Priority-based Event Management using Fuzzy Logic for an IoT-BPM Architecture

Abir Ismaili-Alaoui, Ouafae Kasmi, Amine Baïna, Karim Baïna, Khalid Benali, Mostafa Bellafkih

▶ **To cite this version:**

HAL Id: hal-02413950
https://hal.inria.fr/hal-02413950

Submitted on 16 Dec 2019

Priority-based Event Management using Fuzzy Logic for an IoT-BPM Architecture

Abir Ismaili-Alaoui [1,3]*, Ouafae Kasmi[2]*, Amine Baïna[2] , Karim Baïna[1] , Khalid Benali[3] , and Mostafa Bellafkih[2]

[1]Alqualsadi, Rabat IT Center, ENSIAS, Mohammed V University, Rabat, Morocco

[2]STRS Lab., National Institute of Posts and Telecommunications - INPT,
Department of Telecommunications Systems, Network and Services, Rabat, Morocco

[3]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
Emails: abir.ismaili-alaoui@loria.fr, kasmi@inpt.ac.ma, baina@inpt.ac.ma, karim.baina@um5.ac.ma,

khalid.benali@loria.fr, bellafkih@inpt.ac.ma

Abstract— **Internet of things (IoT) world is growing at a breathtaking pace. This new paradigm shift affects all the enterprise architecture layers from infrastructure to business. Organizations are nowadays faced with new challenges to keep their quality of service and competitive advantage over other rival organizations. Business Process Management (BPM) is a field among others that will be affected by this new technology. Both IoT and BPM communicate through events, and effective and efficient management of those events ensures a better communication channel between the IoT physical layer and the Business layer. However, the huge amount of those IoT generated events and sometimes the subtle difference between their criticality level, generate uncertainty regarding their priority level determination. In this paper, we propose a fuzzy logic-based event management approach to estimate the criticality level of the incoming IoT events using two fuzzy inference systems (FIS) and to manage the priority of business process instances triggered by those events. A case study is presented and the obtained results from our simulations demonstrate the benefit of our approach and allowed us to confirm the efficiency of our assumptions.**

Keywords— **Business Process Management, Fuzzy Logic, Internet of things, Event management, Priority, Criticality.**

I. INTRODUCTION

Managing business processes successfully allows organizations to evolve their performance and achieve their business goals, as it allows them to have a clear vision of their objectives. In fact, the performance level of any organization is indirectly linked to the efficiency of its processes and the quality of their models. Business process management is seen as the perfect solution that helps organizations adapt to the strategic, organizational and technical evolution. When an organization adopts a business process management approach, it is in order to have more visibility and control over its activities and interactions between these processes, to be able to model, manage, improve and optimize these processes continuously, and therefore gain in terms of agility, flexibility, and performance. Several researches have been done in this area and aimed at improving business processes, by focusing on the optimization of business processes issues at build time and at run-time from different perspectives: Control-flow perspective, data and event data perspective, scheduling (time and resources) and event management perspective. Business process instances scheduling and event management are considered as a crucial step in the journey of business process performance improvement. Several research contributions in the literature focus on scheduling business process instances, and optimizing resource allocation based on different methodologies and different algorithms such as [1] [2] [3] [4] [5] just to name a few. However, the new digitized era and the rise of several new technologies especially Internet of Things (IoT) implies new business process scheduling problems and challenges linked basically to limited resources or the need to use those resources (especially in case of human resources) in an optimal and flexible way by achieving priority-based scheduling. According to [6] to overcome those challenges a communication channel must be created between Business Process Management (BPM) domain and IoT domain in order to bridge the gap between the business layer and the IoT physical layer. In real-time IT systems, this communication channel is based on events. An event represents everything that happens or is considered to be happening [7], for example, events generated by IoT devices or sensors. Both IoT devices and business processes communicate with their environment through events. In fact, IoT devices collect data and generate new events by sensing their environment, after that, those events trigger specific business process instances or activities according to the specificities of the detected situation. That is why, efficient exploitation and management of these IoT generated events facilitate more this communication and integration between BPM and IoT for any organization and especially ones that deal with incident management processes in supervision and monitoring systems. In our previous work [6], we proposed a combination approach which is based on unsupervised machine learning algorithm that we apply on a set of event sources so that we can classify those sources on different clusters using a score that we calculate for each event source based on the frequency of previously generated events, in order to generate clusters of priorities, used to estimate incoming events priority. However, there is always an uncertainty level regarding the criticality/priority level of the event generated from sources that belong to the same cluster. In fact, the huge amount of these IoT generated events and sometimes the subtle difference between their criticality level generate uncertainty regarding their priority level determination, especially in critical use cases such as health care. This issue can be addressed with the use of Fuzzy Logic, to achieve priority-based event management and then

---

* This work is equally based on respective PhD thesis work

ensure effective IoT-BPM communication. In this paper, we propose a Fuzzy Logic approach for priority-based management of IoT generated events that trigger business process instances, using two Fuzzy Inference Systems (FIS). The integration of Fuzzy Logic in the IoT-BPM architecture leads to better benefits from monitoring perspective, as it helps to handle the uncertainties regarding the criticality level of those events and than the priority level of the instances triggered by those events. The remainder of the paper is organized as follows. The next Section presents a background about Business Process Management and Fuzzy Logic. In Section 3 we present our context of work and motivation scenario. In Section 4 we present an overview of related work for the problem of IoT and BPM integration and communication. Section 5 outlines our proposed approach and methodology. Section 6 is devoted to the presentation of our experimental results and discussions. We conclude the paper in Section 7 and we give an outlook on future work.

