It should be noted that nodes with no connection and sensor data are not shown. Also, only sensor data with positive sensor values are recorded (i.e., sensor values 0 are not recorded to save memory). Each data file includes sensor data, edge list, and true locations of anomalies.

References

S.-C. Georgescu. Hbmoa applied to design a water distribution network for a town of 50000 inhabitants. *UPB Scientific Bulletin, Series D: Mechanical Engineering*, 74(1):91–102, 2012.

NREL. Bay area synthetic network, https://egriddata.org/dataset/bay-area-synthetic-network, date last accessed 15-april-2021. 2016.

A. Ostfeld, J. Über, E. Salomons, J. Berry, W. Hart, C. Phillips, J.-P. Watson, G. Dorini, P. Jonkergouw, Z. Kapelan, F. di Pierro, S.-T. Khu, D. Savic, D. Eliades, M. Polycarpou, S. Ghimire, B. Barkdoll, R. Gueli, J. Huang, E. McBean, W. James, A. Krause, J. Leskovec, S. Isovitsch, J. Xu, C. Guestrin, J. VanBriesen, M. Small, P. Fischbeck, A. Preis, M. Propato, O. Piller, G. Trachtman, Z. Yi Wu, and T. Walski. The battle of the water sensor networks (bwsn): A design challenge for engineers and algorithms. *Journal of Water Resources Planning and Management*, 134(6):556–568, 2008.

R. A. Rossi and N. K. Ahmed. The network data repository with interactive graph analytics and visualization. In *29 AAAI Conference on Artificial Intelligence*, 2015.