Towards Multi-perspective Conformance Checking with Fuzzy Sets

Sicui Zhang[1,2], Laura Genga[2], Hui Yan[3], Hongchao Nie[4], Xudong Lu[1,2]*, Uzay Kaymak[1,2]

[1] Department of Biomedical Engineering and Instrumental Science, Zhejiang University, Hangzhou (P.R. China)
[2] School of Industrial Engineering, Eindhoven University of Technology, Eindhoven (The Netherlands)
[3] Department of Biomedical Engineering, Hainan University (P.R. China)
[4] Philips Research, Eindhoven (The Netherlands)

Received 24 September 2020 | Accepted 23 November 2020 | Published 26 February 2021

Abstract

Nowadays organizations often need to employ data-driven techniques to audit their business processes and ensure they comply with laws and internal/external regulations. Failing in complying with the expected process behavior can indeed pave the way to inefficiencies or, worse, to frauds or abuses. An increasingly popular approach to automatically assess the compliance of the executions of organization processes is represented by alignment-based conformance checking. These techniques are able to compare real process executions with models representing the expected behaviors, providing diagnostics able to pinpoint possible discrepancies. However, the diagnostics generated by state of the art techniques still suffer from some limitations. They perform a crisp evaluation of process compliance, marking process behavior either as compliant or deviant, without taking into account the severity of the identified deviation. This hampers the accuracy of the obtained diagnostics and can lead to misleading results, especially in contexts where there is some tolerance with respect to violations of the process guidelines. In the present work, we discuss the impact and the drawbacks of a crisp deviation assessment approach. Then, we propose a novel conformance checking approach aimed at representing actors' tolerance with respect to process deviations, taking it into account when assessing the severity of the deviations. As a proof of concept, we performed a set of synthetic experiments to assess the approach. The obtained results point out the potential of the usage of a more flexible evaluation of process deviations, and its impact on the quality and the interpretation of the obtained diagnostics.

Keywords

Business Process, Conformance Checking, Data Perspective, Fuzzy Sets.

DOI: 10.9781/ijimai.2021.02.013

I. Introduction

Nowadays organizations often need to employ data-driven techniques to audit their business processes and ensure they should comply to the predefined *process models* with internal/external regulations, e.g., on the execution time or other data perspective constraints. Failing in complying with the expected process behavior can indeed pave the way to inefficiencies or, worse, to frauds or abuses, which often result in loss of money and/or reputation which can have a strong impact on the organization. In recent years, alignment-based *conformance checking* [1] emerged as a widely used approach for organization process auditing. These techniques allow to automatically detect possible discrepancies between real-world process executions and the expected process behavior, usually represented by means of some modelling formalism (e.g., Petri net, or BPMN) [1]-[5].

However, state of the art techniques suffer from some limitations. Processes often involve several alternative execution paths, whose choice can depend on the values of one or more data variables. While this aspect has been traditionally neglected in conformance checking, typically focused on the control flow perspective [1]-[4], recently a few approaches have been proposed to assess process compliance with respect to multiple perspectives [5], [6]. However, existing techniques consider an activity performed at a given point of an execution either *completely deviated* or *completely correct*. Such a crisp distinction is often not suitable in many real-world processes, where decisions on data-guards are often generated with some level of *uncertainty*, which gives rise to some challenges in drawing exact lines between acceptable/not acceptable values. As a result, in these domains there often exists some tolerance to deviations. For example, let us assume that in a medical process there is a guideline stating that in between two procedures there must be an interval of at most five hours. Adopting a crisp evaluation, 4 hours 59 minutes would be considered fully compliant, while 5 hours and 1 minute would be fully not compliant, which is intuitively unreasonable. Such an approach can lead to generating misleading diagnostics, where executions marked as deviating actually correspond to acceptable behaviors. Furthermore, the magnitude of the deviations is not considered; small or large violations are considered at the same level of compliance, which can easily be misleading to the diagnosis. It is worth noting that this approach can also hamper the overall process resilience, making it very sensible even to small exceptions/disruptions. For instance, if process executions are monitored in a real-time way, every small deviations can lead to raise some alarms and/or to stop the execution.

* Corresponding author.
E-mail address: lvxd@zju.edu.cn

To deal with these challenges, in this work we perform an exploratory study on the use of *fuzzy sets* [7] in conformance checking. Fuzzy sets have been proven to be a valuable asset to represent human decisions making process, since they allow to formalize the uncertainty often related to these processes. In particular, elaborating upon fuzzy theory, we propose a new multi-perspective conformance checking technique that accounts for the degree of deviations. Taking into account the severity of the occurred deviations allows a) improving the quality of the provided diagnostics, generating a more accurate assessment of the deviations, and b) enhancing the flexibility of compliance checking mechanisms, thus paving the way to improve the overall *resilience* of the process management system with respect to unforeseen exceptions [8]. As a proof-of-concept, we tested the approach over a synthetic dataset.

The rest of this work is organized as follows. Section II discusses related work. Section III introduces a running example to discuss the motivation of this work. Section IV introduces basic formal notions. Section V illustrates the approach. Section VI discusses results obtained by a set of synthetic experiments. Finally, Section VII draws some conclusions and future work.

II. Related Work

Conformance checking discipline has evolved significantly in recent times. One of the first automatic approaches was introduced by [9], which proposed a token-based approach to detect deviations by replaying each event of a process execution against a process model, to determine whether the execution was or not allowed by the model. While this seminal work provides detailed diagnostics, supporting the detection of inserted and skipped activities, and it is able to deal with possible infinite behavior (e.g., in the case of loops), further research proved that token-based techniques can lead to misleading diagnostics [10]. Recently, alignments have been proved to be a robust way to check the conformance of the given logs [2]. Alignment-based techniques are able not only to pinpoint occurred deviations, but also to determine the most probable explanation of non conformity. To this end, a cost function is used to determined the cost of alternative explanations, then returning the one with minimum cost. Although most alignment-based approaches apply the standard distance cost function defined by [2], several variants have been suggested to enhance the quality of the compliance assessment. For instance, Alizadeh et al. [11] proposes a method to obtain the probable explanations for nonconformity by computing the cost function from historical logging data. While traditional conformance checking techniques are solely focused on assessing compliance with respect to the control-flow, i.e., the ordering of the activities, recently few approaches in literature investigated how to include other perspectives, e.g., resources, time, data, and so on in conformance checking algorithms. The approach introduced in [6] suggests to align the control-flow first, and then check the executions compliance with respect to the data perspective. While this approach does allow to detect data-related deviations, it still gives more importance to the control flow perspective when it comes to the deviation interpretation, with the results that he can miss some critical deviations in the alignment [5]. With a different interpretation, the work of [12] considers the data perspective prior to control flow, thus aligning the data variables to the data-aware decision paths first for a reference trace, and next replaying it to the execution trace for the mismatches on control flow conformance. The research in [5], instead, aims at balancing the impact of all the different process perspectives when generating the alignment, considering all perspectives equally important. To this end, they propose a cost function which takes into account both data and control flow deviations simultaneously.

