A Probabilistic Framework to Node-level Anomaly Detection in Communication Networks

Batiste Le Bars*† Argyris Kalogeratos*

*Center of Applied Maths, ENS Cachan, CNRS, University Paris-Saclay, France
†Sigfox R&D, Paris, France
Emails: {lebars, kalogeratos}@cmla.ens-cachan.fr

***Abstract*—In this paper we consider the task of detecting abnormal communication volume occurring at node-level in communication networks. The signal of the communication activity is modeled by means of a *clique stream*: each occurring communication event is instantaneous, and activates an undirected subgraph spanning over a set of equally participating nodes. We present a probabilistic framework to model and assess the communication volume observed at any single node. Specifically, we employ non-parametric regression to learn the probability that a node takes part in a certain event knowing the set of other nodes that are involved. On the top of that, we present a concentration inequality around the estimated volume of events in which a node could participate, which in turn allows us to build an efficient and interpretable anomaly scoring function. Finally, the superior performance of the proposed approach is empirically demonstrated in real-world sensor network data, as well as using synthetic communication activity that is in accordance with that latter setting.**

***Index Terms*—Anomaly detection, probabilistic models, communication networks, sensor networks, internet-of-things, link streams, graph signals.**

I. Introduction

Monitoring the activity in communication networks has become a popular area of research and particular attention has been paid to detection tasks such as spotting events or anomalies. An effective way to represent the communication activity is via a dynamic graph where the entities are considered to be nodes, and each communication event (or more simply *event*) to be represented by a set of connecting edges that appear at a specific time interval. Multiple occurring events over time may be seen as a *link stream* [1] with fast creation and deletion of edges. The use of this representation is mainly motivated by the fact that, in reality, content-specific features of the communicated messages are usually kept undisclosed so as to preserve privacy. Consequently, most studies on activity monitoring merely deal with linkage information, i.e. who communicated with whom and at which time; the body of work on anomaly detection is not an exception.

The anomaly detection task on graph-related activity can refer to the node-, the subgraph-, or the whole graph-level [2]. To the best of our knowledge, the existing methods consider *time-aggregated* representations of the dynamic graph. It has been proposed to work with time-series of static graphs, each of them summarizing the link stream during a time interval. In other words, each edge weight of a static graph is a function of the number of events occurring between two nodes during that time interval. Modeling the weights' evolution with counting processes [3, 4] is among the standard approaches. The main drawback of any aggregated representation is that it neglects events that involve more than two nodes (e.g. multiple receivers). Besides, a common limitation of the existing literature is the assumption that the communication volume is generated by a stationary underlying distribution.

In this work, we focus on the detection of *abnormal communication volume at node-level*, which is particularly interesting as a change in the behavior of a node may reveal various types of abnormality (e.g. account hack, antenna breakdown, etc.). We put forward a *content-agnostic* approach supposing access solely to the linkage information observed at each event, that is the set of the involved nodes. The conceptual novelty of our approach is that, contrarily to our predecessors that use time-aggregated representations, we model the activity as a *clique stream*. We track each event independently, we consider it to be instantaneous and thus to activate an undirected subgraph spanning over the set of equally participating nodes, i.e. there are no special roles such as sender and receivers. Hence, we can represent each event with a binary *fingerprint* indicating the involved nodes. Subsequently, we propose to statistically model and infer the probability that a node takes part in an event, knowing the observed event fingerprint indicating the other participating nodes. The assumption is that there is a pattern in the fingerprints of the events in which a node participates. This pattern results from the underlying network structure since it is natural for subsets of neighboring nodes to participate frequently together in events.

This modeling allows us to derive confidence levels for the communication volume to which a node participates in a time interval. Our detection approach has two strong aspects. First, it allows the time-series of the node's communication volume to be non-stationary, since it only assumes regularity in the corresponding event fingerprints. Specifically, knowing the fingerprint of an event for all nodes but for a reference node, then the conditional probability that this node takes part in that event is constant over time, whereas the marginal probability that the node could participate in the event is not necessarily constant. Second, the anomaly score that our approach outputs is easily interpretable as it is simply based on the prediction error of a regression function.

II. Related Work

In the literature, the existing detection methods for abnormal node communication volume mostly analyze a time-aggregated representation of the actual dynamic graph of communication activity. This implies a time-series of static graphs $\{A_t\}_{t=1}^{T}$, where $A_t \in \mathbb{R}^{N \times N}$ is the weighted adjacency matrix representing all the shared communication events between pairs of nodes at in the time interval $t \in \{1,...,T\}$, and N is the total number of nodes in the network. Most methods do not consider self-edges and therefore require each A_t to have zero diagonal. Since these methods consider node-to-node communication events, note that the weighted degree of a node according to A_t gives also the total number of events observed at a node in the time interval t. The multivariate time-series of the total number of events occurring in the network over time can be written as $\{M_t\}_{t=1}^{T}$. This is the variable of our interest which we would like to know when it gets abnormal values.

A feature-based approach for detecting anomalies in such time-series of graphs, is to compute several graph features for each A_t, such as the node degree or centrality, and then apply standard anomaly detection techniques on the derived multivariate time-series of these features [5, 6, 7]. More generally, the literature of anomaly detection in time-series of graphs varies in three aspects:

- *Availability of data labels*: Semi-supervised (access to a dataset of *normal* system operation) [8, 9, 10] or unsupervised (no label available) [11, 12].
- *Type of the utilized method*: Probabilistic model-based [3, 4, 13, 14, 15, 16, 17], distance based [11], decomposition-based [18, 19], compression-based [12], etc.
- *Scale of abnormality*: Node/Edge-level [3, 20], subgraph-level [17], or whole network-level [4].

The reader should refer to [2] for a more detailed survey on anomaly detection in dynamic graphs.

As in [3, 21, 22], our work assumes a semi-supervised setting and proposes a model-based approach for node-level anomaly detection. Moreover, as in [4], graph edges are considered to be undirected, and each event to be shared by two or more nodes without distinguishable roles (e.g. sender and receivers).

In [1], the *link stream* framework is presented for the representation of a dynamic graph as a stream where edges are being created and removed. Therein, the nodes are assumed to be fixed and the dynamics affect only the edges between them. An edge is characterized by a triplet (S,u,v) noting two communicating nodes u, v, and a time interval S which is not necessarily continuous (may even be a union of non-contiguous time intervals). In this work, we adopt this stream framework. In particular, as we will see, we represent the activity as a clique stream, and S is always a finite union of singletons as edges appear instantaneously.

III. Model Description and Methodology

A. The model

Let a communication network have N inter-connected entities, referred to as nodes. In terms of notation style, we differentiate a random from an observed variable (respectively vector) with uppercase and lowercase (respectively bold) letters. Moreover, let $|\cdot|$ denote the size of the input set.

