Modeling Uncertainty in Declarative Artifact-Centric Process Models using Fuzzy Logic

Citation for published version (APA):
Eshuis, R., Firat, M., & Kaymak, U. (2021). Modeling Uncertainty in Declarative Artifact-Centric Process Models using Fuzzy Logic. Information Sciences, 579, 845-862. https://doi.org/10.1016/j.ins.2021.07.075

Document license:
CC BY

DOI:
10.1016/j.ins.2021.07.075

Document status and date:
Published: 01/11/2021

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 01. Jun. 2022
Modeling uncertainty in declarative artifact-centric process models using fuzzy logic

Rik Eshuis, Murat Firat, Uzay Kaymak

Abstract

In many business processes, knowledge workers collect information and make decisions about business entities. Artifact-centric process (ACP) models have been proposed to represent such knowledge-intensive processes. Declarative ACP models use precise business rules to define flexible process executions. However, in many business situations knowledge experts have to deal with uncertainty and vagueness. Currently, how to deal with such situations cannot be expressed in declarative ACP models. We propose to use the fuzzy logic framework to model uncertainty in such models. Using Guard-Stage-Milestone (GSM) schemas as declarative ACP notation, we show how GSM schemas can be adapted for modeling gradual progress of real-life knowledge-intensive processes via a fuzzy logic interface. We evaluate the proposed fuzzy GSM schemas in real-life scenarios of airport ground operations and healthcare services.

1. Introduction

Many business processes are performed by knowledge workers, who are responsible for making decisions and adding information, for instance by analyzing data. These knowledge-intensive processes are centered around key business entities to which information is added and about which decisions are made, for instance a production order or a request for quotes. In Business Process Management (BPM), artifact-centric process models have been proposed to model such knowledge-intensive processes. Business artifacts combine data and process aspects of key entities in a holistic way.

Business artifacts contain business rules that specify conditions under which actions are taken. These rules are modeled in a classical way, assuming clear-cut boundaries between different states of the world. However, the variables in the rules may have some degree of uncertainty in reality or they may not be described precisely. This is caused by either having insufficient data or lacking concrete definitions of the concepts considered. For example, there is not a precise loan amount that distinguishes small loans from large loans. In case we have only a small sample of the requested loan amounts, it may not be reliable to use a precise number as the boundary between small and large loans.

Several approaches for modeling uncertainty exist, such as probabilistic modeling or fuzzy modeling. Probabilistic models rely on frequency-based quantitative analysis, while fuzzy models deal with uncertainties due to imprecise
information in a non-probabilistic way, aiming to model approximate human reasoning [16,24]. We propose modeling the uncertainties in knowledge-intensive processes using fuzzy models rather than probabilistic models, since there is a lack of rich data to generate probabilistic models and a need to support the use of knowledge from human experts, for which fuzzy models are especially suited.

Fuzzy models describe a concept using fuzzy sets that represent linguistic terms using corresponding linguistic labels rather than specifying precise numerical values [16]. Each linguistic label has a membership function defined in the domain of the corresponding variable, the universe of discourse. Examples of linguistic labels are “large”, “high”, or “most”, depending on the context. Another important point of using fuzzy modeling is to model the decision-making of knowledge workers, which represents their domain expertise. Modeling expert knowledge properly is clearly beneficial for enabling automated decisions of high quality.

As an example, suppose that a loan is classified as large if it exceeds 100K$, and not large otherwise. In reality, it can be uncertain what exactly comprises the class of large loans. A loan of 99K$ is not large according to the above classification, but obviously it is larger (“more large”) than a loan of 10K$. Knowledge workers may have their own understanding of when a loan is large or small, which fuzzy modeling can capture [24].

In this paper, we use fuzzy sets for declarative artifact-centric process models in which the decision-making process is subjected to high uncertainty due to imprecise information, and involves expert knowledge. As host notation we use Guard-Stage-Milestone (GSM) schemas [15]. Since the language of GSM schemas has inspired the Case Management Model and Notation (CMMN), the industry standard on Case Management [21], our results can also pave a way for incorporating fuzzy reasoning in CMMN schemas. As we explain in Section 5, existing GSM engines can support fuzzy GSM schemas by invoking fuzzy rule evaluators.

Note that in the remainder of this paper, we use the adjective “crisp” to denote concepts that use exclusively non-fuzzy sets as their components. These crisp sets are defined by their two-valued characteristic function. For example, a “crisp milestone” could take numeric values or non-fuzzy sets. We use the adjective “fuzzy” when fuzzy sets are used, which are defined by their membership function taking values in the closed interval [0, 1).

The main contribution of our work is incorporating fuzzy logic in declarative, artifact-centric GSM schemas for allowing uncertainty about GSM concepts like stage closing and achievement levels of milestones. This allows for flexibility in the execution, for instance, opening the next stage while the current stage has not yet fully completed. The building stones of these decision mechanisms are fuzzy rules that are obtained by extracting the expert knowledge. In this manner, we extend the limited way of using fuzzy modeling in GSM schemas introduced in our previous work [9], where we did not consider fuzzy stages and milestones.

The remainder of this paper is organized as follows. We discuss in Section 2 previous literature that applied fuzzy sets to business process models. Section 3 defines preliminaries on GSM schemas and introduces a running example. It also introduces fuzzy modeling. Section 4 gives the basic idea of how we employ fuzzy modeling in GSM schemas. Section 5 formally defines fuzzy GSM schemas, using the running example as an illustration. Section 6 evaluates the use of fuzzy GSM schemas by modeling two real-life scenarios, one concerning an airport operations and the other one a healthcare process. Section 7 ends the paper with conclusions and an outlook for future work.

2. Fuzzy sets in BPM

In recent years, nature-inspired methods based on computational intelligence have attracted interest in the field of Business Process Management (BPM) [5,17]. In this paper, we focus explicitly on fuzzy models. While fuzzy modeling has been applied to many different application domains, such as control and decision making and optimization [24], only a few papers have applied fuzzy modeling in the context of BPM. Vojtek surveys the use of fuzzy logic in BPM [32]. We next highlight the key papers in this area. Thomas et al. [1,29] explore the combination of fuzzy modeling and Event-driven Process Chains (EPCs). They discuss how the resulting fuzzy workflows can replace the classical, crisp workflows. Ye et al. [34] propose the use of fuzzy logic to handle exceptions in workflow management by using fuzzy Petri nets as the underlying model. They focus on extending existing crisp workflow models and crisp workflow systems to incorporate fuzzy reasoning. Fuzzy logic has also been used to analyze business processes, for instance to represent uncertainty for process simulation [23]. Zakarian [36] proposes a quantitative analysis approach for IDEF3 business processes based on fuzzy logic. Bernasconi et al. [4] model and analyze risks in business processes by linking BPMN process models with fuzzy cognitive maps. In contrast to all these works, which consider procedural process models, we consider declarative, artifact-centric process models. In particular, the notions of achievement level of milestones and completion level of stages have no counterpart in the previous approaches.

Next, there are approaches that use fuzzy logic to enact business processes. Saraeian et al. [25,26] present an architecture for a Business Process Management system that deals with uncertain factors. Agents are responsible for modeling, executing and evaluating uncertain processes [26]. They consider procedural, activity-centric business processes, in which fuzzy sets are used to model uncertainty regarding durations of activities [25]. Landry et al. [18] apply fuzzy logic to distributed workflows to support collaborative crisis management under uncertainty. Slavicek [28] presents a fuzzy ontology-based workflow system. The focus is on integrating fuzzy workflow systems with fuzzy ontologies. These papers consider processes that are modeled in a procedural way, whereas we study processes that are modeled in declarative, artifact-centric style. Procedural-
style process models support neither different levels of achievement of business objectives nor different completion levels of activities, which are key ingredients of fuzzy GSM schemas.