II. CONCEPTUAL BACKGROUND

In this Section, we discuss the different concepts related to Business Process Management and Fuzzy Logic.

A. Business Process Management

Business processes represent a set of activities and tasks that exploit the different resources (human and/or machine) of the organization to achieve one or more objectives previously defined, in order to satisfy an internal or external customer. BPM is defined by M. Dumas, et al. as "the art and science of overseeing how work is performed in an organization to ensure consistent outcomes and to take advantage of improvement opportunities" [8]. C. Moller et al. [9] focus on all sides of BPM, by saying that "BPM is a holistic management discipline that uses technology to control and operate the entire business through rules that clearly define business process. BPM is about continuous improvement and optimizing processes to ensure high performance and by achieving agility and flexibility as a tool to gain competitive advantages". This definition shows that BPM is an open and flexible discipline that can interact with several technologies in order to achieve the goals predefined by each organization, and ensure effective and proper functioning of its activities and to meet the requirements of the market [6].

B. Fuzzy Logic

Fuzzy Logic is a mathematical discipline used to handle nonlinear uncertainties that exist in physical systems to model human experience and human decision making behavior, it has been introduced by L. A. Zadeh in 1965 [10]. This concept is characterized by fuzzy sets, membership functions, linguistic variable, fuzzy operations, fuzzy If-Then rules, and Fuzzy Inference Systems [11] [12] [25] [26].

Definitions related to Fuzzy Logic:

- **Fuzzy Sets:** is defined by its membership function. A point in the universe U, x, belongs to a fuzzy set, A with a membership degree $\mu A(x) \in [0, 1]$.
- **Memberships function:** is a function, which determines the degree to which, a given input or output belongs to a specific set.

- **Linguistic variables:** are the input or output variables of the system that allow the representation of numerical values with fuzzy sets.
- **Fuzzy operations:** logical connectives, which are union, intersection, containment, and complement.
- **Antecedent:** is the condition in a fuzzy rule (or "If").
- **Consequent:** is the result in a fuzzy rule (or "Then").
- **Fuzzy Inference Systems (FIS):** is the regrouping of fuzzy rules, the linguistic variables of membership functions and fuzzy reasoning. The FIS is designed for constructing complex and nonlinear relationships between the input and output. In this paper, we use Mamdani system [13] as FIS.

According to [14], the fuzzy processing procedure is realized in four important steps, including Fuzzification, Fuzzy If-Then rule, Aggregation, and Defuzzification. This process is represented as follows:

- **Fuzzification:** is the first step for fuzzy modeling. It represents the process of transforming a crisp or real value into fuzzy sets by using membership functions. In this step, the crisp input values are transformed into linguistic variables.
- **Fuzzy If-Then rule:** Fuzzy rules consist of three parts, antecedent, fuzzy operation, and consequence. For example: if x_1 is A and/or x_2 is B then x_3=C. where, A, B, C are linguistic values, and x_1, x_2, x_3 are linguistic variables.
- **Aggregation** is the combination of the output of every rule into one fuzzy set before the defuzzification step.
- **Defuzzification:** is the process of converting a fuzzy output of a FIS into a crisp output.

In the next Section, we present our context of work and a motivation example.

III. CONTEXT OF WORK AND MOTIVATION SCENARIO

The case study of our research work belongs to Silver Economy domain which is a new industrial sector officially launched in 2013 in France, in order to create personalized services and new technologies that are expected to improve disability free life expectancy or to help dependent elderly people and their caregivers on a day-to-day basis. Supervision and incident management business processes in health-care are considered as event-driven business processes.

The instances of these processes are, in some cases, triggered by IoT generated events, especially in the current research fields related to health-care. Let us consider a video surveillance company that edits an automatic fall detection system for elderly people and offers a 24/7 automatic alert solution and a quick rescue without the intervention of the person in danger. The incident management process used in this case study is based on real-time analysis of alerts received from 24/7 streaming cameras for detecting falls of elderly people. The events that launch our business process instances in this case study are generated by different cameras placed in each patient room in order to detect possible falls. During the launched instance of this business process, the incident (that triggers this instance) is qualified by human agents and classified during the qualification step into four categories (see Figure. 1):
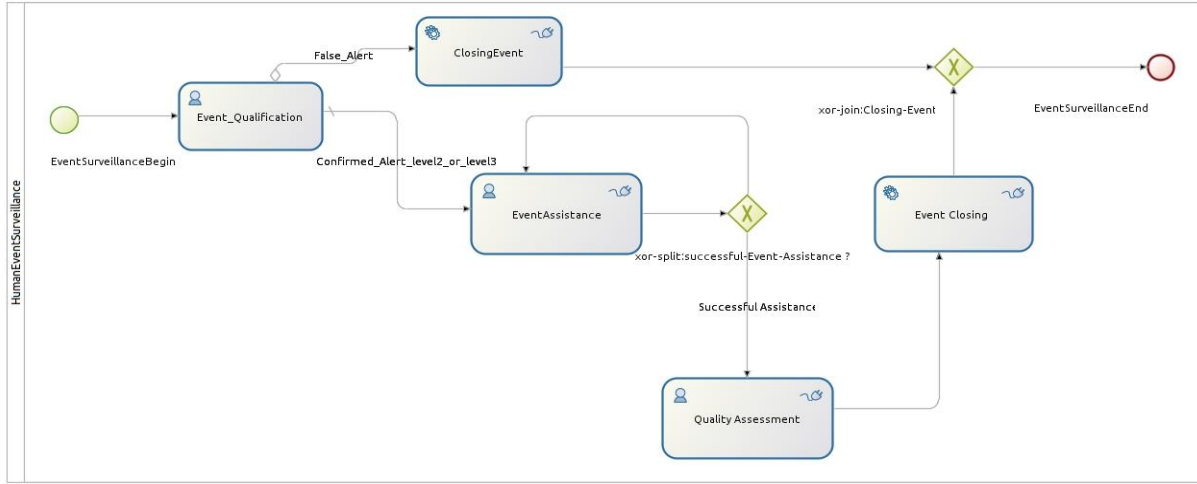
Figure 1: Process model of incident qualification and assessment

