Aberystwyth University

Towards Fuzzy Compositional Modelling
Shen, Qiang; Zhao, Ruiqing; Fu, Xin

Towards Fuzzy Compositional Modelling

Xin Fu, Qiang Shen and Ruiqing Zhao

Abstract— Compositional Modelling (CM) has been applied to synthesize automatically plausible scenarios in many problem domains with promising results. However, it is assumed that the generic and reusable model fragments within the knowledge base can all be expressed by precise and crisp information. This paper presents an initial attempt to extend the existing CM work to allow the generation of scenario spaces which are capable of representing, storing and supporting inference about imprecise or ill-defined data, by the use of fuzzy sets. A knowledge representation formalism for both fuzzy parameters and fuzzy constraints is incorporated into the representation of conventional model fragments. The applicability of the proposed method is illustrated by means of a simple worked example for supporting crime investigation.

I. INTRODUCTION

Compositional Modelling (CM) [3] [7] has been employed to synthesize and store plausible scenario spaces effectively and efficiently in many problem domains (e.g. physical [6] and criminological [8]). One of the most significant advantages of using CM is its ability to construct automatically many variations of an underlying scenario from a relatively small knowledge base. This is rooted in the observation that the constituent parts of different scenarios are not normally unique to any one specific scenario, there are potentially many scenarios that possess common or similar properties locally or globally. The scenario elements and their relationships can therefore be modelled as generic and reusable fragments and they only need to be recorded once in the knowledge base.

While existing work has the capability of automatically generating plausible scenarios from available evidence, it is assumed that the model fragments within the knowledge base can all be expressed by precise and crisp information. However, for applications like crime detection and prevention, the degree of precision of available evidence and intelligent data can vary greatly. In particular, different people may hold different conceptual models of the world.

Furthermore, in the existing work, each scenario fragment employs a set of probability distributions to represent the likelihood of its associated outcomes, and these are described in numerical forms. However, such assessment of likelihood typically reflects the expertise and knowledge of experienced investigators and is normally available in linguistic terms instead [4]. The use of seemingly accurate numeric probabilities suffers from an inadequate degree of precision.

Fuzzy set theory offers a useful means of capturing and reasoning with uncertain information at varying degrees of precision. Although fuzzy set has been applied to addressing

various problems, it has not been integrated with the CM techniques. This paper presents an initial attempt to extend the existing CM work to allow for representing and use of vague knowledge and linguistic probability [2], [5]. It follows the existing literature in applying CM to support crime investigation, which is well suited to illustrating the underlying ideas of integrating fuzzy set in CM, as the scenario fragments are highly subjective and often related to inexact and vague information.

The development of fuzzy CM mechanism involves two conceptually distinct aspects: (i) fuzzification of parameters in the model fragments, including the identification and definition of fuzzy variables in a generic sense; and (ii) fuzzy probabilistic assessment of the constraints between the states and events.

After presenting a brief overview of the basic concepts of CM in Section II, the knowledge presentation of both fuzzy parameters and fuzzy constraints in defining fuzzfied scenario fragments is given in Section III. This is followed by an illustration of applying fuzzy model fragments to a small crime investigation problem in Section IV, showing the composition process of a plausible scenario space from given evidence and facts. Section V concludes this paper and points out future work.

II. BASIC CM CONCEPTS

In CM, the knowledge base of the model-building system consists of a number of generic scenario fragments, interchangeably termed model fragments as above, which represent generic relationships between domain objects and their states for certain types of partial scenario. A scenario fragment has two parts that encode domain knowledge: 1) the relations between domain elements which are often represented in a form that is similar to conventional production rules but in a much more general format where predicates are used to describe the properties of these domain elements; and 2) a set of probability distributions that represent how likely it is that the corresponding relationships are related.

More formally, a scenario fragment μ is a tuple $\langle v^s, v^t, \phi^s, \phi^t, A \rangle$ and is represented as below:

> **If** $\{\phi^s\}$
> **Assuming** $\{A\}$
> **Then** $\{\phi^t\}$
> **Distribution** ϕ^t
> $\{v_1^s \dots v_n^s \rightarrow v_1^t : q_1 \cdots v_m^t : q_m\}$ where

- v^s is a set of variables named source-participants, referring to already identified objects of interest in the partial scenario, which can be real, artificial or conceptual objects.