In other areas of Business Process Management, fuzzy modeling has been applied as well. For instance, Hakim et al. [13] present a methodology based on fuzzy logic to select processes for reengineering. Bazhenova et al. [3] define how to extract fuzzy decision models from event logs. Zhang et al. [37] use fuzzy sets and fuzzy set aggregation operators for multi-perspective conformance checking. Sarno et al. [27] combine fuzzy decision models and process mining. Hsu et al. [14] model contextual information about activities using fuzzy sets. However, in these papers the processes themselves are not modeled in a fuzzy way, while this paper focuses on fuzzy modeling of knowledge-intensive processes.

3. Preliminaries

In this section, we first introduce GSM schemas in more detail. Next, we explain the basics of fuzzy modeling that is needed to understand our proposed fuzzy GSM schemas.

3.1. GSM Schemas

This section presents formal definitions for the variant of GSM schemas used in this paper. Given the restricted space, the presentation is concise, explaining only those GSM details that are important to understand the remainder of this paper. For an extended introduction to classic GSM schemas, we refer the interested reader to other papers [6,10,15].

We informally introduce GSM schemas by means of an example that is revisited throughout the paper. The example is based on a fragment of a real-world process from an international technology company, in which business criteria for partner contracts are assessed [11].

Example 3.1. Fig. 1 shows the lifecycle part of a GSM schema for a Business Criteria Assessment process BCA [11]. First, client data are gathered. Next, a precheck is performed, and if the precheck is successful, a detailed check is performed. Otherwise, the client is rejected.

Rectangles with rounded corners denote stages, in which work is performed. Circles denote milestones, which are business objectives achieved by completing the work in a stage to which a milestone is attached. Diamonds denote guards, which are conditions, also known as sentries in GSM parlance, under which a stage opens. Stages also have terminators, which are conditions under which a stage closed. However, terminators are not visualized. Milestones use sentries to specify the conditions when milestones are achieved and invalidated. Table 1 shows the sentries for some of the stages and milestones.

A GSM schema $\Gamma$ specifies the data and life cycle structure of an entity instance. A GSM schema $\Gamma$ has data attributes, which hold business-relevant data about an entity instance, and status attributes, which hold life cycle data about an entity instance. Status attributes are subdivided into stage attributes and milestone attributes. If a stage attribute is true (false), the stage is said to be open (closed). If a milestone attribute is true (false), the milestone is achieved (invalid). Each data attribute $a$ has a type $\text{type}(a)$ which is scalar, e.g., string, character, integer, float, etc. Status attributes have type Boolean.

External events correspond to state changes in the environment. An external event is either a named external event, denoted with prefix $E:;$, or the completion event of an atomic stage $S$, denoted with prefix $C:$. External events have a payload, consisting of a subset of the data attributes of $\Gamma$ and the new values for those attributes. Typically, in an atomic stage new values are computed or decided for the subset of the data attributes of $\Gamma$.

Sentries specify conditions under which status attributes change value. A sentry that makes status attribute $a$ True is called a plus sentry of $a$, a sentry that makes $a$ False is called a minus sentry of $a$. If $a$ has multiple plus or minus sentries, then one of the sentries becoming True is sufficient to change $a$.

A sentry $\psi$ has the form $e \land \varphi$ or $\varphi$, where $e$ is an external event and $\varphi$ is a condition formula ranging over the attributes of $\Gamma$. We assume a propositional condition language $\mathcal{C}$ that includes fixed predicates over scalars (e.g., '$<$' over integers or

Table 1

Partial list of sentries for BCA. $S$ = Stage; $M$ = Milestone; ';' separates different sentries.

| Type | Name                          | Plus sentry                                      | Minus sentry          |
|------|-------------------------------|-------------------------------------------------|-----------------------|
| $S$  | Business Performance Evaluation Check | $\text{PreCheck} \land \text{employee_count} \geq 300; \text{E: CheckNeeded} \land \text{PreCheck}$ | $\text{BPECS; BPECU}$ |
| $S$  | Credit Check                  | $\text{PreCheck}$                              | $\text{CCS; CCS}$    |
| $M$  | CCS                           | $C : \text{CreditCheck} \land \text{rating} \geq 8$ | $\text{Credit Check;} \text{CCU}$ |
| $M$  | CCU                           | $C : \text{CreditCheck} \land \text{rating} < 8$ | $\text{Credit Check;} \text{CCS}$ |
| $S$  | Addressable Market Check      | $\text{IDGS} \land \text{annual_revenue} \geq 500K$ | $\text{AMCS; AMCU}$  |
| $M$  | PCS                           | $\text{CCS} \land \text{BPECS} \land \text{AMCS}$; $\text{CCS} \land \text{revenueemployee_count} < 300 \land \text{AMCS}$; $\text{CCS} \land \text{employee_count} < 300 \land \text{annual_revenue} < 500K$; $\text{CCS} \land \text{BPECS} \land \text{annual_revenue} < 500K$ | $\text{PreCheck}$ |
| $S$  | Detailed Check                | $\text{PCS}$                                   | $\text{DCS; DCU}$    |
floats), and Boolean connectives. The condition formulas may reference stage, milestone, and data attributes. If in sentry \( e \land \varphi \) the condition \( \varphi \) is equivalent to True, then \( \varphi \) is omitted so the sentry becomes \( e \).

**Example 3.2.** Table 1 shows some of the plus and minus sentries of the GSM schema of Example 3.1. In this example, each stage and each milestone has only one plus sentry, except stage Business Performance Evaluation Check and milestone PCS, each of which has two plus sentries. Each stage and each milestone has two minus sentries, except PCS.

Having introduced GSM schemas, we now define them formally.

**Definition 1.** A GSM schema is a tuple \( \Gamma = (\mathcal{A}, \mathcal{E}, \preceq, \mathcal{R}) \) where:

- \( \mathcal{A} = \mathcal{A}_0 \cup \mathcal{A}_S \cup \mathcal{A}_M \) is a finite set of attributes, where
  - \( \mathcal{A}_0 \) is a finite set of data attributes;
  - \( \mathcal{A}_S \) is a finite set of stage attributes;
  - \( \mathcal{A}_M \) is a finite set of milestone attributes.
- \( \mathcal{E} = \{ E : n \text{ is an event name} \} \cup \{ C : S \in \mathcal{A}_S \land S \text{ is atomic} \} \) is the set of external events, consisting of named external events and stage completion events. Atomic stages are defined below.
- \( \preceq \subseteq (\mathcal{A}_S \times \mathcal{A}_M) \times \mathcal{A}_S \) is a partial order on stages and milestones. If \( a_1 \preceq a_2 \) then \( a_1 \) is a descendant of stage \( a_2 \), depicted by nesting \( a_1 \) inside stage \( a_2 \). Milestones cannot have stage or milestone descendants. We require that \( \preceq \) induces a forest, i.e., if \( a_1 \preceq a_2 \) and \( a_1 \preceq a_3 \), then \( a_2 \preceq a_3 \) or \( a_3 \preceq a_2 \). If stage \( S \) has descendants, \( S \) is called composite, otherwise \( S \) is called atomic.
- \( \mathcal{R} = \mathcal{R}^+ \cup \mathcal{R}^- \) are functions assigning to each stage and each milestone non-empty sets of sentries. Given a stage or milestone \( a \in \mathcal{A}_S \cup \mathcal{A}_M \), \( \mathcal{R}^+(a) \) is the set of plus sentries that open stage \( a \) if \( a \in \mathcal{A}_S \), or achieve milestone \( a \) if \( a \in \mathcal{A}_M \), while \( \mathcal{R}^-(a) \) is the set of minus sentries that close stage \( a \) if \( a \in \mathcal{A}_S \), or invalidate milestone \( a \) if \( a \in \mathcal{A}_M \). As stated above, the sentries are boolean conditions. A sentry \( \varphi \in \mathcal{R}^+(a) \) maps to event-condition-action (ECA) rule “\( \varphi \text{ then } a \)”, while each sentry \( \varphi \in \mathcal{R}^-(a) \) maps to ECA rule “\( \varphi \text{ then } -a \)”, where the actions are setting \( a \) to True or False, respectively.

