Data Fusion & Type-2 Fuzzy Inference in Contextual Data Stream Monitoring

Kostas Kolomvatsos[1], Christos Anagnostopoulos[2], Stathes Hadjiefthymiades[3]
[1] Department of Computer Science, University of Thessaly, 35100, Lamia, Greece
[2] School of Computing Science, University of Glasgow, G12 8QQ, UK
[3] Department of Informatics and Telecommunications, University of Athens, 15784, Athens, Greece
e-mails: [1]kolomvatsos@cs.uth.gr, [2]Christos.Anagnostopoulos@glasgow.ac.uk, [3]shadj@di.uoa.gr

***Abstract*— Data stream monitoring provides the basis for building intelligent context-aware applications over contextual data streams. A number of wireless sensors could be spread in a specific area and monitor contextual parameters for identifying phenomena e.g., fire or flood. A back-end system receives measurements and derives decisions for possible abnormalities related to negative effects. We propose a mechanism, which based on multivariate sensors data streams, provides real-time identification of phenomena. The proposed framework performs contextual information fusion over consensus theory for the efficient measurements aggregation while time-series prediction is adopted to result future insights on the aggregated values. The unanimous fused and predicted pieces of context are fed into a Type-2 fuzzy inference system to derive highly accurate identification of events. The Type-2 inference process offers reasoning capabilities under the uncertainty of the phenomena identification. We provide comprehensive experimental evaluation over real contextual data and report on the advantages and disadvantages of the proposed mechanism. Our mechanism is further compared with Type-1 fuzzy inference and other mechanisms to demonstrate its false alarms minimization capability.**

Keywords*— *Contextual data stream monitoring; Phenomena identification; Data fusion; Type-2 Fuzzy Sets.

I. INTRODUCTION

A. Motivation

A *Wireless Sensor Network* (WSN) consists of distributed wirelessly connected sensors capable of sensing (observing) specific phenomena. Each sensor has specific sensing abilities for performing measurements related to a phenomenon (e.g., fire, flood). The most important advantage of WSNs is the autonomous nature of sensors. When deployed in a field of interest (e.g., a forest), they can automatically perform measurements and disseminate them to their spatial neighbors until reaching a central processing system, hereinafter referred to as *Monitoring System* (MS). An MS collects contextual data and processes them to *(i)* identify certain phenomena and then *(ii)* react to specific events. These events are related to critical aspects like violations of pre-defined constraints.

In security applications, a monitoring infrastructure is imperative by adopting an MS which applies a fast and efficient mechanism to derive alerts when specific criteria are met [1], [2]. Such criteria are related to sensor failures, resources depletion, or other abnormalities. It is of paramount importance the identification of failures to be realized in (near) real time as time-critical applications require immediate responses. For instance, imagine an MS of a power plant where multiple contextual data streams, received by a set of sensors, are aggregated and processed to derive security alerts. Another interesting application domain is environmental monitoring [3], [4], [5]. Environmental monitoring has attracted significant interest as negative effects in the environment heavily affect human lives. Changes in the environment should be immediately identified and decisions should be taken to secure the minimum quality of living for humans. The key aspect of an MS is to be *pro-active* and should immediately respond to any change in the environment. Many MSs adopt *(i)* sensors observing a specific phenomenon (e.g., temperature, water level, pollution) and *(ii)* an intelligent mechanism that responds to the identification of events (e.g., fire, flood). However, only a few MSs exploit the extracted knowledge of sensors as a *whole*.

We propose a mechanism for an MS, which combines *data fusion*, *consensus methods*, *time series prediction*, and *Fuzzy Logic-based inference* to derive the identification of hazardous events. Our mechanism builds on top of streaming contextual data captured by a fixed number of sensors and provides immediate responses to any identified abnormality. The proposed mechanism *derives knowledge from a team of sensors* and does not rely on single sensor observations. A decision based on single sensor measurements could be affected by many reasons (e.g., location where the sensor is placed, network connection, battery level) and the incoming reports could be invalid. Our mechanism aggregates the received measurements and reasons over the opinion of the team of sensors about the identification of an event. Through this collaborative approach, our mechanism introduces an intelligent scheme on top of the *compactness* of the team concerning their *current* and *recent past* opinion about the phenomenon. With the term *compactness*, we characterize the unanimity of all (or a specified subset) sensors about the phenomenon. Our aim is to efficiently identify the event by minimizing false alarms that could affect the reasoning and decision making tasks.

B. Contribution & Organization

The contribution of our paper is as follows: (1) We provide a mechanism for multiple contextual data streams monitoring and phenomena identification by minimizing the rate of false alerts; (2) Our mechanism treats the sensors as a team and does not rely on single sensors observations; (3) Our mechanism combines the following techniques: *(i) Data fusion* for aggregating the current opinion of a number of experts (sensors) excluding outliers; *(ii) A consensus operator* for identifying the unanimity of experts (sensors) on a specific phenomenon; *(iii) Time series forecasting* for estimating the future behavior of the team based on the recent opinions; *(iv) Fuzzy-Logic based inference* for handling the uncertainty related to the decision of when an event takes place; (4) A comprehensive sensitivity analysis of the proposed mechanism with the basic model parameters and a comparative assessment with baseline solutions: the *Single Sensor Alerting* (SSA) mechanism, the *Average Measurements Alerting* (AMA) mechanism, the simple *fusion model* (FM), the *Moving Average Model* (MAM), a *Type-1 fuzzy inference system*, and the *Simple Prediction Mechanism* (SPM).

The paper is organized as follows. Section II reports on the related work while Section III presents our rationale. Section IV introduces the proposed mechanism. In Section V we present our experimental evaluation of the proposed mechanism and provide a comprehensive comparison assessment. Section VI concludes our paper by giving future research directions.

II. Related Work

A. Sensor monitoring systems

Contextual data streams monitoring mechanisms are adopted to support the creation of intelligent applications on top of the observed data. Recent developments create new opportunities to incorporate intelligence into the existing or new systems while they facilitate the participation of citizens [4]. Sensors are adopted to monitor a specific (sub-)area and deliver the collected data to an MS. For instance, a network of sensors has the capability to provide temporal and spatial data regarding the properties of the environment [5]. The MS processes the incoming contextual information and, thus, derives decisions over the state of the environment or producing alerts. The authors in [6] present the architecture of an MS. The system involves a tree construction algorithm that enables the energy efficient calculations and a set of experiments for reporting on its advantages. In [7], a sensor network is proposed for air quality monitoring in indoor environments. A base station collects data of a set of autonomous nodes equipped with sensors measuring temperature, humidity, light, and air quality while specific software is proposed for the supervision of the retrieved measurements. Many researchers have shown significant attention on the identification of hazardous events. In [8], the authors present a framework for early detection of fire. The framework in [8] involves a set of vision-enabled WSN for reliable, early on-site detection of forest fires. A vision algorithm for smoke detection is proposed and the implementation of a power-efficient smart imager is analyzed. The interested readers can also refer to [1] and the references therein for an extensive survey on tools dealing with the automatic detection of fire. Home or critical areas monitoring is another significant field of interest. The authors in [1] present a technique for home monitoring based on the sensors' RSSI signal. The proposed system in [1] builds a smart home that supports scalable services and context aware applications. In [2], a WSN for the surveillance of critical areas is proposed. The system in [2] focuses on ensuring integrity and authenticity of the generated alerts. In [10], a WSN is adopted to build an effective framework for the identification of hazardous underground materials. The framework in [10] is responsible for surveillance and security monitoring of underground mines and confined areas. The focus of [11] is on large scale WSNs oriented to provide monitoring functionalities. The mechanism aims to localized decisions that minimize the propagation of false alerts. A WSN for structural health monitoring is also presented in [12]. A set of nodes are responsible of observing ambient vibrations.

B. Environmental monitoring systems

Environmental monitoring has also attracted significant interest due to the consequences that negative effects in the environment have on humans. In [4], the authors present a framework for combining the concepts of traditional environmental monitoring networks with the ideals of the open source movement. Sensors consist of the fixed infrastructure, citizens play the role of the mobile monitoring network and a back-end system defines the appropriate decisions for environmental protection. In [5], a WSN is deployed to monitor temperature, humidity and soil moisture. An algorithm for scheduling the routes of data is proposed. For monitoring various substances like odour, the authors in [3] propose two approaches: *(i)* working under fixed experimental conditions or *(ii)* measuring the external parameters to numerically compensate for their change. The challenge when dealing with chemical substances is to identify the correlation between the multivariate data and a number of pre-defined indicators (they depict the quality of measurements concerning the chemical substances). The aforementioned efforts present mechanisms where the rate of the incoming information is stable and, thus, bottlenecks could be present in large scale data. For alleviating this problem, a dynamic model adapted into the data is proposed in [13].

C. Fuzzy logic-based monitoring systems

In monitoring activities, FL could be proved a useful technique for delivering high quality systems [14], [15], [16]. The proposed FL systems handle the uncertainty related to the distribution of sensors reports and help in the decision making, i.e., simple alerts or more complex reasoning tasks. In [17], the authors report on a model that tries to predict the peak particle velocity of ground vibration levels. The model adopts an FL scheme and utilizes the parameters of distance from blast face to vibration monitoring point. Another prediction model adopting the FL is discussed in [18] and tries to estimate the Gamma radiation levels in the air. In [19], the authors discuss a FL data fusion technique for fault detection. The FL is adopted to reduce uncertainty and false-positives within the process. In [20], a Type-2 FL system is proposed for ambient intelligence environments. The system learns the user's behavior and, thus, is capable of being adapted to user's profile (i.e., it learns rules and membership functions).

D. Data fusion and outliers detection

A number of techniques can be used for performing fusion, prediction, consensus and decision making. Outliers detection has been widely studied, especially, in the domain of WSNs. An extensive survey is presented in [21]. The outlier detection techniques could be categorized to: *(i)* statistical-based (parametric or non-parametric approaches), *(ii)* nearest neighbor-based, *(iii)* clustering-based, *(iv)* classification-based (Bayesian network-based and support vector machine-based approaches), and *(v)* spectral decomposition-based approaches. Statistical approaches, usually, require the a-priori knowledge of the sensor data distribution. However, in real scenarios, this could not be the case making the parametric approaches useless [21]. Nearest neighbor approaches are not based on assumptions on the streams distribution but are, mainly, efficient only for univariate. In multivariate datasets nearest neighbor approaches are computationally expensive. Moreover, multivariate data require the appropriate choice of input parameters and impose an increased computational cost making the neighbor based approaches to lack scalability [21]. Clustering techniques exhibit similar drawbacks as the neighbor based approaches while the classification methods are computationally expensive [21]. We adopt the CumSum algorithm [22] that is computationally efficient and does not depend on the a-priori knowledge on the streaming data distribution. The selected algorithm is appropriate to manage the concept drift (e.g., changes in streams over time in unforeseen ways) aspect of the examined scenario. In addition, the time complexity of the algorithm is linear to the number of nodes and requires only the knowledge on certain thresholds estimated experimentally. We require a fast outliers detection technique which exploits the temporal statistical characteristics of the data stream, which, in the case of CumSum, is the mean value within a fixed window. Hence, the proposed mechanism can easily support (near) real time decisions over the identification of events 'hidden' in the sensors measurements. This helps in minimizing the time required to produce alerts and initiate actions towards the management of the identified events.