Definition 1. *(Communication event): A communication event $e = (\tau_e, X_e)$ is denoted by a tuple of $N+1$ elements, which contains the timestamp τ_e at which the event occurred and its fingerprint X_e.*

Definition 2. *(Event fingerprint): The fingerprint of an event e is an N-dimensional binary vector $X_e \in \{0,1\}^N$, where $X_e^{(j)} = 1$ if node j is involved in the event, and 0 otherwise.*

Note that the involvement of a node in an event implies its participation regardless its communication role (e.g. sender or receiver). From a probabilistic point of view, a fingerprint follows a multivariate Bernoulli distribution. From a graph point of view, we can see that each event creates a *clique* with all the involved nodes (see an example in Fig. 1). Formally, a clique is defined as a subset of nodes of the graph that are all pairwise adjacent. Therefore, we regard the communication activity as a *clique stream*, and each clique appears instantaneously as events have no duration.

Definition 3. *(Event stream): An event stream $\mathcal{S} = \{(\tau_s, X_s)\}_{s=1}^{n}$ is a sequence of $n \triangleq |\mathcal{S}|$ communication events each creating a clique among the involved nodes. We write as $\mathcal{S}_t \subset \mathcal{S}$ the sub-stream with the events that occurred in a certain time interval t, and $n_t \triangleq |\mathcal{S}_t|$.*

Assumption 1. *The communication events are considered to be independent. The total number of events n recorded during an event stream $\mathcal{S}$ is considered to be deterministic.*

Let us consider a time interval t and the associated event stream $\mathcal{S}_t$ consisting of its n_t recorded events. Let the event realizations be denoted by $\{\mathbf{x}_i\}_{i=1}^{n_t}$, where $\forall i, \mathbf{x}_i \in \{0,1\}^N$. Also, let $M_t^{(j)} = \sum_{i=1}^{n_t} X_i^{(j)}$ be the number of events recorded at node $j \in \{1,...,N\}$ over the time interval t.

For a given node j and time interval t, the goal of our method is to be able to decide if the volume of events in which that node participates is abnormal. To solve this problem, the main idea is to provide confidence levels for $M_t^{(j)}$ based on the fingerprints collected from events of the neighboring nodes. This way, an anomaly can be simply spotted whenever the observed value of $M_t^{(j)}$ lies out of the confidence level.

Definition 4. *(Conditional probability function): Let $\mathbf{x}^{(-j)}$ be the fingerprint of the event X that indicates the participation of all nodes except from node j. Then, we define as $\eta_j^*(\mathbf{x}^{(-j)})$ the probability that node j participates in the event X, provided the fingerprint $\mathbf{x}^{(-j)}$:*

$$\eta_j^*(\mathbf{x}^{(-j)}) \triangleq \mathbb{P}(X^{(j)}=1|X^{(-j)}=\mathbf{x}^{(-j)}) \qquad (1)$$

$$= \mathbb{E}\left[X^{(j)}|X^{(-j)}=\mathbf{x}^{(-j)}\right]. \qquad (2)$$

Knowing the fingerprint over all the other nodes allows us to express the behavior of node j as a Bernoulli random variable:

$$X^{(j)} \sim \mathcal{B}\left(\eta_j^*(\mathbf{x}_i^{(-j)})\right). \qquad (3)$$

Concerning a sub-stream $\mathcal{S}_t$ and the number of events recorded at node j therein, we can note that $M_t^{(j)}$ is a sum of Bernoulli distributions and, thus, we can use concentration inequalities [23], such as Chernoff's or Hoeffding's [24], to derive confidence levels. Below, we apply the *bilateral Hoeffding's inequality* to our case:

$$\mathbb{P}\left(\left|M_t^{(j)}-\mu^*\right| \geq \varepsilon \;\middle|\; \forall i=1\dots n_t, \quad X_i^{(-j)}=\mathbf{x}_i^{(-j)}\right) \leq 2\exp\left(-\frac{2\varepsilon^2}{n_t}\right) \qquad (4)$$

with $\mu^* = \mathbb{E}\left[M_t^{(j)}|X^{(-j)}=\mathbf{x}^{(-j)}\right] = \sum_{i=1}^{n_t}\eta_j^*(\mathbf{x}_i^{(-j)})$. Using this inequality and, as mentioned earlier, knowing the event fingerprints of all other nodes, we have with probability at least $1-\delta$:

$$\left|M_t^{(j)}-\mu^*\right| \leq \sqrt{\frac{n_t\log(2/\delta)}{2}}. \qquad (5)$$

This equation provides, with high probability (as δ is close to 0), a good confidence interval for $M_t^{(-j)}$.

B. Methodology

Suppose we observe the sub-stream $\mathcal{S}_t$ and the associated n_t fingerprints. Let $x_i^{(j)}$ be the observed version of $X_i^{(j)}$ indicating if node j participated in the event i or not, and $m_t^{(j)} = \sum_{i=1}^{n_t} x_i^{(j)}$ be the observed version of $M_t^{(j)}$. If we suppose the access to η_j^* (i.e. the true conditional probability for node j), then an intuitive *anomaly score* for the $m_t^{(j)}$ is:

$$\rho_t^j = 2\exp\left(-\frac{2\,(m_t^{(j)}-\mu^*)^2}{n_t}\right). \qquad (6)$$

This score is obtained by replacing ε with $(m_t^{(j)}-\mu^*)$ in the right-hand side of Eq. 4. Relating to the statistical hypothesis testing theory, this score can be seen as an upper bound on the p-value. Then, a threshold α can be set, conventionally $0 < \alpha \leq 0.05$, to detect anomalies. More specifically, an anomaly is detected when $\rho_t^j < \alpha$. Note that this method is equivalent to replacing δ with the chosen threshold value, and then checking if $(m_t^{(j)} - \mu^*)$ falls out of the confidence interval in Eq. 5. Bear also in mind that, since this method provides an upper bound on the p-value, it is in fact more conservative than in the standard statistical testing. Indeed, the confidence intervals built with Eq. 5 are larger than the confidence intervals that correspond to probability exactly equal to α.

In practice, we cannot have access to the true conditional probability functions $\eta_j^*(\cdot)$ which need to be estimated. To this end, we suppose that we have access to a training data stream $\mathcal{S}_0 = \{(\tau_i^0, X_i^0)\}_{i=1}^{n_0}$ which is an event stream recorded at times of normal communication behavior for all nodes. With our definition of $\eta_j^*(\cdot)$ (Definition 4), the estimation problem refers to the task of estimation of conditional probabilities. However, since we deal with a Bernoulli random variable, the problem actually becomes a regression of the unknown function $\eta_j^* : \{0,1\}^{N-1} \mapsto [0,1]$, which can be performed using the previous normal dataset $\mathcal{S}_0$.

