Anomaly Characterization in Large Scale Networks

Emmanuelle Anceaume, Yann Busnel, Erwan Le Merrer, Romaric Ludinard, Jean-Louis Marchand, Bruno Sericola

Anomaly Characterization in Large Scale Networks

Emmanuelle Anceaume
CNRS / IRISA, France
emmanuelle.anceaume@irisa.fr

Yann Busnel
LINA / Université de Nantes, France
yann.busnel@univ-nantes.fr

Erwan Le Merrer
Technicolor Rennes, France
erwan.lemerrer@technicolor.com

Romaric Ludinard
Inria, France
romaric.ludinard@inria.fr

Jean-Louis Marchand
Ecole Normale Supérieure de Rennes, France
jean-louis.marchand@ens-rennes.fr

Bruno Sericola
Inria, France
bruno.sericola@inria.fr

***Abstract*—The context of this work is the online characterization of anomalies in large scale systems. In particular, we address the following question: Given two successive configurations of the system, can we distinguish massive anomalies from isolated ones, the former ones impacting a large number of nodes while the second ones affect solely a small number of them, or even a single one? The rationale of this question is twofold. First, from a theoretical point of view, we characterize anomalies with respect to their neighborhood, and we show that there are anomaly scenarios for which isolated and massive anomalies are indistinguishable from an global observer point of view. We then relax the definition of this problem by introducing *unresolved* configurations, and exhibit necessary and sufficient conditions that allows any node to determine the type of anomaly it has been impacted by. This condition only depends on the close neighborhood of each node and thus is locally computable. We present an algorithm that implements this condition. We show through extensive simulations the performance of our algorithm. From a practical point of view, distinguishing isolated anomalies from massive ones is of utmost importance for networks providers. For instance, regarding Internet service providers that operate millions of home gateways, it would be very interesting to have procedures that allow gateways to self distinguish whether their dysfunction is caused by network-level anomalies or by their own hardware or software, and to notify the service provider only in the latter case.**

I. Introduction

In this paper we study the online monitoring problem in large scale distributed systems. This problem deals with the capability of collecting and analyzing relevant information provided by monitored devices so as to make the monitoring application continuously aware of the state of the system. In presence of a large number of monitored devices (*i.e.*, a typical scenario is the one encountered by Internet service providers operating millions of home gateways), an approach to solve this problem is to rely on *customers care call centers*. Such call centers are notified by customers that experience degradations of service quality. This approach while commonly adopted, shows several issues in terms of latency (*i.e.*, the latent detection period between the occurrence of an incident and the instant at which the customer observes it is unpredictable), cost (*i.e.*, it requires to mobilize agents for manually handling each customer notification), and inefficiency (*e.g.*, when incidents lie in a part of the network that is not operated by the service provider or when notifications are due to customers negligences). These issues call for automated monitoring procedures that should be able to notify the service provider only for legitimate reasons. Actually, standardized procedures [4] exist at devices level to autonomously trigger investigations in presence of errors or networks events. However, these procedures are never used for practical reasons. Indeed if the cause of a QoS variation lies in the network itself – due to routing loops, router dysfunctions, or configuration errors – this may impact a very large number of devices (more precisely, impact services consumed by these devices), and thus letting thousands of impacted devices reporting the problem to the operator may quickly become a disaster. It is thus of utmost importance to minimize the overall pressure put on the service operator, by giving each device the capability to locally detect whether the local QoS degradation is also observed at many other devices or not, so that only isolated errors or events are reported on the fly by the devices experiencing them. Alternatively, there is a clear need for *over-the-top* operators – that rely on Internet Service Providers to transparently deliver content to their clients – to quickly detect network level events. Indeed, incidents at the network level may impact the quality at which data is received at thousands of clients that will naturally blame their over-the-top operators. Our solution provides each end-device the capability to self distinguish network-based events from local ones, so that only network-events are reported on the fly to the over-the-top operators.

In both cases, the key point is to provide each monitored device a way to estimate the impact on other devices of a locally perceived QoS degradation. The approach we propose boils down for a device to locally detect the presence of similarity features in the abnormal behavior of other devices. This is achieved by modeling the QoS (quality of service) of the different services accessed by a device by a point in a QoS space E, and the temporal evolution of its QoS by a trajectory in E. A trajectory is abnormal if the predicted values of the QoS differ from the observed ones. The problem we tackle amounts for a device to locally identify all the abnormal trajectories that are *close* to its own one, to determine how dense they are, to finally decide whether its services have been impacted by an isolated event or a network one. The

notion of closeness is modeled by the presence of points in a ball centered at a given point. Surprisingly, we show that it exists some trajectories that are indistinguishable (even from the point of view of an omniscient observer) in the sense that it is impossible to decide whether they are due to isolated events or network ones. We formally characterize these unresolved configurations, and derive necessary and sufficient conditions that allow each device to locally decide with certainty whether it belongs to an unresolved configuration or if it has been impacted by an isolated event, or by a network one. In our approach, the frequency at which QoS information is sampled is locally tuned, and only depends on the local occurrence of QoS degradations. Therefore, by avoiding any kind of global synchronization, devices can efficiently provide a fine grain event/errors detection without impacting the rest of the system. The influence of this local tuning has an enjoyable consequence on the number of unresolved configurations: by sampling sufficiently often one's neighborhood, the number of unresolved configurations drastically shrinks. To summarize, our contributions are :

- A modeling of isolated and network based errors or events;
- The derivation of local conditions that allow each device to decide with certainty whether its observed QoS degradation is due to an isolated event or a network one;
- A fine granularity of event detection locally tunable and transparent to the remaining of the system;
- The design of local algorithms whose decisions are as accurate as the one provided by an omniscient observer.

The remaining of the paper is organized as follows. Section II provides an overview of existing monitoring approaches. Section III presents the model of the system, and how errors are modeled. Section IV formalizes the online anomaly detection problem and its relaxed version. Section V presents computable conditions that allow any device to locally solve the relaxed version of the anomaly detection problem. Section VI presents the local algorithms, and their performance are analyzed through extensive simulations (see Section VII). Section VIII concludes and presents future works.

II. Related Work

This section provides an overview of the existing techniques used in large scale systems to continuously and automatically monitor time-varying metrics. The authors in [15] exploit temporal and spatial correlations [3], [8], [11] among groups of monitored nodes to decrease monitoring communication costs, *i.e.*, the cost incurred by the periodic reporting of the updated metrics values from the monitored nodes to the management node. The idea is to prevent any reporting message from occurring when such a reporting would contain metrics values that could be directly inferred by the management node. This is achieved by giving each monitored node the capability to locally detect whether the current values of its monitored metrics are in accordance with predicted ones (through Kalman filters tools [7] installed at both monitored nodes and the management node), and by gathering nodes into clusters (such that, for each monitored metric, a set of clusters groups together nodes that share correlated values of the considered metric according to the Pearson correlation coefficient). At clusters level, an elected leader is in charge of communicating with the management system when the current metric values of its group members differ from each others. Although close to our objectives, the main drawback of this solution lies on the centralized clustering process. All the nodes of the system are continuously organized into clusters computed through the k-means algorithm exclusively run by the management node, which is a clear impediment to the scalability of their approach. Other works aim at minimizing the processing cost for continuous monitoring [13], [9], [14] in the light of the theoretical results of [5], however similarly to [15], all these approaches suffer from a centralized handling of the clustering process. Recently, Choffnes et al. [2] have proposed to leverage structured peer-to-peer architectures (*i.e.*, Distributed Hashing Tables) to guarantee efficient and scalable monitoring management. In contrast to the previous described works that focus on monitoring fine-grained changes on individual nodes, [2] pushes monitoring on end users. Their approach consists in having a set of cooperating edge system monitors (ESM), each having access to a distributed storage system (*i.e.*, based on a DHT) in which they publish aggregated informations about events detected in their own sub-network. This allows any network operators (such as ISPs) to regularly access the storage architecture to analyze system wide detected events, and thus to detect global, or at least massive, network outages. The authors in [1] propose an online error detection mechanism that in contrast to [2] is proactive. Their mechanism fully depends on the tessellation of the overlay, which may lead to numerous false negative and false positive anomalies. Indeed, tessellating the space with large buckets sizes tends to identify each possible anomaly as a massive one, while considering small buckets sizes reduces drastically the probability of having a large number of devices in a single bucket, giving rise to the triggering of false alarms. In our approach, we go a step further by providing end devices the capability to exploit correlation between their state to detect on the fly whether that have been impacted by network errors and isolated errors.

III. System Model

This section details the notations and concepts we need to model the impact of outages on the monitored devices. In the following we adopt the following conventional notations. Variables are represented by lowercase letters as j, ℓ, p and q, sets are denoted by capital letters as S and E, and families of sets are denoted by capital calligraphic letters as $\mathcal{P}$. The set of integers $\{1, \ldots, n\}$ is denoted by $[\![1, n]\!]$.

A. Preliminaries

We consider a set of n monitored devices, such that each one continuously consumes d services $s_1, \ldots, s_d$. At any discrete time k, the QoS of each service s_i at device j is locally measured with an end-to-end performance measurement function

$q_{i,k}(j)$, whose range of values is $[0,1]$. Measurement functions reflect errors (or failures) occurring on the chain of equipments and network links from the providers of consumed services to the monitored devices.