Xin Fu (email: xxf06@aber.ac.uk), Qiang Shen (email: qqs@aber.ac.uk), and Ruiqing Zhao (email: rzz@aber.ac.uk), are with the Department of Computer Science, the University of Wales, Aberystwyth, UK.

- v^t is a set of variables named target-participants, representing new objects that will be added to the partial scenario description if the model fragment is instantiated (i.e. when both the conditions and assumptions are presumed to be true).
- ϕ^s is a set of relations called structural conditions, whose free variables are elements of v^s. Normally, the structural conditions appear in the antecedent part and describe how the source-participants are related to one another, often encoded in the form of predicates.
- ϕ^t is a set of relations called post-conditions, whose free variables are elements of $v^s \bigcup v^t$. Normally, ϕ^t appears in the consequent part and define new relations between source-participants and/or target-participants, also often encoded in the form of predicates.
- A is a set of assumptions, referring to those pieces of information which are unknown or cannot be inferred from other scenario fragments, but they may be presumed to be true for the sake of performing hypothetical reasoning.

The **If** statement describes the required conditions for a partial scenario to become applicable. These conditions must be factually true or logical consequences of other instantiated fragments. The **Assuming** statement indicates the reasoning environment. With the purpose of performing hypothetical reasoning, this environment specifies the uncertain events and states which are presumed in a partial scenario description. The **Then** statement describes the consequent when the conditions and presumed assumptions hold. They may represent a piece of new knowledge or relations which are derived from the hypothetical reasoning. The **Distribution** statement indicates the probability distributions of the consequent variables or those of their relations. The left hand side of the "implication" sign in each instance of such a statement is a combination of variable-value pairs, involving antecedent and assumption variables, and the right hand side indicates the likelihood of each alternative outcome if the fragment is instantiated.

For example, the following scenario fragment shows a piece of generic forensic knowledge that, assuming that suspect S overpowers victim V, there is a 75% chance that fibres will be transferred from S to V:

If {`suspect(S)`, `victim(V)`}

Assuming {`overpowers`(S, V)}

Then {`transfer(fibers,`$S, V)$}

Distribution `transfer(fibers,`$S, V)$ {

    `true, true, true` $\rightarrow$

    `true: 75%, false: 25%` }

Given a collection of such local model fragments and some observations, CM applies an inference procedure to create a space of scenario descriptions at a global level. As the details of this procedure are very similar to what is to be employed in fuzzy CM to be reported later, they are omitted here. Interested readers can refer to [8] for further details.

III. Foundations of Fuzzy CM

This section focuses on the creation of a structured knowledge representation scheme which is capable of storing and managing vague or ill-defined data. The research developed here is loosely based on adapting the crisp knowledge representation given in [8] and its related work.

A. Fuzzy parameters

For many problems, there may be many variables that share similar properties while most of these properties only involve minor variations from one another if encoded computationally, in terms of knowledge representation. This is independent of whether the variables are fuzzy or not. For example, variables such as quantity, volume and proportion all reflect the concept of capacity. This group of variables may all be expressed by linguistic terms such as large, average or small (which can be conveniently represented by fuzzy sets). Therefore, when defining a fuzzy variable, rather than redefining a new quantity space for it completely from scratch each time, it has a natural appeal to group fuzzy variables which share something in common into the same class. In each class, the common features shared by the variables are extracted and represented by an abstract variable with its quantity space specified over a normalized universe of discourse. The quantity space of a variable belonging to a given class is created by inheriting the common features from the abstract variable and by embellishing it with new or modified properties.

To enable this development, fuzzy taxonomies that describe vague states and events are introduced here. A taxonomy is considered to be a hierarchy, where those variables at a lower level are more specific than their ancestors and represent a more specialized group of fuzzy variables. In so doing, fuzzy variables in a CM knowledge base are organized in a structured manner. This does not only improve the efficiency of storing knowledge via reusing abstract fuzzy variables, but also helps reveal both the commonality and speciality of different variables. More importantly, the use of fuzzy taxonomies supports the construction of scenario spaces in a systematic and concise manner due to the inheritance property of the hierarchies.