false alerts (empty place - level 0), false alerts (active person - level 1), true alerts with average risk level (seated person - level 2), and true alerts with high risk level (lying down person - level 3). The human resource determines whether an assistance action is necessary or not, that is why each received event require an in time and vigilant qualification in order to prevent delayed intervention or incorrect qualification. Because falls can have an adverse physical and sometimes psychological impact on elderly people. Once the incident has been confirmed, a handling step will take place to take care of the patient that triggers this instance. After that, the whole activities go through quality assessment step before closing the event. Different instances of the same process may be simultaneously launched depending on the events generated by the IoT devices, and those instances are allocated to the available resources in chronological order. In such a critical case study, managing efficiently the uncertainty of events priority, is the key success to ensure a priority based execution of business process instances triggered by IoT generated events. In the next Section, we present an overview of related work to the problem of IoT-BPM communication and integration.

IV. RELATED WORK

IoT world is growing at a breathtaking pace, from 2 billion objects in 2006 to a projected 200 billion by 2020, and with 26 smart objects for every human being on Earth. With more than 76 million of smart meters, 30 billion RFID tags, 420 million wearable health monitors. IoT infrastructures can vary from instrumented connected devices providing data externally to smart, and autonomous systems. With this recent increase in IoT device usage organizations are faced with new challenges to keep better customer service and competitive advantages over rival organizations. Business Process Management is a field, among others, that will be affected by this new technology. Several researches are underway to ensure a smooth integration of IoT layer within a global and smart enterprise architecture, and to provide effective communication between the IoT layer and the other layers and especially the business one. As a matter of fact, this IoT integration directly affects business processes at both levels: build-time and run-time. In [15] authors present the challenges that

need to be addressed in Business Process Management Systems (BPMS) to achieve an efficient integration of IoT such as the absence of direct interaction between the business layer and the edge network or the problem of complex and inflexible business process models due to a lack of standardization when modeling IoT elements and components in BPM. In [16] authors propose an approach to achieve an interaction between BPM world and IoT world via a conceptual middle-ware to link between IoT services and BPM applications. This gateway transfers events from IoT network to business processes. Those IoT generated events can trigger business process instances. For authors in [17] the combination between IoT and BPM passes through context interpretation to achieve IoT-awareness in BPM, by integrating data generated by IoT sensors as context information into process models. An end-to-end integration architecture of IoT devices (front-end) and business processes applications (back-end) is proposed in [18], authors in this article use a resource oriented approach inspired by the REST communication paradigm to ensure the integration of IoT components and data into business processes and ensure also an event management through an IoT-aware business process. In the same perspective of event management and IoT-BPM integration, authors in [6] propose an approach inspired by Haze Architecture and Cascading Analytics incarnated by a DIKW (Data Information Knowledge Wisdom) discovery pattern crossing the architecture from device then Fog/Edge to the cloud, and a learning feedback loop that feeds forward insight to adjust either Fog/Edge or device algorithms. This incident/event management approach ensures also a beneficial IoT and BPM communication for achieving a smart IoT event management and also a flexible business process instances scheduling under human resource constraints. Even though there are several proposed approaches in the literature to ensure this IoT-BPM communication at different levels within an organization, there is still a lot of work to do to take full advantage of this collaboration between the IoT domain and the BPM one. We propose in this paper an approach for IoT event management that ensures a beneficial IoT and BPM communication using Fuzzy Logic to determine the criticality level of the IoT generated event and the priority

level of the business process instances triggered by those events. In the next Section, we provide the details of our Fuzzy Logic-based event management approach.

V. PROPOSED APPROACH

The global business process of this case study is simple but it represents several hard functional constraints such as: real-time data analysis and the obligation to maintain limited resources for the viability of the business. An instance of this process is handled by one agent (from end-to-end). That is why, in this paper, we focus on estimating the priority of the whole instance instead of the different tasks of this business process. This Section provides details for our proposed approach. Our goal is to efficiently manage and ensure the communication between events received from IoT devices and business process instances using Fuzzy Logic, which provides a process for formulating the mapping from inputs to output through a Fuzzy Logic system.