In this work we do not discuss the regression procedure, but we still need to note that non-parametric methods do seem suitable. Indeed, the estimation of the conditional distributions for every possible combination of fingerprints would lead to the estimation of $N(2^{N-1}-1)$ parameters. Note also that the Binary Tree or Random Forest regression algorithms seem well-adapted to this setting since the explanatory variables are binary. Let $\widehat{\eta_j}(\cdot) := \widehat{\eta_j}(\cdot; X_1^0, \dots, X_{n_0}^0)$ be our regressor. The first anomaly detection method one can think of is the simple ***'plug-in'* method**:

- fix δ;
- replace η_j^* by $\widehat{\eta_j}$ in Eq. 5;
- use Eq. 5 to obtain confidence levels for $M_t^{(j)}$.

Remark. In practice, fixing δ is not trivial and simply taking a value below 0.05 could lead to bad results. One way to fix δ is via cross-validation on the training stream. To do so, one should fix an acceptable false positive rate (e.g. a standard value is 0.05), then via cross-validation find the value of δ that generates a false positive rate lower than that fixed value.

However, Eq. 4 is not true for the estimated version of η_j^* and we must provide a concentration inequality around the estimated expectation $\widehat{\mu} = \sum_{i=1}^{n_t}\widehat{\eta}_j(\mathbf{x}_i^{(-j)})$. In the following subsection, we give an asymptotic concentration inequality around our predicted number of shared events.

C. Model-free prediction intervals

Theorem 1. *Let $\mathcal{S}_0$ be the training (normal) event stream for which we assume that $\forall i=1,\dots,n_0$, $X_i^0 \underset{i.i.d.}{\sim} \mathbb{P}_{X^0}$. Let $\mathcal{S}_t$ be another stream for which the distribution $\mathbb{P}_X$ may be different but having the same support. Assume that both distributions have the same conditional probability function (Definition 4). Assume our estimator $\widehat{\eta_j}$ is weakly consistent* [25]*, and $\forall i = 1,\dots,n_0$,*

$$\max_{x_i, x_i' \in \{0,1\}^{N-1}} |\widehat{\eta_j}(x;\ x_1,\dots,x_i,\dots,x_{n_0}) - \widehat{\eta_j}(x;\ x_1,\dots,x_i',\dots,x_{n_0})| \leq \kappa(n_0), \qquad (7)$$

where, κ tends to 0 when n_0 tends to infinity such that $n_0\kappa^2(n_0) \underset{n_0\to\infty}{\longrightarrow} 0$. Then, we have :

$$\lim_{n_0\to\infty}\mathbb{P}\left(\left|M_t^{(j)} - \sum_{i=1}^{n_t}\widehat{\eta_j}(X_i^{(-j)};\ X_1^0,\dots,X_{n_0}^0)\right| > s\right) \leq \min_{k\in[0,s]}\left\{2\exp\left(-\frac{2k^2}{n_t}\right) + 2\exp\left(-\frac{(s-k)^2}{2n_t}\right)\right\}. \qquad (8)$$

Proof. A sketch follows; the complete proof is provided in the Appendix. The successive use of the triangle, the Cauchy-Schwarz and the Jensen inequalities allows us to upper-bound $\Big|M_t^{(j)} - \underbrace{\sum_{i=1}^{n_t} \widehat{\eta}_j(X_i^{(-j)};\ X_1^0,\dots,X_{n_0}^0)}_{=\widehat{\mu}}\Big|$ by:

$$\underbrace{\Big|M_t^{(j)} - \mu^*\Big|}_{(i)} + \underbrace{\Big|\mu^* - \widehat{\mu} - \mathbb{E}\Big[\mu^* - \widehat{\mu}\Big]\Big|}_{(ii)} + n_t \underbrace{\mathbb{E}\Big[(\eta^*(X) - \widehat{\eta}(X))^2\Big]}_{(iii)}.$$

Since (iii) tends to 0 as n_0 tends to infinity, due to the consistency assumption, we can bound the left-hand side of inequality (8) by:

$$\min_{k\in[0,s]} \left\{ \mathbb{P}\Big((i) > k\Big) + \lim_{n_0\to\infty} \mathbb{P}\Big((ii) > s - k\Big) \right\}.$$

Applying Hoeffding's inequality on the first element of the sum, and McDiarmid's inequality on the second one, leads to the final result.

□

Remark. Replacing k by $\frac{s}{3}$ in the final inequality given by Theorem 1, we obtain:

$$\limsup_{n_0\to\infty} \mathbb{P}\left(\left| M_t^{(j)} - \sum_{i=1}^{n_t} \widehat{\eta}_j(X_i^{(-j)};\ X_1^0,\dots,X_{n_0}^0) \right| > s \right) \le 4\exp\left(-\frac{2s^2}{9n_t}\right). \tag{9}$$

Remarks on Theorem 1. First of all, as mentioned in the first part of the theorem, the training and test event streams may follow different probability distributions. This is very interesting since, in practice, $M_t^{(j)}$ is a non-stationary time-series: i.e. proportion of events in which a node is involved in is not stationary over time. However, we assume that, while in normal state, the probability that a node participates in an event, knowing the participation of the other nodes, does not change over time. From the network viewpoint, this means that the underlying graph structure, on which the events are dynamically created, does not change in that time as well.

Therefore, provided that all hypotheses are verified, we test whether the η function has changed between the training and the test event streams; we test the *stationarity of the conditional distributions*. Falling out of the confidence intervals (built with Eq. 8 or (9)) would indicate a significant change in the conditional probability. Consequently, a property of this method is that it enables the detection of changes in the activity level of a node, having as reference the activity of the other nodes in its close communication environment.

Besides real anomalies, one reason for our statistical test to see the observed communication volume to fall out of the confidence intervals is when the assumptions are not verified. The consistency may has not been reached yet, which means that the number of training samples is not large enough. The other reason may be that the support of the distribution has changed (e.g. nodes sharing events for the first time), which is however important to be able to detect as well.

The consistency assumption is pretty typical for a regression framework. The reader may refer to the large literature that deals with this question [25, 26] in which it has been shown that many regressors are consistent. As clarified earlier, this work does not aim to provide a new regression method, however, we must note that our method largely depends on the convergence rate of the estimator.