The techniques mentioned above adopt a crisp evaluation of the conformance, where a behaviour is completely wrong or completely correct. In this work, we propose to use fuzzy sets theory to assess the magnitude of the detected deviations. Several researches in literature have explored the employment of fuzzy sets in representing expert decision making processes; among them, we can mention, for example, [13], which studies a fuzzy approach to model farmers' decision process in a integrated farming systems; [14], which represents vagueness in linguistic judgements by means of a fuzzy analytic hierarchy process; [15], which applies a fuzzy dynamic method for risk decision making problems for a mine; and the work of [16], which proposes a fuzzy linguistic method for Multiple Criteria Decision Making (MCDM) problem to Prioritize the elective surgery admission in a local public hospital. However, only a few approaches also explored the use of fuzzy theory in process analysis. [17] proposes to characterize the conformance problem by means of an existing fuzzy rule-based framework ; the study of [18] uses a fuzzy process miner on a clinical data-set to support hospital administrators in improving the performance of their processes (e.g., reducing patients' waiting times). However, to the best of our knowledge, no previous work has exploited fuzzy sets theory in the cost function of conformance checking techniques.

III. Motivating Example

Consider, as a running example, a loan management process derived from previous work on the event log of a financial institute made available for the BPI2012 challenge [19], [20]. Fig. 1 shows the process in BPMN notation. The process starts with the submission of an application. Then, the application passes through a first assessment, aimed to verify whether the applicant meets the requirements. If the requested amount is greater than 10000 euros, the application also goes through a more accurate analysis to detect possible frauds. If the application is not eligible, the process ends; otherwise, the application is accepted. An offer to be sent to the customer is selected and the details of the application are finalized. After the offer has been created and sent to the customer, the latter is contacted to discuss the offer with him/her, possibly adjusting according to her preferences. At the end of the negotiation, the agreed application is registered on the system. At this point, further checks can be performed on the application, if the overall duration is still below 30 days, before approving it.

Let us assume that this process is supported by some systems able to track the execution of its activities in a so-called event log. In practice, this is a collection of *traces*, i.e., sequences of activities performed within the same process execution, each storing information like the execution timestamp of the execution, or other data element [1]. Let the following be two example traces extracted by the system supporting the process at hand (note that we use acronyms of the activities names, for the sake of simplicity)[1]:

$\sigma_1 = \langle (A\_S, \{Amount = 9950\}), W\_FIRST\_A, \perp), (W\_F\_C, \perp), (A\_A, \perp), (A\_F, \perp), (O\_S, \perp), (O\_C, \perp), (O\_S, \perp), (W\_C, \perp), (A\_R, \{Duration = 50\}), (A\_AP, \perp) \rangle$;

$\sigma_2 = \langle (A\_S, \{Amount = 2000\}), W\_FIRST\_A, \perp), (W\_F\_C, \perp), (A\_A, \perp), (A\_F, \perp), (O\_S, \perp), (O\_C, \perp), (O\_S, \perp), (W\_C, \perp), (A\_R, \{Duration = 60\}), (A\_AP, \perp) \rangle$;

Both these executions violate the guard on the *Amount* value; indeed, the activity *W_F_C* should have been skipped, being the requested loan amount lower than 10000. It is worth noting, however, that there is

[1] We use the notation $(act, \{att_1 = v_1, ..., att_n = v_n\})$ to denote the occurrence of activity *act* in which variables $att_1 \ldots att_n$ are assigned to corresponding values $v_1, \ldots v_n$. The symbol $\perp$ means that no variable values are changed when executing the activity.

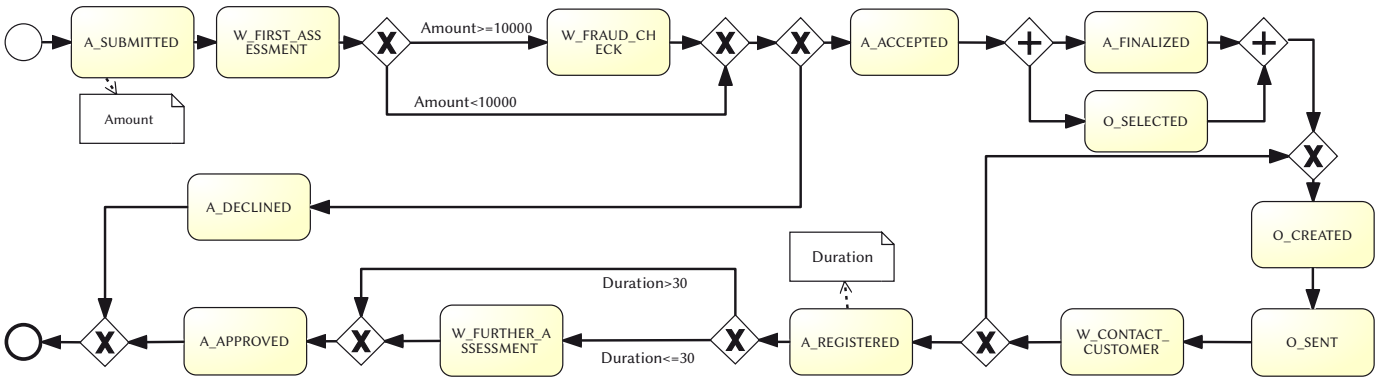


Fig. 1. The Load Management Model.

a significant difference in terms of their *magnitude*. Indeed, while in the first execution the threshold was not reached only by few dozens of euros, the second violation is several thousands of euros below the limit. It is worth noting that applying state-of-the art conformance checking techniques, this difference between σ_1 and σ_2 would remain undetected. Indeed, these techniques adopt a *crisp* logic, where the value of a data variable can be marked only either as correct or wrong.