Our IoT-FIS-BPM architecture is composed of three main layers: (i) The Edge/Fog based IoT Layer which ensures incident data acquisition and filtering. We use a message broker to facilitate the communication and data exchange between the IoT layer (source) and the Fuzzy Inference System (target). Message broker is responsible for routing, storing, retrieving and transforming the information. (ii) The control layer is decomposed into a series of two FIS, where the output of the first FIS is one of the inputs of the second FIS. These two FIS are used here as a decision-making technique to ensure the processing of the IoT generated events, based on several criteria, in order to determine the criticality level of each event source (FIS 1) and then the priority level of those generated events (FIS 2) and their triggered business process instances. (iii) The BPM layer used to coordinate the execution of business process instances and the resource allocation, taking into consideration the priority levels previously determined by our FIS layer. The communication between the control layer and BPM layer is ensured through an application programming interface (API) (see Figure. 2). In this article, we focus only on the implementation of the control layer, as it represents the core phase of our architecture.

A. Fuzzy Inference System Modeling

Our fuzzy system is divided into two Fuzzy Logic Systems, the first one is for the Patient Criticality Level

determination and the second one is for the Event Priority Level determination. Each system consists of four important steps including Fuzzification, If-Then rules, Aggregation, and Defuzzification. Figure 3 describes different blocks diagram of FIS for Patient Criticality Level and Event Priority Level determination respectively.

1. FIS 1 Modeling

a- inputs and output of FIS 1

For modeling the control system, the first FIS (FIS 1) has two fuzzy parameters as inputs: Patient Status and Device Reliability Level.

- Patient Status represents all the characteristics of a specific patient. It contains:

- **Room location** of the patient.
- **Id event:** represents the identification of each event.
- **Emergency level of the previous fall:** presents the level of emergency of the previous fall.
- **Duration of the previous fall:** represents the number of days between the current and the previous critical fall.
- **Repetitive faller:** represents if the patient is considered as repetitive faller or not (the majority of published studies [19] [20] [21] [22] [23] consider at least two falls to retain repetitive character, with an interval between two falls ranging from 6 to 12 months on average).
- **Patient Score:** represents a weighted mean calculated for each patient based on his/her several previous falls of levels 2 and 3 only.

- Device Reliability Level presents all features of the device: its identification, location, date of the first use, a total of false alerts ranged from 2015 to 2018 and whether the device has been changed or not since its first use. This reliability level is determined based on the state of each device. In fact, The state of the device deteriorates with time, so the device can either generate a lot of false alerts or not detect the real ones. The FIS 1 is characterized also by one output parameter, which is the chance value of Patient Criticality Level. The Patient Criticality Level output refers to the criticality level of each patient that generates an event at a time slot t.
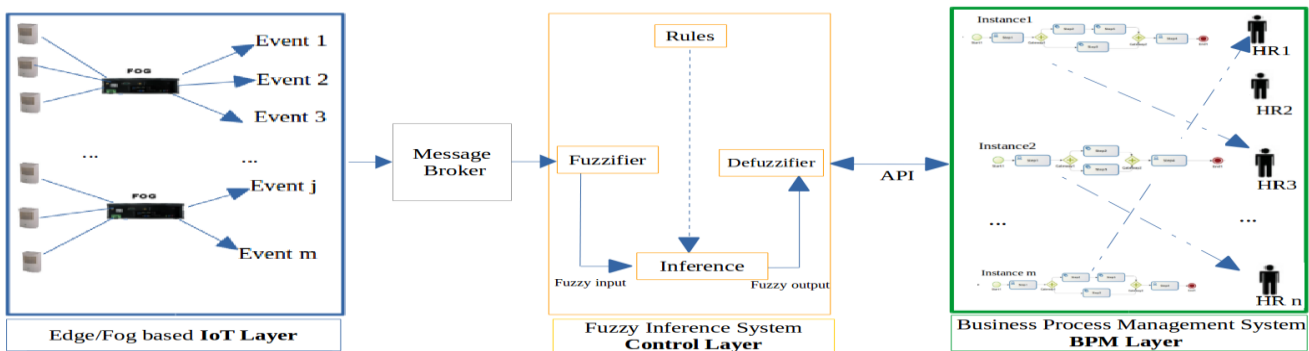

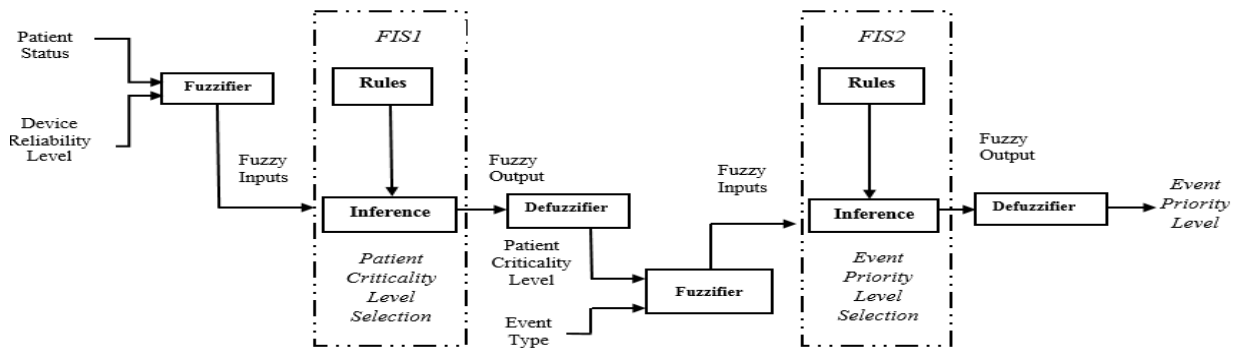Figure 2: IoT-FIS-BPM Architecture for priority-based event management