As shown in Fig. 1, the first taxonomy organises a set of fuzzy variables relating to an abstract fuzzy variable named *Measurement*. Hence, fuzzy variables height, distance, width, depth and length share certain properties in defining their quantity spaces as they inherit such common features from *Measurement*; all of them can be measured with respect to a certain measurable unit and can be described as long, average or short. Similarly, the variables in the second taxonomy are all used to describe levels of different concepts. Although they may denote rather distinct or even seemingly irrelevant properties (e.g. temperature and difficulty), they all take on values from the same underlying abstract quantity space in terms of various levels such as high, average or low.

Note that, in these taxonomies, even the fuzzy variables which are classified into different classes may still have

some more generic common features shared between them. For instance, temperature in the second taxonomy is also a measurable variable. Hence, from a more generic aspect, they may still be allocated to a superclass which is more abstract. In order to maintain the clarity of representation and the comprehensibility of inference drawn from such representations, fuzzy taxonomies are not built in the most generic way possible, but are classified with easy interpretability in mind.
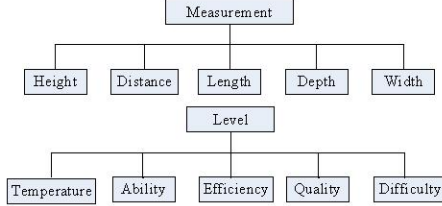

Fig. 1. Example taxonomies of fuzzy variables

From above, it is clear that in defining scenario fragments fuzzy variables can be divided into two types: abstract or non-abstract. Abstract fuzzy variables are actually variable classes that cannot be instantiated themselves in an effort to describe any actual scenario and non-abstract fuzzy variables are those that can be instantiated. Clearly, in Fig.1 *Measurement* and *Level* are abstract fuzzy variables, and depth, distance, efficiency, etc. are non-abstract variables.

In implementation, abstract fuzzy variables are indicated by means of the keyword **abstract**. Defining such a variable involves specifying the following fields:

- **Name:** A constant that uniquely identifies the abstract fuzzy variable.
- **Universe of discourse:** The domain of the abstract variable. Default definition is $[0, 1]$. Any descendant of an abstract fuzzy variable can modify the universe of discourse according to their physical dimension.
- **Cardinality of partition:** The number of fuzzy sets which jointly partition the universe of discourse. It is represented by a symbol n which will be substituted by a positive integer in a lower level non-abstract variable.
- **Quantity Space:** The membership functions of each fuzzy set that jointly cover the partitioned domain.

For example, the aforementioned abstract fuzzy variable *Level* can be defined as follows (adhering to the conventional representation style of model fragments):

Define *abstract* fuzzyvariable {
   Name: Level
   Universe of discourse: $[0, 1]$
   Cardinality of partition: n
   Quantity space:

$$fs_1 = \left[0, 0, \frac{1}{n-1} \right]$$

$$\cdots$$

$$fs_i = \left[\frac{i-2}{n-1}, \frac{i-1}{n-1}, \frac{i}{n-1} \right]$$

$$\cdots$$

$$fs_n = \left[\frac{n-2}{n-1}, 1, 1 \right]$$

}

It would be inefficient and practically unnecessary to store and manipulate fuzzy sets with arbitrarily complex membership functions. Only the triangular membership functions are considered in this initial work. Thus, a quantity space specification consists of an ordered list of triples comprising the start, top and end points of each membership function. For both computational and presentational simplicity, triangular membership functions in which the edge of a fuzzy set's membership function is exactly intersected to the centroid of the neighboring one are used in this paper. For example, assume $n = 5$, then the defined quantity space of *Level* is shown in Fig. 2.
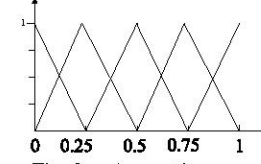

Fig. 2. A quantity space

Non-abstract fuzzy variables are identified by means of the absence of the keyword **abstract**. Such definition involves "is-a" relationships in which a non-abstract fuzzy variable is said to inherit from an abstract fuzzy variable. It requires addition of fields that are specific to the variable under definition, with shared commonalities already defined in the corresponding superior abstract fuzzy variable. In fuzzy CM, such new fields are defined as follows:

- **Is-a:** The name of an abstract fuzzy variable which refers to the immediate parent of the current fuzzy variable in a given taxonomy.
- **Scalar:** A constant which is used to scale up or down the normalized universe of discourse of the corresponding abstract variable.
- **Unit:** The variable's physical dimension. If a fuzzy variable has no unit, a default value of none is set.
- **Name of fuzzy sets:** The name of each fuzzy set in the defined quantity space.
- **Unifiability:** The declaration of a unifiable property of the variable, specified by a predicate.