We argue that taking into account the severity of the violations when assessing execution compliance allows to obtain more accurate diagnostics, especially in contexts where there exists some uncertainty related to the guards definition. Indeed, in these cases guards often represent more guidelines, rather than strict, sharp rules, and there might be some tolerance with respect to violations. In our example, σ_1 could model an execution considered suspicious for some reasons, making a a fraud check worthy, since the amount is only slightly less than 10000. On the other hand, the violation in σ_2 deserves some attention, since the amount is so far from the threshold that the additional costs needed for the fraud check are probably not justified.

Differentiating among different levels of violations also impacts the analysis of possible causes of the deviations. Indeed, conformance checking techniques also attempt to support the user in investigating the *interpretations* of a deviation. In our example, the occurrence of the activity *W_F_C* could be considered either as a control-flow deviation or as a data-flow deviation. In absence of domain knowledge in determining what is the real explanation, conformance checking techniques assess the severity (aka, cost) of the possible interpretations and select the least severe one, assuming that this is the one closest to the reality. In our example, conformance checking would consider both the interpretation equivalent for both the traces; instead, differentiating between the severity of the deviations would make the second interpretation the preferred one when the deviation is limited, like in σ_1, thus providing more guidance to the analyst during process diagnostics.

IV. Preliminaries

This section introduces a set of definitions and concepts that will be used through the paper. First, we recall important *conformance checking* notions; secondly, we introduce basic elements of *fuzzy theory*.

A. Conformance Checking: Aligning Event Logs and Models

Conformance checking techniques detect discrepancies between a process model describing the expected process behavior and the real process execution.

The expected process behavior is typically represented as a *process model*. Since the present work is not constrained to the use of a specific modeling notation, here we refer to the notation used in [2], enriched with data-related notions explained in [6].

Definition 1 (Process model). A process model $M = (P, P_I, P_F, A_M, V, U, T, G, W, Values)$ is a transition system defined over a set of activities A_M and a set of variables V, with states P, initial states $P_I \subseteq P$, final states $P_F \subseteq P$ and transitions $T \subseteq P \times (A_M \times 2^V) \times P$. The function U defines the admissible data values, i.e., $U(V_i)$ represents the domain of V_i for each variable $V_i \in V$; the function G: $A_M \rightarrow Formulas(V \cup \{V_i' \mid V_i \in V\})$ is a *guard* function, that associates an activity to a criterion, i.e., a boolean formula expressing a condition on the values of the data variables; W: $A_M \rightarrow 2^V$ is a write function, that associates an activity with the set of variables which are written/updated by the activity; finally, $Values$: $P \rightarrow \{V_i = v_i, i = 1 .. |V| \mid v_i \in U(V_i) \cup \{\perp\}\}$ is a function that associates each state with the corresponding pairs variable=value.

When a variable $V_i \in V$ appears in a guard $G(A_M)$, it refers to the value just before the occurrence of A_M; however, if $V_i \in W(A_M)$, it can also appear as V_i', and refers to the value after the occurrence. The firing of an activity $s = (a, w) \in A_M \times (V \nrightarrow U)$ in a state p' is *valid* if: 1) a is enabled in p'; 2) a writes all and only the variables in $W(a)$; 3) $G(a)$ is *true* when evaluated over *Values* (p'). To access the components of s we introduce the following notation: $vars(s) = w$, $act(s) = a$. Function *vars* is also overloaded such that $vars(s, V_i) = w(V_i)$ if $V_i \in dom(vars(s))$ and $vars(s, V_i) = \perp$ if $V_i \notin dom(vars(s))$. The set of valid process traces of a process model M is denoted with $\rho(M)$ and consists of all the valid firing sequences $\sigma \in (A_M \times (V \nrightarrow U))^*$ that, from an initial state P_I lead to a final state P_F.

Process executions are often recorded by means of an information system in so-called event logs. In particular, an event log consists of traces, each collecting the sequence of events recorded during the same process execution. Formally, let S_N be the set of (valid and invalid) firing of activities of a process model M; an **event log** is a multiset of traces $\mathbb{L} \in \mathbb{B}(S_N^*)$. Given an event log L, conformance checking builds an *alignment* between L and M, whose goal consists in relating activities occurred in the event log to the activities in the model and vice versa. To this end, we need to map moves" occurring in the event log to possible moves" in the model. However, since the executions may deviate from the model and/or not all activities may have been modeled or recorded [2], we might have log/model moves which cannot be mimicked by model/log moves respectively. These situations are modeled by a "no move" symbol "≫". For convenience, we introduce the set $S_N^{\gg} = S_N \cup \{\gg\}$. Formally, we set S_L to be a transition of the events in the log, S_M to be a transition of the activities in the model. A move is represented by a pair $(s_L, s_M) \in S_N^{\gg} \times S_N^{\gg}$ such that:

- (s_L, s_M) is a *move in log* if $s_L \in S_N$ and $s_M = \gg$
- (s_L, s_M) is a *move in model* if $s_M \in S_N$ and $s_L = \gg$
- (s_L, s_M) is a *move in both without incorrect write operations* if $s_L \in S_N$, $s_M \in S_N$ and $act(s_L) = \text{act}(s_M)$ and $\forall V_i \in V(vars(s_L, V_i) = \text{vars}(s_M, V_i))$
- (s_L, s_M) is a *move in both with incorrect write operations* if $s_L \in S_N$, s_M

$\in S_N$ and $act(s_L) = \text{act}(s_M)$ and $\exists\, V_i \in V \mid vars(s_L, V_i) \neq \text{vars}(s_M, V_i)$)

Let $A_{LM} = \{(s_L, s_M) \in S_N^{\gg} \times S_N^{\gg} \mid s_L \in S_N \vee s_M \in S_N\}$ be the set of all legal moves. The *alignment* between two process executions σ_L, $\sigma_M \in S_N^*$ is $\gamma \in A_{LM}^*$ such that the projection of the first element (ignoring $\gg$) yields σ_L, and the projection on the second element (ignoring $\gg$) yields σ_M.

Given log trace and process model, multiple alternative alignments exist. Our goal is to find the *optimal alignment*, i.e., a complete alignment as close as possible to a proper execution of the model. To this end, the severity of deviations is assessed by means of a *cost function*:

Definition 2 (Cost function, Optimal Alignment). Let σ_L, σ_M be a log trace and a model trace, respectively. Given the set of all legal moves A_N, a *cost function* k assigns a non-negative cost to each legal move: $A_N \to \mathbb{R}_0^+$. The *cost of an alignment* γ between σ_L and σ_M is computed as the sum of the cost of all the related moves: $K(\gamma) = \sum_{(S_L, S_M) \in \gamma} k(S_L, S_M)$. An **optimal** alignment of a log trace and a process trace is one of the alignments with the lowest cost according to the provided cost function.