E. Information aggregation under uncertainty

Data fusion over multiple data sources is affected by the uncertainty about the status and the accuracy of the reporting nodes. A review of multisensor data fusion techniques is presented in [23]. A candidate solution to handle uncertainty is probability theory. Probability theory could provide a well understood way for representing uncertainty and may, thus, be used to build a mechanism for storing uncertain information [24]. The authors in [24] consider the problem of information aggregation using an imprecise probability data model that allows the representation of uncertainty using probability distributions. Any decision making on top of uncertain data has to be accomplished over values that may drastically change over time [25]. For managing dynamic changes, fuzzy information and aggregation operators could be adopted. An aggregation operator that combines a time sequence of hesitant fuzzy information is discussed in [25]. The proposed operator is applied to the service selection problem with the best combination of features based on historical assessments. In [26], a framework for aggregating the linguistic opinions of experts in human decision making is proposed. Type-2 fuzzy sets are adopted to represent the knowledge of experts and an aggregation scheme of intervals for Type-2 fuzzy sets is proposed. In [27], the authors discuss another model for the aggregation of fuzzy information. New operations are proposed like the Einstein sum, the Einstein product, and the Einstein scalar multiplication. Another effort that develops aggregation operators for fuzzy information can be found in [28]. In addition, important research can be found in adopting FL in aggregation mechanisms. In [29], a fuzzy-based data fusion approach for WSNs is proposed. The model aims to increase the QoS while reducing the energy consumption of nodes. In [30], another FL-based data fusion method is combined with Dempster-Shafer theory to derive the aggregated values. The belief function of each node is obtained from extracted eigenvalues by using FL. The FL is also combined with an adaptive Kalman filter in [31]. The proposed mechanism is used to build adaptive federated Kalman filters for adaptive multisensor data fusion. Finally, in [32], multiple image partitions are fused through a FL model. The proposed model integrates the outcomes of multiple image segmentations and provides time consistent spatiotemporal partitions for moving objects.

F. Prediction techniques

Time-series forecasting is widely adopted in many application domains to provide an insight on future values of a specific random variable. Linear [33] or polynomial predictors provide methods that try to estimate future values of a random variable. In both cases (i.e., linear and polynomial methods), a set of coefficients should be determined. The Stoer-Bulirsch methodology adopts the Richardson extrapolation [34], the rational function extrapolation in Richardson applications, and the modified mid-point method [35]. Neural networks can also be adopted for prediction. Neural networks are information processing systems that have certain characteristics in common with biological neural networks [36]. We require a fast prediction mechanism with linear complexity. We adopt the linear predictor defined in [33], [37] as the proposed MS should derive immediate decisions. We choose the discussed technique due to its simplicity and the speed of delivering the final result. The linear predictor is applied over a fixed window of measurements when there is a need for estimating the future aggregated value of the team of sensors. The predictor is executed every time the prediction process is initiated and the final predicted measurement is derived.

G. Consensus methods and metrics

In [38] and [39], the authors provide a comparison for a set of consensus metrics. They compare a consensus technique based on entropy [40], a distance based pairwise consensus [41], a preference similarity-based consensus [42] and a technique that derives the consensus through the standard deviation. In our mechanism, we adopt the metric defined in [41]. It is based on pairwise comparisons of the opinions of members in a group. It determines the overall evaluation of the consensus by considering each pair in contrast to other techniques that search the dissimilarity between each argument and the mean.

H. *Decision making methods under uncertainty*

In an MS, we cannot be sure about when to decide the initiation of an alert. To this end, two approaches can be adopted to support an efficient decision making mechanism: *probabilistic* or *FL reasoning*. A probabilistic approach aims to handle the occurrence or not of an event. FL is a precise system of reasoning, deduction and computation in which the objects of discourse are associated with information which is, or is allowed to be, imprecise, uncertain, incomplete, unreliable, partially true or partially possible [43]. The key difference between the two theories is the meaning (that makes us to choose the FL for the reasoning mechanism), making them to be applied in different application domains. In general, probability theory *does not* [44]: *(i)* support the concept of a fuzzy event; *(ii)* offer techniques to deal with fuzzy quantifiers (e.g., low, medium, high); *(iii)* offer methods to calculate fuzzy probabilities; *(iv)* offer methods to estimate fuzzy probabilities; *(v)* sufficiently express meanings; *(vi)* analyze problems in which data are described in fuzzy terms.

In our case, the MS needs to manage fuzzy concepts (e.g., which values are considered as high to drive an alert), thus, we adopt the FL. Our mechanism focuses on a combination of multiple components. These components aim to provide fusion, forecasting, consensus and decision making towards the identification of events. The difference of our work with other efforts is that we do not focus on a single technology and try to combine a number of technologies with the help of the FL. Various efforts focus on the decision fusion and not in data fusion. For instance, in [45], the authors focus on the fusion of decisions made on top of sensors. The proposed system acquires data from sensors that perform value fusion locally and from sensors without the value fusion capability. In contrast, our mechanism performs value fusion over the sensors measurements assuming that sensors do not perform any fusion process. Our mechanism is fault tolerant when problems in the sensors reporting arise (e.g., hardware problems). As reported in [45], in some cases, decision fusion performs well [46], [47], [48], however, the generic view is that data fusion exhibits better performance.

III. Rationale

A. *Scenario description*

Consider a finite set of n sensors $S = \{s_1, s_2, \ldots, s_n\}$ that monitor a specific area for the same phenomenon and report their real-valued measurements $X = \{x_1, x_2, \ldots, x_n\}$ to an MS. When the MS receives the incoming measurements, it infers whether an event takes place and derives alerts to end users / applications (registered for that event). For instance, if sensors measure ambient temperature or locally identify an increase in the probability of fire, the MS should infer and report a *fire event*. The MS handles the sensors as a team and, through the team reports, attempts to minimize the false alerts while being certain enough on the event identification. The MS relies on the opinion of the majority before deciding on an alert. Further, the MS deals with the fact that sensors are subject to report false values (e.g., missing values, outliers) due to various reasons.

In this context, *uncertainty* is mainly related with sensors measurements. The following question should be answered: *Should the mechanism be confident on the contextual information reported by each single sensor*? Sensors are affected by many issues and, thus, they report faulty values. Reasons that affect sensors measurements are: *(i)* different views on the observed phenomenon and, *(ii)* the existence of noise in measurements. We provide an inference mechanism responsible to derive an alert when an event is identified. We consider that the *fused measurement* of the team combined with the *predicted measurement* and the *consensus* among the team members are the basis of the proposed mechanism. Hence, we pay attention on the current opinion of the team as well as its future estimation. The inference process checks the current state, the future state and whether the team agrees on the presence of the event. Uncertainty makes the decision of deriving an alert difficult. This is because the MS should minimize false alerts as they lead to unnecessary reactions while spending resources for handling responses.

Two types of uncertainty can be identified [49]: *random* and *linguistic*. Random uncertainty is originated in statistical errors in sensors measurements while the linguistic uncertainty deals with the view of each expert on the characterization of the phenomenon. To cover all the aspects of uncertainty both, random and linguistic, we adopt FL and, more specifically, Type-2 Fuzzy Sets. We need an uncertainty handling mechanism on top of the fusion process as crisp thresholds for deriving alerts can easily lead to increased false alarms. In our mechanism, FL manages issues like: *which measurement or aggregated value (i.e., the result of the fusion process) can be considered as faulty (i.e., random uncertainty), high or low (i.e., linguistic uncertainty)*? Such issues cannot be managed by a decision mechanism based on crisp thresholds. The second reason for adopting Type-2 Fuzzy Sets is that in a *Type-2 FL System* (FLS), apart from the management of uncertainty related to sensors measurements, the uncertainty present in membership functions is included in calculations for deriving the final output. Experts defining membership functions cannot be aware, in advance, of the entire picture of a very dynamic environment like multisensory settings.

B. *Type-2 Fuzzy Sets*

We propose a Type-2 FLS to deliver the MS's reaction to the incoming measurements. We do not adopt a Type-1 FLS as it has specific drawbacks when applied in dynamic environments. Type-2 FLSs are better to handle uncertainties than Type-1 FLSs. Two main differences between Type-2 and Type-1 FLSs are [50]: *(i) adaptiveness*, the Type-1 fuzzy sets change as input changes; and *(ii) novelty*, the upper and lower membership functions of the same Type-2 fuzzy set may be used simultaneously in computing each bound of the type-reduced interval. Research efforts have shown the limitations in typical Type-1 FLSs [51], [52]. Type-1 FLSs involve memberships that are crisp values and not intervals. Hence, in Type-1 FLSs, experts should be capable of defining exactly the membership degree of fuzzy variables. However, in some domains, experts cannot be certain about the membership grade. In such cases, uncertainty is observed not only on the environment (e.g., when a measurement is considered as high or low) but also on the membership grades for each variable.

The output of a Type-1 FLS is analogous to the mean of a pdf, in a probabilistic model, while Type-2 FLSs are analogous of the variance providing a measure of dispersion for the mean [52]. Based on these differences, it is clear that Type-2 FLSs can handle large amounts of uncertainty helping developers in the definition of membership functions. Such an approach seems to be appropriate in our scenario. For instance, when a measurement is received, the MS cannot be sure if measurements are correct and if yes, the MS cannot be aware if the specific measurement(s) could be the basis for triggering an alert.