We model the QoS of monitored devices at discrete time k as a set S_k of n points in a space $E = [0,1]^d$, with $d \geq 1$, called the QoS space. The position of device j at time k is represented by point $p_k(j) = (q_{1,k}(j), \ldots, q_{d,k}(j))$. Note that in the following we interchangeably use terms *device* and *point* to speak about the position of a device in the QoS space E. The state S_k of the system at discrete time k is $S_k = (p_k(1), \ldots, p_k(n))$. Each monitored device j has also access at any time k to an error detection function $a_k(j)$ such that $a_k(j) = \texttt{true}$ if there is at least one service consumed by device j at time k whose variation of quality of service is too large to be considered as normal. Error detection functions provide some meaningful predictions of what should be the next output value based on the sequence of past input values. Different kinds of error detection functions exist, ranging from simple threshold based functions to more sophisticated ones like the Holt-Winters forecasting or Cusum methods [6], [12], [10]. Note that implementation of a is out-of the scope of the paper.

As said in the introduction, the impact of errors on devices can either be locally restricted (that is, each error affects a few number of devices, typically no more than τ, with τ a configuration parameter) or spread over a large number of devices (*i.e.*, more than τ devices). A set of devices whose positions in the QoS space E are very close to each other exhibit a similar QoS. We make the assumption that if a set of devices, exhibiting a similar QoS at time $k-1$ are impacted by the same error then they will undergo the same QoS variation. Thus, at time k their modified positions in E will still be close to each other. This closeness assumption is modeled by the presence of points in a ball of radius r. In the following r is called the *consistency impact radius*. Prior to formally modeling the impact of errors or events on devices, we first present the notions we will intensively use in the following.

B. Terminology and Notations

For the sake of simplicity, we use the uniform norm $\|\cdot\|$ defined for any $x = (x_1, \ldots, x_d) \in E$ by $\|x\| = \max\{x_1, \ldots, x_d\}$. As we consider a finite dimension space, all norms are equivalent and differ from a constant factor.

Definition 1 (r-consistent set): For any $r \in [0, 1/4)$, a subset $B \subseteq [\![1,n]\!]$ is said to be *r-consistent* at time k if the maximal distance between any $i, j \in B$ is not larger than $2r$, that is,

$$\forall (i,j) \in B^2, \|p_k(i) - p_k(j)\| \leq 2r.$$

Definition 2 (Maximal r-consistent set): For any $r \in [0, 1/4)$, a subset $B \subseteq [\![1,n]\!]$ is a *maximal r-consistent set* at time k if and only if B is an r-consistent set at time k and $\forall j \in [\![1,n]\!] \setminus B$, $B \cup \{j\}$ is not an r-consistent set at time k.

Figure 1 illustrates these two above notions. It depicts the position of six devices $\{1,2,3,4,5,6\}$ at time k in a one dimension QoS space E (*i.e.*, the number of accessed services is equal to one). The distance between any two devices of $\{1,2,3,4\}$ (resp. of $\{1,2,3,5,6\}$) is small enough to consider that the variations of their perceived QoS are correlated, and thus they may belong to the same r-consistent set. The two r-consistent sets B_1 and B_2 containing device 1, with $B_1 = \{1,2,3,4\}$, and $B_2 = \{1,2,3,5,6\}$ are maximal because adding device 5 or 6 to B_1 or adding 4 to B_2 would make them non r-consistent.

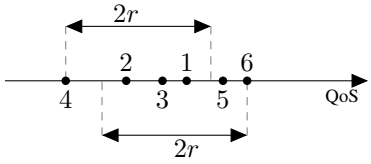


Fig. 1. The two maximal r-consistent sets $B_1 = \{1,2,3,4\}$ and $B_2 = \{1,2,3,5,6\}$ containing point 1. Any subset of B_1 and any subset of B_2 is an r-consistent set.

From these two above notions, we derive the concept of consistent *motions*. This notion reflects the fact that the QoS of a set of devices keep close to each other at successive discrete times. Formally,

Definition 3 (r-consistent motion): For any $r \in [0, 1/4)$, a subset $B \subseteq [\![1,n]\!]$ has an *r-consistent motion* in the time interval $[k-1, k]$ if B is an r-consistent set at both times $k-1$ and k. Moreover, a subset $B \subseteq [\![1,n]\!]$ has a *maximal r-consistent motion* in the time interval $[k-1,k]$ if B has an r-consistent motion in the time interval $[k-1,k]$ and $\forall j \in [\![1,n]\!] \setminus B, B \cup \{j\}$ does not have an r-consistent motion in the time interval $[k-1,k]$.

Remark 1: If B has an r-consistent motion in the time interval $[k-1,k]$, either B has a maximal r-consistent motion or there exists $B' \subseteq [\![1,n]\!], B \subseteq B'$ such that B' has a maximal r-consistent motion.

Finally, we classify r-consistent motions according to the number of devices (or equivalently points) that belong to these motions. This notion will be central for the modeling of errors or events (see Section III-C).

Definition 4 (τ-dense and τ-sparse motions): For any $r \in [0, 1/4)$, $\tau \in [\![1, n-1]\!]$, and for any subset $B \subseteq [\![1,n]\!]$ having an r-consistent motion in the time interval $[k-1,k]$, if $|B| > \tau$ then B is said to have a τ-dense r-consistent motion in $[k-1,k]$, otherwise B has a τ-sparse r-consistent motion in $[k-1,k]$.

In the following, we will simply refer to a "τ-dense motion" (resp. "τ-sparse motion") as a substitute for a "τ-dense r-consistent motion" (resp. "τ-sparse r-consistent motion") when clear from context.

C. Modeling the Impact of Errors

Each device j continuously consumes d services, and for each of them, periodically computes an end-to-end quality of service which is used to feed an error detection function $a_k(j)$. If the variation of quality is considered as abnormal, this function returns $\texttt{true}$. We model the impact of an error

on a device by an abnormal trajectory of this device in the quality space E.

Definition 5 (Abnormal Trajectory): A point $j \in [\![1, n]\!]$ has an abnormal trajectory in the time interval $[k-1, k]$ if $a_k(j) =$ `true`. The subset of points having an abnormal trajectory in the time interval $[k-1, k]$ is denoted by A_k. Formally,

$$A_k = \{j \in [\![1, n]\!] \mid a_k(j) = \texttt{true}\}$$

As previously argued, the main objective of this work is to give each device – whose QoS of consumed services has been degraded by some error – the capability to accurately decide whether such an error has also affected many other devices or solely a few of them. This boils down for each device to locally determine the presence of similarity features in the abnormal behavior of other devices. As presented above, this is achieved by modeling devices QoS by points in the QoS space E, and the temporal evolution of their QoS by trajectories in E so that, at each time k, state S_k represents the QoS of each device. We show in the following that each device only needs to know the trajectories of devices that are at no more than $4r$ from itself. A wider knowledge – as the one got by an omniscient observer that samples at each time k the system state, *i.e.*, S_k – does not bring any additional information and thus does not provide a higher error detection accuracy (see Theorems 5, 6, and 7 and Corollary 8 in Section V).

From these periodic samplings of the system state, one can construct several plausible scenarios of errors that would explain the trajectories of each device. For instance if a group of points follow the same abnormal trajectories at different observations, it should be caused by the same error. Similarly, if some point shows an abnormal trajectory that moves it away from its previous neighbors it should be due to some isolated anomaly. On the other hand, there are scenario of errors that cannot be captured by periodic snapshots, as for example the fact that some device has been hit by simultaneous or temporally close errors between two successive snapshots. We encapsulate these indistinguishable scenarios of errors by imposing the following restrictions on the impact of errors on devices QoS.

- R1: In the time interval $[k-1, k]$, the abnormal trajectory of each device $j \in A_k$ is due to a single error.
- R2: An error has a similar effect on the abnormal trajectories of all impacted devices. In particular if a set of devices belonging to the same r-consistent set are impacted by a given error in the time interval $[k-1, k]$ then all these devices will undergo the same abnormal trajectories and thus by Definition 3 will follow the same r-consistent motion in $[k-1, k]$.
- R3: If strictly less than $\tau + 1$ devices have an abnormal trajectory due to the same error then none of these devices can belong to a τ-dense motion. Moreover, if a device belongs to a τ-dense motion then this device has necessarily been impacted by an error that has impacted many other devices (not necessarily those following the same motion).

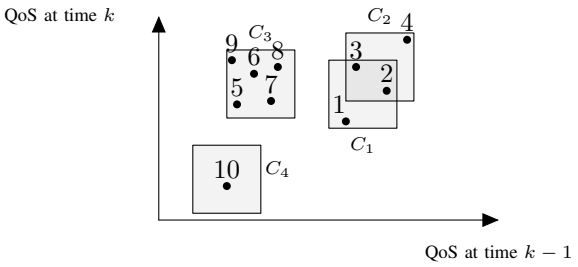


Fig. 2. QoS at time k of ten devices as a function of their QoS at time $k-1$. The four maximal r-consistent motions involving the devices are shown. The density threshold $\tau = 3$.

Note that a single error can impact devices whose QoS can be arbitrarily different. Restrictions R1, R2 and R3 are taken into account by partitioning the set of devices in A_k into τ-dense motion subsets and τ-sparse motion subsets such that *(i)* all the τ-sparse motions subsets are sufficiently "far" from each other so that any combination of several of them cannot form τ-dense motion subsets, and *(ii)* any single τ-sparse motion subset is sufficiently "far" from any τ-dense motion one so that this τ-sparse motion subset cannot merge with a τ-dense one. This partitioning of A_k is formally defined as follows.

Definition 6 (Anomaly partition $\mathcal{P}_k$): For any $k \geq 1, \tau \in [\![1, n-1]\!], r \in [0, 1/4)$, the partition $\mathcal{P}_k$ of A_k is said to be an anomaly partition at time k if it is made of non-empty and disjoint r-consistent motions $B_1, \ldots, B_\ell$ that verify conditions C1 and C2 below. Subsets $B_1, \ldots, B_\ell$ are called anomalies.