The last assumption we need to analyze is the bounded difference of Eq. 7. In simple words, it says that when the size of the training set increases, a change of one sample does not affect much the estimated regression function. The second hypothesis, $n_0\,\kappa^2(n_0) \underset{n_0\to\infty}{\longrightarrow} 0$, is less intuitive. Nonetheless, for many estimators, $\kappa(n_0) = \mathcal{O}(\frac{1}{n_0})$ and thus the hypothesis holds. As an example, take the Nadaraya-Watson regressor [25]. Let here K be the kernel function and h the bandwidth. We then have $\kappa(n_0) = \frac{1/n_0}{\frac{1}{n_0}\sum_{i=1}^{n_0} K_h(X_i^{(-j)} - x)} = \mathcal{O}(\frac{1}{n_0})$, since $\frac{1}{n_0}\sum_{i=1}^{n_0} K_h(X_i^{(-j)} - x)$ converges to $\mathbb{E}[K_h(X^{(-j)} - x)]$.

IV. Experiments

A. State-of-the-art competitors

For our comparative evaluation, we rely on the anomaly detection literature for dynamic graph (see Sec. II). We choose state-of-the-art methods from the literature which, to the best of our knowledge, are the only existing works on the probabilistic anomaly detection at node-level, and hence are natural competitors to our work. They use the aggregated representation of the dynamic graph (see Sec. II and Fig. 1). We set the aggregation's time-scale to one day, hence the edge weight between two nodes at a time interval corresponds to the number of events shared by those two nodes in that interval.

Heard's method [3] consists in fitting, either sequentially (based on all past values) or retrospectively (based on all values but the one to predict), an homogeneous counting process on each edge of the graph independently. However, rather than focusing on edges, here we decided to model the total number of messages received per day by each node. We chose a retrospective fitting, as the number of studied timestamps is not large enough for efficient sequential fitting.

The Scan Statistics-based method in [14], at each timestamp, builds a statistic on the neighborhood around the node of interest, and normalizes it using past values in a time-window. The normalized statistic is used directly as an anomaly score. In our experiments, we used a statistic of order 0, specifically, the weighted degree of the node of interest. With our aggregated graph construction, this corresponds to the sum of weights of the adjacent edges.

Anomaly scores. We can build two anomaly scores. The first one, referred to as *bilateral*, increases when the observed value is 'far' from the expected one, in terms of absolute value. For our method, that simply corresponds to the score described in

III-B, i.e. the Eq. 6 taken negatively so that it increases with the deviation from what is expected.

The second anomaly score, referred to as *unilateral*, is motivated by the fact that in telecommunication networks an interesting type of anomalous behavior is when a node has an abnormal low level of received messages. That may reflect an antenna breakdown. For this reason, the anomaly score should increase only when the observed value is lower than expected. For our method, we simply take $\rho = -\exp(\frac{-2(\widehat{\mu}-m_t^{(j)})}{9n_t})$.

B. IoT dataset

The first results are obtained from a real industrial setting that concerns Sigfox, a telecommunication operator specialized on sensor and *Internet-of-Things* (IoT) networks[1]. Networks like these are dedicated to cover objects or devices that need to exchange only little information with users avoiding standard transmission protocols (such as WiFi, 4G or Bluetooth) that may not be well-adapted to the operational constraints (e.g. for low energy consumption). When a sensor needs to transmit a message, it simply sends a signal which can be received by several nearby Base Stations (BSs) that are in reachable distance. Our objective is to detect abnormal volume of received signals observed at any BS during a day. Hence, we consider that each sent message corresponds to a single event whose fingerprint spans only over the set of receiving BSs of the network. The value of each dimension of the event fingerprint indicates whether the message has been received or not by the corresponding BS (1 or 0, respectively).

In this evaluation study we use the event stream recorded at a subset of 38 BSs over a period of 5 months. Fig. 1 shows the relative geographical locations of the BSs. Each BS is a node in the considered graph representation and each event creates instantaneously a clique in the graph (e.g. see the left column of Fig. 1) among the involved nodes that all receive the same message, sent from the same device).

The results of Fig. 2 concerns two BSs: one with a known anomaly (lying between the two vertical red lines of Fig. 2a), the other with no known issues (Fig. 2b). Note that we have the opinion of Sigfox's experts only about these two BSs, yet we lack labels for the rest of the BSs. According to the experts, network's operation has been normal during January 2017, thus, for both reference BSs the learning phase was performed during that period. We used a Random Forest regressor [27] as implemented in [28]. The testing phase was performed independently on a daily basis for the subsequent 4 months. In other words, and this concerns all our results, we report the raw outcome of the independent daily detection for anomalies without applying any post-processing that could certainly improve the performance of most methods. Fig. 2 refers to the testing phase and shows the evolution of the observed number of received messages (blue) and the evolution of the confidence levels (orange) with $\delta = 0.01$.

The results, especially the ROC curves, show that our method (bilateral and unilateral variations) outperforms the compared approaches. As expected, the tests with unilateral score were always better than those with bilateral, for all the detection methods. Fig. 2b suggests that our model is well-suited for the analysis of the BSs in normal network operation. Indeed, the false positive rates are pretty low in that case.

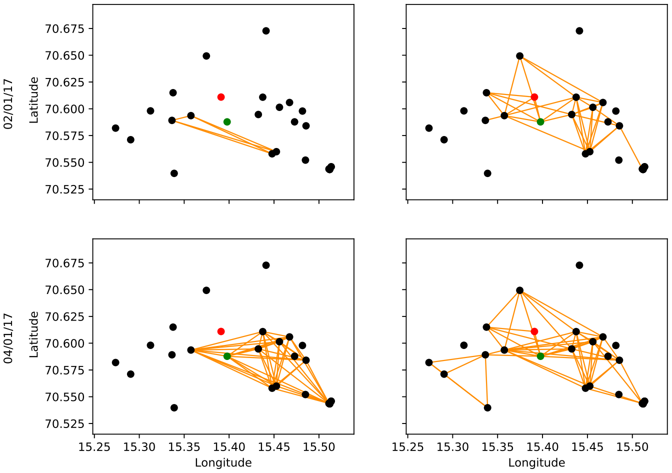


Fig. 1. High-resolution and aggregated representations (columns) of communication activity in a part of the considered Sigfox IoT network, during two consecutive days (rows). Each node corresponds to a Base Station (BS). The red BS has been tagged as *anomalous* by experts and presents *abnormal* behavior at some point during he observation time, while the green BS is taken as reference of *normal* behavior. **Left column:** Each graph represents a single event that occurred at the day indicated on the left. The involved nodes form a clique in the network. **Right column:** Each graph is an aggregated representation of all the events of the respective day. A link is drawn when two nodes share more than 30% of the total number of messages they received during that day.

To prove this latter idea, we applied our method on 5 other BSs which are located close to each other. The predicted confidence region around the predicted value are plotted for each BS in Fig. 3. Once again, we can see that the observed number of received messages falls out of the confidence level very few times. The fact that our method reports long anomalies for many BSs during May 11 - 25, may be a sign that retraining is needed. However, for the third BS, the observed value is persistently very low compared to the predicted confidence intervals, which is a stronger indication for anomalous behavior during that period of time.