The following example defines a non-abstract fuzzy variable named *Chance* that inherits from *Level*.

Define fuzzyvariable {
   Name: Chance
   Is-a: Level
   Cardinality of partition: 5
   Scalar: 1
   Unit: none
   Name of fuzzy sets: {extremely_unlikely, slim_chance, likely, very_likely, good_chance}
   Unifiability: Chance(X)
}

Obviously, this non-abstract fuzzy variable *Chance* is a kind of *Level*. Due to property inheritance, its universe of discourse equals to the normalized universe of discourse multiplied by the scalar over the corresponding physical dimension. Its quantity space is evenly partitioned by five fuzzy sets which are described respectively by the five linguistic terms given. Also, the membership functions of those fuzzy sets are inheritably obtained once again by multiplying the corresponding key points in each fuzzy set by the scalar.

B. Fuzzy constraints

In CM, knowledge is normally expressed as constraints or relations which must be obeyed by certain variables involved

in a given problem domain. For instance, velocity and duration relations often appear in physical reasoning systems; population growth and competition relations often appear in ecological reasoning system; length and angle relations often appear in spatial reasoning systems. Such constraints as used in the existing work require numerical values to quantify the probability of a consequence's occurrence (illustrated in Section II).

Since such subjective probability assessments are often the product of barely articulate intuitions, the seemingly numerically precise expressions may cause loss of efficiency, accuracy and transparency [2], [5], [4]. Under many circumstances, an expert may be unwilling or simply unable to suggest a numerical probability. The initial work developed here models the vagueness of the probability distribution in terms of subjective linguistic probabilities. Rather than using numerical representation as in the literature, a fuzzy variable called *Chance* which inherits the properties of the abstract fuzzy variable *Level* (as defined in Section III-A) is introduced to capture subjective probabilistic assessments.

Similar to the existing approach, a scenario fragment includes a set of probability distributions over the possible assignments of the consequent ϕ^t, for those interested combinations of assignments to the variables within the structural conditions and assumptions. This can be generally represented by:

$$P(a_1 : v_1, \ldots, a_m : v_m \rightarrow c : v_{cp}) = fs_p \quad (1)$$

where $a_i : v_i$, $i \epsilon \{1, 2, \cdots, m\}$ denotes the assignment obtained by assigning v_i to variable a_i, $c : v_{cp}$ has a similar interpretation, and fs_p is a member of the quantity space that specifies the fuzzy variable *Chance*.

The following sample fragment illustrates the concepts and applicability of fuzzy constraints:

> **If** {height(S), height(V)}
>
> **Assuming** {attempted_to_kill(S,V)}
>
> **Then** {difficult_level(overpower(S,V))}
>
> **Distribution** difficult_level(overpower(S,V)) {
>
>   tall, short, true →
>
>   easy: good_chance, difficult: slim_chance }

It describes a casual relation holding among structural condition a_1 and a_2, assumption a_3 and post-condition c. Here, $a_1 = height(S)$ indicates the height of a suspect S, which is a fuzzy variable that takes values from a predefined quantity space of {very_short, short, average, tall, very_tall}; $a_2 = height(V)$ indicates the height of a victim V, whose possible value assignment is the same as S; $a_3 = attempted\_to\_kill(S, V)$ describes that S attempted to kill V, representing a conventional boolean predicate; and $c = difficulty\_level(overpower(S, V))$ describes the difficulty level for S to overpower V, with possible assignments being easy, average and difficult.

Note that, when defining probability distributions in scenario fragments, the names of those variables within the structural conditions, assumptions and post-conditions (e.g. a_1, a_2, a_3 and c) are omitted when such omissions do not affect the interpretation of the meaning of the associated

values, for the sake of presentational simplicity. Thus, the probability distributions can be rewritten as follows:

$$v_1, v_2, \cdots, v_m \rightarrow v_{c1} : fs_1, \cdots, v_{cp} : fs_p$$

The above fragment reveals a general relation between the heights of two people involved in a fight and the difficulty level for one to overpower the other, and it can be applied to modelling various scenarios. For example, this fragment covers a fuzzy production rule which indicates that if suspect S is tall, while victim V is short, and S indeed attempted to kill V, then S stands a good chance of overpowering V easily. Conversely, if S is shorter than V and he indeed attempted to kill V, then there is only a slim chance for S to overpower V easily.