Figure 3: Fuzzy Inference System (FIS) for Patient Criticality Level and Event Priority Level determination

The numerical values of these inputs are converted into linguistic variables using membership functions during Fuzzification step.

b. Membership Functions

The linguistic values for the Patient Status input are Low, Average, Serious and Very Serious. The second fuzzy input parameter is the Device Reliability Level. The linguistic values for this input are Low, Medium, High and Very High. The output of the first FIS is Patient Criticality Level. This output is divided into four linguistic values as follows: Low, Average, Serious and Very Serious.

The membership functions of all inputs and output are represented by triangular membership functions because this type of functions is the most frequently used and gives better results [11].

c. Fuzzy If-Then rules

The chance values (Patient Criticality Level) is accomplished by using predefined fuzzy If-Then rules to handle the uncertainty. We have two inputs, each divided into four linguistic variables, thus we obtain $2^4=16$ possible chance values. The defuzzification step uses these values to obtain crisp output values.

For the defuzzification process, the method of the Center of Area (CoA) is used [24], called also the Center of Gravity (CoG) method. The concept of this method is that the fuzzy controller determines the range of the output variables according to the area under the scaled membership functions.

2. FIS 2 Modeling

The FIS 2 modeling follows the same steps of FIS 1 modeling as discussed above: Fuzzification, Fuzzy rules, and Defuzzification.

a- inputs and output of FIS 2

The second FIS has two fuzzy input parameters: Patient Criticality Level, which is the output of the first FIS and Event Type, and one output which is the Event Priority Level.

- Event Type input represents the type of the last event generated by the device in question. These events could either be false alerts or true alerts.
- Event Priority Level output is used to choose between two or more events that will trigger two or more business process instances. Those instances may need the same resources at the same time (in case of limited

(human) resources). The event with the lowest priority must wait for the resource occupied by executing the event with the highest priority.

b. Membership Functions

The linguistic values of Patient Criticality Level input are Low, Average, Serious and Very Serious. The second fuzzy input parameter is the Event Type. The linguistic values for this input are True and False. The output of the second FIS is Event Priority Level. This output is divided into eight linguistic values as follows: Very Low, Low, Little Average, High Average, Serious, High Serious and Very Serious.

c. Fuzzy If-Then rules

According to the number of inputs and the number of linguistic variables for each input (2x4) 8 rules are determined. The chance values (Event Priority Level) are accomplished by using predefined fuzzy If-Then rules to handle the uncertainty.

VI. RESULTS AND DISCUSSION

In the following, we present a summary of the results obtained from our experiments, and a comparison with Machine learning-based approach [6]. All our experiments were conducted on an Intel(R) Core(TM) i5- 540 M 2.53 GHz.

A. Simulation settings

The performance of our proposed approach is evaluated using MATLAB. The Fuzzy Logic toolbox allows users to create Fuzzy inference for estimating conclusions problems. The simulation of our system is built through the Graphical User Interface (GUI) Tools using five GUI tools that are being executed to have simulations for the input and output: building, editing FIS, Membership function, Rules, and reviewing Rule and surface. In order to achieve our goal which is the estimation of business process instances priority through the estimation of the criticality level of the events that trigger those instances, we used two Fuzzy Inference Systems. The first FIS is dedicated to determining the Patient Criticality Level, and the second FIS is dedicated to determining the event priority level based on the result of the first FIS. We worked with two datasets as input for the Fuzzy Inference System: the first dataset is an event log from our business process, the second dataset contains IoT-device characteristics.

B. Simulation Results using Fuzzy Logic

1) Simulation Scenario Description

In this section, we describe the results obtained after applying the steps of each FIS, explained in the previous section. To evaluate our proposed approach, we analyzed our historical data (this dataset contains 238228 observations generated by 81 patients) of each event source (IoT device in each patient room) and discuss several cases to estimate the priority level of each received event. To this end, we choose eleven different patients with different IoT devices in eleven different rooms (using 11 patients here only for demonstration purposes). We report four cases of simulation to observe the Event Priority Level in different situations (Low, Average, Serious and Very Serious):

Case 1: Low Patient Status (i.e. a patient considered as a non-critical case) and Low Device Reliability Level.

Case 2: Very Serious Patient Status (i.e. a patient considered as a very critical case) and Very High Device Reliability Level.

Case 3: Low Patient Criticality Level and False Event Type.

Case 4: Very Serious Patient Criticality Level and True Event Type.