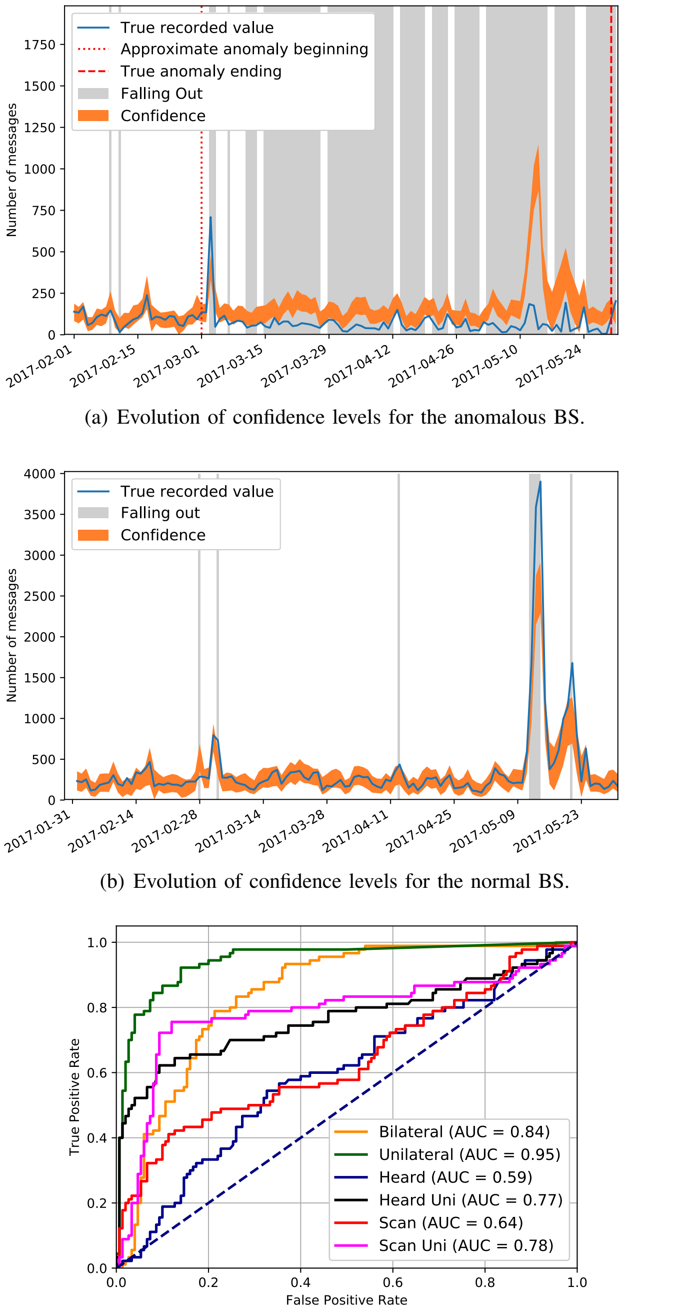
C. Simulated dataset

Aiming to extend the scale of our experimental study, we developed a data generator that simulates network communication activity. To be consistent with the previous experiment of Sec. IV-B, we keep the nodes' spatial arrangement of Sigfox network. We propose the following simulation process:

S1) *Sample the spatial network structure:* Draw N node (i.e. analogous to BSs) locations, according to a mixture model $\mathcal{M}$ of K (bivariate) Gaussian distributions.

S2) *Sample an event/fingerprint:* First generate a transmission location (analogous to a device) $\ell \sim \mathcal{M}$, as in Step 1. Then for each node, let its location x, draw a Bernoulli with a parameter inversely proportional to the distance

[1] The datasets and our implementation of all compared anomaly detection methods are publicly available at http://kalogeratos.com/psite/nad2019.



(a) Evolution of confidence levels for the anomalous BS.

(b) Evolution of confidence levels for the normal BS.

(c) ROC curves for bilateral and unilateral confidence levels. Comparison with stationary counting processes.

Fig. 2. Results on two BSs of the considered IoT network. **(a–b)**: The true number of messages received by the abnormal BS and the normal one over the testing period. The yellow area corresponds to the predicted confidence region for the number of received messages. **(c)**: The ROC curves and their AUC of the proposed method using a bilateral (orange) and an unilateral (green) anomaly score. Comparison with Heard's [3] and Scan Statistics [13, 14].

$d(x,\ell)$. In our experiments, we set the Bernoulli parameters to be equal to $\exp(-\frac{1}{\sigma_x}d(x,\ell))$, where σ_x is a *location-dependent visibility* parameter that controls the density of the graph.

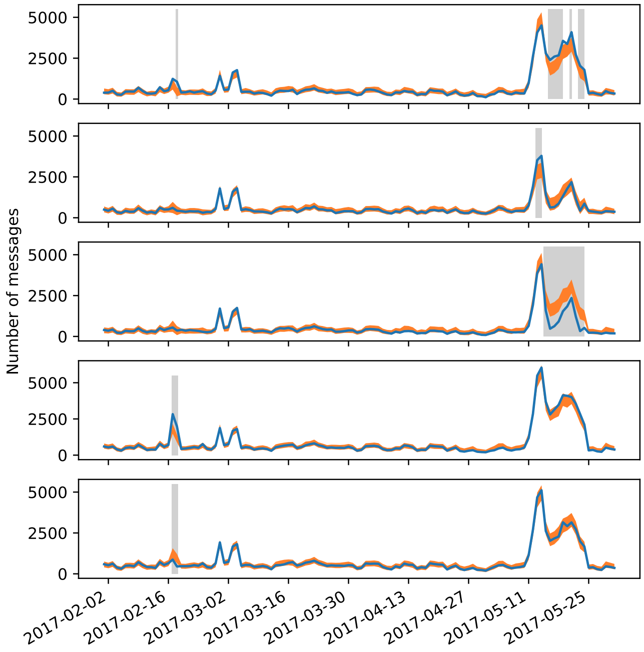


Fig. 3. Evolution of the predicted confidence regions for five BSs.