If milestone \( m \) is attached to stage \( S \), for instance CCS to stage Credit Check in Fig. 1, this denotes that \( m \) is achieved by performing the work in \( S \). Then, \( S \) owns \( m \). This is expressed in terms of sentries: an achieving (plus) sentry of \( m \) is triggered by the completion event \( C : S \) of \( S \), and each minus sentry of \( S \) is triggered if \( m \) gets achieved. Conversely, each minus sentry of \( m \) is triggered if \( S \) becomes open.

**Definition 2.** Let \( S \) be a stage and \( m \) be a milestone. If a plus sentry of \( m \) has trigger event \( C : S \), and if each minus sentry of \( S \) equals either a plus sentry of \( m \) or “\(+m\)” or “if \( m \)”, and each minus sentry of \( m \) equals either a plus sentry of \( S \) or “\(+S\)” or “if \( S \)”, then \( S \) owns \( m \).

Table 2 shows three invariants that GSM schemas must satisfy. The first invariant reflects that stage \( S \) denotes work that needs to be performed to achieve \( m \). If the work is done (\( m \) is achieved) then \( S \) should be closed. Conversely, if \( S \) is open, so the work is being done, then \( m \) should be false. The second invariant ensures that a stage \( S \) is open only if each stage \( S' \) that contains \( S \) is also open. The third invariant states that milestones owned by the same stage denote alternative outcomes. For instance, in BCA (Fig. 1) the outcome of stage Credit Check is either successful (milestone CCS) or unsuccessful (milestone CCU). A state of a GSM schema is called a snapshot.

**Definition 3.** For a GSM schema \( \Gamma = (\mathcal{A}, \mathcal{E}, \preceq, \mathcal{R}) \) a snapshot \( \sigma \) is an assignment of values to attributes, i.e., a mapping from \( \mathcal{A} \) into values of appropriate type. For stage and milestone attributes, the only permitted values are False and True. Each snapshot must satisfy the GSM invariants. Initially, all stage and milestone attributes are False.
Let \( \Gamma = (\mathcal{A}, \mathcal{E}, \preceq, \mathcal{R}) \) be a GSM schema. Let \( \psi = \epsilon \land \varphi \) be a sentry for a status attribute \( a \in \mathcal{A}_S \cup \mathcal{A}_m \), so \( \psi \in \mathcal{R}(a) \). Given a snapshot \( \sigma \) of \( \Gamma \) and an event \( E \) that occurs, where \( E \) has prefix \( C : \) or \( E : \), we write \( \sigma \models \psi \) if \( E = E \) and \( \sigma \models \varphi \), so \( \sigma \) satisfies the formula \( \varphi \) of sentry \( \psi \). In other words, \( \sigma \models \psi \) means that sentry \( \psi \) is enabled in snapshot \( \sigma \) if \( E \) occurs.

If in a snapshot a stage completion event occurs, then first the data values carried in the payload of the event are incorporated in the snapshot by assigning them to the relevant data attributes in \( \mathcal{A} \). Next, several of the event-condition-action rules derived from \( \mathcal{R} \) can be enabled, i.e., their sentries evaluate to true. They are fired one by one, each rule \( \varphi \in \mathcal{R}(a) \) changing the value of the status attribute \( a \). As a result, some additional rules may become enabled and fire. For instance, consider a snapshot in which the stage Credit Check is open and the milestones BPECS and AMCS have been achieved. If completion event \( C : \) Credit Check with payload \( \langle \text{rating}, 9 \rangle \) occurs, then the data attribute rating is updated with this new value first. Next, the achieving sentry for CCS evaluates to true, so the milestone CCS gets achieved, which in turn leads to achieving the milestone PCS, which in turn opens the stage Detailed Check. In the resulting snapshot, no sentries are satisfied and the evaluation of sentries stops and the system waits for the next external event, which is the stage completion event of \( C : \) Detailed Check. Thus, the evaluation of sentries has a cascading effect, which is called a Business step, or B-step for short [6].

To properly compute the B-steps, dependency graphs are used [6]. For each external event \( E \) with prefix \( C : \) or \( E : \), a dependency graph \( DG(E) \) is created whose nodes are \( V = \{E\} \cup \{+S, -S | S \in \mathcal{A}_S\} \cup \{-m, -m | m \in \mathcal{A}_m\} \). For \( a_1, a_2 \in \mathcal{A}_S \cup \mathcal{A}_m \), edges \((+a_1, +a_2), (-a_1, +a_2), (-a_1, -a_2)\) exist if there is a plus sentry of \( a_2 \) that references \( a_1 \); edges \((+a_1, -a_2), (-a_1, -a_2)\) exist if there is a minus sentry of \( a_2 \) that references \( a_1 \); edge \((E, +a_1)\) exists if there is a plus sentry of \( a_1 \) whose trigger event is \( E \); edge \((E, -a_1)\) exists if there is a minus sentry of \( a_1 \) whose trigger event is \( E \).

If the external event \( E \) occurs and there is an edge \((v_1, v_2)\), then the sentries of the status attribute of \( v_1 \) must be evaluated before those of \( v_2 \), in order to ensure that each change in status attribute is justified when examining the starting and ending snapshots of the B-step [6].

**Example 3.3.** Fig. 2 shows the dependency graph \( DG(C : \) CheckCredit\). Each edge indicates a dependency due to a plus or minus sentry. For instance, a minus sentry of Check Credit references CCS, so there are dependencies \((+CCS, -CheckCredit)\) and \((-CCS, -CheckCredit)\). This means that the sentries of CCS should be evaluated before the minus sentries of Check Credit are evaluated.

### 3.2. Fuzzy Modeling

In real-world decision making, obtaining precise parameters for decision models may hardly be possible or precise values for some threshold may not be meaningful.

**Linguistic Modeling of Variables:** A classical example is assessing whether the weather is hot, cold or medium, based on the temperature. The same value of temperature can result in different assessments for different countries. For instance a temperature of 30° is considered hot in the Netherlands, but medium in Turkey.

Fuzzy logic has been proposed by Zadeh [35] to capture this uncertainty [24]. Fuzzy logic uses linguistic variables, i.e., variables whose values are different linguistic labels/terms that are defined in terms of a base variable. For a base variable \( x \), the corresponding linguistic variable is usually denoted by \( \tilde{x} \). However, for simplicity in our notation, we use the same name as the base variable as long as the distinction is clear from the context.