C. *The uncertainty-driven mechanism*

The MS orchestrates a number of processes to handle the incoming contextual data and derive the appropriate decision at each reporting time interval. Fig. 1 presents the envisioned architecture. The proposed mechanism involves a set of processes as follows: **(1)** the **Fusion Process**: It undertakes the responsibility of identifying and eliminating the outlier data and provides an aggregated measurement for the team of sensors; **(2)** the **Prediction Process**: It exploits the trend of historical aggregated measurements of the team of sensors and forecasts short-term aggregated measurements; **(3)** the **Consensus Process**: It produces a *Degree of Consensus* (DoC), which denotes the current unanimity in the opinion of the sensors (experts). The DoC is a real value in [0, 1], where DoC $\rightarrow$ 1 indicates the sensors agreement on their inference about the phenomenon; **(4)** the **Type-2 Fuzzy Inference Process**: This process is realized by a *Type-2 FLS* and combines the fusion, prediction and consensus processes while incorporating Type-2 Fuzzy Sets to handle the uncertainty in the membership functions definition. The fuzzy inference process derives the *Degree of Danger* (DoD) that provides a view on the *existing danger* on top of the aforementioned processes. When the DoD is over a pre-defined threshold, the MS ensures the occurrence of an event and initiates an alert.

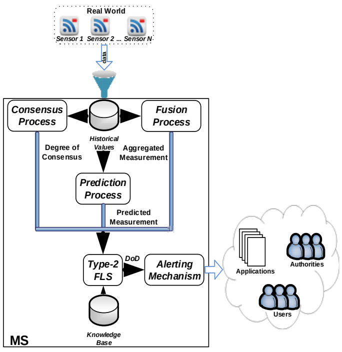


Fig. 1. The proposed MS architecture: results of the fusion, prediction and consensus processes are fed to a Type-2 FLS that derives alerts over sensors measurements.

Finally, in Table 1, we provide the notations adopted throughout the paper.

TABLE 1. The notations adopted throughout the paper.

Notation	Description
S	The set of sensors
n	The number of sensors
s_i	The i^{th} sensor
x_i	The i^{th} sensor measurement (real number)
k^+, k^-, h^+, h^-	The above / below tolerance and thresholds of the CumSum algorithm
g^+, g^-	The above / below detection signal of the CumSum algorithm
w_i	The weight associated with a sensor measurement ($w_i \in \mathbb{R}^+$)
c_i	The i^{th} sensor's confidence value (real number)
W	The examined window (integer number)
m_i	The number of outlier occurences
γ, δ	Parameters for eliminating the confidence of a sensor
y	The aggregated measurement in the Fusion Process
y^*	The predicted measurement in the Prediction Process
M	The window of historical values adopted in the Prediction Process

IV. DATA FUSION & FUZZY INFERENCE MECHANISM

A. *Multi-sensor data fusion*

Multi-sensor data fusion combines contextual data from n sensors to produce reliable fused measurements. The aim is to produce a measurement through the entire set of the incoming observations that will depict the view of the team on the phenomenon. The fused measurement is derived by eliminating sensors that deviate from the rest (defined by the mean of the distribution of measurements). We provide a 'compact' measurement based on these sensors. We eliminate outliers corresponding to measurements that do not 'agree' with the team. Then, we produce the final measurement based on the remaining sensors. We *(i)* adopt the *Cumulative Sum* (CumSum) concept drift algorithm [22] for outliers detection and *(ii)* the linear opinion pool algorithm [53] for deriving the final aggregated measurement.

Outlier detection. The CumSum algorithm attempts to detect if there is any change in the distribution of a contextual time series $x_i[t]$ corresponding to sensor s_i, i = 1, …, n and discrete reporting time instance t = 1, 2, … The algorithm is a change-point detection technique, which is based on the cumulative sum of the differences between the current value at t and the overall average up to t. The overall average is depicted by the mean of the sensor measurements distribution. Slopes depict jumps in the series, thus, corresponding to possible outliers. We adopt a two-side detection scheme where $x_i[t]$ is considered as an outlier when it deviates above a threshold h^+ or below a threshold h^-. Parameters of the algorithm are: (i) the target value (i.e., the mean value-measurement), (ii) the tolerance for the above and below thresholds k^+, k^- and, (iii) the above and below thresholds h^+, h^-.

One crucial decision is the definition of h (h^+ and h^- are symmetrically defined according h). It mainly depends on the signal characteristics and is generally adjusted by experience or by using a training dataset [54]. A number of approaches have been proposed for the definition of h [22], [54], [55], [56], [57], [58], [59], [60], [61]. These efforts handle the research question of automatically defining h through two aspects: *(i)* theoretical and *(ii)* application depended. A few methodologies choose h according to the probability of false alerts and the mean time in the inter-arrivals [55], [56]. The provided formulations are asymptotic and it is difficult to be applied in real scenarios. In [54], the authors propose the adoption of the Kullback–Leibler[1] distance between two probability densities of the examined random variable (i.e., the variable that produces the signal) as h. Other approaches involve the use of the average run length. An approach is to select the mean time between false alerts or the mean detection delay. In any case, the appropriate selection of h plays an important role as it affects the detection of the outliers. A low h will lead to many outliers while a high h will lead to only a few outliers. The best practice is to adopt training data and, accordingly, to set h. In our scheme, we adopt training data for determining h. We choose this approach as it is computationally efficient compared to other approaches (it could be difficult to have an analytical model).

Two signals are the outputs of the CumSum algorithm. The first is related to the above detection signal $g^+ \in \{0,1\}$ i.e., $g^+ = 1$ when $x_i[t] > h^+ + k^+$; otherwise, $g^+ = 0$, while the second corresponds to the below detection signal $g^- \in \{0,1\}$, i.e., $g^- = 1$ when $x_i[t] < h^- - k^-$; otherwise, $g^- = 0$. The detected outliers (i.e., when g^+ and g^- set to 1) are eliminated and, thus, the mechanism is based on the opinion (measurements) of non-outlier sensors. Let I_i be the outlier indicator of s_i, i.e., $I_i = 0$ if x_i of s_i is outlier; otherwise $I_i = 1$.

Data aggregation. Our mechanism, through data fusion, reaches to a consensus based on the linear opinion pool method over the non-outlier measurements. The linear opinion pool is a standard approach adapted to combine experts' opinion (i.e., sensors) through a weighted linear average of measurements. Our aim is to combine single experts' opinions in order to produce the opinion of the team of sensors. We define specific weights for each sensor to 'pay more attention' on its measurement and, thus, to affect more the final aggregated result. The aggregated result corresponds to the measurement that the team reports to the MS. Formally, $F(x_1, \ldots, x_n)$ is the aggregation opinion operator (i.e., the weighted linear average of non-outlier measurements), i.e.,

$$y = F(x_1, \ldots, x_\varepsilon) = w_1 x_1 + \ldots + w_\varepsilon x_\varepsilon \tag{1}$$

where w_i is the weight associated with the measurement of s_i such that $w_i \in [0,1]$ and $w_1 + \ldots + w_\varepsilon = 1$ (ε is the number of the measurements derived by the CumSum algorithm). Weights w_i are calculated based on specific characteristics that affect the confidence on each sensor. Let c_i be the *confidence* that the MS has on s_i. c_i depicts the opinion of the MS that the s_i is 'successfully fulfilling the assigned task'. We set w_i as the normalized confidence c_i, i.e.,

$$w_i = \frac{c_i}{\sum_{k=1}^{\varepsilon} c_k} \tag{2}$$

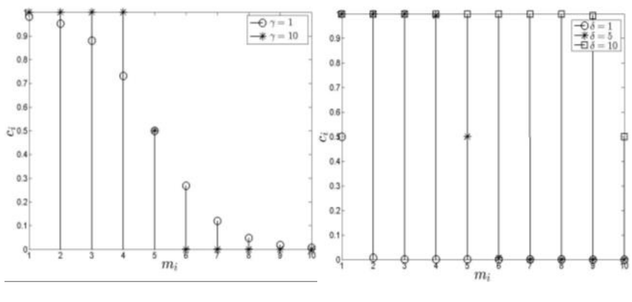
The mechanism assigns high weight on the measurement that corresponds to a sensor having a high c_i. c_i could be determined by the sensor's resources state (e.g., battery level) or historical data of each sensor stored in the back-end system. Historical data consist of the basis for deriving specific intervals that a sensor's measurements are considered as invalid. Such intervals depict the 'trajectory' of a sensor behaving as outlier i.e., series of the outlier indications.

Sensor confidence. We evaluate c_i by adopting a heuristic technique; we require a mechanism that in (near) real time responds to events instead of using resource demanding and time consuming techniques. Our mechanism observes for each sensor the rate of outlier occurrences adopting a periodic window setting. Let W be the window size. The mechanism keeps a record of the behavior of all sensors for the finite horizon W. The MS pays little attention on outlier sensors concluded by a low c_i that, finally, affects a measurement being aggregated with the remaining observations. c_i for a sensor s_i is determined at t involving all values from t-W to t. For each W, our mechanism counts the number of outlier occurrences for each sensor. After the expiration of W, historical values collected in the era $[t_1, t_W]$ are eliminated (e.g., outlier counters are set to zero) and a new era starts off depicted by the interval $[t_{W+1}, t_{2W}]$. Hence, the mechanism eliminates past behavior and does not punish sensors if, in the past (i.e., in the previous window), they exhibit an 'abnormal' behavior in terms of outliers occurrence rate. W should not be low; with a low W, the mechanism does not actually involve many historical values. On the other hand, W should not be very high as the mechanism pays more attention on the past behavior of each sensor eliminating the current behavior. The confidence value c_i, at time instance t, is a function on the number of outlier occurrences $m_i[t] = (1-I_i[t-1]) + (1-I_i[t-2]) + \ldots (1-I_i[t-W])$ for sensor s_i reported within W. As the MS cannot be certain on the real state of sensors, we disregard sensors that for successive measurements report outliers. We adopt a reverse sigmoid function for evaluating c_i to eliminate the confidence of a sensor when $m_i[t]$ exceeds specific pre-defined thresholds. The following equation holds true:

$$c_i[t] = \frac{1}{1+e^{\gamma(\sum_{\tau=1}^{W}(1-I_i[t-\tau]))-\delta)}} \tag{3}$$

where the γ and δ real-valued parameters that define the shape and the threshold for eliminating c_i, respectively. It should be noted that the determination of γ and δ is application specific. Through the adopted sigmoid function, we can easily eliminate the confidence of a sensor and, thus, the sensor measurement is not taken into consideration in the final fused measurement. Indicatively, if we desire a 'strict' mechanism that immediately eliminates c_i (e.g., which results at least one outlier), we should set $\gamma \to 0$ and $\delta \to 0$. In Fig. 2, we depict the c_i value against the shape γ and threshold δ.