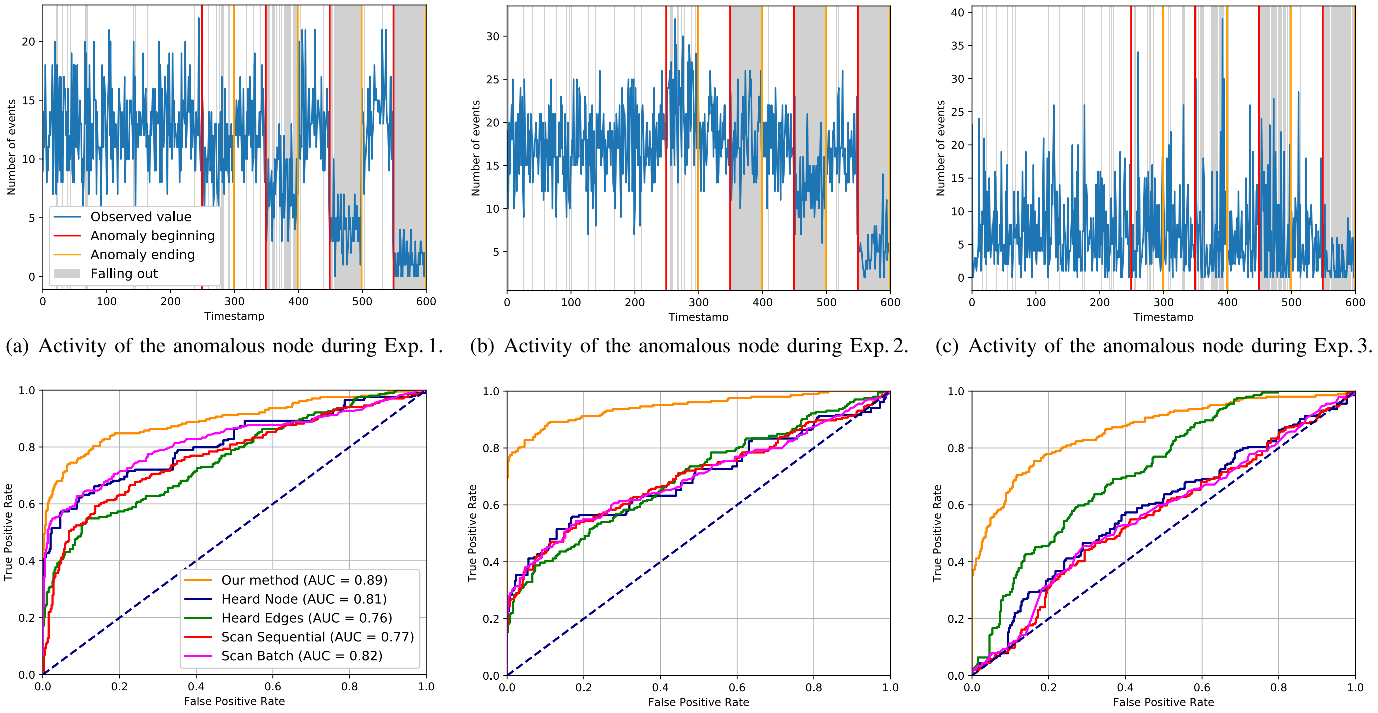
S3) *Generate an event stream (clique stream):* At each timestamp t, draw n_t event fingerprints by applying Steps 2-3, where n_t may be constant, or random, over time.

S4) *Simulate anomalies through non-stationarity:* The simplest way to simulate non-stationarity is to draw the total number of events at each timestamp according to a non-stationary process. To increase the complexity of the phenomena, one may also let the component (or cluster) proportion of the mixture in $\mathcal{M}$ to vary at each timestamp. That would correspond to the case where devices appear following a non-uniform spatial distribution. To simulate anomalies for a node, it is sufficient to let vary the visibility parameter σ_x associated to the node's location.

In order to demonstrate the robustness of our method, we apply the above generative process in three simulations with different 'complexity', whereas sharing the following properties:

- S1: $N = 100$ communication nodes are drawn, for which, $T = 1100$ timestamps are then simulated. The number of Gaussian distributions are fixed to $K = 10$.
- S2: The same set of constant visibility parameters is used.
- S3: The first 500 timestamps are treated as the training stream, while the rest correspond to the test stream.
- S4: A single arbitrary node is chosen to be anomalous. For which, 4 anomaly time intervals are simulated: $[750, 800]$, $[850, 900]$, $[950, 1000]$ and $[1050, 1100]$. Each of these intervals imitates an anomalous behavior at a different scale; this is achieved by decreasing only the visibility parameter σ_x associated with the anomalous node.

Our three experiments (Exp. 1-3) differ in their complexity regarding the stationarity of the respective time-series, i.e. number of events in which the *anomalous node* (the node that



(a) Activity of the anomalous node during Exp. 1.

(b) Activity of the anomalous node during Exp. 2.

(c) Activity of the anomalous node during Exp. 3.

(d) ROC curves for Exp. 1.

(e) ROC curves for Exp. 2.

(f) ROC curves for Exp. 3.

Fig. 4. Results on simulated communication streams. The columns report results related to the three generated streams in order, Exp. $1, 2, 3$, respectively. **(a-c)** The time-series of the number of messages (i.e. communication events) received by the anomalous node during the testing period of each experiment. Four anomaly intervals are simulated for the same fixed node that is chosen to act as anomalous, in the time intervals $[750, 800]$, $[850, 900]$, $[950, 1000]$, and $[1050, 1100]$. The beginning and the end timestamps of each anomalous interval are indicated with red and orange vertical lines, respectively, in the plots. **(d-f)** the ROC curves of the node-level outlier detection task for the three synthetic streams.

at some point develops an anomalous behavior) participates in each experiment. The top row of Fig. 4 presents the time-series of the test streams. The timestamps of the beginning and the end of each simulated anomalous behavior are also indicated in the plots with orange and red vertical lines, respectively.

In Exp. 1 (Fig. 4a, d), the process is perfectly stationary: at each timestamp, exactly 100 events are generated with the same process. In Exp. 2 (Fig. 4b, e), the total number of events participated at each timestamp remains 100, with the difference that a Dirichlet random variable of order K is drawn, with parameters all equal to 1. This corresponds to the mixing variable (i.e. proportion) for the components of $\mathcal{M}$. The last one, Exp. 3 (Fig. 4c, f), is meant to be more difficult: it uses the Dirichlet mixing as well, however, at each timestamp of anomaly, the total number of generated events is increased. This is a 'tricky' setting for the bare human eye as the time-series of interest 'looks' stationary (Fig. 4c) although there are actual changepoints in the node behavior.

For all three experiments, the threshold value that needs to be fixed for building the confidence levels was estimated using cross-validation (see details in Sec. III-B). We fix the acceptable false positive rate at 0.05. The light gray vertical lines in the background of the top row plots in Fig. 4 indicate the timestamps at which the observed values fall out of the confidence levels, and as such they can be spotted as outliers.

The bottom row of Fig. 4 shows the ROC curves of the node-level outlier detection task for the three synthetic streams. The competitors are the same as those of the experiment on real data (Fig. 2), but here only bilateral scores are plotted. In addition, here we employ a second version of both Heard's and Scan Statistics methods. The *Heard Edges* fits an homogeneous Poisson process on each edge independently; each edge denoting the number of common messages received by two nodes. In this case, the node anomaly score is the sum of p-values of its edges. Moreover, the *Scan Batch* method simply outputs anomaly scores equal to the normalized deviation of the statistic of interest (see Sec. IV-A) from a (unordered) batch of the training stream, hence without sequential analysis.