**Definition 4.** Given a linguistic variable \( x \), we define a linguistic label \( i \) whose membership function \( \mu_i : \text{dom}(x) \rightarrow [0, 1] \) maps each value of \( x \) to the degree in which term (label) \( i \) holds. A membership of zero means that \( i \) is not a member at all, and a membership of one that \( i \) is fully a member.

---

**Table 2**

| GSM-1   | If stage \( S \) owns milestone \( m \), then \( S \) and \( m \) cannot both be true. |
|---------|----------------------------------------------------------------------------------|
| GSM-2   | If stage \( S \) is descendant of \( S' \) and \( S' \) is false, then \( S \) is false too. |
| GSM-3   | If \( S \) owns distinct milestones \( m \) and \( m' \), and \( m \) is true, then \( m' \) is false. |

**Fig. 2.** Dependency graph of \( C : \) Check Credit.
Fig. 3 shows the membership functions for the linguistic terms hot, medium, cold, for the linguistic variable Temperature ($T$). These definitions apply for countries like the Netherlands. For a country like Turkey, the membership functions would be different. For instance, $\mu_{T}^{\text{hot/TR}}(30) = 0.5 < \mu_{T}^{\text{hot/NL}}(30) = 1$, where TR represents Turkey and NL the Netherlands.

In fuzzy logic, logical statements contain fuzzy operands, indicating that uncertain or poor information is available. The two states of the satisfaction of a logical statement, namely True (1) and False (0), are extended to take value in the interval $[0, 1]$. Fuzzy aggregation operators have been defined, which are the fuzzy counterparts to logical operands like AND and OR.

Fuzzy Aggregation: The logical aggregation by AND and OR operators are generalized in fuzzy logic by using t-norm (conjunctive) and t-conorm (disjunctive) operators, respectively [16,24]. Other aggregation operators have also been defined, such as the compensative, non-compensative, and the weighted operators.

**Definition 5.** Aggregation operations on fuzzy sets are operations by which several fuzzy sets are combined to produce a single fuzzy set. Weight factors can be defined for the included individual components.

For example, the convenience for sporting is an aggregated decision including two criteria, temperature and day light. The appropriateness of temperature and day light conditions for sporting may linguistically be expressed as

Temperature is medium and Daylight is medium.

Quantifying this expression is done by defining a fuzzy set for each label of the involved terms in the expression such that the degree of satisfaction of the labels is quantified. The label “medium” may, for example, be quantified by using the fuzzy set as shown in Fig. 3. The term “and” requires using an aggregation operator of t-norm type. Finally, the aforementioned expression is formulated as

$$\text{Sport} = \mu_{T}^{\text{med}}(T) \land \mu_{DL}^{\text{med}}(DL)$$

where $T$ and $DL$ denote the temperature and the day light respectively, and the symbol $\land$ denotes a t-norm, for which usually the minimum or the product operators are used.

By definition, any conjunctive (disjunctive) operator should come up with one value that is not greater (smaller) than any individual value in the aggregation. Therefore, the highest (smallest) value is delivered in a conjunctive (disjunctive) aggregation, by the minimum (maximum) operator. Besides these basic optimistic operators for aggregation, other operators are proposed in the literature [16,24]. In this paper, we use the product operator for the conjunctive aggregation in our business cases. The influence of the aggregation operators on the decisions is beyond the scope of our work.

4. **Fuzzifying GSM Schemas**

In this section we introduce the design principles underlying fuzzy GSM schemas. The next section defines fuzzy GSM schemas. We explain for each of the key components of a GSM schema (data attributes, stages, milestones, and sentries) how we fuzzify that component and the benefits of doing so.

4.1. **Fuzzifying Data Attributes**

Artifact-centric process models are abstractions of real-world procedures that include uncertainty due to several reasons like insufficient sampling size, involvement of human perception and so on. Thus, in classic GSM schemas, the values of variables, i.e. data attributes, can actually carry considerable imprecise information. These data attributes can therefore better be modeled as linguistic variables using fuzzy uncertainty modeling as explained in Section 3.2.
Some numerical attributes attached in the classical GSM schemas may be a rough estimation of certain concepts. For example, the threshold parameter 300 in the check of employee_count variable in Table 1 represents the size of a company. Now, 300 is a somewhat arbitrary number and specified by a human expert as a threshold value. However, a razor sharp classification of a company to be small or big may not be appropriate. For the sake of extracting more knowledge from an expert, the following question can be asked: “For which employee numbers is a company considered to be big to which extent?”

Benefit of using fuzzy models of data attributes is that uncertainty about the interpretation of business terms can be expressed. For instance, the stage Business Performance Evaluation Check is opened if the company size is big. In the GSM schema (Fig. 1), this test is operationalized by checking whether the employee count exceeds 300, which is somewhat arbitrary. The related sentry of the fuzzy GSM schema in Fig. 6 defines the notion of big company with a fuzzy set, allowing to express to which degree a company is big.

4.2. Fuzzifying Stages

We fuzzify GSM stages by allowing them to complete gradually, once they are opened. While a stage is open, a completion event signals a partial completion of the stage, along with a set of attributes and their values that have been written so far in the stage. We do not allow gradual opening of stages: once a stage is opened, it is fully active.

A fuzzy stage S represents the degree, between 0 and 1, to which S has completed, i.e., the completion level of S. If S is 0, then S is closed. If S is between 0 and 1, then S is open. The initial starting value of open stage S is an infinitesimally small number ε. Stage S can have finitely many completion levels between 0 and 1. If S is 1, then S has fully completed, i.e., it is closed after having been open. In Fig. 4, we define five labels for the linguistic variable S that represents the completion of the fuzzy stage: open, somewhat completed, partially completed, almost completed, and fully completed. Note that in Fig. 4 ε is small enough such that the intervals [0, ε] and [1 − ε, 1] can be considered as points.

Opening of an atomic stage is done in a crisp way, that is, an atomic stage is either open or closed. Completing a stage is done in a gradual way, that is, a partially opened stage S can have zero or more ‘preliminary’ completion events that signify partial completions of S, followed by a final completion event that sets the completion level to 1, i.e. the stage is closed. Here, we modify slightly the meaning of the completion event in the classical case, where it denotes full completion only. In our case, a completion event implies a new completion level, not necessarily full completion. For instance, when stage Credit Check is open, the value for data attribute rating can be communicated with a preliminary completion event, while stage Credit Check remains open and continues to write other data attributes. The different completion events C for S may carry new values for some of the data attributes in their payloads. These new values are crisp. Typically, if S is open, S increases as work progresses, but S may also decrease, meaning that some work is redone.

The completion level of a composite stage S is defined by a function that takes as input the completion levels of its inner stages. Some inner stages are only opened if the relevant data attributes are in a certain range, for instance Business Performance Evaluation Check and Addressable Market Check in stage PreCheck in Fig. 1. Consequently, the function definition can be complex. We provide some examples in Section 5.
A closed fuzzy stage $S$, so either $S = 0$ (not started) or $S = 1$ (completed), gets opened (again) as follows. For each guard of $S$, its fuzzy truth value is computed, which is between 0 and 1. If one of the guards exceeds a predefined threshold, called the activation degree, then $S$ is opened by assigning $S$ an infinitesimally small number $\epsilon$ greater than 0. The stage is closed and set to 0 if none of its guards exceeds the activation degree.