- C1: $\forall B \subseteq \bigcup_{|B_i| \leq \tau} B_i$, B has a τ-sparse motion or B has not an r-consistent motion
- C2: $\forall B \subseteq \bigcup_{|B_i| \leq \tau} B_i, \forall i \in [\![1, \ell]\!]$, B_i has a τ-dense motion $\Rightarrow B \cup B_i$ has not an r-consistent motion.

By extension, for any point $j \in A_k$, $\mathcal{P}_k(j)$ represents the (unique) subset $B \in \mathcal{P}_k$ such that $j \in B$.

In spite of the apparent complexity of Definition 6, the following lemma shows that given A_k, S_{k-1}, S_k, τ and r, there always exists at least one anomaly partition.

Lemma 2: For any $k \geq 1$, for any $A_k \neq \emptyset$, for any system states S_{k-1} and S_k, $\tau \in [\![1, n-1]\!]$ and $r \in [0, 1/4)$, there exists at least one partition $\mathcal{P}_k$ of A_k such that $\mathcal{P}_k$ is an anomaly partition. In the general case, it is not unique.

Proof: We first prove the existence of anomaly partitions, and then their non uniqueness.

- A simple way to build an anomaly partition $\mathcal{P}_k$ of A_k is described in Algorithm 1. After having initialized $\mathcal{P}_k$ to an empty set and S to A_k, all the points of S are examined as follows. Let j be any random point taken from S, and B any subset of S that has a maximal r-consistent motion in S involving j. B is added to $\mathcal{P}_k$ and all the elements of B are removed from S. The size of S is monotonically decreasing and thus this algorithm terminates. We now prove by induction that at each iteration, $\mathcal{P}_k$ satisfies conditions C1 and C2 of Definition 6. The first element added in $\mathcal{P}_k$ has a maximal motion. Being the first

Algorithm 1: Constructing an anomaly partition from A_k.

Data: $S_{k-1}, S_k, \tau \in [\![1, n-1]\!], r \in [0, \frac{1}{2})$.
Requires: A_k
Output : An anomaly partition

```
begin
    S ← A_k;
    P_k ← {};
    while S ≠ ∅ do
        Take any j ∈ S;
        Let B ⊆ S be such that j ∈ B and B ∉ P_k and B has
        a maximal r-consistent motion in S;
        S ← S \ B ;
        P_k ← P_k ∪ {B};
    return P_k;
```

element of $\mathcal{P}_k$, both conditions C1 and C2 hold. Now suppose that up to iteration $n \geq 1$ both conditions C1 and C2 hold. At the end of iteration $m = n + 1$, the new element B of $\mathcal{P}_k$ has a maximal r-consistent motion among all the remaining points of S. By construction, $\forall \ell \in [\![1, n]\!], B_\ell \in \mathcal{P}_k$ has a maximal r-consistent motion among all the remaining points of $S \setminus \cup_{i<\ell} B_i$. Thus, by Definition 3, $\forall j \in S \setminus \cup_{i<\ell} B_i, B_\ell \cup \{j\}$ has not an r-consistent motion. In particular, $\forall j \in B, B_\ell \cup \{j\}$ has not an r-consistent motion. Thus conditions C1 and C2 hold. Thus, by the induction hypothesis, C1 and C2 hold for all iteration steps. Finally, as A_k is non empty, for any j in A_k, it exists a subset $B \in \mathcal{P}_k$ such that $j \in B$. By construction of $\mathcal{P}_k$, each element of A_k belongs to only one element of $\mathcal{P}_k$ and thus $\mathcal{P}_k$ is a partition of A_k, which completes the proof.

- We now prove with a counterexample that given A_k, S_{k-1} and S_k, the anomaly partition that leads the system from S_{k-1} to S_k is in the general case not unique. Consider Figure 2 that shows the variation of QoS of a service consumed by ten devices $S = [\![1, 10]\!]$ in the time interval $[k-1, k]$. Suppose that the density threshold τ is equal to 3 and all devices in S have abnormal trajectories. Four maximal r-consistent motions C_1, C_2, C_3, C_4 are depicted. By direct application of Algorithm 1, if device 1 is first chosen, C_1, then C_3 and C_4 and finally $\{4\}$ constitutes the members of a possible anomaly partition. We have: $\mathcal{P}'_k = \{\{1, 2, 3\}, \{4\}, \{5, 6, 7, 8, 9\}, \{10\}\}$. Now if device 4 is initially chosen, we have $\mathcal{P}'_k = \{\{1\}, \{2, 3, 4\}, \{5, 6, 7, 8, 9\}, \{10\}\}$. This completes the proof.

■

Finally, according to the number of devices belonging to each $B_1, \ldots, B_\ell$ of $\mathcal{P}_k$, we differentiate between *isolated anomalies* and *massive anomalies*. Specifically,

Definition 7 (Massive / Isolated Anomalies): Let $\mathcal{P}_k$ be an anomaly partition. An element $B \in \mathcal{P}_k$ is called a *massive anomaly* in the time interval $[k-1, k]$ if $|B| > \tau$. Otherwise it is called an *isolated anomaly*. The set of devices impacted by a massive anomaly in the time interval $[k-1, k]$ is denoted by $M_{\mathcal{P}_k}$. Formally, we have $M_{\mathcal{P}_k} = \{j \in A_k \mid |\mathcal{P}_k(j)| > \tau\}$.

Notation	Meaning
E	QoS Normed space (Section III-A)
$p_k(j)$	Position of point j at time k in E (Section III-A)
r	Consistency impact radius (Section III-A)
τ	Density threshold (Definition 4)
$a_k(j)$	Anomaly detection function on device j at time k (Definition 5)
A_k	Set of points involved in an anomaly in $[k-1, k]$ (Relation 5)
S_k	System state at time k (Section III-C)
$\mathcal{P}_k$	Anomaly partition at time k (Definition 6)
$\mathcal{R}_k$	Real scenario of errors that occurred in the time interval time $[k-1, k]$
$M_{\mathcal{P}_k}$	Set of points impacted by a massive anomaly in $[k-1, k]$ w. r. t. $\mathcal{P}_k$ (Definition 7)
$I_{\mathcal{P}_k}$	Set of points impacted by an isolated anomaly in $[k-1, k]$ w. r. t. $\mathcal{P}_k$ (Definition 7)
M_k	Set of points involved in a massive anomaly in $[k-1, k]$ in any anomaly partition (Section IV)
I_k	Set of points involved in an isolated anomaly in $[k-1, k]$ in any anomaly partition (Section IV)
U_k	Set of points involved in an unresolved configuration $[k-1, k]$ (Definition 8)
$\mathcal{W}_k(j)$	Family of all τ-dense motions involving j in $[k-1, k]$ (Section V)
$\overline{\mathcal{W}}_k(j)$	Family of all maximal τ-dense motions involving j in $[k-1, k]$ (Section V)
$D_k(j)$	Set of points in A_k that could belong to a τ-dense motion containing point j in $[k-1, k]$ (Section V)
$J_k(j)$	Set of points in $D_k(j)$ for which j belongs to all their maximum τ-dense motions in $[k-1, k]$ (Section V)
$L_k(j)$	Set of points in $D_k(j)$ for which j does not belong to all their maximum τ-dense motions in $[k-1, k]$ (Section V)

TABLE I
LIST OF SYMBOLS AND NOTATIONS

Similarly, the set of devices impacted by an isolated anomaly in the time interval $[k-1, k]$ is denoted by $I_{\mathcal{P}_k}$. We have $I_{\mathcal{P}_k} = \{j \in A_k \mid |\mathcal{P}_k(j)| \leq \tau\}$.
To summarize, let $\mathcal{P}_k$ be an anomaly partition, we have

$$A_k = M_{\mathcal{P}_k} \cup I_{\mathcal{P}_k} \text{ and } M_{\mathcal{P}_k} \cap I_{\mathcal{P}_k} = \emptyset. \quad (1)$$

We consider in the following that all the errors or events that occur in the system respect restrictions R1, R2 and R3. In this (ideal) context, there exists an anomaly partition that reconstructs exactly what really happens in the system. In the following we denote by $\mathcal{R}_k$, $k \geq 1$, this real scenario of errors, and by respectively $M_{\mathcal{R}_k}$ and $I_{\mathcal{R}_k}$ the set of devices that have been involved in respectively massive and isolated anomalies. We show in Theorem 3, that even in this ideal context, an omniscient observer is not always capable of building $M_{\mathcal{R}_k}$ and $I_{\mathcal{R}_k}$ if it has not access to $\mathcal{R}_k$.

IV. The Addressed Problems

Consider an omniscient observer that is able to read, at any time k, the state vector S_k, and knows for any point $j \in S$ the output of the error detection function $a_k(j)$. Based on this

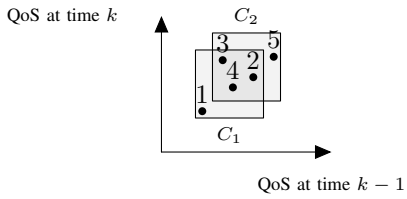


Fig. 3. A simple scenario that yields to an unresolved configuration.

knowledge, the goal of the omniscient observer is to infer the set of devices that have been involved in massive and isolated anomalies. The question that naturally crosses our mind is whether these inferred sets exactly match both $M_{\mathcal{R}_k}$ and $I_{\mathcal{R}_k}$. We reformulate this question as the Anomaly Characterization Problem (ACP). Specifically, for any $k \geq 1$, for any system states S_{k-1} and S_k, for any A_k, and $\tau \in [\![1, n-1]\!]$, let M_k and I_k be the two sets built by the omniscient observer that contained all the devices that have been impacted by respectively massive and isolated anomalies.

Problem 1 (Anomaly Characterization Problem (ACP)): Is the omniscient observer always capable of building M_k and I_k such that $M_k = M_{\mathcal{R}_k}$ and $I_k = I_{\mathcal{R}_k}$ without knowing $\mathcal{R}_k$?