IV. APPLICATION TO CRIME INVESTIGATION: OUTLINE OF SCENARIO COMPOSITION

This section covers how the proposed knowledge representation formulism is used to support CM along with a sample application. Relevant evidence and the key scenario fragments are presented in Appendix A. From the given facts, collected evidence and this knowledge base, a structural scenario space can be generated by joint use of two conventional inference techniques named *abduction* and *deduction*. Note that since the degree of precision of the information (including both predefined knowledge and available evidence/facts) can vary greatly, the collected evidence and the knowledge base cannot in general be matched precisely. Thus, a fuzzy matching method is applied for scenario fragment instantiation.

A. Initialization

Collected evidence and facts are firstly entered to the emerging scenario space. The present example shows a piece of evidence in which a number of fibers collected from Dave's body have been identified as matching the fibers of Bob's clothes, and two available facts in which Dave is known to be the victim and Bob is under suspicion.
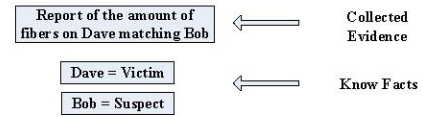

Fig. 3. Result of initialization

B. Backward chaining phase

This involves the *abduction* of all domain objects and their states which lead to the available evidence. These plausible causes are created by instantiating the conditions and assumptions of the scenario fragments in the knowledge base, whose consequences match the collected evidence. After that, the newly created instances of all plausible causes are recursively used in the same manner as the original piece of evidence, instantiating all relevant fragments and adding new nodes that correspond to the instantiated conditions and assumptions to the emerging scenario space. This phase leads to what is shown in Fig. 4.

A brief explanation of how such *abduction* phase works with respect to the following sample fragment and collected evidence/facts is given below:

Since the collected evidence matches the consequent variable of the above scenario fragment, the variables within the structural conditions and assumptions X, S and V are firstly instantiated with fibers, Bob and Dave, respectively. The resulting instantiated nodes (e.g. Transfer fibers from Bob to Dave, Degree of flight between Bob and Dave and Find fibers on Dave matching Bob) are then added to the emerging scenario space.
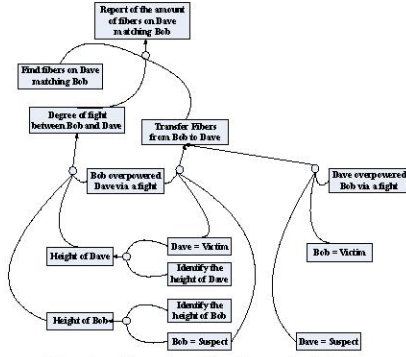

Fig. 4. Result of backward chaining

1) Fuzzy matching: To allow instantiation of a fuzzy scenario fragment when given a piece of evidence, the extended compositional modeller requires matching specific data items with broader and relatively subjective information in the knowledge base. As aforementioned, the evidence and the knowledge base cannot always be matched precisely. Under many circumstances, the values of the involved fuzzy variables do not have to be identical, partial matching suffices. Such matching is done by the following process.

First, find those scenario fragments that involve the same variables as the underlying fuzzy variables that describe the collected evidence. For example, the consequence and collected evidence in the above example both contain the amount of the transferred substance X (with the amount being a fuzzy variable). Second, identify the degree of the match between the evidence and the found scenario fragments. Third, return a matched scenario fragment for instantiation if the match degree is larger than a predefined threshold, otherwise, no match between them is found.

A match degree is obtained by calculating the maximum membership value over the overlapping area between the relevant fuzzy sets. Note that more complex calculi for matching degree may be developed; however, for computational simplicity and thanks to the employment of triangular fuzzy sets only, this straightforward matching method is adopted here. Clearly, much remains to be done in order to have a more general approach regarding the set-up of the important threshold used in the third step. Yet, this does not affect the understanding of the underlying inference techniques introduced herein.

C. Forward chaining phase

While all plausible causes of the collected evidence and some pieces of additional evidence may be introduced to the emerging scenario space during the backward chaining phase, the forward chaining phase is responsible for extending the scenario space by adding all plausible consequences of the fragments whose conditions and assumptions match the instances created in the last phase. This produces potential pieces of evidence that have not yet been identified, but may be used to improve the plausible scenario description.