If $S \leq S'$ and both $S$ and $S'$ are 0, then the activation degree of $S$ must be higher than or equal to the activation degree of $S'$ to ensure invariant GSM-2. In classical GSM schemas, this is ensured by the convention that each guard of $S$ has the additional conjunct “$\wedge S'$”, so if $S$ is opened, then $S'$ must be open [10]. Using the open label of linguistic variable $S$ in conjunction with the rest of the sentry, see Fig. 4, gives a similar result.

A fuzzy stage $S$ has no explicit terminating (minus) sentries. Stage $S$ is closed if the completion level of $S$ has reached 1, fully completed in Fig. 4. This is done via the final completion event of $S$. As for classical GSM schemas, we assume that for each open atomic stage the environment eventually generates a final completion event. Stage $S$ is closed too if all its guards fall below activation degree. This resembles an interrupt of $S$.

Benefit of fuzzy stages is that the different completion levels of a fuzzy stage can be referred to in the guard sentries of other stages. For instance, we can specify in a sentry that stage $S$ has almost completed, or is halfway completed. Having user input regarding the progress of a stage makes fuzzy GSM schemas more sensitive to status changes than classical GSM schemas and allows to express fine-grained ordering dependencies. Similar concerns are also addressed by generalized precedence relations [23], which have been used in project management.

4.3. Fuzzifying Milestones

A fuzzy milestone generalizes a classical milestone by representing the degree to which the corresponding business objective has been achieved. If milestone $m$ is attached to stage $S$, then the gradual completion of $S$ may change the truth value of $m$. After the final completion of $S$, the truth value of $m$ can be 1 (which means fully achieved) but not necessarily. For instance, after stage Credit Check is closed, the value for milestone CCS can be 0.9, indicating that the credit check was successful to a large extent.

A fuzzy milestone allows to express multiple, exclusive business conditions. For instance, the outcome of credit check is either successful or unsuccessful, denoted with two exclusive milestones CCS and CCU in the classical GSM schema in Fig. 1. In the fuzzy GSM schema $BCA_{fuzzy}$ (Fig. 6), there is a single fuzzy milestone CCS. If its value is close to 1, it denotes success. If its value is close to 0, it denotes fail, which corresponds to CCU in Fig. 1. Naturally, fuzzy GSM schemas do allow to include a fuzzy milestone CCU in this case, and multiple milestones attached to the same stage in general.

It is a design choice how to model achievement at milestones. One way is to define one membership function for the complete milestone. We prefer to define three membership functions to represent intermediate achievement levels linguistically. Hence, fuzzy milestones are referenced in combination with linguistic labels such as fully, almost, and somewhat, allowing sentries like “$m$ has been fully achieved”, “$m$ has been almost achieved”, “$m$ has been achieved to some extent”, see Fig. 5. Note that using linguistic labels with fuzzy milestones in sentries is different from using fuzzy milestones directly. For instance, if $m$ has value 0.7, then “$m$ has almost been achieved” can have value 0.9.

The value for a fuzzy milestone is the maximum truth value of its sentries. Consequently, fuzzy milestones do not need the notion of invalidating (minus) sentries. For instance, the fuzzy sentry for CCS in Fig. 6 is “CC has fully completed $\wedge$ rating is high”. If CCS is fully achieved and next stage CC (Credit Check) is reopened, then the truth value for the fuzzy sentry becomes small, so CCS has a small value close to zero too.

![Fig. 5. Labels for the achievement of fuzzy milestone $m$.](image-url)
Benefit of fuzzy milestones is that they allow to express uncertainty about the achievement of business conditions, which is natural in complex business environments. Further, there is no need to have multiple exclusive milestones for the same stage: different outcomes can be represented by different valuations of the same milestone, rather than by using different milestones.

4.4. Fuzzifying Invariants

Fuzzifying stages and milestones impact the GSM invariants, i.e., they need to be rephrased. Table 3 shows the invariants for fuzzy GSM schemas.

Invariant GSM-1 is too strong for fuzzy GSM schemas, since a completed stage \( S \) is false in classical GSM schemas and has value 1 in fuzzy GSM schemas. We therefore formulate another invariant FGSM-1 that states that a non-closed fuzzy stage \( S \), so \( S < 1 \), cannot have a (completely) true fuzzy milestone, or equivalently, if the milestone is completely true, the stage has fully completed. This invariant is similar in spirit to what GSM-1 expresses for classical GSM schemas: an open stage cannot own an achieved milestone.

Note that the invariant allows that a gradual completion of a fuzzy stage increases the truth value of a fuzzy milestone, which reflects that progressing in the stage is progressing towards the related business objective, as denoted by the milestone.

Since a fuzzy stage can be closed in two different ways (values 0 and 1), invariant GSM-2 translates into two subconstraints FGSM-2. If a fuzzy stage \( S' \) is closed and not yet been open, so its value is 0, and \( S' \) has a descendant stage \( S \), then that stage \( S \) must be closed and not yet been open too. If \( S' \) is closed because it has been completed, its descendant stage \( S \) is either 0, if \( S \) was not needed to complete \( S' \), or 1, if \( S \) has been fully completed.

The first subconstraint is enforced by letting conjunct "\(^\land S\)" be part of every guard of \( S \). If \( S' \) is 0, then the truth value of each guard of \( S \) is 0. The second subconstraint needs to be enforced by the definition of the completion level of the composite stage \( S' \).

Invariant GSM-3 translates into the fuzzy invariant FGSM-3 that the sum of the milestones owned by the same stage \( S \) does not exceed 1. However, if a stage owns a single milestone that represents different outcomes, invariant GSM-3 is too strong. Therefore FGSM-3 only applies to stages that own multiple milestones.

4.5. Fuzzifying Sentries

In the classical GSM schemas, sentries may contain logical AND and OR operators in their condition formulas. In our fuzzy modeling of GSM schema, fuzzy sentries are evaluated using fuzzy aggregations, whose basics are introduced in Section 3.2.

A crisp sentry can reference stages, milestones or data attributes in its condition formula. A fuzzy sentry can use linguistic labels of fuzzy stages or fuzzy milestones or fuzzy data attributes, which need to be aggregated when evaluating the fuzzy sentry. For example, Table 1 contains crisp sentry "CCS \( \land \) BPECS \( \land \) AMCS" of milestone PCS, where each term in the expression is Boolean. The corresponding fuzzy sentry has the following linguistic form:

\[
CCS \text{ is almost achieved } \land \text{ BPECS is partially achieved } \land \text{ AMCS is almost achieved.}
\]

| FGSM-1 | Given \( S \) owns \( m \): if \( S < 1 \), then \( m < 1 \). |
| FGSM-2 | Given \( S \) is a descendant of \( S' \), i.e., \( S \leq S' \): if \( S' \) is 0, then \( S \) is 0; if \( S' \) is 1, then \( S \) is 0 or 1. |
| FGSM-3 | Given multiple milestones \( \{m_1, \ldots, m_n\} \) owned by \( S \): \( \sum_{i=1}^{n} m_i \leq 1 \). |
where the terms like “almost achieved” are represented by fuzzy sets.

Fuzzy guards for our business case example are listed in Table 4. A fuzzy guard evaluates to a value in the interval \([0, 1]\). If the value exceeds the activation degree of the stage, the stage is opened and the stage is assigned the value \(c\), as explained in Section 4.2. If the value is below the activation degree, the stage is closed and assigned value 0.

Fuzzy sentries of milestones for our business case example are listed in Table 5. Each milestone is assigned the maximum of the truth values of each sentry.