B. Basic Fuzzy Sets Concepts

Classic sets theory defines crisp, dichotomous functions to determine membership of an object to a given set. For instance, a set N of real numbers smaller than 5 can be expressed as $N = \{n \in \mathbb{R} \mid n < 5\}$. In this setting, an object either belongs to N or it does not. Although crisp sets have proven to be useful in various applications, there are some drawbacks in their use. In particular, human thoughts and decisions are often characterized by some degree of uncertainty and flexibility, which are hard to represent in a crisp setting [21].

Fuzzy sets theory aims at providing a meaningful representation of measurement uncertainties, together with a meaningful representation of vague concepts expressed in natural language and close to human thinking [22]. Formally, a *fuzzy set* is defined as follows:

Definition 3 (Fuzzy Set). Let N be a collection of objects. A *fuzzy set* F over N is defined as a set of ordered pairs $\text{F} = \{n, \mu_F(n) \mid n \in N\}$. $\mu_F(n)$ is called the membership function (μ) for the fuzzy set F, and it is defined as $\mu_F: N \to [0,1]$. The set of all points n in N such that $\mu_F(n) > 0$ is called the **support** of the fuzzy set, while the set of all points in N in which $\mu_F(n) = 1$ is called **core**.

It is straightforward to see that fuzzy sets are extensions of classical sets, with the characteristic function allowing to any value between 0 and 1. In literature several standard functions have been defined for practical applications (see, e.g., [22] for an overview of commonly used functions).

V. Methodology

The goal of this work is introducing a compliance checking approach tailored to take into account the severity of the deviations, in order to introduce some degree of flexibility when assessing compliance of process executions and to generate diagnostics more accurate and possible closer to human interpretation. To this end, we investigate the use of *fuzzy theory*. In particular, we propose to use fuzzy membership functions to model the cost of moves involving data; then, we employ off-shelf techniques based on the use of A* algorithm to build the optimal alignment. The approach is detailed in the following subsections.

A. Fuzzy Cost Function

The computation of an optimal alignment relies on the definition of a proper cost function for the possible kind of moves (see Section [sec:preliminaries]). Most of state-of-the art approaches adopt (variants of) the standard distance function defined in [2], which sets a cost of 1 for every move on log/model (excluding invisible transitions), and a cost of 0 for synchronous moves. Furthermore, the analyst can use *weights* to differentiate between different kind of moves.

The standard distance function is defined only accounting for the control-flow perspective. However, in this work we are interested in the data-perspective as well. In this regards, a cost function explicitly accounting for the data perspective has been introduced by [5] and it is defined as follows.

Definition 4 (Data-aware cost function). Let (S_L, S_M) be a move between a log trace and a model execution, and let, with a slight abuse of notation, $W(S_M)$ to represent write operations related to the activity related to S_M. The cost $k(S_L, S_M)$ is defined as:

$$k(S_L, S_M) = \begin{cases} 1 & \text{if it is a move in log} \\ 1 + |W(S_M)| & \text{if it is a move in model} \\ |\{V_i \in W(S_M): var(S_L, V_i) \neq var(S_M, V_i)\}| & \text{if it is a move in both} \end{cases} \quad (1)$$

In this definition, data costs are computed as a) number of missing data variables because the corresponding activity was skipped, i.e., for a move in model, b) number of data variables in a synchronous move whose values are not allowed according to the process model, i.e., for a move in both.

Compared to Definition 4, in this paper we integrate both data violation situations a) and b), by considering the missing variables as a noncompliance to the rule as well, thereby counting the data cost with a move in both. Besides, the cost function in (1) uses a dichotomous function which considers every move either as *completely wrong* or *completely correct*. To differentiate between different magnitude of deviations, in this work we propose to use fuzzy membership functions as cost functions for the alignment moves. Note that here we focus on data moves. Indeed, when considering other perspectives the meaning of the severity of the deviation is not that straightforward. For example, when considering control-flow deviations, usually an activity is either executed or skipped. Nevertheless, fuzzy costs can be defined also for other process perspectives, for instance, to differentiate between skip of activities under different conditions. We plan to explore these directions in future work.

Following the above discussion, we define our *fuzzy cost function* as follows:

Definition 5 (Data-aware fuzzy cost function). Let (S_L, S_M) be a move between a process trace and a model execution, and let $\mu(var(S_L, V_i))$ be a fuzzy membership function returning the degree of deviation of a data variable in a move in both with incorrect data. The cost $k(S_L, S_M)$ is defined as:

$$k(S_L, S_M) = \begin{cases} 1 & \text{if a move in log} \\ 1 & \text{if a move in model} \\ \sum_{\forall V_i \in V} \mu(var(S_L, V_i)) & \text{if a move in both} \end{cases} \quad (2)$$

To define the fuzzy cost function in (2), we first need to determine over which data constraints we want to define a μ [2]. Then, for each of them first we need to define a tolerance interval; in turn, this implies to define a) an interval for the core of the function, and b) an interval for the support of the function (see Section IV). This choice corresponds to determine, for a given data constraint, which values should be considered equivalent and which ones not optimal but still acceptable. Once the interval is chosen, we need to select a suitable membership function. In literature, several different μ have been defined (see, e.g., [22] for an overview), with different level of complexity and different

[2] Note that multiple μ functions can be defined for the same data variable, if it is used in multiple guards.

interpretations. It is straightforward to see that determining the best μ to explicit the experts' knowledge is not a trivial task. For the sake of space, an extended discussion over the μ modeling is out of the scope of this paper, and left for future work. Nevertheless, we would like to point out that this is a well-studied issue in literature, for which guidelines and methodologies have been drawn like, e.g., the one presented by [23]. The approach can be used in combination of any of these methodologies, since it does not depend on the specific μ chosen.