In the affirmative, we say that ACP can be solved.

*Theorem 3 (*ACP *Impossibility):* ACP cannot be solved.

Proof: Proof by counterexample. We consider the scenario depicted in Figure 3 which illustrates the variation of QoS of a service consumed by five devices $S = \{1, 2, 3, 4, 5\}$ in the time interval $[k-1, k]$. Suppose that the density threshold τ is equal to 3 and that all the devices in S have abnormal trajectories. The two maximal r-consistent motions $C_1 = \{1, 2, 3, 4\}$ and $C_2 = \{2, 3, 4, 5\}$ are represented. Let $\mathcal{P}_k^1 = \{\{1, 2, 3, 4\}, \{5\}\}$ and $\mathcal{P}_k^2 = \{\{1\}, \{2, 3, 4, 5\}\}$ be the two anomaly partitions of A_k. Now, given $\tau = 3$, we have by definition that $M_{\mathcal{P}_k^1} = \{1, 2, 3, 4\}$ and $M_{\mathcal{P}_k^2} = \{2, 3, 4, 5\}$. An omniscient observer is unable to tell whether $\mathcal{P}_k^1 = \mathcal{R}_k$ or $\mathcal{P}_k^2 = \mathcal{R}_k$, and thus does not know whether $M_{\mathcal{P}_k^1} = M_{\mathcal{R}_k}$ or $M_{\mathcal{P}_k^2} = M_{\mathcal{R}_k}$, and similarly for $I_{\mathcal{R}_k}$. Thus ACP cannot be solved. ∎

We have just shown that there exist configurations that do not allow an omniscient observer to decide with certainty which devices have been impacted by massive anomalies and which ones have been impacted by isolated anomalies. We propose to relax Problem 1 by partitioning A_k into three sets M_k, I_k and U_k such that M_k and I_k contain all the devices for which it is certain that these devices have been impacted by respectively massive and isolated anomalies. We have

$$I_k = \{\ell \in A_k \mid \forall \mathcal{P}_k, |\mathcal{P}_k(\ell)| \leq \tau\} \quad (2)$$

$$M_k = \{\ell \in A_k \mid \forall \mathcal{P}_k, |\mathcal{P}_k(\ell)| > \tau\} \quad (3)$$

Thus, whatever the anomaly partition $\mathcal{P}_k$, $M_k \subset M_{\mathcal{P}_k}$ and $I_k \subset I_{\mathcal{P}_k}$. In particular $M_k \subset M_{\mathcal{R}_k}$, $I_k \subset I_{\mathcal{R}_k}$. On the other hand, set U_k contains all the other devices $j \in A_k$ for which an omniscient observer cannot decide with certainty whether j belongs to a massive anomaly or an isolated one.

Definition 8 (Unresolved configuration): Any device $j \in A_k$ is in an unresolved configuration if there exist two anomaly partitions $\mathcal{P}_k$ and $\mathcal{P}'_k$ such that $j \in I_{\mathcal{P}_k}$ and $j \in M_{\mathcal{P}'_k}$. The set of devices belonging to an unresolved configuration in the time interval $[k-1, k]$ is denoted by U_k.

We have,

Corollary 4: For any time $k \geq 1$,

$$U_k = \emptyset \Longrightarrow \text{ ACP can be solved.}$$

Proof: Suppose that $U_k = \emptyset$. By Definition 8, it means that for any j in $[\![1, n]\!]$, either j belongs to M_k or j belongs to I_k. Let us suppose that j belongs to M_k. The same argument applies if j belongs to I_k. By Relation 2, $j \in M_k \Leftrightarrow \forall \mathcal{P}_k, |\mathcal{P}_k(j)| > \tau$. In particular, $\forall \mathcal{P}_k, M_{\mathcal{P}_k} = M_{\mathcal{R}_k}$. Any execution of Algorithm 1 allows us to build an anomaly partition $\mathcal{P}_k$, and thus $M_{\mathcal{P}_k}$. ∎

We now formulate a relaxed version of ACP. Specifically, for any $k \geq 1$, for any system states S_{k-1} and S_k, for any A_k, and $\tau \in [\![1, n-1]\!]$, let M_k, I_k and U_k be respectively the set of devices involved in massive and isolated anomalies and those being in an unresolved configuration.

Problem 2 (Relaxed ACP*):* Is the omniscient observer always capable of building M_k, I_k and U_k such that

$$M_k \subseteq M_{\mathcal{R}_k} \text{ and } I_k \subseteq I_{\mathcal{R}_k} \text{ and } M_k \cup I_k \cup U_k = A_k$$

without knowing $\mathcal{R}_k$?

The following section presents necessary and sufficient conditions for any device $i \in A_k$ to belong to one of these three sets M_k, I_k and U_k.

V. Locally deciding whether one belongs to M_k, I_k, or U_k

In this section, we show how each device $j \in A_k$, $k \geq 1$, decides whether it belongs to M_k, I_k or U_k. A naive approach consisting in generating all admissible anomaly partitions and then in deciding for each device whether it belongs to M_k, I_k, or U_k is clearly impractical. Indeed, the number of these partitions is proportional to the Bell numbers, which is itself a sum of Stirling numbers of the second kind that is equal to $S(n,t) = \frac{1}{t!}\sum_{j=0}^{t}(-1)^{t-j}\binom{t}{j}j^n$, where n is the number of devices and t the number of sets of the partition. Therefore, $S(n,t)$ grows exponentially with n. We propose to solve the relaxed ACP through a cheaper and local computation which relies uniquely on the knowledge of all the maximal r-consistent motions j is involved in. Theorem 5 provides a necessary and sufficient condition (NSC) for $j \in A_k$ to belong to I_k. Theorems 6 and 7 give respectively a sufficient condition and a NSC for $j \in A_k$ to belong to M_k. Finally, Corollary 8 exhibits a NSC for $j \in A_k$ to belong to U_k.

A. A Necessary and Sufficient Condition to Belong to I_k

We introduce the following two families $\mathcal{W}_k(j)$ and $\overline{\mathcal{W}}_k(j)$ representing the family of all τ-dense motions (resp. maximal τ-dense motions) j belongs to. We have

$$\mathcal{W}_k(j) = \{B \subseteq A_k \mid j \in B \text{ and } B \text{ has a } \tau\text{-dense motion}\} \text{ and}$$

$$\overline{\mathcal{W}}_k(j) = \{B \subseteq A_k | j \in B \text{ and } B \text{ has a maximal } \tau\text{-dense motion}\}.$$

Theorem 5: For any $k \geq 1$, and for any $j \in A_k$, we have

$$\overline{\mathcal{W}}_k(j) = \emptyset \Longleftrightarrow j \in I_k.$$

Proof: ($\Rightarrow$) Let $j \in A_k$. By assumption of the theorem, $\overline{\mathcal{W}}_k(j) = \emptyset$. Now, for any subset $B \subseteq [\![1, n]\!]$ having an r-consistent motion there exists a subset $B' \subseteq [\![1, n]\!]$ such that $B \subseteq B'$ and B' is maximal (See Definition 3 and Remark 1). Thus, $\overline{\mathcal{W}}_k(j) = \emptyset \Rightarrow \mathcal{W}_k(j) = \emptyset$. Therefore, by Definition 4, j solely belongs to τ-sparse motions, and by Definition 5, for any anomaly partition $\mathcal{P}_k$, we have $|\mathcal{P}_k(j)| \leq \tau$. Therefore, by Definition 7, j can only be impacted by an isolated anomaly in the time interval $[k-1, k]$, thus $j \in I_k$.

($\Leftarrow$) We prove that $\overline{\mathcal{W}}_k(j) \neq \emptyset \Longrightarrow j \notin I_k$. Suppose that $\overline{\mathcal{W}}_k(j) \neq \emptyset$. Thus, $j \in A_k$ and $\mathcal{W}_k(j) \neq \emptyset$. Let subset B be such that $B \in \overline{\mathcal{W}}_k(j)$. We run Algorithm 1 by initially selecting j and B such that $j \in B$ and B has a maximal r-consistent motion. By Lemma 2, Algorithm 1 builds a valid anomaly partition $\mathcal{P}_k$, such that B is an element of $\mathcal{P}_k$. Thus, $\mathcal{P}_k(j) = B$. By construction of the proof, $B \in \overline{\mathcal{W}}_k(j)$, thus $|B| > \tau$. Thus we have exhibited an anomaly partition for which $\mathcal{P}_k(j) > \tau$. By Relation (2), $j \notin I_k$, which completes the proof. ∎

In the following we give a necessary and sufficient condition for device j to belong to M_k (Theorem 7). Prior to this theorem, we provide a sufficient condition for j to belong to M_k. The rationale of this weaker condition (Theorem 6) is that, from a computation point of view, it is more efficient than the NSC one and meanwhile, misses to detect that $j \in M_k$ in a very small number of scenario (simulations show that in average less than 0.4% of the scenario are not covered by Theorem 6).