The benefit of fuzzy sentries is twofold. First, they allow eager execution of stages: it is no longer needed for a guard sentry to be fully true, true enough is already sufficient. For example, a truth level 0.87 may be enough to open a stage if the activation degree of that stage is defined as 0.85. Second, they are convenient tools for incorporating expert knowledge into a business model by capturing the uncertainty and imprecision of real-life conditions.

### 4.5.1. Evaluation of fuzzy sentries

So far, we have presented how sentries can fuzzified. We now sketch a potential solution how the evaluation of fuzzy sentries can be done at run time. In regular GSM semantics sentries are evaluated to perform a B-step in response to an external event (see Section 3.1). The evaluation order of sentries is according to the dependency graph of the event. For fuzzy GSM schemas, fuzzy sentries also need to be evaluated one by one according to the order of the dependency graphs. However, fuzzy aggregation is needed to determine the value for the status attributes, i.e., fuzzy stages and milestones. Section 5.5 explains in more detail how fuzzy GSM schemas are executed.

### 5. Fuzzy GSM Schemas

In the previous section, we introduced the design choices of fuzzy GSM schemas. In this section, we introduce fuzzy GSM schemas.

#### 5.1. Linguistic Variables

As explained in Section 4, the structure of the fuzzy GSM schema leads us to define three types of linguistic variables: completion level of stages, achievement level of milestones, and other linguistic variables of data attributes. The last group of linguistic variables should be obtained by consulting the expert knowledge of the specific business case. Before giving
details about each linguistic variable type, we first give a formal definition of a linguistic variable and its membership function for a label.

**Definition 6.** Let a data attribute \( a \in \mathcal{A} \) with its domain \( \text{dom}(a) \) be defined as a linguistic variable. Let \( i \) be the label of the linguistic variable \( a \) and \( \mu^i : \text{dom}(a) \to [0, 1] \) denote its membership function with \( \mu^i(v) \) being the membership to \( i \) of the value \( v \in \text{dom}(a) \).

To show how this affects the possible expressions used in sentries, consider the parameter \( \text{employee\_count} \) in Table 1. As explained in Section 4, this parameter is the quantification of the company size, and we choose \( \text{company size} \) as the corresponding linguistic variable. Then we have the following transformation from a numerical check into a fuzzy statement:

“\( \text{employee\_count} \geq 300 \)” becomes “\( \text{company size is big} \),”

where the truth value of the first formula corresponds to a degree of membership of a fuzzy set. The membership function in part (b) of Fig. 7 is used to find the degree of a company to be considered as big instead of having a knife-cut threshold value 300.

5.2. Fuzzy Sentries

In the classical GSM schemas, sentries may contain logical AND operators in their condition formula. In our fuzzy modeling of GSM schemas, condition formulas are computed using fuzzy aggregations whose basics are introduced in Section 3.2. For example, as seen in Table 5, we have the following transformation for the sentry of CCS:

**Classical** : \( \text{C} : \text{CreditCheck} \land \text{rating} \geq 8 \)

**Fuzzy** : \( \text{CC is almost completed} \land \text{rating is high} \)

where \( \text{rating} \) is modeled as a linguistic variable with the linguistic label “\( \text{high} \)” that could be interpreted as “being approximately greater than or equal to 8”.

Fuzzy aggregation is performed by using conjunctive and disjunctive operators, as mentioned in Section 3.2. Then, the condition formula of this fuzzy sentry is evaluated as

\[ \mu^a(\text{CC}) \mu^{\text{high}}(\text{rating}) \]

where the product operator is used for the conjunctive fuzzy aggregation.

5.3. Fuzzy Stages

5.3.1. Stage openings

Given a stage \( S \in \mathcal{S} \), a fuzzy aggregation is used to obtain the maximum truth value of the guards of \( S \), called the activation level. The activation level is computed as \( \mathcal{L}^S = \max_{i \in [1, n(S) \setminus \emptyset]} \{ \tilde{\psi}_i \} \) where \( \tilde{\psi}_i \) is the aggregated result of the \( i^{th} \) fuzzy guard of stage \( S \). A threshold checking is needed to decide whether stage \( S \) becomes open, i.e. it changes from zero to a positive value. Assuming for stage \( S \) an activation degree \( \alpha_S \in [0, 1] \), we define a stage status function that assigns to \( S \) an infinitesimally small value \( \epsilon \) if one of the fuzzy guards of \( S \) reaches the activation degree \( \alpha_S \):

![Fig. 7. Binary and fuzzy conditions for employee count check.](image-url)
fuzzy sentries is " and can trigger new Business steps. As another example, fuzzy milestone

Fuzzy aggregation of the sentry, and (3) defuzzify the aggregation output for the relevant decision of the sentry, e.g., a stage starts if the sentry output exceeds the threshold value. The functions used in these steps have been defined above.

The opening of stage \( S \) is almost achieved if the sentry output exceeds the threshold value. The functions used in these steps have been defined above.

Fuzzy stage \( S \) gets assigned a completion level from \( \mathcal{S} \). In atomic stages, the intermediate completion levels, i.e., \( \mathcal{S} \setminus \{0, \epsilon\} \), are reported by system users by generating completion events. For instance, the atomic stage GFD of our business example in Fig. 6 may have the completion values \( \mathcal{S}_{\text{GFD}} = \{0, \epsilon, 1/3, 2/3, 1\} \) meaning that the completion occurs in increments of one third.

Composite stages have completion levels depending on the involved atomic stages and their influences on the completion. As mentioned in Section 4.2, discrete membership functions of the completion of composite stages contain a subset of all possible, and feasible, states of the involved components of the stage. For example, the completion level of stage IDG may be \( 1/8 \) if GFD \( = 1/3 \), \( P = \epsilon \), and GMD \( = 0 \). The set of completion levels of composite stage IDG, in Fig. 6, may be defined as

\[
\mathcal{S}_{\text{IDG}} = \{0, \epsilon, 1/8, 1/6, 1/3, 2/3, 1\}
\]

for the aforementioned \( \mathcal{S}_{\text{GFD}} \) and sets of completion levels \( \mathcal{S}_P = \mathcal{S}_{\text{GMD}} = \{0, \epsilon, 1\} \).

A fuzzy stage completion has five labels as shown in Fig. 4. There are two special labels, “opened” and “fully completed” that indicate boundary completion states of a stage. Intermediate completion levels are expressed by labels “somewhat”, “partially”, and “almost”. The linguistic expression “Stage \( S \) is almost completed” is quantified by the value \( \mu^f(S) \) using the corresponding membership function in Fig. 4.

5.4. Fuzzy Milestones

Fuzzy milestones represent the level of achievement of the corresponding business objectives. Given a milestone \( m \in \mathcal{M} \), its achievement level is the maximum value of the aggregations of its fuzzy sentries, denoted \( \max_{\mathcal{M}} \{\psi_i\} \). For example, the achievement level of milestone IDGS in Table 5 is obtained by evaluating the following aggregation

\[
\mu^f(\text{GFDC})\mu^f(\text{ES})\mu^f(\text{GAMD}),
\]

where \( f \) denote fully and \( a \) denote almost, respectively.

Any change of the term values in the aggregation leads to a recalculation of the achievement level of the milestone, which can trigger new Business steps. As another example, fuzzy milestone PCS in Table 5 has four fuzzy sentries. One of these fuzzy sentries is “CCS is almost achieved \& BPECS is partially achieved \& AMCS is almost achieved”, whose aggregation is

\[
\mu^f(\text{CCS})\mu^f(\text{BPECS})\mu^f(\text{AMCS}).
\]

The achievement level of milestone PCS is denoted as \( \max_{\psi_1, \ldots, \psi_4} \{\psi_i\} \), where \( \psi_1, \ldots, \psi_4 \) denote the aggregated results of the four fuzzy sentries listed in Table 5 for fuzzy milestone PCS.