2) Analysis and discussion

Case 1: If (Patient Status is Low) and (Device Reliability Level is Low) then the Patient Criticality Level is Low. This rule expresses that if the linguistic value of Patient Status is Low and Device Reliability Level is also Low (the device generates a lot of false alerts), then the output will be Low. As we can see in Figure 5, the input vector contains two linguistic variables (the numerical values corresponding to these linguistic values are 26.6 and 441 respectively) used to determine the Patient Criticality Level as an output parameter. In this case, the patient is considered as a non-critical patient.
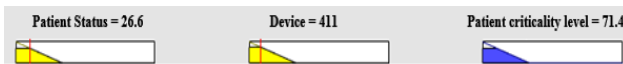

Figure 5: Rule viewer for Case 1

Case 2: If (Patient Status is Very Serious) and (Device Reliability Level is Very High) then Patient Criticality Level is Very Serious. This rule expresses that if the linguistic value of Patient Status is Very Serious and Device Reliability Level is Very High (the device has a low rate of false alerts), then the output will be Very Serious. Figure 6 describes the input vector, which contains two linguistic variables (233 and 4000 respectively) used to determine the Patient Criticality Level (230). In this case, the patient is critical.

Figure 6: Rule viewer for Case 2

Case 3: If (Patient Criticality Level is Low) and (Event Type is False) then the Event Priority Level is Low. This rule expresses that if the linguistic value of the Patient Criticality Level is Low, and the Event Type is False, then the output will be Low. As we can see in Figure 7, the input vector contains two linguistic variables (29.8 and 0.123), used to determine the Event Priority Level output. In this case, the Patient is non-critical and she/he has a lower priority level.
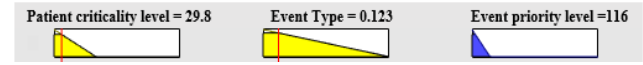

Figure 7: Rule viewer for Case 3

Case 4: If (Patient Criticality Level is Very Serious) and (Event Type is True) then Event Priority Level is Very Serious. This rule expresses that if the linguistic value of the Patient Criticality Level is Very Serious and the Event Type is True, then the output will be Very Serious. Figure 8 describes the input vector, which contains two linguistic variables (456 and 0.959 respectively), used to determine the priority level (423). In this case, the patient is critical and has the highest priority level.
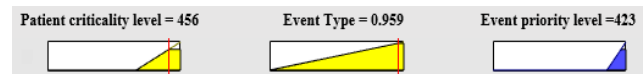

Figure 8: Rule viewer for Case 4

As we can see in Figure 9, the eleven patients that we have randomly chosen for this simulation (represented in the Figure by the ID of the events generated by their devices) have almost the same score value (2.5), except the patient with ID event (313647) which is considered as a non-critical patient. However, those ten patients do not have the same criticality level. So, it is important to efficiently manage the criticality levels of each patient even when the Device Reliability Level is Low (until this device is changed or repaired). As we can see, the criticality level values fall into the range of [65, 500].
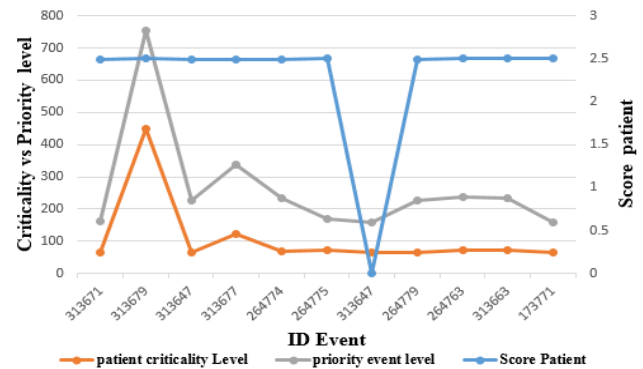

Figure 9: Score patient vs Event priority level vs patient criticality level.

There are 3 important points to consider in Figure 9: the first point is that the score all alone is not enough to determine the criticality/priority level of an event. For example, the Id event (313671) has the same patient score compared to other events, but it has the lower level of criticality and lower event priority. The second point is that the events generated by patients with the highest criticality level, have the highest priority level, as we can see for the Id event (313679). The third point is that the events generated by patients with close values of criticality level have finally different priority level. If we take, for example, the events with Id (264779) and (173771) they are both generated from two different patients

with approximately the same criticality level (64,81) and (64,69) respectively. However, they have different priority levels (162, 45) and (93,81) respectively.

C. Simulation Results using Machine Learning

In our previous work [6] the criticality level was estimated using the score value to create clusters of priority (with K-Means algorithm for clustering). We obtain four clusters with the K-means algorithm based on the score of each patient calculated using the total number of his/her falls, taking into consideration level 2 and 3 only. So the patients in the same cluster have the same profile and then the same criticality/priority level.

TABLE I: Id Event and Clusters of patient with the same profile based on their score value

Id Event	Score	Id Patient	Id Cluster
313671	2.4875	70	Cluster 3
313679	2.4979	49	Cluster 3
264779	2.4995	30	Cluster 3
173771	2.4989	22	Cluster 3

The Event Priority Level is estimated based on the criticality level of the patient (source) that generates this event. With the clustering-based approach, there is a bijection between criticality levels and priority levels. Two scenarios are encountered when applying this approach: (i) the patients belong to different clusters. (ii) both patients belong to the same cluster. The subtle difference between the criticality level in the second scenario generates uncertainty regarding their priority level determination (See Table 1). As we can see in this Table, the four patients belong to the same cluster (Cluster 3) in spite of their different scores, and in this case their generated events will have eventually the same priority level.