All the reported results indicate that the proposed method outperforms clearly its competitors. As expected by its design, our approach is shown to be robust to the non-stationarity introduced at arbitrary timestamps during our simulations. The performance of all other methods seems to decrease fast with the increase of non-stationarity (i.e. behavior complexity). An important closing remark is to remind that, in our evaluation, we have been applying anomaly detection independently on each day. As this work is related to the detection of change-points in nodes' behavior rather than instantaneous anomalies, post-processing (such as filtering) of the raw detection outcome could increase the accuracy of most methods.

V. Conclusions

In this paper we presented a probabilistic framework for node-level anomaly detection in communication networks. We went beyond the aggregated representations that the existing literature has used to model the communication activity. Instead, we modeled such activity as a *clique stream* where each event creates an instantaneous clique among the communicating nodes of the graph. The detection approach we proposed is to infer the conditional probabilities of cliques to be generated. This allowed the derivation of node anomaly scores which are efficient in detecting when the communication volume deviates from the 'normal' behavior (estimated using a training stream of normal communication behavior), while also being statistically interpretable. We applied our method on both real-world and synthetic sensor network data, and demonstrated that it outperforms other probabilistic approaches found in the related literature.

As future work, there is room to further improve the accuracy of the statistical modeling, consider that events can create more complex structures of connected nodes than cliques, include dynamics coming from (dis)appearance of nodes, and finally bring our method closer to the link prediction or structure inference tasks, using for instance the learned conditional probabilities.

Acknowledgments

Part of this work was funded by the IdAML Chair hosted at ENS-Paris-Saclay.

References

[1] M. Latapy, T. Viard, and C. Magnien, "Stream graphs and link streams for the modeling of interactions over time," *preprint arXiv:1710.04073*, 2017.

[2] S. Ranshous, S. Shen, D. Koutra, S. Harenberg, C. Faloutsos, and N. Samatova, "Anomaly detection in dynamic networks: a survey," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 7, no. 3, pp. 223–247, 2015.

[3] N. Heard, D. Weston, K. Platanioti, and D. Hand, "Bayesian anomaly detection methods for social networks," *The Annals of Applied Statistics*, vol. 4, no. 2, pp. 645–662, 2010.

[4] M. Corneli, P. Latouche, and F. Rossi, "Multiple change points detection and clustering in dynamic networks," *Statistics and Computing*, pp. 1–19, 2017.

[5] H. Cheng, P.-N. Tan, C. Potter, and S. Klooster, "Detection and characterization of anomalies in multivariate time series," in *Proc. of the SIAM Intern. Conf. on Data Mining*, 2009, pp. 413–424.

[6] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier detection for temporal data: A survey," *IEEE Trans. on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2250–2267, 2014.

[7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *Computing Surveys*, vol. 41, no. 3, pp. 15:1–15:58, 2009.

[8] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[9] C. D. Scott and R. D. Nowak, "Learning minimum volume sets," *J. of Machine Learning Research*, vol. 7, no. Apr, pp. 665–704, 2006.

[10] J. Di and E. Kolaczyk, "Complexity-penalized estimation of minimum volume sets for dependent data," *J. of Multivariate Analysis*, vol. 101, no. 9, pp. 1910–1926, 2010.

[11] M. Breunig, H. Kriegel, R. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *ACM SIGMOD Record*, vol. 29, no. 2, 2000, pp. 93–104.

[12] L. Huang, X. Nguyen, M. Garofalakis, M. Jordan, A. Joseph, and N. Taft, "In-network PCA and anomaly detection," in *Advances in Neural Information Processing Systems*, 2007, pp. 617–624.

[13] C. E. Priebe, J. M. Conroy, D. J. Marchette, and Y. Park, "Scan statistics on Enron graphs," *Computational & Mathematical Organization Theory*, vol. 11, no. 3, pp. 229–247, 2005.

[14] H. Wang, M. Tang, Y. Park, and C. Priebe, "Locality statistics for anomaly detection in time series of graphs," *IEEE Trans. on Signal Processing*, vol. 62, no. 3, pp. 703–717, 2014.

[15] L. Peel and A. Clauset, "Detecting change points in the large-scale structure of evolving networks," in *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 15, 2015, pp. 1–11.

[16] C. C. Aggarwal, Y. Zhao, and S. Y. Philip, "Outlier detection in graph streams," in *Proc. of the IEEE Intern. Conf. on Data Engineering*, 2011, pp. 399–409.

[17] J. Neil, C. Hash, A. Brugh, M. Fisk, and C. Storlie, "Scan statistics for the online detection of locally anomalous subgraphs," *Technometrics*, vol. 55, no. 4, pp. 403–414, 2013.

[18] J. Sun, Y. Xie, H. Zhang, and C. Faloutsos, "Less is more: Compact matrix decomposition for large sparse graphs," in *Proc. of the SIAM Intern. Conf. on Data Mining*, 2007, pp. 366–377.

[19] T. G. Kolda and J. Sun, "Scalable tensor decompositions for multi-aspect data mining," in *Proc. of the IEEE Intern. Conf. on Data Mining*, 2008, pp. 363–372.

[20] T. Ji, D. Yang, and J. Gao, "Incremental local evolutionary outlier detection for dynamic social networks," in *Proc. of the Joint European Conf. on Machine Learning and Knowledge Discovery in Databases*. Springer, 2013, pp. 1–15.

[21] B. Pincombe, "Anomaly detection in time series of graphs using arma processes," *Asor Bulletin*, vol. 24, no. 4, p. 2, 2005.

[22] X. Wan, E. Milios, N. Kalyaniwalla, and J. Janssen, "Link-based event detection in email communication networks," in *Proc. of the ACM Symp. on Applied Computing*, 2009, pp. 1506–1510.

[23] S. Boucheron, G. Lugosi, and P. Massart, *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013.

[24] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *J. of the American Statistical Association*, vol. 58, no. 301, pp. 13–30, 1963.

[25] L. Györfi, M. Kohler, A. Krzyzak, and H. Walk, *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

[26] L. Györfi, *Principles of nonparametric learning*. Springer, 2002, vol. 434.