5.5. Incorporating Fuzzy Sentries in the GSM Execution

So far, we have presented how sentries can fuzzified. We now sketch a potential solution how the evaluation of fuzzy sentries can be done at run time (cf. Fig. 8). The only difference with the regular GSM semantics, in which sentries are evaluated to perform a B-step (see Section 3.1) is that sentries are now evaluated using fuzzy reasoning, consisting of three steps that are standard in evaluating fuzzy rules [24]: (1) fuzzify the crisp input variables referenced in the sentry, (2) calculate the fuzzy aggregation of the sentry, and (3) defuzzify the aggregation output for the relevant decision of the sentry, e.g., a stage starts if the sentry output exceeds the threshold value. The functions used in these steps have been defined above.
6. Evaluation

In this section we provide two examples of fuzzy GSM models to show that our proposed GSM structure can capture the uncertainty encountered in real-life applications.

6.1. Airport ground operations

As our first example, we show how fuzzy GSM schemas are applied to a real-life process from the airport ground operations management. Airport ground operations are faced with high uncertainty resulting in delays which requires dynamic decision making. Using fuzzy GSM schemas allows specification of fine-grained executions that cannot be specified easily with classical process models. Fig. 9 shows the GSM schema of the ground operations regarding to passenger boarding and departure preparation of the airplane. The actions for a departing flight start upon receiving the expected arrival time of the airplane that is assigned to the flight.

The evaluation example in Fig. 9 includes three composite stages. The first one concerns boarding. In the second one, the atomic stage executions are sequential, i.e. one requires completion of the previous one in the order Precheck documents, Preboarding, and Embarking of passengers. In the third composite stage Departure preparation the precedence relations are not as precise as in the first one. Especially, the stand-alone milestone Waiting Time Milestone, shortly WTM, has a decision mechanism that outputs the amount of waiting time for closing the gate. This decision of waiting time amount ($W_t$) depends on several parameters like number of passengers (NrPass) taking the flight, the ones who did not show up (MissingPass), and the expected delay time ($ExpDly$). The corresponding membership functions are illustrated in Fig. 10. Table 8 lists the fuzzy sentries of WTM and Fig. 11 shows the membership function to map membership values obtained from the milestone to a crisp number in minutes. The gate is closed after waiting $W_t$ long time, which is one of the conditions checked to start the atomic stage Baggage removal and final checks. The other stand-alone milestone Gate Opening Milestone, abbreviated GOM, is used to make the decision of opening the gate when the airplane is highly ready and preboarding is fully completed.

The data attributes of the business model AGO are self-explaining like $GateOpened$, $GateClosed$, $CounterClosed$, $AirplaneLanded$, $DispatcherOffBoard$. Moreover, the linguistic variable $AirPlaneReadiness$ is input information that is received from the teams that are responsible for deboarding, cleaning, and refueling of the airplane.

We note that determining the opening time of a gate and the waiting time to close the gate are multi-criteria decisions. Having fuzzy sentries (see Table 6), hence using fuzzy sets and aggregations, to determine these continuous values is the
state-of-the-art way. Table 7 shows the progress of one flight instance where the extra waiting time decision is made. We see that the expected delay is found for every gradual completion step; the step where the milestone PBS is fully achieved is shown in bold. At that moment, the aggregation value of the WTM milestone (see Table 8) has a fuzzy membership value of 0.75 that is converted to 13 minutes by using the membership function in Fig. 11. It is also seen that in this waiting time 4 passengers have arrived. The ultimate goal of the waiting time is to let all passengers arrive, but this strongly depends on particular cases. In practice, having extra waiting time allows to get late-arriving passengers on board, but there may be ones who changed their minds for taking the flight (no shows).

In real-life practice of the airport ground operations, efficiency in turnaround times comes from using generalized precedence relations, where gradual completions are taken into account, rather than the conventional precedence relations of start-to-finish type. This evaluation example shows that fuzzy GSM schemas allow to express such generalized precedence relations. The example also shows that the proposed fuzzy GSM schemas are suitable to model projects involving generalized activity precedence relations.

6.2. Healthcare case: screening and intake of Unstable Angina patients

As our second example, we show how fuzzy GSM schemas are applied to a real-world process from the healthcare domain. Healthcare processes are often very complex, spanning different departments and comprising different subprocesses. Using fuzzy GSM schemas allows specification of fine-grained executions that cannot be specified easily with classical process models.

---

**Table 6**

| Stage                                      | Sentry                                                                 |
|-------------------------------------------|-----------------------------------------------------------------------|
| Boarding Preparation                      | true                                                                  |
| Gate Assignment                           | Boarding Preparation is opened                                        |
| Gate Preparation                          | GAC is fully achieved                                                 |
| Passenger Boarding                        | BPS is fully achieved                                                 |
| Precheck documents                        | Passenger Boarding is opened                                          |
| Preboarding                               | Precheck documents is fully completed \( \land \) GOM is fully achieved |
| Embarking of passengers                   | Preboarding is almost completed                                       |
| Load baggage                              | CounterClosed \( \land \) AirplaneLanded                               |
| Check cabin baggage                       | Load Bagage is fully completed \( \land \) NrPass is high             |
| Bagage removal & Final checks             | GateClosed \( \land \) MissingPass is high                           |

---

Fig. 10. Fuzzy membership function for Large Number of Passengers and Large Expected delay.

Fig. 11. Fuzzy membership function for waiting time.
The evaluation example is based on the standardized care trajectory of Unstable Angina patients, hospitalized because of Acute Coronary Syndrome (ACS) in a Dutch hospital [31]. The current process is designed in the classical way in BPMN, forcing composite activities to be completely finished before the next activities can start. Also, different possible outcomes and their required followup actions need to be specified explicitly. Here we evaluate the use of fuzzy GSM schemas for modeling this process. Due to space limitations, we show here model fragments.

Fig. 12 shows a GSM schema fragment of the screening and intake subprocesses that belong to this care process. Fig. 13 gives a detailed view of the screening subprocess. Table 9 shows fuzzified guards for some stages, including stage Discussion risk + consent. The guards allow that the stage activation depends on the level of completion of compound stage Check and the outcome of the ECG Interpretation. Achievement of milestone EI is based on a linguistic variable ECG Improvement which can get linguistic values high (i.e. significant), mod or low, which indicate how the latest ECG relates to the previous ECG. High achievement of EI depends on the success of ECG Improvement. This way, fine-grained precedence relations can be specified that cannot be expressed easily in classical BPMN models.

To illustrate this, Table 10 shows a sample execution scenario for the GSM schema in Fig. 13 considering a classical (crisp) and a fuzzy setting for a case in which the Euroscore of a patient is above 10. The Euroscore of a patient is based on data that is gathered in compound stage Check. In the fuzzy setting, the stages following Check, including final stage Determine risk + consent, are started much earlier than in the crisp setting. Reason is that in the Check stage data is gathered that can already be used in subsequent stages before Check has completed. In particular, after stage Diabetes Check has completed (minute 17), the Euroscore is already high, which enables in the fuzzy setting that stage Consult CTC surgeon starts at minute 20; in the crisp setting, the Euroscore is only computed after completing stage Check at minute 30. This way, the throughput time in the fuzzy setting (32 min) is significantly less than in the crisp setting (45 min).