B. A Sufficient Condition to Belong to M_k

We have just shown that if there are not enough devices, in the vicinity of j, that belong to dense motions, then j has necessarily been impacted by an isolated anomaly. Suppose now that $\overline{\mathcal{W}}_k(j) \neq \emptyset$, that is j belongs to a family of τ-dense motions, and denote by $D_k(j)$ the set of all these devices that belong to $\overline{\mathcal{W}}_k(j)$. We have

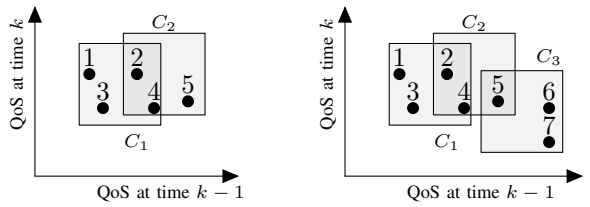
$$D_k(j) = \bigcup_{B \in \overline{\mathcal{W}}_k(j)} B.$$

We split set $D_k(j)$ into two subsets $J_k(j)$ and $L_k(j)$, such that the former one contains all the devices whose all their maximal τ-dense motions also contain j, while the latter one contains all the devices that have at least one maximal τ-dense motion that does not contain j. Notice that we have $j \in J_k(j)$ and $j \notin L_k(j)$. Formally,

$$J_k(j) = \{\ell \in A_k \mid \exists B \in \overline{\mathcal{W}}_k(j), \ell \in B \text{ and } \forall B' \in \overline{\mathcal{W}}_k(\ell),\ j \in B'\}$$

$$L_k(j) = \{\ell \in A_k \mid \exists B \in \overline{\mathcal{W}}_k(j), \ell \in B \text{ and } \exists B' \in \overline{\mathcal{W}}_k(\ell),\ j \notin B'\}.$$

Figure 4 illustrates, for device $4 \in S$ the decomposition of its neighborhood $D_k(4)$ into $J_k(4)$ and $L_k(4)$. We assume that $\forall i \in S$, $i \in A_k$ and $\tau = 2$. In Figure 4(a), $S = \{1, 2, 3, 4, 5\}$. In this configuration, we have $\overline{\mathcal{W}}_k(4) = \{\{1, 2, 3, 4\}, \{2, 4, 5\}\}$ and thus $D_k(4) = \{1, 2, 3, 4, 5\}$. By definition, device $4 \in J_k(4)$. Devices $1, 3$ have a single maximal τ-dense motion $C_1 = \{1, 2, 3, 4\}$, and C_1 contains 4. Thus $1, 3 \in J_k(4)$. In the same way, device 5 has also a single maximal τ-dense motion $C_2 = \{1, 2, 3, 4, 5\}$, C_2 contains 4. Thus $5 \in J_k(4)$. Finally, by applying the same argument for device 2, we get $2 \in J_k(4)$. Putting altogether, we have $J_k(4) = \{1, 2, 3, 4, 5\}$ and $L_k(4) = \emptyset$. In Figure 4(b), $S = \{1, 2, 3, 4, 5, 6, 7\}$. Device 5 belongs to both $C_2 = \{2, 4, 5\}$ and $C_3 = \{5, 6, 7\}$, while device 4 does not belong to C_3. Thus $J_k(4) = \{1, 2, 3, 4\}, L_k(4) = \{5\}$.



(a) $D_k(4) = \{1, 2, 3, 4, 5\}$, $J_k(4) = \{1, 2, 3, 4, 5\}$, and $L_k(4) = \emptyset$, with $\tau = 2$

(b) $D_k(4) = \{1, 2, 3, 4, 5\}$, $J_k(4) = \{1, 2, 3, 4\}$, and $L_k(4) = \{5\}$, with $\tau = 2$

Fig. 4. Splitting the neighborhood of device 4 into $J_k(4)$ and $L_k(4)$.

Based on this neighborhood division, we enunciate the following theorem.

Theorem 6: For any $k \geq 1$ and for any $j \in A_k$,

$$\exists B \in \mathcal{W}_k(j) \text{ such that } B \subseteq J_k(j) \Longrightarrow j \in M_k$$

Proof: The proof is done by contradiction. Let B be a set such that $B \in \mathcal{W}_k(j)$ and $B \subseteq J_k(j)$. Suppose that there exists an anomaly partition $\mathcal{P}_k = \{B_1, \ldots, B_\ell\}$ such that $j \in I_{\mathcal{P}_k}$. Two cases must be considered.

1) Let $\ell \in B$ be such that $|\mathcal{P}_k(\ell)| > \tau$. By Definition 6, $\mathcal{P}_k(\ell)$ has a τ-dense motion, thus $\mathcal{P}_k(\ell) \in \mathcal{W}_k(\ell)$. By Remark 1, let L be a set of $\overline{\mathcal{W}}_k(\ell)$ such that $\mathcal{P}_k(\ell) \subseteq L$. By assumption of the theorem, $B \subseteq J_k(j)$, and thus, for any set of $\overline{\mathcal{W}}_k(\ell)$, j belongs to this set, and in particular j belongs to L. By assumption of the proof, $j \in I_{\mathcal{P}_k}$, therefore $j \notin \mathcal{P}_k(\ell)$. Therefore, we have $\mathcal{P}_k(\ell) \cup \{j\} \subseteq L$ and so $\mathcal{P}_k(\ell) \cup \{j\}$ has an r-consistent motion, and its size is greater than $\tau + 1$. Hence $\mathcal{P}_k(\ell) \cup \{j\}$ is τ-dense, which contradicts condition C2 of Definition 6.

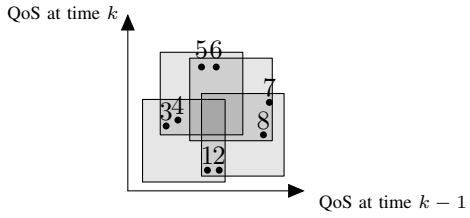


Fig. 5. Configuration where device $j \in [\![1,8]\!]$ belongs to M_k while one cannot build a dense motion with devices in $J_k(j)$. Settings: $\tau = 3$.

2) Suppose now that for any ℓ in $B, |\mathcal{P}_k(\ell)| \leq \tau$. We have $B \subseteq \bigcup_{\ell \in B} \mathcal{P}_k(\ell) \subseteq \bigcup_{|B_i| \leq \tau} B_i$. By assumption, $B \in \mathcal{W}_k(j)$. Therefore, B is a τ-dense motion, which contradicts condition C1 of Definition 6.

Both contradictions conclude the proof. ∎

We now present a necessary and sufficient condition that finds all the devices that belong to M_k. Note that this condition is more intricate and requires substantially more computation than the necessary one exhibited in Theorem 6.

C. Necessary and Sufficient Condition to Belong to M_k

We first give an intuition of this condition by exhibiting the type of scenario that Theorem 6 does not cover. Figure 5 shows a system $S = \{1,2,3,4,5,6,7,8\}$ where each device in S belongs to A_k. Suppose that $\tau = 3$. We focus on device 1. The same argument holds for the other devices by symmetry of the system states. We have $\overline{\mathcal{W}}_k(1) = \{\{1,2,3,4\},\{1,2,7,8\}\}$ and $D_k(1) = \{1,2,3,4,7,8\}$. By definition, $J_k(1) = \{1,2\}$ and $L_k(1) = \{3,4,7,8\}$. As $|J_k(1)| < \tau$, there are no τ-dense motions made of devices of $J_k(1)$. Thus the sufficient condition of Theorem 6 does not hold. Nevertheless, device 1 belongs to M_k as for any anomaly partition $\mathcal{P}_k$ of A_k, we have $|\mathcal{P}_k(1)| > \tau$. Indeed, the only two anomaly partitions $\mathcal{P}_k$ and $\mathcal{P}'_k$ of A_k for $\tau = 3$ are respectively equal to $\{\{1,2,7,8\},\{3,4,5,6\}\}$ and $\{\{1,2,3,4\},\{5,6,7,8\}\}$. In both cases, we have $|\mathcal{P}_k(1)| = |\mathcal{P}'_k(1)| > \tau$. Thus we have a configuration where device 1 belongs to M_k while one cannot build a dense motion with devices in $J_k(1)$. The following theorem provides a necessary (and sufficient) condition for a device to belong to M_k.

Theorem 7: For any time $k \geq 1$ and for any $j \in A_k$, $j \in M_k$ if and only if $\overline{\mathcal{W}}_k(j) \neq \emptyset$ and for all collections $\mathcal{C}$ of pairwise disjoint sets defined by $\mathcal{C} \subseteq \{B \in \mathcal{W}_k(\ell) \mid \ell \in L_k(j), j \notin B\}$ one of the following two relations holds:

$$\exists A \in \mathcal{W}_k(j) : A \subseteq D_k(j) \setminus \bigcup_{B \in \mathcal{C}} B, \tag{4}$$

$$\exists B \in \mathcal{C} : B \cup \{j\} \in \mathcal{W}_k(j). \tag{5}$$

Proof: For both senses of the equivalence, their contraposive is proven.