This procedure applies logical *deduction* to all the scenario fragments in the knowledge base, whose conditions and assumptions match the existing nodes in the emerging scenario space. The actual matching method used is basically the same as that used previously (except step 1 obviously). For the running example, based on those newly introduced nodes, their deduced corresponding consequences are then created and added to the emerging scenario space. Fig. 5 depicts the resulting scenario space that may be the outcome of this phase (depending on the actual knowledge base used).
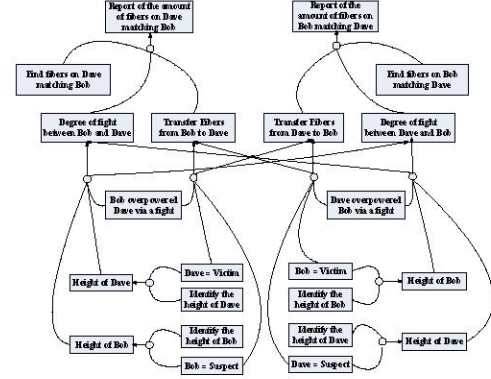

Fig. 5. Result of forward chaining

D. Removal of spurious nodes

In the backward chaining phase, some spurious nodes may have been added to the emerging scenario space. Such nodes are root nodes in the space graph which are neither facts or instantiated assumptions nor the justifying nodes that support the instantiated assumptions. The removal procedure recursively examines the root nodes in the emerging scenario space and removes those nodes that do not correspond to a fact/assumption and the justifications they occur in. It terminates when each root node in the emerging scenario space corresponds to either a fact or an assumption, guaranteeing all the spurious nodes are removed. In this example, the emerging scenario space contains the following information that Dave is both the suspect and victim at the same time, and the same for Bob. As Dave is known to be the victim whereas Bob is known as the suspect, hence, the nodes "Dave = suspect" and "Bob = Victim" as well as their directly supported nodes can be removed from this emerging space. The remaining scenario space is shown in Fig. 6.

E. Use of generated scenario space

Once the plausible scenario space is generated, it provides effective assistance for crime investigators by allowing them
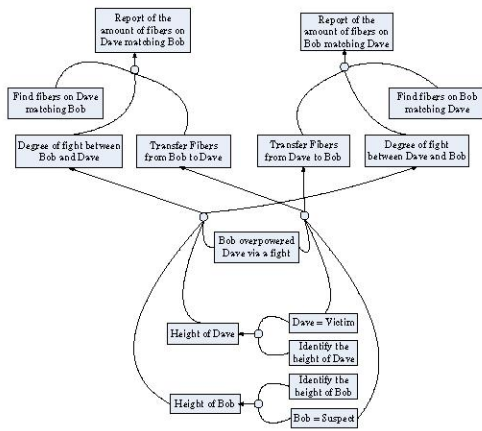
Fig. 6. Result of spurious node removal

to seek potential answers to a range of possible queries. For instance, an investigator may query the system for scenarios by inputing his/her interested evidence or hypotheses. Also, the investigator might discover that a tall person was observed entering the crime scene on a CCTV camera, and wonders whether this would rule out homicidal death. The system can answer this type of question by adding this new evidence to the set of collected pieces of evidence and modifying the generated scenario description to establish whether the new evidence indeed supports the hypothesis. Note that compared with previous work, the present approach provides more flexible query support, as it has the capability to deal with fuzzy queries.

V. Conclusions

This paper has extended the existing CM work to allow for representing, storing and supporting inference about vague and imprecise data, by the use of fuzzy sets. A knowledge representation formalism for both fuzzy parameters and fuzzy constraints is incorporated into the representation of conventional model fragments. The applicability of the proposed method is illustrated by means of a simple worked example for providing crime investigation support.

However, the ideas presented here require considerable further development. In particular, the proposed method is not yet able to provide evidence collection strategies for decision support. If the generated plausible scenarios can be evaluated by means of calculating the most likely scenario, the effectiveness of evidence collection may be greatly improved. Also, the fuzzy constraints within a single scenario fragment are defined by employing a fuzzy variable named *Chance*. However, when dynamically composing these potential relevant scenario fragments into plausible scenario descriptions, the fuzzy constraints will be propagated from individual fragments to their related ones. How to combine and propagate fuzzy probabilities in an emerging model space is a tough problem that needs to be taken into account in further research. Original work as represented in [5], [4] may serve as a starting point for this.