D. Comparison: Fuzzy Logic Vs Machine Learning

- Methods Comparison

In the following, we provide a brief comparison between Fuzzy Logic and K-Means algorithm. As we can see in Table 2, when using Fuzzy Logic, the decision-making process for determining criticality/priority level is based on multicriteria (Patient Status, Events Type, etc), which are represented by linguistic variables, and it is based also on the uncertainty management problem in the case of the events that have the same characteristics (score).

TABLE II: Comparison: Fuzzy Logic Vs K-Means

Fuzzy Logic	K-Means
Uncertainty-based decisions	Clustering-based decisions
Human reasoning: inspired by the processes of human perception and cognition	Based on distance (Euclidean distance or Manhattan distance): to identify the set of objects with similar characteristics
Linguistic Variables	Features
Building decisions using multicriteria	Building decisions through partitioning n observations into k clusters

The whole decision-making process in Fuzzy Logic is inspired by human reasoning. While the k-Means method is based on the partition of n observations into k cluster for decision-making. This algorithm works by choosing k initial cluster centers and then assigns every data point to the nearest cluster (based on distance) based on the provided features that define the resulting clusters (using feature similarity).

- Complexity analysis

The computational complexity of the Fuzzy Logic method is presented as follows: $O(N_{rule} N_{dim})$, where N_{rule} is the number of fuzzy rules and N_{dim} is the number of dimensions of the input [27]. For K-Means method, the computational complexity is expressed as $O(n^2)$, where n is the input data size. So its performance is directly proportional to the square of the size of the dataset used as input. Generally, K-Means performance deteriorates when using very large datasets. With this computational complexity, K-Means can be seen as a greedy algorithm [28]. Improving the computational complexity of the K-Means algorithms is basically linked to the initial inputs (selecting of better initial centroids, choosing an effective number of clusters, ...), and to the execution conditions (using parallel and distributed execution environment, reducing the number of scans over the dataset, especially for very large datasets,...). As we can see, Fuzzy Logic method offers a lower computational complexity in our case compared to the K-Means algorithm.

The simulation results prove that the integration of the Fuzzy Logic in the IoT-BPM architecture has been advantageous for the priority-based event management. From the results above, we can consider that the performance of the proposed architecture, at this step, has been achieved. The highest priority level is assigned to the highest criticality level (i.e a critical patient). Thus, it has a positive impact on managing those events and on reducing the waiting time of some critical cases, especially when the organization has a limited number of human resources. Both methods have advantages and disadvantages. In fact, K-means is considered as an easy to implement algorithm that gives easy to interpret results. However, it can be difficult to predict the K value, besides the initial inputs (k, features, the order of the data, dataset quality and tidiness, ...) have a huge impact on the final clustering results. Same thing for Fuzzy Logic. In fact, it is characterized by its simplicity and flexibility. However, in a high complex system, using Fuzzy Logic become an obstacle to verify the system reliability.

VII. CONCLUSION

In this paper, we presented an IoT-FIS-BPM architecture to ensure effective communication between IoT layer and BPM layer through event management. This approach is based on the integration of a Fuzzy Inference System in order to determine the criticality level of the IoT generated events on the one hand (FIS 1), and the priority level of the business process instances triggered by those events on the other hand (FIS 2). The main advantage of this approach is that the Fuzzy logic allows us to handle the degree of uncertainty regarding the criticality/priority level of the events generated by some sources that may have the same characteristics. This proposed approach can be applied for (IoT) generated events triggering any other entities not only

business processes, for example, telecommunication application for management services, or any system that deals with concurrent access to shared resources (human and/or machine) where priority/criticality level determination is important for the viability of the business. The simulation results prove that the events generated by different patients are not treated and managed in the same way using Fuzzy Logic, even if the patients have similar profiles, unlike the K-Means approach. To avoid the limitations encountered in this paper and to enhance event management in this IoT-BPM architecture, we focus in our future work on how the advantages of both methods can be merged in order to develop a hybrid algorithm based on these technologies. And we evaluate the end-to-end architecture with more experiments.

REFERENCES

[1] K. Bessai, S. Youcef, A. Oulamara, C. Godart, and S. Nurcan, "Business process scheduling strategies in cloud environments with fairness metrics," in International Conference on Services Computing, pp. 519-526, IEEE, 2013.

[2] M. Arias, E. Rojas, J. Munoz-Gama, and M. Sepúlveda, "A framework for recommending resource allocation based on process mining," in International Conference on Business Process Management, pp. 458-470, Springer, 2016.

[3] A. Ismaili-Alaoui, K. Benali, K. Baïna, and J. Baïna, "Business process instances scheduling with human resources based on event priority determination," in International Conference on Big Data, Cloud and Applications, pp. 118-130, Springer, 2018.

[4] Z. Huang, W. M. van der Aalst, X. Lu, and H. Duan, "Reinforcement learning based resource allocation in business process management," Data & Knowledge Engineering, vol. 70, no. 1, pp. 127-145, 2011.