- Suppose that $j \notin M_k$. This means that there exists at least one anomaly partition $\mathcal{P}_k = \{B_1, \ldots, B_\ell\}$ such that $|\mathcal{P}_k(j)| \leq \tau$. Consider the collection $\mathcal{C}_1$ defined by $\mathcal{C}_1 = \{B \in \mathcal{P}_k \mid |B| > \tau\}$. By definition of an anomaly partition, we have $\forall B, B' \in \mathcal{C}_1, B \cap B' = \emptyset$. Now, by condition C2 of Definition 6, $\forall B \in \mathcal{C}_1, B \cup \{j\}$ is not a r-consistent motion. As a consequence, $B \cup \{j\} \notin \mathcal{W}_k(j)$. Moreover, by definition of $\mathcal{C}_1$, $\forall \ell \in D_k(j) \setminus \bigcup_{B \in \mathcal{C}_1} B$, we have $|\mathcal{P}_k(\ell)| \leq \tau$. Thus, $D_k(j) \setminus \bigcup_{B \in \mathcal{C}_1} B \subseteq \bigcup_{|B_i| \leq \tau} B_i$. By condition C1 of Definition 6, $\forall B \subseteq \bigcup_{|B_i| \leq \tau} B_i$, B is not a τ-dense motion, and thus $B \notin \mathcal{W}_k(j)$. Hence, we have $\forall A \in \mathcal{W}_k(j), A \nsubseteq D_k(j) \setminus \bigcup_{B \in \mathcal{C}_1} B$, which concludes the first part of the proof.
- We prove the second contraposive.
  1) Suppose that $\mathcal{W}_k(j) = \emptyset$. By Theorem 5, $j \in I_k$ and thus we have $j \notin M_k$.
  2) Suppose that $\mathcal{W}_k(j) \neq \emptyset$ and $\exists \mathcal{C}_1 \subseteq \{B \in \mathcal{W}_k(\ell) \mid \ell \in L_k(j) \wedge j \notin B\}$ satisfying $[B \cap B' = \emptyset, \forall B, B' \in \mathcal{C}_1]$ such that the following two relations hold:

$$\forall A \in \mathcal{W}_k(j), A \nsubseteq D_k(j) \setminus \bigcup_{B \in \mathcal{C}_1} B, \tag{6}$$

$$\forall B \in \mathcal{C}_1, B \cup \{j\} \notin \mathcal{W}_k(j). \tag{7}$$

  Consider an anomaly partition $\mathcal{P}_k$ such that $\forall B \in \mathcal{C}_1, B \in \mathcal{P}_k$. By Relation (6), j cannot belong to a dense motion compounded of points that are not in an element of $\mathcal{C}_1$. Moreover, by relation (7), j cannot belong to any dense motion of $\mathcal{C}_1$. Thus, by construction of $\mathcal{P}_k$, we have $|\mathcal{P}_k(j)| \leq \tau$ and thus $j \notin M_k$, which completes the proof.

∎

Coming back to the example shown in Figure 5, we have $L_k(1) = \{3,4,7,8\}$, and thus the family of sets $\{B \in \mathcal{W}_k(\ell) \mid \ell \in L_k(1) \wedge 1 \notin B\}$ is equal to $\{\{3,4,5,6\},\{5,6,7,8\}\}$. Two cases need to be considered.

1) $B = \{3,4,5,6\}$. Then $D_k(1) \setminus \left(\bigcup_{B \in \mathcal{C}} B\right) = \{1,2,5,6\}$ which is a τ-dense motion and thus device 1 belongs to a massive anomaly in this configuration.
2) $B = \{5,6,7,8\}$. Then $D_k(1) \setminus \left(\bigcup_{B \in \mathcal{C}} B\right) = \{1,2,3,4\}$ which is a τ-dense motion and thus 1 belongs to a massive anomaly in this configuration.

Therefore, as there are no other configuration satisfying $[B \cap B' = \emptyset, \forall B, B' \in \mathcal{C}]$, all collections have been tested and thus device $1 \in M_k$.

D. A Necessary and Sufficient Condition to belong to U_k

We end this section, by enunciating a corollary that derives from both Theorems 6 and 7. This corollary gives a necessary and sufficient condition for a device to belong to an unresolved configuration.

Corollary 8: For all time $k \geq 1$ and $j \in A_k$, $j \in U_k$ if and only if $\overline{\mathcal{W}}_k(j) \neq \emptyset$ and it exists a collection $\mathcal{C}$ of pairwise disjoint sets defined by $\mathcal{C} \subseteq \{B \in \mathcal{W}_k(\ell) \mid \ell \in L_k(j), j \notin B\}$ such that the following two conditions hold:

$$\forall A \in \mathcal{W}_k(j) : A \nsubseteq D_k(j) \setminus \bigcup_{B \in \mathcal{C}} B, \text{ and}$$
$$\forall B \in \mathcal{C} : B \cup \{j\} \notin \mathcal{W}_k(j). \tag{8}$$

Proof: Straightforward from case 2) of the proof of Theorem 7. ∎

To summarize, we have derived conditions that allow any impacted device to decide whether many other devices have been

Algorithm 2: j.maxMotions($N(j), i, \ell, \mathcal{M}(j)$)

Data: $N(j)$: set of devices whose positions are at no more than distance $2r$ from j in E at both time $k-1$ and time k; $\mathcal{M}(j)$: family of maximal r-consistent motions j belongs to; i: current dimension; $\ell = 0$ current configuration, $\ell = 1$ previous one.
Output : $\mathcal{M}(j)$

begin
  if $i > d$ and $\ell = 0$ **then**
    $\ell \leftarrow \ell + 1$;
    $i \leftarrow 0$;
  $i \leftarrow i + 1$;
  if $N(j) \notin \mathcal{M}(j)$ and $i \leq d$ and $\ell \leq 1$ **then**
    $x_{set} \leftarrow \{q_{i,k-\ell}(x) | x \in N(j)\}$;
    $x_m \leftarrow \min(x_{set})$;
    $N(j) \leftarrow \{x \in N | x_m \leq q_{i,k-\ell}(x) \leq x_m + 2r\}$;
    while $x_{min} < q_{i,k-\ell}(j)$ and $N(j) \neq \emptyset$ **do**
      $\mathcal{M}(j) \leftarrow$j.maxMotions(N(j),i,$\ell$,$\mathcal{M}(j)$);
      $x_{set} \leftarrow x_{set} \setminus \{x_m\}$;
      $x_m \leftarrow \min(x_{set})$;
      $N \leftarrow \{x \in N(j) | x_m \leq q_{i,k-\ell}(x) \leq x_m + 2r\}$;
  else if $\forall M \in \mathcal{M}(j), N(j) \not\subseteq M$ **then**
    $\mathcal{M}(j) \leftarrow \{M \in \mathcal{M}(j) | M \not\subseteq N(j)\}$;
    $\mathcal{M}(j) \leftarrow \mathcal{M}(j) \cup$ {N(j)};
  return $\mathcal{M}(j)$;

Algorithm 3: j.characterize()

Data: $N(j) = N_k(j) \cap N_{k-1}(j)$: set of devices whose positions are no more than distance $2r$ from j in E at both time $k-1$ and k; τ: density threshold.
Output : Type of the anomaly impacting j (*i.e.,* I, M, or U

begin
  $\mathcal{M}(j) \leftarrow$ j.maxMotions($N(j)$,0,0,{});
  $\overline{\mathcal{W}}_k(j) \leftarrow \{M \in \mathcal{M}(j) \mid |M| > \tau\}$;
  if $\overline{\mathcal{W}}_k(j) = \emptyset$ **then**
    anomaly$\leftarrow$ Isolated;
  else
    $J \leftarrow \emptyset$;
    $L \leftarrow \emptyset$;
    for $M \in \overline{\mathcal{W}}_k(j)$ **do**
      for $\ell \in M$ **do**
        $\mathcal{M}_\ell \leftarrow \ell$.maxMotions($N(\ell)$,0,0,{}) ;
        $\overline{\mathcal{W}}_k(\ell) \leftarrow \{M' \in \mathcal{M}_\ell \mid |M'| > \tau\}$;
        if $\exists M' \in \overline{\mathcal{W}}_k(\ell)$ such that $j \notin M'$ **then**
          $L \leftarrow L \cup \{\ell\}$;
        else
          $J \leftarrow J \cup \{\ell\}$;
    if $\exists M \in \overline{\mathcal{W}}_k(j)$ such that $|M \cap J| > \tau$ **then**
      anomaly$\leftarrow$ Massive;
    else
      anomaly$\leftarrow$ Unresolved;
  return anomaly;

Algorithm 4: j.fullcharacterize()

$S \leftarrow \bigcup_{\ell \in L, B \in \overline{\mathcal{W}}_k(\ell)} B$;
$S \leftarrow S \setminus \{j\}$;
$\mathcal{C} \leftarrow \{\}$;
$R \leftarrow L$;
while $R \neq \emptyset$ and $\neg$ j.check($\mathcal{C}$) **do**
  Take any $\ell \in R$;
  $R \leftarrow R \setminus \{\ell\}$;
  $\mathcal{C} \leftarrow$j.isolate(S,$L \setminus \{\ell\}$,$\mathcal{C}$,ℓ);
if j.check($\mathcal{C}$) **then**
  anomaly$\leftarrow$ Unresolved;
else
  anomaly$\leftarrow$ Massive;

impacted by the very same error or not. We have shown that the concomitance of errors may lead to unresolved scenarios that do not allow devices to distinguish which error they have been impacted by. Finally, we have shown that each device j only needs to know the trajectories of its neighbors (*i.e.*, the devices that belong to j maximal r-consistent motions), and possibly the trajectories of the neighbors of the devices that belong to $L_k(j)$. Thus j only needs to know the trajectories that are at no more than $4r$ from itself. A larger radius of knowledge – as the one got by an omniscient observer that samples at each time k the system state, *i.e.*, S_k – does not bring any additional information and thus does not provide a higher error detection accuracy. The following two sections present respectively the local algorithms run by impacted devices, and the performance of these algorithms in terms of accuracy and complexity through extensive simulations.

VI. Algorithms

This section presents the local algorithms implementing Theorems 5, 6, and 7 and Corollary 8.