Remark: *It should be noted that the mechanism calculates the $m_i[t]$ value at every time instance t where a fusion process takes place.*



[1] Kullback, S., Leibler, R.A., 'On information and sufficiency', Annals of Mathematical Statistics, vol. 22(1), 1951, pp. 79–86.

(a) $\gamma \in \{1, 10\}$ *(b) $\delta \in \{1, 5, 10\}$*

Fig. 2. Plot of c_i for different γ and δ values. c_i over δ is very low. If a sensor is considered as outlier for $m_i > \delta$ times, the MS decreases the confidence level of that sensor.

B. Consensus process

The consensus process evaluates the *Degree of Consensus* (DoC) taking values in [0,1]. The DoC represents the unanimity on the opinion of sensors (i.e., measurements) about the observed phenomenon. When DoC $\rightarrow$ 1, the team of sensors unanimously agree on a specific opinion i.e., occurrence or non-occurrence of an event. A DoC value close to zero denotes a team of sensors that cannot conclude on the occurrence or the non-occurrence of an event. Our mechanism identifies if all sensors agree on their measurements as a team (i.e., have the same opinion – measurement for the observed phenomenon). The evaluation of the DoC is based on the technique discussed in [41], which compares the opinion of each sensor with the remaining sensors of the team. Specifically, the DoC, at the reporting time instance t, is defined as follows:

$$DoC = 1 - \frac{2}{n^2} \sum_{\forall i, j, i \neq j} \left(x_i - x_j\right)^2 \quad (4)$$

where x_i and x_j are contextual data from sensors s_i and s_j, respectively. The specific technique does not require any complex calculations and relies on pairwise comparisons. Pairwise comparisons between sensor measurements aim to identify the difference in the opinion of the team members. The smaller the difference is, the higher the unanimity in the sensors opinions becomes. Hence, the DoC indicates whether the sensors have the same opinion on the observed phenomenon as realized by their measurements or divergent opinions. The DoC is calculated over the sensor measurements that are not characterized as outliers.

C. The prediction process

Every reporting time, the mechanism triggers the fusion process and obtains the current aggregated measurement as depicted in Fig. 1. In the sequel, the mechanism updates the historical aggregated measurements given a window $M > 0$ corresponding to all sensors. Historical aggregated measurements consist of the basis for predicting the future aggregated measurement. The idea is to see how the fused measurement 'evolves' over time. To achieve these goals, the mechanism engages a time series prediction algorithm. The future behaviour, as realized by the prediction model, could be adopted to 'calibrate' the current behaviour and, thus, to lead to the most appropriate decisions according to the identification of events. Consider two scenarios: ***Case A***: the current behaviour depicts that no event is triggered while the prediction scheme indicates that the event is possible; ***Case B***: the current behaviour depicts that the event is possible while the prediction scheme indicates that the event will not be present. In these scenarios, both models deal with current and future behaviour and the MS can 'calibrate' each other when inserted into the FLS. The FLS will be responsible to regulate possible contradictions that could be present, as the aforementioned scenarios indicate, while handling the uncertainty related to the event identification. Since the mechanism should derive the predicted aggregated measurement in the minimum time (i.e., the provision of alerts should be realized in (near) real time), we adopt a linear predictor over the available historical values. Specifically, for a history of the latest M aggregated measurements $y[t\text{-}1]$, $y[t\text{-}2]$, …, $y[t\text{-}M]$, with $y[t\text{-}k] = F(x_1[t\text{-}k], \ldots, x_n[t\text{-}k])$ be the aggregated measurement at $t\text{-}k$, $k = 1, \ldots, M$. We predict the aggregated measurement $y^*[t]$ through a linear combination of the $y[t\text{-}k]$ historical aggregated measurements with real-valued a_k coefficients. The set of coefficients $\{a_k\}$, $k = 1, \ldots, M$ are estimated to minimize the error between the predicted $y^*[t]$ and the actual aggregated measurement $y[t] = F(x_1[t], \ldots, x_n[t])$ at the reporting time t.

$$y^*[t] = \sum_{k=1}^{M} a_k y[t-k] \quad (5)$$

A number of algorithms have been proposed for the calculation of a_k coefficients, with most known being the minimum mean square estimate of $E[(y-y^*)^2]$ getting the Yule-Walker equations. We adopt the Levinson-Durbin algorithm [33], [37].

D. Type-2 fuzzy inference process

The *Type-2 FLS* (T2FLS) is a non-linear mapping between l inputs $u_i \in U_i$, $i = 1, \ldots, l$ (e.g., the fused – predicted value of the team or the DoC) and z outputs $v_i \in V_i$, $i = 1, \ldots, z$ (e.g., the DoD). The proposed T2FLS is based on a set of rules (i.e., the FL rule base) that combine the available inputs and derive if the event is present. FL rules have the following structure:

R_j: **If** u_{1j} is A_{1j} and/or u_{2j} is A_{2j} and/or ….and/or u_{lj} *is* A_{lj} **Then** v_{1j} *is* B_{1j} and…and v_{zj} *is* B_{zj},

where R_j is the j^{th} fuzzy rule, u_{ij} are the inputs of the j^{th} rule, v_{kj} are the outputs of the j^{th} rule and A_{ij} and B_{kj} are membership functions. Membership functions are adopted to represent the fuzzy sets for each input and output variables. The structure of the FL rules is the same as in Type-1 FLSs, however, a Type-2 FL rule base involves Type-2 fuzzy sets in the antecedents and consequents (i.e., the combination of inputs and outputs). Membership functions in Type-2 FLSs are intervals defining the upper and the lower bounds for each fuzzy set [62]. The area between the two bounds is referred to as *Footprint of Uncertainty* (FoU). A_{ij} are interval Type-2 fuzzy sets and B_{ij} are the centroids of a consequent Type-2 fuzzy set. In general, the following steps are adopted when activating a Type-2 FLS: *(i)* Calculate the membership functions for each set based on inputs crisp values; *(ii)* Calculate the firing interval of each FL rule; *(iii)* Perform type reduction to combine the firing intervals; *(iv)* Produce the interval of the consequent. The defuzzification phase defines the final output. The most common method for type reduction is the *center of sets type reducer* [52].

The T2FLS has three inputs: *(i) The current aggregated measurement y*; *(ii) The current degree of consensus (DoC) given fixed γ and δ*; *(iii) The predicted aggregated measurement y^**. The DoD is generated by the T2FLS on top of the three inputs. Sensor measurements are fed to the fusion process where outliers are eliminated and the linear opinion pool derives the final fused measurement. Sensor measurements are also fed to the consensus process to derive the DoC. The predicted value is calculated over the historical values of the fused measurements. The scalability of the system is affected only by the scalability of the fusion and the consensus processes. We assume that inputs are normalized in [0,1] based on the minimum and maximum values as depicted by the application domain. For instance, if the MS monitors the existence of fire, we can set a maximum temperature that sensors could measure. We also define that DoD $\in [0,1]$. A DoD close to unity denotes the danger at high levels, i.e., there is a high belief that a hazardous phenomenon actually occurs. The opposite stands when DoD $\rightarrow 0$.

For inputs and the output, we consider three linguistic values: *Low*, *Medium*, and *High*. *Low* represents that the fuzzy variable takes values close to the lowest limit while *High* depicts that the variable takes values close to the upper level. *Medium* depicts that the fuzzy variable takes values close to the average (i.e., around 0.5). For instance, a *Low* value for y shows that the fused measurement is close to the lower limit of the sensors measurements (e.g., for a temperature sensor could be close to zero). A similar rationale stands for the remaining linguistic values. For each fuzzy set, human experts define the upper and the lower limits for the Type-2 fuzzy sets. For simplicity, we consider *triangular* membership functions as they are widely adopted in the literature. In Table 2, we present our FL rule base. Finally, when the DoD is over a pre-defined threshold $\theta \in [0,1]$, the MS triggers an alert, otherwise, it continues with the upcoming sensors reports.

TABLE 2. The FL rule base. The rules are designed for scenarios where sensors data reaching the upper limit exhibit a 'danger' case.

	Inputs			Output
Rule	***y***	***y******	**DoC**	**DoD**
1	Low	Low	*Any*[2]	*Low*
2	Low	Medium	Low or Medium	*Low*
3	Low	Medium	High	*Medium*
4	Low	High	Low	*Low*
5	Low	High	Medium or High	*Medium*
6	Medium	Low	Low or Medium	*Low*
7	Medium	Low	High	*Medium*
8	Medium	Medium	*Any*	*Medium*
9	Medium	High	Low or Medium	*Medium*
10	Medium	High	High	*High*
11	High	Low	Low	*Low*
12	High	Low	Medium or High	*Medium*
13	High	Medium	Low or Medium	*Medium*
14	High	Medium	High	*High*
15	High	High	*Any*	*High*

V. PERFORMANCE EVALUATION

We elaborate on the performance of the proposed mechanism to evaluate if it minimizes false alerts while accurately and early identifies any event. We experiment with datasets collected by real sensors. Our experimental scenarios involve the environmental parameters: temperature and water levels.

[2] With the term *Any*, we depict the entire set of the available Type-2 Fuzzy Sets.

A. *Perfrormance metrics & simulation setup*

We define the following performance metrics:

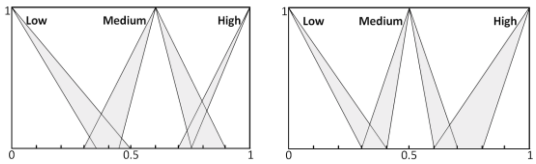
- *Rate of false alerts* (RoFA). RoFA $\in [0,1]$ represents the rate of false alerts derived by the mechanism. As RoFA $\rightarrow 1$, the MS results a lot of false alerts and no efficient conclusion could be drawn about the true state of the phenomenon. As RoFA $\rightarrow 0$, the MS decreases the rate of false alerts and efficient conclusions could be drawn concerning the observed event. RoFA is the ratio of false alerts out of a total number of measurements.
- *Index of Alert* (IoA). The IoA refers to the (natural number) index of the measurement that triggers an alert (i.e., IoA $\in \{1, 2, 3, \ldots\}$). Through the IoA, we can see how close to the real case an event is triggered (not at early stages to avoid false alerts and not many stages after the real event). In that sense, we want to have an IoA close to the real event as depicted by sensors measurements.