Appendix

Key Sample Data and Scenario Fragments

```
Define action{
```

```
name = find_match
description = find the substance X on V matching S
unifiability = find_match(X,V,S) }

Define action{
name = identify_height
description = identify the height of P
unifiability = identify(height(P)) }

Define evidence{
name = report_of_amount
description = report of the amount of X
unifiability = evidence(amount(X)) }

Define fuzzyvariable {
name = height
is-a = measurement
cardinality of partition = 5
scalar = 250
unit = centimeter
names of fuzzy sets = {very_short, short, average, tall, very_tall}
unifiability = height(P) }

Define fuzzyvariable {
name = amount
is-a = capacity
cardinality of partition = 5
scalar = 1
unit = none
names of fuzzy sets = {none, few, several, a_number_of, many}
unifiability = amount(X) }
```

If {suspect(S),victim(V) }
Assuming {overpower(S,V) }
Then { transfer(X,S,V) }
Distribution transfer(X,S,V) {
true,true,true→true:good_chance, false:slim_chance}

If {suspect(S),victim(V) }
Assuming {overpower(S,V) }
Then { transfer(X,V,S) }
Distribution transfer(X,V,S) {
true,true,true→true:good_chance, false:slim_chance}

If {person(P) }
Assuming {Identify(height(P)) }
Then { height(P) }
Distribution height(P) {
true,true→true:1, false:0}

If {degree_of_fight(S,V) }
Assuming {transfer(X,S,V),find_match(X,V,S) }
Then { evidence(amount(transferred(X,V,S))) }
Distribution evidence(amount(transferred(X,V,S)))
{intensive,true,true→many:good_chance,few:slim_chance
weak,true,true→many:slim_chance,few:good_chance}

If {height(V), height(S) }
Assuming {overpower(S,V) }
Then {degree_of_fight(S,V) }
Distribution degree_of_fight(S,V)
{tall,short,true→intensive:slim_chance,weak:good_chance
short,tall,true→intensive:slim_chance,weak:good_chance
tall,tall,true→intensive:good_chance,weak:slim_chance
short,short,true→intensive:good_chance,weak:slim_chance}

If {height(V), height(S) }
Assuming {overpower(S,V) }
Then {degree_of_fight(V,S) }
Distribution degree_of_fight(V,S)
{tall,short,true→intensive:slim_chance,weak:good_chance
short,tall,true→intensive:slim_chance,weak:good_chance
tall,tall,true→intensive:good_chance,weak:slim_chance
short,short,true→intensive:good_chance,weak:slim_chance}

References

[1] G.Bordogna and G.Pasi. "Modeling vagueness in information retrieval," *ESSIR*, 207–241, 2000.

[2] G.de Cooman. "A behavioural model for vague probability assessments," *Fuzzy Sets and Systems*, 154(3):305–358, 2005.

[3] B.Falkenhainer and K.Forbus. "Compositional modelling: Finding the right model for the job," *Artificial Intelligence*, 51:95–143, 1991.

[4] J.Halliwell, J.Keppens and Q.Shen. "Lingustic bayesian networks for reasoning with subjective probabilities in forensic statistics," In *Proceedings of the 9th International Conference on Artificial Intelligence and Law*, 42-50, 2003.

[5] J.Halliwell and Q.Shen. "Linguistic Probabilities: Theory and Application," To appear in *Soft Computing*.

[6] W.Hamscher, L.Console, and J.de Kleer (eds.). "Readings in Model-Based Diagnosis," Morgan-Kaufmann, San Mateo, 1992.

[7] J.Keppens and Q.Shen. "On compositional modelling," *Knowledge Engineering Reivew*, 16(2):157–200, 2001.

[8] Q.Shen, J.Keppens, C.Aitek, B.Schafer and M.Lee. "A Scenario Driven Decision Support System for Serious Crime Investigation," To appear in *Law, Probability and Risk*, 2007.

[9] Y.Wang. "On fuzzy multiattribute decision-making models and methods with incomplete information," *Fuzzy Sets and Systems*, 151:285-301, 2005.