From an algorithmic point of view, the determination of maximal r-consistent motions allows us to efficiently derive all the sets or families of sets needed to determine for each device $j \in A_k$ whether it belongs to M_k, I_k, or to an unresolved configuration U_k. Algorithm 2 presents the pseudo-code run by any device $j \in A_k$ to build the family of maximal r-consistent motions j belongs to. Let $\mathcal{M}(j)$ be this family. The set $N(j) \subseteq [\![1, n]\!]$ contains all the devices whose positions are at no more than distance $2r$ from j in E at both time $k-1$ and time k. $N(j)$ is initialized to $N_k(j) \cap N_{k-1}(j)$. We denote by $S_k(j)$ and $S_{k-1}(j)$ the sets of their respective positions. Recall that E is a d-dimensional space. The core of the algorithm lies in moving, along each of the d dimensions, two sliding-windows W_{k-1} and W_k of width $2r$ with dimension d (*i.e.*, hyper-rectangle of dimension d) over all the points in respectively $S_{k-1}(j)$ and $S_k(j)$ and in updating progressively $\mathcal{M}(j)$ with the new set of points B covered by the sliding-windows. Specifically, the position of a device ℓ in $S_k(j)$ is given by $p_k(\ell) = (q_{1,k}(\ell), q_{2,k}(\ell), \ldots, q_{d,k}(\ell))$. Sliding-window W_k is initially positioned at pivot p_0 at position $p_k(p_0) = (x_1, \ldots, x_d)$ with $\forall i \in [\![1, d]\!], x_i = \min_{\ell \in N_k(j)} q_{i,k}(\ell)$. The same construction applies for the initial positioning of W_{k-1} at pivot $p_{k-1}(0)$. $\mathcal{M}(j)$ is fed with the set of points that are covered by both sliding-windows W_k and W_{k-1}. Then W_k is moved to its next pivot $p1$ whose

Algorithm 5: j.isolate(S,L,$\mathcal{C}$,ℓ)

Data: S : set of devices that belong to a dense motion containing a device of $L_k(j)$
Output : A collection $\mathcal{C}$ that satisfies Relation 8 of Corollary 8

```
1  begin
2      for M ∈ {M ∈ W̄_k(ℓ) | j ∉ M} do
3          s ← |M ∩ S \ ⋃_{B_i∈C} B_i|;
4          while s > τ do
5              for B ∈ {B ⊆ M ∩ S | |B| = s} do
6                  C ← C ∪ {B};
7                  Take any m ∈ L;
8                  if L \ {m} ≠ ∅ then
9                      C ←j.isolate(S,L \ {m},C, m);
10                 if j.check(C) then
11                     return C;
12             s ← s − 1;
13     return C;
```

position is given by $p_k(p_1) = (p_{1,k}(p_1), x_2, \ldots, x_d)$ with $p_{1,k}(p_1) = \min_{\ell \in N_k(j)}\{q_{1,k}(\ell) \,|\, q_{1,k}(\ell\} > p_{1,k}(p_0))$, and the set of points that are covered by both sliding-windows W_k and W_{k-1} are appended to $\mathcal{M}(j)$. Note that if this new set B includes an existing one B' in $\mathcal{M}(j)$ then B replaces B' (lines 17–19). Sliding-window W_k continues to move along the first dimension until either the first coordinate of the pivot is equal to $p_{1,k}(j)$ or the intersection between W_k and W_{k-1} is empty. At this point, W_k is moved to its next position along the second dimension and the same process is reiterated until W_k has been moved along the d dimensions. If $N_k(j) \cap N_{k-1}(j) \notin \mathcal{M}(j)$ then sliding-window W_{k-1} is moved to its next pivot (which is computed as for W_k) and W_k is re-positioned to its initial pivot $p_k(0)$. This is achieved in a recursive way.

Algorithm 3 presents the pseudo-code run by any device $j \in A_k$ to determine whether it has been impacted by an isolated anomaly (lines 4–5, direct application of Theorem 5), by a massive one (lines 6–21, direct application of Theorem 6) or whether it is in an unresolved configuration (line 23). In the latter case, if j wants to know whether it has impacted by a massive anomaly then it runs the necessary and sufficient condition presented in Theorem 7. This consists in replacing line 20 of Algorithm 3 by the pseudo code presented in Algorithm 4. This algorithm iterates over all the devices in $L_k(j)$ and looks for a collection of dense motions containing devices in $L_k(j)$ (procedure isolate()) until one satisfies Relation 8 of Corollary 8 which is achieved by procedure check($\mathcal{C}$).

VII. Performance Evaluation

A. Simulation settings

This section is devoted to the performance study of our algorithms. Four main metrics have been analyzed. The complexity in time has been evaluated by measuring the average number of operations executed per device to be able to take a decision (*i.e.*, Isolated/Massive/Unresolved). Then the effectiveness of Theorem 6 with respect to Theorem 7 has been analyzed by measuring the percentage of massive anomalies that Theorem 6 misses to detect. The adaptivity of our algorithms to various sampling frequencies has been studied. Finally, the pertinence of our model with respect to a ground truth has been evaluated by measuring the number of isolated errors that are considered as massive ones due to restriction R3 of the model. All these results have been obtained by running extensive simulations. We have tested around $10,000$ different settings of the following parameters: n, the number of devices in the system, τ the threshold that distinguishes massive from isolated anomalies, and r the consistency impact radius. The number of services d accessed by each devices has been set to 2, leading to a 2-dimensional QoS space E. The initial distribution of the devices in E follows a uniform distribution, denoted by S_0. Then for each discrete time $k \geq 1$, configuration S_k is generated as follows. A number A of points with $A \in [\![1, 80]\!]$ are randomly chosen in S_{k-1}. Then, for each chosen point j, with probability G less than τ points are randomly chosen in a ball of radius r centered at j, and with probability $1 - G$, t points are randomly chosen in a ball of radius r centered at j, with t varying from τ to the number of points in this ball. This allows to respectively simulate isolated and network errors. In both cases, all these chosen points ℓ are moved to another location uniformly chosen in E, and $a_k(\ell)$ is set to `True`.

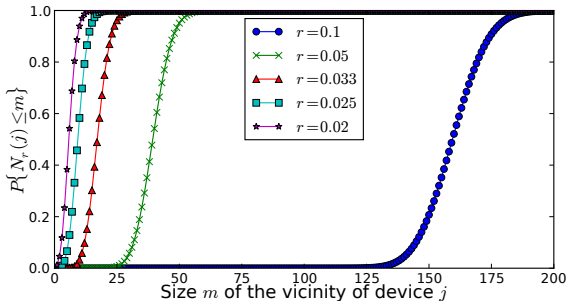
We now precise how both parameters r and τ are selected. As previously said in Section III-C, if strictly less than $\tau + 1$ devices have an abnormal trajectory due to the same error then none of these devices can belong to a τ-dense motion. In other words, we need to tune parameters r and τ in such a way that the probability of having more than τ errors impacting devices, which are at no more than $2r$ from each other, is negligible. We denote by $N_r(j)$ the random variable representing the number of devices in the vicinity of device j and by $F_r(j)$ the random variable equal to the number of devices impacted by an isolated error in the vicinity of device j. We have $\mathbb{P}\{N_r(j) = i\} = \binom{n-1}{i} q_j^i (1-q_j)^{n-1-i}$ with q_j the probability that a device ℓ is in the vicinity of device j. Given the position $p(j)$ of device j, the vicinity $V \subset E$ of device j is defined by $V = \{x \in E \mid \|x - p(j)\| \leq 2r\}$. We are interesting in computing $\mathbb{P}\{F_r(j) > \tau\}$. We have

$$\begin{aligned}
\mathbb{P}\{F_r(j) > \tau\} &= 1 - \sum_{\ell=0}^{\tau} \mathbb{P}\{F_r(j) = \ell\} \\
&= 1 - \sum_{m=0}^{n-1} \sum_{\ell=0}^{\tau} \mathbb{P}\{F_r(j) = \ell \mid N_r(j) = m\} \mathbb{P}\{N_r(j) = m\} \\
&= 1 - \sum_{m=0}^{n-1} \sum_{\ell=0}^{\tau} \binom{m}{\ell} b^\ell (1-b)^{(m-\ell)} \mathbb{P}\{N_r(j) = m\}.
\end{aligned}$$

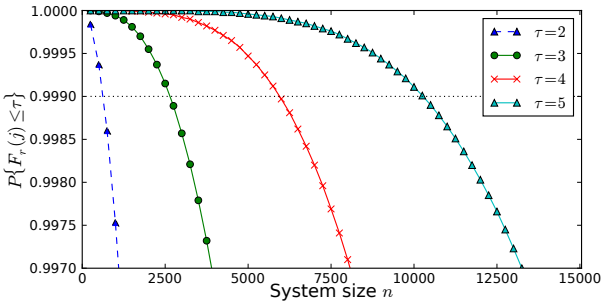
where b is the probability that an isolated error impacts a device in the time interval $[k-1, k]$. This leads to

$$\begin{aligned}
&\mathbb{P}\{F_r(j) > \tau\} \\
&= 1 - \sum_{m=0}^{n-1} \sum_{\ell=0}^{\tau} \binom{m}{\ell} b^\ell (1-b)^{(m-\ell)} \binom{n-1}{m} q_j^m (1-q_j)^{n-1-m}.
\end{aligned}$$

Now, given a small constant ε, r and τ are tuned so that the probability of having more than τ independent errors that impact close devices is negligible, that is $\mathbb{P}\{F_r(j) \leq$



(a) $\mathbb{P}\{N_r(j) \leq m\}$ as a function of m and different values of r. We have $n = 1000$.



(b) $\mathbb{P}\{F_r(j) \leq \tau\}$ as a function of the system size n and different values of τ. We have $r = 0.03$ and $b = 0.005$

Fig. 6. Dimensioning parameters r and τ.

$\lvert I_k \rvert$ with Theorem 5	$\lvert M_k \rvert$ with Theorem 6	$\lvert U_k \rvert$ with Corollary 8	$\lvert M_k \rvert$ with Theorem 7
2.54%	88.34%	8.72%	0.4%

TABLE II
AVERAGE REPARTITION OF POINTS OF A_k IN EACH SET I_k,M_k AND U_k FOR $A = 20$, $n = 1000$, $r = 0.03$, $\tau = 3$ AND $|A_k| = 95.7$.

$\lvert I_k \rvert$ with Theorem 5	$\lvert M_k \rvert$ with Theorem 6	$\lvert U_k \rvert$ with Corollary 8	$\lvert M_k \rvert$ with Theorem 7
1.85	1.17	31,107.9	2,450,150

TABLE III
AVERAGE COMPUTATIONAL COST FOR EACH DEVICE IN I_k,M_k,U_k FOR $A = 20$, $n = 1000$ $r = 0.03$, $\tau = 3$ AND $|A_k| = 95.7$.