In Table 3, we present our system's parameters adopted in our experiments. We experiment with two real datasets. The dataset of the Intel Berkeley Research Lab[3] that contains millions of measurements retrieved by $n = 54$ sensors deployed in a lab. From this dataset, we get 15,000 measurements i.e., 15 sensors produced 1,000 temperature measurements. As no hazardous event is identified by these measurements (i.e., the probability of a true event is zero), we consider the injection of faulty values to see whether the proposed mechanism produces false alerts. We assume that a high temperature (e.g., around 600^o Celsius) defines the case of a fire incident and inject faulty measurements as indicated in [63] i.e., every actual measurement x_i is replaced as follows: $x_i \leftarrow (1 + a)\, x_i$, $a \in \{2, 3, 5\}$. We provide experiments with scenarios where a portion of measurements $p \in \{1\%, 5\%, 10\%, 20\%, 40\%, 60\%, 80\%\}$ are considered as faulty. p represents how many faulty measurements are included into our dataset. When $p = 1\%$, we inject 150 faulty measurements, when $p = 5\%$, we inject 750, and so on. Through faulty measurements, we simulate a setting where a number of sensors deliver faults. Our mechanism aims to minimize the false alerts to face faulty values and to minimize the need for addressing each issue producing faulty values. Finally, we experiment with the adoption of missing values. We define ω which depicts the probability of a missing measurement; when $\omega \rightarrow 0$, missing values are limited while a value of ω close to 1 means that only a few measurements are present.

We also adopt data retrieved by a real flood event as reported by a number of water level sensors[4]. Sensors were located in the shore of a river. We get the measurements of $n = 10$ sensors (we adopt the entire dataset of the available sensors). In this dataset, the probability of the alert is equal to unity as the flood event was realized in the past. We adopt 265 measurements just before and during the flood event and normalize them in [0,1]. Based on these data, we try to identify if the MS is capable of identifying the event at the right time (not too early to avoid false alerts and not after the real event in order to avoid a disaster).

[3] Intel Lab Data, http://db.csail.mit.edu/labdata/labdata.html
[4] The data were retrieved in the past by http:// www.pegelonline.wsv.de

Our T2FLS utilizes triangular membership functions for each fuzzy set. In Fig. 3, we plot membership functions for *y* and DoD. Membership functions for the remaining fuzzy variables are similarly defined. We compare the proposed T2FLS with the following monitoring mechanisms: (a) the *Single Sensor Alerting* (SSA) mechanism and (b) the *Average Measurements Alerting* (AMA) mechanism. SSA delivers an alert when at least one sensor reports a measurement over a pre-defined threshold. Through simulations, we set the alert threshold equal to 0.7 ($\theta = 0.7$). SSA consists of the worst case scenario where the system is based on a single sensor for delivering an alert. The decision process in the SSA is as follows: **SSA-D1** Stop and deliver an alert, if $|\{x_i[t] \geq \theta, i=1,2,\ldots,n\}| \geq 1$; **SSA-D2** Continue to monitor the team of sensors and receive their measurements, if $|\{x_i[t] \geq \theta, i=1,2,\ldots,n\}| = 0$, where | | depicts the cardinality of the set of sensors reporting measurements over θ. AMA produces alerts when the average measurement of sensors is over θ. AMA realizes a linear opinion pool [62], [66] where the opinions of the sensors are of equal weight. The decision process in the AMA is as follows: **AMA-D1** Stop and derive an alert, if $\frac{1}{n}\sum_{i=1}^{n} x_i[t] \geq \theta$; **AMA-D2** Continue to monitor sensors and receive their measurements, if $\frac{1}{n}\sum_{i=1}^{n} x_i[t] < \theta$.



(a) Input variable y *(b) Output variable DoD*

Fig. 3. Type-2 membership functions. The left plot presents the input y's membership functions while the right plot depicts the membership functions for the output variable DoD.

Finally, for the confidence calculation, we get $\gamma = 2$ and $\delta = 5$ as we want to eliminate the confidence of a sensor that reports outliers for more than 5 measurements. In the CumSum algorithm, we get low h^+, h^-, k^+, k^- to produce 'more easily' the outliers and, thus, to derive the aggregated measurement on top of values very close to the mean.

TABLE 3. Values for the adopted parameters.

Parameter	Value / Range
Number of sensors n	{5, 10, 54}
Probability of a faulty value p	{0.01, 0.05, 0.1, 0.2, 0.4, 0.6, 0.8}
Prediction history length M	{5, 10, 50, 100, 500, 1000}
Confidence sliding window W	{10, 20, 50, 100}
Confidence parameter γ	2
Confidence parameter δ	5
CumSum params h^+, h^-, k^+, k^-	0.1
Probability of a missing value ω	{0.4, 0.6}

B. *Performance assessment*

We evaluate the proposed mechanism for a set of scenarios. When non-missing values are present, we still consider the dataset containing values altered as reported in the previous section (i.e., as discussed in [63]). When missing values are the case, we consider that datasets contain 'gaps' in the reported measurements in order to depict the scenario where a single or a subset of sensors does not report anything to the MS (realized by the probability ω).

1) *Data streams without missing values*

We execute a number of experiments based on the Intel Berkeley Research Lab dataset that does not incorporate any indication of an observed event. We compare our mechanism with the SSA mechanism. In Table 4, we present our results for different p values. The SSA mechanism exhibits an increased RoFA. When $p \in \{20\%, 40\%\}$, our mechanism results to RoFA = {0.017, 0.020} while, the SSA mechanism results false alerts that are above the half of the reported measurements. When $p \in \{60\%, 80\%\}$, the majority of the measurements are considered as faulty, however, the T2FLS results 14 false alerts while the SSA results an RoFA close to unity. The false alerts produced by the SSA are 65 times greater than the false alerts produced by the T2FLS. The reason is that the SSA mechanism based on the opinion of single sensors could easily derive an alert, however, this is not the right approach as single measurements could be the result of problems in the reporting process. The proposed T2FLS does not rely on single measurements and identifies the event only when the team measurements accompanied by similar future estimations of the aggregated measurement indicate the presence of an event. In addition, we perform a t-test evaluation for revealing if the improvement offered by our model is significant when compared with the SSA. In Table 5, we depict the t-test critical values. The t-test value achieved in our results is equal to 4.044 which shows that our model offers significant improvement over the SSA for the 99.5% confidence interval.

TABLE 4. False alerts comparison. For p ∈ {1%, 10%} the proposed mechanism does not result any false alert while for p = 5%, the T2FLS results a single alert.

p	RoFA_{SSA}	RoFA_{T2FLS}
1%	0.058	0.000
5%	0.179	0.001
10%	0.356	0.000
20%	0.535	0.017
40%	0.755	0.020
60%	0.831	0.014
80%	0.915	0.014

TABLE 5. T-test critical values.

	Critical Values	
Confidence Interval	**Population Size: 7**	**Population Size: 4**
90%	1.415	1.533
95%	1.895	2.132
99%	2.998	3.747
99.9%	3.499	4.604

We also perform a set of experiments for the contextual data retrieved by a real flood event. In the real dataset streams of data values produced by a set of n = 10 sensors are the case and not faulty values are injected in the dataset. Each measurement is characterized by the *Index of Measurement* (IoM) for each sensor (i.e., the discretized report time). In the real dataset, approximately, from the 45th measurement a flood event is realized. We compare our mechanism with the AMA mechanism concerning the IoA metric. For n = 5, our mechanism derives an alert at the 45th measurement (in average and for W = 10) while the AMA mechanism derives an alert at the 88th measurement. When W = 100, the T2FLS derives the alert at the 37th measurement. For n = 10 and W = 10, our mechanism derives an alert at the 47th measurement (in average) while the AMA mechanism derives the alert at the 45th measurement. When W = 100, the T2FLS derives the alert also at the 47th measurement. We observe that the proposed mechanism is able to identify the event just in its beginning. The interesting is that the T2FLS can identify the event even for a low number of sensors n. The SSA mechanism derives alerts at the 34, 35, 36, 39 measurements for the same dataset. This means that the SSA mechanism derives false alerts many stages before the real event commences. If we consider that the sampling interval for identifying flood is e.g., half of a day, the SSA mechanism will result false alerts five (approximately) days before the real event.

2) *Data streams with missing values*

We study the performance of our mechanism coping with missing values combined with faulty values (i.e., the Intel Berkeley Research Lab dataset with injected faulty values). We experiment with $\omega \in \{0.4, 0.6\}$, i.e., at t, a measurement is observed / received by the MS with probability 1 - ω. In Table 6, we present our results. The T2FLS behaves well due to that incorporates a set of values as inputs depicting different aspects of the team behavior. Concerning the real contextual data, the proposed mechanism results an alert in the 44th measurement irrespectively the value of ω. This means that the MS adopting the T2FLS is robust against missing values as it can identify the event at the correct IoM (i.e., a round before the actual real event). It should be noted that in the case of missing values, we can adopt a moving average exponential smoothing process for filling / imputing the missing values before proceeding with the prediction. Hence, our mechanism will be capable of efficiently providing an insight on future estimations no matter the missing values.

TABLE 6. Experiments with missing values. Missing values affect the RoFA only when combined with a lot of faulty measurements. For instance, when p = 80% and ω = 0.6, the proposed T2FLS produces 76 false alerts instead of 14 where no missing values are present.

	ω = 0.0	**ω = 0.4**	**ω = 0.6**
P	**$RoFA_{T2FLS}$**	**$RoFA_{T2FLS}$**	**$RoFA_{T2FLS}$**
1%	0.000	0.000	0.000
5%	0.001	0.000	0.000
10%	0.000	0.000	0.000
20%	0.017	0.015	0.019
40%	0.020	0.018	0.022
60%	0.014	0.014	0.039
80%	0.014	0.025	0.076

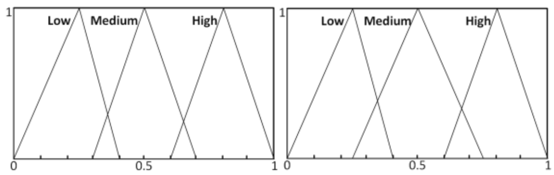
We perform another set of experiments where we try to study the detection rate of the proposed mechanism. We consider the real flood dataset and replicate the available measurements. Through this approach, we create an extended dataset where 14 flood events are present. We get $W = M = 10$. We run our mechanism and see if it can identify the correct measurement where the event takes place. We assume that when the mechanism results the event at the index of the measurement indicating the start of the real event, the mechanism successfully concludes the identification of the event. Our experimental results show that the proposed mechanism exhibits a detection rate equal to 100% (approximately) for ω = 0.0. When ω = 0.4, the detection rate is equal to 86% and when ω = 0.6, the detection rate is 71%. The 'calibration' process performed by the T2FLS leads to an increased detection rate even many missing values.