[27] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

APPENDIX

Theorem 1. *See Sec. III-C.*

Proof. In the following, we note $\widehat{\eta}_{n_0}(\cdot) := \widehat{\eta}_j(\cdot\,;\, X_1^0,\dots,X_{n_0}^0)$. We also assume the distribution described in Theorem 1: $\forall i = 1,\dots,n_0,\ X_i \underset{i.i.d.}{\sim} \mathbb{P}_{X^0}$ and $\forall j = 1,\dots,n_t,\ X_j \underset{i.i.d.}{\sim} \mathbb{P}_X$. With an abuse of notation, $\mathbb{P}_{X^0}$ and $\mathbb{P}_X$ also refer to the marginal distributions. Using the triangle inequality, we get:

$$
\begin{aligned}
&\left| M_t^{(j)} - \sum_{i=1}^{n_t} \widehat{\eta}_{n_0}(X_i^{(-j)}) \right| \\
&\le \left| M_t^{(j)} - \sum_{i=1}^{n_t} \eta^*(X_i^{(-j)}) \right| + \left| \sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right| \\
&\le \left| M_t^{(j)} - \sum_{i=1}^{n_t} \eta^*(X_i^{(-j)}) \right| \\
&\quad + \left| \sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) - \right. \\
&\qquad\qquad \left. \mathbb{E}_{X^0 \otimes X}\left[\sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right] \right| \\
&\quad + \underbrace{\left| \mathbb{E}_{X^0 \otimes X}\left[\sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right] \right|}_{(*)}.
\end{aligned}
$$

In the above, $\mathbb{E}_{X^0 \otimes X}$ means that the expectation is taken with distribution $\mathbb{P}_X^0$ for $\mathcal{S}_0$ and $\mathbb{P}_X$ for $\mathcal{S}_t$. Using Jensen's inequality and the fact that all $X_i^{(-j)}$ are i.i.d., we get:

$$
\begin{aligned}
(*) &\le \mathbb{E}_{X^0 \otimes X}\left[\left| \sum_{i=1}^{n_t} \eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)}) \right| \right] \\
&\le n_t \mathbb{E}_{X^0 \otimes X}\left[|\eta^*(X) - \widehat{\eta}_{n_0}(X)| \right] \\
&\le n_t \mathbb{E}_{X^0}\left[\int |\eta^*(\mathbf{x}) - \widehat{\eta}_{n_0}(\mathbf{x})| \mathbb{P}_X(d\mathbf{x}) \right].
\end{aligned}
$$

Using Cauchy-Schwarz inequality:

$$
\begin{aligned}
&\le n_t \mathbb{E}_{X^0}\left[\sqrt{\int (\eta^*(\mathbf{x}) - \widehat{\eta}_{n_0}(\mathbf{x}))^2 \mathbb{P}_{X^0}(d\mathbf{x})} \times \right. \\
&\qquad\qquad \left. \times \sqrt{\int \frac{\mathbb{P}_X(\mathbf{x})}{\mathbb{P}_{X^0}(\mathbf{x})} \mathbb{P}_{X^0}(d\mathbf{x})} \right].
\end{aligned}
$$

With the same support hypothesis:

$$
= n_t \mathbb{E}_{X^0}\left[\sqrt{\int (\eta^*(\mathbf{x}) - \widehat{\eta}_{n_0}(\mathbf{x}))^2 \mathbb{P}_{X^0}(d\mathbf{x})} \right].
$$

Using Jensen inequality:

$$
\le \sqrt{\mathbb{E}_{X^0 \otimes X^0}\left[(\eta^*(X) - \widehat{\eta}_{n_0}(X))^2 \right]} := r(n_0).
$$

Due to the assumption of weak consistency, $r(n_0)$ converges to zero, so as $(*)$. In the following, we assume that $r(n_0) < s$ which is always true after a certain rank. We note $\tilde{s}(n_0) = s - r(n_0)$. Back to the first inequality of the proof, we get $\forall k \in (0, \tilde{s}(n_0))$:

$$
\begin{aligned}
&\mathbb{P}(|M_t^{(j)} - \sum_{i=1}^{n_t} \widehat{\eta}_{n_0}(X_i^{(-j)})| > s) \\
&\le \mathbb{P}(|M_t^{(j)} - \sum_{i=1}^{n_t} \eta^*(X_i^{(-j)})| > k) \\
&+ \mathbb{P}\left(\left| \sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right. \right. \\
&\quad \left. \left. - \mathbb{E}_{X^0 \otimes X}\left[\sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right] \right| > \tilde{s}(n_0) - k \right).
\end{aligned}
$$

This is due to the fact that $k + \tilde{s}(n_0) - k + r(n_0) = s$. We now need to find an upper bound on the two elements of the right-hand side of the previous inequality. The first element of the sum is easily bounded using Jensen's inequality:

$$
\mathbb{P}\left(|M_t^{(j)} - \sum_{i=1}^{n_t} \eta^*(X_i^{(-j)})| > k \right) \le 2 \exp\left(-\frac{2k^2}{n_t} \right).
$$

For the second element, we must note that $\sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)}))$ is a function, with bounded differences, of $n_0 + n_t$ independent random variables. Thus, we can apply McDiarmid's inequality to bound our probability:

$$
\begin{aligned}
&\mathbb{P}\left(\left| \sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right. \right. \\
&\qquad \left. \left. - \mathbb{E}_{X^0 \otimes X}\left[\sum_{i=1}^{n_t} (\eta^*(X_i^{(-j)}) - \widehat{\eta}_{n_0}(X_i^{(-j)})) \right] \right| > \tilde{s}(n_0) - k \right) \\
&\le 2 \exp\left(-2 \frac{(\tilde{s}(n_0) - k)^2}{4n_t + n_t^2 n_0 \kappa^2(n_0)} \right).
\end{aligned}
$$

This implies that:

$$
\begin{aligned}
&\mathbb{P}(|M_t^{(j)} - \sum_{i=1}^{n_t} \widehat{\eta}_{n_0}(X_i^{(-j)})| > s) \\
&\qquad \le 2 \exp\left(-\frac{2k^2}{n_t} \right) + 2 \exp\left(-\frac{2(\tilde{s}(n_0) - k)^2}{4n_t + n_t^2 n_0 \kappa^2(n_0)} \right).
\end{aligned}
$$

This is true $\forall k \in (0, \tilde{s}(n_0))$. Furthermore, since $\tilde{s}(n_0) \underset{n_0 \to \infty}{\longrightarrow} s$ and $n_0 \kappa^2(n_0) \underset{n_0 \to \infty}{\longrightarrow} 0$, passing to the limit on both side of the previous equation, we get our final result:

$$
\begin{aligned}
&\lim_{n_0 \to \infty} \mathbb{P}(|M_t^{(j)} - \sum_{i=1}^{n_t} \widehat{\eta}_{n_0}(X_i^{(-j)})| > s) \\
&\qquad \le \min_{k \in [0,s]} \left\{ 2 \exp\left(-\frac{2k^2}{n_t} \right) + 2 \exp\left(-\frac{(s-k)^2}{2n_t} \right) \right\}.
\end{aligned}
$$

□