$\tau\} < 1 - \varepsilon$. Figure 6(a) plots the curve of the cumulative distribution function of $N_r(j)$ as a function of the size m of j neighborhood and for different values of r. This curve clearly illustrates the impact of r on the size m of j neighborhood. An interesting value for a total population of $n = 1000$ devices is $r = 0.03$ which guarantees that for a value of m logarithmic in the size of the population of the system, the probability of having more than τ independent errors that impact close devices is negligible. Figure 6(b) plots $F_r(j)$ as a function of the system size n when r is set to 0.03. In the following $r = 0.03$ and $\tau = 3$.

B. What brings Theorem 7 with respect to Theorem 6

We have derived in Theorem 7 a necessary and sufficient condition for any device $j \in A_k$ to decide with certainty that it has been impacted by a massive error. This condition requires first that j builds, for all the devices in $L_k(j)$, all the sets of collections of disjoint r-consistent motions. Then, for each set of these collections, j verifies that there always exists a dense motion containing j such that none of the elements of this dense motion can belong to one of these collections. The question that naturally comes up is whether this computation complexity is worth regarding the performance of the sufficient condition of Theorem 6. To answer this question, we have generated configurations of errors that maximize the number of devices that exhibit massive anomalies. This has been achieved by setting the probability G that an isolated error impacts a device to a small constant ε. Table II and Table III summarize the main obtained trends. Table II provides the repartition of each set I_k, M_k (detected by Theorem 6 and by Theorem 7), and U_k for $A = 20$ generated errors and $|A_k| = 100$ impacted devices. The main result is that in average no more than 0.4% of devices impacted by massive anomalies are missed by Theorem 6. This result is very interesting given the computational cost incurred by the NSC of Theorem 7. In Table III, the cost corresponding to column I_k represents the average number of maximal motions that device $j \in I_k$ belongs to. For the second column, the cost represents the number of maximal dense motions that device $j \in M_k$ belongs to. For the third column, the cost represents the average number of tested collections of dense motions containing the devices in $L_k(j)$, while the fourth one represents all the collections of dense motions containing the devices in $L_k(j)$.

C. Granularity of the snapshots

Any online detection system should be able to quickly identify the presence of isolated or massive anomalies to rapidly fix or confine the events or errors that lead to these anomalies. Typically, this largely depends on the frequency at which the system can sample the QoS information of the devices it monitors. In our approach, the frequency of QoS information sampling is locally tuned, and only depends on the local occurrence of anomalies. Thus by avoiding any kind of global synchronization, devices can efficiently provide a fine grain event/errors detection without impacting the rest of the system. An enjoyable consequence of this local tuning is that devices can afford to increase the frequency at which they sample their neighborhood, decreasing accordingly the number of concomitant errors and thus the number of unresolved configurations. This is illustrated in Figure 7. This figure shows the percentage of unresolved configurations as a function of the number of errors generated between two snapshots of the system and the type of errors (that is when $G = 1$, only isolated errors are generated, while for

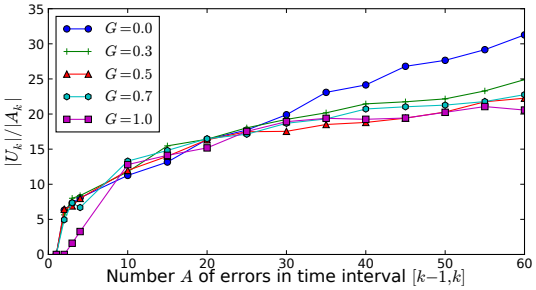


Fig. 7. Ratio $|U_k|/|A_k|$ as a function of A and G. We have : $n = 1000$ and $b = 0.005$.

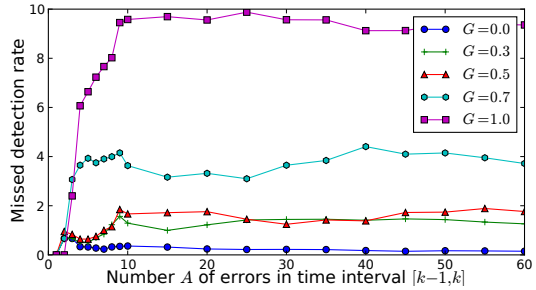


Fig. 8. Proportion of missed detection as a function of A and G when restriction R3 does not hold. We have : $n = 1000$ and $b = 0.005$.

$G = 0$ up to all the devices in the vicinity of an impacted device can be impacted). This figure confirms the fact that when a single error (isolated or massive) is generated then no unresolved configurations exists. Now for an increasing number of errors, the number of unresolved configurations augments. Note that the impact of massive errors is more significant on the number of unresolved configurations because it increases the number of configurations where a device can belong to several maximal dense motions.

D. Pertinence of Restriction R3

We finally show that the impact of Restriction R3 on the accuracy of I_k and M_k is relatively weak. Indeed, to model the impact of errors, we have assumed that if a device belongs to a τ-dense motion then this device has necessarily been impacted by an error that has impacted many other devices. While relevant in many situations, we show in the following the proportion of devices for which the second part of Restriction R3 does not hold. Figure 8 shows the proportion of devices that claim to have been impacted by a massive error (in accordance with our model) although it was an isolated one as a function of the frequency of the snapshots (represented by the number of generated errors between two snapshots). This figure shows that in the worst case, this proportion is less than 10%, and it remains constant whatever the number of errors. Which is an interesting result. Finally, Figure 9 shows that Restriction R3 has no impact on the number of unresolved configurations, which comes from the fact that unresolved configurations are essentially due to the superposition of massive errors.

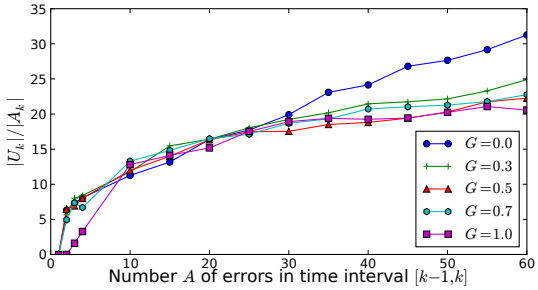


Fig. 9. Ratio $|U_k|/|A_k|$ as a function of A and G when restriction R3 does not hold. We have : $n = 1000$ and $b = 0.005$.

VIII. Conclusion

This paper has been devoted to the online detection of errors or events in large scale systems according to the extent of their damage. We have proposed a new approach that fully relies on the local knowledge of each impacted device to provide the monitoring application the essential information that should help them to be continuously aware of the state of the system. This has been achieved by modeling the impact of errors on devices as consistent and close trajectories in a QoS space. We have derived necessary and sufficient conditions locally applicable. We have validated the pertinence of our model by comparing the output of our algorithms with a large spectrum of scenarios of errors. Finally, by design, our approach is scalable. As future work, we plan to extend our characterization to take into account malicious devices. In particular, we will study the presence of collusion of malicious devices whose aim would be to prevent an impacted device to be detected by the monitoring application.

References

[1] E. Anceaume, R. Ludinard, E. Le Merrer, B. Sericola, and G. Straub. FixMe: A Self-organizing Isolated Anomaly Detection Architecture for Large Scale Distributed Systems. In *Proc. of the 16th International Conference On Principles Of Distributed Systems (OPODIS)*, 2012.

[2] D. R. Choffnes, F. E. Bustamante, and Z. Ge. Crowdsourcing service-level network event monitoring. In *SIGCOMM*, pages 387–398, 2010.

[3] A. Desphand, E. Guestrin, and S. Madden. Model-driven data acquisition in sensor networks. In *Proc. of the International Conference on Very Large Databases (VLDB)*, 2002.

[4] B. Forum. TR-069 CPE WAN Management Protocol, 2011. Issue 1, Amend.4.

[5] S. Har-Peled and B. Sadri. How fast is the k-means method? *Algorithmica*, 41(3):185–202, 2005.

[6] C. C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5–10, 2004.

[7] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[8] S. Krishnamurthy, T. He, G. Zhou, J. A. Stankovic, and S. H. Son. RESTORE: A Real-time Event Correlation and Storage Service for Sensor Networks. In *Proc. of the International Conference on Network Sensing Systems (INSS)*, 2006.

[9] K. Mouratidis, D. Papadias, S. Bakiras, and Y. Tao. A Threshold-Based Algorithm for Continuous Monitoring of K Nearest Neighbors. *IEEE Transactions on Knowledge and Data Engineering*, 17(11):1451–1464, 2005.

[10] E. S. Page. Continuous Inspection Schemes. *Biometrika*, 41(1/2):100–115, June 1954.

[11] M. C. Vuran and I. F. Akyildiz. Spatial correlation-based collaborative medium access control in wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 14(2):316–329, 2006.

[12] P. R. Winters. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6:324–342, 1960.

[13] X. Xiong, M. Mokbel, and W. Aref. SEA-CNN: Scalable Processing of Continuous K-Nearest Neighbor Queries in Spatio-Temporal Databases. In *Proc. of the IEEE International Conference on Data Engineering (ICDE)*, 2005.

[14] Z. Zhang, Y. Yang, A. K. H. Tung, and D. Papadias. Continuous k-means monitoring over moving objects. *IEEE Transactions on Knowledge and Data Engineering*, 20(9):1205–1216, 2008.

[15] Y. Zhao, Y. Tan, Z. Gong, X. Gu, and M. Wamboldt. Self-correlating predictive information tracking for large-scale production systems. In *Proc. of the International Conference on Autonomic Computing (ICAC)*, 2009.