Table 11 shows a few fuzzy milestones and their fuzzy sentries for the composite stage Screening in Fig. 13. Each truth value of a fuzzy sentry specifies a different success level for its milestone. Thus, a single milestone allows for a succinct representation of different outcomes of a stage. In the BPMN model, these different outcomes and their followup actions need to be explicitly specified with different paths in the process model. In the fuzzy GSM schema, a single milestone EI, ESU or...
Table 9
Some stages and sentries for Fig. 12 and 13.

| Stage                                      | Sentry                                      |
|--------------------------------------------|---------------------------------------------|
| Discussion risk + consent                  | Check has almost completed                  |
| Discussion risk + consent                  | Check has partially completed \( \land \) Ei is low |
| Consult CTC Surgeon                       | Euroscore is high                           |
| Consult CTC Surgeon                       | Medical objection                           |
| Start intake anesthetist                   | Screening has almost completed              |

Table 10
Execution scenario in the crisp and the fuzzy setting for a case that the Euroscore is high; numbers are minutes; the last minute in each scenario is marked in bold.

| Stage                                      | Activation time (classical GSM) | End time (classical GSM) | Activation time (fuzzy GSM) | End time (fuzzy GSM) |
|--------------------------------------------|--------------------------------|--------------------------|-----------------------------|----------------------|
| Data Gathering                             | 0                              | 3                        | 0                           | 3                    |
| Intake, history and information providing  | 3                              | 5                        | 3                           | 5                    |
| Check                                      | 5                              | 30                       | 5                           | 30                   |
| Diabetes Check                             | 7                              | 17                       | 7                           | 17                   |
| Start GLK schema                           | 7                              | 11                       | 7                           | 11                   |
| Consult ICC Internist                      | 12                             | 17                       | 12                          | 17                   |
| COPD Check                                 | 25                             | 30                       | 25                          | 30                   |
| Consult ICC pulmonologist                  | 25                             | 30                       | 25                          | 30                   |
| Physical examination and ...               | 11                             | 16                       | 11                          | 16                   |
| Interpret ECG                              | 17                             | 19                       | 17                          | 19                   |
| Update EPR                                | 30                             | 35                       | 18                          | 31                   |
| Update Euroscore                           | 35                             | 36                       | 19                          | 32                   |
| Consult CTC Surgeon                       | 36                             | 41                       | 20                          | 25                   |
| Discussion risk + consent                  | 41                             | 45                       | 26                          | 30                   |

Table 11
Some milestones and sentries for Fig. 12 and 13.

| Milestone | Full Name           | Sentry                                      |
|-----------|---------------------|---------------------------------------------|
| EI        | ECG Interpreted     | ECG improvement is high                     |
| ESU       | Euroscore Updated   | Euroscore is high                           |
| Screened  | Screened            | Checked is fully achieved \( \land \) ESU is somewhat achieved \( \land \) RCD is somewhat achieved |
Screened captures the different outcomes. For instance, in the BPMN model there are two outcomes of screening, screened (ready for next phase) or withdrawn. If the Euroscore is low and the outcome of the Discussion risk + consent stage is that the risk is low, then Screened is almost or fully achieved. Conversely, if the Euroscore is high, a CTC surgeon is consulted for a discussion, which may result in the decision to withdraw the patient. Also if RCD is high, so the outcome of the discussion risk and consent stage is that there is a high risk, a CTC surgeon may decide to withdraw the patient. These outcomes are represented here by different achievement levels of a single milestone Screened, due to different truth values for its fuzzy sentry. Thus, with a fuzzy milestone different outcomes can be represented in a succinct way.

Both examples in this section show that the notion of fuzzy stages, fuzzy milestones, and fuzzy sentries can be easily incorporated in real life business processes, allowing fine-grained specification of precedence relations and a succinct representation of different outcomes. Moreover, the fuzzy rules are much closer to human reasoning than the two-valued logic in the BPMN models.

7. Conclusion

We have proposed fuzzy GSM schemas as a technique for modeling uncertainty in declarative artifact-centric process models. Key elements are fuzzy stages, which represent work that can be completed gradually, and fuzzy milestones, which allow to express to which extent business objectives have been achieved. We showed, using real-life scenarios from airport ground operations and healthcare services, how fuzzy stages and milestones enable a fine-grained specification of precedence relations, based on different levels of completion of stages.

Fuzzy GSM schemas have several advantages over classical GSM schemas. Firstly, they are capable of capturing expert knowledge via defining rules in linguistic terms that resemble human understanding of the problem domain more closely than the rules in classical GSM schemas. Secondly, fuzzy GSM schemas are more useful for the cases where it is hard, even impossible, to collect enough information to make concrete decisions while performing the business process. The resulting uncertainty is better captured using linguistic variables in decision making, as done in fuzzy models, rather than relying on numeric variables as used in classical GSM schemas. Thirdly, fuzzy GSM schemas are more succinct than classical GSM schemas, since a single fuzzy milestone can represent many different outcomes for a single business objective, while a crisp milestone corresponds to one possible outcome.

One potential use of fuzzy GSM schemas is modeling executions of projects that are subjected to generalized precedence relations. It is pointed out in the literature that the real-life precedence relations are results of imprecise estimations and subjected to some degree of uncertainty [22]. A potential research direction is developing a mapping method from project management involving generalized precedence relations to fuzzy GSM schemas. In this context, we intend to explore the use of fuzzy GSM schemas for enabling dynamic multi-criteria decision making.

Also, we plan to extend the feature composition approach defined for classical GSM schemas [8] to deal with fuzzy GSM schemas. Finally, another direction for future work is to leverage IoT-enabled monitoring of artifact-centric processes [19] with fuzzy GSM schemas, in order to allow fine-grained monitoring constraints that can be expressed in linguistic terms.

CRediT authorship contribution statement

Rik Eshuis: Conceptualization, Validation, Writing - original draft, Writing - review & editing. Murat Firat: Conceptualization, Validation, Writing - original draft, Writing - review & editing. Uzay Kaymak: Conceptualization, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

We are thankful to Viggo Eindhoven Airport B.V. for cooperating with us in modeling their operations management case.

References

[1] O. Adam, O. Thomas, G. Martin, G. Fuzzy workflows - enhancing workflow management with vagueness. In Proc. EURO/INFORMS 2003, 2003.
[2] A. Azadet, M. Haghnejati, Y. Khodadadegan, M. & Madadi, Modeling and improvement of the integrated business and production processes by fuzzy simulation. In Proc. SpringSim 2009 (pp. 42:1–42:8). SCS/ACM, 2009.
[3] E. Bazhenova, S. Haarmann, S. Ihde, A., Solti, M. & Weske, Discovery of fuzzy DMN decision models from event logs. In Proc. CAiSE 2017 (pp. 629–647). Springer volume 10253 of Lecture Notes in Computer Science, 2017.
[4] E. Benasconi, F. Filippi, B. Lazzerini, B. Niccolini, C. Petronella, An integrated approach based on business process modeling and fuzzy logic for risk identification and evaluation in production processes, Intell. Decision Technol., 7 (2013) 113–122.
[5] M. Comuzzi, Optimal directed hypergraph traversal with ant-colony optimisation, Inf. Sci. 471 (2019) 132–148.
[6] E. Damaggio, R. Hull, R. Vaculín, On the equivalence of incremental and fixpoint semantics for business artifacts with guard-stage-milestone lifecycles, Inform. Syst. 38 (2013) 561–584.