C. *Comparison with a Type-1 fuzzy inference system*

We compare the T2FLS with a Type-1 FLS (T1FLS) for the Intel Berkeley Research Lab dataset with injected faulty values. T1FLS has two inputs: *(i)* the aggregated measurement y and *(ii)* the predicted aggregated measurement y^*. T1FLS does not rely on the DoC as the T2FLS does. Additionally, the confidence level c_i for each sensor s_i as adopted in the T1FLS is randomly selected in [0, 1]. Finally, the FL rule base of the T1FLS is presented in Table 7 while example membership functions are depicted in Fig. 4.

TABLE 7 The T1FLS rule base.

Rule	***y***	***y^****	**DoD**
1	Low	Low or Medium	*Low*
2	Low	High	*Medium*
3	Medium	Low or Medium	*Medium*
4	Medium	High	*High*
5	High	Any[5]	*High*



(a) Input variable y *(b) Output variable DoD*

Fig. 4. Example membership functions for the T1FLS. The left plot depicts the y membership functions while the right plot presents the membership functions for DoD. Functions for y^* are defined similarly.

We define the metric d that quantifies the difference in the RoFA metric for both T1FLS and T2FLS, i.e.,

$$d = \frac{RoFA_{T2FLS} - RoFA_{T1FLS}}{RoFA_{T1FLS}} \cdot 100\% \tag{6}$$

[5] With the term *Any*, we depict the entire set of the available Fuzzy Sets.

where the $RoFA_{T1FLS}$ and $RoFA_{T2FLS}$ depict the RoFA for T1FLS and T2FLS, respectively. A negative *d* means that the T2FLS exhibits better performance than the T1FLS in terms of false alarms, and the opposite stands when $d > 0$. In Table 8, we present our results for $W \in \{10, 20, 50, 100\}$. False alerts that both mechanisms result are below 20 for the majority of cases when $p \in \{1\%, 5\%, 10\%, 20\%\}$. The number of false alerts, for $p \in \{1\%, 5\%, 10\%, 20\%\}$, are depicted in Table 9. We observe that the difference in the number of alerts between the T2FLS and the T1FLS is low. Based on these values, results related to $p \in \{1\%, 5\%, 10\%\}$ could not considered as representative of the performance of the examined mechanisms. The maximum number of false alerts is 34 for the T2FLS while the maximum number of false alerts for the T1FLS is 55. The T2FLS performs better than the T1FLS in the presence of many faulty measurements. The T1FLS adopting crisp membership functions for each fuzzy set is not efficient when multiple faulty values are reported by sensors (membership degrees could be wrongly evaluated). Hence, the number of false alerts increases as the T1FLS identifies multiple occurrences of events. In the T2FLS, the membership degree of each fuzzy set is 'smoothly' evaluated as it is not a crisp value but an interval. As natural, intervals cover greater areas than the crisp values of the T1FLS and, thus, the membership degree of each variable is efficiently evaluated. We perform a t-test evaluation for the T2FLS and T1FLS results. Critical values are depicted in Table 5 (population size equal to 4). In this set of experiments the t-test value is equal to 2.584 that depicts a difference in the performance between the T2FLS and the T1FLS for the 95% confidence interval.

TABLE 8 T2FLS and T1FLS comparison. The T2FLS exhibits better performance than the T1FLS for high W, e.g., W close to 100 and $p \in \{40\%, 60\%, 80\%\}$. The T1FLS mainly outperforms the T2FLS when p = 20%.

	Metric *d*			
p	***W* = 10**	***W* = 20**	***W* = 50**	***W* = 100**
1%	00.00%	00.00%	00.00%	00.00%
5%	-66.67%	33.33%	33.33%	00.00%
10%	00.00%	00.00%	00.00%	00.00%
20%	46.24%	63.44%	37.63%	183.87%
40%	-21.88%	-2.34%	-25.78%	1.56%
60%	-68.33%	-52.49%	-41.18%	-47.96%
80%	-74.66%	-56.56%	-42.08%	-54.75%

TABLE 9 T2FLS and T1FLS False Alerts ($p \in \{1\%, 5\%, 10\%, 20\%\}$ over 1000 values). The number of false alerts is derived as the rounded average value retrieved from a set of experiments over our datasets.

	***p* ∈ {1%, 10%}**		***p* = 5%**		***p* = 20%**	
W	***T2FLS***	***T1FLS***	***T2FLS***	***T1FLS***	***T2FLS***	***T1FLS***
10	0	0	1	3	18	12
20	0	0	4	3	20	12
50	0	0	4	3	17	12
100	0	0	3	3	34	12

In Fig. 5, we present our results for different *M* and the same dataset (i.e., the Intel Berkeley Research Lab dataset with faulty values). In these experiments, we get $p = 40\%$. We observe that as *M* increases the RoFA increases as well (for $M \leq 500$).

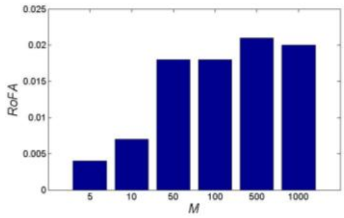


Fig. 5. RoFA against different values of *M*. The more historical data are taken into consideration, the greater the RoFA becomes.

D. Comparison with other fusion techniques

A number of fusion techniques have been proposed over the past years [67]. For depicting the strength of the FL when combined with legacy fusion techniques, we compare our mechanism with two 'typical' fusion models for the Intel Berkeley Research Lab dataset with injected faulty values. The aim is to show how the FL built on top of fusion results adds efficiency to the MS. We consider the fusion model realized by the combination of the CumSum and the linear opinion pool algorithms, i.e., the *Fusion Model* (FM). FM is a 'typical' fusion technique performed by our fusion process (part of the MS). Recall that the result of this process is fed to the proposed T2FLS. In the FM, when applying the linear opinion pool algorithm, we consider that sensors have the same weight. In addition, we compare our mechanism with the *Moving Average Model* (MAM) discussed in [68]. MAM is widely adopted in digital signal processing. The model calculates the mean of a set of input measurements to produce each point of the output. In general, if $\mathbf{z} = (z(1), z(2), \ldots)$ is the input signal, the true signal is $\mathbf{x} = (\hat{x}(1), \hat{x}(2), \ldots)$ where (V is the number of measurements taken into consideration):

$$\hat{x}(k) = \frac{1}{V} \cdot \sum_{i=0}^{V-1} z(k-i) \tag{7}$$

In Table 10, we see our results concerning the RoFA. FM exhibits fair performance when $p \in \{1\%, 5\%, 10\%, 20\%\}$. However, when $p \geq 40\%$, FM produces more alerts than the T2FLS. FM keeps the performance stable (for $p \geq 40\%$), however, false alerts increase as *p* increases. The aforementioned results stand for the Intel Berkeley Research Lab dataset. Concerning the real flood dataset, FM derives an alert in the 42^{nd} and 34^{th} measurements for $\omega = 0.0$ and $\omega \in \{0.4, 0.6\}$, respectively. Recall that the T2FLS derives an alert at the 45^{th} measurement. The identification of the event is realized before the 45^{th} measurement (the real event). Such an approach could cause problems if sensors report their measurements e.g., twice per day (FM will identify the event in the previous or five days before). The interesting is that MAM cannot identify the flood event due to the adopted averaging mechanism. The averaging mechanism requires an increased unanimity in the incoming measurements that could cause problems when an event is realized by only a sub-set of sensors. Finally, in this set of experiments, the t-test value is equal to 1.314 (critical values are depicted in Table 5 for population size equal to 7) that does not depict a significant difference in the performance between the T2FLS and the FM.

TABLE 10. Comparison with other fusion techniques. MAM is not presented as it results no alerts because the arithmetic mean of the sensor measurements does not produce any value over the pre-defined threshold.

p	**RoFA$_{T2FLS}$**	**RoFA$_{FM}$**
1%	0.000	0.000
5%	0.001	0.000
10%	0.000	0.001
20%	0.017	0.005
40%	0.020	0.034
60%	0.014	0.074
80%	0.014	0.142

E. The effect of the alert threshold

We perform a set of experiments for different θ values in order to reveal the effect of the alert threshold in to the discussed mechanisms. We focus on the dataset without missing values (i.e., the Intel Berkeley Research Lab dataset) and we get $\theta \in \{0.5, 0.9\}$. In Table 11, we present the comparison results between the T2FLS and the SSA. The T2FLS results less false alerts than the SSA even for low θ. The interesting is that when $\theta = 0.9$, the T2FLS does not result any false alerts, however, a high θ value could lead, possibly, to missing alerts. Hence, we set $\theta = 0.9$ and experiment with the real flood data. As expected, the T2FLS was not able to identify the event due to the high θ that makes the model more strict to conclude the final decision. The t-test values for this set of experiments are equal to 1.531 ($\theta = 0.5$) and 3.554 ($\theta = 0.9$) - the critical values are depicted in Table 5 for population size equal to 7. T-test values show a significant difference in the performance between the T2FLS and the SSA for the 90% and the 99.5% confidence intervals.

F. Comparison with other models

In [63], the authors discuss a flood prediction scheme and the description of specific installations. In the Massachusetts Dover field test realized on the upper Charles River, the proposed scheme is based on three distinct nodes (i.e., one for rainfall, one for temperature and one for pressure). The retrieved dataset is not provided, however, in this installation, the model produces, in average, 20 false alerts (RoFA = 0.02). We also compare the T2FLS with a model that is based on a *Simple Prediction Model* (SPM) over the sensors measurmements adopting the Intel Berkeley Research Lab dataset with injected faulty values. The SPM is applied on the historical values reported by each sensor and when one of them exceeds the pre-defined threshold, the SPM triggers an alert. The SPM adopts the average of two predictors, the one discussed in Section IV-D (linear predictor) accompagnied by a polynomial predictor. In average, the SPM results 181% more false alerts compared to the T2FLS. Similar outcomes stand for a high number of sensors ($n \rightarrow 15$). In general, the SPM is similar to the SSA mechnism, however, it is based on the future estimate of sensor measurements and, thus, it ‘conveys' the drawbacks of the SSA.