It is worth noting that on one hand, the cost function (2) can be seen as a direct extension of (1) to the fuzzy case, where the cardinality of a set of differences has been replaced by the cardinality of a fuzzy set (denoting the compliance to a soft constraint). On the other hand, there is also some reasoning behind this formulation of the fuzzy cost function from an aggregation of information perspective. There are various problems in which the deviation from a control-flow perspective is comparable to a deviation in the data perspective in terms of the consequences of the deviation. In this case, an additive cost function makes sense in which the cost incurred from a gradual violation in the data perspective is comparable (or is the same) as the cost incurred from a violation of an activity in the control- flow perspective. Additionally, the cost function in (2) is essentially a penalty function in which different costs are aggregated in additive fashion, implying that a small compliance along one data dimension can be compensated by a large compliance along another data dimension. There is a large class of problems in which such an additive cost function makes sense [24], since good properties in one variable (criterion) can be compensate the poor qualities along another variable (criterion).

In general, it is possible to consider different, more advanced and/or more complex aggregation of the information regarding the violations. Fuzzy set theory provides a rich set of aggregation functions, pre-aggregation functions, and other mathematical formalisms for aggregating the cost information regarding violations [25]. A thorough analysis beyond the additive function is not within the scope of this preliminary paper. However, an initial investigation of using more complex fuzzy set aggregations can be found in [26].

B. Alignment Building: Using A* to Find the Optimal Alignment

The problem of finding an optimal alignment is usually formulated as a search problem in a directed graph [27]. Let $Z = (Z_V, Z_E)$ be a directed graph with edges weighted according to some cost structure. The A* algorithm finds the path with the lowest cost from a given source node $v_0 \in Z_v$ to a node of a given goals set $Z_G \subseteq Z_V$. The cost for each node is determined by an evaluation function $f(v) = g(v) + h(v)$, where:

- g: $Z_V \rightarrow \mathbb{R}^+$ gives the smallest path cost from v_0 to v;
- h: $Z_V \rightarrow \mathbb{R}_0^+$ gives an estimate of the smallest path cost from v to any of the target nodes.

If h is *admissible*,i.e. underestimates the real distance of a path to any target node v_g, A* finds a path that is guaranteed to have the overall lowest cost.

The algorithm works iteratively: at each step, the node v with lowest cost is taken from a priority queue. If v belongs to the target set, the algorithm ends returning node v. Otherwise, v is expanded: every successor v_0 is added to priority queue with a cost $f(v_0)$.

Given a log trace and a process model, to employ A* to determine an optimal alignment we associate every node of the search space with a prefix of some complete alignments. The source node is an empty alignment $\gamma_0 = \langle\rangle$, while the set of target nodes includes every complete alignment of σ_L and M. For every pair of nodes (γ_1, γ_2), γ_2 is obtained by adding one move to γ_1.

The cost associated with a path leading to a graph node γ is then defined as $g(\gamma) = K(\gamma) + \epsilon\,|\gamma|$, where $K(\gamma) = \sum_{(s_L, s_M)\in\gamma} k\,(s_L, s_M)$, with $k\,(s_L, s_M)$ defined as in (2); $|\gamma|$ is the number of moves in the alignment; and ϵ is a negligible cost, added to guarantee termination when implementing the A* algorithm (see [5] for a formal proof). Note that the cost g has to be strictly increasing. While a formal proof is not possible for the sake of space, it is however straight to see that g is obtained in our approach by the sum of all non negative elements; therefore, while moving from an alignment prefix to a longer one, the cost can never decrease. For the definition of the heuristic cost function $h(v)$ different strategies can be adopted. Informally, the idea is computing, from a given alignment, the minimum number of moves (i.e., the minimum cost) that would lead to a complete alignment. Different strategies have been defined in literature, e.g., the one in [2], which exploits Petri-net marking equations, or the one in [28], which generates possible states space of a BPMN model.

VI. Implementation and Experiments

This section describes a set of experiments we performed to obtain a proof-of-concept of the approach. To this end, we compared the diagnostics returned by a crisp conformance checking approach with the outcome obtained by our proposal. In order to get meaningful insights on the behavior we can reasonably expect by applying the approach in the real world, we employ a realistic synthetic event log, introduced in a former paper [29], obtained starting from one real-life logs, i.e., the event log of the BPI2012 challenge[3]. We evaluated the compliance of this log against a simplified version of the process model in , to which we added few data constraints (see Fig. 1). The approach has been implemented as an extension to the tool developed by [28], designed to deal with BPMN models. In the following we describe the experimental setup and the obtained results.

A. Settings

The log in [29] consists of 5000 traces, where a predefined set of deviations was injected. The values for the variable "*Amount*" were collected the from the BPI2012 log, while for calculating "*Duration*" a random time window ranging from 4 to 100 hours has been put in between each pair of subsequent activities, and the overall duration was then increased of by 31 days for some traces. For more details on the log construction, please check [29].

Our process model involves two constraints for the data perspective, i.e., *Amount* >= 10000 to execute the activity *W_F_C*, and *Duration* <= 30 to execute the activity *W_FURTHER_A*. For the crisp conformance checking approach, we use the cost function provided by (1); while for the fuzzy approach, the cost function in (2). Here we assume that *Amount* ∈ (3050,10000) and *Duration* ∈ (30,70) represent a tolerable violation range for the variables. Since we do not have experts' knowledge available for these experiments, we derived these values from simple descriptive statistics. In particular, we draw the distributions of the values for each variable, considering values falling within the third quartile as acceptable. The underlying logic is that values which tend to occur repeatedly are likely to indicate acceptable situations. Regarding the shape of the membership function, here we apply a special trapezoidal function, reported below. *Amount* and *Duration* are abbreviated to *A* and *D*.

$$\mu_1(A) = \begin{cases} 0 & \text{, if } A \geq 10000 \\ 1 & \text{, if } A \leq 3050 \\ \dfrac{10000 - A}{6950} & \text{, if } 3050 < A < 10000; \end{cases}$$

3 https://www.win.tue.nl/bpi/doku.php?id=2012:challenge

$$\mu_2(D) = \begin{cases} 0 & \text{, if } D \leq 30 \\ 1 & \text{, if } D \geq 70 \\ \dfrac{D-30}{40} & \text{, if } 30 < D < 70 \end{cases}$$

B. Results

We compare the diagnostics obtained by the crisp approach and by our approach in terms of a) kind of moves regarding the activities ruled by the guard, and b) distribution of fitness values, computed according to the definition in [6]. Table I shows differences in terms of number and kind of moves detected for the activities *W_F_C* and *W_FURTHER_A* within the crisp/fuzzy alignments respectively, considering also the possible existence of multiple optimal alignments. Namely, when the same move got different interpretations in different alignments, we count the move as both move in log and move in data. Note, however, that the multiple optimal alignments with the same interpretation for the move count one. It is worth noting that while we obtained the same result for both the move-in-log and move-in-data amount for the crisp approach, these values change considerably when considering the fuzzy approach, which returned a significantly smaller amount of move-in-log. The reason for this difference becomes clear by analyzing the boxplots in Fig. 2, which shows the distributions of data deviation severity. We can see that the ranges are similar for both the constraints, with most of the values remaining below 0.65. These distributions suggest that data deviations are mostly within the tolerance range in our dataset; as a consequence, we expect that in most of the cases the move-in-data will have a smaller cost than the move-in-log and will hence be preferred when building the optimal alignment, which justifies the numbers reported in Table I.