[5] K. Bessai and F. Charoy, "Business process tasks-assignment and resource allocation in crowdsourcing context," in 2nd International Conference on Collaboration and Internet Computing (CIC), pp. 11-18, IEEE, 2016.

[6] A. Ismaili-Alaoui, K. Baïna, K. Benali, and J. Baïna, "Towards smart incident management under human resource constraints for an iot-bpm hybrid architecture," in International Conference on Web Services, pp. 457-471, Springer, 2018.

[7] D. Luckham and R. Schulte, "Epts event processing glossary v2. 0, "Event Processing Technical Society, 2011.

[8] M. Dumas, M. La Rosa, J. Mendling, H. A. Reijers, et al., Fundamentals of business process management, vol. 1, Springer, 2013.

[9] C. Moller, C. J. Maack, and R. D. Tan, "What is business process management: A two stage literature review of an emerging field," in Research and practical issues of enterprise information systems II, pp. 19-31, Springer, 2007.

[10] B. Bouchon-Meunier, "La logique floue et ses applications," 1995.

[11] O. Kasmi, A. Baina, and M. Bellafkih, "Multi level integrity management in LTE/LTE-A networks," Advances in Science, Technology and Engineering Systems Journal (ASTESJ), Vol. 2, No. 3, pp. 658-668, 2017.

[12] C. W. De Silva, Intelligent control: fuzzy logic applications. CRC press, 2018.

[13] V. Kamboj and A. Kaur, "Comparison of constant sugeno-type and mamdani-type fuzzy inference system for load sensor," International Journal of Soft Computing and Engineering, vol. 3, no. 2, pp. 204-207, 2013.

[14] K. Valášková, T. Klieštik and M. Mišánková, "The role of fuzzy logic in decision making process," 2014.

[15] C. Chang, S. N. Srirama, and R. Buyya, "Mobile cloud business process management system for the internet of things: a survey," ACM Computing Surveys (CSUR), vol. 49, no. 4, pp. 70, 2017.

[16] S. Cherrier and V. Deshpande, "From bpm to iot," in International Conference on Business Process Management, pp. 310-318, Springer, 2017.

[17] R. Song, Y. Wang, W. Cui, J. Vanthienen, and L. Huang, "Towards improving context interpretation in the iot paradigm: a solution to integrate context information in process models," in Proceedings of the 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, pp. 223-228, ACM, 2018.

[18] K. Dar, A. Taherkordi, H. Baraki, F. Eliassen, and K. Geihs, "A resource oriented integration architecture for the internet of things: A business process perspective," Pervasive and Mobile Computing, vol. 20, pp. 145-159, 2015.

[19] P. A. Stalenhoef, J. P. Diederiks, L. P. de Witte, K. H. Schiricke, and H. F. Crebolder, "Impact of gait problems and falls on functioning in independent living persons of 55 years and over: a community survey," Patient Education and Counseling, vol. 36, no. 1, pp. 23-31, 1999.

[20] P. A. Stalenhoef, J. P. Diederiks, J. A. Knottnerus, L. P. de Witte, and H. F. Crebolder, "The construction of a patient record-based risk model for recurrent falls among elderly people living in the community," Family Practice, vol. 17, no. 6, pp. 490-496, 2000.

[21] P. Stalenhoef, J. Diederiks, J. Knottnerus, A. Kester, and H. Crebolder, "A risk model for the prediction of recurrent falls in community-dwelling elderly: a prospective cohort study," Journal of clinical epidemiology, vol. 55, no. 11, pp. 1088-1094, 2002.

[22] V. S. Stel, S. M. Pluijm, D. J. Deeg, J. H. Smit, L. M. Bouter, and P. Lips, "A classification tree for predicting recurrent falling in community-dwelling older persons," Journal of the American Geriatrics Society, vol. 51, no. 10, pp. 1356-1364, 2003.

[23] K. A. Faulkner, M. S. Redfern, J. A. Cauley, D. P. Landsittel, S. A. Studenski, C. Rosano, E. M. Simonsick, T. B. Harris, R. I. Shorr, H. N. Ayonayon, et al., "Multitasking: association between poorer performance and a history of recurrent falls," Journal of the American Geriatrics Society, vol. 55, no. 4, pp. 570-576, 2007.

[24] T. Korol, "A fuzzy logic model for forecasting exchange rates," Knowledge-Based Systems, vol. 67, pp. 49-60, 2014.

[25] S. Chaudhari, M. Patil, & J. Bambhori. "Study and review of fuzzy inference systems for decision making and control". American International Journal of Research in Science, Technology, Engineering & Mathematics, 14(147), pp. 88-92, 2014.

[26] O. Kasmi, A. Baina, and M. Bellafkih, "Multi level integrity management in critical infrastructure." In 11th International Conference on Intelligent Systems: Theories and Applications (SITA), pp. 1-6, IEEE, 2016.

[27] K. Balázs, L.T. Kóczy, and J. Botzheim, "Comparison of fuzzy rule-based learning and inference systems," In Proceedings of the 9th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, pp. 61-75, 2008.

[28] K. Pakhira Malay, "A linear time-complexity k-means algorithm using cluster shifting," International Conference on Computational Intel-ligence and Communication Networks. IEEE, pp. 1047-1051, 2014.