TABLE 11. False alerts for different θ and *p*.

p	θ = 0.5		θ = 0.9	
	RoFA$_{SSA}$	**RoFA$_{T2FLS}$**	**RoFA$_{SSA}$**	**RoFA$_{T2FLS}$**
1%	0.063	0.011	0.033	0.000
5%	0.241	0.080	0.061	0.000
10%	0.413	0.144	0.122	0.000
20%	0.632	0.301	0.228	0.000
40%	0.822	0.473	0.306	0.000
60%	0.908	0.584	0.378	0.000
80%	0.956	0.684	0.501	0.000

G. Complexity analysis

The complexity of the proposed mechanism is affected by the complexity of the fusion, the consensus and the prediction processes. The complexity of the fusion process is equal to $O(n)$ while the worst case complexity of the consensus part is equal to $O(n^2)$. The reason is that the adopted consensus process is based on pairwise comparisons and, thus, we should check the difference of the measurement of a sensor with the remaining observations. The prediction part has complexity of $O(M^2)$. Concerning the FLS part, this depends on the number of inputs, the fuzzy sets, the outputs and the number of the FL rules. For more details, the interested reader could refer in [69] to find more information on the complexity of FL systems. In our case, the proposed T2FLS (recall that it adopts triangular membership functions), the fuzzification phase has complexity of $O(N_I \cdot N_F)$, the inference process has complexity of $O((M_{OD}+N_I) \cdot R)$ and the defuzzification phase has complexity of $O(M_{OD} \cdot R)$ where N_I is the number of inputs, N_F the number of input fuzzy sets, M_{OD} is the number of discretization of the output universe of discourse and R is the number of the FL rules. As parameters N_I, N_F, M_{OD} and R are constant, the complexity of the T2FLS could be considered as negligible. Hence, the worst case complexity of our mechanism is $O(max(M^2, n^2))$. The complexity is mainly affected by the number of sensors and the window of the historical values over which we predict the future behavior of the team.

H. Discussion on the sensors coverage impact

The number of sensors that are enough for the monitoring process depends on the application, the area under consideration and their hardware. This is because all these issues affect the coverage of the area where sensors are placed. A number of research efforts deal with the optimal sensors placement and the coverage problem [70], [71], [72], [73]. Sensor coverage models reflect sensors' sensing capability and quality [71]. They are abstraction models that quantify how well sensors can sense physical phenomena at some locations. A sensor coverage model is mathematically formulated as a coverage function of the distance between a point and the location of the sensor. It has been observed and postulated that different applications would require different degrees of coverage [74]. For instance, a military surveillance application would need a high degree of coverage, because it would require a region to be monitored by multiple nodes simultaneously, such that, even if some nodes cease to function, the security of the region will not be compromised. Environmental monitoring applications might require a low degree of coverage.

Certain coverage problems have been addressed in different research fields. The art gallery problem [70] and the disc covering problem [71] are related to the coverage problem. The art gallery problem refers to the minimum number of observers required to monitor a polygon area and assumes that an observer can watch all the points within its line of sight. The disc covering problem asks for the smallest radius of n identical discs which can be arranged to cover a unit disc. However, solutions of these problems are not directly applicable to our scenario due to the nature of sensor nodes. For instance, sensors have limited sensing range and battery life whereas the observers of the art gallery problem have infinite visibility unless any obstacle appears. In addition, unlike the observers, sensors should communicate with the MS within their limited communication range, thus, the communication range is not infinite.

Moreover, in k-coverage problems, we aim to have a point covered by at least k sensors. Two categories of k-coverage problems have been identified: *(i)* The k-coverage verification: the problem is formulated as a decision problem where an area needs to be verified whether it is k-covered or not; *(ii)* The nodes subset selection k-coverage problem: For a desired coverage degree select a minimal subset of already deployed nodes in an area so that every point of the area is within the sensing range of at least k different sensors. This k-coverage problem has been proved to be NP-hard by reduction to the minimum dominating set problem [73].

Based on the above algorithms (e.g., a modified version of the solution to the art gallery problem), we are able to specify the minimum number of nodes to have a specific coverage level for an area. This number of nodes is involved in the complexity of our proposed mechanism for decision making. As noted, the complexity of our mechanism is $O(max(M^2, n^2))$ where n is the number of nodes and M is the discrete window size (adopted by the prediction process). A number of research efforts focus on the art gallery problem and provide us with an indication of the number of nodes required to monitor an area. For a simple polygon with β vertices, we need: *(i)* at most $n = \lfloor \beta / 4 \rfloor$ nodes according to [75] and [76] and *(ii)* at most $n = \lfloor (3\beta + 4)/16 \rfloor$ nodes according to [77]. For an analysis on the appropriate number of nodes for an area containing 'holes', the interested reader could refer in [78], [79]. The proposed MS spends only a few computational resources and time in order to deliver an efficient event identification methodology. It should be noted that our mechanism can work on any setting including any number of sensors. The definition of the minimum number of sensors for a specific area is beyond the scope of this paper.

VI. Conclusions

We propose a mechanism that combines a set of techniques for the aggregation of sensors measurements (contextual data streams) observing a specific phenomenon. We rely on fusion techniques for aggregating sensors values while outliers are detected and eliminated. We deal with techniques for deriving the consensus degree of the team of sensors according to their observations. We also adopt time-series prediction for estimating future measurements of the team. The discussed techniques are combined and applied into the reported measurements and their results are fed to a Fuzzy Logic System responsible to derive the degree of danger for a specific event. The proposed Type-2 Fuzzy Logic System undertakes the responsibility of handling the uncertainty present in dynamic environments as well as in the definition of membership functions and provides an effective decision making mechanism. A set of simulations over real contextual data exhibit the performance of the proposed mechanism. We show that our mechanism is capable of early identifying events related to a phenomenon while minimizing false alerts.

Our future research plans include the definition of an adaptation mechanism that will be aligned with the environmental characteristics. The adaptation mechanism will provide membership functions and fuzzy rule-base adaptation according to specified performance metrics. The discussed metrics will be affected by the environmental characteristics as reported by the team of sensors. Hence, the system will be fully aligned with changes in the environment and could support more generic applications.

References

[1] Kausar, F., Al Eisa, E., Bakhsh, I., 'Intelligent Home Monitoring Using RSSI In Wireless Sensor Networks', International Journal of Computer Networks and Communications, vol. 14(6), 2012.

[2] Rothenpelier, P., Kruger, D., Pfisterer, D., Fischer, S., 'FleGSens – Secure Area Monitoring Using Wireless Sensor Networks', in International Conference on Sensor Networks, Information, and Ubiquitous Computing, 2009.

[3] Bourgeois, W., Romain, A. C., Nicolas, J., Stuetz, R. M., 'The Use of Sensor Arrays for Environmental Monitoring: Interests and Limitations', Journal of Environmental Monitoring, vol. 5(6), 2003.

[4] Gouveia, C., Fonseca, A., 'New Approaches to Environmental Monitoring: the Use of ICT to Explore Volunteered Geographic Information', GoeJournal, vol. 72, 2008, pp. 185-197.

[5] Hardas, B. M., Asutkar, G. M., Kulat, K. D., 'Environmental Monitoring Using Wireless Sensors: A Simulation Approach', in 1st International Conference on Emerging Trends in Engineering and Technology, 2008, pp. 255-257.

[6] Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., Anderson, J., 'Wireless Sensor Networks for Habitat Monitoring', in WSNA, 2002.

[7] Lozano, J., Suarez, J. I., Arroyo, P., Ordiales, J. M., Alvarez, F., 'Wireless Sensor Network for Indoor Air Quality Monitoring', Chemical Engineering Transactions, vol. 30, 2012.

[8] Fernández-Berni J., Carmona-Galán R., Martínez-Carmona J. F., Rodríguez-Vázquez Á., 'Early forest fire detection by vision-enabled wireless sensor networks', in the International Journal of Wildland Fire, vol. 21, 2012, 938–949.

[9] Bahrepour, M., Meratnia, N., Havinga, P. J. M., 'Automatic Fire Detection: A Survey from Wireless Sensor Network Perspective', Technical Report, University of Twente, 2008.

[10] Chehri, A., Fortier, P., Tardif, P. M., 'Security Monitoring Using Wireless Sensor Networks', in the 5th Annual Conference on Communication Networks and Services Research, 2007.

[11] Hsin, C. F., Liu, M., 'A Distributed Monitoring Mechanism for Wireless Sensor Networks', in Proc. of the WiSe, 2002.

[12] Kim, S., Pakzad, S., Culler, D., Demmel, J., 'Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks', in the IPSN, 2007.

[13] Cardell-Oliver, R., Smettem, K., Kranz, M., Mayer, K., 'Field Testing a Wireless Sensor Network for Reactive Environmental Monitoring', SSNIP, 2004.

[14] Abdulah, P., Waseem, S., Bai, R., Mohsin, I., 'Development of New Water Quality Model Using Fuzzy Logic System for Malaysia', Open Environmental Sciences, vol. 2, 2008, pp. 101-106.

[15] Hatzikos, E., Bassiliades, N., Asmanis, L., Vlahavas, I., ‘Monitoring Water Quality through a Telematic Sensor Network and a Fuzzy Expert System’, Expert Systems, vol. 24(3), 2007, pp. 143-161.

[16] Muhammetoglu, A., Yardimci, A., ‘A Fuzzy Logic Approach to Assess Groundwater Pollution Levels below Agricultural Fields’, Environmental Monitoring Assessment, vol. 118, 2006, pp. 337-354.

[17] Fisne, A., Kuzu, C., Hudaverdi, T., ‘Prediction of Environmental Impacts of Quarry Blasting Operation Using Fuzzy Logic’, Environmental Monitoring Assessment, vol. 174, 2011, pp. 461-470.

[18] Ramadan, A. B., El-Garhy, A., Zaky, F., Hefnawi, M., ‘New Environmental Prediction Using Fuzzy Logic and Neural Networks’, International Journal of Computer Sciences Issues, vol. 9(3), 2012.

[19] Shell, J., Coupland, S., Goodyer, E., ‘Fuzzy data Fusion for Fault Detection in Wireless Sensor Networks’, in Proc. of the UK Workshop on Computational Intelligence, 2010.

[20] Doctor, F., Hagras, H., Lopez, A., Callaghan, V., ‘An Incremental Adaptive Life Long Learning Approach for Type-2 Fuzzy Embedded Agents in Ambient Intelligent Environments’, IEEE TFS, vol. 15(1), 2007.

[21] Zhang, Y., Meratnia, N., Havinga, P., ‘Outlier Detection techniques For Wireless Sensor Networks: A Survey’, IEEE Communications Surveys & Tutorials, vol. 12(2), 2010, pp. 159-170.

[22] Page, E. S., 'Continuous Inspection Scheme', Biometrika, vol. 41(1/2), 1954, pp. 100–115.

[23] Khaleghi, B., Khamis, A., Karray, F. O., Razavi, S. N., ‘Multisensor Data Fusion: A Reviewe of the State-of-the-Art’, Information Fusion, vol. 14, 2013, pp. 28-44.