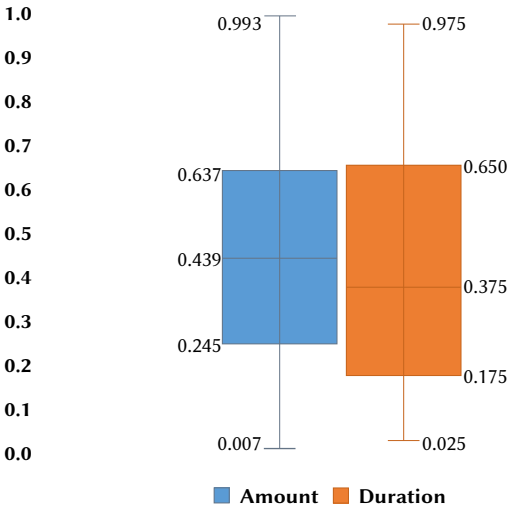


Fig. 2. Boxplots of data deviation.

TABLE I. Number of Different Moves Kinds for Activities *W_F_C* and *W_F URTHER_A*

	W_F_C		*W_FURTHER_A*	
	move-in-log	move-in-data	move-in-log	move-in-data
Crisp	744	744	958	958
Fuzzy	177	744	245	958

From these observations, it follows that we also expect relevant differences in fitness values computed by the fuzzy and the crisp approaches. In particular, we expect to obtain higher values of fitness with the fuzzy approach, being the fuzzy costs less severe than the crisp ones. Fig. 3 shows a scatter plot in which each point represents one trace. The x-axis represents the fitness level of alignment with crisp costs, while the y-axis represents the value corresponding to the fuzzy cost. For the traces on the main diagonal, the fitness level remains unchanged between the two approaches; while for traces that are above the main diagonal, the fuzzy approach obtained higher values of fitness. From the graph we can see that the fuzzy approach never returned lower values of fitness than the crisp one; instead, it returned (also significantly) improved level of fitness for a relevant percentage of the examined cases. Delving into this observation, we found out that the fuzzy approach returns higher value of fitness for 24.3% of the traces.

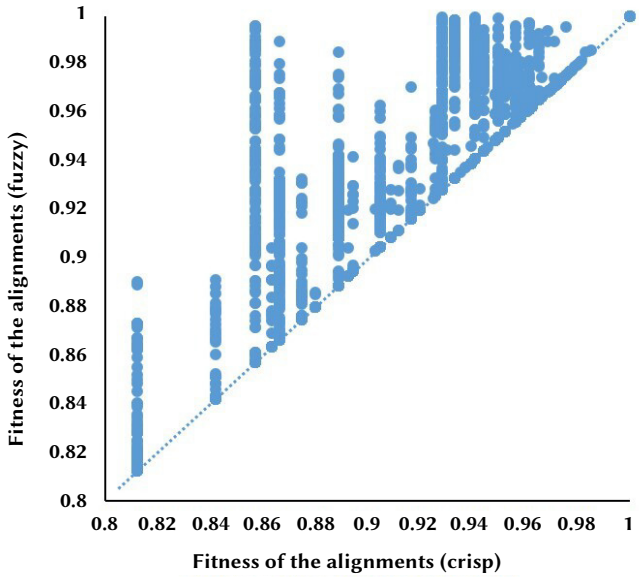


Fig. 3. Comparison of the fitness values obtained with crisp and fuzzy cost.

It is worth noting that, since alignments aim at supporting also the interpretation of the detected deviations, as discussed in Section [sec:motivation], different cost functions also impact the interpretation of the output from an human analyst. To better clarify this aspect, in the following example, we discuss the alignments obtained on one of the traces of our dataset in which the fuzzy and the crisp approach returned different outputs.

Example 1. Let us consider $\sigma = \langle (A\_S, \{Amount = 8160\}), (W\_FIRST\_A, \perp), (W\_F\_C, \perp), (A\_D, \perp), (A\_A, \perp), (A\_F, \perp), (O\_S, \perp), (O\_C, \perp), (O\_S, \perp), (W\_C, \perp), (O\_C, \perp), (O\_S, \perp), (W\_C, \perp), (O\_C, \perp), (O\_S, \perp), (W\_C, \perp), (A\_R, \{Duration = 97\}), (W\_FURTHER\_A, \perp), (A\_AP, \perp), \rangle$. Table II and Table III show the alignment obtained adopting the crisp cost function the fuzzy cost function, respectively. For the sake of space, here we report only the lines of the alignments related to the activities ruled by the data guards. For each move, we report the position of the move in the alignment followed by "#". We can observe that for the second deviation multiple alternative interpretations were returned by both the approaches, either as move-in-log or a move-in-data; indeed, the data deviation is outside the tolerance range, with the result that the costs are equal to 1 both for the move-in-log and for the move-in-data. Instead, the first deviations is always considered as a move-in-data in the fuzzy approach, since the deviation is within the tolerance range and, hence, the cost is less than 1. We argue that this interpretation is reasonably closer to the human's interpretation than the crisp one. Indeed, we can expect that a human analyst would consider the execution of *W_F_A* as correct in this trace, being the data violation negligible. Furthermore, the fuzzy approach returned a higher fitness value for the trace than the crisp one; this is reasonable, since the first deviation is still close enough to the ideal value.