[24] McClean, S., Scotney, B., Shapcott, M., ‘Aggregation of Imprecise and Uncertain Information in Databases’, IEEE Transactions on Knowledge and Data Engineering, vol. 13(6), 2002, pp. 902-912.

[25] Torres, R., Salas, R., ‘Time-Based Hesistant Fuzzy Information Aggregation Approach for Decision Making Problems’, International Journal of Intelligent Systems, vol. 29(6), 2014, pp. 579-595.

[26] Zhou, S.-M., John, R. I., Chiclana, F., Garibaldi, J. M., ‘On Aggregating Uncertain Information by Type-2 OWA Operators for Soft Decision Making’, International Journal of Intelligent Systems, vol. 25, 2010.

[27] Wang, W., Liu, X., ‘Intuitionistic Fuzzy Information Aggregation Using Einstein Operations’, IEEE TFS, vol. 20(5), 2012, pp. 923-938.

[28] Xia, M., Xu, Z., ‘Hesitant Fuzzy Information Aggregation in Decision Making’, International Journal of Approximate Reasoning, vol. 52, 2011, pp. 395-407.

[29] Izadi, D., Abawajy, J. H., Ghanavati, S., Herawan, T., ‘A Data Fusion Method in Wireless Sensor Networks’, Sensors, vol. 15(2), 2015.

[30] Ding, H., Ji. H., Huang, Z., Li, H., ‘Data Fusion Algorithm Based on Fuzzy Logic’, in Proceedings of the 5th World Congress on Intelligent Control and Automation, 2004.

[31] Escamilla-Ambrosio, P. J., Mort, N., ‘Hybrid Kalman Filter – Fuzzy Logic Adaptive Multisensor Data Fusion Architectures’, in Proceedings of the 42nd IEEE Conference on Decision and Control, 2003.

[32] Ioannou, S., Avrithis, Y., Stamou, G., Kollias, S., ‘Fuzzy Data Fusion for Multiple Cue Image and Video Segmentation’, chapter in Fuzzy Filters and Image Processing, vol. 122, 2003, pp. 195-215.

[33] Levinson, N., ‘The Wiener RMS error criterion in filter design and prediction’, J. Math. Phys., vol. 25, 1947, pp. 261-278.

[34] Brezinski, C., Redivo, Z. M., ‘Extrapolation Methods. Theory and Practice’, North-Holland, 1991.

[35] Griffiths, D.V., Smith, L. M., ‘Numerical methods for engineers: a programming approach’, Boca Raton: CRC Press, 1991.

[36] Fausett, L., ‘Fundamentals of Neural Networks. Architectures, Algorithms and Applications’, Prentice Hall, 1994.

[37] Durbin, J., ‘The fitting of time series models’, Rev. Inst. Int. Stat., vol. 28, 1960, pp. 233-243.

[38] Beliakov, G., James, S., Calvo, T., ‘Aggregating Fuzzy Implications to Measure Group Consensus’, in the Joint IFSA World Congress and NAFIPS Annual Meeting, 2013, pp. 1016-1021.

[39] Beliakov, G., Calvo, T., James, S., ‘Consensus Measures Constrcted from Aggregation Functions and Fuzzy Implications’, Knowledge Based Systems, vol. 55(1-8), 2014.

[40] Tastle, W. J., Wierman, M. J., Dumdum, U. R., ‘Ranking ordinal scales using the consensus measure’, Issues in Information Systems, vol. 185(96–102), 2005.

[41] Szmidt, E., Kacprzyk, J., ‘Analysis of consensus under intuitionistic fuzzy preferences’, in Proc. of the International Conference in Fuzzy Logic and Technology, 2001, pp. 79–82.

[42] Cabrerizo, F. J., Alonso, S., Perez, I. J., Herrera-Viedma, E., ‘On consensus measures in fuzzy group decision making’, in MDAI 2008.

[43] Zadeh, L. A., ‘Fuzzy Logic, Neural Networks, and Soft Computing’, Communications of the ACM, vol. 37(3), 1994, pp. 77-84.

[44] Zadeh, L., ‘Discussion: Probability Theory and Fuzzy Logic are Complementary Rather than Competitive’, Technometrics, vol. 37(3), 1995.

[45] Tsai, P. H., Lin, Y. C., Ou, Y. Z., Chu, E. T. H., Liu, J. W., S., ‘A Framework for Fusion of Human Sensor and Physical Sensor Data’, IEEE Transactions on Systems Man and Cybernetics: Systems, vol. 44(9), 2014, pp.1248-1261.

[46] Katenka, N., Levina, E., Michailidis, G., ‘Local vote decision fusion for target detection in wireless sensor networks’, IEEE Trans. Signal Process., vol. 56(1), 2008, pp. 329–338.

[47] D’ Costa, A. M., Sayeed, A. M., ‘Data versus decision fusion for classification in sensor networks,” in Proc. IEEE Military Commun. Conf., 2003, pp. 585–590.

[48] Clouqueur, T., Saluja, K., Ramanathan, P., ‘Fault-tolerance in collaborative target detection in sensor networks,” *IEEE Trans. Comput.*, vol. 53(3), 2004, pp. 320–333.

[49] Mendel, J., ‘Type-2 Fuzzy Sets: Some Questions and Answers’, IEEE Neural Networks Society, vol. 1, 2003.

[50] Wu, D., ‘On the Fundamental Differences Between Inerval Type-2 and Type-1 Fuzzy Logic Controllers’ IEEE Transaction on Fuzzy Systems, vol. 20(5), 2012, pp. 832-848.

[51] Hagras, H., ‘A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots’, IEEE TFS, vol. 12, 2004, pp. 524–539.

[52] Mendel, J. M., ‘Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions’, Upper Saddle River, NJ: Prentice-Hall, 2001.

[53] Manyika, J., Durrant-Whyte, H., 'Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach', Ellis Horwood, New-York, London, 1994.

[54] El Falou, W., Khalil, M., Duchene, J., Hewson, D., ‘Automatic Threshold Determination for a Local Approach of Change Detection in Long-Term Signal Recordings’, EURASIP Journal on Advances in Signal Processing, vol. 2007, 2007.

[55] Baron, M. I., ‘Nonparametric adaptive change-point estimation and on-line detection’, Sequential Analysis, vol. 19(1-2), 2000, pp. 1–23.

[56] Basseville, M., Nikiforov, I., ‘Detection of Abrupt Changes: Theory and Application’, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[57] Brook, D., Evans, D. A., ‘An Approach to the Probability Distribution of the Cusum Run Length’, Biometrika, vol. 59(3), 1972, pp. 539-549.

[58] Lucas, J. M., ‘Combined Shewhart-CUSUM Quality Control Schemes’, Journal of Quality Technology, vol. 14(2), 1982, pp. 51-59.

[59] Page, E. S., ‘An improvement to Wald's approximation for some properties of sequential tests’, Journal of Royal Statistical Society – Series B, vol. 16(1), 1954, pp. 136-139.

[60] Siegmund, D., ‘Sequential analysis - tests and confidence intervals’, New York: Springer-Verlag, 1985.

[61] Wald, A., ‘Sequential analysis’, New York: John Wiley and Sons, 1947.

[62] Mendel, J. M., ‘Type-2 Fuzzy Sets and Systems: An Overview’, IEEE Computational Intelligence Magazine, vol. 2(2), 2007.

[63] Sharma, A., Golubchik, L., Govidnan, R., ‘On the Prevalence of Sensor Faults in Real World Deployments’, in SECON ’07, 2007.

[64] Baljak, V., Tei, K., Honiden, S., ‘Classification of Faults in Sensor Readings with Statistical Pattern Recognition’, in the 6th International Conference on Sensor Technologies and Applications, 2012.

[65] Zhao, J., Govidnan, R., Estrin, D., ‘Computing Aggregates for Monitoring Wireless Sensor Networks’, in the 1st IEEE SNPA, 2003.

[66] Bates, J. M., Granger, C. W. J., ‘The combination of forecasts’, Operational Res. Q., vol. 20, 1969, pp. 451 - 467.

[67] Nakamura, E., Loureiro, A., Frery, A., Information Fusion for Wireless Sensor Networks: Methods, Models and Classifications’, ACM Computing Surveys, vol. 39(3), art. 9, 2007.

[68] Smith, S., ‘The Scientist and Engineer’s Guide to Digital Signal Processing’, 2nd Ed., California Technical Publishing, San Deigo, CA, 1999.

[69] Kim, Y. H., Ahn, S. C., Kwon, W. H., ‘Computational Complexity of General Fuzzy Logic Control and its Simplification for a Loop Controller’, Fuzzy Sets and Systems, vol. 111, 2000, pp. 215-224.

[70] Marengoni, M., Draper, B. A., Hanson, A., Sitaraman, R. A., ‘A System to Place Observers on a Polyhedral Terrain in Polynomial Time’, Image and Vision Computing, vol. 18(773), 1996.

[71] Wang, B., ‘Coverage Control in Sensor Networks’, Computer Communications and Networks, 1617–7975, Springer Verlag, 2010.

[72] Williams, R., ‘The geometrical foundation of natural structure’, Dover Publications, 1979.

[73] Yang, S., Dai, F., Cardei, M., Wu, J., Patterson, F., ‘On connected multiple point coverage in wireless sensor networks’, International Journal of Wireless Information Networks, vol. 13, 2006, pp. 289–301.

[74] Ghosh, A., Das, S. K., ‘Coverage and Connectivity Issues in Wireless Sensor Networks’, chapter in Mobile, Wireless and Sensor Networks: Technology, Applications and Future Directions, Editors: Ananda, A., Chan, M. C., Ooi, W. T., Wiley, 2006.

[75] Kahn, J., Klawe, M., Kleitman, D., ‘Traditional galleries require fewer watchmen’, SIAM Journal of Algebraic and Discrete Methods, vol. 4, 1983, pp. 194-206.

[76] O' Rourke, J., ‘An alternative proof of the rectilinear art gallery theorem’, Journal of Geometry, vol. 211, 1983, 118-130.

[77] Aggarwal, A., ’The art gallery theorem: its variations, applications, and algorithmic aspects’, Ph.D. Thesis, John Hopkins University,1984.

[78] Hoffmann, F., Kaufmann, M., Kriegel, K., ‘The art gallery theorem for polygons with holes’, In Proceedings of the 32nd IEEE Symposium on the Foundation of Computer Science, 1991, pp. 39-48.

[79] O' Rourke, J., ‘Art Gallery Theorems and Algorithms’, Oxford University Press, New York, 1987.