TABLE II. The Optimal Alignments Returned by the Crisp Cost Function

No.	Model	Log	δcost
...	...	...	...
3#	≫	*W_F_C* (*Amount* = 8160)	1
...	...	...	...
18#	≫	*W_F_A* (*Duration* = 97)	1

TABLE III. The Optimal Alignment Returned By a Fuzzy Cost Function

No.	Model	Log	δcost
...	...	...	...
3#	*W_F_C*	*W_F_C* (*Amount* = 8160)	0.265
...	...	...	...
18#	≫	*W_F_A* (*Duration* = 97)	1

Summing-up, the performed comparison did highlight how the use of a fuzzy cost led to improved diagnostics. On the overall fitness level, the fuzzy cost function has obtained higher level of fitness, which represents a more accurate diagnostics [9]. It proves that the fuzzy approach provides a more precise evaluation of the deviation level, taking into account actors' acceptance. In particular, the results show that the fuzzy approach allows to obtain a more fine-grained evaluation of traces compliance levels, allowing the analyst to differentiate between reasonably small and potentially critical deviations. Furthermore, they pointed out the impact that the cost function has on the interpretation of the alignments. Indeed, the approach allows to establish a preferred interpretation in cases in which the crisp function would consider possible options as equivalent, thus reducing ambiguities in interpretation, and providing interpretations for the detected deviations reasonably closer to human analysts' ones.

VII. Conclusion

The present work investigated the use of fuzzy sets concepts in multi-perspective conformance checking. In particular, we showed how fuzzy set notions can be used to take into account the severity of deviations when building the optimal alignment. We implemented the approach and performed a proof-of-concept over a synthetic dataset, comparing results obtained adopting a standard crisp logic and our fuzzy logic. The obtained results confirmed the capability of the approach of generating more accurate diagnostics, as shown both by a) the difference in terms of fitness of the overall set of executions, due to a more fine-grained evaluation of the magnitude of the occurred deviations, and b) by the differences obtained in terms of the different preferred explanations provided by the alignments of the different approaches.

Our results indicate that by exploiting the flexibility in the definition of gradual concepts, conformance analysis from the data perspective is improved. By using fuzzy sets to represent gradual constraints, the penalization of slight violations of the constraints is also made gradual, which reduces the cost associated with a slight violation, and this seems to improve the results of matching between a process model and the event log. Effectively, the fuzzy sets are used to represent a weighting of the violation of business (clinical) rules, which renders the conformance analysis less sensitive to small violations of such rules.

Since this is an exploratory work, there are several research directions that can still be explored. First, in future work we plan to test our approach in real-world experiments, to generalize the results obtained so far. When dealing with real-world experiments, we expect handling of missing values to be an important step in our analysis. There are various methods in which this could be done, such as imputation methods or approaches based on possibility theory in order to deal with the unknown nature of the missing data. Another research direction we intend to explore consists of introducing interval valued fuzzy sets or type-2 fuzzy sets for dealing with the variability that might occur when obtaining the fuzzy sets in our cost function from experts. Inter-expert variability can best be handled with more generic forms of fuzzy sets, which will allow us to extend the flexibility of the analysis process to the process analysts' needs.

Finally, in future work we intend to investigate how to exploit our flexible conformance checking approach to enhance the system on-line resilience to exceptions and unforeseen events.

Acknowledgment

The research leading to these results has received funding from the Brain Bridge Project sponsored by Philips Research.

References

[1] W. Van der Aalst, A. Adriansyah, A. K. A. De Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. Van Den Brand, R. Brandtjen, J. Buijs, et al., "Process mining manifesto," in International Conference on Business Process Management, 2011, pp. 169–194, Springer.

[2] W. Van der Aalst, A. Adriansyah, B. van Dongen, "Replaying history on process models for conformance checking and performance analysis," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 2, no. 2, pp. 182–192, 2012.

[3] A. Adriansyah, B. F. van Dongen, W. M. van der Aalst, "Memory-efficient alignment of observed and modeled behavior," BPM Center Report, vol. 3, 2013.

[4] A. Adriansyah, J. Munoz-Gama, J. Carmona, B. F. van Dongen, W. M. van der Aalst, "Alignment based precision checking," in International Conference on Business Process Management, 2012, pp. 137–149, Springer.

[5] F. Mannhardt, M. De Leoni, H. A. Reijers, W. M. Van der Aalst, "Balanced multi-perspective checking of process conformance," Computing, vol. 98, no. 4, pp. 407–437, 2016.

[6] M. De Leoni, W. M. Van Der Aalst, "Aligning event logs and process models for multi-perspective conformance checking: An approach based on integer linear programming," in Business Process Management, Springer, 2013, pp. 113–129.

[7] S.-C. Cheng, J. N. Mordeson, "Fuzzy linear operators and fuzzy normed linear spaces," in First International Conference on Fuzzy Theory and Technology Proceedings, Abstracts and Summaries, 1992, pp. 193–197.

[8] G. Müller, T. G. Koslowski, R. Accorsi, "Resilience-a new research field in business information systems?," in International Conference on Business Information Systems, 2013, pp. 3–14, Springer.

[9] A. Rozinat, W. M. Van der Aalst, "Conformance checking of processes based on monitoring real behavior," Information Systems, vol. 33, no. 1, pp. 64–95, 2008.

[10] A. Adriansyah, B. F. van Dongen, W. M. van der Aalst, "Towards robust conformance checking," in International Conference on Business Process Management, 2010, pp. 122–133, Springer.

[11] M. Alizadeh, M. de Leoni, N. Zannone, "History-based construction of alignments for conformance checking: Formalization and implementation," in International Symposium on Data-Driven Process Discovery and Analysis, 2014, pp. 58–78, Springer.

[12] W. Song, H.-A. Jacobsen, C. Zhang, X. Ma, "Dependence-based data-aware process conformance checking," IEEE Transactions on Services Computing, 2018.

[13] R. Bosma, U. Kaymak, J. Berg, van den, H. Udo, "Fuzzy modelling of farmer motivations for integrated farming in the vietnamese mekong delta," in The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ'05, United States, 2005, pp. 827–832, Institute of Electrical

and Electronics Engineers.
[14] E. S. Pane, A. D. Wibawa, M. H. Purnomo, "Event log-based fraud rating using interval type-2 fuzzy sets in fuzzy ahp," in 2016 IEEE region 10 conference (TENCON), 2016, pp. 1965–1968, IEEE.
[15] Z. Hao, Z. Xu, H. Zhao, H. Fujita, "A dynamic weight determination approach based on the intuitionistic fuzzy bayesian network and its application to emer-gency decision making," IEEE Transactions on Fuzzy Systems, vol. 26, no. 4, pp. 1893–1907, 2017.
[16] J. Li, L. Luo, X. Wu, C. Liao, H. Liao, W. Shen, "Prioritizing the elective surgery patient admission in a chinese public tertiary hospital using the hesitant fuzzy linguistic oreste method", Applied Soft Computing, vol. 78, pp. 407–419, 2019.
[17] S. Bragaglia, F. Chesani, P. Mello, M. Montali, D. Sottara, "Fuzzy conformance checking of observed behaviour with expectations," in Congress of the Italian Association for Artificial Intelligence, 2011, pp. 80–91, Springer.
[18] K. Ganesha, S. Dhanush, S. S. Raj, "An approach to fuzzy process mining to reduce patient waiting time in a hospital," in 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2017, pp. 1–6, IEEE.
[19] A. Adriansyah, J. M. Buijs, "Mining process performance from event logs: The bpi challenge 2012," in Case Study. BPM Center Report BPM-12-15, BPM-center. org, 2012, Citeseer.
[20] L. Genga, M. Alizadeh, D. Potena, C. Diamantini, N. Zannone, "Discovering anomalous frequent patterns from partially ordered event logs," Journal of Intelligent Information Systems, vol. 51, no. 2, pp. 257–300, 2018.
[21] J.-S. R. Jang, C.-T. Sun, E. Mizutani, "Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [book review]," IEEE Transactions on Automatic Control, vol. 42, no. 10, pp. 1482–1484, 1997.
[22] G. J. Klir, B. Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1995.
[23] A. Cornelissen, J. Berg, van den, W. Koops, U. Kaymak, "Elicitation of expert knowledge for fuzzy evaluation of agricultural production systems," Agriculture, Ecosystems & Environment, vol. 95, no. 1, pp. 1–18, 2003.
[24] J. M. da Costa Sousa, U. Kaymak, Fuzzy Decision Making in Modeling and Control, vol. 27 of World Scientific Series in Robotics and Intelligent Systems. New Jersey: World Scientific, 2002.
[25] G. Beliakov, A. Pradera, T. Calvo, Aggregation Functions: A Guide for Practitioners. Berlin: Springer, 2007.
[26] S. Zhang, L. Genga, L. Dekker, H. Nie, X. Lu, H. Duan, U. Kaymak, "Towards multi-perspective conformance checking with aggregation operations," in Information Processing and Management of Uncertainty in Knowledge-Based Systems, Cham, 2020, pp. 215–229, Springer International Publishing.
[27] R. Dechter, J. Pearl, "Generalized best-first search strategies and the optimality of a," Journal of the ACM (JACM), vol. 32, no. 3, pp. 505–536, 1985.
[28] H. Yan, P. Van Gorp, U. Kaymak, X. Lu, L. Ji, C. C. Chiau, H. H. Korsten, H. Duan, "Aligning event logs to task-time matrix clinical pathways in bpmn for variance analysis," IEEE Journal of Biomedical and Health Informatics, vol. 22, no. 2, pp. 311–317, 2017.
[29] L. Genga, C. Di Francescomarino, C. Ghidini, N. Zannone, "Predicting critical behaviors in business process executions: when evidence counts," in International Conference on Business Process Management, 2019, pp. 72–90, Springer.

Sicui Zhang

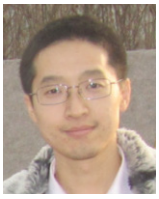
Sicui Zhang is a Ph.D. candidate at Department of Biomedical Engineering of Zhejiang University, China, and Department of Industrial Engineering, Eindhoven University of Technology, the Netherlands. Her research focuses on clinical decision support systems, business process management, conformance checking, and decision making processes.

Laura Genga

Laura Genga, received her Ph.D. in science of engineering at the Università Politecnica delle Marche, Italy, in 2016. Since January 2019, she has been an assistant professor in the Information Systems Group at the Eindhoven University of Technology, the Netherlands. Her core topics involve automated discovery and analysis of flexible pro-cesses, compliance analysis, and on-line process monitoring and prediction to support human analysts in detecting potential threats and in taking decisions regarding current process executions.

Hui Yan

Hui Yan received the first Ph.D. degree in biomedical engineering from Zhejiang University, China, in 2019 and second Ph.D. degree in industrial engineering from Eindhoven University of Technology, the Netherlands, in 2020. She is now working in Hainan University as an assistant professor. Her research interests include care pathway analysis, conformance checking, business process management.



Hongchao Nie

Hongchao Nie obtained the Ph.D. degree from Zhejiang University, China, in 2014 . He is currently a research scientist at Philips Research Eindhoven, the Netherlands. His academic activities have covered image processing, health IT, interoperability and process management. At Philips Research, his active research areas include process analysis, clinical informatics, machine learning and operation research.

Xudong Lu

Xudong Lu received the M.Sc. degree and Ph.D. degree in Biomedical Engineering from Zhejiang University in 1998 and 2001. He is a full professor in Biomedical Informatics Laboratory, Department of Biomedical Engineering, Zhejiang University. He is an openEHR Foundation Management Board Member, member of American Medical Informatics Association, and Hospital Information Management Sys-tem Society since 2007. He achieved several contributions on Business Process Management with Medical Intelli-gence, Guideline-based Clinical Decision Support Systems, Integrated EMR-S in China, and Integrated Physiology Information System through Knowledge Transfer.

Uzay Kaymak

Uzay Kaymak received the M.Sc. degree in electrical engineering, the degree of chartered designer in information technology, and the Ph.D. degree in control engineering from the Delft University of Technology, Delft, The Netherlands, in 1992, 1995, and 1998, respectively. From 1997 to 2000, he was a Reservoir Engineer with Shell International Exploration and Production. He is currently a Full Professor with the School of Industrial Engineering, Eindhoven University of Technology, Eindhoven, the Netherlands. He has co-authored more than 250 academic publications in the fields of intelligent decision support systems, computational intelligence, data mining, and computational modeling methods. His current research interests include fuzzy decision support, interpretable fuzzy modeling, computational intelligence, and intelligent systems design. Dr. Kaymak is an Associate Editor for the IEEE TRANSACTIONS ON FUZZY SYSTEMS and is member of the Editorial Board of multiple journals. He is a Past Chair of the Fuzzy Systems Technical Committee and the Computational Finance and Economics Technical Committee of the IEEE Computational Intelligence Society. He is also a board member of DSC/e (Data Science Centre Eindhoven) and of the Clinical Informatics study program (two-year post-master PDEng study) of TU/e and a member of the program and/or organization committee of multiple international conferences. Dr. Kaymak also holds a visiting professor position at the Zhejiang University, China.